Abstract

Using fuzzy method to adjust adaptive controller is fuzzy adaptive controlling, so-called adaptive controlling is that using modern control theory to recognize object's characteristic parameters online and change their control strategies timely so that control system's quality index could keep in the best range. Fractional order controller $P^\alpha D^\beta$ became a hot pot in recently years, when compared it with the common PID controller’s system we found that the former has shorter rise time, no overshoot or oscillations, and has stronger robustness. Especially, when the system has nonlinear and time-varying links, the adjusting effects will be excellent. Then the simulation of the fractional order controller $P^\alpha D^\beta$ combination with the adaptive controlling is carried out based on Matlab in this text.

Keywords: fractional order PID controller, fuzzy adaptive controller, robustness

1. Introduction

In industrial production process, many charged objects often face the effects of load or disturbance, so its characteristic parameters or structure are frequently changed. It is impractically, if the controller has been using fixed strategy to control or manually modifying strategy to meet the change of the object's characteristic parameters, thus adaptive controller was used by more and more people in this situation. Adaptive controlling recognize object's characteristics parameters online by using modern control theory, change control strategy on time, keep the control system's quality index in the best range. But the fuzzy control is established on the basis of artificial experience. In industrial production, some complex production process using common control strategy can't achieve satisfactory control effect, for a skilled operator often take ably appropriate measures to control a complicated process by using rich practical experience. If these experiences were summarized and described by words we will get a qualitative, not precise control rule, when use the fuzzy mathematical method to quantify the rule just formed today's the fuzzy control theory. Now we combine the fuzzy control theory that has the ability to solve complex problems, the
adaptive controller that can automatically adjust the control strategy and the fractional controller that has very good adjust effect to nonlinear time-varying links, we will get the best control effect.

2. Fractional order \(Pl^\lambda D^\mu\) controllers (FOPID)

Fractional order control system is described by fractional order differential equations. Fractional calculus allows the derivatives and integrals to be arbitrary order. The FOPID controller is the expansion result of the conventional PID controller based on fractional calculus. FOPID controllers design parameters have five and improve the design flexibility.

The general calculus operator (including fractional order and integral order) is defined as:

\[
\alpha D^\alpha_t f(t) = \begin{cases} 
\frac{d^\alpha}{dt^\alpha}, & \text{Re}(\alpha) > 0 \\
1, & \text{Re}(\alpha) = 0 \\
\int (d\tau)^{-\alpha}, & \text{Re}(\alpha) < 0
\end{cases}
\] (1)

There are several definitions of fractional derivatives as fellows:

(1) Grunwald-Letnikov definition

\[
\alpha D^\alpha_t f(t) = \lim_{h \to 0} \left[ \frac{1}{h^\alpha} \sum_{j=0}^{\lfloor (t-j)/h \rfloor} (-1)^j \binom{\alpha}{j} f(t - jh) \right] (2)
\]

Where \([\bullet]\) is a truncation; \(\binom{\alpha}{j}\) is binomial coefficients

(2) Riemann-Liouville definition

\[
\alpha D^\alpha_t f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_0^t \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau
\] (3)

Where \(n-1<\alpha<n\); \(\Gamma(\bullet)\) is the Euler Gamma function

(3) Caputo definition

\[
\alpha D^\alpha_t f(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{f^n(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, \quad (n-1<\alpha<n)
\] (4)

FOPID is developed on the basis of fractional order calculus, it is the general form of the integer traditional PID, besides the three parameters of integral PID controller, FOPID has two parameters that is integral order \(\lambda\) and differential order \(\mu\) and the value is any real. After introducing the two parameters, the system transfer function is described by:

\[
u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d D^\mu e(t)
\] (5)

The continuous transfer function of FOPID is obtained through Laplace transform, which is given by:

\[
G(s) = k_p + \frac{k_i}{s} + k_d s^\mu
\] (6)

After the discrete time, the (5) is described by:

\[
u(k) = k_p e(k) + h^\lambda \sum_{j=0}^k q_j e(k-j) + h^\mu \sum_{j=0}^k d_j e(k-j)
\] (7)

Where \(e(k) = rin - yout(k)\), \(q_j = (1 - \frac{\lambda j}{j})d_{j-1}\), \(d_j = (1 - \frac{\mu j}{j})d_{j-1}\), \(q_0 = d_0 = 1\)
From (5) we could obtained the conclusions that when $\lambda = 1$ and $\mu = 1$ the controller is traditional integral PID; when $\lambda = 1$ and $\mu = 0$ the controller is PI controller; when $\lambda = 0$ and $\mu = 1$ the controller is PD controller; when $\lambda = 0$ and $\mu = 0$ the controller is a proportion amplifier.

3. The tuning principle of adaptive fuzzy FOPID controller parameters

The adaptive fuzzy FOPID controller has two inputs which are error and error change can meet the self-setting requirements of FOPID controller parameters in different time. The adaptive fuzzy controller is that the controller parameter was modified online by using the fuzzy control rules, the chart is as followed:

![Figure 1: The adaptive fuzzy controller structure](image)

The tuning of FOPID controller $k_p$, $k_i$, $k_d$ is roughly same with integral PID controller, the tuning principle is to find out the fuzzy relation between $k_p$, $k_i$, $k_d$ with error and error change then establish fuzzy rules. Through continually testing error and error change in the entire process to adjust the three parameters online so that make charged object could has good dynamic and static performance. The modifier formula of $k_p$, $k_i$, $k_d$ is given by:

$$
\begin{align*}
\dot{k}_p &= k_p + \{e_i, ec_i\}_p \\
\dot{k}_i &= k_i + \{e_i, ec_i\}_i \\
\dot{k}_d &= k_d + \{e_i, ec_i\}_d
\end{align*}
$$

Where $k_p$, $k_i$, $k_d$ is the initial value of the three parameters, for another two parameters $\lambda$ and $\mu$ is given an initial value and programming in matlab by using (7) then continuously sample and iterate finally find the optimal value of five parameters.

The following is control objective:

$$G(s) = \frac{400}{s^2 + 50s}$$

The sampling period is 0.001s. The input $r$ is step signal. Firstly establish the fuzzy ruler form of $k_p$, $k_i$, $k_d$ respectively.

| $e$   | NB | NM | NS | ZO | PS | PM | PB |
|-------|----|----|----|----|----|----|----|
| NB    | PB | PB | PM | PM | PS | ZO | ZO |
| NM    | PB | PB | PM | PM | PS | ZO | NS |
| NS    | PM | PM | PM | PS | ZO | NS | NS |
| ZO    | PM | PM | PS | ZO | NS | NM | NM |
| PS    | PS | PS | ZO | NS | NS | NM | NM |
| PM    | PS | ZO | NS | NM | NM | NM | NB |
| PB    | ZO | ZO | NM | NM | NM | NB | NB |
| e      | NB | NM | NS | ZO | PS | PM | PB |
|--------|----|----|----|----|----|----|----|
| NB     | NB | NB | NM | NM | NS | ZO | ZO |
| NM     | NB | NB | NM | NS | NS | ZO | ZO |
| NS     | NB | NM | NS | NS | ZO | PS | PS |
| ZO     | NM | NM | NS | NS | ZO | PS | PM |
| PS     | NM | NS | ZO | PS | PS | PM | PB |
| PB     | ZO | ZO | PS | PM | PM | PB | PB |

| e      | NB | NM | NS | ZO | PS | PM | PB |
|--------|----|----|----|----|----|----|----|
| NB     | PS | NS | NB | NB | NB | NM | PS |
| NM     | PS | NS | NB | NM | NM | NS | ZO |
| NS     | ZO | NS | NM | NM | NS | NS | ZO |
| ZO     | ZO | NS | NS | NS | NS | NS | ZO |
| PS     | ZO | ZO | ZO | ZO | ZO | ZO | ZO |
| PM     | PB | NS | PS | PS | PS | PS | PB |
| PB     | PB | PM | PM | PM | PM | PS | PB |

4. Simulation researches

The following figures are the optimization results, the figure 2 is the dynamic response after optimization and the figure 3 is the error status of system.

![Figure 2. Step response of Fractional-order PID & traditional PID controller](image-url)
As can be seen from the graph, the system rise time of FOPID controller system was significantly shorter than the traditional PID controller, in figure 2, under the control of FOPID controller the rise time of system is 0.132 second, but under the control of traditional PID controller the rise time is 0.439 second. The FOPID controller error dropped speed significantly faster than traditional PID controller. In figure 3 the FOPID controller system can make the system error become to zero by high speed.

5. Conclusions

In industrial production process the FOPID controller system can improve the efficiency and robustness of equipment thus the system can reach the ideal target, the design and application of FOPID will be appeared in various fields.

6. Acknowledge

The work is supported by the National Natural Science Foundation of China (61073101).

References

[1] Tanq KS, Man KF. A realizable architecture for genetic algorithm parallelism. Control Engineering Practice 1998; 6(7): 897-903.
[2] Podlubny I. Fractional order system and fractional order PID controllers. IEEE Transactions on Automatic Control 1999; 44(1): 208-214.
[3] Lubich Ch. Discretized fractional calculus. SIAM J Math Anal 1986; 17(3): 704-719.
[4] Cao Junyi, Cao Binggang. Design of fractional order controller based on particle swarm optimization. International Journal of Control, Automation and Systems 2006; 6(4): 775-781.
[5] Podlubny I. The Laplace transform method for linear differential equations of the fractional order. Fractional Calculus Applied Analysis 1999; 365-386.
[6] Maki Y, Kenneth S Loparo. A neural network approach to fault detection and diagnosis in industrial process. IEEE Transactions on Control Systems Technology 1997; 5(6): 529-541.
[7] Oustaloup A, Moreau X, Nouillant M. The CRONE suspension. Control Engineering Practice 1996; 4(8): 1101-1108.
[8] Jia D, Krogh B H, Talukdar S. Distributed model predictive control. IEEE Control Systems Magazine 2002; 22(1): 44-52.
[9] Wu Zhijian, Kang Lishan, Zou Xiufen. An elite subspace evolutionary Algorithm for solving function optimization problems. Computer Application 2003; 23(2): 13-15.
[10] Srinivasan A, Batur C. Hopfield/APRT-1 neural network-based fault detection and isolation. IEEE Transaction on Neural Network 1994; 5(6): 890-899.