

Parameters turning of the Active-Disturbance Rejection Controller based on RBF Neural Network

Baifen Liu

School of Electronic and Electrical Engineering
East China Jiaotong University
Nanchang, Jiangxi, China
bfliu@4y.com.cn

Ying Gao

School of Basic Sciences
East China Jiaotong University
Nanchang, Jiangxi, China
chair0088@yahoo.com.cn

Abstract

Abstract—the Active-disturbance rejection control (ADRC) has the advantage of strong robustness, anti-interference capability, and it does not rely on the accurate math model of controlled plant. But the parameter self-turning of ADRC isn't as easy as PID controller because there are more parameters to turn in ADRC. In this paper the parameters are self-turning by the Radial Basis Function (RBF) Neural Network. The results of the simulation indicate that the controller has good anti-interference capability and fast response. The robustness of the system is improved.

Keywords- Active-Disturbance Rejection Control (ADRC), Radial Basis Function (RBF), Robustness

1. Introduction

Proportional-integral-derivative (PID) controller, a technique that dates back to 1920s, is still the most widely used control technique in process control although the control hardware has already entered the digital era. To overcome the limitations of PID [1], model based control, such as model predictive control, has been successfully developed. Plenty of linear MPC [2] applications can be found in various industries. The differential geometry method, the inverse system method, the output-tracked method have also been developed in the past years.

The theory of auto-disturbance rejection control proposed these years is a simple and practical method. It was invented by Professor Han Jingqing who served in Chinese Academy of Sciences. This method does not depend on a precise mathematical model of controlled object. It can estimate and compensate the influences of all internal and external disturbances in real time when system is activated. The control has the advantage of simple algorithm, strong robustness, fast

system response and high anti-interference ability. At present, this method has been applied to a number of fields of top science and technology, such as robotics, satellite attitude control, missile flight control, the fire control of tank and the inertia navigation. However, the parameters of ADRC need to turn in these occasions. The study of the parameters self-turning is only at an exploratory stage at home and abroad.

RBF Basis function (RBF) neural network has the ability of expressing arbitrary nonlinear mapping, learning and self-adapting, and also has an effective control to complex and uncertain system. This method becomes independent of model. Neural network is applied to fields of technology, and it gains a number of valuable research results.

So in this paper, parts of the parameters in ADRC are turned by the RBF neural network. And the simulation results indicate the method is practical.

2. Active-disturbance rejection control

Many plants' model can be simplified as follows:

$$x^{(n)} = f(x, \dot{x}, \dots, x^{(n-1)}, t) + w(t) + u \quad (1)$$

Where, w is external disturbance variable, u is control variable, $f(x, \dot{x}, \dots, x^{(n-1)}, t)$ is an uncertain plant. For the two-order plant, its standard ADRC controller structure is usually illustrated as Figure 1[3].

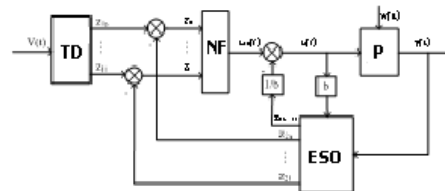


Figure 1. The structure of Active-Disturbance Rejection Controller

According to Figure 1, ADRC controller consists of three main parts: “Transition Process Arranged”, “Nonlinear Feedback” and “Extended-State Observer”.

2.1 Transition Process Arranged

$$\begin{cases} e = v_1 - v_0 \\ fh = fhan(e, v_2, r_0, h) \\ v_1 = v_1 + h \cdot v_2 \\ v_2 = v_2 + h \cdot fh \end{cases} \quad (2)$$

Where, v_0 is the control objective, v_1 is the track signal of v_0 , $fhan(\square)$ is a time optimal integrated function, which detailed expression is described as (3).

$$\begin{cases} d = r \cdot h, d_0 = r \cdot d, y = x_1 + h \cdot x_2 \\ a_0 = \sqrt{d^2 + 8r \cdot |y|} \\ a = \begin{cases} x_2 + \frac{(a_0 - d)}{2} \text{sign}(y), |y| > d_0 \\ x_2 + \frac{y}{h}, |y| \leq d_0 \end{cases} \\ fhan = - \begin{cases} r \cdot \text{sign}(a), |a| > d \\ r \frac{a}{d}, |a| \leq d \end{cases} \end{cases} \quad (3)$$

2.2 Extended-State Observer (ESO)

$$\begin{cases} e = z_1 - y \\ z_1 = z_1 + h \cdot (z_2 - \beta_{01} e) \\ z_2 = z_2 + h \cdot (z_3 - \beta_{02} \cdot fal(e, 0.5, \delta) + b_0 u) \\ z_3 = z_3 + h \cdot (-\beta_{03} \cdot fal(e, 0.25, \delta)) \end{cases} \quad (4)$$

Where, h is the control cycle.

2.3 Output of Nonlinear Feedback (NF)

$$\begin{cases} e_1 = v_1 - z_1, e_2 = v_2 - z_2, \\ u_0 = \beta_1 \cdot fal(e_1, \alpha_1, \delta) + \beta_2 \cdot fal(e_2, \alpha_2, \delta) \end{cases} \quad (5)$$

$$fal(\varepsilon, \alpha, \delta) = \begin{cases} |\varepsilon|^\alpha \cdot \text{sgn}(\varepsilon), |\varepsilon| > \delta \\ \varepsilon / \delta^{1-\alpha}, |\varepsilon| \leq \delta \end{cases} \quad (6)$$

Where, the $\text{sgn}(\varepsilon)$ is the symbolic function. And the other parameters can be found in [4-5].

3. ADRC optimized by RBF Neural Network

The structure of ADRC based on RBF neural network control system is shown in Figure 2.

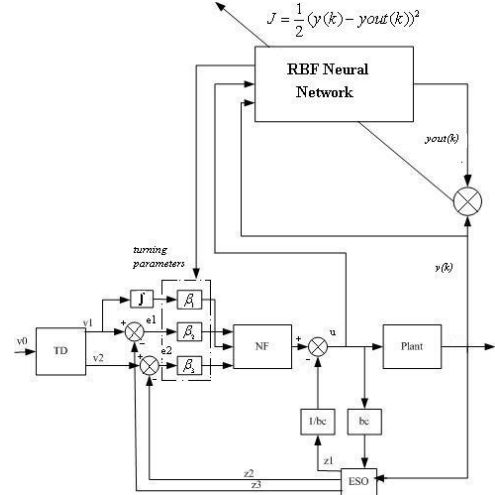


Figure 2. Schematic of ADRC based on RBF Neural Network

3.1 RBF neural network

The RBF neural network is presented by J.Moody and C.Darke in the 1980s which is a feed-forward network with three layers [6-7]. It possesses the capability of local adjustment and can approximate any continuous function in any accuracy. The structure of a typical RBF neural network is shown as Figure 3.

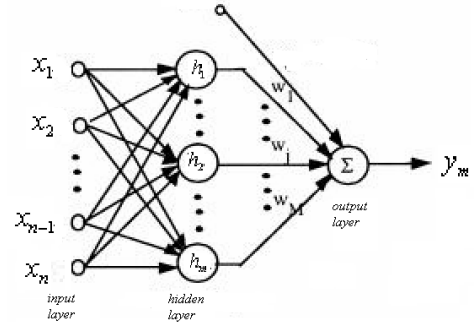


Figure 3. Schematic of a RBF neural network

Radial vector of RBF network adopts multivariate Gaussian function as radial basis function in this paper.

$$h_j = \exp\left(-\frac{\|X_i - C_j\|^2}{2b_j^2}\right) \quad j = 1, 2, \dots, m \quad (7)$$

Where h_j is the output of hidden unit; b_j is the width of hidden unit; C_j is the center of hidden units; X_i is the output vector; $\|\cdot\|$ is Euclid norm.

For an input pattern X_i , the output of single output networks are given by

$$y_{out}(k) = \sum_{j=1}^m \omega_j h_j \quad (8)$$

Where ω_j is the weight between the j_{th} node in the hidden layer and the output unit; m is the number of hidden unit.

3.2 Control of RBF neural network

1) The input and output of controller.

RBF neural network can get more information from speed regulating system to adapt the chances of condition with effective measures. The structure of neural network is determined by genetic algorithm.

The vector of input is:

$$X(k) = [r(k), e(k), e(k) - e(k-1)/T, \sum e(k)]^T \quad (9)$$

Where $r(k)$ is the given input; $\sum e(k)$ is the sum of error.

2) The parameters of neural network

Array of parameter denotes all the parameters of the hidden unit connected with output layer in RBF network. Applications of RBF network have much difficulty in determining the array and the number of the hidden unit. The selection of hidden unit has a very huge influence on mapping ability and effectiveness of network. If there are too few hidden unit, the network can't complete the mission of classification and mapping function; if there are too many hidden units, influence generalization ability and learning efficiency. In order to improve the performance of whole system, it is necessary to seek more suitable learning algorithm of RBF network to determine array of parameter; and the number of hidden unit.

3) Target function of neural network

$$J = \frac{1}{2} (y(k) - y_{out}(k))^2 \quad (10)$$

Where $y(k)$ is the output of the target; $y_{out}(k)$ is the output of the network;

4) Self-turning of the parameters in NF

Use the gradient descent method to turn the parameters in NLESF:

$$\begin{aligned} \Delta\beta_1 &= \eta e(k) \frac{\partial y}{\partial u} xc(1) \\ \Delta\beta_2 &= \eta e(k) \frac{\partial y}{\partial u} xc(2) \\ \Delta\beta_3 &= \eta e(k) \frac{\partial y}{\partial u} xc(3) \end{aligned} \quad (11)$$

η is the learning rate.

4. Simulation and analysis

In this paper, the simulation model adopts a

nonlinear discrete system, whose transfer function is described by

$$\begin{aligned} rin(k) &= 0.5 * \text{sign}(\sin(0.2\pi * t)) + \\ &0.25 * \text{sign}(\sin(0.2\pi * t - \pi / 2)) \end{aligned} \quad (12)$$

Suppose input signal is

$$\begin{aligned} rin(k) &= 0.5 * \text{sign}(\sin(0.2\pi * t)) + \\ &0.25 * \text{sign}(\sin(0.2\pi * t - \pi / 2)) \end{aligned} \quad (13)$$

We adopt proposed algorithm and controller in the simulation. The outputs of ADRC controller based on RBF identification are shown in Figure 4. The simulation results are shown the outputs of the identification network can match the output of the closed-loop controlled plant well. The adaptive turning of NF in ADRC controller parameters are shown in Figure 5. At the same time Jacobian information of identification is shown in Figure 6 in simulation. From Figure 7, we can see that the system output trace the reference information well while using ADRC based on BP neural network. And in Figure 8, we can clearly see that the method of PID control cannot trace the reference signal as well as the previous method.

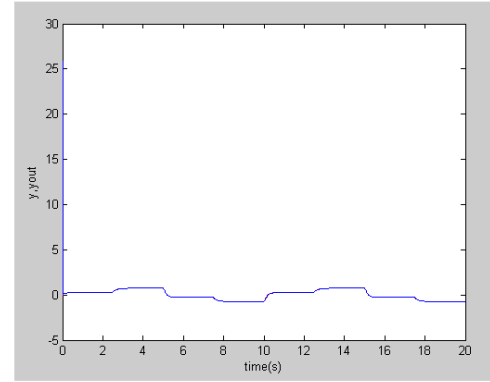


Figure 4. Output of the target and the network

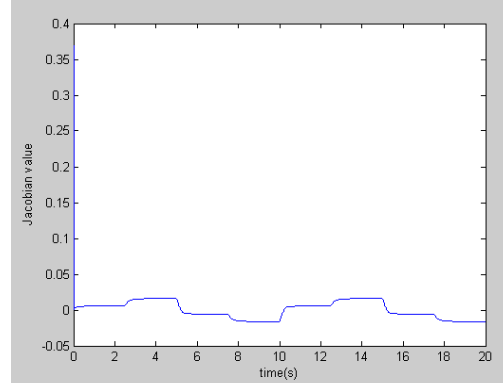


Figure 5. Jacobian information of identification

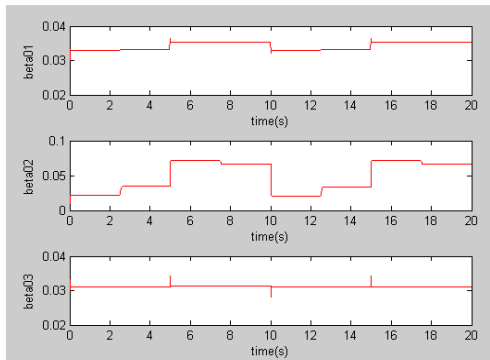


Figure 6. Adaptive turning curve of ADRC parameters

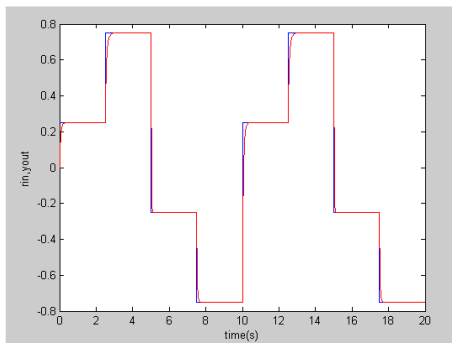


Figure 7. Response of ADRC based on RBF Neural Network

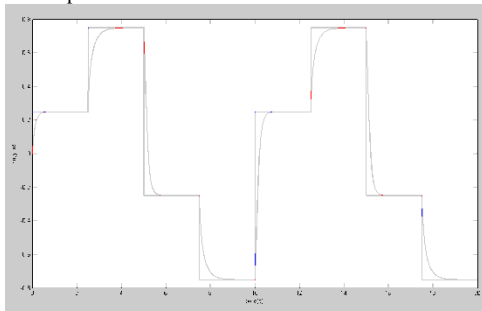


Figure 8. Response of PID

5. Conclusion

Conventional PID controller can hardly work well at different operating condition. A novel controlling method-ADRC based on Radial Basis Function (RBF) Neural Network (NN) is presented in this paper. The controller has advantages of both self-learning capability of neural network and simplicity of ADRC. During the practice, ADRC based on RBF neural network has the superiority such as strong robustness, simple theory. Simulation result shows that the proposed controller has adaptability, strong robustness and satisfactory control performance in the nonlinear and time variable system.

6. References

- [1] Suresh kumar,A;Subba Rao,M;Babu,Y.S.K, "Model reference linear adaptive control of DC motor using fuzzy controller,"TENCON 2008-2008,TENCON 2008, IEEE Region 10 Conference 19-21 Nov.2008 Page(s):1-5
- [2] Buchnik.Y, Rabinovici.R,"Speed and position estimation of brushless DC motor in very low speeds,"Electrical and Electronics Engineers in Israel, 2004.Proceedings,2004 23rd IEEE Convention.2004,pp.317-230
- [3] Pan,Y.;Furuta,K.,"Variable Structure Control with Sliding Sector for Hybrid Systems,"Variable Structure Systems,2006.VSS'06.International Workshop on 5-7 June 2006 Page(s):286-291;
- [4] Ji Zhi-cheng,SHEN Yan-xia, Xue Hua."Study on the adaptive fuzzy control for brushless DC motor,"Proceedings of the CSEE.2005,25[5],pp.104-109.
- [5] Ren yu; Zhou Li-meng;"PMSM Control Research Based on Particle Swarm Optimization BP Neural Network,"Cyberworlds,2008 International Conference on 22-24 Sept.2008 Page(s):832-836.
- [6] Jingqing Han,"From PID to Active Disturbance Rejection Control",Industrial Electronics,IEEE transactions on Volume 56 issue 3, March 2009 Page(s):900-906.