

# A Method Tuning Control System of Thermal Process in Startup Period

Do Cao Trung

*Department of Automation and Control of Thermal Process, Hanoi University of Science and Technology (HUST), Vietnam*

**Abstract.** The tuning control system of thermal process in startup period up today is very difficult problem due to difficulties are to model the process and to tune controller guaranteeing robustness of the system. In the paper the process is approximated second-order-plus-delay model defined by “cleft-over” algorithm [1]. And the parameters of controller defined based on required robustness of the system for startup case. Finally, the illustrative examples are presented.

## 1 Introduction

Let consider the typical control structure in thermal technology (Figure 1), where,  $R(s)$  and  $O(s)$  are transfer functions of process and controller.

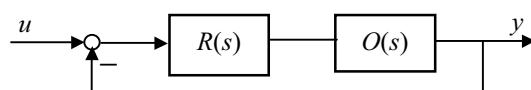


Figure 1. Typical control structure.

Currently, the tuning control system of the thermal process in start-up period is a difficult problem and unresolved up today.

Because of the existence, the startup and commissioning of thermal processes are often protracted, that makes implemented cost rises.

The obstacle is mostly the system identification in startup mode, i.e. in terms of change and lack of information. It is a difficult, complex problem that so far almost no research attention.

The article presents the problem of how to solve the tuning of control systems in start-up period consists of two phases: Phase 1 – system identification and Phase 2 – compute the parameters of controller.

## 2 System identification

### 2.1 Model definition

In thermal control engineering there are two main classes of control objects. The first includes dominant-lag processes with dead time and the other includes dominant-integral ones with dead time. The mentioned processes can be approximated by 2-nd-order lag plus delay model under the transfer function:

$$O_{Q2T}(s) = \frac{K}{(1+T_1s)(1+T_2s)} \cdot e^{-\tau s} \quad (1)$$

Where:  $K$  – gain factor;  $T_1, T_2$  – lag constants ( $T_1 \neq T_2$ );  $\tau$  - dead time;  $s$  – complex variable.

The model (1) quite matches the dominant-lag processes [6]. In the other hand, for dominant-integral processes, the model (1) gives also very good approximation. Indeed, we may transform (1) to:

$$O_{Q2T}(s) = \frac{K/T_1}{(1/T_1 + s)(1+T_2s)} \cdot e^{-\tau s}$$

It is easy to see that if  $1/T_1 \rightarrow 0$  when keep  $K_0 = K/T_1 \rightarrow a < \infty$ , then model becomes dominant-integral:

$$O_{Q2T}(s) = \frac{K_0}{s(1+T_2s)} \cdot e^{-\tau s} \quad (2)$$

The unit step response of (1) is

$$y(t) = h(t) = K \left[ 1 - \frac{T_1 e^{-\frac{t-\tau}{T_1}}}{T_1 - T_2} + \frac{T_2 e^{-\frac{t-\tau}{T_2}}}{T_1 - T_2} \right] \quad (3)$$

Where:  $t$  – time variable;  $T_1 \neq T_2$ .

In the time  $t = t_i$ , the response is

$$y(t_i, x) = K \left[ 1 - \frac{T_1 e^{-\frac{t_i-\tau}{T_1}}}{T_1 - T_2} + \frac{T_2 e^{-\frac{t_i-\tau}{T_2}}}{T_1 - T_2} \right] \quad (4)$$

and the measuring process value is  $y_i$ . The parameters of vector  $x = \{K, \tau, T_1, T_2\}$  have to be calculated to warranty that the second-order lag plus delay (1) presents as exactly as possible the practical thermal process. This requirement will be satisfied if the below problem of optimization is solved.

$$F(\mathbf{x}) = \sum_{i=1}^N [y(t_i, \mathbf{x}) - y_i]^2 \rightarrow \min_{\mathbf{x}} \quad (5)$$

Here: N is the number of measuring points.

$F(\mathbf{x})$  is non-linear function of K,  $\tau$ ,  $T_1$  and  $T_2$  variables. Problem (5) will be solved by using “cleft-over” algorithm [1].

## 2.2 Optimizing model by cleft-over algorithm

We propose use cleft-over algorithm to solve the problem (5) summarized as below:

### 2.2.1 Specifying objective function [1, 2]

Problem (5)-(6) needs to be determined with constraints of parameters based on each practical process. The constraints are:

$$0 \leq K; 0 \leq T_1, T_2; 0 \leq \tau \quad (6)$$

Problem (5)-(6) may be driven to an equivalent following unconstraint problem:

$$J(\mathbf{x}) = F(\mathbf{x}) + p^* \Psi(\mathbf{x}) \rightarrow \min_{\mathbf{x}} \quad (7)$$

Where  $\Psi(\mathbf{x})$  is penalty function determined as:

$$\Psi(\mathbf{x}) = [|\tau| - \tau]^2 + [|T_1| - T_1]^2 + [|T_2| - T_2]^2 + [|K| - K]^2$$

$p = [10 \div 10^6]$  - the penalty coefficient.

### 2.2.2 Cleft-over step algorithm [1, 2]

The algorithm is built base on two concepts: moving direction of searching orbit and cleft-over step. The iterative equation of the algorithm is

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_{k+1} \mathbf{s}^k$$

Where:  $\mathbf{x}^k$ ,  $\mathbf{x}^{k+1}$  - the begin and end of the k+1-step,  $\alpha_{k+1}$  is the step length,  $\mathbf{s}^k$  is moving direction.

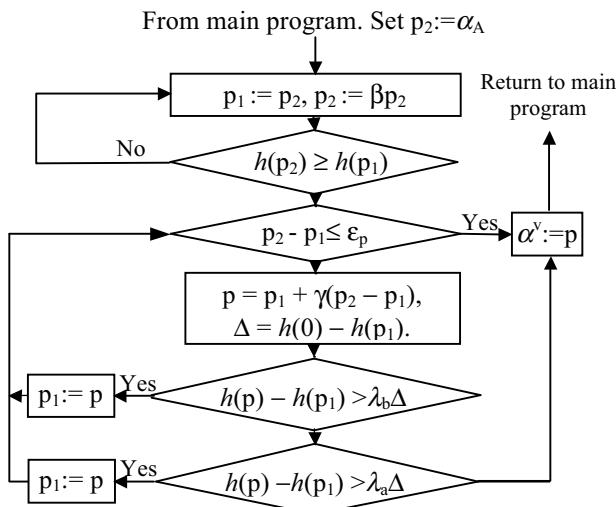


Figure 2. Flow chart of finding step length.

In the cleft-over-step algorithm the  $\alpha_{k+1}$  defined based on “cleft-over principle” [2] that provides the begin and end points of each iteration lay on different sides of minimum point of objective function in  $\mathbf{s}^k$  direction.

Length of each iterative step of the algorithm is defined by flow chart in Figure. 3.

For each iterative step, suppose:

$$h(\alpha) = J(\mathbf{x}^{k+1}) = J(\mathbf{x}^k + \alpha \mathbf{s}^k)$$

Each iterative step of the algorithm satisfies following condition:

$$\lambda_a[h(0) - h(\alpha^*)] < h(0) - h(\alpha^v) < \lambda_b[h(0) - h(\alpha^*)] \quad (8)$$

Where:  $\alpha^v$  – iterative cleft-over step;  $\alpha^*$  - minimum point of  $h(\alpha)$ ;  $\lambda_a, \lambda_b$  - coefficients:  $0 \leq \lambda_a \leq \lambda_b \leq 1$ .

The algorithm of finding cleft-over step is shown as below:

Initial values:  $\epsilon_p > 0$ ,  $\beta > 1$ ,  $0 < \gamma < 1$ ,  $\alpha_A > 0$ .

- Step 1: Set  $p_1 = 0$ ,  $p_2 = \alpha_A$  and increase the values repeatedly by the rule:  $p_1 := p_2$ ,  $p_2 := \beta p_2$  until  $h(p_2) \geq h(p_1)$ , get the length ( $p_1, p_2$ ) and go to step 2.

- Step 2: If the condition  $p_2 - p_1 \leq \epsilon_p$  is satisfied, means not possible to find cleft-over step with exact level  $\epsilon_p$ . Set  $\alpha^v := p$  and end, return to main program. If not, go to the step 3.

- Step 3: Calculate  $p := p_1 + \gamma(p_2 - p_1)$  and  $\Delta = [h(0) - h(p_1)]$ , go to step 4.

- Step 4: If  $h(p) - h(p_1) > \lambda_b \Delta$ , set  $p_2 := p$ ; if  $h(p) - h(p_1) < \lambda_a \Delta$ , then set  $p_1 := p$ ; repeat from step 2. The other case,  $\lambda_b \Delta > h(p) - h(p_1) > \lambda_a \Delta$  and  $\alpha^v = p$  is left-over step. Set  $\alpha_A := \alpha^v$  and return to main program.

The direction  $\mathbf{s}^k$  is determined to make the target function reduce after each iterative step, means  $J(\mathbf{x}^{k+1}) < J(\mathbf{x}^k)$ . The combination of cleft-over step adjusting rule and one move direction  $\mathbf{s}^k$  will create one cleft-over step algorithm. The algorithm is shown in Figure. 3.

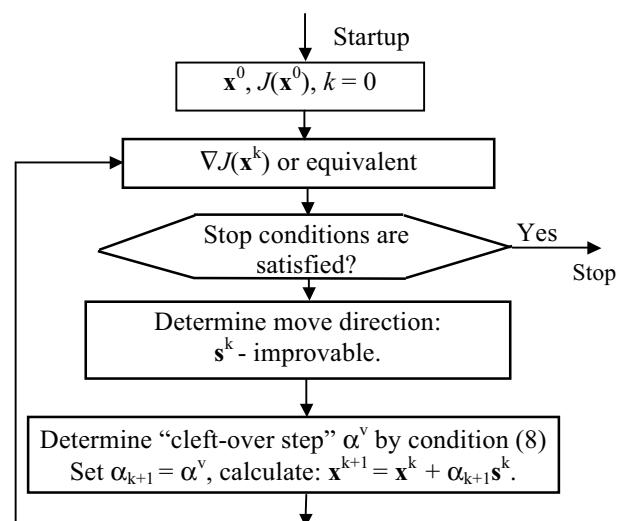


Figure 3. Cleft-over step algorithm.

The initial parameters are set as below:

$$\gamma = 0,382; \lambda_a = 0; \lambda_b = 0,5.$$

In the first step ( $k = 0$ ), choose start point  $\mathbf{x}^0$ , the first step value  $\alpha_A = 0,1$ .

Normally, the first move vector is chosen:  $\mathbf{s}^0 = -\nabla J(\mathbf{x}^0)$  (minus gradient value of target function in start point). The progress of optimization by using cleft-over step algorithm in step no. ( $k+1$ ) consists of two parts as below:

- Calculate gradient  $\nabla J(\mathbf{x}^k)$  and determine the direction  $\mathbf{s}^k$ .

- Move by direction  $\mathbf{s}^k$  until reach the cleft uptrend, means the condition (8) is satisfied. End the iterative step no. ( $k+1$ ), get the new point:  $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_{k+1} \mathbf{s}^k$  with  $\alpha_{k+1} = \alpha^y$  is cleft-over step.

The repeat is continuous until fixed conditions are satisfied. One of the most efficient moving directions is Affine [1].

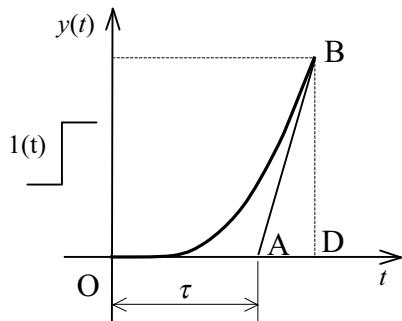
### 3 Tuning controller

When the model of process is achieved, the calculation of controller parameters is able to fulfill by using some popular rules as Zigler-Nichol 1, SIMC and Robust-based.

#### 3.1 Zigler-Nichol 1 method [3]

This method uses approximate model of process. With the unit step response is shown in Figure. 4, the chosen model is time delay integral:

$$O_{Z-NI}(s) = \frac{K}{s} \cdot e^{-\tau s}$$



**Figure 4.** Process unit step response.

Here:  $K = BD/AD$  (Figure 4)

The PID controller equation is:

$$R(s) = k_p \left( 1 + \frac{1}{T_I s} + T_D s \right) e^{-\tau s}$$

The parameters are set:

$$k_p = \frac{1,2}{K\tau}, T_I = 2\tau, T_D = \frac{\tau}{2}$$

#### 3.2 SIMC method [4]

The original Internal Model Control based on model of process. The best improve of IMC is SIMC by Skogestad [4]. The PID controller is

$$R(s) = k_c \left( \frac{1 + T_I s}{T_I s} \right) (1 + T_D s)$$

Its parameters for model (1) are set as below:

$$k_c = \frac{1}{K} \cdot \frac{T_I}{\tau_c + \tau}, T_I = \min\{T_I, 4(\tau_c + \tau)\}, T_D = T_2$$

Normally, the tuning parameter  $\tau_c$  is chosen equal to time delay value  $\tau$  of processes.

#### 3.3 Robust-based method [5]

The method is based on model of process, with the general equation is

$$O(s) = \frac{B(s)}{A(s)} \cdot e^{-\tau s}$$

The robust-based controller is defined:

$$R(s) = \frac{1}{\theta s} \cdot \frac{A(s)}{B(s)}$$

The coefficient  $\theta$  is lag time constant which is defined by the stable requirement of system.  $\theta$  is chosen in the condition that the oscillation index of dominant pole of the control system varies in the range from 0,461 to 2.

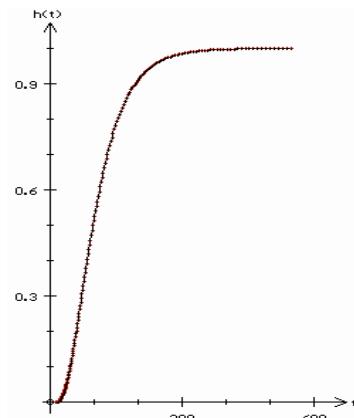
### 4 Examples

#### 4.1 Lag process

A superheated steam process of one thermal power plant has the transfer function:

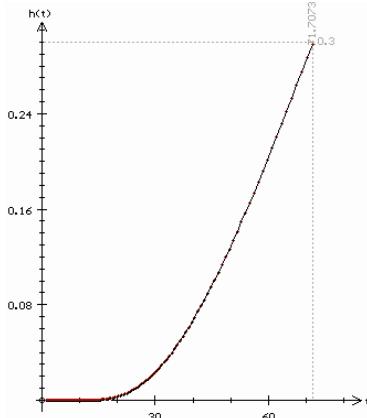
$$O_{sh}(s) = \frac{1}{(1 + 54s)(1 + 22,8s)(1 + 22,8s)} \cdot e^{-11,4s}$$

The unit step response of it is shown in Figure. 5.



**Figure 5.** Unit step response of  $O_{sh}$ .

Collect the data from 0 to 30% of output at the time  $t_{30\%}$ .



**Figure 6.** Initial data from characteristic of  $O_{sh}$ .

Identify the obtained data by model (1) with the limitation:

$$0 \leq K; 0 \leq T_1, T_2; 0 \leq \tau \leq t_{30\%} \leq 74 \quad (9)$$

Get 15 measuring points at the time  $t_i = 0, 5, 10, \dots, 70$ , values  $y_i$  is shown respectively in the Table 1.

**Table 1.** Fifteen measuring points.

$t_i$	0	5	10	15	20	...	65	70
$y_i$	0	0	0	0,001	0,003	...	0,243	0,286

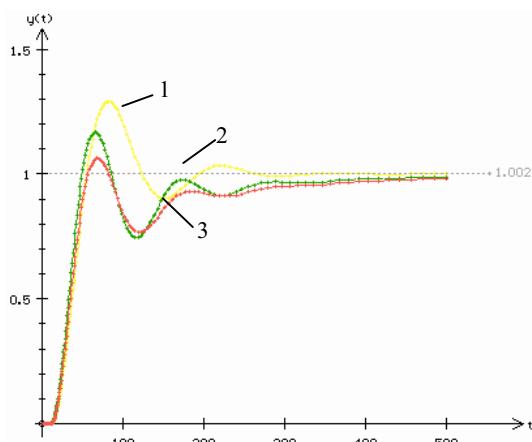
Input the values in (5) will have function  $F(x)$ , the penalty function  $\Psi(x)$  is determined from (8), coefficient  $p$  is set by  $10^3$ . Finally, function  $J(x)$  in (7) is formed with variable  $x = \{K, \tau, T_1, T_2\}$ .

Using cleft-over step algorithm to identify the system, after 37 iterative steps achieves the process model in startup period is

$$O'_{sh}(s) = \frac{31,813}{(1+2580s)(1+41,218s)} \cdot e^{-16,565s}$$

By this model, we use the Zigler-Nichol 1 [3], SIMC [4] and robust-based [5] tuning methods to compute controllers.

After that input three obtained controllers in closed control loop of the original  $O_{sh}(s)$ . The closed-loop unit step responses of these different controllers are shown in Figure. 7.



**Figure 7.** Closed-loop response with different controllers 1\_Zigler-Nichol; 2\_SIMC; 3\_Robust-based.

The results express:

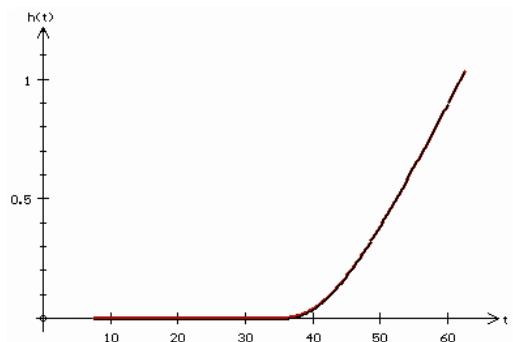
- All three controllers give good tuning quality.
- Three qualitative factors are used to compare are Decay, Overshoot and Steady time. The results of each method are expressed:
  - + Decay: Robust-based is lower than SIMC, SIMC is lower than Zigler-Nichol 1.
  - + Overshoot: Robust-based is lower than SIMC, SIMC is lower than Zigler-Nichol 1.
  - + Steady time: Robust-based and SIMC are the same, both of them are a bit shorter than Zigler-Nichol 1.

## 4.2 Integral processes

A level drum process of one thermal power plant has the transfer function [7]:

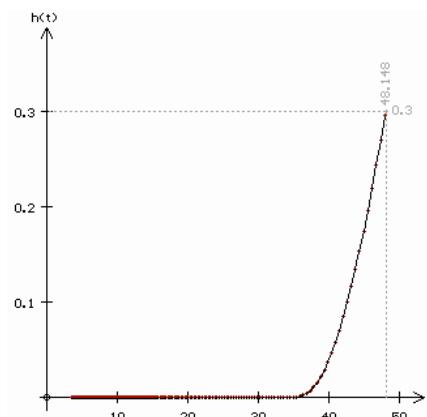
$$O_{dr}(s) = \frac{0,055}{s(1+4,608s)^2} \cdot e^{-34,671s}$$

The unit step response of it is shown in Figure. 8.



**Figure 8.** Unit step response of  $O_{dr}$ .

Collect the data from 0 to 30% of output at the time  $t_{30\%}$ .



**Figure 9.** Initial data from characteristic of  $O_{dr}$ .

Identify the obtained data by model (1) with the limit conditions:

$$0 \leq K; 0 \leq T_1, T_2; 0 \leq \tau \leq t_{30\%} \leq 49 \quad (10)$$

Get 17 measuring points at the time  $t_i = 0, 3, 6, \dots, 48$ , values  $y_i$  is shown respectively in the Table 2.

**Table 2.** Seventeen measuring points.

$t_i$	0	3	...	36	39	42	45	48
$y_i$	0	0	...	0	0,056	0,163	0,228	0,293

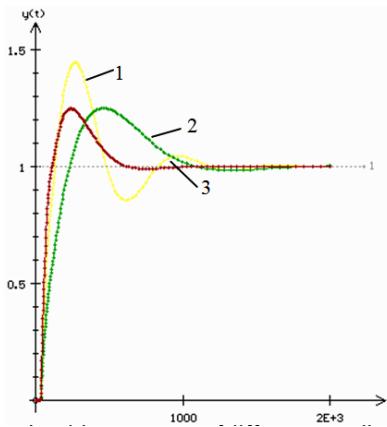
Replace the values into (5) will have function  $F(x)$ , the conditional function  $\Psi(x)$  is determined from (9), coefficient  $p$  is set by  $10^3$ . Finally, function  $J(x)$  in (7) is formed by vector  $x = \{K, \tau, T_1, T_2\}$ .

Using cleft-over step algorithm to identify the system, after 55 iterative steps achieves the process model in startup period is

$$O'_{dr}(s) = \frac{41,21}{(1+329,616s)(1+26,852s)} \cdot e^{-35,763s}$$

By this model, we use the Zigler-Nichol 2 [3], SIMC [4] and robust-based [5] tuning rules to compute controllers.

After that, input three computed controllers into closed control loop of the root  $O_{bh}(s)$ . The closed-loop unit step responses of these different controllers are shown in Figure 10.



**Figure 10.** Closed-loop response of different controllers  
1\_Zigler-Nichol 1; 2\_SIMC; 3\_Robust-based.

The results express:

- All three controllers give good tuning quality.
- Three qualitative factors are used to compare are Decay, Overshoot and Steady time. The results of each method are expressed:
  - + Overshoot: Robust-based is the same SIMC and much lower than Zigler-Nichol 1.
  - + Decay: Robust-based is lower than SIMC, SIMC is much lower than Zigler-Nichol 1.
  - + Steady time: Robust-based is much shorter than SIMC, SIMC is a bit shorter than Zigler-Nichol 1.

## 5 Discussion

- Thermal process identification algorithm is efficient.
- Second-order lag plus delay model is suitable for modeling thermal processes including dominant integral and lag classes.
- The characteristic of Second-order lag plus delay model is expressed via the relationship between  $T_1$  and  $T_2$ . If they are so different, the model presents integral manner, otherwise it behaves as a lag model.

- The robust-based tuning rule gives general control quality better than SIMC and Zigler-Nichol 2 do in the two given examples.

- The private result of example is significant base in the choice of tuning rule for thermal process control systems.
- The identification of thermal process by Second-order lag plus delay model is very useful for tuning activity owing to the obtained controller in PID structure.

## 6 Recommendation

- The identification algorithm will give result faster and more exactly if the start values are sensibly chosen.
- The start vector should be set to satisfy:  $\tau \approx T_2 \approx 10\% * T_1$  and  $\tau + T_1 + T_2 \approx t_{30\%}$ . The identification result will more exactly present practical process.

## 7 Conclusions

- The article proposes a tuning algorithm for control system of thermal process in startup period.
- The proposed identification method provides receive rather precise model of process.
- The method tuning controller for startup period should be robust-based method that is most flexible and able give desired robustness of the system.
- The proposed methodology of implementation is clear, comprehensible and general. It is able to effectively apply to tune thermal process control systems in startup period.

## References

1. Nguyen Van Manh, MPEI, Moscow, *Diss. Doc. Sc.: Optimizing methods of uncertainty control systems* (1999)
2. Мань Н.В. Применение “Оврагоперешагового” метода нелинейной оптимизации для идентификации передаточной функции объектов управления //Теплоэнергетика. № 6. С. 71-77, (1995)
3. Zigler, J.G.; N.B. Nichols, Trans. Am. Soc. Mech. Eng., *Optimum settings for automatic controllers*, Vol. 64, pp. 759-768 (1942)
4. Sigurd Skogestad, Chriss Grimholt, PID Control in the Third Millennium, Advances in Industrial Control, *The SIMC Method for Smooth PID Controller Tuning*, Chapter 5, pp. 147-175 (2012)
5. Nguyen Van Manh, The fifth Vietnamese national conference of Automation, Science report, *Robust synthesizing method of uncertainty control system*, Vol. 5, pp. 155-161 (2002)
6. Nguyen Van Manh, Thermal Energy Review, A modeling method of thermal control processes by using second-order lag plus delay model, Vol. 125, pp. 15-19 (2015)
7. Do Cao Trung, Hanoi University of Science and Technology, *MSc.: A method synthesizing optimal-robust PID for thermal processes* (2011)