An Identification and Tuning Method for Integrating Processes with Deadtime and Inverse Response

Do Cao Trung

Abstract— *The methods of process identification and controller tuning for integrating processes with dead time and inverse response are presented in this study. Firstly, a numerical procedure is proposed to identify the processes from its step response data by using models of integrating plus first order with deadtime and a zero (IFOPDTZ) or integrating plus first order with dead time (IFOPDT). Secondly, the robust-based controller is introduced for controller tuning. Finally, some examples are given for illustration and comparison. The identification procedure contributes to enrich the inverse response process identification works and the tuning methodology gives the opportunity to preset the system robustness index, which is the main interest of studies addressing controller tuning for integrating inverse response processes.*

Keywords— **Integrating processes with inverse response, process identification, PID tuning, integrating plus first order plus dead time with a zero (IFOPDTZ), integrating plus first order plus dead time (IFOPDT).**

1. INTRODUCTION

Integrating processes with deadtime and inverse response are the self-imbalance ones, which react inversely to input change at the initial period. The trend is the opposite direction of the process final steady state. The typical process open-loop step response is shown in Fig. 1 [1]. The processes will have inverse response with input pulse at the beginning.

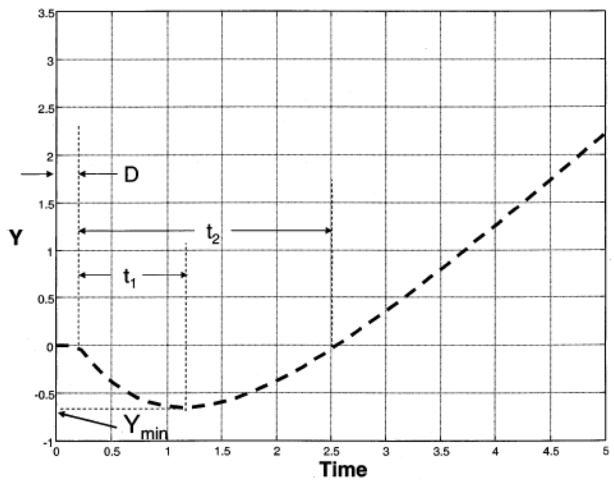

Fig.1. Step response of self-imbalance process with inverse response [1].

For example, the water boiler drum level process in thermal power plant is one of the most popular of this kind. Fig. 2 shows the step dynamic response of drum water level of one steam generator of a nuclear power plant with the load range of 5% to 100% of the rated output [2]. At the initial period, when feed-water is fed to the boiler drum, the level of the drum is reduced. This phenomenon is known as "Swell and Shrink" effect.

The processes are controlled by PI/PID (Proportional-Integral-Derivative) controllers, which will be tuned from identified models or be synthesized directly. In both ways, IFOPDTZ model is used popularly.

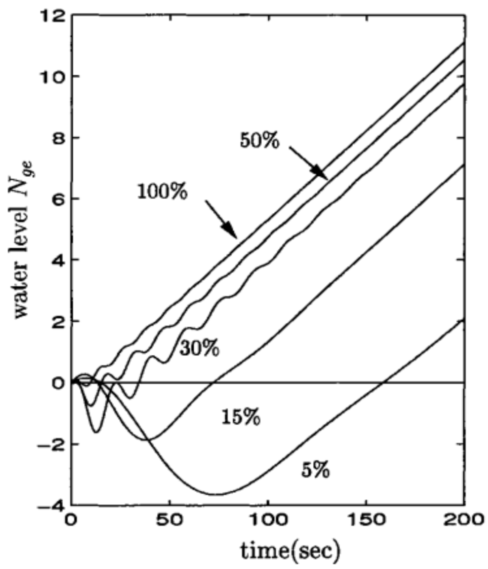

Fig.2. Water drum level step response [2].

To identify the processes, their open-loop step responses are used. The result of the identification will decide the quality of PID setting and control system. Previously, the manual analysis of step response curve was applied commonly for process model, which caused the drawback of poor identifying quality. Factually, in the literature, there have not been many identification methods to be given. In [1], William L. Luyben proposed a procedure including two separate phases, which were the process identification technique by IFOPDTZ model

---

Do Cao Trung is with Hanoi University of Science and Technology, No. 1, Dai Co Viet Street, Hanoi, Vietnam. Phone: +84-913689199; E-mail: trung.docao@hust.edu.vn.

based on step open-loop response and the frequency-domain controller tuning approach using Matlab software. In [3], Danying Gu at al. expressed relay feedback technique for process identification in combination with the tuning approach of H_∞ optimization and internal model control (IMC) theory. While Luyben's identification has been a typical, useful technique applied in many works of controller tuning study, the Gu's one shows the limitation and inconvenience in application.

In tuning work, the proposal of Luyben [1] has a significant drawback of poor system robustness, which is one of the greatest interested features of PID setting for integrating processes with inverse response. In [4], Chi-Tsung Huang at al. presented a direct synthesis method, which improved the system robustness margin in comparison to that of Luyben approach, but it still is a poor index. Moreover, this method also results an oscillatory closed-loop output. In [5], Jeng JC at al. designed controller for the processes based on a Smith-type compensator for non-minimum phase dynamics. In [6], Kaya proposed a PI–PD tuning technique developed Smith predictor scheme for improvement of the process closed-loop performances. The work is continuously developed in [7] to upgrade its performances.

For integrating processes without inverse response, there are a lot of papers concerning PID controller tuning. A. Ali at al. [8] developed the method of direct synthesis with the same disadvantage of paper [4]. Q. Liu at al. [9] proposed an analytical internal model controller (IMC) PID tuning rules. Although the proposal provides the higher system robustness, it still results an oscillatory output of closed-loop response. Anil at al. [10] presented a PID controller technique using direct synthesis approach, Ajmeri at al. [11] and many other authors also applied direct synthesis method for PID tuning of three integrating process forms with large time delays.

While there are plenty of tuning methods for integrating process, the number is only a few for integrating processes with deadtime and inverse response. However, it is still many more comparing to the number of identification approaches. There are few works for identification of integrating processes with deadtime and inverse response. The mentioned tuning works and others use IFOPDTZ model for the process representative, few papers addressing the complex integrating processes with deadtime and inverse response like [3], which also applied Luyben's procedure to approximate them to IFOPDTZ model. The problem narrows the application of tuning methods for practical processes due to the lack of identified transfer function. Moreover, in literature, the tuning studies for both kinds of integrating processes achieved significant system robustness improvement, but there are difficulties for designers get the achievement of those techniques in application.

In this research, a method including a numerical identification technique and a "robust-based" tuning procedure is proposed. The paper is organized as follows. Section 2 presents a numerical identification method for integrating process with deadtime and inverse response using IFOPDTZ or IFOPDT models of which parameters are defined by "step-over-cleft" algorithm [12]. Section 3 develops the PID controller tuning method based on robust control viewpoint [13] followed by examples given in section 4 and finally, section 5 shows the conclusion.

2. MODEL FOR PROCESS IDENTIFICATION

Identify by measuring unit response

Normally, the unit response curve (Fig. 2) will be measured and used to identify processes by IFOPDTZ model with transfer function as below,

$$O_{IFOPDTZ}(s) = \frac{K(1-cs)}{s(1+Ts)} e^{-\tau s} \qquad (1)$$

Where, K – gain factor; T – lag constant; τ - dead time; c – inverse element; s – complex variable.

$$K > 0, \tau \geq 0, T, c \geq 0 \qquad (2)$$

Besides, by this method, the curve is also able to identify by IFOPDT model with transfer function,

$$O_{IFOPDT}(s) = \frac{K}{s(1+Ts)} e^{-\tau s} \qquad (3)$$

where: K – gain factor; T – lag constant; τ - dead time; s – complex variable.

$$K > 0, \tau \geq 0, T \geq 0 \qquad (4)$$

Using IFOPDT model means that the inverse response is ignored.

Model reduction

Practically, the processes have high order. There is a popular situation is a complex transfer function given and need to be deduced. In this case, commonly, the type of transfer function will be,

$$O(s) = \frac{(1-cs)B(s)}{sA(s)} e^{-\tau s} \qquad (5)$$

where A(s), B(s) are polynomials of s variable.

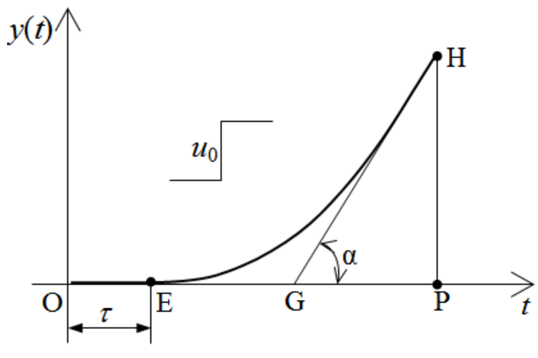

Fig. 3. Normal integrating process.

Generally, there were two ways widely used for function deduction were to model by IFOPDTZ using process step response in Fig. 2 or to approximate the inverse element converting (5) become normal integrating processes. The inverse, commonly, is able to approximate by using the formula: $1-cs=e^{-cs}$, then model (5) becomes.

$$O(s)=\frac{(1-cs)\,B(s)}{sA(s)}e^{-\tau s}\approx\frac{B(s)}{sA(s)}e^{-(\tau+c)s} \qquad (6)$$

Processes with functions type (6) have step dynamic response is shown in Fig. 3.

This curve will be identified by model (3) meaning that the function (5) will be deduced to the IFOPDT model.

3. DEFINE PARAMETERS OF MODEL

Target function

In model (1), if c = 0 then (1) will become (3), so that model (1) is able to be considered to be the common transfer function for normal integrating processes.

Its unit step response is:

$$y(t)=L^{-1}\{Y(s)\}=L^{-1}\{U(s)O_{IFOPDTZ}(s)\}=$$

$$L^{-1}\left\{\frac{u_0}{s}\frac{K(1+cs)}{s(1+Ts)}e^{-s}\right\}=Ku_0\left[t-\tau-(T-c)\left(1-e^{-\frac{t-\tau}{T}}\right)\right]$$

In the time t_i, the value of model is:

$$y(t_i,X)=Ku_0\left[t_i-\tau-T\left(1-e^{-\frac{t_i-\tau}{T}}\right)\right] \qquad (7)$$

Process values of step response (in Fig. 2 or Fig. 3) are y_i $(i=\overline{1,M})$. The constant vector X = {K, T, c, τ} of (1) is chosen based on condition,

$$F(X)=\sum_{i=1}^{M}\left[y(t_i,X)-y_i\right]^2\to\min_X \qquad (8)$$

where, N – Number of measuring points.

X also must be satisfied the bound (2). The function (8) together with bound (2) is driven to unconstraint optimization problem:

$$J(X)=F(X)+p\Pi(X)\to\min_X \qquad (9)$$

where:

$$P(X)=p\{[|T|-T]^2+[|c|+c]^2+[|\tau|-\tau]^2+[|K|-K]^2\}$$

The penalty coefficient p is set equally 10^3.

It is not difficult to prove that (8) and (9) have the same roots. Indeed, if any parameter of X does not satisfy (4) then Π(X) > 0 and pΠ(X) → ∞ making J(X) → ∞, this means the root of (9) is not able to find. Otherwise, pΠ(X) ≡ 0, means J(X) ≡ F(X).

Solving target function

The unconstrained optimization problems (9) will be solved by "Cleft-over step" algorithm [2]. The algorithm requires a constant start vector.

The start vector X_0 = {K_0, T_0, c_0, τ_0} are set as start point of "Cleft-over step" algorithm to solve unconstrained optimization problems (9). The start vector is very important to help the algorithm find the roots fast and exactly. This start vector will be set based on the step response analyzed in Fig. 4.

H is the end and HG is tangent of the curve.

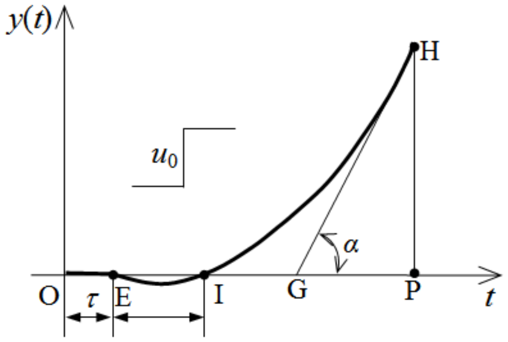

Fig.4. Inverse response analysis.

Start vector X_0 for model (1) is set as below:

$$K_0\approx tg\alpha/u_0=\frac{HP/GP}{u_0},\ \tau_0\approx OE,\ T_0\approx OG-\tau_0,\ c_0\approx EI$$

If the response curve in Fig. 4 is identified by model (3), the start X_0 vector will be set as:

$$K_0\approx tg\gamma/u_0=\frac{HP/GP}{u_0},\ \tau_0\approx OI,\ T_0\approx OG-\tau_0,\ c_0=0$$

In the iterative optimization steps, the variable c of vector X is not optimized (is remained by 0).

Nevertheless, in the case of model reduction or normal integrating process (no inverse) identification, the parameters y_i now is extracted from the curve as shown in Fig. 3 and the start vector X_0 now is set:

$$K_0\approx tg\gamma/u_0=\frac{HP/GP}{u_0},\ \tau_0\approx OE,\ T_0\approx OG,\ c_0=0,$$

The variable c also is not optimized.

4. TUNING CONTROLLER

Robust control viewpoint [13]

Simple closed-loop control is shown in Fig. 5 including: Process O(s), Controller R(s), Input z and Output y.

O(s) is general type: $O(s)=O_{PL}(s)e^{-\tau s}$ ($O_{PL}(s)$ is a rational fraction of variable s, τ - the dead time).

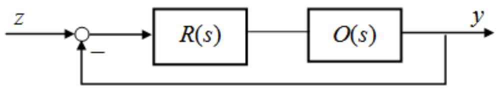

Fig.5. Simple closed-loop.

From Fig. 5, it is denoted $W_O(s) = R(s)O(s)$ being open-loop transfer function and $W_C(s) = W_O(s)/(1+W_O(s))$ being closed-loop one.

By replacing the approximation:

$$e^{\tau s} \simeq \left(1+\frac{\tau}{n}s\right)^n \ (n \in N)$$

Then, the closed-loop transfer function is able to alter as $W_C(s) = C(s)/D(s)$, D(s) now is the system characteristic polynomial and p pair of conjugate-complex numbers $s_i = -\beta_i \pm j\omega_i$ (i = 1 → p) satisfying the equation D(s) = 0 are system characteristic solutions. The factor $m_i = \beta_i/|\omega_i|$ is called as oscillation index of the solution s_i and $m_s = \min_i\{m_i\}$ is accepted as oscillation index of the system [13]. The concept "soft oscillation index" being a function of frequency is defined as below.

$$m = m(\omega) = m_0 \frac{1-e^{-\alpha|\omega|}}{\alpha|\omega|} \qquad (9)$$

where, α is softening factor (α ≥ τ > 0) and m_0 is initial value of oscillation index m which is a decreasing function of ω variable, m ∈ (m_0; 0) with ω ∈ (0; +∞).

Now, the complex number $s = -m|\omega|+j\omega$ is called "soft variable". With ω ∈ (–∞; +∞), the soft variable draws on the complex plane a symmetrical curve MON (Fig. 6), named "soft boundary". If $m = m_0$ (unchanged by ω) then $s = -m_0|\omega|+j\omega$ will draw out "hard boundary" COD being tangents of the soft boundary.

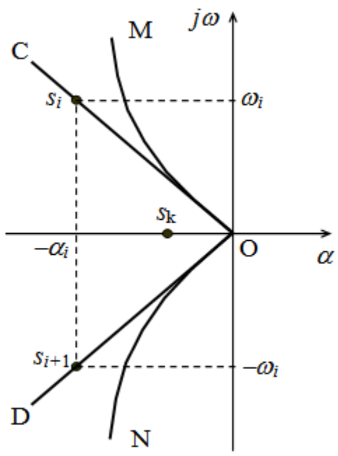

Fig.6. "Soft" and "hard" boundaries MON and COD.

A solution has oscillation index lower, equal or higher m will be located in the left, in or in the right of MON boundary, respectively. If a system in Fig. 5 has oscillation index m_s, then it will have all characteristic solutions located in the left or in the soft boundary created by soft variable $s = -m_s|\omega|+j\omega$. This is suggested that m_s is able to be considered to be system robustness index and soft boundary is robustness boundary. From this point of view, it is obvious that if m_s is able to be pre-chosen then the robustness of the system is fixed.

Robust-based controller

The system in Fig. 5 will have robustness index is +∞ if it has only one characteristic being a negative real number. It is able to choose:

$$W_C(s) = \frac{R(s)O(s)}{1+R(s)O(s)} = \frac{1}{1+\theta s},\ \theta > 0$$

where, θ is a real number being the lag constant. So,

$$W_O(s) = R(s)O(s) = \frac{1}{\theta s}.$$

From here we have a robust controller:

$$R(s) = \frac{1}{\theta s}O(s)^{-1} = \frac{1}{\theta s}O_{PL}(s)^{-1}e^{\tau s}.$$

To perform the controller, $e^{\tau s}$ will be eliminated. So,

$$R(s) = \frac{1}{\theta s}O_{PL}(s)^{-1} \qquad (10)$$

This R(s) is called robust-based controller. From this controller, it is given:

$$W_O(s) = R(s)O(s) = \frac{e^{-\tau s}}{\theta s} \qquad (11)$$

$$W_C(s) = \frac{W_O(s)}{1+W_O(s)} = \frac{e^{-\tau s}/\theta s}{1+e^{-\tau s}/\theta s} = \frac{e^{-\tau s}}{\theta s + e^{-\tau s}} \qquad (12)$$

Put $s = -m|\omega|+j\omega$ into (11) will have $W_O(-m\omega+j\omega)$ is named as the soft characteristic of open loop.

In R(s) formula (10), if process O(s) is given then it need only to determine the lag constant θ.

In [13], [14], a theorem for system of controller (10) is stated that: *"If the open-loop has the robustness index is m_s = m then the necessary and sufficient requirement that the closed-loop also has robustness index not less than m_s is the soft characteristic of the open loop does not cover the point (-1, j0)".*

The theorem is explained that if the open-loop has robustness index equal m, which means all its characteristic solutions are not located in the right of a fixed soft boundary. From this, the necessary and sufficient condition for all solutions of closed-loop system are not located on the right side of mentioned soft boundary is the soft characteristic of the open-loop does not cover the point (-1, j0).

Basing on the theorem, if the system in Fig. 5 with the controller (10) has a robustness index m_s to be prechoosen then the open-loop soft characteristic $W_O(-m_s|\omega|+j\omega)$ will cut the real axis of complex plane many times. By regulating the value of m_0, it is always able to force the first cut being the point (-1, j0). From the doing, the lag constant θ of R(s) will be calculated as below.

$$\theta = \frac{\tau e^{m_s(\frac{\pi}{2} - \mathrm{arctg} m_s)}}{(\frac{\pi}{2} - \mathrm{arctg} m_s)\sqrt{m_s^2 + 1}} \qquad (13)$$

The "hard" oscillation index m_0 is also able to inversely calculate from m_s by the formula as below.

$$m_0 = \frac{m_s(\frac{\pi}{2} - \mathrm{arctg} m_s)}{1 - e^{(\frac{\pi}{2} - \mathrm{arctg} m_s)}}$$

The formula points out that it is always able to find the index m_0 for every chosen value of m_s.

Robust-based PID controller for integrating processes

If process is IFOPDT (3), the robust-based controller will be:

$$R(s) = \frac{1}{\theta K}(1 + Ts) = \frac{1}{\theta K} + \frac{T}{\theta K}s = c_1 + c_2 s \qquad (14)$$

This PD controller will be supplemented integrating element (I) for the anti-disturbance, the PID robust-based controller now is.

$$R^*(s) = \frac{c_0}{s} + c_1 + c_2 s \qquad (15)$$

$$c_1 = \frac{1}{\theta K},\ c_2 = \frac{T}{\theta K}.$$

The integral element is set based on proportional and derivative coefficients of the controller.

$$c_0 = c_1(1 + B)^2 A \qquad (16)$$

where,

$$B = \frac{0{,}112\omega_c c_2/\tau}{c_1},\ A = \frac{0{,}233\omega_c c_1/\tau}{2c_1 + c_2/\tau}$$

$$\omega_c = \frac{1}{\tau}(\frac{\pi}{2} - \arctan m_s)\cdot$$

The m_s is often chosen in the range from 0.132 to 2.318 to achieve the optimal control error [14]. Especially, if m_s = 0.461 then the system will have the minimum integral of square error between output and input [14].

In practice, for integrating process with and without inverse response, the suitable value of system robustness index is 0.71.

5. EXAMPLES

This part will show some examples of system identification and tuning, the results are also compared with some other studies to monitor the effectiveness of proposed methods.

Identify by ISOPDTZ

The Integrating processes with inverse response is studied by Chi-Tsung Huang et al. [4] as,

$$O_1(s) = \frac{0.5(-0.5s + 1)}{s(0.4s + 1)(0.1s + 1)(0.5s + 1)}e^{-0.7s}$$

Unit step response of $O_1(s)$ is shown in Fig. 7. Get 32 points of this curve as in Table 1.

Table 1. 24 measuring points from $O_1(s)$ uint step response in summary

t_1	…	t_4	t_5	t_6	t_7
0	…	0.75	1.0	1.25	1.5
y_1	…	y_2	y_5	y_6	y_7
0	…	0	-0.0175	-0.0415	-0.0438
t_8	t_9	…	t_{22}	t_{23}	t_{24}
1.75	2.0		5.5	5.75	6
y_8	y_9	…	y_{22}	y_{23}	y_{24}
-0.0160	0.0394		1.6515	1.7765	1.9016

Here: $t_{i+1} = t_i + 0.25$ (i = 1 → 23).

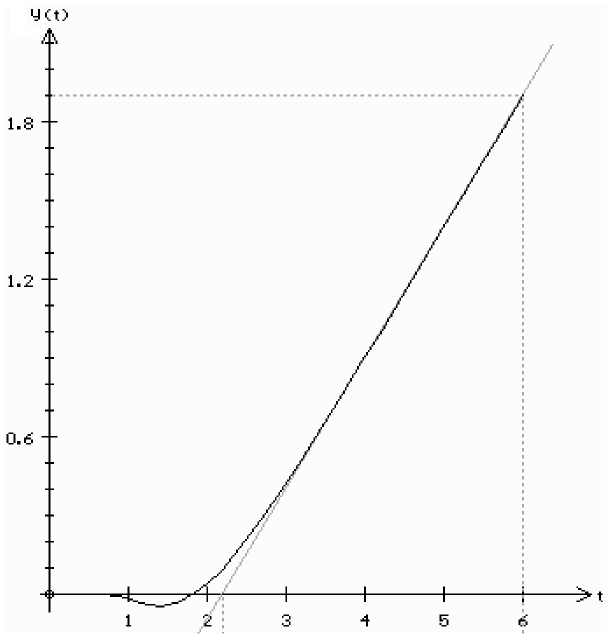

Fig. 7. Step dynamic response of $O_1(s)$.

Choose $X_0 = \{K_0, T_0, c_0, \tau_0\}$ based on curve in Fig. 7.

$K_0 = 0.5$; $\tau_0 = 0.8$; $T_0 = 0.6$; $c_0 = -1.0$

Table 1 parametters, X_0 are used for function (9). Finally, the roots are:

K = 0.5106; τ = 1.0107; T = 0.7774; c = - 0.4782

Minimun value of optimization function is J = 3.3E-3. The modeled transfer function is:

$$O_1^{'}(s)=\frac{0.5106(1-0.4782s)}{s(1+0.7774s)}e^{-1.0107s}$$

Chi-Tsung Huang et al. [4] used Luyben' method to appriximate $O_1(s)$ by

$$O_1^{''}(s)=\frac{0.5183(1-0.4699s)}{s(1+1.1609s)}e^{-0.81s}$$

Q. Liu et al. [9] modeled $O_1(s)$ by,

$$O_1^{'''}(s)=\frac{0.5183(1-0.4699s)}{s(1+1.1609s)}e^{-1.2799s}$$

Unit step responses of $O_1(s)$ and $O_1^{'}(s)$, $O_1^{''}(s)$, $O_1^{'''}(s)$ are shown in Fig. 8. The curves show that $O_1^{'}(s)$ is the most similar to $O_1(s)$.

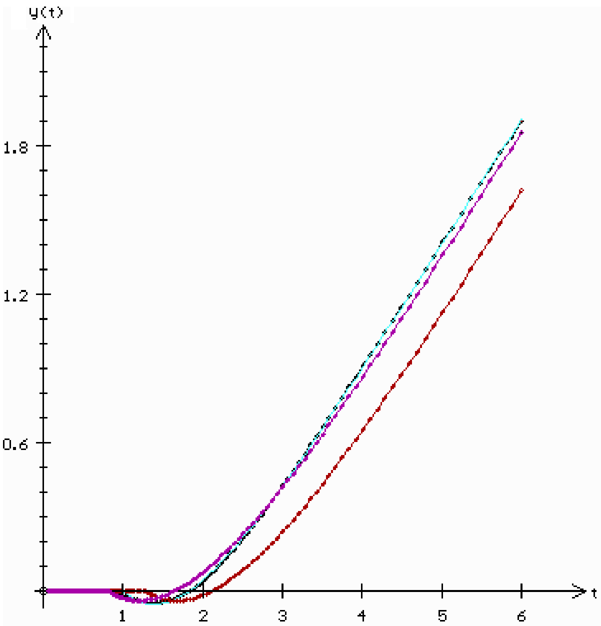

Fig. 8. Step responses of $O_1(s)$ and $O_1^{'}(s)$, $O_1^{''}(s)$, $O_1^{'''}(s)$

$O_1(s)$-black, $O_1^{'}(s)$-blue, $O_1^{''}(s)$-red, $O_1^{'''}(s)$-purple

From the $O_1^{'}(s)$ model, by approximating the inverse element and choosing m_s = 0.71, the PD robust-based controller (10) is:

$$R_1^{*}(s)=0.7816(1+0.7774s)$$

And the PID robust-based controller (15) is set as.

$$R_1^{**}(s)=0.7816(1+\frac{1}{23.972s}+0.7774s)$$

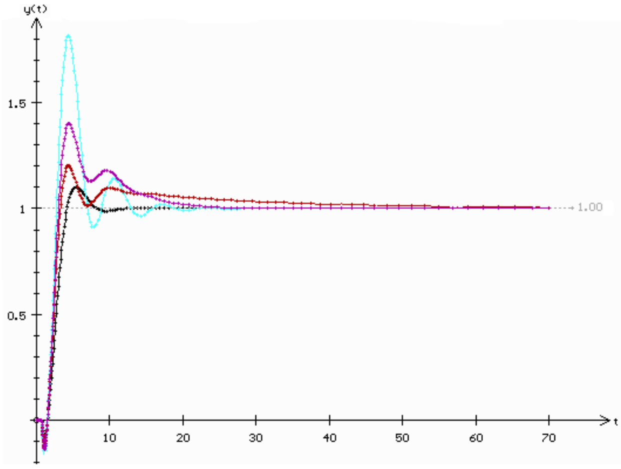

Fig. 9. Simple closed-loop

***Huang**-blue, **Liu**-purple, **Luyben**-red, $R_1^{*}(s)$-black*

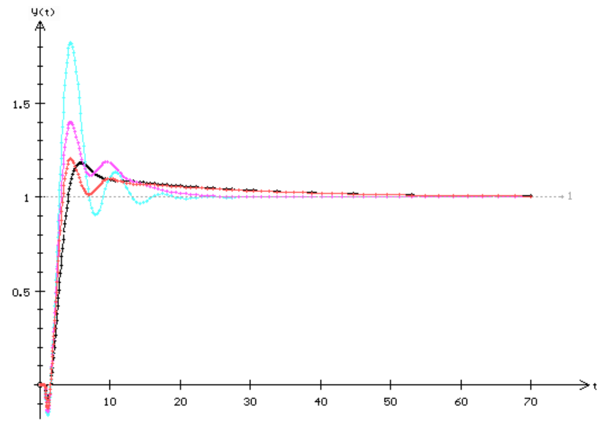

Fig. 10. Simple closed-loop

***Huang**-blue, **Liu**-purple, **Luyben**-red, $R_1^{**}(s)$-black*

Compare the control qualities of those $R_1^{*}(s)$ and $R_1^{**}(s)$ controllers with the other three controllers are set for $O_1(s)$, including:

- Chi-Tsung Huang et al. [4]

$$1.267(1+\frac{1}{5.782s}+0.925s)$$

- Q. Liu et al. [9]

$$0.97(1+\frac{1}{8.698s}+1.12s)$$

- Chi-Tsung Huang et al. [4] based on method of William L. Luyben [1],

$$0.867(1+\frac{1}{24s}+1.12s)$$

$O_1(s)$ closed-loop responses of the three controllers and $R_1^*(s)$, $R_1^{**}(s)$ to unit step input are shown in Fig. 9 and Fig. 10, respectively.

The robust-based PID controller $R_1^{**}(s)$ gives good closed-loop output qualities with no oscillation.

Reduction model

The inverse response of $O_1(s)$ is able to be approximate by $1-0.5s=e^{-0.5s}$ then $O_1(s)$ becomes,

$$O_2(s)=\frac{0.5}{s(0.4s+1)(0.1s+1)(0.5s+1)}e^{-1.2s}$$

It is the Integrating processes without inverse response. The unit step response of $O_2(s)$ is shown in Fig. 11. It will be modeled by (3).

Get 24 measuring points of the curve in table 2.

Table 2. 24 measuring points from O2(s) uint step response in sumary

t_1	…	t_6	t_7	…	t_{24}
0	…	1.25	1.5	…	6
y_1	…	y_6	y_7	…	y_{24}
0	…	0	0.0040	…	1.9016

Here: $t_{i+1} = t_i + 0.25$ ($i = 1 \rightarrow 23$).

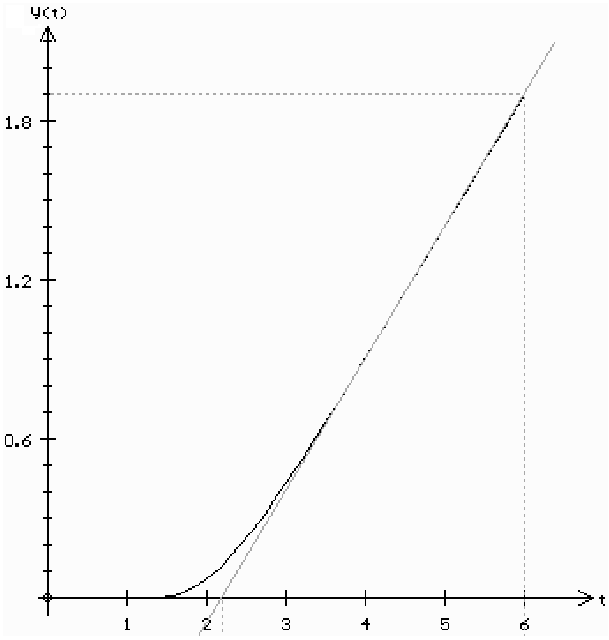

Fig. 11. Step dynamic response of O2(s)

Choose $X_0 = \{K_0, T_0, c_0, \tau_0\}$ based on curve in Fig. 11 as below,

$$K_0 = 0.5;\ T_0 = 0.7;\ c_0 = 0;\ \tau_0 = 1.5$$

where, c is set unchangingly.

The identified model is:

$$O_2^{'}(s)=\frac{0.5019}{s(1+0.7042s)}e^{-1.5128s}$$

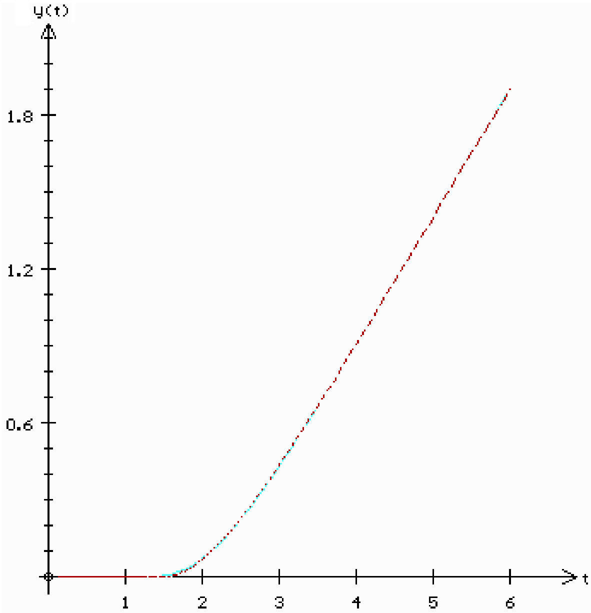

Fig. 12. Unit step responses of O2(s) and O'2(s)

O2(s)_Green, O2'(s)_ Red (Dash)

The unit step responses and frequency characteristics of $O_2^{'}(s)$ and $O_2(s)$ in Fig. 12 and 13.

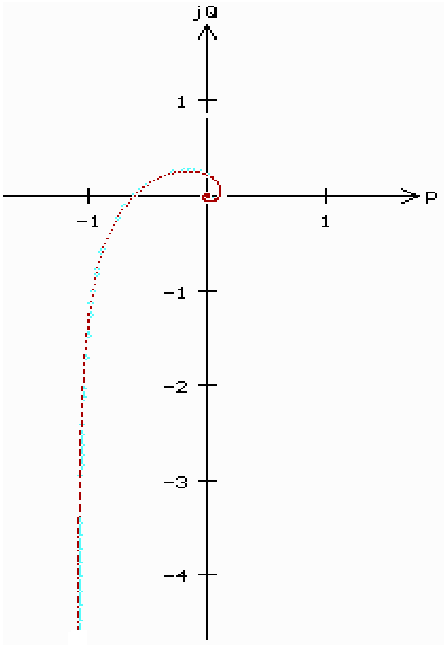

Fig. 13. Frequency characteristics of O2(s)

O2(s)_Green, O2'(s)_ Red (Dash)

Robust-based controllers (10) of $O_2(s)$ and $O_2^{'}(s)$ are set as below.

For $O_2(s)$:

- Choose $m_s = 0.71 \rightarrow \theta = 2.0196$, then

$$R_2(s) = 0.9903(1+0.4s)(1+0.1s)(1+0.5s)$$

For $O_2^{'}(s)$:
- Choose $m_s = 0.71 \rightarrow \theta = 2.546$, then

$$R_2^{'}(s) = 0.7826(1+0.7042s)$$

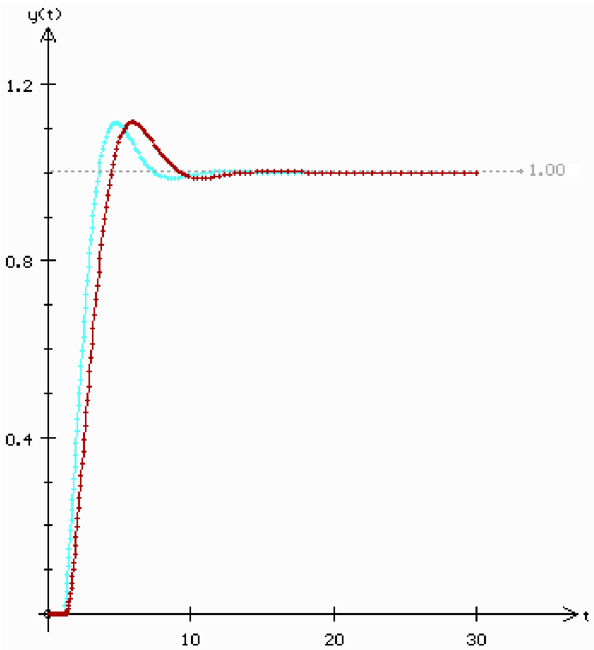

Fig. 14. Unit step responses of $O_2(s)$ closed-loop
$R_2(s)$*_Green,* $R_2^{'}(s)$*_Red*

$O_2(s)$ closed-loop responses of $R_2(s)$ and $R_2^{'}(s)$ controllers are shown in Fig. 14.

The responses are similar, which means $O_2(s)$ being modeled very exactly by $O_2^{'}(s)$.

Identify by IFOPDT

The integrating process with inverse response is able to identify directly by IFOPDT model. Indeed, if use (3) to identify $O_1(s)$ will get the model,

$$O_1^{*}(s) = \frac{0.4975}{s(1+0.4962s)} e^{-1.6864s}$$

The unit step responses and frequency characteristics of $O_1(s)$, $O_1^{'}(s)$ and $O_1^{*}(s)$ are shown in the Fig. 15&16.

The curves show that $O_1^{*}(s)$ function also is very efficient for $O_1(s)$ modeling.

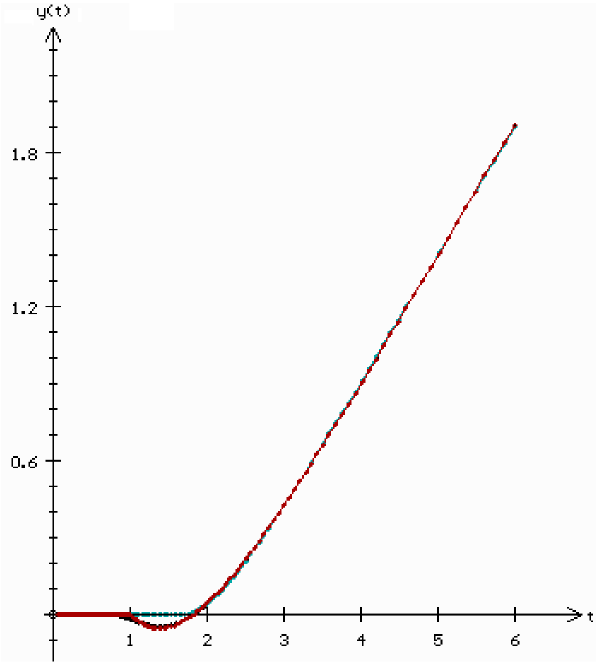

Fig. 15. Unit step responses of $O_1(s)$, $O_1^{'}(s)$ and $O_1^{*}(s)$
$O_1(s)$_Black, $O_1^{'}(s)$*_Red,* $O_1^{*}(s)$*_Green*

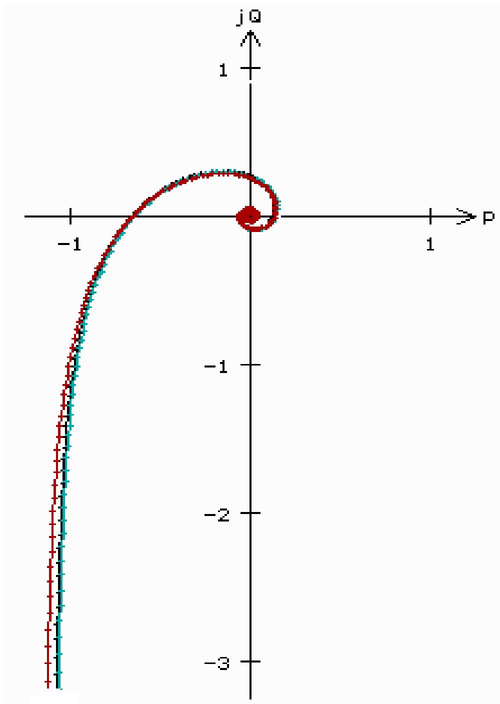

Fig. 16. Frequency characteristics $O_1(s)$, $O_1^{'}(s)$, $O_1^{*}(s)$
$O_1(s)$*_Black,* $O_1^{'}(s)$*_Red,* $O_1^{*}(s)$*_Green*

The PD robust-based controller (10) with $m_s = 0.71$ ($\theta = 2.838$) for $O_1^{*}(s)$ is,

$$R_2^{*}(s) = 0.7083(1+0.4962s)$$

$O_1(s)$ closed-loop responses of $R_1^{*}(s)$ and $R_2^{*}(s)$

controllers to unit step input are shown in Fig. 17.

The controller $R_2^*(s)$ even gives a bit better control quality than $R_1^*(s)$, this point demonstrates that the method models inverse response process $O_1(s)$ by $O_1^*(s)$ process without inverse response is also efficient.

Thus, the identification technique enables to exactly identify integrating processes with deadtime and inverse response directly by IFOPDT instead of IFOPDTZ model being more complex and more difficult to set PID controller.

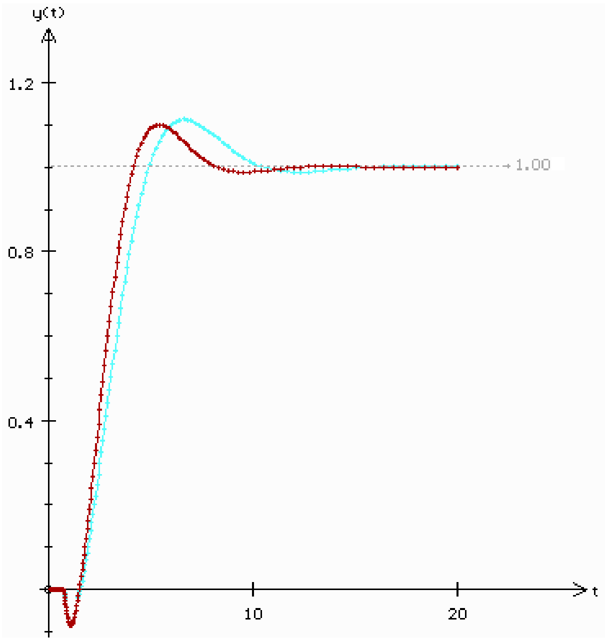

Fig. 17. Unit step responses of $O_1(s)$ closed-loops

$R_1^*(s)$*_Green,* $R_2^*(s)$*_Red*

6. CONCLUSION

The proposed identification technique proves the effectiveness via examples, its better result compared to that of Luyben's method thanks to the cleft-over algorithm. The proposal is able to be used to model the integrating processes with dead time and inverse response from measured data or to deduct complex given transfer function by the IFOPDTZ model. With identified models, the designers are able to apply any tuning methods at their convenience. Additionally, the technique permits identifying exactly the integrating processes with and without inverse response by IFOPDT model, which makes the controller setting period is much simpler with much more choices of approaches.

The presented tuning procedure allows the system robustness margin to be pre-set, by only this index it is easy to calculate the lag constant θ and gain controller for any given transfer function. Owing to IFOPDTZ or IFOPDT models from the proposed identification technique, it is convenient to synthesize the PID controller for integrating processes with dead time and inverse response. The suitable robustness index is 0.71 but, it is stated that the value enabled to choose in the range of [0.132; 2.318]. The permission of system robustness setting in a range is also flexible in the controller tuning phase according to practical integrating processes. The procedure tuning effectiveness is also demonstrated in the examples.

The proposed methods of identification and tuning are not only applicable for integrating processes with dead time and inverse response but also applicable to other types of processes.

ACKNOWLEDGMENT

This research is funded by the Hanoi University of Science and Technology (HUST) under project number T2018-PC-082.

REFERENCES

[1] William L. Luyben. 2003. Identification and Tuning of Integrating Processes with Deadtime and Inverse Response. *Ind. Eng. Chem. Res* 42: 3030-3035.

[2] Mayuresh V. Kothare, Bernard Mettler, Manfred Morari, Pascale Bendotti, Clément-Marc Falinower. 2000. Level Control in the Steam Generator of a Nuclear Power Plant. *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.* 8 (1), JANUARY 2000.

[3] Gu D, Ou L, Wang P, Zhang W (2006) Relay feedback autotuning method for integrating processes with inverse response and time delay. Ind Eng Chem Res 45:3119–3132.

[4] Neng-Sheng Pai, Shih-Chi Chang, Chi-Tsung Huang. 2010. Tuning PI/PID controllers for integrating processes with deadtime and inverse response by simple calculations. *Journal of Process Control* 20: 726–733.

[5] Jeng JC, Lin SW .2012. Robust proportional-integral-derivative controller design for stable/integrating processes with inverse response and time delay. Ind Eng Chem Res 51:2652–2665.

[6] Kaya I (2016) PI-PD controllers for controlling stable processes with inverse response and dead time. Electr Eng 98: 55–65.

[7] Ibrahim Kaya .2018. Controller design for integrating processes with inverse response and dead time based on standard forms. *Electrical Engineering* 100:2011-2022.

[8] A. Ali, S. Majhi. 2010. PID controller tuning for integrating processes, *ISA Trans.* 49: 70–78.

[9] Q.B. Jin, Q. Liu. 2014. Analytical IMC-PID design in terms of performance/robustness tradeoff for integrating processes: From 2-Dof to 1-Dof. *Journal of Process Control*, vol. 24, pp. 22–32.

[10] Anil Ch, Sree RP. 2015. Tuning of PID controllers for integrating processes using direct synthesis method. *ISA Trans.* 57: 211–219.

[11] Ajmeri M, Ali A. 2015. Simple tuning rules for integrating processes with large time delay. *Asian J Control* 17: 2033–2040.

[12] Manh N.V., Tri B.M. 1990. Method of "Cleft-Over-Step" by Perpendicular direction for solving the unconstrained non-linear optimization problem. *ACTA Mathem. Vietnam* 2: 73-83.

[13] Manh N.V., (1999) *Optimizing methods of uncertainty control systems.* Diss. Doc. Sc., MPEI, Moscow.

[14] Trung D.C., (2019) *Research tuning method of thermal process control loops for power plant wide-range operation.* Doctoral dissertation, Hanoi University of Science and Technology (HUST).