CS-Based Optimal PI$^{\lambda}$D$^{\mu}$ Controller Design for Induction Motor Speed Control System

Chaiyo Thammarat and Deacha Puangdownreong

Department of Electrical Engineering
Graduate School, Southeast Asia University, Bangkok, Thailand, 10160
deachap@sau.ac.th

Abstract: Over two decades, the PI$^{\lambda}$D$^{\mu}$ (or fractional-order PID) controller was introduced and demonstrated to perform the better responses in comparison with the conventional integer-order PID controller. The PI$^{\lambda}$D$^{\mu}$ controller consists of five parameters, i.e. proportional gain (K_p), integration gain (K_i), derivative gain (K_d), integration order (λ) and derivative order (μ). Effects of the PI$^{\lambda}$D$^{\mu}$ parameter tuning on the system responses are studied and summarized in this paper. Based on the modern optimization, designing the optimal PI$^{\lambda}$D$^{\mu}$ controller for an induction motor speed control system by using the cuckoo search (CS), one of the most efficient metaheuristic optimization search techniques, is proposed. Five parameters of the PI$^{\lambda}$D$^{\mu}$ controller will be optimized by the CS to meet the response specifications of the three-phase induction motor (3ϕ-IM) speed control system which is defined as particularly constraint functions. Results obtained by the PI$^{\lambda}$D$^{\mu}$ controller are compared with those obtained by the conventional PID controller designed by Ziegler–Nichols tuning rule, Cohen–Coon tuning rule and the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS. As simulation results, it was found that the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS can provide faster and smoother speed responses than others, significantly. The stability of the 3ϕ-IM speed controlled system with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS is also investigated.

Keywords: PI$^{\lambda}$D$^{\mu}$ Controller; Cuckoo Search; Ziegler–Nichols Type Tuning Rule.

1. Introduction

Based on the fractional calculus, the PI$^{\lambda}$D$^{\mu}$ (or fractional-order PID) controller was firstly proposed by Podlubny in 1994 [1, 2] as an extended version of the conventional integer-order PID controller. The PI$^{\lambda}$D$^{\mu}$ controller possesses five tuning parameters: proportional gain (K_p), integration gain (K_i), derivative gain (K_d), integration order (λ) and derivative order (μ), whereas the conventional PID controller consists of only three tuning parameters: proportional gain (K_p), integration gain (K_i) and derivative gain (K_d). Podlubny proved the superiority of the PI$^{\lambda}$D$^{\mu}$ to the conventional PID controller when applied for control systems [1, 2]. Once the PI$^{\lambda}$D$^{\mu}$ is compared with the conventional PID controller, there are two extra parameters λ and μ making the PI$^{\lambda}$D$^{\mu}$ controller more efficient, but more complicate than the conventional PID controller in design and implementation procedures. By literatures, the PI$^{\lambda}$D$^{\mu}$ controller has been successfully conducted in many applications, for instance, process control [3], automatic voltage regulator [4], DC motor control [5], power electronic control [6], inverted pendulum control [7] and gun control system [8]. Several design and tuning methods for the PI$^{\lambda}$D$^{\mu}$ controller have been consecutively launched, for example, rule-based methods [9, 10] and analytical methods [11, 12]. Review and tutorial articles of the PI$^{\lambda}$D$^{\mu}$ controller providing the state-of-the-art and its backgrounds have been completely reported [13, 14]. Control theorists believe that since the conventional PID controller dominates the industry, the PI$^{\lambda}$D$^{\mu}$ controller will gain increasing impact and wide acceptance. Based on real-world applications, fractional order control with the PI$^{\lambda}$D$^{\mu}$ controller will be ubiquitous. Although, the effects of tuning PID parameters on the system responses are well-known for practicing engineers [15, 16], the effects of the PI$^{\lambda}$D$^{\mu}$ controller on the system responses are not reported.

Received: February 30th, 2019. Accepted: August 24th, 2019
DOI: 10.15676/ijeei.2019.11.4.2

Nowadays, control synthesis has been changed from the conventional paradigm to the new framework based on modern optimization using metaheuristics as an optimizer [17, 18]. By literatures, the cuckoo search (CS), firstly proposed by Yang and Deb in 2009, is one of the most powerful population-based metaheuristic optimization search techniques [19]. The CS was proved for the global convergent property [20] and successfully applied to many real-world engineering problems, such as wind turbine blades [21], antenna arrays [22], power systems [23], travelling salesman problem [24], structural optimization problem [25], wireless sensor network [26], flow shop scheduling problem [27], job shop scheduling problem [28], model order reduction [29] and control systems [30]. The state-of-the-art and its applications of the CS have been reviewed and reported [31].

In this paper, effects of the PI$^{\lambda}$D$^{\mu}$ parameter tuning on the system responses are studies for practicing control engineers. An application of the CS to optimally design the PI$^{\lambda}$D$^{\mu}$ controller for the 3ϕ-IM speed control system is then proposed. This paper is arranged as follows. After an introduction is provided in Section 1, fractional calculus, the PI$^{\lambda}$D$^{\mu}$ controller and stability analysis of linear, time-invariant (LTI) fractional order system are briefly described in Section 2. The CS algorithm is briefly described in Section 3. Study of the effects of the PI$^{\lambda}$D$^{\mu}$ parameter tuning on system responses is performed in Section 4. Results and discussion of the CS-based PI$^{\lambda}$D$^{\mu}$ design for the 3ϕ-IM speed control system are given in Section 5. Stability analysis of the 3ϕ-IM speed control system with the PI$^{\lambda}$D$^{\mu}$ controller is investigated in Section 6, while conclusions are given in Section 7.

2. Fractional Order Control System

In this section, fractional calculus, the PI$^{\lambda}$D$^{\mu}$ controller and stability analysis of linear, time-invariant (LTI) fractional order system are briefly described as follows.

A. Fractional Calculus

In fractional calculus, a generalization of integration and differentiation can be represented by the non-integer order fundamental operator ${}_aD_t^{\alpha}$, where a and t are the limits of the operator. The continuous integro-differential operator is defined as expressed in (1), where $\alpha \in \Re$ stands for the order of operation. There are three definitions used for the generally fractional differintegral. The first definition is Grunwald-Letnikov (GL) as stated in (2), where $[\cdot]$ is integer part and n is an integer satisfying the condition $n-1<\alpha<n$. The binomial coefficient is stated in (3), while the Euler's gamma function $\Gamma(\cdot)$ is defined by (4).

$$ {}_aD_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt^{\alpha}} & \Re(\alpha) > 0 \\ 1 & \Re(\alpha) = 0 \\ \int_a^t (d\tau)^{-\alpha} & \Re(\alpha) < 0 \end{cases} \tag{1} $$

$$ {}_aD_t^{\alpha} f(t) = \lim_{h\to 0} h^{-\alpha} \sum_{r=0}^{\left[\frac{t-a}{h}\right]} (-1)^r \binom{n}{r} f(t-rh) \tag{2} $$

$$ \binom{n}{r} = \frac{\Gamma(n+1)}{\Gamma(r+1)\Gamma(n-r+1)} \tag{3} $$

$$ \Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t} dt \tag{4} $$

The second definition is Riemann-Liouville (RL) as expressed in (5), for $n-1<\alpha<n$. The third definition is Caputo definition as shown in (6), where n is an integer and $n-1<\alpha<n$. Among those, the Caputo definition is most popular in engineering applications [13].

$$ {}_aD_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \left(\frac{d}{dt}\right)^n \int_a^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{5} $$

$$ {}_aD_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \int_a^t \frac{f^{n}(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{6} $$

$$L\{\, {}_aD_t^{\alpha} f(t)\} = \int_0^{\infty} e^{-st}\ {}_0D_t^{\alpha} f(t)\, dt = s^{\alpha}F(s) - \sum_{k=0}^{n-1} s^k\ {}_0D_t^{\alpha-k-1} f(t)\Big|_{t=0} \tag{7}$$

For solving engineering problems, the Laplace transform is routinely conducted. The formula of the Laplace transform of the RL fractional derivative in (5) is stated in (7), for $n-1<\alpha\leq n$, where $s\equiv j\omega$ denotes the Laplace transform (complex) variable. Under zero initial conditions for order α ($0<\alpha<1$), the Laplace transform of the RL fractional derivative in (5) can be expressed in (8).

$$L\{\, {}_aD_t^{\pm\alpha} f(t)\} = s^{\pm\alpha}F(s) \tag{8}$$

B. $PI^{\lambda}D^{\mu}$ controller

The $PI^{\lambda}D^{\mu}$ controller is an extended version of the conventional PID controller. It can be regarded as the general controller for all PID family members. The generalized transfer function of the $PI^{\lambda}D^{\mu}$ controller is given by the differential equation as stated in (9), where $u(t)$ is the control signal, $e(t)$ is the error signal, λ and $\mu\geq 0$, and by the Laplace transform as expressed in (10).

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^{\mu} e(t) \tag{9}$$

$$G_c(s)\big|_{PI^{\lambda}D^{\mu}} = K_p + \frac{K_i}{s^{\lambda}} + K_d s^{\mu} \tag{10}$$

Relationship between the conventional PID controller and the $PI^{\lambda}D^{\mu}$ controller can be represented by a graphical way as visualized in Figure 1. In general, the range of fractional orders (λ and μ) is varied from 0 to 2. However, in most research works, the range of λ and μ is varied from 0 to 1. Referring to Figure 1, it was found as follows.

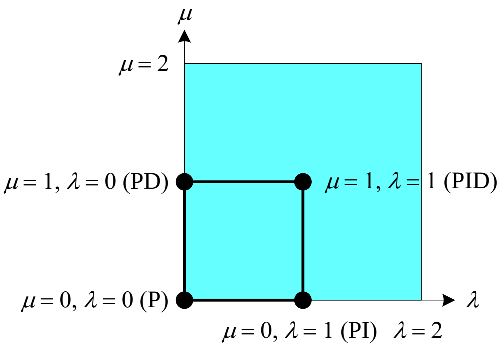


Figure 1. Relationship between PID and $PI^{\lambda}D^{\mu}$ controllers

if $\lambda = 0$ and $\mu = 0$, it is the P controller,
if $\lambda = 1$ and $\mu = 0$, it is the PI controller,
if $\lambda = 0$ and $\mu = 1$, it is the PD controller and
if $\lambda = 1$ and $\mu = 1$, it is the PID controller.

C. Stability of LTI Fractional Order System

In classical control theory, the LTI system is stable if the roots of the characteristic polynomial (or poles) are negative or have negative real parts if they are complex conjugate. It means that they are located on the left half of the complex plane. For the fractional order LTI case, the stability is different from the integer one. However, Matignon's stability theorem based on the Riemann sheet can be utilized for stability analysis of both integer order LTI and fractional order LTI systems [13, 14, 32, 33].

A general fractional order system can be formulated by a fractional differential equation as stated in (11) or by the corresponding transfer function of incommensurate real orders as expressed in (12), where a_k ($k = 0,\ldots,n$) and b_k ($k = 0,\ldots,m$) are constant, and α_k ($k = 0,\ldots,n$)

and β_k (k = 0,…,m) are arbitrary real numbers which can be arranged as $\alpha_n > \alpha_{n-1} > \cdots > \alpha_0$ and $\beta_m > \beta_{m-1} > \cdots > \beta_0$.

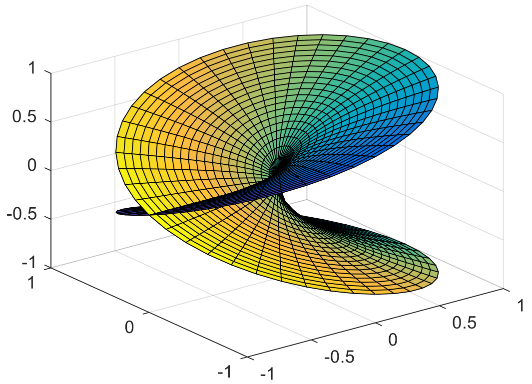
$$a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \cdots + a_1 D^{\alpha_1} y(t) + a_0 D^{\alpha_0} y(t) = b_m D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \cdots + b_1 D^{\beta_1} u(t) + b_0 D^{\beta_0} u(t) \quad (11)$$

$$G(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \cdots + b_1 s^{\beta_1} + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \cdots + a_1 s^{\alpha_1} + a_0 s^{\alpha_0}} = \frac{Q(s^{\beta_k})}{P(s^{\alpha_k})} \quad (12)$$

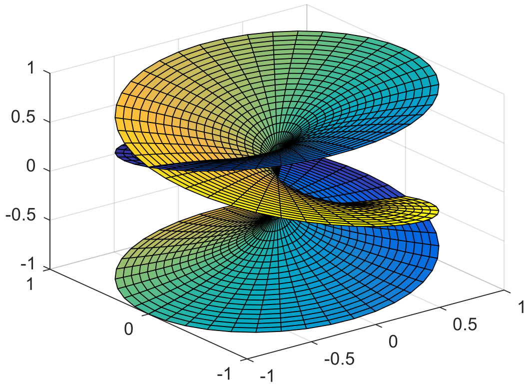
The incommensurate order system can be transformed into the commensurate form by the multi-valued transfer function as stated in (13). This implies that every fractional order system can be expressed in the form of (13) and domain of the $H(s)$ definition is a Riemann surface with v Riemann sheets [32, 33, 34, 35].

$$H(s) = \frac{b_m s^{m/v} + b_{m-1} s^{m-1/v} + \cdots + b_1 s^{1/v} + b_0}{a_n s^{n/v} + a_{n-1} s^{n-1/v} + \cdots + a_1 s^{1/v} + a_0}, \quad v > 1 \quad (13)$$

For example, the Riemann surfaces of multi-valued functions $w = s^{1/2}$ and $w = s^{1/3}$ are depicted in Figure 2(a) and Figure 2(b), respectively.



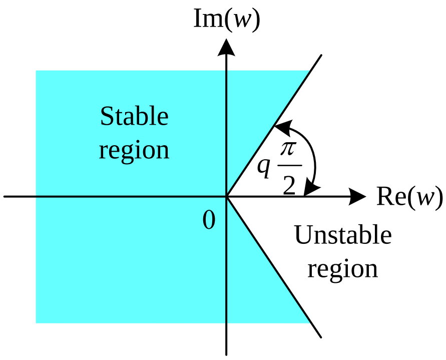
(a) Functions $w = s^{1/2}$



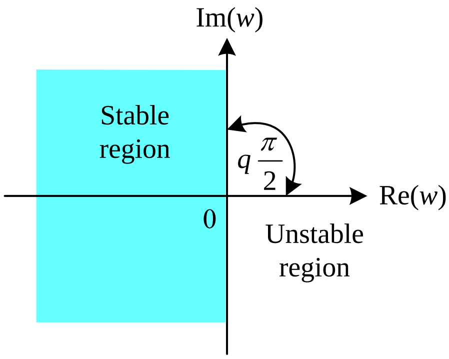
(b) Functions $w = s^{1/3}$

Figure 2. Riemann surfaces

The Riemann surface of any fractional order system will be transformed into the Riemann sheets [32, 33, 34, 35]. The stability region of the fractional order system can be performed by the first (or principal) Riemann sheet on the complex w-plane as visualized in Figure 3.



(a) $0 < q < 1$



(b) $q = 1$

Figure 3. Stability regions

For the case of commensurate-order systems, whose characteristic equation is a polynomial of the complex variable $\lambda = s^{\alpha}$, the stability condition is expressed in (14), where λ_i are the roots of the characteristic polynomial $P(s^{\alpha}) = 0$.

$$|arg\,(\lambda_i)| > \alpha\frac{\pi}{2}, \quad i = 1,\ 2, \dots, n \tag{14}$$

A commensurate-order system described by a rational transfer function in (15), where $w = s^q$, $q \in \Re^+$, $(0 < q < 2)$, is stable if and only if the stability condition stated in (16) is satisfied, where w_i are the roots of the characteristic polynomial $P(w) = 0$.

$$G(w) = \frac{Q(w)}{P(w)} \tag{15}$$

$$|arg\,(w_i)| > q\frac{\pi}{2}, \quad i = 1,\ 2, \dots, n \tag{16}$$

Referring to (12), the characteristic equation of the fractional order system can be formulated in (17). It may be rewritten as (18), where u_i and v_i are the integer numbers. Then, it can be transformed into the w-plane as (19), where $w = s^{1/m}$ and m is the least common multiple (LCM) of v_i.

$$a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \dots + a_1 s^{\alpha_1} + a_0 s^{\alpha_0} = \sum_{i=0}^{n} a_i a^{\alpha_i} = 0 \tag{17}$$

$$\sum_{i=0}^{n} a_i a^{u_i/v_i} = 0 \tag{18}$$

$$\sum_{i=0}^{n} a_i w^i = 0 \tag{19}$$

Let $|arg(w_i)|$ be the absolute phase of all roots w_i. Roots in the s-plane have corresponding roots in the w-plane by the transformation $s = w^m$.

The condistion for stability is $\pi/2m < |\arg(w_i)| < \pi/m$. Condition for oscillation or undamped response is $|\arg(w_i)| = \pi/2m$, otherwise the system is unstable. This means that if there is not root in the unstable region of the w-plane, the system will always be stable [32, 33, 34, 35].

3. CS Algorithm

The algorithm of the CS is briefly reviewed in this section. The CS algorithm is based on the general cuckoo bird's behavior which can be described by three following idealized rules [19, 20].

- Each cuckoo lays one egg at a time, and dumps it in a randomly chosen nest.
- The best nests with high quality of eggs (solutions) will carry over to the next.
- The number of available host nests is fixed, and a host can discover an alien egg with a probability $p_a \in [0, 1]$. In this case, the host bird can either throw the egg away or abandon the nest, and build a completely new nest in a new location.

The last assumption can be approximated by a fraction p_a of the n nests being replaced by new nests (with new random solutions at new locations). This means that each egg in a nest represents each solution, while a cuckoo egg represents a new solution. The worse solutions will be replaced by the new solution (cuckoo egg). Based on three rules, the CS algorithm proposed by Yang and Deb [19, 20], can be summarized by the flow diagram shown in Figure 4.

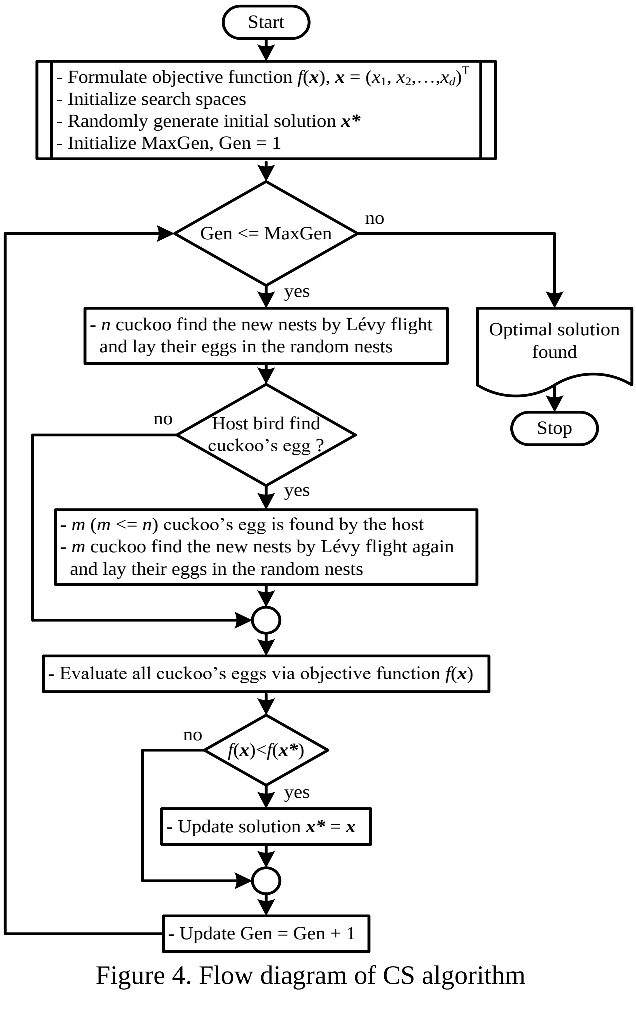


Figure 4. Flow diagram of CS algorithm

New solutions $x^{(t+1)}$ for cuckoo i can be generated by using a Lévy flight as stated in (20), where $\alpha > 0$ stands for the step size. A symbol $\oplus$ means entry-wise multiplications, while a symbol $\text{Lévy}(\lambda)$ represents a Lévy flight providing random walk with random step drawn from a Lévy distribution having an infinite variance with an infinite mean as expressed in (21). In the other hands, the step length s can be calculated by (22), where u and v are drawn from normal distribution as stated in (23). Standard deviations of u and v are expressed in (24). With fraction p_a, the CS can effectively escape from any local entrapment.

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Lévy(\lambda) \tag{20}$$

$$Lévy \approx u = t^{-\lambda}, \quad (1 < \lambda \leq 3) \tag{21}$$

$$s = \frac{u}{|v|^{1/\beta}} \tag{22}$$

$$u \approx N(0,\ \sigma_u^2), \quad v \approx N(0,\ \sigma_v^2) \tag{23}$$

$$\sigma_u = \sqrt[\beta]{\frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\beta 2^{(\beta-1)/2}}}, \quad \sigma_v = 1 \tag{24}$$

4. Effects of PI$^{\lambda}$D$^{\mu}$ Controller

Starting from the basic control loop represented by the block diagram of closed-loop system with fractional-order control actions shown in Figure 5, the effects of the basic control actions of type Ks^{μ} for $\mu \in [-1, 1]$ will be examined. The basic control actions traditionally considered will be particular cases of this general case, in which:

- = 0 : proportional action (P),
- = −1 : integral action (I) and
- = 1 : derivative action (D).

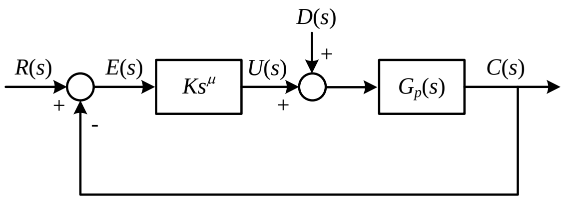


Figure 5. Closed-loop system with fractional-order control actions

For the fractional-order integral action, $\mu \in (-1, 0)$, Figure 6 shows the integral control actions for a square wave error signal (regarded as an input signal) and $\mu = 0, -0.2, -0.5, -1$. As can be observed, the effects of the control action over the square wave error signal vary between the effects of a proportional action ($\mu = 0$, square wave) and an integral action ($\mu = -1$, straight lines curve). For intermediate values of μ, the control action increases for a constant error, which results in the elimination of the steady-state error, and decreases when the error is zero, resulting in a more stable system.

For the fractional-order derivative action, $\mu \in (0, 1)$, Figure 7 shows the derivative control actions for the trapezoidal wave error signal (regarded as an input signal) and $\mu = 0, 0.2, 0.5, 1$. As can be observed, the effects of the control action over the trapezoidal wave error signal vary between the effects of a proportional action ($\mu = 0$, trapezoidal signal) and a derivative action ($\mu = 1$, square signal). For intermediate values of μ, the control action corresponds to intermediate curves. It must be noted that the derivative action is not zero for a constant error, and the growth of the control signal is more damped when a variation in the error signal occurs, which implies a better attenuation of high-frequency noise signals [32].

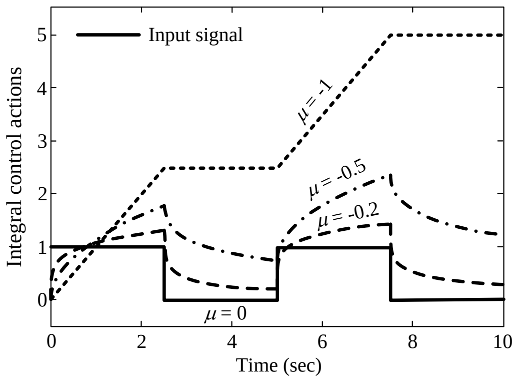


Figure 6. Integral control actions

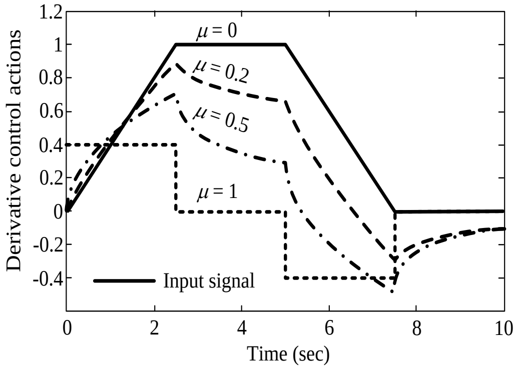


Figure 7. Derivative control actions

The step-response of the PI$^{\lambda}$D$^{\mu}$ controller can be depicted in Figure 8 – Figure 12. Once λ = 1 and μ = 1, the PI$^{\lambda}$D$^{\mu}$ controller is corresponding to the PID controller as shown in Figure 8. The portion *a* varies directly as K_p which results in speed up of the transient response and decreasing the steady-state error. The portion *b* varies directly as K_d to speed up the transient response and decrease the oscillation. The slope *d* varies directly as K_i to eliminate or decrease the steady-state error.

Once λ is varied and μ = 1, the step-responses of the PI$^{\lambda}$D$^{\mu}$ controller are depicted in Figure 9 – Figure 10. From Figure 9, λ < 1, The exponential curve portion *e* varies inversely as λ. The less the value of λ, the more the curve portion *e*. For Figure 10, λ > 1, The parabolic curve portion *e* varies directly as λ. The more the value of λ, the more the curve portion *e*. The effects of λ will reinforce those of K_i to eliminate or decrease the steady-state error.

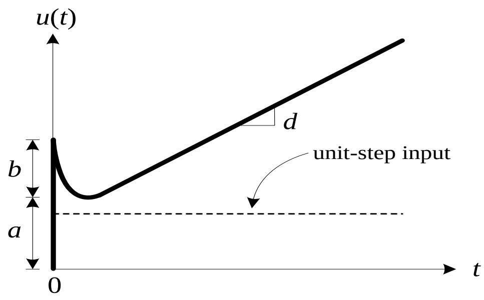


Figure 8. Step response of PI$^{\lambda}$D$^{\mu}$
(λ = 1 and μ = 1) corresponding to conventional PID controller

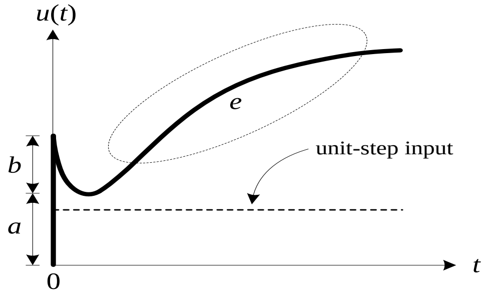


Figure 9. Step response of PI$^\lambda$D$^\mu$
($\lambda < 1$ and $\mu = 1$)

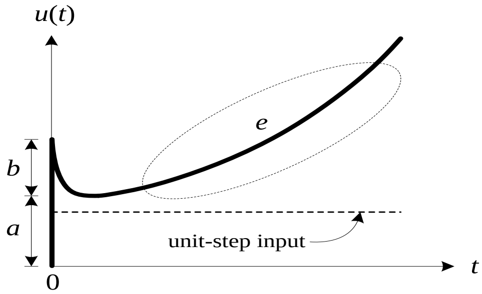


Figure 10. Step response of PI$^\lambda$D$^\mu$
($\lambda > 1$ and $\mu = 1$)

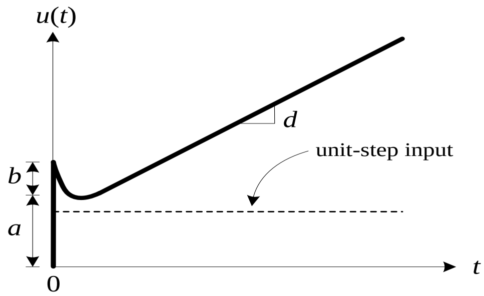


Figure 11. Step response of PI$^\lambda$D$^\mu$
($\lambda = 1$ and $\mu < 1$)

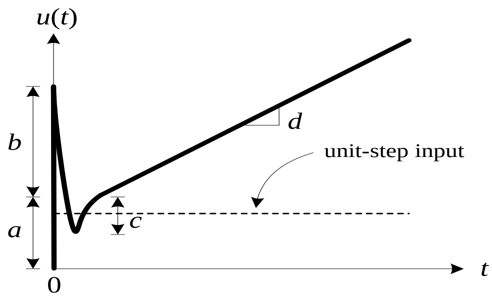


Figure 12. Step response of PI$^\lambda$D$^\mu$
($\lambda = 1$ and $\mu > 1$)

Once $\lambda = 1$ and μ is varied, the step-responses of the PI$^\lambda$D$^\mu$ controller are depicted in Figure 11 – Figure 12. From Figure 11, $\mu < 1$, The portion b varies directly as μ. The less the value of μ, the less the portion b. For Figure 12, $\mu > 1$, The portions b and c vary directly as μ. The more the value of μ, the more the portions b and c. The effects of μ will reinforce those of K_d to speed up the transient response and decrease the oscillation of the system response.

Study of the effects of the PI$^\lambda$D$^\mu$ parameter tuning on system responses is conducted by using the second-order prototype system as stated in (25), where ζ is a damping ratio and ω_n is an undamped natural frequency.

$$G_p(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{25}$$

Five tuning parameters of the PI$^\lambda$D$^\mu$ controller are K_p, K_i, K_d, λ and μ. By varying $\zeta = 0.7$, 0.8, 0.9, 1, 2, 3 and, $\omega_n = 1$, 2, 5, 10, the effects of increasing K_p (K_i and K_d are fixed, $\lambda = \mu = 1$) on system responses is shown in Figure 13. It was found that the rise time (t_r) is decreased, the maximum percent overshoot (M_p) is increased, the settling time (t_s) is minor changed and the steady-state error (e_{ss}) is decreased.

Figure 14 shows the effects of increasing K_i (K_p and K_d are fixed, $\lambda = \mu = 1$) on system responses. It was found that t_r is minor changed, M_p are t_s are increased and e_{ss} is eliminated. From Figure 15, the effects of increasing K_d (K_p and K_i are fixed, $\lambda = \mu = 1$) on system responses are plotted. It was found that t_r is minor changed, M_p are t_s are decreased and e_{ss} is not impact.

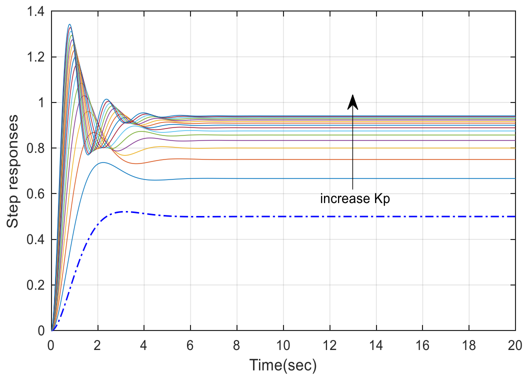


Figure 13. Effects of increasing K_p

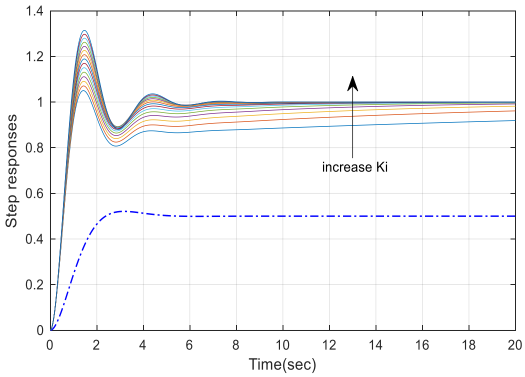


Figure 14. Effects of increasing K_i

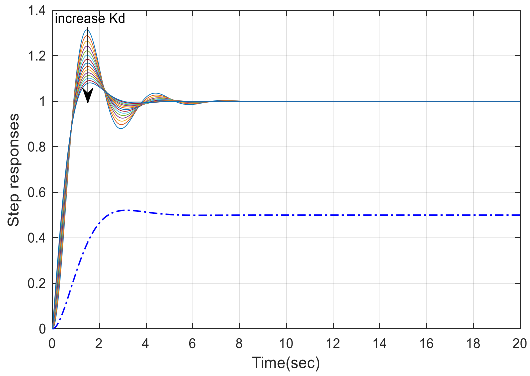


Figure 15. Effects of increasing K_d

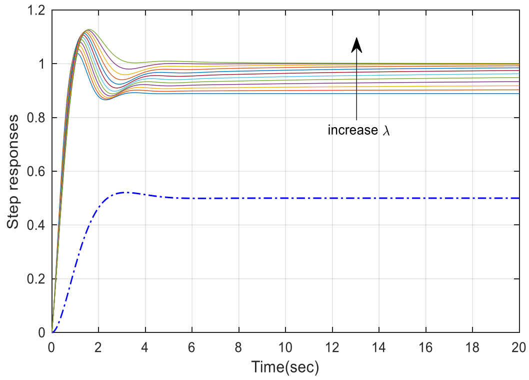


Figure 16. Effects of increasing λ

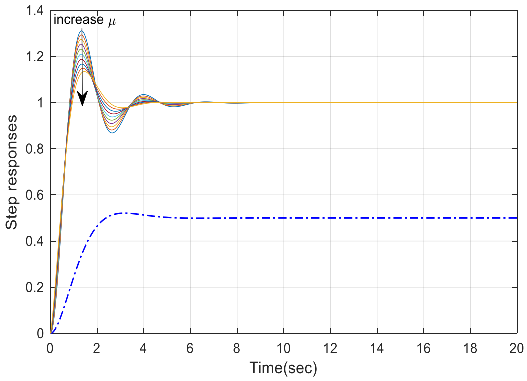


Figure 17. Effects of increasing μ

Figure 16 shows the effects of increasing λ (K_p, K_i and K_d are fixed, μ = 1) on system responses. It was found that t_r is minor changed, M_p and t_s are increased and e_{ss} is decreased. From Figure 17, the effects of increasing μ (K_p, K_i and K_d are fixed, λ = 1) on system responses are plotted. It was found that t_r is minor changed, M_p and t_s are decreased and e_{ss} is minor changed. Results of the study of the effects of the PI$^{\lambda}$D$^{\mu}$ parameter tuning on system responses are summarized in Table 1.

Table 1. Effects of PI$^\lambda$D$^\mu$ controller on system responses

PI$^\lambda$D$^\mu$ parameters	Step responses			
	Rise time (t_r)	Maximum percent overshoot (M_p)	Settling time (t_s)	Steady-state error (e_{ss})
Increase K_p	Decreased	Increased	Minimal impact	Decreased
Increase K_i	Minimal impact	Increased	Increased	Eliminated
Increase λ	Minimal impact	Increased	Increased	Decreased
Increase K_d	Minimal impact	Decreased	Decreased	No impact
Increase μ	Minimal impact	Decreased	Decreased	Minimal impact

5. CS-Based PI$^\lambda$D$^\mu$ Controller Design

Problem formulation of the CS-based optimal PI$^\lambda$D$^\mu$ controller design for the 3ϕ-IM speed control system is presented. For comparison, results obtained by the PI$^\lambda$D$^\mu$ controller designed by the CS will be compared with those obtained by the conventional PID controller designed by Ziegler–Nichols (ZN) [36, 37], Cohen–Coon (CC) [38] tuning rules and the PI$^\lambda$D$^\mu$ controller designed by the CS, respectively. In this section, the PI$^\lambda$D$^\mu$ control loop, the 3ϕ-IM model, PID controller design by ZN and CC tuning rules, PID and PI$^\lambda$D$^\mu$ controllers design by the CS are consecutively proposed as follows.

A. PI$^\lambda$D$^\mu$ Control Loop

The PI$^\lambda$D$^\mu$ control loop is represented by the block diagram as shown in Figure 18, where $G_p(s)$ and $G_c(s)$ are the plant and the PI$^\lambda$D$^\mu$ controller models, respectively. The PI$^\lambda$D$^\mu$ controller receives the error signal $E(s)$ and produces the control signal $U(s)$ to control the output signal $C(s)$ and to regulate the disturbance signal $D(s)$, referring to the reference input $R(s)$.

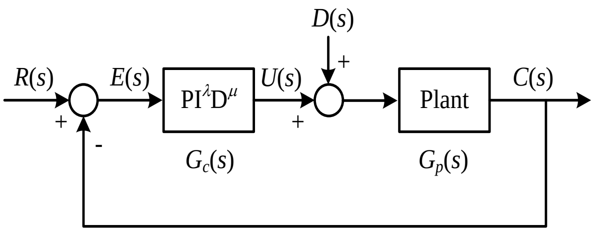


Figure 18. Closed-loop system with PI$^\lambda$D$^\mu$ controller

B. Induction Motor Model

In this work, a 0.37 kW, 1400 rpm, 50 Hz, 4-pole three-phase induction moter (3ϕ-IM) was conducted. Such the motor was tested as shown in Figure 19 to record its speed dynamics. By using MATLAB and system identification toolbox [39], the third-order transfer function was identified as given in (26). Good agreement between the model plot and the experimental speed (sensory data) can be observed in Figure 20. The plant model in (26) will be used as the plant $G_p(s)$ in Figure 18.

$$G_p(s) = \frac{1{,}360}{s^3 + 39.16s^2 + 398.14s + 1{,}360} \tag{26}$$

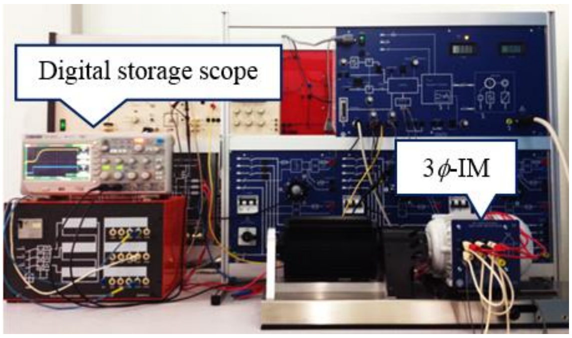


Figure 19. 3ϕ-IM testing rig

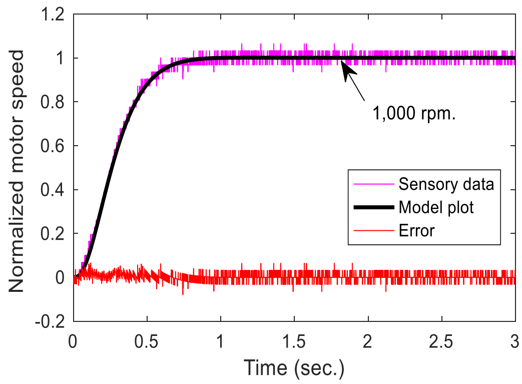


Figure 20. Model plot against sensory data

C. PID Controller Design by ZN

Referring to Figure 18, once $G_c(s)$ is the conventional PID controller, the model $G_c(s)$ of the conventional PID controller is expressed in (27), where K_c is controller gain, τ_i is integral time constant and τ_d is derivative time constant. For the first method of ZN tuning rule [36, 37], the delay time L and time constant T of the S-shaped curve obtained from the open-loop step response of the plant $G_p(s)$ are requested. Once L and T are obtained, the PID parameters (K_c, τ_i and τ_d) in (27) can be determined from Table 2.

$$G_c(s)|_{PID} = K_p + \frac{K_i}{s} + K_d s = K_c\left(1 + \frac{1}{\tau_i s} + \tau_d s\right) \tag{27}$$

Table 2. ZN tuning rule based on S-shaped step response of plant

Controllers	Parameters		
	K_c	τ_i	τ_d
P	T/L	∞	0
PI	$0.9T/L$	$L/0.3$	0
PID	$1.2T/L$	$2L$	$0.5L$

From the 3ϕ-IM model in (26) and its open-loop step response in Figure 20, it was found that L = 0.0574 sec. and T = 0.4344 sec. The PID parameters can be calculated: $K_c = 1.2T/L$ = 9.0815, $\tau_i = 2L$ = 0.1148 sec. and $\tau_d = 0.5L$ = 0.0287 sec. Therefore, the PID controller designed by the ZN tuning rule for the 3ϕ-IM speed control system is stated in (28).

$$G_c(s)|_{PID\_ZN} = 9.0815 + \frac{79.1074}{s} + 0.2606s \tag{28}$$

The step-input responses of the 3ϕ-IM speed control system without and with the PID controller designed by the ZN tuning rule are depicted in Figure 21, while the step-disturbance response of the 3ϕ-IM speed control system with the PID controller designed by the ZN tuning rule is plotted in Figure 22.

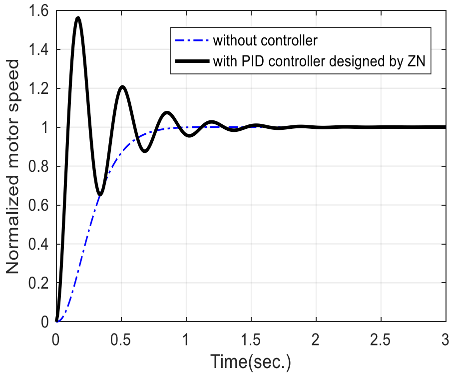


Figure 21. Step-input responses of the 3ϕ-IM speed controlled system without and with the PID controller designed by the ZN

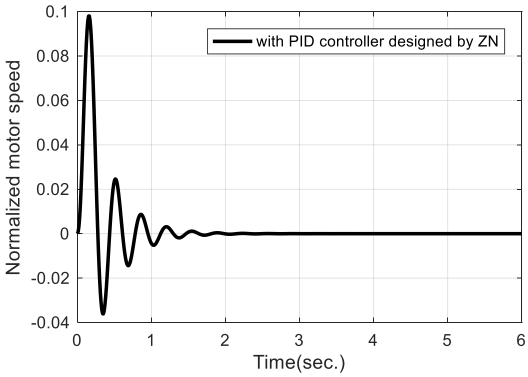


Figure 22. Step-disturbance response of the 3ϕ-IM speed controlled system with the PID controller designed by the ZN

From Figure 21, the step-input response of the 3ϕ-IM speed control system without controller provides t_r = 0.51 sec, M_p = 0%, t_s = 0.75 sec. and e_{ss} = 0, while that of the 3ϕ-IM speed controlled system with PID controller designed by ZN tuning rule gives t_r = 0.09 sec, M_p = 56.14%, t_s = 1.23 sec. and e_{ss} = 0. From Figure 22, the 3ϕ-IM speed controlled system with PID controller designed by the ZN tuning rule can regulate the step load disturbance. It yields the maximum percent overshoot from load disturbance regulation M_{p_reg} = 9.81% and the regulating time t_{reg} = 0.54 sec.

D. PID Controller Design by CC

The CC tuning rule [38] requires the delay time L and time constant T of the S-shaped curve obtained from the open-loop step response of the plant $G_p(s)$. Once L and T are measured, the PID parameters (K_c, τ_i and τ_d) in (27) can be determined from Table 3.

The 3ϕ-IM model in (26) and its open-loop step response in Figure 20 provide L = 0.0574 sec. and T = 0.4344 sec. Therefore, the PID controller designed by the CC tuning rule for the 3ϕ-IM speed control system is stated in (29).

$$G_c(s)|_{PID\_CC} = 10.4665 + \frac{76.9448}{s} + 0.2170s \tag{29}$$

The step-input responses of the 3ϕ-IM speed control system without and with the PID controller designed by the CC tuning rule are depicted in Figure 23, while the step-disturbance response of the 3ϕ-IM speed control system with the PID controller designed by the CC tuning rule is plotted in Figure 24.

From Figure 23, the step-input response of the 3ϕ-IM speed controlled system with PID controller designed by the CC tuning rule provides t_r = 0.09 sec, M_p = 61.24%, t_s = 1.62 sec. and e_{ss} = 0. From Figure 24, the 3ϕ-IM speed controlled system with PID controller designed by the CC tuning rule can regulate the step load disturbance. It gives M_{p_reg} = 9.54% and t_{reg} = 0.53 sec.

Table 3. CC tuning rule based on S-shaped step response of plant

Controllers	Parameters		
	K_c	τ_i	τ_d
P	$\frac{1.03}{K}\left(\frac{T}{L}+0.34\right)$	∞	0
PI	$\frac{0.9}{K}\left(\frac{T}{L}+0.092\right)$	$3.33L\left(\frac{T+0.092L}{T+2.22L}\right)$	0
PD	$\frac{1.24}{K}\left(\frac{T}{L}+0.129\right)$	∞	$0.27L\left(\frac{T-0.324L}{T+0.129L}\right)$
PID	$\frac{1.35}{K}\left(\frac{T}{L}+0.185\right)$	$2.5L\left(\frac{T+0.185L}{T+0.611L}\right)$	$0.37L\left(\frac{T}{T+0.185L}\right)$

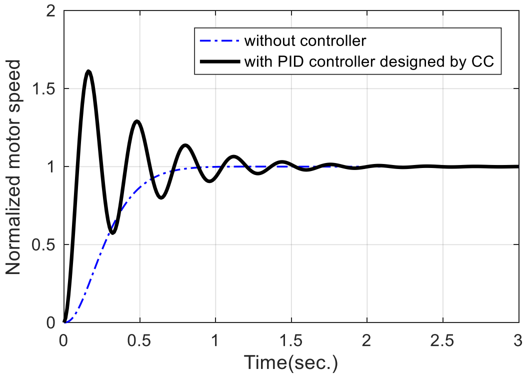


Figure 23. Step-input responses of the 3ϕ-IM speed controlled system without and with the PID controller designed by the CC

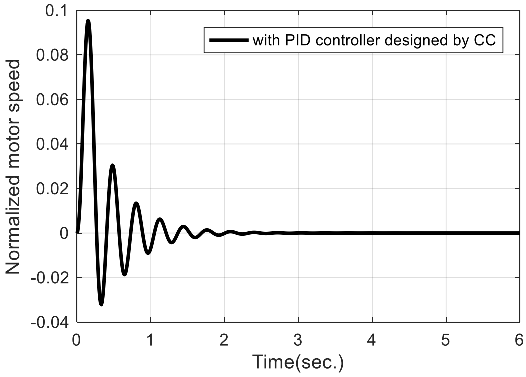


Figure 24. Step-disturbance response of the 3ϕ-IM speed controlled system with the PID controller designed by the CC

E. PID Controller Design by CS

Based on modern optimization framework, application of the CS to design an optimal PID controller for 3ϕ-IM speed control system can be represented by the block diagram in Figure

25. The objective function f, sum-squared error between the reference speed $R(s)$ and the actual speed $C(s)$ in (30), will be fed to the CS to be minimized by searching for the appropriate values of the PID parameters, i.e. K_p, K_i and K_d subject to inequality constraint functions satisfying to the predefined response specifications as stated in (31).

$$Min\ f(K_p, K_i, K_d) = \sum_{i=1}^{N}[r_i - c_i]^2 \tag{30}$$

$$\left.\begin{aligned} Subject\ to\quad & t_r \leq 0.2\ sec., \\ & M_p \leq 10.0\ \%, \\ & e_{ss} \leq 0.01\ \%, \\ & 0 < K_p \leq 10, \\ & 0 < K_i \leq 20, \\ & 0 < K_d \leq 1.0 \end{aligned}\right\} \tag{31}$$

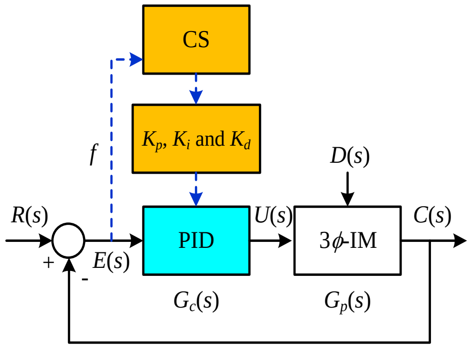


Figure 25. CS-based PID controller design by CS

The CS algorithm was coded by MATLAB version 2018b (License No.#40637337) run on Intel(R) Core(TM) i5-3470 CPU@3.60GHz, 4.0GB-RAM. Number of cuckoos n = 40 and fraction p_a = 0.2 are set according to recommendations of Yang and Deb [19, 20]. The maximum generation Max_Gen = 200 is then set as the termination criteria (TC). 50 trials are conducted to find the best solution (optimal PID controller for the 3ϕ-IM speed control system).

Once the search process stopped, the CS can successfully provide the optimal parameters of the PID controller for the 3ϕ-IM speed control system as expressed in (32). The convergent rates of the objective functions in (30) associated with inequality constraint functions in (31) proceeded by the CS over 50 trials are depicted in Figure 26. The step-input responses of the 3ϕ-IM speed control system without and with the PID controller designed by the CS are depicted in Figure 27, while the step-disturbance response of the 3ϕ-IM speed control system with the PID controller designed by the CS is plotted in Figure 28.

From Figure 27, the step-input response of the 3ϕ-IM speed controlled system with PID controller designed by the CS tuning rule gives t_r = 0.11 sec, M_p = 9.05%, t_s = 0.43 sec. and e_{ss} = 0. From Figure 28, the 3ϕ-IM speed controlled system with PID controller designed by the CS tuning rule can regulate the step load disturbance. It gives M_{p_reg} = 11.28% and t_{reg} = 0.80 sec.

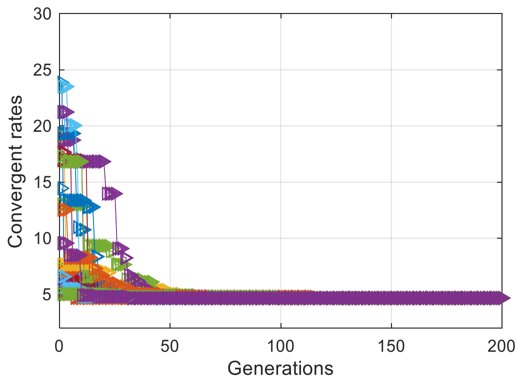


Figure 26. Convergent rates of CS-based PID design over 50 trials

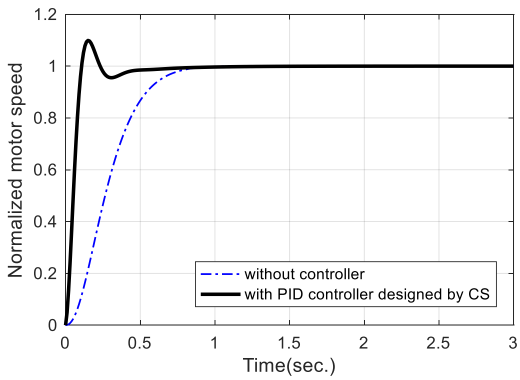


Figure 27. Step-input responses of the 3ϕ-IM speed controlled system without and with the PID controller designed by the CS

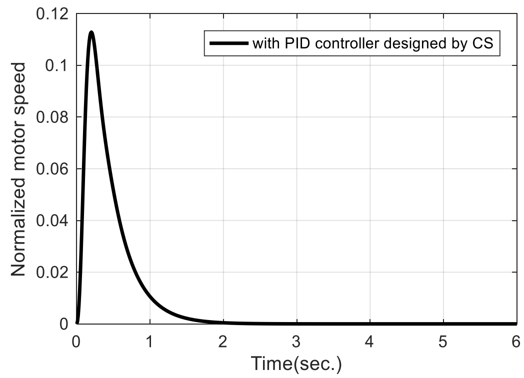


Figure 28. Step-disturbance response of the 3ϕ-IM speed controlled system with the PID controller designed by the CS

$$G_c(s)|_{PID\_CS} = 7.0071 + \frac{18.8821}{s} + 0.4709s \tag{32}$$

E. PI$^\lambda$D$^\mu$ Controller Design by CS

Application of the CS to design an optimal PI$^\lambda$D$^\mu$ controller for 3ϕ-IM speed control system can be represented by the block diagram in Figure 29. The objective function f, sum-squared error between the reference speed $R(s)$ and the actual speed $C(s)$ in (33), will be fed to the CS to be minimized by searching for the appropriate values of the PI$^\lambda$D$^\mu$ parameters, i.e. K_p, K_i, K_d,

λ and μ subject to inequality constraint functions satisfying to the predefined response specifications as stated in (34).

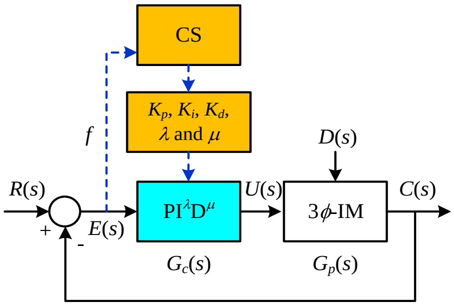


Figure 29. CS-based $PI^{\lambda}D^{\mu}$ controller design by CS

$$Min\ f(K_p, K_i, K_d, \lambda, \mu) = \sum_{i=1}^{N}[r_i - c_i]^2 \tag{33}$$

$$\left.\begin{aligned} Subject\ to\quad & t_r \leq 0.2\ sec., \\ & M_p \leq 10.0\ \%, \\ & E_{ss} \leq 0.01\ \%, \\ & 0 < K_p \leq 10, \\ & 0 < K_i \leq 20, \\ & 0 < K_d \leq 1.0, \\ & 0 < \mu < 1.0, \\ & 0 < \lambda < 1.0 \end{aligned}\right\} \tag{34}$$

Like a case of the CS-based PID controller design, the CS algorithm was coded by MATLAB. In this case, the $PI^{\lambda}D^{\mu}$ is implemented by MATLAB with FOMCON toolbox [40, 41, 42] where Oustaloup's approximation is realized for fractional order numerical simulation. Number of cuckoos n = 40 and fraction p_a = 0.2 are set according to recommendations of Yang and Deb [19, 20]. The maximum generation Max_Gen = 200 is then set as TC. 50 trials are conducted to find the best solution (optimal $PI^{\lambda}D^{\mu}$ controller for the 3ϕ-IM speed control system).

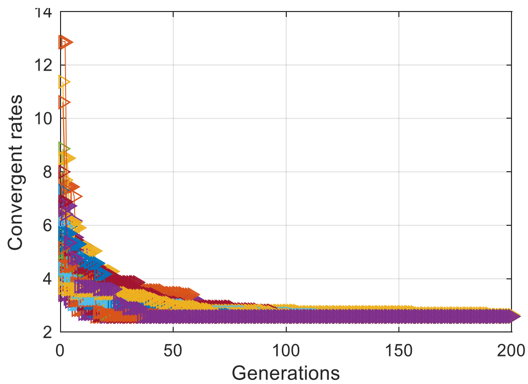


Figure 30. Convergent rates of CS-based $PI^{\lambda}D^{\mu}$ design over 50 trials

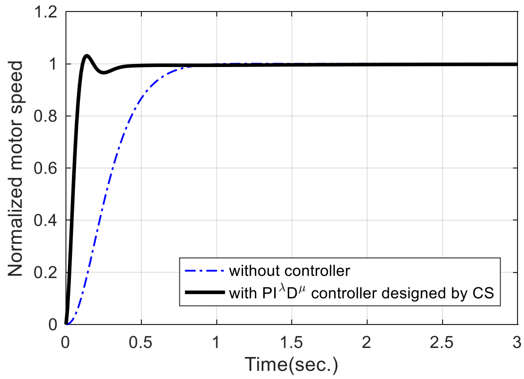


Figure 31. Step-input responses of the 3ϕ-IM speed controlled system without and with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS

When the search process stopped, the CS can successfully provide the optimal parameters of the PI$^{\lambda}$D$^{\mu}$ controller for the 3ϕ-IM speed control system as expressed in (35). The convergent rates of the objective functions in (33) associated with inequality constraint functions in (34) proceeded by the CS over 50 trials are depicted in Figure 30. The step-input responses of the 3ϕ-IM speed control system without and with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS are depicted in Figure 31, while the step-disturbance response of the 3ϕ-IM speed control system with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS is plotted in Figure 32.

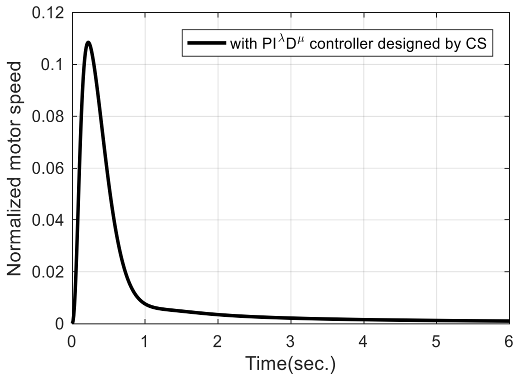


Figure 32. Step-disturbance response of the 3ϕ-IM speed controlled system with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS

$$G_c(s)|_{PI^{\lambda}D^{\mu}\_CS} = 5.5979 + \frac{19.9749}{s^{0.91}} + 0.6501\,s^{0.99} \tag{35}$$

From Figure 31, the step-input response of the 3ϕ-IM speed controlled system with PI$^{\lambda}$D$^{\mu}$ controller designed by the CS tuning rule provides t_r = 0.11 sec, M_p = 3.11%, t_s = 0.32 sec. and e_{ss} = 0. From Figure 32, the 3ϕ-IM speed controlled system with PI$^{\lambda}$D$^{\mu}$ controller designed by the CS tuning rule can regulate the step load disturbance. It yields M_{p_reg} = 10.85% and t_{reg} = 0.73 sec.

For comparison, the step-input responses and the step-disturbance responses of the 3ϕ-IM speed control system without, with the PID controllers designed by ZN, CC and CS and with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS are plotted in Figure 33 and Figure 34, respectively. Entire results of step-input responses are summarized in Table 4, while those of step-disturbance responses are summarized in Table 5. From Figure 33 – 34 and Table 4 – 5, it was found that the 3ϕ-IM speed controlled systems with PID and PI$^{\lambda}$D$^{\mu}$ controllers are stable. The

PI$^{\lambda}$D$^{\mu}$ controller designed by the CS can yield faster and smoother speed response than others, significantly.

Table 4. Entire step-input responses

Controllers	Step-input responses			
	t_r (sec.)	M_p (%)	t_s (sec.)	e_{ss}
PID-ZN	0.09	56.14	1.23	0.00
PID-CC	0.09	61.24	1.62	0.00
PID-CS	0.11	9.05	0.43	0.00
PI$^{\lambda}$D$^{\mu}$-CS	0.11	3.11	0.32	0.00

Table 5. Entire step-disturbance responses

Controllers	Step-input responses		
	M_{p_reg} (%)	t_{reg} (sec.)	e_{ss}
PID-ZN	9.81	0.54	0.00
PID-CC	9.54	0.53	0.00
PID-CS	11.28	0.80	0.00
PI$^{\lambda}$D$^{\mu}$-CS	10.85	0.73	0.00

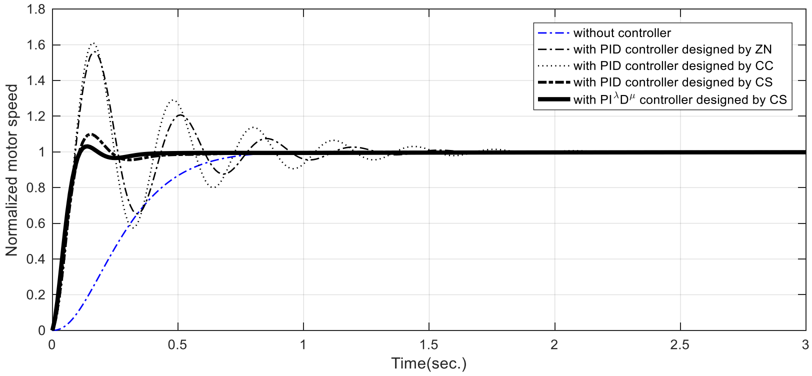


Figure 33. Step-input responses of the 3ϕ-IM speed controlled system without and with PID and PI$^{\lambda}$D$^{\mu}$ controllers

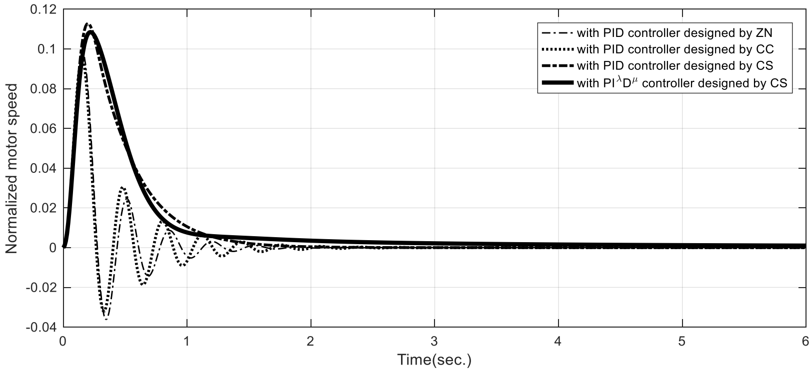


Figure 34. Step- disturbance responses of the 3ϕ-IM speed controlled system with PID and PI$^{\lambda}$D$^{\mu}$ controllers

6. Stability Analysis

Based on Matignon's stability theorem associated with Riemann sheet [13, 14, 32, 33, 34, 35], the stability analysis of the 3ϕ-IM speed system controlled with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS is investigated in this section. Referring to Figure 18, the closed loop transfer function is formulated as shown in (36).

$$\frac{C(s)}{R(s)}=\frac{G_c(s)G_p(s)}{1+G_c(s)G_p(s)} \tag{36}$$

By substituting the plant model $G_p(s)$ in (26) and the PI$^{\lambda}$D$^{\mu}$ controller model $G_c(s)$ in (35), the closed loop transfer function in (36) becomes the transfer function in (37).

$$\frac{C(s)}{R(s)}=\frac{884.14s^{1.90}+7{,}613.1s^{0.91}+27{,}166}{s^{3.91}+39.16s^{2.91}+398.14s^{1.91}+884.14s^{1.90}+8{,}973.1s^{0.91}+27{,}166} \tag{37}$$

The characteristic equation and the corresponding commensurate characteristic equation of system are performed in (38) and (39), respectively.

$$s^{3.91}+39.16s^{2.91}+398.14s^{1.91}+884.14s^{1.90}+8{,}973.1s^{0.91}+27{,}166=0 \tag{38}$$

$$s^{\frac{391}{100}}+39.16s^{\frac{291}{100}}+398.14s^{\frac{191}{100}}+884.14s^{\frac{190}{100}}+8{,}973.1s^{\frac{91}{100}}+27{,}166=0 \tag{39}$$

When $w=s^{1/100}$ and $m=100$ (LCM), a polynomial of complex variable w is then obtained as expressed in (40).

$$w^{391}+39.16w^{291}+398.14w^{191}+884.14w^{190}+8{,}973.1w^{91}+27{,}166=0 \tag{40}$$

All 391 roots (poles) of a polynomial in (40) can be obtained by MATLAB. The Riemann surface of $w=s^{1/100}$ is depicted in Figure 35. All 391 roots are then plotted in w-plane as shown in Figure 36. It was found that all roots are satisfy the condition $|\arg(w_i)| > \pi/2m > \pi/200$ radian = 0.0157 radian ($\approx$ 0.9 degree). It means that the 3ϕ-IM speed controlled system with the PI$^{\lambda}$D$^{\mu}$ controller designed by the CS is stable.

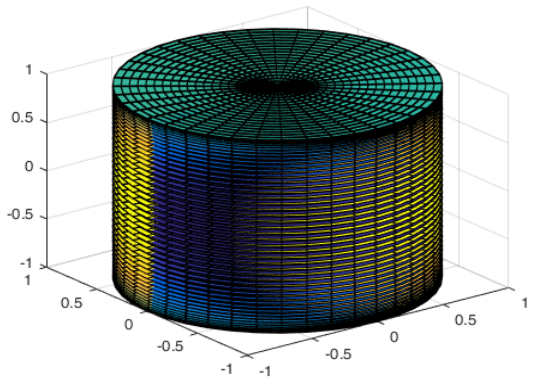


Figure 35. Riemann surface of $w=s^{1/100}$

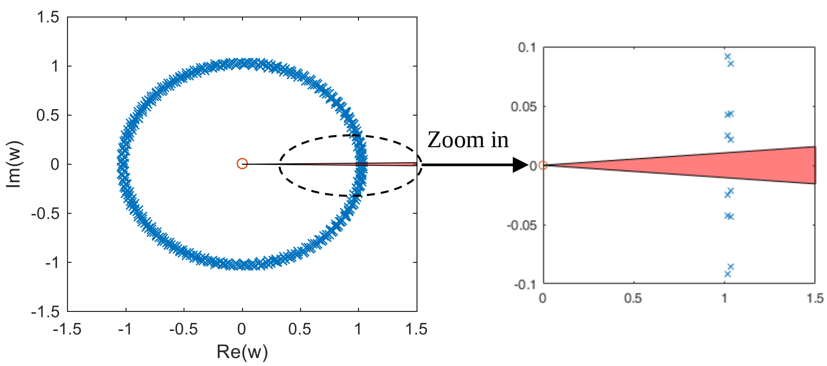


Figure 36. Stability investigation of 3ϕ-IM speed system controlled with the PI$^{\lambda}$D$^{\mu}$ controller

7. Conclusions

In this paper, designing an optimal $PI^{\lambda}D^{\mu}$ controller for the 3ϕ-IM speed controlled system by the CS has been proposed. Characterized by five parameters, the $PI^{\lambda}D^{\mu}$ controller could perform the better responses once compared with the conventional PID controller. Effects of the $PI^{\lambda}D^{\mu}$ parameter tuning on the system responses have been studies and summarized for control engineers. It was noticed that λ will reinforce the integral gain K_i to decrease or eliminate the steady-state error, while μ will reinforce the derivative gain K_d to speed up and decrease the oscillation of the system response. The CS algorithm has been briefly reviewed. The CS-based $PI^{\lambda}D^{\mu}$ controller design framework has been formulated according to modern optimization context. By numerical simulation with MATLAB and FOMCON toolbox, five parameters of the $PI^{\lambda}D^{\mu}$ controller have been successfully optimized by the CS meeting the predefined response specifications as inequality constraint functions. As simulation results, it was found that the $PI^{\lambda}D^{\mu}$ controller designed by the CS performs superior to others with faster and smoother speed response. The stability analysis of the 3ϕ-IM speed controlled system has been also investigated. It was noticed that the 3ϕ-IM speed controlled system with the $PI^{\lambda}D^{\mu}$ controller designed by the CS is stable based on Matignon's stability theorem associated with Riemann sheet. For the future research, the fractional-order PIDA ($PI^{\lambda}D^{\mu}A^{\gamma}$) controller designed by the CS (or other potential metaheuristics) will be alternatively conducted to extend the fractional order control systems. In addition, multobjective $PI^{\lambda}D^{\mu}$/ $PI^{\lambda}D^{\mu}A^{\gamma}$ controllers design framework will be studied.

8. References

[1] I. Podlubny, Fractional-Order Systems and Fractional-Order Controllers, UEF-03-94, Slovak Academy of Sciences, Kosice, 1994.

[2] I. Podlubny, "Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers," *IEEE Transactions on Automatic Control*, vol. 44, no. 1, 1999, pp. 208–214.

[3] C. A. Monje, B. M. Vinagre, V. Feliu, and Y. Chen, "Tuning and auto-tuning of fractional order controllers for industry applications," *Control Engineering Practice*, vol. 16, 2008, pp. 798–812.

[4] M. Zamani, M. Karimi-Ghartemani, N. Sadati, and M. Parniani, "Design of a fractional order PID controller for an AVR using particle swarm optimization," *Control Engineering Practice*, vol. 17, 2009, pp. 1380–1387.

[5] D. Xue, C. Zhao, and Y. Q. Chen, "Fractional order PID control of a DC-motor with elastic shaft: a case study," *Proc. of the 2006 American Control Conference*, 2006, pp. 3182–3187.

[6] A. J. Calderón, B. M. Vinagre, and V. Feliu, "Fractional order control strategies for power electronic buck converters," *Signal Processing*, vol. 86, no. 10, 2006, pp. 2803–2819.

[7] S. K. Mishra and D. Chandra, "Stabilization and tracking control of inverted pendulum using fractional order PID controllers," *Journal of Engineering*, vol. 2014, 2014, pp. 1–9.

[8] Q. Gao, J. Chen, L. Wang, S. Xu, and Y. Hou, "Multiobjective optimization design of a fractional order PID controller for a gun control system," *The Scientific World Journal*, vol. 2013, 2013, pp. 1–8.

[9] D. Valério and J. C. Sáda, "Tuning of fractional PID controllers with Ziegler-Nichols type rules," *Signal Processing*, vol. 86, no. 10, 2006, pp. 2771–2784.

[10] Y. Chen, T. Bhaskaran, and D. Xue, "Practical tuning rule development for fractional order proportional and integral controllers," *Journal of Computational and Nonlinear Dynamics*, vol. 3, 2008.

[11] R. Caponetto, L. Fortuna, and D. Porto, "A new tuning strategy for a non integer order PID controller," *Fractional Differentiation and its Applications, Bordeaux*, 2004.

[12] C. Zhao, D. Xue, and Y. Chen, "A fractional order PID tuning algorithm for a class of fractional order plants," *Proc. of the International Conference on Mechatronics & Automation, Niagara Falls*, 2005.

[13] Y. Q. Chen, I. Petráš, and X. Dingyü, "Fractional order control - a tutorial," *Proc. of the American Control Conference*, 2009, pp. 1397–1411.

[14] P. Shah and A. Agashe, "Review of fractional PID controller," Mechatronics, vol. 38, 2016, pp. 29–41.

[15] R. C. Dorf and R. H. Bishop, Modern Control Systems, 13th-edition, Pearson Prentice Hall, 2017.

[16] K. Ogata, Modern Control Engineering, 5th-edition, Pearson Prentice Hall, 2010.

[17] V. Zakian, Control Systems Design: A New Framework, Springer-Verlag, 2005.

[18] V. Zakian and U. Al-Naib, "Design of dynamical and control systems by the method of inequalities," *Proc. of the IEE International Conference*, vol. 120, 1973, pp.1421–1427.

[19] X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," Proc. of the World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), 2009, pp. 210–214.

[20] X. S. Yang, "Cuckoo search and firefly algorithm: overview and analysis," Studied in Computational Intelligence: Cuckoo search and Firefly Algorithm Theory and Applications, Springer, 2014, pp. 1–26.

[21] B. Ernst, M. Bloh, J. R. Seume, and A. G. González, "Implementation of the cuckoo search algorithm to optimize the design of wind turbine rotor blades," *Proc. of the European Wind Energy Association (EWEA)* 2012, 2012.

[22] M. Khodier, "Optimisation of antenna arrays using the cuckoo search algorithm," *IET Microwaves, Antennas & Propagation*, vol. 7, no. 6, 2013, pp. 458–464.

[23] S. Rangasamy and P. Manickam, "Stability analysis of multimachine thermal power systems using nature inspired modified cuckoo search algorithm," *Turkish Journal of Electrical Engineering & Computer Sciences*, DOI: 10.3906/elk-1212-39, 2013.

[24] A. Ouaarab, B. Ahiod, and X. S. Yang, "Discrete cuckoo search algorithm for the travelling salesman problem," *Neural Computing and Applications*, 2013, pp. 1–11.

[25] A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with computers*, vol. 29, no. 1, 2013, pp. 17–35.

[26] M. Dhivya and M. Sundarambal, "Cuckoo search for data gathering in wireless sensor networks", *International Journal of Mobile Communications*, vol. 9, no. 6, 2011, pp. 642–656.

[27] M. K. Marichelvam, "An improved hybrid cuckoo search (ihcs) metaheuristics algorithm for permutation flow shop scheduling problems," *International Journal of Bio-Inspired Computation*, vol. 4, no. 4, 2012, pp. 200–205.

[28] S. Hanoun, S. Nahavandi, D. Creighton, and H. Kull, "Solving a multiobjective job shop scheduling problem using pareto archived cuckoo search, *Proc. of the 17th IEEE Conference on Emerging Technologies & Factory Automation (ETFA)*, 2012, pp. 1–8.

[29] S. Hlungnamthip and D. Puangdownreong, "Model order reduction of linear time–invariant dynamic systems via cuckoo search," *KMUTT Research & Development Journal*, vol. 40, no. 2, 2017, pp. 237–253.

[30] D. Puangdownreong, A. Nawikavatan, and C. Thammarat, "Optimal design of I–PD controller for DC motor speed control system by cuckoo search," *Procedia Computer Science*, vol. 86, 2016, pp. 83–86.

[31] I. Fister Jr., X. S. Yang, D. Fister, and I. Fister, "Cuckoo search: a brief literature review," Cuckoo Search and Firey Algorithm: *Theory and Applications, Studies in Computational Intelligence*, vol. 516, 2014, pp. 49– 62.

[32] C. A. Monje, Y. Q. Chen, B. M. Vinagre, D. Xue, and V. Feliu, Fractional-order Systems and Controls Fundamentals and Applications, Springer-Verlag, 2010.

[33] D. Matignon, "Stability result on fractional differential equations with applications to control processing," *Proc. of the International Meeting on Automated Compliance Systems and the International Conference on Systems, Man, and Cybernetics (IMACS-SMC '96)*, 1996, pp. 963–968.
[34] D. Matignon, "Stability properties for generalized fractional differential systems," Proc. of the Fractional Differential Systems: Models, Methods and Applications, 1998, pp. 145–158.
[35] I. Petráš, "Stability of fractional-order systems with rational orders: a survey," *Fractional Calculus & Applied Analysis*, vol. 12, no. 3, 2009, pp. 269–298.
[36] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," Transactions on ASME, vol. 64, 1942, pp. 759–768.
[37] J. G. Ziegler and N. B. Nichols, "Process lags in automatic control circuits," *Transactions on ASME*, vol. 65, 1943, pp. 433–444.
[38] G. H. Cohen and G. A. Coon, "Theoretical consideration of retarded control," Transactions on ASME, vol. 75, 1953, pp. 827–834.
[39] MathWorks, System Identification Toolbox™, MATLAB-R2018b, 2017.
[40] A. Tepljakov, E. Petlenkov, and J. Belikov, "FOMCON: fractional-order modeling and control toolbox for MATLAB," *Proc. of the 18th International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES '11)*, 2011, pp. 684–689.
[41] A. Tepljakov, E. Petlenkov, and J. Belikov, "FOMCON: a MATLAB toolbox for fractional-order system identification and control," *International Journal of Microelectronics and Computer Science*, vol. 2, no. 2, pp. 51–62, 2011.
[42] A. Tepljakov, E. Petlenkov, and J. Belikov, FOMCON toolbox, [Available Online: http://www. fomcon.net], 2011.

Chaiyo Thammarat was born in Chonburi, Thailand, in 1953. He received the B.Eng. degree in electrical engineering from King Mongkut Institute of Technology Thonburi (KMITT), Bangkok, Thailand, in 1976, and in M.Eng. degree in electrical engineering from Lamar University, Beuamont, Texas,U.S.A, in 1986. From 1979 to 1998, he worked with Royal Thai Navy, Thailand. Since 1998, he has been with Department of Electrical Engineering, Faculty of Engineering, Southeast Asia University (SAU), Bangkok, Thailand, where he is an assistant professor of electrical engineering. Now, he is a Ph.D. candidate of Doctor of Philosophy Program in Electrical Engineering at SAU. His research interests include intelligent control system and novel metaheuristics as well as their applications.

Deacha Puangdownreong was born in Phra Nakhon Si Ayutthaya, Thailand, in 1970. He received the B.Eng. degree in electrical engineering from Southeast Asia University (SAU), Bangkok, Thailand, in 1993, the M.Eng. degree in control engineering from King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 1996, and the Ph.D. degree in electrical engineering from Suranaree University of Technology (SUT), Nakhon Ratchasima, Thailand, in 2005, respectively. Since 1994, he has been with the Department of Electrical Engineering, Faculty of Engineering and Graduate School, Southeast Asia University, where he is currently an associated professor of electrical engineering. His research interests include control synthesis and identification, novel metaheuristics as well as their applications.