SICE Annual Conference 2008
August 20-22, 2008, The University Electro-Communications, Japan

PID Optimal Tuning Method by Particle Swarm Optimization

Akihiro Oi [1], Chikashi Nakazawa[1], Tetsuro Matsui[1],
Hiroe Fujiwara[2], Kouji Matsumoto[2], Hideyuki Nishida[2]

[1] Fuji Electric Advanced Technology, Tokyo, Japan
(Tel : +81-42-586-1116; E-mail: oi-akihiro@fujielectric.co.jp)
[2]Fuji electric Systems, Tokyo, Japan

Abstract: This paper proposes a new PID parameter tuning method by Particle Swarm Optimization (PSO) without tuning operator's know-how. The method searches the PID parameter that realizes the expected step response of the plant. The response is defined by the overshoot ratio, the rising time, the settling time and so on. The method is implemented into the PID tuning tool on a personal computer. The plant model represented by the transfer function is obtained by system identification on the PID tuning tool. The PID parameter is computed by PSO-based PID tuning method according to the obtained model. The numerical results show the effectiveness of the proposed tuning method and the developed tool.

Keywords: PID Control, Particle Swarm Optimization, System Identification.

1. INTRODUCTION

PID control is a fundamental control technology and it makes up 90% of automatic controllers on process control fields [1]. It is also necessary for the total energy saving system or the model predictive control to operate each single loop control system appropriately, and thus the PID control is absolutely essential.

For an ideal control performance by the PID controller, an appropriate PID parameter tuning is necessary [1]. In fact, PID parameter tuning depends on operator's know-how, therefore a PID parameter has not been frequently optimal from the viewpoint of qualities.

The CHR (Chien-Hrones-Reswick) tuning rule has been applied to the PID tuning frequently [2]. This method is for the basic PID controller. Therefore, it is not optimal for the practical controller such as PI-D or I-PD controllers of the industries. The tuning method based on PSO (Particle Swarm Optimization) [3] has been proposed to obtain the optimal PID parameter [4]. This method has not considered the constraints of the overshoot ratio, the rising time, the settling time and so on. This paper presents the new method that obtains the PID parameter realizing the expected step response with the constraints.

The proposed method is implemented into the PID tuning tool on a personal computer. The developed tool provides the PID parameter that realizes the expected step response of the plant. The numerical results show the effectiveness of the proposed PSO-based PID tuning method.

2. SUMMARY OF THE PID TUNING TOOL

The proposed PID tuning method needs the plant model of the controlled object, thus it needs system identification to obtain the model. Therefore, the procedure of system identification and the PID tuning is implemented as PID tuning tool on a personal computer. On the tool, the plant model represented by the transfer function is obtained by system identification. And based on the model, PID parameter is obtained by PSO-based PID tuning method. This PID tuning tool has following features.

・System identification by start-up or step response data in operation (under the identifiable condition).

・The PID parameter that realizes desirable process value (PV) response is obtained by PSO.

・The PID parameter tuning that has depended on a operator's know-how is automated

3. SYSTEM IDENTIFICATION

System identification is conducted with data of PV and manipulated value (MV). The gain, the time constant, and the delay time of the transfer function that defined by the user beforehand are obtained. The construction types of transfer function are prepared, for example, Table 1. Where G is the process gain, T, T_1, T_2 are the time constant, L is time delay on the transfer function on Table 1. The construction types are able to be customized with user need.

System identification is based on the optimization that minimizes the objective function. The objective function J is defined as integral square error as follows.

$$J = \int_0^T (\hat{P}V(\rho) - PV)^2 \, dt \tag{1}$$

where J is the object function, ρ is the parameter of the transfer function (the objective valuables), T is the

Table 1. A configuration diagram of FePIDtune.

Type of transfer function	Transfer function
First order with time lag	$\frac{G}{1+Ts}e^{-Ls}$
Second order with time lag	$\frac{G}{(1+T_1 s)(1+T_2 s)}e^{-Ls}$
First order with integral and time lag	$\frac{G}{s(1+Ts)}e^{-Ls}$

evaluate period, ^ is the estimate value.

To obtain the transfer function minimize Eq. (1), Newton-Raphson method is applied as follows.

$$\hat{\rho}^{(i+1)} = \hat{\rho}^{(i)} - \mu[H(\hat{\rho}^{(i)})]^{-1}\nabla J(\hat{\rho}^{(i)}) \tag{2}$$

where i is the iteration number, μ is the step length, ∇J is the gradient of J, H is the Hessian of J.

From Eq. (1-2), the parameter of the transfer function is obtained.

4. CONVENTIONAL PID TUNING

The PID tuning tool has two selectable features on the PID tuning. One is the conventional PID tuning rules from the formulation on the process model, another is the PID optimal tuning based on PSO. The results of the PID tuning parameter and simulation data are displayed on the PC screen, and the operator is able to check the results.

On the PID tuning tool, the conventional rules of PID tuning are available. The PID parameter is obtained by these rules on the plant model computed by system identification. Available tuning rules are as follows [2].

・Zigler-Nichols Rule
・CHR Rule
・Cohen-Coon Rule

5. PID OPTIMAL TUNING METHOD BY PSO

The conventional tuning rules need trial and error on the experiment or the simulation in order to adjust PV to expected one. The procedure has depended on the user know-how and has had issue of the tuning cost and period. For the resolution of these problems, the PID optimal tuning method based on PSO is developed. The proposed method computes the PID parameter that realizes expected PV without the user know-how. The concept of the proposal method is shown in Fig. 1.

The response is defined by overshoot ratio, rising time, settling time and so on. The method minimizes the objective function composed by the objective valuables that is the PID parameter.

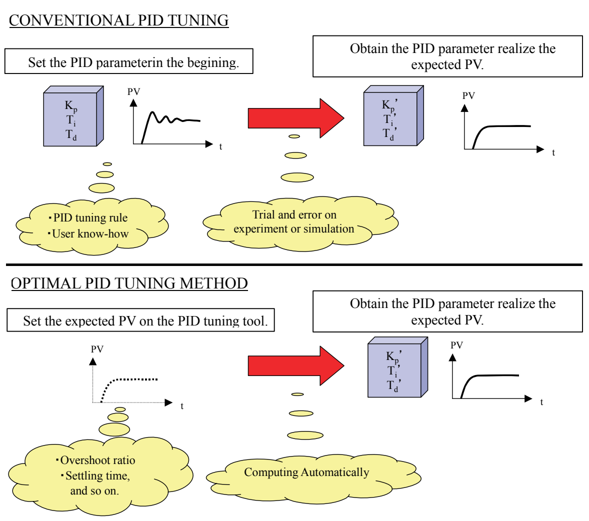


Fig. 1. Concept of the PID optimal tuning

The objective function f is defined as integral square error as follows.

$$F = \int e(t)^2 dt \tag{3}$$

where $e(t)$ is an error of set point value (SV) and PV.

The constraints are as follows.

$$\begin{aligned}
&T_r^L \le T_r \le T_r^U,\ T_d^L \le T_d \le T_d^U \\
&p_0^L \le p_0 \le p_0^U,\ T_p^L \le T_p \le T_p^U \\
&T_s^L \le T_s \le T_s^U,\ E_s^L \le E_s \le E_s^U
\end{aligned} \tag{4}$$

where T_r is a rising time, T_d is a delay time, p_o is an overshoot ratio, T_p is a peak time, T_s is a settling time, E_s is a steady state error, L is lower bound, U is upper bound. The objective valuables (the PID parameter) that minimize the objective function under the constraints are computed by PSO.

6. PARTICLE SWARM OPTIMIZATION (PSO)

PSO is one of the evolutionary computation methods to solve optimization problems. The method can be applied to non-linear optimization problem that includes constraints without the graduate of the objective function. The procedure of PSO is to iterate the following equation.

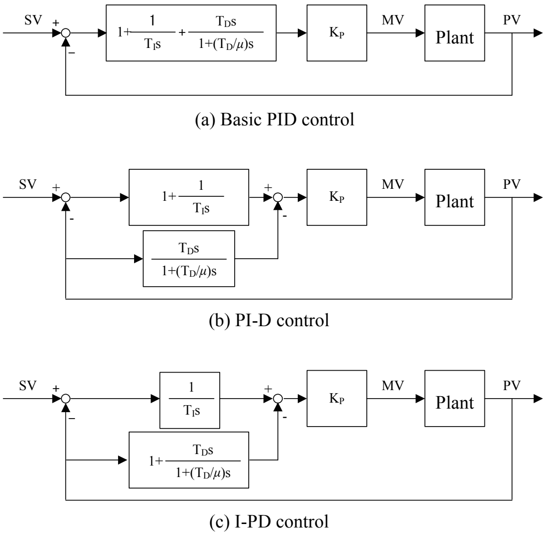
$$\begin{aligned}
v_{ij}^{(k+1)} &= w \times v_{ij}^{(k)} \\
&\quad + c_1 \times rand() \times (pbest_{ij} - x_{ij}^{(k)}) \\
&\quad + c_2 \times rand() \times (gbest_j - x_{ij}^{(k)})
\end{aligned} \tag{5}$$

$$x_{ij}^{(k+1)} = x_{ij}^{(k)} + v_{ij}^{(k+1)} \tag{6}$$

where i is a particle number, j is the PID parameter specie number, k is a iteration number, x is the PID parameter, v is a moving vector, *pbest* is a personal best of particle i, *gbest* is a global best of all particles, w, $c1$ and $c2$ are weight parameters, *rand*( ) is a uniform random number from 0 to 1. The pbest and the gbest are objective function value of Eq. (3), and update each iteration process. After the iteration, the objective valuable that is the PID parameter are obtained.

7. PID CONTROLLER

Various tuning rules have been proposed such as Ziegler Nichols rule or CHR rule and so on. However, these methods assume the basic PID control not such as I-PD control and IP-D control as shown in Fig.2, where K_p is the proportional gain, T_I is the integral time, T_d is the derivative time, μ is the derivative constant. Also on non-linear characteristics of the controller are not



(a) Basic PID control

(b) PI-D control

(c) I-PD control

Fig. 2. Block Diagram of PID controls.

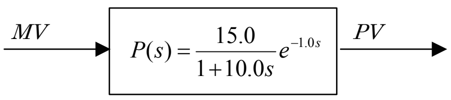


Fig.3. A block diagram for the numerical example (Case 1).

regarded to conventional rules, for example, MV limiter or integral cut action. On most controllers in the control field, the I-PD or the IP-D control is applied including MV limiter, integral cut action, or any functions including non-liner characteristics. The PID tuning tool is able to reflect these features of the real controller by using the PID simulator of the real controller. The PID tuning method based on PSO uses the simulator to obtain the evaluate value of J, Eq (3).

8. NUMERICAL EXAMPLES

For the purpose of showing the effectiveness of the proposed method and the developed tool, numerical examples are studied. For the numerical examples, the developed PID tuning tool is applied to the PID tuning to the closed loop system that is composed by the PID controller and the process models. The process models are two cases defined as the first order system and the second order system that include the time lag.

8.1 First order system (Case 1)

The plant model $P(s)$ by the transfer function is shown in Fig.3. The identification data is shown in Fig.4, and the identification result is shown in Table 2.

Based on the plant model, the optimal PID tuning method is applied. The expected PV responses are shown in Table 3.

As a result, the PID parameter of the traditional method (e.g. CHR rule) and the proposed method are shown in Table 2, PV data are shown in Fig.5, evaluation of PV data are shown in Table 5. According to the results, it is observed that the proposed method is able to obtain the desirable PID parameter under constraints against CHR tuning.

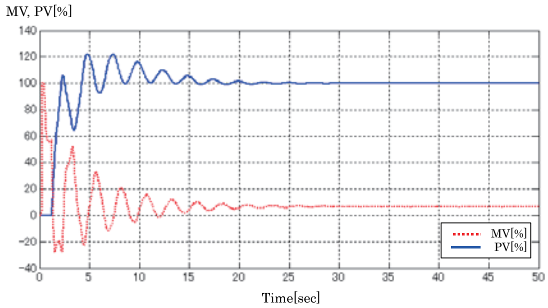


Fig.4. Data for the identification (Case 1).

Table 2. Result of the identification (Case 1).

Parameter of model	True value	Result
Process gain	15.0	14.9
Time constant [sec]	10.0	9.9
Time Lag [sec]	1.0	1.0

Table 3. Constraints for the optimal PID tuning (Case 1).

Evaluation of PV data	Constraint
T_r: Rising time[sec]	$1.0 \leqq T_r \leqq 5.0$
T_s: Settling time[sec]	$7.0 \leqq T_s \leqq 15.0$
p_0: Overshoot ratio[%]	$0.0 \leqq p_0 \leqq 2.0$
E_s: Steady state error[%]	$0.0 \leqq E_s \leqq 0.1$

Table 4. Results of the CHR Tuning and the PID Optimal Tuning (Case 1).

PID parameter	CHR tuning	Proposed method
Proportional gain	0.63	0.49
Integral time [sec]	13.50	14.02
Derivative time [sec]	0.47	0.48

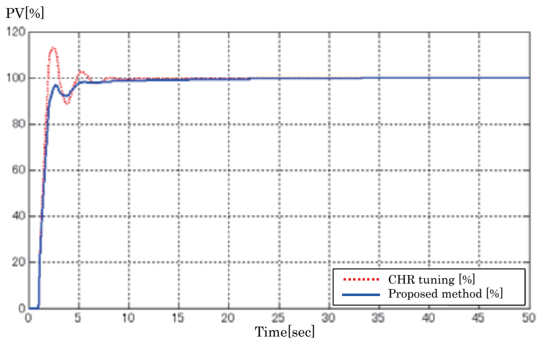


Fig.5. PV data of the CHR tuning and the PID optimal tuning (Case 1).

Table 5. Results of the CHR Tuning and the PID Optimal Tuning (Case 1).

Evaluation of PV data	CHR tuning	Optimal tuning
T_r: Rising time[sec]	0.8	1.1
T_s: Settling time[sec]	7.1	7.3
p_0: Overshoot ratio[%]	13.3	0.0
E_s: Steady state error[%]	0.0	0.0

8.2 Second order system (Case 2)

The plant model P(s) by the transfer function is shown in Fig.6. The identification data is shown in Fig.7, and the identification result is shown in Table 6.

Based on the plant model, the optimal PID tuning method is applied. The expected PV responses are shown in Table 7.

As a result, the PID parameter of the traditional method and the proposed method are shown in Table 8, PV data are shown in Fig. 8, evaluation of PV data are shown in Table 9. In this case, in order to apply CHR rule to this problem, two time constants are considered as one time constant. According to the results, it is observed that the proposed method is able to obtain the desirable PID parameter under constraints.

9. CONCLUSIONS

This paper presents the new method that obtains the PID parameter realizing the expected step response with the constraints. The proposed method is implemented into the PID tuning tool on a personal computer. The developed tool provides the PID parameter that realizes the expected step response of the plant. The numerical results show the effectiveness of the proposed PSO-based PID tuning method.

Future works are to develop the method to obtain the optimal PID parameter to disturbance.

REFERENCES

[1] Carl Knospe: “PID Control”, IEEE Control Magazine, pp.30-31, 2006.

[2] Karl J. Astrom and Tore Hagglund: “PID Controllers: Theory, Design, and Tuning ”, International Society for Measurement and Con, 1995.

[3] J.Kennedy, R.C.Eberhart: “Swarm Intelligence”, Morgan Kaufmann Publishers, 2001.

[4] Z.L.Gaing, “A Particle Swarm Optimization Approach For Optimum Design of PID Controller in AVR system”, IEEE Trans. on Energy Conversion, Vol.19, No.2, pp.384-391, 2004.

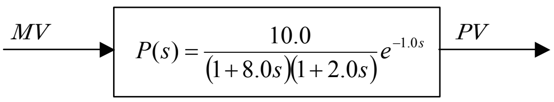


Fig.6. A block diagram for the numerical example (Case 2).

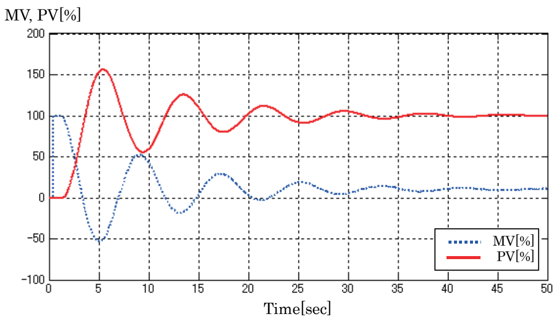


Fig.7. Data for the identification (Case 2).

Table 6. Results of the identification (Case 2).

Parameter of model	True value	Result
Process gain	10.0	9.9
Time constant 1 [sec]	8.0	8.0
Time constant 2 [sec]	2.0	2.0
Time Lag [sec]	1.0	0.9

Table 7. Constraints for the optimal PID tuning (Case 2).

Evaluation of PV data	Constraint
T_r: Rising time[sec]	$1.0 \leqq T_r \leqq 5.0$
T_s: Settling time[sec]	$20.0 \leqq T_s \leqq 30.0$
p_0: Overshoot ratio[%]	$0.0 \leqq p_0 \leqq 5.0$
E_s: Steady state error[%]	$0.0 \leqq E_s \leqq 0.5$

Table 8. Results of the CHR Tuning and the PID Optimal Tuning (Case 2).

PID parameter	CHR tuning	Proposed method
Proportional gain	0.60	0.44
Integral time [sec]	10.00	17.36
Derivative time [sec]	0.50	1.00

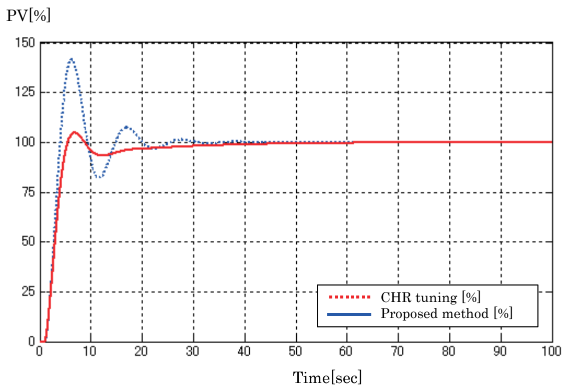


Fig.8. PV data of the CHR tuning and the PID optimal tuning (Case 2).

Table 9. Results of the CHR Tuning and the PID Optimal Tuning (Case 2).

Evaluation of PV data	CHR tuning	Optimal tuning
T_r: Rising time[sec]	2.1	3.3
T_s: Settling time[sec]	24.0	29.9
p_0: Overshoot ratio[%]	41.7	4.9
E_s: Steady state error[%]	0.0	0.0