www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
Vol. 02, Issue 04, April-2022, pp: 5-14

e-ISSN : 2583-1062
Impact Factor : 2.205

TUNING OF CONTROLLERS FOR REFERENCE INPUT TRACKING OF A BLDC MOTOR

Galal Ali Hassaan*1

*1Emeritus Professor, Department of Mechanical Design and Production, Faculty of Engineering, Cairo University, Egypt.

ABSTRACT

The paper proposes fives controllers to control a BLDC motor having relatively bad dynamics. The analysis coves a PID controller from the first generation and four controllers from the second generation. The five controllers are tuned using the MATLAB optimization toolbox with functional constraints on maximum overshoot, settling time and steady state error. The performance is judged through the maximum percentage overshoot, settling time and steady state error characteristics of the time response of the control system for reference input tracking. The characteristics of the closed loop control system are compared and used to reveal the best controller suitable to control the BLDC motor.

Keywords: BLDC motor speed control, PID controller, I-PD controller, PD-PI controller, PI-PD controller, 2DOF-PID controller.

1. INTRODUCTION

Brushless direct current (BLDC) motors have the advantages of good speed control, high speed, high torque and silent operation. Therefore, they have wide industrial application in single speed applications, variable speed applications, low noise applications and applications requiring precise control [1]. The author is going to examine their automatic control using five controllers, one from the first generation of PID controllers and four from the second generation.

Kim, Yang and Kim (2007) presented a control algorithm for a BLDC motor using general purpose microcontroller having one-chip timer. They used pulse-width modulation signals to control a three phase permanent magnet brushless DC motor [2]. Rambabu (2007) presented the use of a Fuzzy logic controller for speed control of a BLDC motor through simulation. He used a PI controller for implementation on an experimental BLDC motor set up [3]. Safwat, Elwakeel, Eliwa and Abdelsattar (2010) used a BLDC motor with a resolver to exchange the momentum of a satellite. They used a PID and FLC to control the motor angular position with comparison of the performance of the control system using the two controllers [4]. Ramesh, Amarnath, Kamakshaiah and Rao (2011) presented a fuzzy-PI controller for speed control of a BLDC motor. They presented also the modeling of the BLDC motor. They uses Simulink (of MATLAB) to simulate the proposed scheme for variable load torque [5].

Abkenar (2014) modeled a BLDC motor and its control drive in simulink. He discussed the direct torque control switching technique of the BLDC motor. He implemented a digital PWM technique to control the speed of the motor [6]. Purnalal and Kumar (2015) simulated a closed loop speed control of a BLDC motor and its performance. They used the simulink software to simulate their work at different load torques [7]. Wen and Li (2017) modeled a BLDC motor and used three control strategies to control: PID, fuzzy control and fuzzy-PID control. Through comparison of the step time response of the control system, they concluded that the fuzzy-PID controller was the best of the three [8]. Anshory, Robandi and Wirawan (2018) obtained the mathematical model of a brushless DC motor in the form of a transfer function. They used sensors to measure current, voltage and speed of a prototype BLDC motor. They used the MATLAB System Identification Toolbox to obtain the transfer function model of the motor [9].

Ramachanaran, Ganeshaperumal and Subathra (2019) presented a closed loop simulation for a brushless direct current motor used in an electric vehicle application. They presented a comprehensive study covering conventional and auto-tuned optimal method for PID controller. They concluded that the auto-tuned PID controller had good track efficiency in achieving the target speed [10]. Mahmoud, Motakabber, Alam and Nordin (2020) described the design of a BLDC motor control using MATLAB/Simulink for PID controller. They compared with PI and fuzzy logic controllers and concluded that the PID controller provided the best performance compared with the other two controllers (!) [11]. Beladjine et. al. (2021) discussed a comparative study between closed loop PI control and Artificial Neural Network (ANN) based control for a brushless DC motor. They used MATLAB in their analysis and concluded that the ANN controller was better than the PI controller [12].

2. DYNAMICS OF THE BLDC MOTOR

The dynamics of the selected brushless DC motor (BLDC) depend on its transfer function. A typical assigned transfer function of a BLDC motor, $G_p(s)$ is given by [9]:

$$G_p(s) = (3.473s^2+27.11s+0.7475) / (s^3+3.24s^2+12.55s+0.4053) \qquad (1)$$

The transfer function of the BLDC motor under study has a 2/3 order. Its unit step time response is obtained using the 'step' command of MATLAB [13]. It is shown in Fig.1.

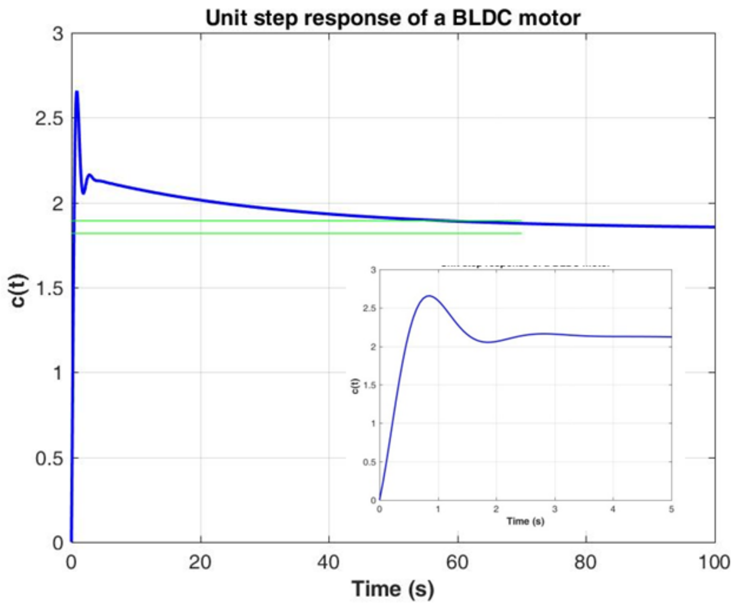

Figure 1: Unit Step Time Response of the BLDC motor.

The step response in Fig.1 reveals the following time-based characteristics of the BLDC motor:

- Maximum percentage overshoot: 43.59 %
- Settling time: 62.00 s
- Time of peak time response: 0.86 s
- Steady state error (step magnitude-steady state response): -0.8508

3. CONTROLLING THE BLDC MOTOR USING A CONVENTIONAL PID CONTROLLER

The block diagram of the control system incorporating the PID controller and the BLDC motor is shown in Fig.2.

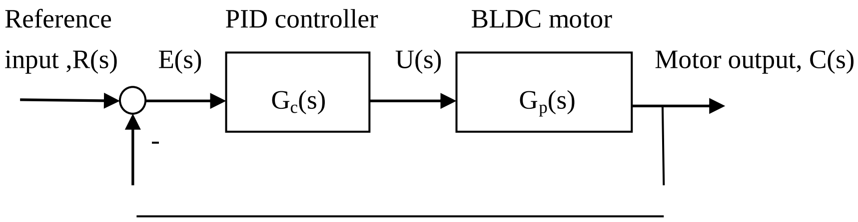

Figure 2: PID Controlled BLDC motor .

The transfer function of an ideal PID controller, $G_c(s)$ is given by:

$$G_c(s) = K_{pc} + (K_i/s) + K_d s \qquad (2)$$

where: K_{pc} = proportional controller gain.

K_i = integral controller gain.

K_d = derivative controller gain.

The transfer function of the closed loop control system is obtained using the block diagram in Fig.2 and the transfer functions of the controller (Eq.2) and the process (Eq.1). This transfer function of the closed loop control system is used to assign the step response for reference input tracking using the MATLAB commands 'step' and 'plot' [13].

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
Vol. 02, Issue 04, April-2022, pp: 5-14
e-ISSN : 2583-1062
Impact Factor : 2.205

Tuning of the Conventional PID Controller:

The PID controller is tuned by the author using the MATLAB optimization toolbox using its command 'fmincon' for constrained optimization [14], ISTSE objective function and functional constraints on the maximum percentage overshoot, settling time and stability. This tuning procedure resulted in the following PID controller parameters:

$$K_{pc} = 2.0076 \quad , \quad K_i = 9.2272 \quad , \quad K_d = 0.6529 \qquad (3)$$

Now, the unit step response of the control system is plotted using the transfer function of the closed loop control system, the tuned PID controller parameters in Eq.3 using the ‘step’ and 'plot' commands of MATLAB [13]. The step time response of the control system for step input tracking is shown in Fig.3.

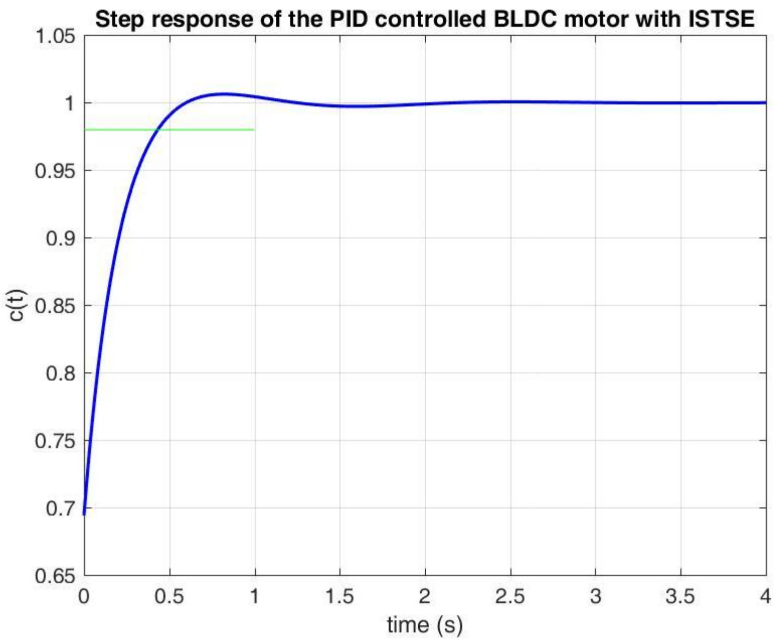

Figure 3: Unit Step Time Response using a Tuned PID Controller.

The step time response of the control system in Fig.3 reveals the following time-based characteristics:

- Maximum percentage overshoot: 0.6365 %
- Settling time: 0.425 s
- Steady state error: 0

4. CONTROLLING THE BLDC MOTOR USING A PD-I CONTROLLER

The PD-I controller is a one of the second generation of PID controllers investigated by the author [15]. The structure of a PD-I controller is shown in Fig.4 [16]. It has the transfer function, $G_c(s)$:

$$G_c(s) = (K_{pc} + K_d s)(K_i/s) \qquad (4)$$

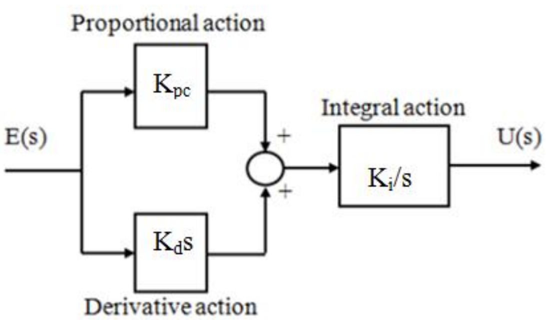

Figure 4: Structure of the PD-I Controller [16].

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
Vol. 02, Issue 04, April-2022, pp: 5-14
e-ISSN : 2583-1062
Impact Factor : 2.205

The transfer function of the closed loop control system is obtained using the block diagrams in Fig.2 and Fig.4, the transfer functions of the controller (Eq.4) and the process (Eq.1). This transfer function of the closed loop control system is used to assign the step response for reference input tracking using the MATLAB commands 'step' and 'plot' [13].

Tuning of the PD-I Controller:

The PD-I controller is tuned by the author using the MATLAB optimization toolbox using its command 'fmincon' for constrained optimization [14], ISTSE objective function and functional constraints on the maximum percentage overshoot, settling time and stability. This tuning procedure resulted in the following PD-I controller parameters:

$$K_{pc} = 70.2802 \quad , \quad K_d = 1.7402 \quad , \quad K_i = 4.6311 \qquad (5)$$

Now, the unit step response of the control system is plotted using the transfer function of the closed loop control system, the tuned PD-I controller parameters in Eq.5 using the 'step' and 'plot' commands of MATLAB [13]. The step time response of the control system for step input tracking is shown in Fig.5.

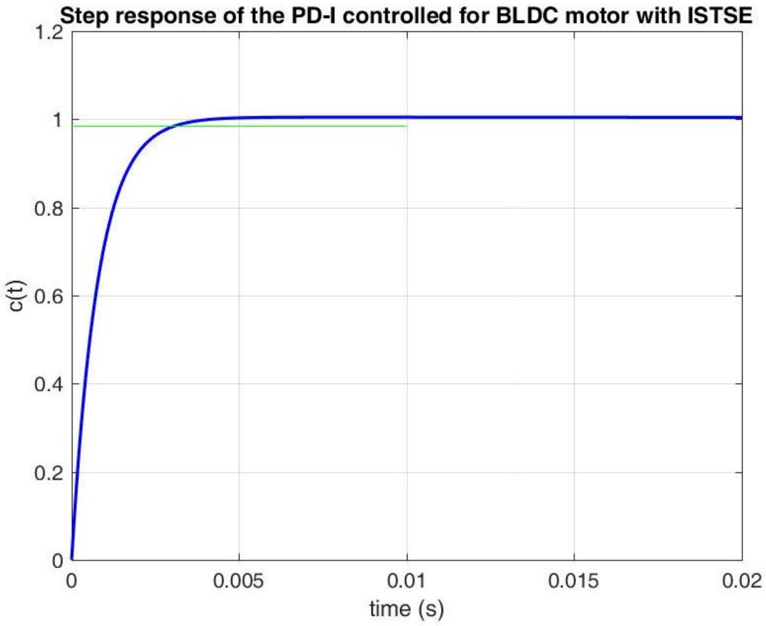

Figure 5: Unit Step Time Response using a Tuned PD-I Controller.

The step time response of the control system in Fig.5 reveals the following time-based characteristics:

- Maximum percentage overshoot: 0.048 %
- Settling time: 0.003 s
- Steady state error: 0

5. CONTROLLING THE BLDC MOTOR USING A PD-PI CONTROLLER

The structure of the PD-PI controller is shown in Fig.6 [17,18]. It is composed of two elements, a PD mode element with unit proportional gain and a derivative gain K_d and a PI mode element with proportional gain K_{pc} and an integral gain K_i.

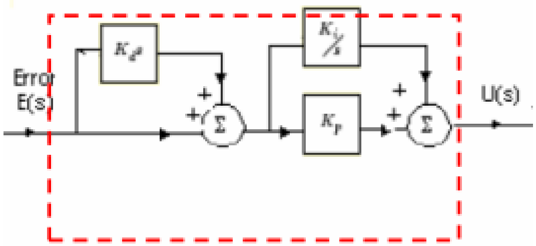

Figure 6: Structure of the PD-PI Controlled [17,18].

Using the block diagram in Fig.4, the transfer function of the PD-PI controller, $G_c(s)$ is given by:

$$G_c(s) = (1 + K_d s)[K_{pc} + (K_i/s)] \qquad (6)$$

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
www.ijprems.com
editor@ijprems.com
Vol. 02, Issue 04, April-2022, pp: 5-14
e-ISSN : 2583-1062
Impact Factor : 2.205

The transfer function of the closed loop control system is obtained using the block diagrams in Fig.2 and Fig.6, the transfer functions of the controller (Eq.6) and the process (Eq.1). This transfer function of the closed loop control system is used to assign the step response for reference input tracking using the MATLAB commands 'step' and 'plot' [13].

Tuning of the PD-PI Controller:

The PD-PI controller is tuned by the author using the MATLAB optimization toolbox using its command 'fmincon' for constrained optimization [14], ITAE objective function and functional constraints on the maximum percentage overshoot, settling time and stability. This tuning procedure resulted in the following PD-PI controller parameters:

$$K_{pc} = 12.572 \quad , \quad K_i = 204.0773 \quad , \quad K_d = 18.9161 \qquad (7)$$

Now, the unit step response of the control system is plotted using the transfer function of the closed loop control system, the tuned PD-PI controller parameters in Eq.7 using the ‘step’ and 'plot' commands of MATLAB [13]. The step time response of the control system for step input tracking is shown in Fig.7.

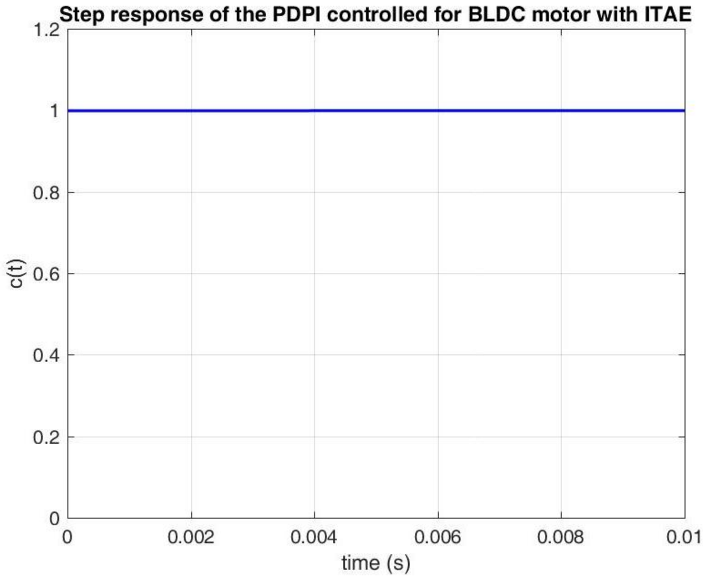

Figure 7: Unit Step Time Response using a Tuned PD-PI Controller.

The step time response of the control system in Fig.7 reveals the following time-based characteristics:

- Maximum percentage overshoot: 0 %
- Settling time: 0 s
- Steady state error: 0

6. CONTROLLING THE BLDC MOTOR USING A PI-PD CONTROLLER

The block diagram of the control system incorporating a PI-PD controller and the controlled motor is shown in Fig.8[19,20].

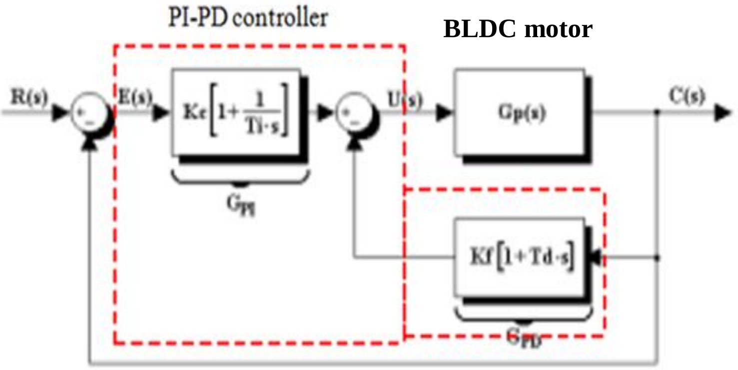

Figure 8: PI-PD Controlled BLDC motor [19,20].

Using the block diagram in Fig.8, the mathematical model of the PI-PD controller is:

$$U(s) = [K_c + (K_i/s)[R(s) - C(s)] - (K_f + K_d s)C(s) \quad (8)$$

where: K_c = proportional gain of the PI mode.
K_i = integral gain of the PI mode.
K_f = proportional gain of the PD mode.
K_d = derivative gain of the PD mode.

The transfer function of the closed loop control system is obtained using the block diagrams in Fig.2 and Fig.8, the transfer functions of the controller (Eq.8) and the process (Eq.1). This transfer function of the closed loop control system is used to assign the step response for reference input tracking using the MATLAB commands 'step' and 'plot' [13].

Tuning of the PI-PD Controller:

The PI-PD controller is tuned by the author using the MATLAB optimization toolbox using its command 'fmincon' for constrained optimization [14], ISTSE objective function and functional constraints on the maximum percentage overshoot, settling time and stability. This tuning procedure resulted in the following PI-PD controller parameters:

$$K_c = 0.8271 \quad , \quad K_i = 96.8597 \quad , \quad K_f = 3.2465 \quad , \quad K_d = 0.0289 \quad (9)$$

Now, the unit step response of the control system is plotted using the transfer function of the closed loop control system, the tuned PI-PD controller parameters in Eq.9 using the 'step' and 'plot' commands of MATLAB [13]. The step time response of the control system for step input tracking is shown in Fig.9.

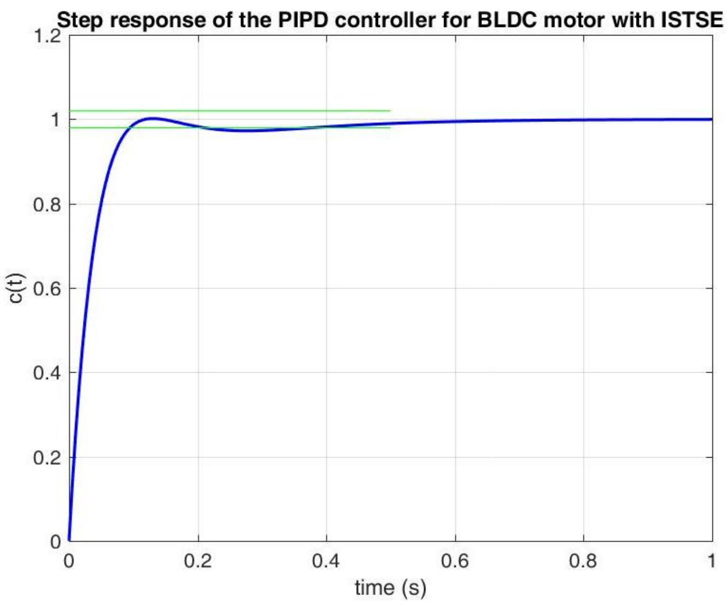

Figure 9: Unit Step Time Response using a Tuned PI-PD Controller.

The step time response of the control system in Fig.9 reveals the following time-based characteristics:

- Maximum percentage overshoot: 0.194 %
- Settling time: 0.376 s
- Steady state error: 0

7. CONTROLLING THE BLDC MOTOR USING A 2DOF-PID CONTROLLER

The block diagram of the control system incorporating a 2DOF-PID controller and the controlled motor is shown in Fig.10 [21,22].

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
Vol. 02, Issue 04, April-2022, pp: 5-14
e-ISSN : 2583-1062
Impact Factor : 2.205

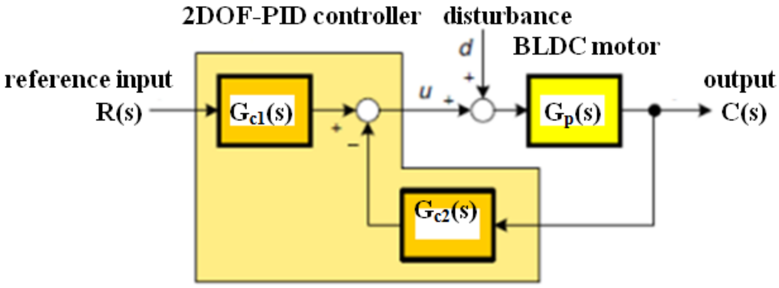

Figure10: 2DOF-PID Controlled BLDC Motor [21, 22].

In 2015 the author used a PI-controlling mode for the elements of the 2DOF controller when controlling a highly oscillating second order process with different proportional gains and same integral gain [22]. Here, in this application I propose a PID controlling mode for G_{c1} (s) and G_{c2} (s) with different proportional gain and same integral and derivative gains.

Using the block diagram in Fig.10, the mathematical model of the 2DOF-PID controller is:

$$U(s) = G_{c1}(s)R(s) - G_{c2}(s)C(s) \qquad (10)$$

where $G_{c1}(s) = K_{pc1} + (K_i/s) + K_d s$

$$G_{c2}(s) = K_{pc2} + (K_i/s) + K_d s$$

The transfer function of the closed loop control system is obtained using the block diagrams in Fig.2 and Fig.10, the mathematical model of the controller (Eq.10) and the transfer function of the process (Eq.1). This transfer function of the closed loop control system is used to assign the step response for reference input tracking using the MATLAB commands 'step' and 'plot' [13].

Tuning of the PI-PD Controller:

The 2DOF-PID controller is tuned by the author using the MATLAB optimization toolbox using its command 'fmincon' for constrained optimization [14], ISTSE objective function and functional constraints on the maximum percentage overshoot, settling time and stability. This tuning procedure resulted in the following 2DOF-PID controller parameters:

$$K_{pc1} = 8.1695 \quad , \quad K_i = 28.4913 \quad , \quad K_d = 2.4918 \quad , \quad K_{pc2} = 8.0581 \qquad (11)$$

Now, the unit step response of the control system is plotted using the transfer function of the closed loop control system, the tuned 2DOF-PID controller parameters in Eq.11 using the 'step' and 'plot' commands of MATLAB [13]. The step time response of the control system for step input tracking is shown in Fig.11.

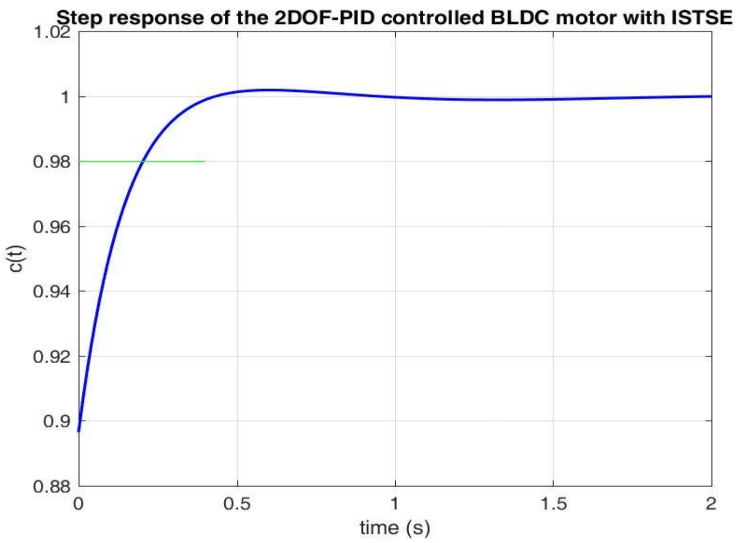

Figure 11: Unit Step Time Response using a Tuned 2DOF-PID Controller.

The step time response of the control system in Fig.11 reveals the following time-based characteristics:

- Maximum percentage overshoot: 0.195 %
- Settling time: 0.200 s
- Steady state error: 0

Comparison of Controllers for BLDC Motor Control

The analysis presented in the previous sections was focused on the control of a BLDC motor using five controllers: PID controller from the first generation of PID controllers and PD-I, PD-PI, PI-PD and 2DOF-PID controllers from the second generation presented by the author [15].

The unit step response of the control system representing the reference input tracking of the control system is shown in Figure 12 using PID, PD-I, PD-PI, PI-PD and 2DOF-PID controllers.

The characteristics of the reference input tracking associated with the BLDC motor using the five controllers are collected in Table 1.

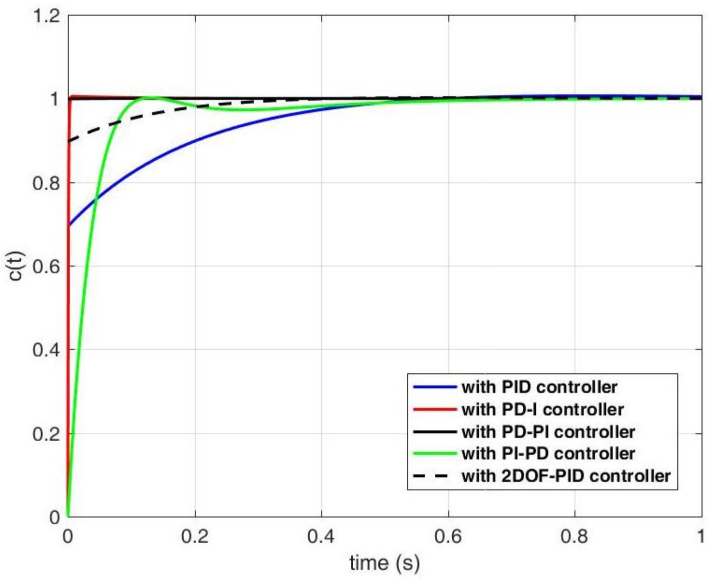

Figure 12: Step Time Response of the BLDC Motor Controlled using Five Controllers.

Table 1: Reference input tracking characteristics associated with the BLDC motor.

Controller	PID	PD-I	PD-PI	PI-PD	2DOF-PID
Number of Parameters	3	3	3	4	4
Error Criterion	ISTSE	ISTSE **(2)**	ITAE **(1)**	ISTSE	ISTSE
OS_{max} (%)	0.637	0.048	0	0.194	0.1953
T_s (s)	0.425	0.003	0	0.376	0.200
e_{ss}	0	0	0	0	0

8. CONCLUSION

- The paper investigated the use of a PID controller from the first generation and four controllers from the second generation of PID controllers to control a BLDC motor.
- The four used controllers from the second generation of PID controllers are the I-PD, PD-PI, PI-PD and 2DOF-PID controllers.
- The controlled BLDC motor had relatively bad dynamics characterized by 43.6 % maximum percentage overshoot, 62 s settling time and 0.851 steady state error.
- In this control application the step time response using a PID controller to control the BLDC motor did not show any kick associated reference input tracking.

- The four controllers from the second generation didn't show any kick in the reference input tracking step response.
- The tuned PID controller proposed to control the motor could reduce the maximum percentage overshoot to 0.6365 %, the settling time to 0.425 s and the steady state error to zero.
- The tuned PD-I controller proposed to control the motor could reduce the maximum percentage overshoot to 0.048 %, the settling time to 0.003 s and the steady state error to zero.
- The tuned PD-PI controller proposed to control the motor generate a step-like time response with zero overshoot, settling time and steady state error..
- The tuned PI-PD controller proposed to control the motor could reduce the maximum percentage overshoot to 0.194 %, the settling time to 0.376 s and the steady state error to zero.
- The tuned PID controller proposed to control the motor could reduce the maximum percentage overshoot to 0.195 %, the settling time to 0.200 s and the steady state error to zero.
- The time based characteristics of the control system were compared using the five controllers graphically and numerically.
- The comparison revealed the best controller to control the BLDC motor which is the PD-PI controller, then the PD-I controller from the second generation of PID controllers.
- The PID controller was the last controller suitable for the control of the BLDC motor.

9. REFERENCES

[1] Sontain Motor, "Brushless DC motor application", https://sontianmotor.com/brushless-dc-motor-applications/ , 9 September, 2021.

[2] N. Kim, O. Yang and M. Kim, “Transfer function model based analysis of application using a general purpose processor”, Journal of Power Electronics, vol.7, issue 2, 2007 , pp.132-139.

[3] S. Rambabu, “Modeling and control of a brushless DC motor", Master of Technology Thesis, National Institute of Technology, Rourkela, 2007.

[4] I. Safwat, A. Elwakeel, A. Eliwa and A. Abdelsattar, "Control of BLDC motor for satellite momentum exchange", 7th International Conference on Electrical Engineering, Military Technical College, Cairo, Egypt, 25-27 May, 2010 15 pages.

[5] M. Ramesh, J. Amarnath, S. Kamakshaiah and G. Rao, "Speed control of brushless DC motor by using fuzzy logic PI controller", ARPN Journal of Engineering and Applied Sciences, vol.6, issue 9, 2011, pp.55-62.

[6] A. T. Abkenar, "BLDC motor drive controller for electric vehicles", Ph. D. Thesis, Faculty of Science, Engineering and Technology, University of Technology, Swinburne, 2014.

[7] M. Purnalal and S. Kumar, "Development of mathematical model and speed control of BLDC motor", International Journal of Electrical and Electronics Engineers, vol.7, issue 1, 2015, pp.271-280.

[8] X. Wen and Z. Li, "Brushless DC motor speed control strategy of simulation research", MATEC Web of Conference, vol.139, 00172, 2017, 6 pages.

[9] I. Anshory, I. Roband and W. Wirawan, "Parameters identification BLDC motor: Instrumentation and transfer functions", MATEC Web of Conferences, issue 197, 11012, 2018, 4 pages.

[10] R. Ramachandran, D. Ganeshaperumal and B. Subathra, "Closed loop control of BLDC motor in electric vehicle", IEEE International Conference on Clean Energy and Energy Efficient Electronics Circuit for Sustainable Development, Krishnanankoil, India, 18-20 December 2019.

[11] M. Mahmud, S. Motakabber, A. Alam and A. Nordin, "Control BLDC motor speed using PID controller", International Journal of Advanced Computer Science and Applications, vol.11, issue 3, 2020, pp.477-481.

[12] D. Beladjine, D. Boudana, MA. Moualdia, M. Hallouz and P. Wira, "A comparative study of BLDC motor speed control using PI and ANN regulator", 18th IEEE International Multi- Conference on Systems, Signals and Devices, Monastir, Tunisia, March 22-25, 2021, 6 pages.

[13] C. Houpis and S. Shelden, “Linear control system analysis and design with MATLAB”, 6th Edition, CRC Press, 2013.

[14] A. Massac, “Optimization in practice with MATLAB for engineering students and professionals”, 1st Edition, Cambridge University Press, 2015.

[15] G. A. Hassaan, “Second generation of PID controllers for reference input tracking”, International Journal for Progressive Research in Engineering Management and Science, vol.1, issue 3, 2021 , pp.25-30.

[16] G. A. Hassaan, "Novel PD-I controller for underdamped second order-like processes", International Journal of Engineering and Techniques, vol.4, issue 2, 2018 , pp.181-187.

[17] J. Jain and M. Nigram, "Optimization of PD-PI controller using swarm intelligence", Journal of Theoretical and Applied Information Technology, vol.4, issue 11, 2008 , pp.1013-1018.

[18] G. A. Hassaan, "Tuning of an PD-PI controller used with first order delayed processes", International Journal of Engineering Research and Technology, vol.3, issue 4, 2014 , pp.2751-2755.

[19] I. Kaya, P. Derek and P. Atherton, "Simple analytical rules for PI-PD controllers to tune integrating and unstable processes", International Control Conference, Glasgow, August 2006, 5 pages.

[20] G. A. Hassaan, "Tuning of a PI-PD controller used with a highly oscillating second order process", International Journal of Research and Innovative Technology, vol.1, issue 3, 2014 , pp.42-45.

[21] D. Vrancic, S. Strmcnik, M. Huba and P. Oliveira. "Comparison of some tuning methods for integrating processes", 9th International Ph. D. Workshop on Systems and Control, Izola, Sloveria, 1-3 October, 2008, 6 pages.

[22] G. A. Hassaan, "Tuning of an 2DOF controller used with a highly oscillating second order process", International Journal of Modern Trends in Engineering and Research, vol.2, issue 8, 2015 , pp.292-298.