Journal of Engineering Science and Technology
6th EURECA 2016 Special Issue May (2017) 43 - 57

HARDWARE SIMULATION OF A NEW ANTI-WINDUP PI CONTROL FOR MOTOR SPEED APPLICATION

THONG LING CHIAH, CHOON LIH HOO*, EDWIN CHIN YAU CHUNG

School of Engineering, Taylor's University, Taylor's Lakeside Campus,
No. 1 Jalan Taylor's, 47500, Subang Jaya, Selangor DE, Malaysia
*Corresponding Author: ChoonLih.Hoo@taylors.edu.my

Abstract

Proportional-Integral (PI) controller consists of both proportional and integral parameters which enable the controller to eliminate error in the response of a control system. However, PI controller always accompanied by long settling time and oscillation in its response due to the windup phenomenon. Windup occurs when the controller output transcended the limit of the system actuator and the control is under saturated state. The control system will unable to react and respond to the incoming error signal. This uncontrollable state introduces performance deterioration and even instability in the system. Many anti-windup controllers have been established to overcome windup such as the Steady-state Integral Proportional Integral Controller (SIPIC). SIPIC is a robust controller with tuning gain decoupling feature which was developed through the generic control template. The control template can be used to design any PI related controller. This research aims to develop another anti-windup PI controller that is able to lessen the theoretical treatment of the windup phenomenon from the control template. It was observed that the existing anti-windup methods tend to apply the adaptive control switching mechanism that requires switching control methods between non-saturation and saturation range whereas the proposed anti-windup PI controller does not require adaptive control. The speed control performance of the proposed anti-windup PI controller is compared with the conventional PI controller through hardware simulation using Scilab/Scicoslab version 4.4.1 software. The parameter specification was identified through system identification of a direct current motor. The proposed anti-windup PI controller shows better performance improving and eliminating the drastic high overshoot, settling time and rise time as compared to the conventional PI controller. Therefore, this research is able to contribute to the body of knowledge as an alternative controller in the industrial motor application.

Keywords: Anti-windup, PI controller, parameter tuning, steady state, speed control.

1. Introduction

Proportional-integral (PI) controller remain the widely used closed-loop controller among researchers in industrial application because PI controller is easy to implement and able to provide satisfying result where it has short settling time with no overshoot. PI controller is able to eliminate the errors or disturbance in a system but subjected to oscillating response, tremendous overshoot and prolonged settling time [1]. This unfavorable response is due to the windup phenomenon which can even cause speed response instability besides the high overshoot and long settling time [2, 3]. Windup happened when a control system operates in a nonlinear region in which the controller output transcended the limit of the system actuator. The system will be unable to continuously detect the error signal in the system and provide necessary changes in the control when the control system falls into this saturated state. Windup phenomenon will cause physical degradation or damage to the system and soon leading to system malfunction [4].

Anti-windup was commonly used in control system to resolve system performance and stability deterioration especially for linear systems due to windup [5]. Anti-windup control works in ensuring the control output to stay within the limit of the system input. The anti-windup controller allows the control state regains its linear or non-saturated control as soon as possible when falls into the saturation region. The control system is saturated when the control output exceed the limiter's range [5].

There is various type of anti-windup PI controllers proposed by control engineers. These controllers can be categorised to three main scheme type which is conditional integral (CI) scheme, tracking back calculation (TBC) scheme and integral state prediction (ISP) scheme. All controllers consist their very own mathematical equation and the construction of the components for the controller. It allows the controller to be presented in the form of a block diagram. Furthermore, the mathematical formula reduced the challenges in the theoretical treatment of the anti-windup phenomenon in PI controller.

In this paper, a new anti-windup PI controller for motor speed application is proposed. The existing anti-windup involves adaptive control which switches control operation when the control state crosses between linear range and saturation range whereas the proposed anti-windup PI controller does not require such switching mechanism. Switching in an operating system may cause system instability and affect the system performance as well. The proposed anti-windup PI controller is a robust controller without the need of switching and equipped with tuning gain decoupling feature which is able to prevent the weaknesses and system instability caused by adaptive control switching.

This paper will continue with the discussion on the existing anti-windup controller schemes in Section 2. Section 3 introduces the proposed new anti-windup controller for motor speed application. Section 4 shows the simulated results of the newly proposed anti-windup controller and followed by conclusion for the research in Section 5.

2. Comparison of Existing Anti-windup Controller Schemes

PI controllers are widely used nowadays for motor control system due to its simplicity in application. There are consequences for PI controller where the controller can result in a higher overshoot and longer settling time which may lead to system degradation [6]. Therefore, many researches on anti-windup PI controller were done in the past and three schemes of the anti-windup controller that are applied most often are summarised. They are the CI, TBC and ISP schemes. These schemes have a common behavior where they alter the integral control during saturation whereas it switches back to the conventional PI controller under non-saturation. Steady-state integral proportional integral controller (SIPIC) is a recent anti-windup controller which require no switching mechanism [7].

2.1. Conditional integral (CI) scheme

A few switching method was done in the past and the most common CI scheme is to turn off the integration mode of the controller saturation point. However, the integral action will be excluded if the control input reaches saturation together with proportional control being activated. The PI control is only effective below the saturation point. Therefore, the controller input must be within the limit of saturation in order to obtain the most effective performance [8-10].

Conditional integral scheme uses the saturation range and linear range of the controller as a trigger to switch the integral action on or off. It can be simplified below in Eq. (1) where $\dot{q}$, e, u and v denotes first derivative integral state, error signal, controller output and plant input respectively [11].

$$\dot{q} = \begin{cases} e & if\ u = v \\ 0 & if\ u \neq v \end{cases} \tag{1}$$

This method is able to produce non-overshoot performance. However, it is challenging in deciding and select the gain to fulfill performance of an anti-windup and large integral value when plant input and controller output are different [8, 11]. The block diagram of anti-windup controller using conditional integral scheme is shown in Fig. 1 where $\omega_r{}^*$, ω_r, s, k_p, k_i, T_L, k_T, J and B are denoted as set reference of motor rotational speed, motor rotational speed, Laplace variable, proportional gain, integral gain, external load, torque constant, moment of inertia of motor and viscous damping coefficient [9].

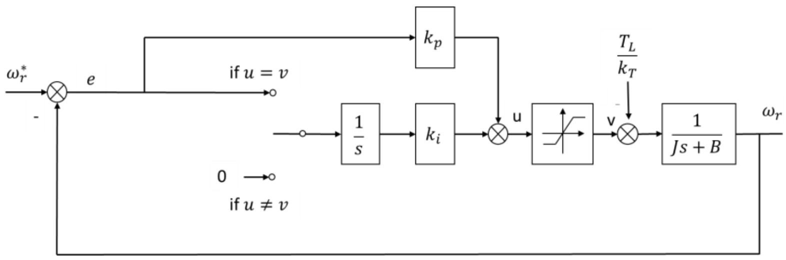

Fig. 1. Conditional integral scheme anti-windup controller [8].

2.2. Tracking back calculation (TBC) scheme

Tracking back calculation is also a method that is widely used by the control engineers to develop anti-windup controller in the past and Fig. 2 shows the block diagram of this scheme [10]. As shown in Fig. 2, the difference between non-saturated and saturated signal is identified and the error is being integrated in order to provide feedback for better control in the saturation range in Eq. (2):

$$\dot{q} = \begin{cases} e & if\ u = v \\ e - k_a(u - v) & if\ u \neq v \end{cases} \quad (2)$$

The advantage of this scheme is that a large range of gain, k_a can be chosen as it is able to limit the integrator of the controller. However, the error might occur when the chosen gain is too high and cause integrator to be reset by affecting the saturation of the controller. This scheme also bears the high risk of experiencing speed error when proportional integral (PI) is activated [10].

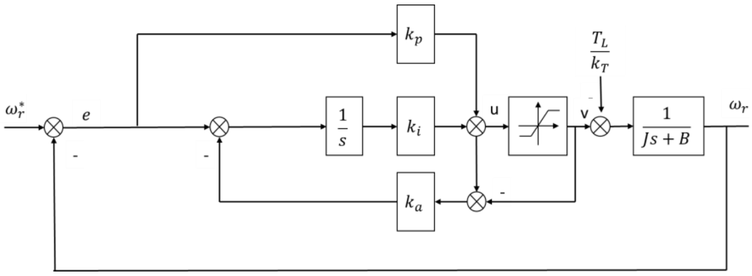

Fig. 2. Tracking back calculation scheme anti-windup controller [9].

2.3. Integral state prediction (ISP) scheme

The other anti-windup scheme is the integral state prediction (ISP) and the block diagram for its controller is shown in Fig. 3 [12-13]. The integrator value of this controller is restricted using a large gain in the feedback of the control so that the controller is in the linear range. This controller can function in two range which is linear and saturation range. The error from the output is connected directly to the integrator input to function as a feedback in linear range whereas the integral state is reset to the steady state according to the system prediction as shown in Fig. 3 at the state shown in Eq. (3) [11]:

$$\dot{q} = \begin{cases} e & if\ u = v \\ \omega_i(q_{ss} - q) & if\ u \neq v \end{cases} \quad (3)$$

where ω_i, q_{ss} and τ_m denotes the positive parameter of the low pass filter, the integral value at steady state and time constant.

Integral state loading time plays a significant role in allowing the controller to achieve the desired performance outcome. The integral state loading time is required to be a step ahead of the system dynamic so that the performance can be maintained at the top desired point [11, 13].

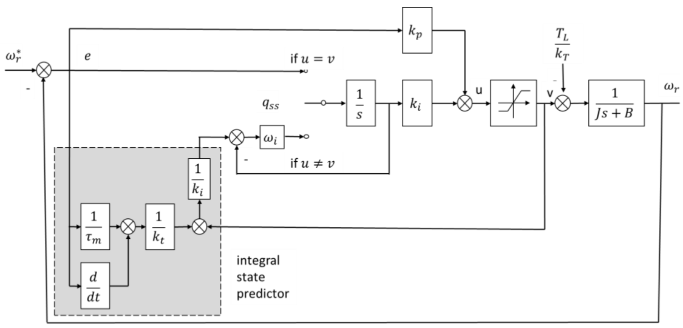

Fig. 3. Integral state prediction scheme anti-windup controller [12].

2.4. Comparison of anti-windup schemes

Table 1 shows the advantages and disadvantages of the existing anti-windup schemes. The existing anti-windup controllers have their very own integral control. The existing anti-windup controllers are applying the adaptive control which switches for the controller back to conventional PI controller under the unsaturated state. The functionality of the existing anti-windup controllers are limited only to saturation state and switching schemes at different states will eventually cause the system to be unstable.

Table 1. Comparison of anti-windup schemes.

Controller Scheme	Switching	Advantage	Disadvantage
Conditional Integration (CI)	Yes	• No overshoot speed response performance. • Effective when control input within saturation limit.	• Integral will be large when the value of controller output is different from plant output. • Difficult to choose value of gain.
Tracking Back Calculation (TBC)	Yes	• Large range of anti-windup gain alteration. • Improve overshoot.	• Error will occur if the gain is too big. • High risk of speed error.
Integral State Prediction (ISP)	Yes	• Functioning well in both linear and saturation range. • Steady state value is predicted to prevent state change.	• Integrator value is limited by feeding control back into linear state. • Derivation function is constraining the band-width of low-pass filter.

2.5. Steady-state integral proportional integral controller (SIPIC)

SIPIC is a robust controller which is able to overcome the instability caused by the adaptive control switching mechanism of the other anti-windup controllers which switches back to PI control under non-saturated state stated in Table 1 [7]. SIPIC has the tuning gain decoupling feature which was developed through the generic control template stated in Eq. (4) [7, 14].

$$\frac{q_{ss}}{s} - Q(s) = As^n f(s) + C \qquad (4)$$

where *A* and *C* indicate constant value, *f(s)* as the Laplace function, *Q(s)* as the Laplace form of integral state and *n* is a non-negative value [14].

3. Proposed Anti-windup PI Controller

A new anti-windup PI controller is proposed according to the generic control template similar to SIPIC which is able to produce a satisfying result with no overshoot. The proposed anti-windup PI controller performance are validated in motor speed control. A conventional PID controller is composed by Eq. (5).

$$u = k_p e + k_i q + k_d \dot{e} \quad (5)$$

Equation (5) shows a combination of proportional, integral and derivative parameters in a PID controller. u, k_p, e, k_i, q, k_d *and* $\dot{e}$ denotes controller output, proportional gain, error, integral gain, integral state, derivative gain and derivative error. A speed motor is usually constructed by controlling the inner current where the dynamic of the motor is in view of the loading effect shown in Eq. (6) [7]. T_l, ω_r, τ_m and v are denoted as external torque or disturbance, rotational motor speed, time constant and plant input respectively.

$$\frac{d\omega_r}{dt} = -\frac{\omega_r}{\tau_m} + k_t v - T_l \quad (6)$$

The difference between the current motor speed and the speed at set reference, $\omega_r{}^*$ shows the error in the closed loop system shown in Eq. (7):

$$e = \omega_r{}^* - \omega_r \quad (7)$$

The speed of motor, ω_r remain unchanged, error is excluded and plant input is $k_i q_{ss}$ at steady state where q_{ss} denotes integral state at steady state. Therefore, $\frac{d\omega_r}{dt} = 0$ which direct Eqs. (6) to (8) and rearranged as Eq. (9) to show $k_i q_{ss}$.

$$0 = -\frac{\omega_r{}^*}{\tau_m} + k_t k_i q_{ss} - T_l \quad (8)$$

$$k_i q_{ss} = \frac{\omega_r{}^*}{\tau_m k_t} + \frac{T_l}{k_t} \quad (9)$$

According to Hoo [14], the error (e) and its first derivative ($\dot{e}$) will be zero at steady state whereas q will be at steady state, q_{ss}. The error of the system or disturbance that occurs in the system can be simplified as (10) [15-17]:

$$-\dot{e} = \left(\frac{1}{\tau_m} + k_t k_p\right) e + k_t k_i q - \frac{\omega_r{}^*}{\tau_m} - T_l \quad (10)$$

where τ_m, $\omega_r{}^*$, T_l denotes the time constant, the rotational speed of motor at set reference. Followed by the steady-state error shown in Eq. (11):

$$E(s) = \frac{e(0) + k_t k_i\left(\frac{q_{ss}}{s} - Q(s)\right)}{\left(s + \frac{1}{\tau_m} + k_t k_p\right)} \quad (11)$$

In order to eliminate windup in a control system, the steady state error should be as low as possible or it should not exist [14]. When the steady-state error is assumed to be zero, the equation is written as Eq. (12) by limiting $s \to 0$.

$$\lim_{s\to 0} k_t k_i s \left[\frac{q_{ss}}{s} - Q(s)\right] = 0 \quad (12)$$

The function in bracket leads to a generic form of Eq. (4) as proposed by Hoo [14]. The possible function for Eq. (4) is described in Eqs. (13) and (14) shows the application of the integral component in the controller. The proposed controller with the integral component (9) is shown in Fig. 4 where k_t, J and B denotes the torque constant, motor's moment of inertia and viscous damping coefficient. The proposed controller was simulated using Scilab/Scicoslab version 4.4.1 and it shows satisfying result where the control can reach steady state under any circumstances and it took a shorter time to reach the steady state without high overshoot.

$$\frac{q_{ss}}{s} - Q(s) = \frac{(sQ(s) - q(0) + E(s))}{k_i} \quad (13)$$

$$k_i(q_{ss} - q) = \dot{q} + e \quad (14)$$

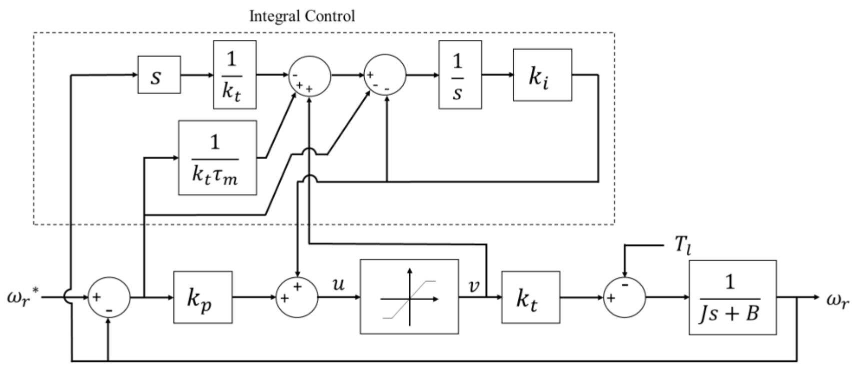

Fig. 4. Block diagram of the proposed anti-windup PI controller.

Characteristic analysis

From Eq. (13), the integral component of the proposed PI controller can be presented by Eq. (15) based on Eq. (13). The error equation of the proposed PI controller can be obtained by substituting Eq. (15) into Eq. (11) which leads to Eq. (16). The steady state error for the proposed anti-windup controller appeared to be zero when *s* approaches zero based on Eq. (12). The proposed PI controller does not behave like the existing anti-windup schemes which switching back to conventional PI during linear state. This allows the proposed anti-windup PI controller to be free from the disadvantageous effect of adaptive control switching mechanism.

$$Q(s) = \frac{q(0) + k_i \frac{q_{ss}}{s} - E(s)}{s + k_i} \quad (15)$$

$$E(s) = \frac{e(0)(s+k_i) + k_t k_i \left(q_{ss} - q(0)\right)}{(s+k_i)\left(s + \frac{1}{\tau_m} + k_t k_p\right) - k_t k_i} \quad (16)$$

According to Eq. (16), the tuning parameters (k_p and k_i) are separated into distinct poles and can be tuned to desired performance with no overshoot while remaining zero steady state error. The proposed anti-windup PI controller is compared with the conventional PI controller with different tuning parameters under different speed. The results are shown in Section 4.

4. Comparison of Simulated Result for Speed Control

In order to verify the concept in the previous section, the performances of the proposed anti-windup PI controller is compared with the conventional PI controller scheme using software Scilab/Scicoslab version 4.4.1. Figure 5 illustrates the hardware simulation block diagram incorporated with the proposed controller in Fig. 4. The comparison of the controllers was done with different motor speed under different loading conditions. The specification for loading condition is listed in Table 2. The parameters for DC motor speed are set based on the values listed in Table 3 in order to obtain a stable performance. The hardware simulation requires a setup of DC servo motor, motor encoder and the host computer with Scilab/Scicoslab 4.4.1 software and Real-Time Application Interface (RTAI) installed under Linux. The details of these setup components are stated in Tables 4 and 5. These specifications are calibrated according to the administrative and technical requirements of ANSI/ESD S20.20 and EN61340-5-1 electrostatics control standards as testing standards for the experimental machhine. The hardware simulation was done for step input speed for ±50 rad/s and ±100 rad/s with a different set of tuning parameters, k_p and k_i as shown in Tables 6-8.

Table 2. Specification for load condition.

Case	Characteristics	Values
No load	Material	-
	Moment of inertia	0 kgm^2
Load 1	Material	Mild steel black plating
	Moment of inertia	8.63×10^{-5} kgm^2
Load 2	Material	Aluminum plating
	Moment of inertia	2.83×10^{-5} kgm^2

Table 3. Parameters for DC motor speed test.

Characteristics	Values
Viscous damping coefficient, *B*	2.12×10^{-4} kg m^2/s
Inductance, *L*	0.005 H
Moment of inertia of motor, *J*	2.14×10^{-5} kg m^2
Torque constant, k_t	0.09 Nm/A
Back-emf constant, k_m	0.09 Nm/A
Efficiency, η	0.8
Resistance, *R*	7.8 Ω

Table 4. Specification for DC servo motor.

Characteristics	Values
Maximum supply voltage	40 Vdc
Maximum continuous torque	14 Ncm
Maximum peak torque	36 Ncm
Motor voltage constant	10.3 V at 1000 rpm
Motor torque constant	9.0 Ncm/A
Mechanical time constant	20 ms
Rotor inertia	0.214 kg cm
Terminal resistance	7.8 Ohms
Rated speed	1600 rpm
No load speed	2600 rpm @ 24 Vdc
Rated torque	12 Ncm
Peak torque	27 Ncm

Table 5. Specification for motor encoder.

Characteristics	Values
Number of lines	500 lines (2000 @ quad)
Number of channel	2 channel (A, B)
Output pulse	5 Vdc
Supply voltage	± 5 Vdc

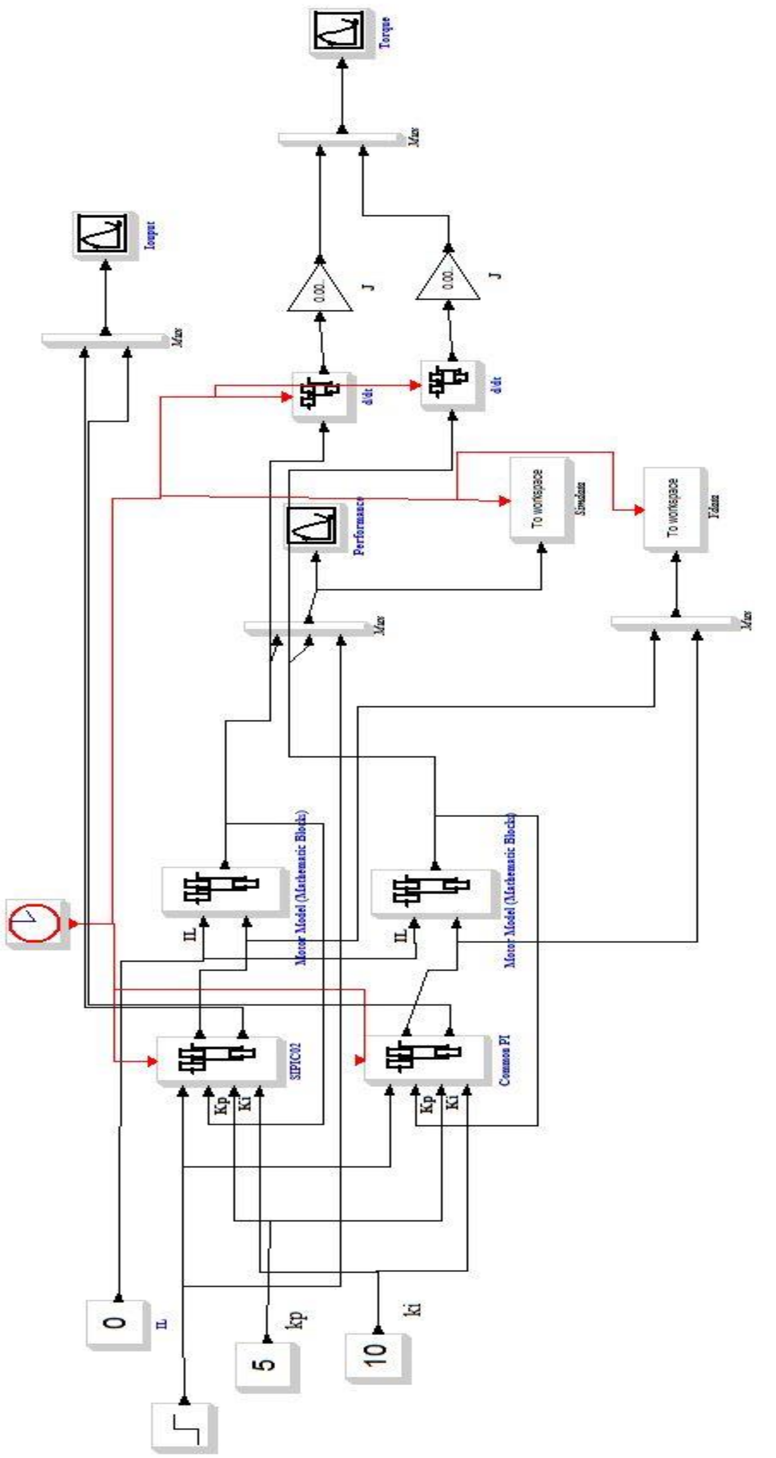

Fig. 5. Block diagram for simulation in Scilab/Scicoslab software.

4.1. Speed control under no load condition

The conventional PI controller and the proposed anti-windup PI controller were simulated at the motor speed of 100 rad/s as shown in Fig. 6. Conventional PI controller is faster integral speed as compared with the proposed anti-windup PI controller which causes overshoot phenomenon to occur in the performance. The proposed anti-windup controller has no overshoot but it has a longer settling time as compared to conventional PI controller.

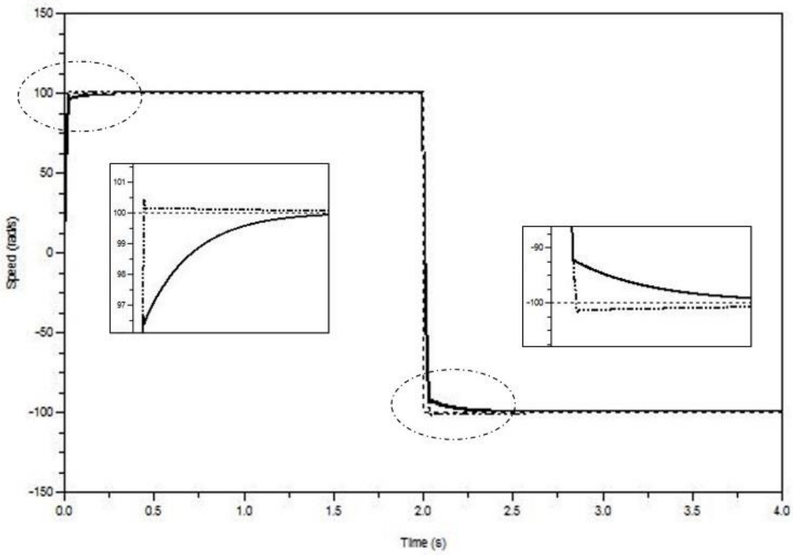

Fig. 6. Comparison of speed control for conventional PI controller (dash-dot line) and proposed anti-windup PI controller (solid line) with changing step input under no load condition (speed = 100 rad/s, k_p = 5, k_i = 10).

The simulation of motor speed control was done with a different set of tuning parameters. The result of the simulation was summarised in Table 6 in terms of the rising time of the signal, settling time and percentage of the overshoot for conventional PI and proposed anti-windup PI controller. The comparison of conventional PI and the proposed anti-windup PI controller was made for the speed of 50 rad/s and 100 rad/s under no load condition. According to Table 6, increasing the value of parameter k_p and k_i is able to reduce the rise time and settling time for both the controllers. However, the rise time and settling time of the proposed anti-windup PI controller are higher than the conventional PI controller whenever k_p is 1. Both rise time and settling time are observed to be the same as k_p and k_i increases. This is because the performance of the controller has reached the lowest boundary of rise time and settling time when k_p is tuned to 5 for both speeds of 50 rad/s and 100 rad/s. The increment of the parameter k_p and k_i is no longer giving any significant effect on rise time and settling time. However, the occurrence of overshoot still exists and the percentage of having an overshoot increases with the increment in k_i value for conventional PI controller. The proposed anti-windup PI controller does not show any overshoot performance due to the existence of the decoupling effect.

Table 6. Performance at 50 rad/s and 100 rad/s under no load condition.

Speed, ω (rad/s)	k_p	k_i	Rise Time		Settling Time		Overshoot (%)	
			PI	Proposed	PI	Proposed	PI	Proposed
50	1	1	0.012	0.521	0.742	2.427	0	0
	1	5	0.011	0.380	0.106	1.758	0	0
	1	10	0.010	0.337	0.016	1.032	0.161	0
	5	1	0.009	0.009	0.010	0.010	0	0
	5	5	0.009	0.009	0.010	0.010	0	0
	5	10	0.009	0.009	0.010	0.010	0.468	0
	10	1	0.009	0.009	0.010	0.010	0	0
	10	5	0.009	0.009	0.010	0.010	0.160	0
	10	10	0.009	0.009	0.010	0.010	0.464	0
100	1	1	0.025	0.916	0.690	2.341	0	0
	1	5	0.024	0.681	0.028	2.021	0	0
	1	10	0.023	0.567	0.026	1.262	0.635	0
	5	1	0.024	0.024	0.027	0.027	0	0
	5	5	0.024	0.024	0.026	0.026	0	0
	5	10	0.024	0.024	0.026	0.026	0.377	0
	10	1	0.024	0.024	0.027	0.027	0	0
	10	5	0.024	0.024	0.026	0.026	0	0
	10	10	0.024	0.024	0.026	0.026	0.339	0

4.2. Speed control with load condition

4.2.1. Loading with mild steel plating (Loading condition 1)

Figure 7 shows the comparison between the conventional PI controller and the proposed anti-windup PI controller at the motor speed of 100 rad/s under loading condition 1 (mild steel plating). The simulated rise time, settling time and overshoot percentage under loading condition 1 is tabulated in Table 7.

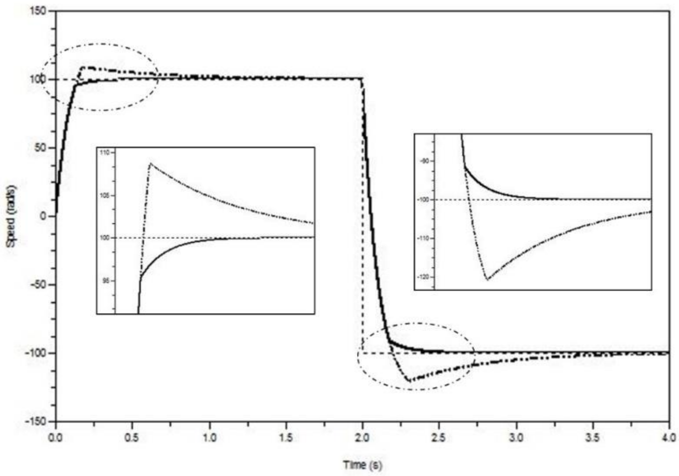

Fig. 7. Comparison of speed control for conventional PI controller (dash-dot line) and proposed anti-windup PI controller (solid line) with changing step input under loading condition 1 (speed = 100 rad/s, k_p = 5, k_i = 10).

Table 7. Performance at 50 rad/s and 100 rad/s under loading condition 1.

Speed, ω (rad/s)	k_p	k_i	**Rise Time**		**Settling Time**		**Overshoot (%)**	
			PI	Proposed	PI	Proposed	PI	Proposed
50	1	1	0.055	1.091	0.565	2.266	0	0
	1	5	0.045	0.799	0.051	2.001	0.820	0
	1	10	0.044	0.647	0.047	1.336	9.919	0
	5	1	0.044	0.044	0.047	0.047	0	0
	5	5	0.044	0.044	0.047	0.047	0.364	0
	5	10	0.044	0.044	0.047	0.047	2.696	0
	10	1	0.044	0.044	0.047	0.047	0	0
	10	5	0.044	0.044	0.047	0.047	0.827	0
	10	10	0.044	0.044	0.047	0.047	1.965	0
100	1	1	0.118	2.001	0.160	2.287	0	0
	1	5	0.118	1.630	0.130	2.124	13.409	0
	1	10	0.120	1.141	0.130	1.830	23.939	0
	5	1	0.118	0.118	0.130	0.130	0	0
	5	5	0.118	0.118	0.130	0.130	3.405	0
	5	10	0.118	0.118	0.130	0.130	8.736	0
	10	1	0.118	0.118	0.130	0.130	0	0
	10	5	0.118	0.118	0.130	0.130	1.837	0
	10	10	0.118	0.118	0.130	0.130	4.621	0

The rise time and settling time for both conventional PI controller and the proposed anti-windup PI controller are longer as compared to no load condition. In addition, the overshoot percentage of the conventional PI controller increases drastically as compared to no load condition. A small percentage of overshoot is observed at low k_p and k_i values. Conventional PI controller has faster integral speed as compared with the proposed anti-windup PI controller which causes overshoot phenomenon to occur in the performance. The proposed anti-windup controller has no overshoot but with a longer settling time as compared to conventional PI controller. The higher integral gain is the reason behind the changes which brought the system into saturation and proportional gain could not react to correct the error immediately. The proposed anti-windup controller still shows a satisfying result with no overshoot due to its decoupling feature.

4.2.2. Loading with aluminum plating (Loading condition 2)

The comparison between the conventional PI controller and the proposed anti-windup PI controller at the motor speed of 100 rad/s under loading condition 2 (aluminum plating) is shown in Fig. 8 and the result is tabulated in Table 8. The rise time, settling time and overshoot percentage is very similar to loading condition 1 case. The rise time and settling time for both conventional PI controller and the proposed anti-windup PI controller is shorter as compared to loading condition 1 case but longer when compared to no load condition. The overshoot percentage of the conventional PI controller increases with proportional gain and integral gain. A small percentage of overshoot is observed at low k_p and k_i values as well. According to Table 8, the integral speed of the conventional PI controller is faster than the proposed anti-windup PI controller which leads to overshoot phenomenon. The proposed anti-windup controller has no overshoot but has a longer settling time and the simulated performance is more satisfying compared to conventional PI controller.

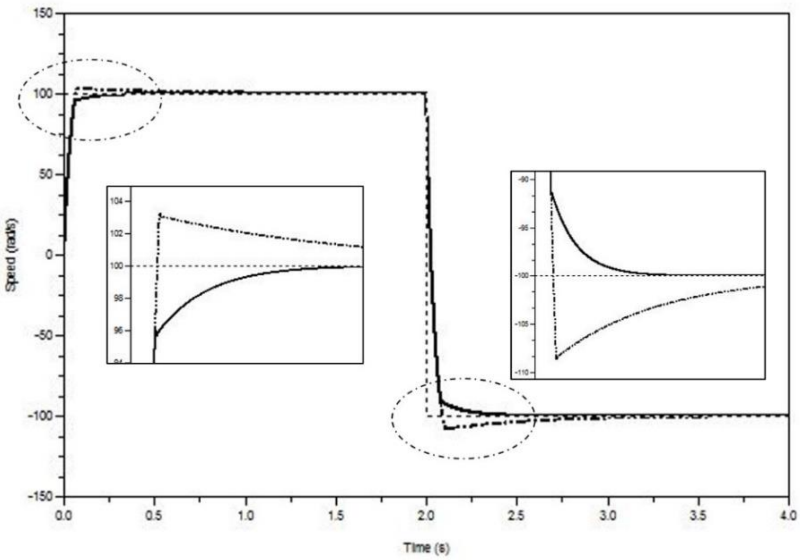

Fig. 8. Comparison of speed control for conventional PI controller (dash-dot line) and proposed anti-windup PI controller (solid line) with changing step input under loading condition 2 (speed = 100 rad/s, k_p = 5, k_i = 10).

Table 8. Performance at 50 rad/s and 100 rad/s under loading condition 2.

Speed, ω (rad/s)	k_p	k_i	**Rise Time**		**Settling Time**		**Overshoot (%)**	
			PI	Proposed	PI	Proposed	PI	Proposed
50	1	1	0.027	0.733	0.685	2.398	0	0
	1	5	0.022	0.549	0.031	1.925	0.155	0
	1	10	0.021	0.470	0.024	1.164	1.762	0
	5	1	0.021	0.021	0.022	0.022	0	0
	5	5	0.021	0.021	0.022	0.022	0.643	0
	5	10	0.021	0.021	0.022	0.022	1.272	0
	10	1	0.021	0.021	0.022	0.022	0	0
	10	5	0.021	0.021	0.022	0.022	0.449	0
	10	10	0.021	0.021	0.022	0.022	1.272	0
100	1	1	0.055	1.578	0.574	2.174	0	0
	1	5	0.055	1.094	0.060	2.103	1.918	0
	1	10	0.055	0.840	0.060	1.533	12.633	0
	5	1	0.055	0.055	0.060	0.060	0	0
	5	5	0.055	0.055	0.060	0.060	0.752	0
	5	10	0.055	0.055	0.060	0.060	3.058	0
	10	1	0.055	0.055	0.060	0.060	0	0
	10	5	0.055	0.055	0.060	0.060	0.417	0
	10	10	0.055	0.055	0.060	0.060	1.961	0

5. Conclusions

The proposed anti-windup controller shows a promising potential to contribute to the society by serving as an anti-windup controller for better motor speed control. The proposed anti-windup PI controller shows a better result regardless of the loading condition as compared to the conventional PI controller. The performance of this controller is promising where the integral control is able to reach steady state at both saturated and non-saturated state. The proposed anti-windup controller reaches the steady state without having overshoot in the speed control performance. However, the proposed anti-windup PI controller has extremely long rise time and settling time when the proportional gain, k_p is 1 as compared to the conventional PI controller. Therefore, future work will focus on reducing the rise time and settling time for low proportional gain, k_p value.

References

1. Gadoue, S.M.; Giaouris, D.; and Finch, J.W. (2009). Artificial intelligence-based speed control of DTC induction motor drives - A comparative study, *Electric Power Systems Research*, 79(1), 210-219.
2. Astrom, K.J.; and Hagglund, T. (1995). *PID controllers: Theory, design, and tuning*. Research Triangle Park, NC: ISA, 59-92.
3. Bahrani, B.; Kenzelmann, S.; and Rufer, A. (2011). Multivariable-PI-based DQ current control of voltage source converters with superior axes decoupling capability. *IEEE Transactions on Industrial Electronics*, 58(7), 3016-3026.
4. Wai, R.J.; Lee, J.D.; and Chuang, K.L. (2016). Real-time PID control strategy for Maglev transportation system via particle Swarm optimization. *IEEE Transactions on Industrial Electronics*, 58(2), 629-646.
5. Tarbouriech, S.; and Turner, M. (2009). Anti-windup design: An overview of some recent advances and open problems. *IET Control Theory and Application*, 3(1), 1-19.
6. Al-Abbas, I.K.; and Issa, R. (2012). Overshoot elimination in cascade control of separately excited DC motors. *European Journal of Scientific Research*, 85(1), 98-104.
7. Hoo, C.L.; Haris, S.M.; Chung, E.C.Y.; and Mohamed, N.A.N. (2015). Steady-state integral controller for PI motor speed controllers. *Journal of Power Electronics*, 15(1), 177-189.
8. Hanus, R.; Kinnaert, M.; and Henrotte, J.L. (1987). Conditioning technique, a general anti-windup and bumpless transfer method. *Automatica*, 23(6), 729-739.
9. Seok, J.K.; Kim, K.T.; and Lee, D.C. (2007). Automatic mode switching of P/PI speed control for industry servo drives using online spectrum analysis of torque command. *IEEE Transactions on Industrial Electronics,* 54(5), 2642-2647.
10. Choi, J.W; and Lee, S.C. (2009). Antiwindup strategy for pi-type speed controller. *IEEE Transactions on Industrial Electronics*, 56(6), 2039-2046.
11. Li, X.L.; Park, J.G.; and Shin, H.B. (2011). Comparison and Evaluation of Anti-Windup PI Controllers. *Journal of Power Electronics*, 11(1), 45-50.

12. Park, J.G.; Chung, J.H.; and Shin, H.B. (2002). Anti-windup integral-proportional controller for variable-speed motor drives. *Journal of Power Electronics*, 2(2), 130-138.
13. Ohishi, K.; Hayasaka, E.; Nagano, T.; Harakawa, M.; and Kanmachi, T. (2006). High-performance speed servo system considering voltage saturation of a vector-controlled induction motor. *IEEE Transactions on Industrial Electronics*, 53(3), 795-802.
14. Hoo, C.L.; Haris, S.M.; Chung, E.C.Y.; and Mohamed, N.A.N. (2015). The generalization and decoupling mode of PI-based control: Theoretical approach. *ICIC Express Letters*, 9(7), 1991-1996.
15. Hoo, C.L.; Haris, S.M.; Chung, E.C.Y.; and Mohamed, N.A.N. (2015). Steady-state integral proportional integral controller for PI motor speed controllers: A theoretical approach. *ICIC Express Letters*, 9(6), 1777-1782.
16. Hoo, C.L.; Haris, S.M.; Chung, E.C.Y.; and Mohamed, N.A.N. (2015). The boundary integral-error state, q(0)-e(0) plane for PI control. *ICIC Express Letters*, 9(7), 1797-1802.
17. Hoo, C.L.; Haris, S.M.; Chung, E.C.Y.; and Mohamed, N.A.N. (2016). New integral antiwindup scheme for PI motor speed control. *Asian Journal of Control*, 18(3), 1-18.