Evaluation of SIPIC01 and SIPIC02 on Motor Speed Control

Kah Kit Wong¹, Choon Lih Hoo², Mohd Hardie Hidayat Mohyi¹,*
¹School of Engineering, Taylor’s University, Malaysia
²School of Engineering and Physical Sciences, Heriot-Watt University, Malaysia

Abstract. Due to its simplicity, Proportional-Integral (PI) controller still remains as the widely used controller for motor speed control system. However, PI controller exhibits windup phenomenon when the motor operates in a saturated state, which may cause degradation to the control system. In order to overcome the windup phenomenon, many researches have introduced various types of anti-windup methods such as the Conditioning Technique (CI), Tracking Back Calculation (TBC), Integral State Prediction (ISP), Steady-state Integral Proportional Integral Controller-01 (SIPIC01) and Steady-state Integral Proportional Integral Controller-02 (SIPIC02). These are anti-windup techniques with integral control switching mechanism, coupling of proportional gain, $k_p$, and integral gain, $k_i$. Due to the coupled $k_p$ and $k_i$, tuning motor performance is a difficult task with short settling time without experiencing overshoot. SIPIC01 and SIPIC02 are robust anti-windup methods without a switching mechanism and exhibit decoupling feature. SIPIC01 and SIPIC02 have shown better dynamic performance compared to CI, TBC and ISP. However, SIPIC01 has not been compared to SIPIC02 in terms of their decoupling effect flexibility and dynamic performance. The decoupling effect was verified using MATLAB simulation, while the performance analysis was verified through hardware simulation and testing by using Scilab. The results obtained from the simulation showed that both SIPIC01 and SIPIC02 consist of decoupling features that allow a performance with coexistence of zero or minimum overshoot with short settling time. However, SIPIC02 consists of longer rise and settling time as compared to SIPIC01. Therefore, it can be concluded that SIPIC01 is better than SIPIC02 in term of dynamic performance.

1. Introduction

A control system is one of the important components in a system which is responsible in producing desired output. Control system comprises of open-loop and closed-loop systems.

Open-loop control system is a simple control system without any feedback loop in which the input is totally independent from the output of a system [1]. Other than its
simplicity, open-loop control system is generally stable and economical. However, the output of the open-loop control system is inaccurate and unreliable. This is due to the system disability to auto-correct any error signals since there is no feedback loop. In contrast, a closed-loop control system is a control system with feedback loop ability [1]. This type of control system contains the error auto-correction operation that gives a more accurate and reliable output. However, the design of a closed-loop control system is complicated.

In modern control, Artificial Intelligence (AI) is a type of closed-loop control system that utilises the complex computation mechanism to imitate human thinking and thought process. Fuzzy logic controller (FLC) [2], artificial neural network (ANN) [3], and genetic algorithm (GA) [4] are some of the commonly discussed AI controls which are robust with good dynamic performance in motor speed control. Due to the classical Proportional Integral (PI) controller ease of use and simplicity in applications, it is still remain as the current widespread controller choice for motor speed control system.

However, PI controller experiences windup which normally happens when the output of the controller exceed the input limit of the plant which caused the system to operates in a saturated state [5]. This phenomenon will lead to poor control response such as increased overshoot, long settling time and system instability.

In order to improve the performance of the PI controller, the windup phenomenon should be eliminated. There are many anti-windup techniques developed to avoid windup phenomenon such as the Conditional Integration (CI), Tracking Back Calculation (TBC), Integral State Prediction (ISP), Steady-state Integral Pi Controller-01 (SIPIC01) and Steady-state Integral Pi Controller-02 (SIPIC02) [5]. Currently, both SIPIC01 and SIPIC02 have been compared with the existing anti-windup PI controllers and the result shows that both of them perform better in term of dynamic performance on motor speed control [6]. However, the evaluation between the two has yet to be studied. Hence, the dynamic performance between SIPIC01 and SIPIC02 is investigated in this study. This work serves to expand the knowledge of the field of anti-windup PI controller.

2. Types of Anti-Windup PI Controller

In order to overcome the windup phenomenon, many researches introduced different types of anti-windup schemes. Out of the schemes available; CI, TBC and ISP appeared to be the choice among researchers. Though they operate differently by altering their integral control component in the saturated state, they have the similarity that resume to the conventional PI controller under unsaturated state. Consequently, they experience difficulty in tuning when operating as PI controller due to their coupled tuning gains [5].

CI, TBC and ISP have similar integral control under unsaturated state but consist of different integral control under saturated state. Table 1 shows the integral control of the respective anti-windup method under different states. Table 2 shows the advantages and disadvantages of these anti-windup methods [5]. The $e$, $k_p$, $u$, $v$, $ω_i$, $q_{ss}$ and $q$ shown in Table 1 denote the input error, anti-windup gain, controller output, plant input, positive parameter of the low-pass filter, steady state integral component and integral state respectively.

Table 1. Comparison of integral control of anti-windup schemes under different states

| Controllers | Integral Control Under Unsaturated State, $u = v$ | Integral Control Under Saturated State, $u ≠ v$ | Switching Mechanism |
|-------------|-----------------------------------------------|-----------------------------------------------|-------------------|
| Conditional Integration (CI) | $e$ | $0$ | Yes |
The disadvantage of these anti-windup methods [5]. The respective anti-windup method under different states. Table 2 shows the advantages and disadvantages of anti-windup schemes. Out of the schemes available; CI, TBC and ISP appeared to be the choice among researchers. Though they operate differently by altering their integral component in the saturated state, they have the similarity that resume to the CI, TBC and ISP have similar integral control under unsaturated state. Consequently, they experience difficulty in the evaluation between the two has yet to be studied. Hence, the dynamic performance of the controller exceed the input limit of the plant which caused the system to operate in a saturated state [5]. This phenomenon will lead to poor control response such as increased overshoot, long settling time and system instability.

There are many anti-windup techniques developed to avoid windup and the phenomenon such as the Conditional Integration (CI), Tracking Back Calculation (TBC), Integral State Prediction (ISP). The main purpose of generalisation of PI controllers is to develop a generic expression which can be used to represent all the PI-related control methods. According to [7], the function \( k_i k_q (q_{ss} - q) \) can generalise any PI-related controller and the general equation is shown in Equation (1) where \( l, u, v \) and \( w \) are natural numbers while \( A_l, B_u \) and \( C \) are integers.

\[
k_i k_q (q_{ss} - q) = \sum_{l=0}^{l=v} A_l q^{(l)} + \sum_{u=0}^{u=w} B_u e^{(u)} + C
\]

(1)

The \( k_i, k_q, q_{ss}, q \) and \( e \) represent the integral gain, external load constant, steady state integral component, integral state and input error respectively while the \( l \) and \( u \) denote the number of coefficient of \( A \) and \( B \) for \( q \) and \( e \) respectively. Both \( l \) and \( u \) also denote the number of order for \( q \) and \( e \) respectively while \( C \) represent any constant. Each term in the function of \( k_i k_q (q_{ss} - q) \) carries certain characteristic of the control system. For example, the existence of the constant \( C \) will give a pole at the origin which will result in steady state error [7]. In order to obtain a zero steady state error control performance, Equation (2) must be met.

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

(2)

Following this condition (2), the expression of \( \frac{q_{ss}}{s} - Q(s) \) needs to be reduced into a function of \( s \) which consists of no pole at the origin [7]. However, with the presence of constant \( C \) in the function of \( k_i k_q (q_{ss} - q) \), Equation (2) will never be satisfied and the steady state error cannot be eliminated.

Besides, the existence of error term, \( e \) will lead to the coupling of \( k_p \) and \( k_i \) parameter which increase the difficulty to obtain a performance of zero overshoot and short settling time. According to [7], the absence of the error term will decouple the \( k_p \) and \( k_i \) which allows a performance with coexistence of zero overshoot and short settling time.

| Controllers                  | Advantages                                      | Disadvantages                           |
|------------------------------|------------------------------------------------|-----------------------------------------|
| Conditional Integration (CI) | • No overshoot                                  | • Integral value is large when the controller output differ from the plant input. |
|                              | • Integral value is large when the controller output differ from the plant input. | • Difficulty to choose value of gain   |
|                              | • Large value of anti-windup gain, \( k_i \) is allowed to be chosen | • Consists of coupling effect          |
|                              | • Improve overshoot                             |                                         |
| Tracking Back Calculation (TBC) | Improving overshoot, \( k_i \) is allowed to be chosen | • Consists of coupling effect          |
| Integral State Prediction (ISP) | Prevent state change                           | • Integrator value is limited.         |
|                              | • Steady state value is predicted to prevent state change. | • Consists of coupling effect          |

3. Generalisation of PI Controllers

The main purpose of generalisation of PI controllers is to develop a generic expression which can be used to represent all the PI-related control methods. According to [7], the function \( k_i k_q (q_{ss} - q) \) can generalise any PI-related controller and the general equation is shown in Equation (1) where \( l, u, v \) and \( w \) are natural numbers while \( A_l, B_u \) and \( C \) are integers.
4. Steady-state Integral Proportional Integral Controller 01 (SIPIC01)

According to the characteristic of the term in the general Equation (1) stated in section 3, Hoo et al. proposed a structure of the controller as shown in Equation (3) where $A_1$ can be any constant [8], [9].

$$k_ik_t(q_{ss} - q) = A_1q\dot{q}$$

(3)

In order to develop a controller with no steady state error and decoupling effect, $A_1$ can be equal to $k_i$ as $A_1$ can be any constant. By substituting $A_1 = k_i$ into Equation (3), Equation (4) which describes the integral control of SIPIC01 will be formed. The Laplace transformation of Equation (4), with $Q(s)$ the subject is shown in Equation (5).

$$k_i(q_{ss} - q) = \dot{q}$$

(4)

$$Q(s) = \frac{q(0) + k_iq_{ss}}{s + k_i}$$

(5)

Equation (6) is a second order error dynamic equation (EDE) formed from a common first order system plant, $P(s)$ controlled by a PI controller as shown in Figure 1, where $V(s)$ and $T_i(s)$ are the control output and external load respectively. The error equation of SIPIC01 (7) can be derived through the substitution of Equation (5) into Equation (6). Equation (8) corresponds to the damping ratio of SIPIC01, which is formed from the denominator of Equation (7) by comparing with the general characteristic equation, $s^2 + 2\zeta\omega s + \omega^2$, where $\zeta$ and $\omega$ are the damping ratio and natural frequency respectively.

$$E(s) = \frac{e(0)(b + k_dk_t) + k_ik_t\left(\frac{q_{ss}}{s}-q(s)\right)}{(b + k_dk_t)s + \frac{a}{\tau} + k_pk_t}$$

(6)

$$E(s) = \frac{e(0)b}{(b + \frac{a}{\tau} + k_pk_t)} + \frac{k_ik_t(q_{ss} - q(0))}{(s + k_i)(\frac{b}{s} + \frac{a}{\tau} + k_pk_t)}$$

(7)

$$\zeta = \frac{a + bk_pk_tk_i}{2\sqrt{b}\sqrt{ak_t + bk_pk_tk_i}}$$

(8)

As shown in Equation (7), there is no pole at the origin and this indicates that there is no steady state error. Besides, only $k_p$ controls the $e(0)$ while $k_p$ and $k_i$ control the $k_ik_t(q_{ss} - q(0))$ in two separate terms. Hence, it can be concluded that SIPIC01 consists of decoupling effect which allows the controller to obtain short settling and rise time without experiencing overshoot.

5. Steady-state Integral Proportional Integral Controller 02 (SIPIC02)
Chiah et al. [6] proposed another anti wind-up PI controller based on general Equation (1) with the structure of the controller shown in Equation (9) where $A_1$ and $A_2$ can be any constant.

$$k_i k_e (q_{ss} - q) = A_1 \dot{q} + A_2 e$$

(9)

One of the possible option for $A_1$ and $A_2$ is $k_i$. By substituting $A_1 = A_2 = k_i$ into Equation (9), the integral control of the SIPIC02 can be obtained (10). The Laplace transformation of Equation (10), with $Q(s)$ the subject is shown in Equation (11).

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

(10)

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

(11)

From Equation (11) and Equation (6) with zero derivative gain, the error equation of SIPIC02 is shown in Equation (12). Equation (13) is the damping ratio of SIPIC02 obtained from the denominator of Equation (12) by comparing with the characteristic equation.

$$E(s) = \frac{e(0) p}{f} (s+k_i) + k_i k_e (q_{ss}-q(0))$$

\[
(12)
\]

$$\zeta = \frac{a+f k_p k_e + b k_i}{2 \sqrt{E(a k_i+f k_p k_e) - f k_i k_e}}$$

(13)

As shown in Equation (12), there is no pole at the origin and this indicates that there is no steady state error. Theoretically, SIPIC02 consists of coupling effect which will prevent the controller from achieving a performance of zero overshoot and short settling and rise time. This is because the $k_i k_e$ in Equation (12) does not allow distinctive separable factors in the denominator. The coexistence of $k_p$ and $k_i$ in the pole hinder the decoupling. However, according to Chiah et al. [6], SIPIC02 shows a promising performance with short settling and rise time while maintaining low or zero overshoot.

6. Simulation and Experimental Testing

The theoretical analysis of SIPIC01 and SIPIC02 done so far will be verified in this section by conducting simulation and hardware testing. The simulation for decoupling effect will be described in section 6.1 while the hardware simulation and experiment testing for speed control performance will be described in section 6.2.

6.1 Simulation for Decoupling Effect

MATLAB/Simulink software was used in to verify the decoupling effect of SIPIC01 and SIPIC02 through simulation for a second order EDE system. The block diagram of the simulation for decoupling mode is illustrated in Figure 2. For simplicity, the
other parameters are set to unity, the input is set as 1000 rpm and the system plant is set as \( P(s) = \frac{1}{s+1} \). The simulation is conducted under two conditions which are with and without load. For simulation with load, the load, \( L \) is set as 200. The simulation for both conditions are repeated for input equal to 2000 rpm.

![Fig 1. The block diagram for general control system](image)

![Fig. 2. The block diagram of the simulation for decoupling mode](image)

### 6.2 Simulation and Experiment Testing for Control Performance

The simulation testing of SIPIC01 and SIPIC02 for control performance is done through Scilab/Scicoslab. They are tested under three cases which are no load, load 1 and load 2. Load 1 is a mild steel black plating with moment of inertia of \( 8.63 \times 10^{-5} \) kgm\(^2\) whereas the material for load 2 is an aluminium plating with moment of inertia of \( 2.83 \times 10^{-5} \) kgm\(^2\). Figure 3 shows the block diagram used for the simulation testing of SIPIC01 and SIPIC02. In order to ensure that the model is reliable, system identification is performed by using Scilab/Scicoslab. The simulation result is later compared with the real hardware testing result.

For the experimental testing, the setup for the DC servo motor speed control is shown in Figure 4. The setup includes Scilab/Scicoslab, DC servo motor, host controller
computer, load tuner, motor encoder, power amplifier, terminal system unit and real time interfacing platform data acquisition. Similar to the simulation, the experimental testing for SIPIC01 and SIPIC02 are also tested under the three cases. For each case, the testing was done for two speeds input, which are 50 rad/s and 100 rad/s. The specification of the DC motor is shown in Table 3.

![Fig. 1. The block diagram for general control system](image1)

![Fig. 2. The block diagram of the simulation for decoupling mode](image2)

### Fig. 3. The block diagram for the simulation testing

![Fig. 4. Setup for the experimental testing](image3)

### Fig. 4. Setup for the experimental testing

#### Table 3. Specification of the DC 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 1000rpm           |
Motor torque constant  9.0 Ncm/A
Mechanical time constant  20ms
Rotor inertia  0.214 kgcm
Terminal resistance  7.8 Ohms
Rated speed  1600 rpm
No load speed  2600 rpm @ 24 Vdc
Rated torque  12 Ncm
Peak torque  27 Ncm

7. Simulation Result

7.1 Decoupling Effect

Figures 5 and 6 show the result of simulation for decoupling effect for $k_p = 1$, $k_i = 20$ at 1000 rpm under no load and loading condition respectively. PI has the shortest rise time and greatest overshoot, while both SIPIC01 and SIPIC02 have no overshoot. The response of simulation for decoupling effect for PI, SIPIC01 and SIPIC02 at 1000 rpm and 2000 rpm under no load condition is shown in Table 4. The response of simulation for decoupling effect under loaded condition is not shown but will be discussed in this section.

As shown in Table 4, the rise time and settling time for PI reduced with increasing $k_p$ or $k_i$ gains. However, the overshoot percentage for PI increased when $k_i$ is increased. This observation can be explained based on the damping ratio equation (14). By referring to Equation (14), the damping ratio of PI is directly proportional to $k_p$ but inversely proportional to $k_i$. The damping ratio of PI increases with increasing $k_p$ but conversely with increasing $k_i$.

![Fig. 5. Simulation for decoupling effect for $k_p = 1$, $k_i = 20$ at 1000 rpm under no load condition.](image-url)
Fig. 5. Simulation for decoupling effect for $k_p = 1$, $k_i = 20$ at 1000 rpm under no load condition.

Fig. 6. Simulation for decoupling effect for $k_p = 1$, $k_i = 20$ at 1000 rpm under loaded condition.
The system is said to experience overshoot when the damping ratio is $< 1$. The lower the damping ratio, the higher the percentage of overshoot.

$$\zeta = \frac{a + f k_p k_i}{2b \sqrt{k_i k_t}}$$  \hspace{1cm} (14)

As for SIPIC01 and SIPIC02, their rise time and settling time also reduced with increasing $k_p$ or $k_i$ gains. However, both of them do not experience overshoot. This phenomenon can be explained based on Equations (8) and (13). Since the tuning parameters, $k_p$ and $k_i$ exist in the numerator and denominator, both of them increase or decrease together with the damping ratio. As mentioned above, increasing $k_p$ or $k_i$ gains will reduce the rise and settling time of SIPIC01 and SIPIC02. However, increasing $k_p$ or $k_i$ gains will only have slight effect on the damping ratio due to the existence of $k_p$ and $k_i$ gains in the numerator and denominator of Equations (8) and (13). Hence, it can be said that both SIPIC01 and SIPIC02 consist of decoupling features that allows a performance with coexistence of zero overshoot and short settling time. By comparing SIPIC01 and SIPIC02, SIPIC02 consists of longer rise and settling time. This is due to the $-f k_i k_t$ term in the denominator of Equation (13). With the $-f k_i k_t$ term, the denominator of Equation (13) will always be smaller as compared to the denominator of Equation (8). This will cause the SIPIC02 to have a greater damping ratio as compared to SIPIC01. The over damped response with greater damping ratio will lead to longer rise and settling time. By increasing $k_p$ or $k_i$ gains, the rise and settling time of SIPIC02 have a greater amount of reduction as compared to SIPIC01. The performance under loaded condition is similar to the result under no load condition.

7.2 Control Performance

The simulation for speed control at 100 rad/s for $k_p = 5$, $k_i = 10$ under no load condition is shown in Figure 7 while the experimental testing for speed control at 100 rad/s for $k_p = 5$, $k_i = 10$ under no load condition is shown in Figure 8. Based on Figure 7, PI consists of the shortest rise and settling time, whereas SIPIC02 has the longest rise and settling time. PI experienced overshoot while both SIPIC01 and SIPIC02 did not experience overshoot. Tables 5 and 6 summarise the rise time, settling time and percentage overshoot of PI, SIPIC01 and SIPIC02 under no load condition. According to Table 5, increasing $k_p$ or $k_i$ gains will reduce the rise and settling time for PI, SIPIC01 and SIPIC02. When the $k_p$ is 1, PI has the shortest rise and settling time while SIPIC02 has the longest rise and settling time. The controllers are said to have reached their lowest boundary of rise and settling time when the $k_p$ is $\geq 5$. Therefore, all controllers consist of similar rise and settling time. According to Table 6, there is no overshoot for all controllers at 50 rad/s under no load condition. The same performance can be observed when the controllers are tested at 100 rad/s under the same condition. However, PI experienced overshoot when $k_i$ is equal to 10. When $k_p$ increases, the percentage overshoot decreases. This can be explained by Equation (14) where the $k_p$ is directly proportional to the damping ratio. When the damping ratio increases, the percentage overshoot decreases. The rise time and settling time for both SIPIC01 and SIPIC02 reduced by increasing $k_p$ or $k_i$ gains without experiencing overshoot. This is due to the decoupling effect of $k_p$ or $k_i$ gains in SIPIC01 and SIPIC02. By comparing SIPIC01 and SIPIC02, SIPIC01 has a much shorter rise and settling time while both of them do not experience overshoot. Hence, it can be said that SIPIC01 is better in dynamic performance as compared to SIPIC02.
This is due to the decoupling effect of longer rise and settling time. This can be explained by Equation (14) where the denominator, both of them increase or decrease together with the damping ratio. As mentioned above, increasing $kp$ or $ki$ gains will reduce the rise and settling time for PI, SIPIC01 and SIPIC02. When the term in the denominator of Equation (13). With $kp=10$ under no load condition is shown and $ki=10$ under no load condition.

| $kp$ | $ki$ | Rise Time (s) | Settling Time (s) | Overshoot (%) | Rise Time (s) | Settling Time (s) | Overshoot (%) |
|------|------|---------------|-------------------|--------------|---------------|-------------------|--------------|
| 1    | 1    | 2.196         | 2.196             | 5.069        | 2.996         | 2.997             | 6.997        |
| 1    | 5    | 0.595         | 1.202             | 2.696        | 2.138         | 1.642             | 3.667        |
| 1    | 10   | 0.392         | 1.128             | 2.435        | 2.42          | 1.558             | 3.316        |
| 1    | 15   | 0.309         | 1.111             | 2.352        | 2.712         | 1.536             | 3.205        |
| 1    | 20   | 0.263         | 1.105             | 2.312        | 2.952         | 1.526             | 3.152        |
| 5    | 1    | 2.201         | 0.752             | 0.983        | 6.264         | 1.392             | 1.846        |
| 5    | 5    | 0.438         | 0.439             | 0.555        | 0.600         | 0.601             | 0.786        |
| 5    | 10   | 0.318         | 0.394             | 0.496        | 1.088         | 0.534             | 0.681        |
| 5    | 15   | 0.265         | 0.380             | 0.475        | 1.079         | 0.518             | 0.650        |
| 5    | 20   | 0.233         | 0.374             | 0.465        | 0.977         | 0.512             | 0.636        |
| 10   | 1    | 0.345         | 0.307             | 0.329        | 5.656         | 0.704             | 0.857        |
| 10   | 5    | 0.261         | 0.240             | 0.268        | 0.435         | 0.342             | 0.400        |
| 10   | 10   | 0.219         | 0.219             | 0.248        | 0.300         | 0.301             | 0.348        |
| 10   | 15   | 0.195         | 0.211             | 0.239        | 0.252         | 0.288             | 0.330        |
| 10   | 20   | 0.178         | 0.207             | 0.234        | 0.225         | 0.283             | 0.321        |
| 15   | 1    | 0.187         | 0.184             | 0.189        | 2.650         | 0.360             | 0.409        |
| 15   | 5    | 0.170         | 0.164             | 0.017        | 0.294         | 0.240             | 0.266        |
| 15   | 10   | 0.156         | 0.153             | 0.166        | 0.228         | 0.216             | 0.239        |
| 15   | 15   | 0.145         | 0.149             | 0.161        | 0.200         | 0.207             | 0.228        |
| 15   | 20   | 0.137         | 0.146             | 0.158        | 0.183         | 0.203             | 0.222        |
| 20   | 1    | 0.130         | 0.139             | 0.140        | 0.251         | 0.232             | 0.244        |
| 20   | 5    | 0.124         | 0.129             | 0.135        | 0.202         | 0.189             | 0.203        |
| 20   | 10   | 0.118         | 0.124             | 0.131        | 0.176         | 0.175             | 0.189        |
| 20   | 15   | 0.113         | 0.121             | 0.128        | 0.161         | 0.169             | 0.182        |
| 20   | 20   | 0.109         | 0.120             | 0.127        | 0.150         | 0.166             | 0.178        |
Fig. 7. Simulation for speed control at 100 rad/s for $k_p = 5$, $k_i = 10$ under no load condition.

Fig. 8. Experimental testing for speed control at 100 rad/s for $k_p = 5$, $k_i = 10$ under no load condition.

Tables 7 and 8 summarise the rise time, settling time and percentage overshoot of PI, SIPIC01 and SIPIC02 under load 1 (mild steel) condition. According Table 7, the average rise and settling time of PI, SIPIC01 and SIPIC02 are longer compared to no load condition which is due to the higher moment of inertia. The PI experienced inconsistent settling time with increasing $k_p$ or $k_i$. This is due to the coupling effect of $k_p$ or $k_i$ as both of them affect each other in the response. According to Table 8, SIPIC01 experience insignificant amount of overshoot which is less than 1.2% when $k_i$ is equal to 10. Tables 9 and 10 summarise the rise time, settling time and percentage overshoot of PI, SIPIC01 and SIPIC02 under load 2
(aluminium) condition. As shown in Table 9 and 10, the controllers have the similar performance to the load 1 condition. The only difference is that all controllers have shorter average rise and settling time which is due to the lower moment of inertia. Besides, SIPIC02 experienced overshoot under load 2 condition. However, the amount is insignificant.
Table 5. The rise time and settling time of PI, SIPIC01 and SIPIC02 under no load condition

| Speed (rad/s) | \( k_p \) | \( k_i \) | Rise Time (s) | Settling Time (s) |
|--------------|----------|----------|---------------|-------------------|
|              |          |          | PI | SIPIC01 | SIPIC02 | PI | SIPIC01 | SIPIC02 |
| 50           | 1        | 1        | 0.009 | 0.011 | 0.518 | 0.742 | 0.708 | 2.427 |
|              | 1        | 5        | 0.008 | 0.009 | 0.377 | 0.106 | 0.143 | 1.758 |
|              | 1        | 10       | 0.007 | 0.008 | 0.334 | 0.016 | 0.072 | 1.032 |
|              | 5        | 1        | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
|              | 5        | 5        | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
|              | 5        | 10       | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
|              | 10       | 1        | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
|              | 10       | 5        | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
|              | 10       | 10       | 0.006 | 0.006 | 0.006 | 0.010 | 0.010 | 0.010 |
| 100          | 1        | 1        | 0.021 | 0.023 | 0.885 | 0.690 | 0.709 | 2.341 |
|              | 1        | 5        | 0.020 | 0.021 | 0.678 | 0.029 | 0.143 | 2.021 |
|              | 1        | 10       | 0.020 | 0.020 | 0.564 | 0.027 | 0.073 | 1.263 |
|              | 5        | 1        | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |
|              | 5        | 5        | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |
|              | 5        | 10       | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |
|              | 10       | 1        | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |
|              | 10       | 5        | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |
|              | 10       | 10       | 0.020 | 0.020 | 0.020 | 0.027 | 0.027 | 0.027 |

Table 6. The overshoot percentage of PI, SIPIC01 and SIPIC02 under no load condition

| Speed (rad/s) | \( k_p \) | \( k_i \) | Overshoot (%) |
|--------------|----------|----------|---------------|
|              |          |          | PI | SIPIC01 | SIPIC02 |
|              |          |          |     |          |          |
| 50           | 1        | 1        | 0   | 0        | 0        |
|              | 1        | 5        | 0   | 0        | 0        |
|              | 1        | 10       | 0   | 0        | 0        |
|              | 5        | 1        | 0   | 0        | 0        |
|              | 5        | 5        | 0   | 0        | 0        |
|              | 5        | 10       | 0   | 0        | 0        |
|              | 10       | 1        | 0   | 0        | 0        |
|              | 10       | 5        | 0   | 0        | 0        |
|              | 10       | 10       | 0   | 0        | 0        |
| 100          | 1        | 1        | 0   | 0        | 0        |
|              | 1        | 5        | 0   | 0        | 0        |
|              | 1        | 10       | 0.693 | 0        | 0        |
|              | 5        | 1        | 0   | 0        | 0        |
|              | 5        | 5        | 0   | 0        | 0        |
|              | 5        | 10       | 0.462 | 0        | 0        |
|              | 10       | 1        | 0   | 0        | 0        |
|              | 10       | 5        | 0   | 0        | 0        |
|              | 10       | 10       | 0.221 | 0        | 0        |
Table 7. The rise time and settling time of PI, SIPIC01 and SIPIC02 under no load condition

| Speed (rad/s) | k_p | k_i | Rise Time (s) | Settling Time (s) |
|---------------|-----|-----|---------------|-------------------|
|               |     |     | PI            | SIPIC01 | SIPIC02 | PI | SIPIC01 | SIPIC02 |
| 50            | 1   | 1   | 0.048         | 0.054   | 1.084   | 0.565 | 0.632   | 2.226   |
|               | 1   | 5   | 0.038         | 0.042   | 0.792   | 0.051 | 0.067   | 2.001   |
|               | 1   | 10  | 0.037         | 0.038   | 0.640   | 0.155 | 0.052   | 1.336   |
|               | 5   | 1   | 0.037         | 0.037   | 0.037   | 0.047 | 0.047   | 0.047   |
|               | 5   | 5   | 0.037         | 0.037   | 0.037   | 0.047 | 0.047   | 0.047   |
|               | 10  | 1   | 0.037         | 0.037   | 0.037   | 0.047 | 0.047   | 0.047   |
|               | 10  | 5   | 0.037         | 0.037   | 0.037   | 0.047 | 0.047   | 0.047   |
| 100           | 1   | 1   | 0.107         | 0.109   | 2.001   | 0.160 | 0.630   | 1.830   |
|               | 1   | 5   | 0.107         | 0.107   | 1.619   | 0.424 | 0.131   | 2.142   |
|               | 1   | 10  | 0.107         | 0.107   | 1.130   | 0.445 | 0.130   | 1.830   |
|               | 5   | 1   | 0.107         | 0.107   | 0.107   | 0.130 | 0.130   | 0.130   |
|               | 5   | 5   | 0.107         | 0.107   | 0.107   | 0.130 | 0.130   | 0.130   |
|               | 5   | 10  | 0.107         | 0.107   | 0.107   | 0.460 | 0.130   | 0.130   |
|               | 10  | 1   | 0.107         | 0.107   | 0.107   | 0.130 | 0.130   | 0.130   |
|               | 10  | 5   | 0.107         | 0.107   | 0.107   | 0.130 | 0.130   | 0.130   |
|               | 10  | 10  | 0.107         | 0.107   | 0.107   | 0.130 | 0.130   | 0.130   |

Table 8. The overshoot percentage of PI, SIPIC01 and SIPIC02 under load 1 condition

| Speed (rad/s) | k_p | k_i | Overshoot (%) |
|---------------|-----|-----|---------------|
|               |     |     | PI            | SIPIC01 | SIPIC02 |
| 50            | 1   | 1   | 0             | 0       | 0       |
|               | 1   | 5   | 0.822         | 0       | 0       |
|               | 1   | 10  | 9.934         | 0.339   | 0       |
|               | 5   | 1   | 0             | 0       | 0       |
|               | 5   | 5   | 0.362         | 0       | 0       |
|               | 5   | 10  | 2.694         | 0.296   | 0       |
|               | 10  | 1   | 0             | 0       | 0       |
|               | 10  | 5   | 0.829         | 0       | 0       |
|               | 10  | 10  | 1.974         | 0.804   | 0       |
| 100           | 1   | 1   | 0             | 0       | 0       |
|               | 1   | 5   | 13.417        | 0       | 0       |
|               | 1   | 10  | 23.940        | 1.173   | 0       |
|               | 5   | 1   | 0             | 0       | 0       |
|               | 5   | 5   | 3.406         | 0       | 0       |
|               | 5   | 10  | 8.743         | 0.353   | 0       |
|               | 10  | 1   | 0             | 0       | 0       |
|               | 10  | 5   | 1.834         | 0       | 0       |
|               | 10  | 10  | 4.608         | 0.302   | 0       |
Table 9. The rise time and settling time of PI, SIPIC01 and SIPIC02 under load 2 condition

| Speed (rad/s) | $k_p$ | $k_i$ | Rise Time (s) | Settling Time (s) |
|---------------|-------|-------|---------------|-------------------|
|               |       |       | PI | SIPIC01 | SIPIC02 | PI | SIPIC01 | SIPIC02 |
| 50            | 1     | 1     | 0.022 | 0.025 | 0.728 | 0.685 | 0.684 | 2.398 |
|               | 1     | 5     | 0.017 | 0.020 | 0.544 | 0.031 | 0.116 | 1.925 |
|               | 1     | 10    | 0.016 | 0.018 | 0.465 | 0.024 | 0.041 | 1.164 |
|               | 5     | 1     | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
|               | 5     | 5     | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
|               | 5     | 10    | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
|               | 10    | 1     | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
|               | 10    | 5     | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
|               | 10    | 10    | 0.016 | 0.016 | 0.016 | 0.022 | 0.022 | 0.022 |
| 100           | 1     | 1     | 0.049 | 0.052 | 1.572 | 0.547 | 0.685 | 2.174 |
|               | 1     | 5     | 0.049 | 0.049 | 1.088 | 0.060 | 0.115 | 2.103 |
|               | 1     | 10    | 0.049 | 0.049 | 0.834 | 0.194 | 0.061 | 1.533 |
|               | 5     | 1     | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |
|               | 5     | 5     | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |
|               | 5     | 10    | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |
|               | 10    | 1     | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |
|               | 10    | 5     | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |
|               | 10    | 10    | 0.049 | 0.049 | 0.049 | 0.060 | 0.060 | 0.060 |

Table 10. The overshoot percentage of PI, SIPIC01 and SIPIC02 under load 2 condition

| Speed (rad/s) | $k_p$ | $k_i$ | Overshoot (%) |
|---------------|-------|-------|---------------|
|               |       |       | PI | SIPIC01 | SIPIC02 |
| 50            | 1     | 1     | 0  | 0  | 0  |
|               | 1     | 5     | 0  | 0  | 0  |
|               | 1     | 10    | 0.697 | 0 | 0 |
|               | 5     | 1     | 0  | 0  | 0  |
|               | 5     | 5     | 0.635 | 0.081 | 0 |
|               | 5     | 10    | 1.723 | 0.485 | 0 |
|               | 10    | 1     | 1.325 | 1.263 | 1.132 |
|               | 10    | 5     | 1.772 | 1.496 | 0.888 |
|               | 10    | 10    | 2.144 | 1.697 | 0.588 |
| 100           | 1     | 1     | 0  | 0  | 0  |
|               | 1     | 5     | 0  | 0  | 0  |
|               | 1     | 10    | 12.639 | 0 | 0 |
|               | 5     | 1     | 0  | 0  | 0  |
|               | 5     | 5     | 0.746 | 0 | 0 |
|               | 5     | 10    | 3.351 | 0 | 0 |
|               | 10    | 1     | 0  | 0  | 0  |
|               | 10    | 5     | 0.650 | 0 | 0 |
|               | 10    | 10    | 1.959 | 0.095 | 0 |
Based on the rise time, settling time and overshoot percentage under three load cases, it can be said that SIPIC01 and SIPIC02 are better in dynamic performance as compared to PI. By comparing SIPIC01 and SIPIC02, SIPIC01 has a much faster rise and settling time. Although the percentage overshoot of SIPIC01 is slightly greater as compared to SIPIC02, the amount of overshoot experienced by SIPIC01 and SIPIC02 are insignificant. Therefore, it can be said that SIPIC01 is better in dynamic performance as compared to SIPIC02.

8. Conclusions

In conclusion, both SIPIC01 and SIPIC02 consist of decoupling feature which allow them to produce a performance with coexistence of zero or minimum overshoot and short settling time. However, by comparing SIPIC01 and SIPIC02, SIPIC02 consists of longer rise and settling time. Hence, it can be concluded that SIPIC01 is better than SIPIC02 in term of dynamic performance. As shown in Tables 6, 8 and 10, SIPIC02 does not experience overshoot except under loading 2 condition at 50 rad/s. The zero overshoot performance of SIPIC02 may be due to its longer rise and settling time and cannot reach its maximum speed with the simulated period. In future, work will be conducted to verify the stability of SIPIC02. From this research, increasing $k_p$ or $k_i$ gains gives a reduction trend for the rise and settling time of SIPIC02. A more in-depth study of the dynamic performance and mathematical model of SIPIC02 by determining the integral output and torque will be performed to investigate this observation.

References

1. R. C. Dorf and R. H. Bishop, “Introduction to Control Systems,” Mod. Control Syst., 2–23 (2000).
2. W. Yan, D. Wang, P. Jia, and W. Li, “The PWM speed regulation of DC motor based on intelligent control,” Syst. Eng. Procedia, 3, 259–267 (2012).
3. P. M. Menghal and A. J. Laxmi, “Dynamic analysis & artificial intelligent control of induction motor drives,” Annu. IEEE India Conf., December, 1–6 (2014).
4. S. K. Suman and V. K. Giri, “Performance Analysis of optimization techniques and Intelligence Techniques for speed control of DC motor,” African J. Basic Appl. Sci., 9, no. 2, 97–104 (2017).
5. X. L. Li, J. G. Park, and H. B. Shin, “Comparison and evaluation of anti-windup PI controllers,” J. Power Electron., 11, no. 1, 45–50 (2011).
6. T. L. Chiah, C. L. I. H. Hoo, E. Chin, and Y. A. U. Chung, “Hardware Simulation of a New Anti-Windup PI Control For Motor Speed Application,” J. Eng. Sci. Technol., 43–57 (2017).
7. C. L. Hoo, S. M. Haris, E. C. Y. Chung, and N. A. N. Mohamed, “The generalisation and decoupling mode of PI-based control: theoretical approach,” ICIC Express Lett., 4, no. 5, 1–6 (2010).
8. C. L. Hoo, E. C. Y. Chung, S. M. Haris, and N. A. N. Mohamed, “Steady-state integral proportional integral controller for PI motor speed controllers,” J. Power Electron., 9, no. 6, 1777–1782 (2015).
9. C. L. Hoo, S. M. Haris, C. Y. E. Chung, and N. A. N. Mohamed, “Steady-state integral proportional integral controller for PI motor speed controllers: a theoretical approach,” ICIC Express Lett., 4, no. 5, 1–6 (2010).