# Self-tuning PID Control of Induction Motor Speed Control System Based on Diagonal Recurrent Neural Network

Chong Chen

## School of Electrical Engineering, Yancheng Institute of Technology Yancheng 224051, PR China cc082120@126.com

#### Abstract

The performance optimization of induction motor speed control system is studied and self-tuning PID controller based on diagonal recurrent neural network (DRNN) is presented in this paper. Neural network control does not require the precise mathematical model of the system, and it only needs to train neural network online or offline, then use the training results to design the control system. It is applicable of the nonlinear, strong coupling and multi variable system, which is composed of inverter and induction motor. The speed regulation control performances are tested on the experimental platform constructed by SIMATIC S7-300 power PLC. The results of experiment indicate that, compared with conventional PID controller, induction motor speed control system, which is controlled by self-tuning PID controller based on DRNN, has better static-dynamic and following performances, stronger anti-interference ability and robustness.

*Keywords:* Induction Motor, Performance Optimization, Speed Control, DRNN Neural Network, PLC

#### **1. Introduction**

Recently, AC induction motor variable frequency speed regulating system fed by a general inverter is widely used in the industrial production field. But it is a nonlinear, strong coupling and multi-variable system. Traditional PID control can hardly meet the control requirements. Therefore, many researchers start to study new control theories and methods, and apply them into the speed control system, such as fuzzy control [1], neural network inverse control [2-5] and ADRC control [6, 7] and so on. The performances of the control system are greatly improved by adopting the methods above. But the adaptation range of parameters is finite and it is difficult to design the control system and the calculation is relatively complicated.

Neural network has strong learning, adaptive capability and nonlinear mapping capability. When it is used to design control system, it only needs to train neural network online or offline, and then use the training results to design, without the mathematical model of controlled object. For nonlinear and uncertain system, it will have strong adaptability and robustness. Diagonal recurrent neural network can not only realize dynamic mapping and memory, but also reduce the adjusted parameters with its self-feedback characteristics, and the amount of computation. Therefore, it is widely used in adaptive control, system identification and modeling field and so on.

In this paper, self-tuning PID control of induction motor variable frequency speed control system based on DRNN is designed. The experiments of anti-disturbance and tracking performance were performed based on S7-300 experimental platform. The results prove that the control system has better static-dynamic and following performances, stronger anti-interference ability and robustness. Thus, the control effect is satisfactory.

## 2. Mathematical Model Of Induction Motor Speed Control System

When the inverter works in V/F mode, the inverter and induction motor is regarded as a whole controlled object, then its mathematical model can be depicted by the state equation of induction motor, as is shown in Figure 1.



Figure 1. Variable Frequency Speed Control System in V/F Mode

The mathematical model of voltage control type induction motor speed control system at  $\alpha, \beta$  stationary coordinate is

$$\left| \frac{d\omega_r}{dt} = \frac{n_P}{J} (T_e - T_L) = \frac{n_P^2}{J} (\psi_{sa} i_{s\beta} - \psi_{s\beta} i_{s\alpha}) - \frac{n_P}{J} T_L$$

$$\frac{d\psi_{s\alpha}}{dt} = V \cos(\omega_l t) - R_s i_{s\alpha}$$

$$\frac{d\psi_{s\beta}}{dt} = V \sin(\omega_l t) - R_s i_{s\beta}$$

$$\left\{ \frac{di_{s\alpha}}{dt} = \frac{R_r}{L_s L_r - L_m^2} \psi_{s\alpha} + \frac{L_r}{L_s L_r - L_m^2} \omega_r \psi_{s\beta} - \frac{R_r L_s + L_r R_s}{L_s L_r - L_m^2} i_{s\alpha} - \omega_r i_{s\beta}$$

$$+ \frac{L_r}{L_s L_r - L_m^2} V \cos(\omega_l t)$$

$$\frac{di_{s\beta}}{dt} = \frac{R_r}{L_s L_r - L_m^2} \psi_{s\beta} + \frac{L_r}{L_s L_r - L_m^2} \omega_r \psi_{s\alpha} - \frac{R_s L_r + L_s R_r}{L_s L_r - L_m^2} i_{s\beta} + \omega_r i_{s\alpha}$$

$$+ \frac{L_r}{L_s L_r - L_m^2} V \sin(\omega_l t)$$
(1)

Where,

$i_{s\alpha}, i_{s\beta}$ : stator current at $\alpha, \beta$ axis;	$\psi_{s\alpha}$ , $\psi_{s\beta}$ : stator flux at $\alpha$ , $\beta$ axis
$R_s, L_s$ : stator resistance and inductance;	$R_r, L_r$ : rotor resistance and inductance
$T_L$ : load torque	J: moment of inertia
$L_m$ : mutual inductance	$n_P$ : number of pole pairs
$\omega_r$ : rotor speed	$\omega_1$ : synchronous frequency

The input variables of the variable frequency speed control system in V/F mode are synchronous frequency  $\omega_1$  and voltage magnitude V, and the output variables are angular velocity  $\omega_r$  and stator flux  $\psi_s$ . Because the stator flux keeps constant fundamentally, when the inverter works in V/F mode with low frequency compensation,

then  $V = k\omega_1 + C$ . Where, c is the compensation value of stator voltage, and k is the proportional coefficient between voltage and frequency. When the power frequency fall into low frequency, the voltage does not fall in tandem accordingly, so as to compensate the voltage-drop caused by stator impedance. The curve of relationship between voltage and frequency is shown in Figure 2.

According to the reference [3], the variable frequency speed control system in V/F mode can be simplified as a single input and single output system.



Figure 2. The Curve of Relationship between Voltage and Frequency

## **3.** The Application of Neural Network Control in Induction Motor Speed Control System

In order to further improve the control performance of induction motor speed control system, a new digital control speed control system is designed based on neural network control in this paper. The control block diagram of induction motor speed control system in V/f mode using self-tuning PID control based on DRNN is shown in Figure 3. In the control system, self-tuning PID controller realizes adaptive adjustment in speed control loop, making the system have stronger adaptive ability, better real-time and robustness[8]-[10].



Figure 3. Actual Speed Control System in V/F Mode with Self-Tuning PID Control Based on DRNN

#### 3.1. Self-Tuning PID Controller Based on DRNN

As is known, the PID controller has the characteristics of simple structure, good stability and high reliability. Therefore, it still occupies important status in modern industrial control situations. But it also has two flaws inherent, one is that the parameters tuning of PID controller is difficult, the other is that self-adaptability is poor. Then the Self-tuning PID controller based on DRNN is adopted in this paper. It uses DRNN network as online identification, according to the external environment variation, adjusting the network weights automatically and tracking the output of the object. Jacobian information obtained by DRNN network can be used to adjust proportional, integral, differential coefficient of PID controller. This controller has the advantages of fast response, strong adaptive and anti-interference ability [11].

#### 3.1.1 Algorithm Principle of Parameter Self-Learning PID Control Based on DRNN

Position type PID controller is adopted in this paper, and the control error is

$$error(k) = r(k) - y(k)$$
<sup>(2)</sup>

Where, r(k) is the given speed  $\omega_{r1}^*$ , y(k) is the actual output speed  $\omega_{r1}$ . The three inputs of PID controller are

$$\begin{cases} x_1(k) = error(k) \\ x_2(k) = \sum_{i=1}^{k} (error(k) \times T) \\ x_3(k) = \frac{error(k) - error(k-1)}{T} \end{cases}$$
(3)

Where, T is sampling time, in this experiment system, it is 0.1s. The three inputs of PID controller,  $k_p$ ,  $k_i$  and  $k_d$  are tuning online by DRNN network.

The control algorithm is

ſ

$$u(k) = k_p(k)x_1(k) + k_i(k)x_2(k) + k_d(k)x_3(k)$$
(4)

Index function can be defined as follows:

$$E(k) = \frac{1}{2}(r(k) - y(k))^2 = \frac{1}{2}e(k)^2$$
(5)

The adjustment of  $k_p$ ,  $k_i$ ,  $k_d$  adopts gradient descent method:

$$Dk_{p} = -\eta_{p} \frac{\Xi}{\Xi} = -\eta_{p} \frac{E}{y} \frac{\Xi}{\Xi} \frac{u}{k_{p}} = \eta_{p} e(k) \frac{?y}{?u} x_{1}(k)$$

$$Dk_{i} = -\eta_{i} \frac{\Xi}{\Xi} = -\eta_{i} \frac{E}{y} \frac{\Xi}{\Xi} \frac{u}{k_{i}} = \eta_{i} e(k) \frac{?y}{?u} x_{2}(k)$$

$$Dk_{d} = -\eta_{d} \frac{\Xi}{\Xi} = -\eta_{d} \frac{E}{y} \frac{\Xi}{\Xi} \frac{u}{u} \frac{u}{k_{d}} = \eta_{d} e(k) \frac{?y}{?u} x_{3}(k)$$
(6)

 $\frac{\|y\|}{\|u\|}$  is the Jacobian information of controlled object, which can be got by identification of DRNN.

**3.1.2. Jacobian Information Identified Of DRNN Network:** DRNN is a kind of recurrent neural network. It is made of three layers network structure, whose hidden layer is recurrent layer. The 3-7-1 network structure is shown in Figure 4.



Figure. 4 The Structure of DRNN

In DRNN network,  $I = [I_1, I_2, ..., I_n]$  is set as input vector, and  $I_i(k)$  is set as the ith neuron input of the input layer. In this paper, the inputs of network are  $I = \{u(k-1), y(k), 1.0\}$ . The algorithm of DRNN is

$$\begin{cases} y_{m}(k) = O(k) = \sum_{j} W_{j}^{O} X_{j}(k) \\ X_{j}(k) = f(S_{j}(k)) \\ S_{j}(k) = W_{j}^{D} X_{j}(k-1) + \sum_{i} W_{ij}^{I} I_{i}(k) \end{cases}$$
(7)

Where

 $X_{i}(k)$ : the jth neuron output of recurrent layer

 $S_{i}(k)$ : the jth recurrent neuron input sum

f(g): double S function

O(k),  $y_m(k)$ : output of DRNN

 $W^D$ : weigh vector of recurrent layer

 $W^{O}$ : weigh vector of output layer

 $W^{I}$ : weigh vector of input layer

The identification error is

$$em(k) = y(k) - ym(k) \tag{8}$$

The identification index is

$$Em(k) = \frac{1}{2}em(k)^2 \tag{9}$$

Learning algorithm adopts gradient descent method

$$\begin{cases} \Delta w_j^o(k) = -\frac{\partial Em(k)}{\partial w_j^o} = em(k) \frac{\partial ym}{\partial w_j^o} = em(k) X_j(k) \\ w_j^o(k) = w_j^o(k-1) + \eta_o \Delta w_j^o(k) + \alpha(w_j^o(k-1) - w_j^o(k-2)) \\ \Delta w_{ij}^I(k) = -\frac{\partial Em(k)}{\partial w_{ij}^I} = em(k) \frac{\partial ym}{\partial X_j} \frac{\partial X_j}{\partial w_{ij}^I} = em(k) w_j^o Q_{ij}(k) \\ w_{ij}^I(k) = w_{ij}^I(k-1) + \eta_I \Delta w_{ij}^I(k) + \alpha(w_{ij}^I(k-1) - w_{ij}^I(k-2)) \\ \Delta w_j^o(k) = -\frac{\partial Em(k)}{\partial w_j^D} = em(k) \frac{\partial ym}{\partial X_j} \frac{\partial X_j}{\partial w_j^D} = em(k) w_j^o P_j(k) \\ w_j^D(k) = w_j^D(k-1) + \eta_D \Delta w_j^D(k) + \alpha(w_j^D(k-1) - w_j^D(k-2)) \end{cases}$$
(10)

Double S function of recurrent layer neuron is

$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \tag{11}$$

$$P_{j}(k) = \frac{\partial X_{j}}{\partial w_{j}^{D}} = f'(S_{j})X_{j}(k-1)$$
(12)

$$Q_{ij}(k) = \frac{\partial X_j}{\partial w_{ij}^l} = f'(S_j)I_i(k)$$
(13)

Where

 $\eta_i$ : learning rate of input layer

 $\eta_D$ : learning rate of recurrent layer

 $\eta_o$ : learning rate of output layer

 $\alpha$  : inertial coefficient

Jacobian information of the controlled object 
$$\frac{\P y}{\P u}$$
 is

$$\frac{\partial y}{\partial u} \approx \frac{\partial ym}{\partial u} = \sum_{j} w_{j}^{O} f^{\dagger}(S_{j}) w_{ij}^{I}$$
(14)

### 4. Experiments

#### 4.1. System Composition

Figure 5 is self-tuning PID control of induction motor speed control experiment system in V/F mode. Industrial computer is used as upper computer, running the monitor software WinCC6.0 and compile software STEP7-V5.4. Siemens S7-300 PLC is adopted to control one set of the Siemens MMV inverter. The parameters of induction motor are  $p_e = 2.2kW$ ,  $U_e = 220/380V$ ,  $I_N = 5A$ , and  $n_N = 1420r/\min$ . When the experiments are performed, the inverter is set in V/F mode and rated frequency is set at 50Hz [2].



Figure 5. Experiment System

#### 4.2. Design of Structured Program

According to the control requirements of induction motor speed control system and the characteristics of structured programming in STEP7, the program structure diagram is designed, shown in Figure 6. The whole programm is realized in the form of blocks.



Figure 6. Program Structure Diagram of the Speed Control System

OB100 is a starting organization block. Before applications launched by user, this block is always first executed. Users can program in the block to initialize CPU. In this paper, we initialize the inverter in the block. The inverter is set stop mode at first and the given frequency is set 0 Hz. We also initialize the counter module FM350-1, and specify the addresses of counter module and channel, and the length of data in the block.

OB1 is the main program of user applications. After running the start program, OB1 is excuted cyclically. The main task of OB1 is to realize the communication between PLC and the inverter, run counter module and deal with timing cycle interrupt. Figure 7 is the flowchart of OB1. In this paper, the communication between PLC and the inverter is realized by FC10. Then FC0 is called to count the pulses emitted by photoelectric encoder, realizing the counting function.

OB35 is cyclic interrupt organization block, and it is also the core of the whole program design. If we wanted to realize self-tuning PID control algorithm based on DRNN in OB35, the function blocks of FB2 is needed to call. Figure 8 is the flowchart of OB35.







Figure 8. The Flowchart of OB35

#### 4.3. Realization of the System Communication

The system communication is composed of three parts: the OPC communication between WinCC and Excel, the MPI communication between industrial computer and PLC, and the PROFIBUS-DP communication between PLC and inverter. WinCC monitor PLC through MPI communication, and collect real-time speed data from PLC through OPC communication.

The PROFIBUS-DP communication realizes PLC control of inverter by constructing the network structure of control system, and finally realizes long distance field control. In this paper, we will mainly introduce the realization of PROFIBUS-DP communication, because it plays the most important role in the system control [12].

**4.3.1. Hardware Connection:** PROFIBUS cable and connector are adopted to connect DP port in CPU module, which is RS485 port, with RS485 port in PROFIBUS-DP module CB15, which is installed in the front panel of MMV inverter. Where, CPU315-2DP is selected as master station and MMV inverter is selected as slave station.

**4.3.2. The Setting Of Mmv Inverter In Electric Drive System:** MMV inverter supports two communication modes, which are USS and PROFIBUS-DP protocol. If the latter is used, CB15 is used accordingly. Firstly, install CB15 in front panel of the inverter and then set the parameter P099 as one, which means selecting the PROFIBUS module. Lastly, the parameters appointed for CB15 are as follows:

P918=7: set PROFIBUS-DP slave station address,

P927=1: set PROFIBUS-DP remote parameter control,

P928=1: set PROFIBUS-DP complete remote control,

P963=7: display PROFIBUS-DP network baud rate, in this paper it is 1.5Mbps.

**4.3.3. Determine User Data Structure:** Variable transmission outline defines five PPO types, while PROFIBUS-DP module CB15 only supports PPO1 and PPO3. PROFIBUS data vertical structure is shown in Figure 9.



Figure 9. The User Data Structure in the PROFIBUS Information Frame

PKW: The parameters setting area of inverter, in this area, the data may be selected. The PPO3 type data structure is selected in this paper, which has process data area, but no parameter area.

PZD: Four bytes process data are adopted, including PZD1 and PZD2, and the data length is two bytes. During the transmission process from master station to slave station (Master $\rightarrow$ Slave), PZD1 is control word, controlling the motor start, stop, and positive negative turns and so on. PZD2 outputs the setting frequency value to MMV inverter. While, during the transmission process from the slave station to the master station (Slave $\rightarrow$ Master), PZD1 is status word, real-time dynamically display the running state of MMV inverter. PZD2 displays the actual speed. Every bit meaning of control word is shown in Tab.1.

 Table 1. Every Bit Meaning of Control Word

Bit	Function description
0	start and stop 1
1	stop 2

2	stop 3
3	enable and forbid operation
4	forbid slope generator
5	enable slope generator
6	enable and forbid set value
7	confirm fault bit
8	crawl clockwise and without crawling
9	crawl anticlockwise and without crawling
10	enable and forbid PZD
11	idle bit
12	idle bit
13	idle bit
14	rotate clockwise and anticlockwise
15	idle bit

**4.3.4. Design of Communication Program:** If we want to start a motor, we should set PZD1=W#16#447E to reset the inverter first, and then set PZD1=W#16#447F to start the motor. The running frequency of the motor is set by PZD2. If we set PZD1=W#16#447E again, then the motor will stop running. The process data PZD1 is transferred to control word interface area PQW256 of the inverter by MOVE instruction in STEP7. Similarly, the process data PZD2 is transferred to set frequency value interface area PQW258 of the inverter in the same way.

#### 4.4. Experimental Results and its Analysis

(A)PID Control

Many experiments are done on the whole system platform, so as to test the control effect. Figure 10(a) and Figure 11(a) are the system response curves with PID control when load sudden change under different given speed. Figure 10(b) and Figure 11(b) are the contrast diagrams with neural network control in the same condition. As can be seen from the diagrams, neural network control can make corresponding control adjustment according to load change, so as to keep speed constant maximum no matter the influence of load change. Therefore, neural network control has stronger anti-interference ability, better adaptability and robustness to parameters change and nonlinear factor of AC motor. The performance index of output speed with load change when the motor runs at different speed are shown in Tab.2 and Tab.3.



(B) Self-Tuning PID Control Based On DRNN

Figure 10. Load Increase at 80s and Decrease at 120s under Two Control Methods



(B) Self-Tuning PID Control Based On DRNN

Figure 11. Load Decrease at 80s and Increase At 120s under Two Control Methods

Tab.2 The Performance Index of Speed With Load Change When	Given
Speed Is 150r/M	

performance	load increase and decrease			load decrease and increase				
index	PID control		proposed		PID control		proposed	
		method				method		
speed variation	-	14.7	-	6.7	14.	-	6.9	-
$(\Delta \omega \%)$	13.3%	%	2.9%	%	2%	10.3%	%	6.7 %
recovery time	5.4	5.8	1.2	1.5	5.4	4.9	1.8	1.6
(s)								

# Table 3. The Performance Index of Speed With Load Change When Given Speed Is 300r/M

performance	load increase and decrease			load decrease and increase				
index	PID control		proposed		PID control		proposed	
			method		method		method	
speed variation	-	7%	-	2.4	6.3	-	2.1	-
$(\Delta \omega \%)$	5.6%		1.8%	%	%	5%	%	1.6%
recovery time	3.4	3.8	0.6	0.8	3.7	3.3	0.7	0.6
(s)								

In order to test the tracking performance of the control system, different expected output trajectory are given as reference input, the response curves of speed are studied to test the control effect. We give the tracking response curves with both PID control and neural network control as comparative study. Initial speed are respectively given periodic rectangular and triangle waveform signals, then the responses of close-loop are obtained, shown in Figure 12 and Figure 13. As can be seen from the curves, no matter the given signal is rectangular or triangle, neural network control of induction speed control system has faster response and higher steady precision, and it can better trace the given signal. The error of both is almost zero. Therefore, the tracking effect is satisfactory. The performance index of speed tracking when it is given square wave and triangular wave, shown in Table 2.

(A)PID Control

International Journal of Control and Automation Vol.8, No.10 (2015)



(B) Self-Tuning PID Based On DRNN

Figure 12. Square Wave Response Curve of Speed Tracking with Two **Control Methods** 



Figure 13. Triangle Wave Response Curve of Speed Tracking with Two **Control Methods** 

Table 4. The Performance Index of Speed Tracking When it is Given Square, **Triangular Wave** 

The index of speed tracking	traditional PID control	self-tuning PID control based on DRNN
response time of square wave	12.6s	2.4s
steady error of square wave steady error of triangular wave	$\pm 1.63\%$	± 1.13%
	$\pm$ 6.86%	± 1.25%

## 5. Conclusion

This paper considers the inverter and the induction motor as a whole controlled object, established the mathematical model of induction motor speed control system in V/F mode at  $\alpha, \beta$  stationary coordinate. According to the structural characteristics and control requirements of the system, self-tuning PID control method based on DRNN is presented. The load and tracking experiments are performed based on S7-300 experimental platform. Compared with traditional PID control, the effectiveness and advancement of this control method is proved. Therefore, the better control effect is acquired, meeting the requirements of high performance control of induction motor speed control system in V/F mode.

# References

- [1] D. Asija, "Speed control of induction motor using fuzzy-PI controller", Proceedings of the 2nd International Conference on Mechanical and Electronics Engineering (**2010**), pp.2460-2463.
- [2] G. H. Liu, H. Zhang, X. Z. Dai, "The application of artificial neural network inverse system in speed control of AC variable frequency induction motor system. Transactions of China Electrotechnical Society, vol. 18, no. 3, (2003), pp.67–71.
- [3] X. Z. Dai, G. H. Liu "The neural network inverse control of the constant ratio of voltage and frequency adjusting speed system", Chinese Electrical Engineering College Journal, vol.25, no.7, (2005), pp.109–114.
- [4] G.H. Liu, C. L. Teng and Z.L. Liao, "Robust control of induction motor speed regulation systan based on fuzzy neural networks inversion", Micromotor, vol.43, no.11, (2010), pp.48–51.
- [5] G.H. Liu and Y. Zhang, "LSSVM Inverse control of two-motor variable frequency speed-regulating system", Advanced Science Letters, vol.4, no.3, (2011), pp.1208-1213.
- [6] X.Q. Liu, L. Tang and L. Zhou, "Active disturbance rejection control of three-motor synchronous control system", Proceedings of the CSEE vol. 30, no. 12, (2010), pp.80–85.
- [7] X.Q. Liu, L. Tang and L. Zhou, "Second-order active disturbance rejection controller applied in threemotor synchronous system", Transactions of China Electrotechnical Society, vol. 27, no. 2, (2012), pp.179-184.
- [8] M. Garcia and AlFredo, "New induction motor V/f control method capable of high-performance regulation at low speeds", IEEE Transactions on Industry Applications, vol.34, no.4, (1998), pp.813– 821.
- [9] X.L. Li, L.H. Shi and J. Li, "Neural network PID control for a water level system", International Journal of Modelling, Identification and Control, vol.11, no.1-2, (**2010**). pp.124-129.
- [10] X.J. Cai, Z.H. Cui, J.C. Zeng and T. Ying, "Self-Adaptive PID-controlled particle swarm optimization", Proceedings of the 26th Chinese Control Conference, (2007), pp.799-803.
- [11] D.Y. Xue, "Computer aided control system design-MATLAB language and its application", Tsinghua University press, (2006).
- [12] C. Chen and G.W. Hu, "Fuzzy PID control of induction motor speed regulating system", International Journal of Wireless and Mobile Computing , vol. 6, no. 4, (**2013**), pp.321-330.

## Author



**Chong Chen,** He received his master degree in Electrical Engineering from Jiangsu University in 2008. He is currently at School of Electrical Engineering, YanCheng Institute of Technology, China. His current research interest includes motion control and power electron application.