Proc. Of the 8th International E-Conference on Advances in Engineering, Technology and Management - ICETM 2022

ISBN: 978-1-63248-195-5 DOI: 10.15224/ 978-1-63248-195-5-06

Design and Comparison of Integer and Fractional PI Controllers for Field Oriented Control of PMSM

Omkar Bhanap, Meera Murali, Shayok Mukhopadhyay

Abstract— **Permanent magnet synchronous motor is a commonly used motor in electric vehicles. This paper focuses on the design and comparative analysis of integer and fractional order proportional integral controllers on Permanent Magnet Synchronous Motor (PMSM) used for electric vehicle application. Field oriented control (FOC) is used for control of PMSM in EV application. Dynamic equations in rotor reference frame are used to obtain the state space representation of the PMSM. Perturb and disturb method is used to convert this state space representation into transfer function. Frequency response-based approach is used to design integer order PI (IOPI) and fractional order PI (FOPI) controllers for current as well as speed loops. Four different combinations of inner and outer loops are then considered for comparative analysis of time and frequency domain response. MATLAB software is used to validate the proposed PMSM drive system. It is observed that, the FOPI controller-based system is more effective than the conventional IOPI controller-based system due to its non-linear nature.**

Keywords—PMSM, field-oriented control, fractional order control, frequency response-based design.

I. Introduction

High efficiency, quick power output and zero tailpipe emission are advantages of electric vehicles over conventional gasoline vehicles[1]. An electric vehicle broadly consists of a battery pack, traction motor, e-powertrain, and vehicle controller. The battery pack consists of lithium-ion cells connected in a series-parallel combination to achieve the required current and voltage ratings. Various researchers have done a detailed study on the type and construction of motors suitable for the electric vehicle (EV) application. These studies show that induction motors and permanent magnet alternating current motors (BLDC, SRM and PMSM) are best suited for electric vehicle application [2][4]. Among these two, PMSM motors offer many advantages over induction motors, such as high torque to weight ratio, high efficiency, and higher power density and hence, have captured the place of induction motors in the electric vehicle industry.

The PMSM motor can be controlled by direct torque control (DTC) and field-oriented control (FOC). Both the methods have certain advantages and shortcomings over one another. A detailed comparison of these speed control methods is given in [3]. This study shows that FOC provides lower ripple and distortion in motor torque over DTC and is commonly used in electric vehicles. Three PI controllers are required in the FOC of PMSM, one for speed loop and the two for direct and quadrature axis current control. From the perspective of control engineering, the PMSM motor is highly non-linear since its torque equation contains a term that has the product of direct and quadrature axis currents (both are the state variables for the torque equation). Hence, by using non-linear FO-PI controllers instead of IO-PI controllers, the performance of the PMSM drive system can be enhanced.

A PMSM is a singly electrically excited motor where the rotor has permanent magnets placed on its surface (called surface PMSM) or embedded inside (interior PMSM). The stator has three phase windings. The mathematical modelling and speed control methods of PMSM are given in [5]. A three-phase inverter is required for driving the motor. The inverter can be triggered by either sine wave pulse width modulation (SPWM) or space vector pulse width modulation (SVPWM) technique. Here, the sine wave PWM is easy to implement but SVPWM can provide 15% better utilization of the supply voltage and hence is considered in this work[6]. The simulation of IOPI controller based FOC, field weakening and maximum torque per ampere (MTPA) of PMSM is presented in [7]. The design of fractional order controllers for an interior PMSM is presented in the paper [8][9]. In this, the authors have used perturb and disturb method to obtain the transfer function for the interior PMSM plant model. The MATLAB control system toolbox is not useful for the fractional-order PI controller simulation. Hence the authors in [10] and [11] have created 'Ninteger' and 'FOMCON' toolboxes for the simulation of FOPI controller-based systems.

This paper's contribution lies in the frequency response-based design of IOPI and FOPI controllers for vector control of PMSM and comparative analysis of the different combinations IOPI and FOPI controllers when used for inner and outer loops of the PMSM FOC system.

The remainder of the paper is organized as follows; section II contains a brief description of fractional order calculus and fractional-order controllers. Sections III and IV deal with the mathematical modelling of FOC PMSM and design of IOPI and FOPI in the frequency domain. The last two sections cover MATLAB results and discussion, followed by the conclusion.

Omkar Bhanap

Department of Electrical Engineering, College of Engineering Pune
India

Meera Murali

Department of Electrical Engineering, College of Engineering Pune
India

Shayok Mukhopadhyay

Department of Electrical Engineering, College of Engineering,

American University of Sharjah
United Arab Emirates (UAE)

II. A Brief Description of Fractional Order Calculus and Fractional Order Controllers

Fractional calculus is a branch of mathematics which deals with non-integer order of differentiation and integration. It corresponds to generalization of conventionally used differentiation and integration to the non-integer orders. The properties of integral calculus are not applicable to fractional calculus. The various approaches for representation of fractional order differentiation are given as follows:

A. *Grunwald-Letnikov Definition*

A fractional order derivative is represented according to the Grunwald-Letnikov definition as,

$$ {}_aD_t^{\alpha} = \lim_{h\to 0} \frac{1}{h^{\alpha}} \sum_{r=0}^{\left[\frac{t-a}{h}\right]} (-1)^r \binom{n}{r} f(t-rh) $$

(1)

Where, $\left[\frac{t-a}{h}\right]$ is an integer and a and t are limits of operator. The binomial expression $\binom{n}{r}$ is given by,

$$ \binom{n}{r} = \frac{\Gamma(n-1)}{\Gamma(\mathrm{r}+1)\,\Gamma(\mathrm{n}-\mathrm{r}+1)} \qquad (2) $$

Where, the gamma function (Γ) in the above expression is given by, $\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t}\, dt\,, \Re(Z) > 0$. (3)

B. *Riemann-Liouville (R-L) Definition*

As per Riemann-Liouville (R-L) definition, fractional order differentiation is defined as,

$$ {}_aD_t^{\alpha} = \frac{1}{\Gamma(n-\alpha)} \left(\frac{d}{dt}\right)^n \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}}\, d\tau \qquad (4) $$

Where, α is a real value called as order of differentiation, n is an integer with the condition $(n-1) < \alpha < n$, t and a are limits of integration. The R-L method of representing fractional derivatives is generally used to determine analytical solution of trigonometric and exponential functions [12].

C. *FOPI Controllers*

The transfer function for a FOPI controller is given by,

$$ C_{(s)} = K_p + \frac{K_i}{s^{\lambda}} \qquad (5) $$

Where λ is the order of integration, is a positive real number. Comparing with the IOPI controller, in case of FOPI, one additional tunable parameter λ is available. It provides an additional benefit of robustness to system gain variations. In case of a PMSM, the inductances, back emf and stator resistance are not always constant and are dependent on rotor position, load current etc. Thus, using a fractional order controller instead of integer order controller can improve the performance of the drive system[13].

III. Mathematical Modelling of Permanent Magnet Synchronous Motor

The generalized non-linear mathematical model of PMSM in rotor reference frame is given as

$$ \left.\begin{aligned} v_d &= (R_s + L_d\, p)i_d - \omega_r\, L_q\, i_q \\ v_q &= \left(R_s + L_q\, p\right)i_q + \omega_r\, L_d\, i_d + \omega_r\, F_x \\ T_e &= \frac{3}{2} * \frac{P}{2}\, i_q \left(F_x + \left(L_d - L_q\right) * i_d\right) \\ \frac{d}{dt}\,\omega_r &= \frac{1}{J}\,(\,T_e - T_l - B\,\omega_r\,) \end{aligned}\right\} \qquad (6) $$

Where,
$v_{d,}\, i_d, v_q, i_q$ – d and q axis stator voltages and currents
L_d, L_q – direct and quadrature axis self-inductances
R_s – Stator resistance,
p – differential operator $\frac{d}{dt}$,
ω_r – Rotor speed in (rad/s),
T_l – Load torque,
T_e – Electromagnetic torque,
J – Rotor inertia,
F_x – Stator flux linkages due to rotor PM,
B – Friction constant,
P – number of rotor poles

Once the mathematical model of PMSM is obtained, we linearize the above model using 'perturb and disturb' method around an operating point. The detailed explanation is given in [5]. For the ease of calculations, the rotor torque is taken as input. Thus, the dynamic equations in rotor reference frame are converted into state space representation as,

$$ pX = AX + BU \quad \text{and} \quad Y = CX + DU \text{ where,} $$
$$ X = \left[\delta i_q\ \delta i_d\ \delta\omega_r\right]^T \text{ and } U = \left[\delta v_q\ \delta v_d\ \delta T_l\right]^T $$

The A, B, C and D matrices mentioned in the above equation are

$$ A = \begin{bmatrix} -\frac{R_s}{L_q} & -\frac{L_d}{L_q}\,\omega_{ro} & -(F_x + L_q I_{d0}) \\ -\frac{L_q}{L_d}\,\omega_{ro} & -\frac{R_s}{L_d} & -\frac{L_q}{L_d}\, I_{d0} \\ k\left(F_x + (L_d - L_q)I_{d0}\right) & k(L_d - L_q)I_{q0} & -\frac{B}{J} \end{bmatrix} $$

$$B = \begin{bmatrix} \frac{1}{L_q} & 0 & 0 \\ 0 & \frac{1}{L_d} & 0 \\ 0 & 0 & -\frac{P}{2J} \end{bmatrix}, C = [1\ 0\ 0]\ and\ D = [0\ 0\ 0]$$

$$k = \frac{3}{2J}\left(\frac{P}{2}\right)^2$$

From the state model, transfer function can be obtained as

$$G_{i(s)} = \frac{Y_{(s)}}{U_{(s)}} = [C\ (sI - A)^{-1}\ B + D] \qquad (7)$$

Using the above C matrix (i.e. [1 0 0]), the transfer function between quadrature axis current and voltage is obtained. For direct axis current-voltage transfer function, the C matrix is modified to $C =\ [0\ 1\ 0]$ [8]. Fig. 1 shows the block diagram for field-oriented control of PMSM. For implementation the FOC control strategy on PMSM, a total of three control loops needs to be tuned, two PI controller are used to track i_d and i_q and one is required to control the rotor speed ω_r.

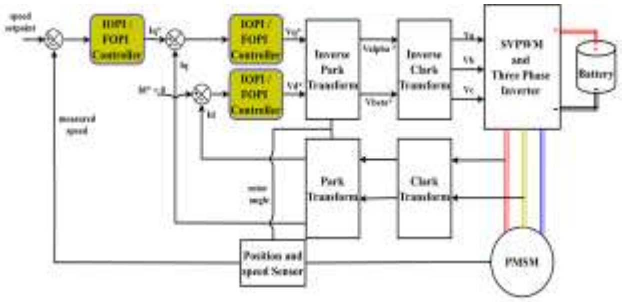

Fig. 1: Field Oriented Control of PMSM: Block diagram

A. *Current control loop*

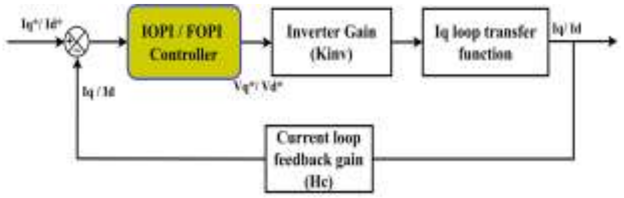

Fig. 2: Current control loop

Fig. 2 shows the current loop for FOC PMSM. The control loop consists of a PI controller, inverter gain K_{inv} and the plant transfer function for the electrical system. This control loop is used twice (i.e for tracking of i_d and i_q) and the complete i_q control loop is used as plant transfer function for the speed control loop. The gain of the inverter is obtained as

K_inv = $(0.65 * \text{V\_dc})/V\_(cm\ )$ [13].

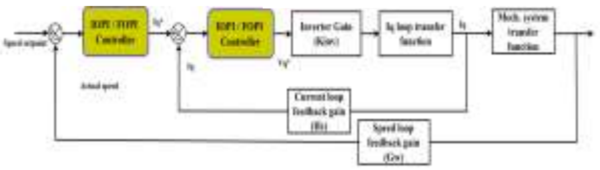
Speed setpoint
IIOPI / FOPI Controller
Iq*
IOPI / FOPI Controller
Vq*
Iq
Inverter Gain (Kinv)
Iq loop transfer function
Iq
Mech. system transfer function
Actual speed
Current loop feedback gain (Hc)
Speed loop feedback gain (Gw)

Fig. 3: Speed control loop

B. *Speed control loop*

Fig. 3 shows the speed control loop. Here, the current loop block represents the closed loop transfer function of the q-axis loop, and the mechanical system block is the part of transfer function of speed loop which takes care of mechanical time constant. It is given by $G_{mech(s)}$.

$$G_{mech(s)} = \frac{K_m\, K_t}{(1{+}s\, T_m)} \qquad (8)$$

$$K_m = \frac{1}{B}, K_t = \frac{3}{2}\left(\frac{P}{2}\right)^2 \lambda_{af}\ , T_m = \frac{J}{B}, G_w\ = \frac{H_w}{1{+}\ T_w}$$

Where, K_m- Mechanical gain
T_m - Mechanical time constant
K_t - Torque constant
T_w - Time constant of speed filter

IV. Frequency Response Based Design of Integer and Fractional Order PI Controllers

A. *Integer Order PI Controller*

From Fig. 2, the open loop transfer functions for the i_q and i_d loops are given by,

$$G_{IOL(iq)(s)} = \left(K_{p1} + \frac{K_{i1}}{s}\right) * K_{inv} *\ G_{iq(s)} \qquad (9)$$

$$G_{IOL(id)(s)} = \left(K_{p2} + \frac{K_{i2}}{s}\right) * K_{inv} *\ G_{id(s)} \qquad (10)$$

where, $G_{iq(s)}$ is obtained from (8), taking the value of C equal to [1 0 0] and $G_{id(s)}$ is obtained by taking the value of C equal to [0 1 0]. To design the integer order PI controller using the frequency response, we need to specify two parameters, viz. desired phase margin of the system (PM) and desired crossover frequency (ω_c). From these two parameters and according to the definition of phase margin, (9) in frequency domain becomes,

$$|G_{IOL(iq)(jw)}| = 1 \ \text{ at } \omega = \omega_c \qquad (11)$$

$$\arg\left(G_{IOL(iq)(j\omega c)}\right) =\ -180^o + \phi_{PM} \qquad (12)$$

Therefore, $\sqrt{K_{p1}^2 + \left(\frac{K_{i1}}{\omega_c}\right)^2} * K_{inv} * \left|G_{iq(j\omega_c)}\right| = 1$

on squaring both sides,

$$K_{p1}^2 + \left(\frac{K_{i1}}{\omega_c}\right)^2 = \frac{1}{K_{inv}^2 * \left|G_{iq(j\omega_c)}\right|^2} \qquad (13)$$

Also, from (13), we get

$$-\tan^{-1}\left(\frac{K_{i1}}{\omega_c\ *\ K_{p1}}\right) = \phi_{PM} - 180^O -\ \arg\left(G_{iq(j\omega c)}\right)$$

$$\frac{\frac{K_{i1}}{\omega_c}}{K_{p1}} = \tan\left(\arg\left(G_{iq(j\omega_c)}\right) + 180^O - \phi_{PM}\right) \qquad (14)$$

Thus, by solving (13) and (14), we get the equations for K_{p1} and K_{i1} as follows:

$$\left.\begin{aligned} K_{p1} &= \frac{\cos\left(\left(arg\left(G_{iq(j\omega_c)}\right)+180^{O}-\phi_{PM}\right)\right)}{\left|G_{iq(j\omega_c)}\right|} \\ K_{i1} &= -\frac{\sin\left(\left(arg\left(G_{iq(j\omega_c)}\right)+180^{O}-\phi_{PM}\right)\right)*\omega_c}{\left|G_{iq(j\omega_c)}\right|} \end{aligned}\right\} \quad (15)$$

The K_p and K_i gains for both d and q-axis loops are determined using the above equations. Once it is obtained, then the values of K_p and K_i for the speed loop are determined. The open loop transfer function for speed loop is given by,

$$G_{\omega OL(s)} = \left(K_{p3} + \frac{K_{i3}}{s}\right) * G_{ICL(iq)(s)} * \ G_{mech(s)} \quad (16)$$

where , $G_{ICL(iq)(s)}$ is the closed loop transfer function of the iq current loop. In this case also, we need to specify the phase margin and the crossover frequency. The procedure to calculate the K_p and K_i for the speed loop is exactly same as the current loop, the only difference here is that the closed loop transfer function of i_q loop is present instead of the electrical system open loop as in previous case.

B. *Fractional Order PI Controller*

$C_{(S)}$ and $G_{p(s)}$ are the controller and plant transfer functions respectively, where $G_{p(s)} = K_{inv}\ G_{iq(s)}$. The complete transfer function is given by $G_{OL(s)} = C_{(s)} * G_{p(s)}$.

The transfer function of FOPI controller is given by

$$C_{(s)} = K_p\left(1 + \frac{K_i}{s^{\lambda}}\right) \quad 0 < \lambda < 1 \quad (17)$$

In the frequency domain, the above transfer function is written as,

$$C_{(j\omega)} = K_p\left(1 + \frac{K_i}{(j\omega)^{\lambda}}\right) \quad (18)$$

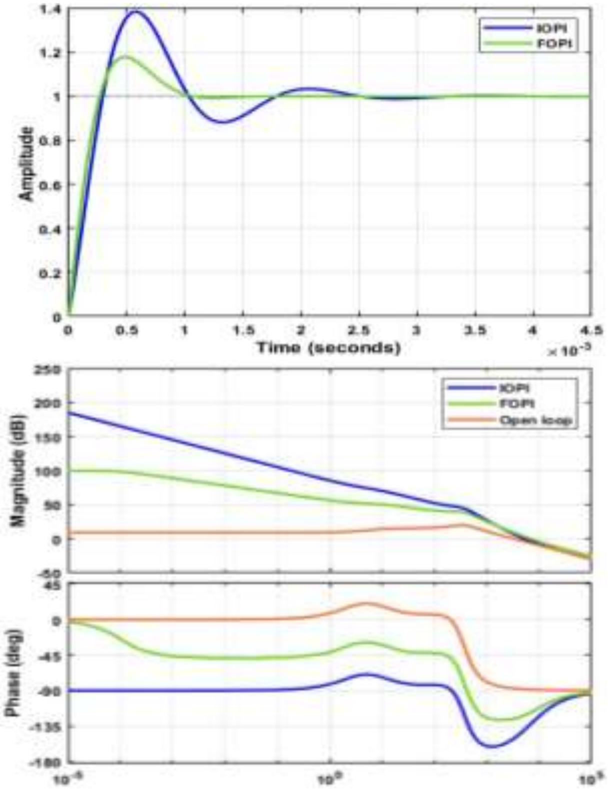

Fig. 4: Q axis current loop response

$$C_{(j\omega)} = K_p\left[\left(1 + K_i\ \omega^{-\lambda}\cos\left(\frac{\lambda\pi}{2}\right) - jK_i\omega^{-\lambda}\sin\left(\frac{\lambda\pi}{2}\right)\right)\right] \quad (19)$$

The magnitude and phase of the above transfer function are given by,

$$\left.\begin{aligned} \left|C_{(j\omega)}\right| &= K_p\sqrt{\left(1 + K_i\omega^{-\lambda}\cos\left(\frac{\lambda\pi}{2}\right)\right)^2 + \left(K_i\omega^{-\lambda}\sin\left(\frac{\lambda\pi}{2}\right)\right)^2} \\ Arg\left(C_{(j\omega)}\right) &= -\tan^{-1}\left(\frac{K_i\omega^{-\lambda}\sin\left(\frac{\lambda\pi}{2}\right)}{1+K_i\omega^{-\lambda}\cos\left(\frac{\lambda\pi}{2}\right)}\right) \end{aligned}\right\} (20)$$

For the tuning of the FOPI controller using frequency response analysis, three conditions should be satisfied as follows:

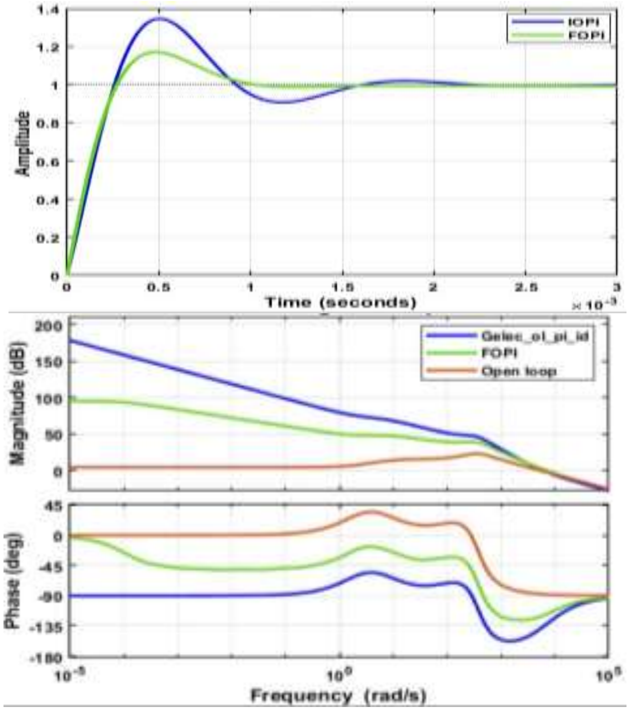

Fig. 5: D axis current loop response

1. Phase margin specification:

$$Arg\left[G_{OL(s)}\right] = Arg\left[C_{(s)}\right] + Arg\left[G_{OL(s)}\right] = -180 + \phi_{PM} \quad (21)$$

2. Robustness to gain variation of system parameters:

$$\frac{d}{d\omega}\ Arg\left[G_{OL(s)}\right] = \frac{d}{d\omega}\left(Arg\left[C_{(s)} *\ G_{OL(s)}\right]\right)|_{(\omega=\omega_c)} = \ 0 \quad (22)$$

The above two equations are implicit equations between variable K_i and fractional order λ. The solution of these equations is calculated by the graphical method(see Fig. 6). Once the value of K_i and λ is obtained, we proceed to the calculation of K_p using the definition of phase margin as follows:

3. Amplitude specification:

$$\left|G_{OL(j\omega)}\right| = \left|C_{(j\omega)} * G_{OL(j\omega)}\right||_{(\omega=\ \omega_c)} \ = 1 \quad (23)$$

The values of K_i and λ are substituted in the above equation and K_p is determined.

V. Results and Discussion

A. *Plant Description*

The parameters of the PMSM plant model are as follows[5]:

Table 1: PMSM parameters

Parameter	Symbol	Value
Pole pairs	P	6
Stator Resistance	R_s	$1.4\,\Omega$
Direct axis inductance	L_d	$0.0056\,H$
Quadrature axis inductance	L_q	$0.009\,H$
DC link supply voltage	V_{dc}	$285\,V$
Rotor inertia	J	$0.006\,kgm^2$
Viscous Friction constant	B	$0.01\,Nms/r$
Speed loop time constant	T_w	$0.002\,s$
Maximum control voltage	V_{cm}	$10\,V$
Rotor flux	Fx	$0.1546\,Wb$
Gain of speed filter	Hw	$0.05\,V/rad/s$
Rated speed	N	$3000\,rpm$
Carrier frequency	f_c	$20\,kH_z$

Parameters mentioned in table 1 are used to obtain the state space model and transfer function of PMSM. Here, from the torque equation in (6), the value of I_{q0} is taken as 6A for a steady state torque of 4.17 Nm. The open loop transfer functions for q and d-axis current loops are obtained as:

$$G_{p\ I_q(s)} = \frac{111.11\ (s + 248.2)(s + 3.462)}{(s + 7.09)(s^2 + 400.1s + 1.359e5)}$$

$$G_{p\ I_d(s)} = \frac{178.57\ (s + 155.2)(s + 2.017)}{(s + 7.09)(s^2 + 400.1s + 1.359e5)}$$

B. *Integer Order PI controller tuning*

For designing an IOPI controller using frequency response, we need to specify desired phase margin and the gain crossover frequency. The choice of phase margin and crossover frequency is made after determining the relation between time domain parameters (rise time, settling time etc.) and frequency domain parameters. Here, for simplicity, the phase margin and crossover frequency are directly taken as 45^0 and 1000Hz (6283 rad/s) [13]. Based on these design requirements and following the design procedure in section IV(A), IOPI controllers for the current loop are designed. The value of K_p and K_i are mentioned in table 2.

Table 2: IOPI controller Kp and Ki values

Control loop	K_p	K_i
I_d	0.753	5126
I_q	0.97	6406
ω	0.872	95.57

The bode plot and step response for IOPI controller tuned q and d axis current loops are depicted in Fig. 4 and 5 respectively.

Once the values of controller gains for the inner loop are obtained, then we proceed for the speed loop. The plant transfer function for the speed loop is as follows:

$$G_{p\ \omega(s)} = \frac{1.3192e06\ (s + 3839)(s + 249.2)}{(s + 247.4)(s + 1.667)(s^2 + 3957s + 1.469e07)}$$

For designing of IOPI controller for the speed loop, exactly same procedure is followed as mentioned in section IV(A). The values of controller gains obtained are mentioned in table 2. The output of speed loop for a step command of 3000 rpm is presented in Fig. 7**.**

C. *Fractional Order PI controller tuning*

For the design of fractional order controller for current and speed loops, the system should satisfy three conditions as mentioned in the section IV(B). For the sake of comparison, the values of desired phase margin and crossover frequency are considered same as of the IOPI controller-based system. Graphical method is used for determination of λ and K_i. The values of λ and K_i are then substituted in (24) to obtain the value of K_p. The outputs of the graphical method for can be seen in Fig. 6.

The values of controller gains are mentioned in table 3. For the current loops (Iq and Id), the response of system with FOPI controller and its comparison with IOPI controller is shown in Fig. 4 and 5. Fractional order transfer functions obtained for the various loops are mentioned in Table 4. Once the fractional order transfer function is obtained, Oustaloup filter is used to approximate it into the integer order transfer function. The filter settings are kept as follows: approximation order 5 and frequency range [0.0001 10000].

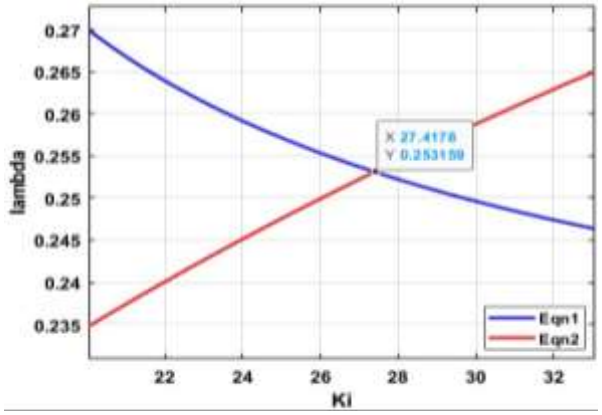

Fig. 6: Graphical method for calculation of FOPI controller gains (calculation of speed loop controller gains where; inner loop is fractional)

D. *Comparative analysis of IOPI and FOPI controllers*

There are a total of four possible combinations of using integer order and fractional order controllers in the PMSM FOC system. These are as follows: (a) Both speed and current loops are integer order (IO-IO configuration) ,(b) Speed loop fractional and current loop integer order (FO-IO configuration), (c) speed loop integer order and current loop fractional order (IO-FO) and (d) Both speed and current loops are fractional (FO-FO configuration).

The speed and current loops were tuned by all the four combinations of IOPI and FOPI controllers and their time and frequency domain performance is calculated. Table 5 gives variation in time domain parameters of the same.

Table 3: Comparative analysis of IOPI and FOPI controllers

Speed loop (ω)	Current loops (I_q and I_d)		K_p	K_i	λ
IOPI	IOPI	I_d	0.753	5126	-
		I_q	0.97	6406	-
		ω	0.872	95.57	-
IOPI	FOPI	I_d	0.117	1475	0.573
		I_q	0.126	1790	0.5465
		ω	0.886	94.57	-
FOPI	IOPI	I_d	0.753	5126	-
		I_q	0.97	6406	-
		ω	0.132	24.88	0.261
FOPI	FOPI	I_d	0.117	1475	0.573
		I_q	0.126	1790	0.5465
		ω	0.117	27.39	0.253

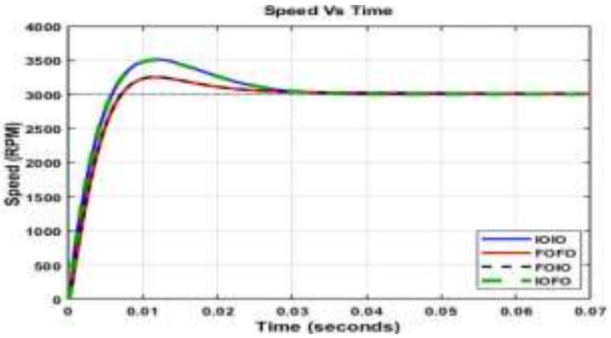

Fig. 7: Speed vs Time curve for all combination of controllers

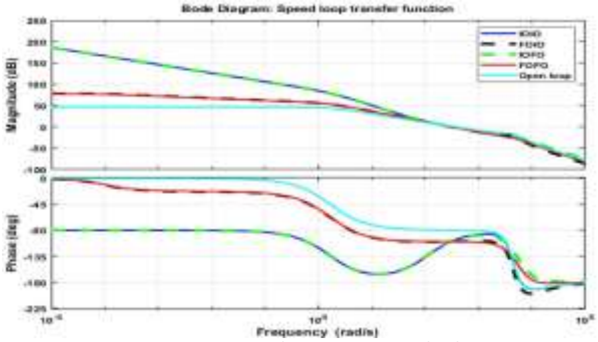

Fig. 8: Frequency response of open loop transfer function

Table 4: FO-PI controller transfer functions

Control loop	FO-PI controller transfer function
Iq	$\frac{0.126\, s^{0.546} + 225.6}{s^{0.546}}$
Id	$\frac{0.1175\, s^{0.573} + 173.5}{s^{0.573}}$
Speed (FO-FO)	$\frac{0.117\, s^{0.253} + 3.211}{s^{0.253}}$
Speed (FO-IO)	$\frac{0.1324\, s^{0.2618} + 3.29}{s^{0.2618}}$

Table5: Step Response for all combinations of IOPI and FOPI controllers

Speed loop (ω)	Current loops (I_q and I_d)		Rise Time (ms)	Settling time (ms)	% Overshoot
IO	IO	I_d	0.2	1.5	34.6
		I_q	0.229	0.88	38.2
		ω	4.3	28.0	16.87
IO	FO	I_d	0.18	0.924	17.11
		I_q	0.206	0.83	17.64
		ω	4.3	28.4	16.55
FO	IO	I_d	0.18	1.5	34.6
		I_q	0.229	0.88	38.2
		ω	3.7	24.6	8.51
FO	FO	I_d	0.205	0.924	17.11
		I_q	0.206	0.83	17.64
		ω	3.7	24.1	8.36

From table 5, it can be seen that, for inner loop, by replacing the IOPI controller with FOPI, the rise time is reduced by 10.04 %, settling time by 5.61 % and the overshoot is reduced by 55.26 %. Hence the superior performance of FOPI based system is observed. Also, for the overall system, FO-FO controller combination gives lowest rise time, settling time and overshoot compared to the other three.

From the bode plot of speed loop i.e., Fig. 8, the FO-IO and FO-FO controllers have flat nature of phase plot near the crossover frequency of 6283 rad/s. This signifies to the robustness to gain variations property of the fractional order controllers for the outer loop.

Comparing the FO-IO and FO-FO schemes, the rise time is exactly same, settling time is 2 % less and overshoot is 1.762% less in FO-FO system. Thus, implementation of FOPI controller to the inner loop does not make any significant difference to the overall system performance, but it consumes more time for the I_d and I_q calculations compared to the FO-IO scheme due to an additional factor λ.

VI. Conclusion

Fractional order PI controllers were designed in this work for all three control loops of the PMSM, and a comparative analysis of these with the conventional integer order PI controllers is done. From the simulations, it is evident that fractional order controllers perform better than integer order controllers in terms of time and frequency domain response.

If the choice has to be made on selecting a particular type of controller combination, the FO-IO configuration performs best. In case of an EV application, where a limited power ECU provides computation power, it is essential to make the I_d and I_q computations in as little time as possible. In such cases, implementing FOPI controllers for all three loops does not hold good; hence inner loops IOPI and outer loop FOPI in the best suited scheme.

References

[1] K. Poornesh, K. P. Nivya and K. Sireesha, "A Comparative study on Electric Vehicle and Internal Combustion Engine Vehicles," 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020 , pp. 1179-1183.

[2] Dr. J.R Hadji-Minaglou and Prof. Dr.-Ing G. Henneberger, "Comparison of Different Motor Types for Electric vehicle Application", EPE Journal (1999), pp. 8-34.

[3] M. Abdul, W. Begh and H. Herzog, "Comparison of field oriented and direct torque control", MAW Begh, HG Herzog - Technical University of Munich, Germany, 2018.

[4] Jape Swaraj and Thosar, Archana. (2017). Comparison of electric motors for electric vehicle application. International Journal of Research in Engineering and Technology.

[5] R. Krishnan, "Permanent magnet synchronous and Brushless DC motor drives", CRC Press, Boca Ranton, Florida, 2010.

[6] A. Apte, R. Walambe, V. Joshi, K. Rathod and J. Kolhe, "Simulation of a permanent magnet synchronous motor using Matlab-Simulink," 2014 Annual IEEE India Conference (INDICON), 2014, pp. 1-5.

[7] S. Halder, S. P. Srivastava and P. Agarwal, "Flux weakening control algorithm with MTPA control of PMSM drive," 2014 IEEE 6th India International Conference on Power Electronics (IICPE), 2014, pp. 1-5. International Conference on Smart Electronics and Communication (ICOSEC), 2020, pp. 1179-1183.

[8] Thakar, U., Joshi, V., Vyawahare, V.A., 2015. Design of fractional-order PI speed controller for permanent magnet synchronous motor. Int. Symp. Fract. Signals Syst., 36-42. ISBN 978-606-737-084-3.

[9] U. Thakar, V. Joshi, U. Mehta, V. A. Vyawahare, "Fractional-Order PI Controller for Permanent Magnet Synchronous Motor: A Design-Based Comparative Study", Chapter 18 - In Advances in Nonlinear Dynamics and Chaos (ANDC), Fractional Order Systems, Academic Press, 2018, 553-578.

[10] A. Tepljakov, E. Petlenkov, J. Belikov and J. Finajev, "Fractional-order controller design and digital implementation using FOMCON toolbox for MATLAB," 2013 IEEE Conference on Computer Aided Control System Design (CACSD), 2013, pp. 340-345, doi: 10.1109/CACSD.2013.6663486.

[11] Valério, Duarte and Costa, José. (2004). Ninteger: a non-integer control toolbox for MatLab. Proc. of the First IFAC Workshop on Fractional Differentiation and Applications.

[12] P. Shah, S. Agashe, "Review of fractional PID controller", Mechatronics, Volume 38, 2016, Pages 29-41, ISSN 0957-4158.

[13] Thakar, U., Joshi, V. & Vyawahare, V. "Design of fractional-order PI controllers and comparative analysis of these controllers with linearized, nonlinear integer-order and nonlinear fractional-order representations of PMSM". Int. J. Dynam. Control 5, pp. 187–197 (2017).