DC MOTOR SPEED CONTROL USING INTERNAL MODEL CONTROLLER: AN INDUSTRIAL TRANSFORMATION STRATEGY

[1]USMAN OMEIZA AHMED AND [2]PATRICK AYUBA AVONG

Department of Electrical and Electronic Engineering Technology, Federal Polytechnic Nasarawa, P.M.B 01, Nasarawa, Nasarawa State, Nigeria.

Abstract

In developed nations, industries are made to function at control engineering costs. For industrial applications, adopted speed control scheme is fundamentals and determine the dc motor's performance. This paper seeks to analyze and validate performance of dc motor in Internal Model Control (IMC) and Proportional–Integral–Derivative (PID) control schemes. The IMC was realized by cascading inverse of the target control process model with the model of the reference process transfer function. This is aim to achieve unity gain that could track the set-point. The methodology involved getting specifications for the dc motor (model RMCS-3011) and then derivation of motor's IMC controller transfer function. Simulated plots in Matlab-Simulink using the PID and IMC were presented. The quantitative results of the IMC method compared with the Matlab-Simulink tuned PID controller provides a commendable performance. Thus, its being recommended for consideration and application in Nigeria industries.

KEYWORDS:
Industries, controller, dc motor, performance, model.

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

INTRODUCTION

According to Sao and Singh (2015), dc motors are used to generate rotational speed and linear position control industrial applications which may include electric cranes, steel rolling mills, electric vehicles and robotic manipulators etc. The achievable efficiency controlling the motor's speed and torque is critical and of which higher performance value is vital for mentioned industrial applications. This necessitate the need for higher performing dynamic control specifically targeting transient response and steady-state response of the motor. Proportional–Integral–Derivative (PID) is the most used among other control schemes over the years; it enables flexible, precise and simplified speed and position control of the dc motor. In this paper, the Internal Model Control (IMC) controller was examined and verified using referenced Matlab-tuned PID controller model.

The identified inefficiencies of the conventional PID controller which is attribute of susceptibility to noise at high frequency, sensitivity to controller gain, undesirable speed overshoot at start and sluggish response due to sudden change in load torque (Al Nisa et al., 2013). Adopting PID controllers could be attributed to its unsatisfactory performance; reason been dc motors though modeled as linear systems are often subjected to nonlinearities.

Therefore, implementing others control scheme could be optimization of system's stability and achieving satisfactory performance. This research's objective will include to validate performance of the IMC and PID controllers.

Al Nisa et al. (2013) provided detailed description of dc motor and its areas of applications where larger bandwidth speed is needed. The ease in speed control gives dc motors greater advantage in applications where variable speed drives is required. Theoretical and real analysis have identified relationship between dc motor speed, armature voltage, magnetic flux per poles; Thus, speed of rotor can be set by controlling armature voltage and/or the field current.

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

Model of a DC Motor

Figure 1 depicts the control equivalent circuit of a dc motor (Meshram & Kanojiya, 2012). With reference to Figure 1, the dc motor model can be described as follows:

R_a : armature resistance
L_a : armature inductance
i_a : armature current
i_f : field current
v_a : input voltage
e_b : back electromotive force (e. m. f)
T_m : motor torque

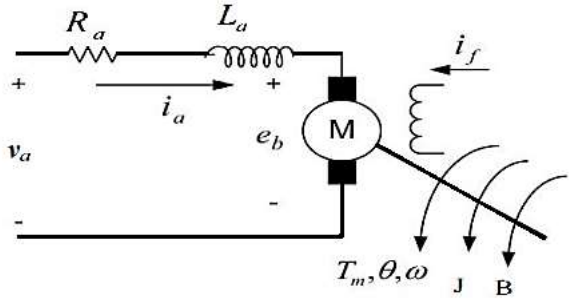

Figure 1: Control Equivalent Circuit of DC Motor (Meshram & Kanojiya, 2012)

θ : rotor angular speed
ω : rotor angular velocity
J : moment of inertia
b : viscous friction coefficient of the motor

The armature voltage equation is given by

$$v_a = R_a i_a(t) + L_a \frac{d\, i_a(t)}{dt} + e_b(t) \quad (1)$$

The back e.m.f equation of the motor will be

$$e_b = K_b\, \omega(t) \quad (2)$$

where

K_b : back e. m. f constant

From Newton's law, the motor torque can be obtained as

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

$$T_m(t) = J\frac{d^2\theta(t)}{dt} + b\frac{d\theta}{dt} \quad (3)$$

$$T_m(t) = K_T\, i_a(t) \quad (4)$$

Thus,

$$K_T\, i_a(t) = J\frac{d^2\theta(t)}{dt} + b\frac{d\theta}{dt} \quad (5)$$

$$K_T\, i_a(t) = J\frac{d\omega(t)}{dt} + b\,\omega(t) = T_m(t) \quad (6)$$

where

K_T : Torque constant

Taking Laplace transform of equations (1), (2) and (6) respectively gives

$$V_a(s) = (R_a + L_aS)I_a(s) + E_b(s) \quad (7)$$

$$E_b(s) = K_b\,\omega(s) \quad (8)$$

$$K_TI_a(s) = Js\,\omega(s) + B\,\omega(s) = T_m(s) \quad (9)$$

Figure 2 describes the dc motor armature control system function block diagram derived from equation (7) to (9). The input voltage $V_a\,(s)$ and the output angular velocity $\omega(s)$ are related by the transfer function:

$$G(s) = \frac{\omega(s)}{V_a(s)} = \frac{K_T}{(sL_a + R_a)(Js + B) + K_bK_T} \quad (10)$$

Equation (10), on expansion and factorization of its denominator can be rewrite as:

$$G(s) = \frac{\omega(s)}{V_a(s)} = \frac{K_T}{(L_aJ)s^2 + (L_aB + R_aJ)s + (R_aB + K_bK_T)} \quad (11)$$

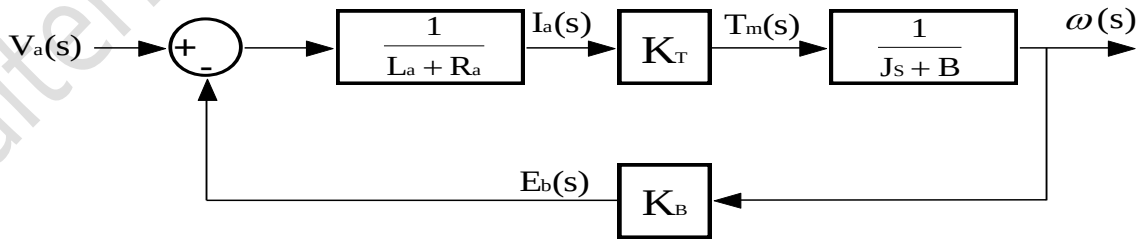

Figure 2: Function Block Diagram of a DC Motor Armature Voltage Control System (Meshram & Kanojiya, 2012)

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

Speed Control of DC Motor

Electric motor convert electrical energy to mechanical energy by interactions of magnetic fields. One field is produced by a magnet of poles assembly; the other produced when an electrical current flowing in the motor windings. Kushwah and Patra, 2014) have highlighted advantages of dc motor to include high start torque, high response performance, easier to be linear controlled and low cost etc. this makes them find widely applicable in the industries. Singh *et al.* (2013) outlined characteristics equation that describes the relationship of speed of a dc motor as

$$N = \frac{V_a - I_a R_a}{K\emptyset} \tag{12}$$

where

N is the DC motor speed

V_a is the armature terminal voltage

I_a is armature current

R_a is armature circuit resistance

$\emptyset$ is the field flux

K is armature constant and is given by

$$K = \frac{PZ}{2\pi a} \tag{13}$$

where

$P =$ No. of poles

$Z =$ Total no. of armature conductors

$a =$ No. of parallel path

Equation (2.12) shows that for a DC motor there are basically three methods of speed control:

1. Variation of resistance in armature circuit.
2. Variation of field flux
3. Variation of armature terminal voltage.

Control Schemes used for Speed Control of DC Motor

The load on a dc motor may vary over a range of speed depending on the type of application. Certain applications may demand high-speed control

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

accuracy and good dynamic responses. Thus, the motor should be precisely controlled to give the desired performance. There are several controller types: proportional integral controller, PID controller, neural network controller, fuzzy logic controller and linear quadratic regulator controller (Tripathi et al., 2015).

Proportional-Integral-Derivative (PID) Controller

Moradi and Saeedi (2016) explains the PID controller as a control loop feedback mechanism that calculates an "error" value as the difference between a measured process variable and a desired set point; the error is minimized by adjusting the process control inputs. A PID controller consists of parallel connections of three elements: proportional element, integral element and derivative element. These elements take the error as input, use it to compute new input process. Each new input tries level the measured process to the desired set point values. In addition, Meshram & Kanojiya (2012) describes the PID controller as a design aim to achieve optimal control performance by adjusting an appropriate proportional gain K_p, integral gain K_i, and differential gain K_d. Figure 3, shows a PID controller system block diagram. The mathematical expression of the PID controller is:

$$y(t) = K_p \left[e(t) + \frac{1}{K_i} \int_0^t e(t)\, dt \; + \; K_d \frac{d}{dt}\, e(t) \right] \qquad (14)$$

The Laplace domain representation of PID controller is:

$$\frac{Y(S)}{E(S)} = K_p \left[1 + \frac{1}{K_i S} + K_d S \right] \qquad (15)$$

PID controllers have the following features:

- They have the capacity to eliminate steady-state error of the response to a step reference signal (because of integral action).
- They have the ability to anticipate output changes (when derivative action is employed).

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

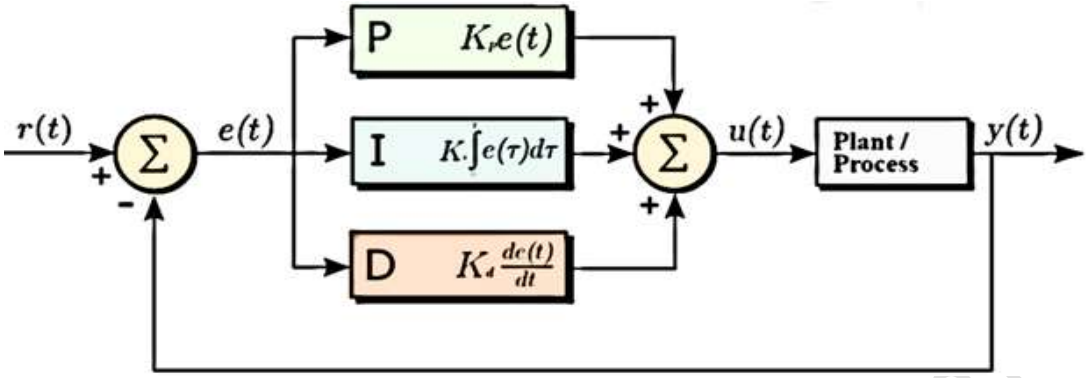

Figure 3: PID Controller System Block Diagram (Urquizo, 2011)

Internal Model Control (IMC) Controller

Saranya & Pamela, (2012) assert linear control schemes may not guaranty satisfactory performance when implemented upon a linear modeled system such as a dc motor. This may be as a result of variation in motor-load dynamics and due to dc motor armature reaction introducing nonlinearities. The use of internal model control (IMC) controller can give better performance amidst external disturbances that risk the stability of a close loop system.

Ahmad et al. (2014) in describing the IMC stated the theory of IMC thus: "control can be achieved only if the control system encapsulates, either implicitly or explicitly, some representation of the process to be controlled". This encapsulation is obvious, as the IMC controller is designed using the inverse of the intended control process model. This is realized by cascading the IMC controller with the process transfer function, such that the process model is in parallel with the actual process. Thus, unity gain is achieved that results in an accurate tracking of the set-point. Figure 4 is the block diagram of an IMC controller as illustrated by Saranya & Pamela, (2012). The block symbols and signals are defined as follows:

$Y(s)$: Output signal
$R(s)$: Set-point
$d(s)$: Disturbance signal
$G_{IMC}(s)$: Internal Model Control (IMC) Controller
$G_P(s)$: Actual process to be controlled

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

$G_M(s)$: Model of the actual process

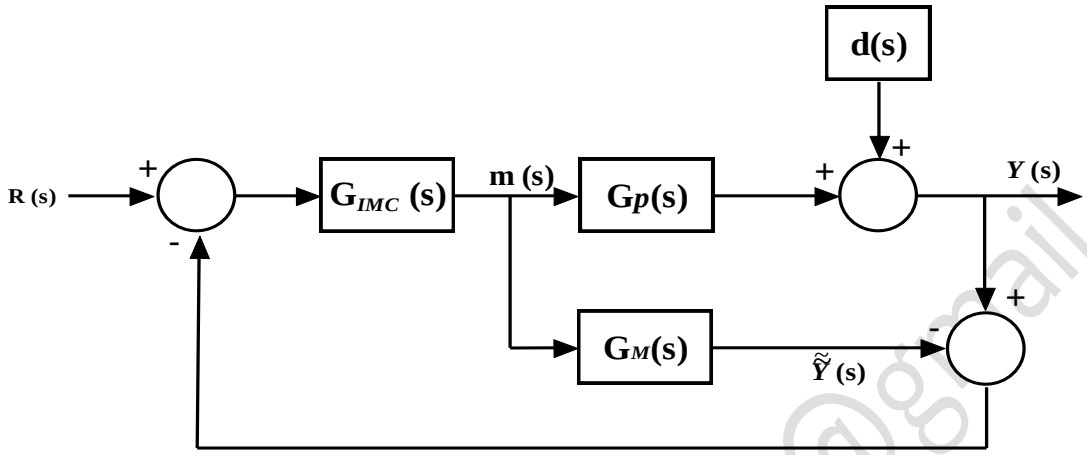

Figure 4: Block Diagram of IMC (Saranya & Pamela, 2012)

For the purpose of simplification, the IMC block diagram of Figure 4 can be redrawn as shown in Figure 5. From Figure 4,

$$Y(s) = G_P(s)\, m(s) + d(s) \qquad (16)$$

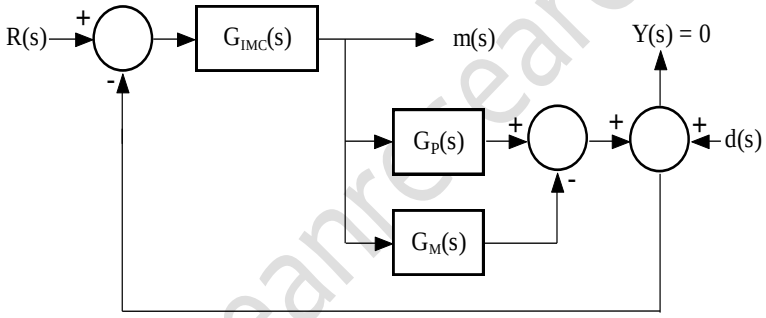

Figure 5: Restructured Block Diagram of IMC

Using block diagram reduction technique, Figure 5 is reduced to Figure 6.

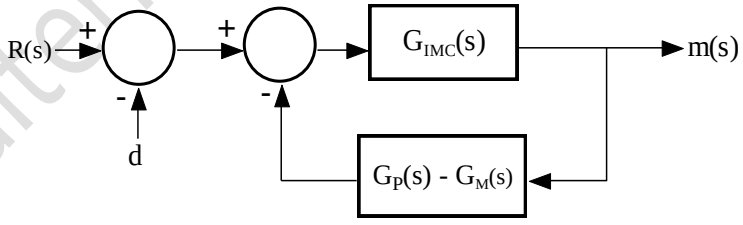

Figure 6: Resultant IMC Equivalent Block Diagram

With respect to Figure 9 and letting $Y(s) = 0$.

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

$$m(s) = \frac{G_{IMC}(s)}{1 + G_{IMC}(s)\,[G_P(s) - G_M(s)]}\,[R(s) - d(s)] \qquad (17)$$

According to Garcia and Morari (1982), substituting equation (17) into equation (16) yields

$$Y(s) = d(s) + \frac{G_P(s)\;G_{IMC}(s)}{1 + G_{IMC}(s)\,[G_P(s) - G_M(s)]}\,[R(s) - d(s)] \qquad (18)$$

"Perfect" control could be achieved by selecting

$$G_{IMC}(s) = \frac{1}{G_P(s)} \qquad (19)$$

This can be verified by substituting equation (19) into equation (18) while applying the equality of the process transfer function to that of the process model, i.e., $G_P(s) = G_M(s)$ (Garcia and Morari, 1982).

$$Y(s) = d(s) + \frac{G_P(s)\;G_{IMC}(s)}{1 + G_{IMC}(s)\,[G_P(s) - G_M(s)]}\,[R(s) - d(s)]$$

$$Y(s) = d(s) + \frac{G_P(s)\;\frac{1}{G_P(s)}}{1 + G_{IMC}(s)\,[G_M(s) - G_M(s)]}\,[R(s) - d(s)]$$

$$Y(s) = R(s) \qquad (20)$$

Equation (20) shows that the IMC whose transfer function is symbolized by $G_{IMC}(s)$ can make the output, $Y(s)$ to track the input, $R(s)$. Since $G_P(s) = G_M(s)$, equation (19) can be written as

$$G_{IMC}(s) = \frac{1}{G_M(s)} \qquad (21)$$

In accordance with dual stability criterion, when the model is exact representation of the process, stability of both controller and plant (the actual process) is sufficient for overall system stability. That in itself

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

depends on if the actual process is open loop stable. However, due to modelling error, the dual stability criterion may not be satisfied (Garcia and Morari, 1982). There is the need for the internal model control controller, $G_{IMC}(s)$, to be stable and realizable. Thus, a factorization of $G_M(s)$ in equation (21) is introduced such that

$$G_{IMC}(s) = \frac{1}{G_{MM}(s)\ G_{MA}(s)} \tag{22}$$

where,

$G_{MM}(s)$ is the transfer function component of $G_M(s)$ that has minimum phase characteristics (both $G_{MM}(s)$ and its inverse $\frac{1}{G_{MM}(s)}$ are causal and stable- by not having zeros and poles on the right-half plane). $G_{MA}(s)$ is the transfer function component of $G_M(s)$ that has non-minimum phase (NMP) characteristics which is undesirable part largely responsible for the instability of the IMC controller. Thus, for a stable IMC, the approximation of equation (23) results having factored out $\frac{1}{G_{MA}(s)}$.

$$G_{IMC}(s) \cong \frac{1}{G_{MM}(s)} \tag{23}$$

$$G_{MM}(s) \cong \frac{1}{G_{IMC}(s)} \tag{24}$$

Practical IMC controller, $G_{IMC\_P}(s)$ that is stable and robust can be realized by factoring in (in place of the factored out unstable NMP component, $\frac{1}{G_{MA}(s)}$) the transfer function of an nth-order low pass filter, $G_F(s)$ to the IMC controller of equation (23). The filter is selected such that it has all roots inside the unit circle.

$$G_{IMC\_P}(s) = G_{IMC}(s)\, G_F(s) = \frac{1}{G_{MM}(s)} G_F(s) \tag{25}$$

where,

$$G_F(s) = \frac{1}{(\lambda S + 1)^n} \tag{26}$$

where

λ is the time constant

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

n is the nth order of the filter

Figure 7 depicts the block diagram for the implementation of the internal model control scheme. An arbitrary gain of 0.001 was inserted along the feedback path to reduced and subsequently eliminate the external disturbances signal interference.

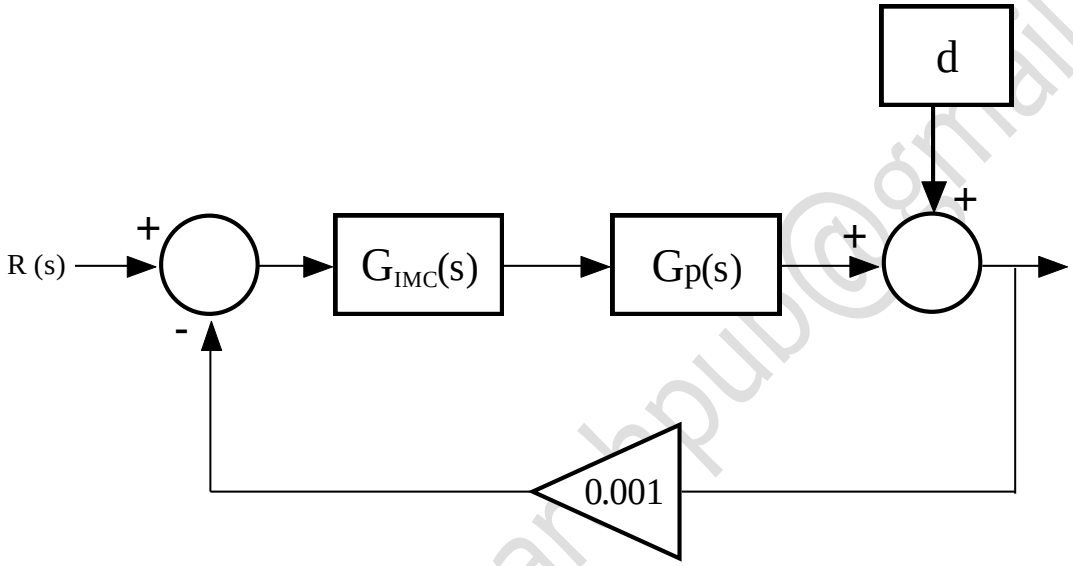

Figure 7: Implementation Block Diagram for IMC Scheme

Control System Performance Requirement

Performance of a control system is often measured by applying a step function as the set point command variable, and then measuring the response of the process variable. Commonly, as illustrated in Figure 8, the response is quantified by measuring defined waveform characteristics (National Instrument, 2018).

Rise Time: is the amount of time the system takes to go from 10% to 90% of the steady-state, or final, value.

Percent Overshoot: is the amount that the process variable overshoots the final value, expressed as a percentage of the final value.

Settling time: is the time required for the process variable to settle to within a certain percentage (commonly 5%) of the final value.

Steady-State Error: is the final difference between the process variable and set point.

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

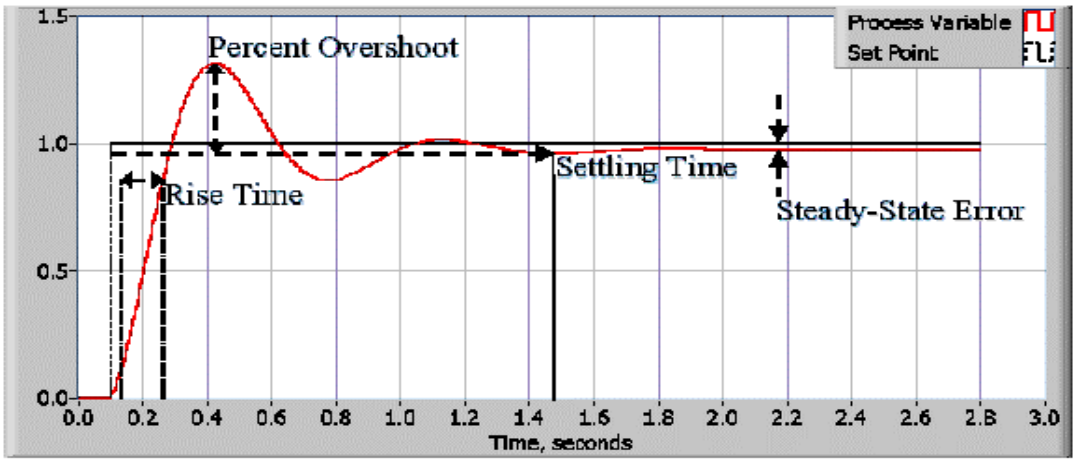

Figure 8: Response of a Typical PID Closed Loop System (National Instrument, 2018)

Method

DC Motor Specifications

The dc motor of choice is a geared brushed dc motor model RMCS-3011 with the following relevant manufacturer information:

Rated Voltage, V_a : 12 V
Rated Torque, T : 11.76798 Nm
Rated Speed, N : 10 rpm
Torque Constant, K_T : 66.4935 $Nm\ A^{-1}$
Back e.m.f constant, K_b : 18.2 $mVs\ rad^{-1}$
Moment of Inertia, J : 0.224 $Kg\ m^2$
Motor armature resistance, R_a: 2.96 Ω
Motor armature inductance, L_a: 2.51 mH

DC Motor (RMCS-3011) Transfer Function

Substituting the relevant manufacturer information and measured data of geared dc motor (model RMCS-3011) into equation (11) yields.

$$G_M(s) = \frac{1}{0.000008422s^2 + 0.010394s + 30.51677} \quad (27)$$

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

Simulation and Tuning of PID Controller

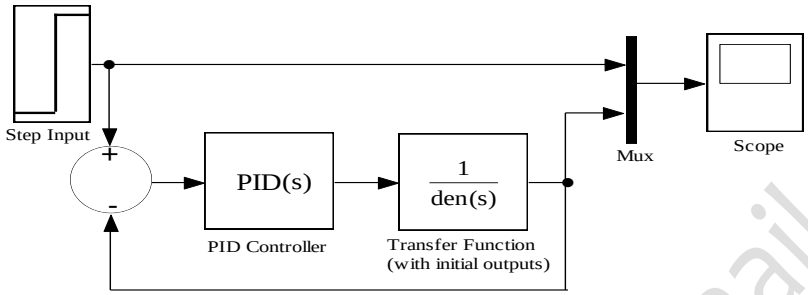

Figure 9: PID Controller Tuning Simulated Modelled Diagram for a DC Motor

Transfer Function of Practical IMC Controller

Considering the transfer functions $G_M(s)$ of equation (11) as the model of the process (i.e. the dc motor), $G_F(s)$ of equation (26) as the model of nth order filter, equations (23), (24) and (25). With reference to the derivation of the IMC formula and derivations of equation (25) and (26), a program to generate transfer function for practical IMC controller, $G_{IMC\_}p(s)$, using Matlab m-file environment is thus:

```
% Program to generate transfer function of practical internal model
% controller, gIMC_P
% The numerator and denominator of the process (dc motor) model
numM = [1]; denM = [0.000008422 0.010394 30.51677]; gM = tf(numM,denM);
% Computing the transfer function of the inverse of the process model
gM_inv = inv(gM);
% Enabling transfer function to be entered in the 's' domain
S = tf('s');
% Generate an 'n' order filter (n = 1) transfer function whilst setting
% time constant, Y = 'certain variable value'
Y = 0.1; n = 1; gF = 1/((Y*S + 1)^n);
% Generating IMC transfer function
gIMC = gM_inv*gF;
% The transfer function that has minimum phase characteristics, gMM
gMM = inv(gIMC);
```

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

% Generating the Practical Internal Model Control Controller, gIMC_P
gIMC_P = gF/gMM

Within the program, the time constant, Y, can be varied whilst keeping the order of the filter, $n = 1$, constant. The numerator and denominator values of the generated transfer function, $G_{IMC\_P}(s)$, are entered into the IMC block of Simulink modeled IMC dc motor diagram (Figure 10). The Scope block gives the performance response of the IMC.

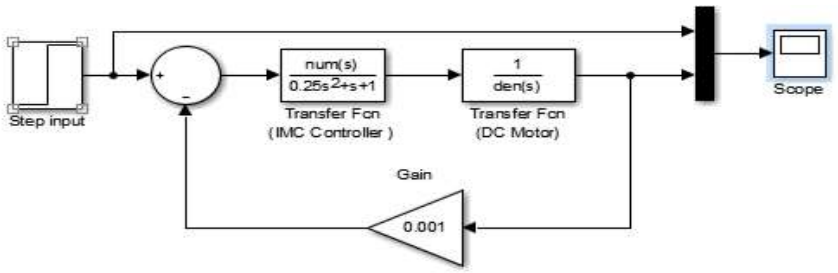

Figure 10: Simulink Modelled IMC DC Motor Diagram

Results

The unit step response for tuned PID controlled dc motor speed control was plotted (Figure 11). The unit-step response plots were generated for IMC controlled dc motor speed control at three varying time constant, $(\lambda = 0.1)$, $(\lambda = 0.5)$ and $(\lambda = 0.9)$ for both first order filter $(n = 1)$ and second order filter $(n = 2)$. Though, the best plot having $\lambda = 0.1$ with $n = 2$ is shown in Figure 12. The performance parameters of each IMC plot were recorded and tabulated in Table 1. Lastly, the performance parameters of both controllers based on their respective plots were tabulated (Table 2) for the purpose of verification.

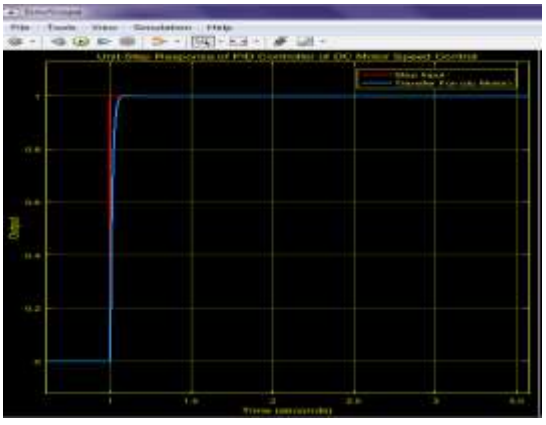

Figure 11: Step Response of Tuned PID Controlled DC Motor Speed Control

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

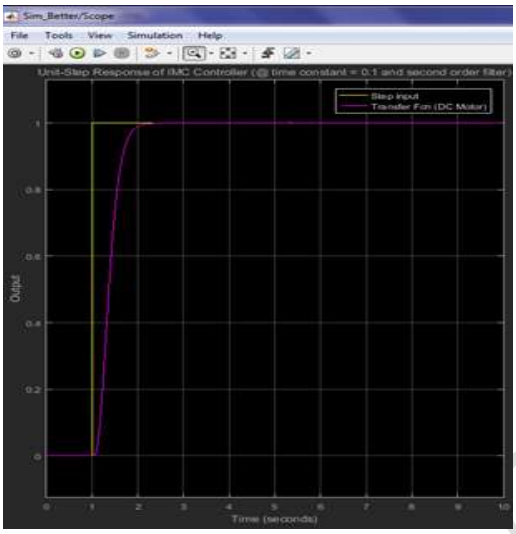

Figure 12: Step Response of IMC Controlled DC Motor Speed Control (Time Constant, $\lambda = 0.1$; Second Order Filter, $n = 2$)

Table 1: Comparison of IMC Controller Response

Parameters	IMC Controller					
	(λ = 0.9; n = 1)	(λ = 0.9; n = 2)	(λ = 0.5; n = 1)	(λ = 0.5; n = 2)	(λ = 0.1; n = 1)	(λ = 0.1; n = 2)
Overshoot (%)	Nil	Nil	0.505	Nil	0.437	0.026
Rise time (s)	2.966	4.247	1.649	2.425	0.330	0.492
Steady-state value	0.9900	0.9801	0.9904	0.9903	0.9909	0.9952
Steady-state error	0.0100	0.0199	0.0096	0.0097	0.0091	0.0048

Table 2: Comparison of Tuned PID Controller with IMC Controller

Parameter	Tuned PID Controller	IMC Controller ($\lambda = 0.1; n = 2$)
Overshoot (%)	**0.505**	**0.026**
Rise time (s)	**0.0289**	**0.492**
Steady-state value	**0.9925**	**0.9952**
Steady-state error	**0.0075**	**0.0048**

Conclusion

This research has successfully achieved its set objective and the conclusion (within the scope of the study) is as follows:

1. Comparatively, percentage overshoot with IMC controller is more desirably less than that of tuned-Matlab PID controller. The IMC Controller shows consistency in very low overshoot for varying time constant and order of filter.
2. Steady-state values for both IMC and tuned-Matlab PID controller are about equal. However, the IMC controller has a value much closer to the set-point.

Therefore, IMC controller would be the preferred controller where the robustness and accuracy of the dc motor speed control counts more than faster response.

References

Ahmad, M. A., Kishor, K. and Rai, P. (2014). Speed Control of a DC Motor using Controllers. *Automation, Control and Intelligent Systems*. [Online] 2 (6-1): 1-9. Available from: doi:10.11648/j.acis.s.2014020601.11 [Accessed 17th October 2018].

Al Nisa, S., Mathew, L. and Chatterji, S. (2013). Comparative Analysis of Speed Control of DC Motor using AI Technique. *International Journal of Engineering Research and Applications*, 3 (3): 1137-1146.

Garcia, C.E. and Morari, M. (1982). Internal Model Control. A Unifying Review and Some New Results. *Industrial & Engineering Chemistry Process Design and Development*, 21 (2): 308-323.

Kushwah, M. and Patra, A. (2014). Tuning PID controller for speed control of dc motor using soft computing techniques-a review. *Advance in Electronic and Electric Engineering*, 4 (2): 141 - 148.

Meshram, P.M. and Kanojiya, R.G. (2012). Tuning of PID controller using Ziegler -Nichols method for speed control of DC motor. In: IEEE 2012. International Conference On Advances In Engineering, Science And Management, ICAESM 2012, 30 - 31 March 2012, IEEE. pp. 117 – 122.

Moradi, S. Y. & Saeedi, E., 2016. Controling DC Motor Position, Using PID Controller Made by PIC

Microcontroller. *ZANCO Journal of Pure and Applied Sciences,* 28(2), pp. 82 - 89.

INTERNATIONAL JOURNAL – ERT
VOL. 11 NO.5 JUN-2020 ISSN: 1074-4741

Singh, A.P., Narayan, U. and Verma, A. (2013). Speed Control of DC Motor using PID Controller based on Matlab. Innovative Systems Design and Engineering, 4 (6): 22 –28.

Sao, K. and Singh, D.K. (2015). A Review of Transient and Steady State Response of DC Motor Position Control. *International Journal of Engineering Research & Technology*, 4 (6): 756 – 762.

Saranya, M. and Pamela, D. (2012). A Real Time IMC Tuned PID Controller for DC Motor. *International Journal of Recent Technology and Engineering*, 1 (1): 65-70

Tripathi, N., Singh, R. and Yadav, R. (2015). Analysis of Speed Control of DC Motor –A review Study. *International Research Journal of Engineering and Technology*, 2 (8):1616-1621.

Urquizo, A. (2011). *PID controller overview*, image, viewed 2. [Online] Available from: http://commons.wikimedia.org/wiki/File:PID.svg [Accessed 27th September 2018].