

# Control Strategy on a Class of Control System

Wei Wu

College of Information Science and Technology  
Hainan University  
Haikou, P.R.China, 570228  
wuweilu@vip.qq.com

Wencai Du<sup>†</sup>

College of Information Science and Technology  
Hainan University  
Haikou, P.R.China, 570228  
<sup>†</sup>wencai@hainu.edu.cn

**Abstract**—It is acknowledged to build nonlinear system model is the crucial point for control problems, the utilization of conventional control strategies can improve the system performance, i.e., system accuracy, stability and fast convergence. From fundamentally view, a model does not touch changes of its structure often does not provide a perfect solution. Bionic science and technology, especially intelligent neural network technology and genetic evolution, with its excellent adaptability and intelligent evolution, provides effective strategy for complex control systems. This article first describes a general structure of the model two-dimensional input control system, then simulate to verify the suitability of its strategy from the Back-Propagation Neural Networks and Genetic Algorithms for a comparison.

**Keywords**—nonlinear system; intelligent bionic; artificial neural network; genetic algorithm

## I. INTRODUCTION

In actual field, the control systems usually not described by traditional model, the expression transfer function, but matrix. of higher of the input order, the control process will be more complex, large nonlinear multi-channel interference, hysteresis and other constraints, these local problems cant be accurately grasped. Consider these factors to establish a priori-model is extremely complex and whether that model can be applied to an unknown environment is uncertain. If ignore these factors, then results will be very unsatisfied and unstable, cannot be put into operation. Additional control strategies to compensate control will become inevitable. Traditional, such as Smith predicts, fuzzy strategy, pulse modulation can improve its performance on the system limited, but the model structure is remarkable complicated. In strict occasion, it is urgent to adopt an intelligent technology to solve these problems. Bionics control is thereby introduced the control system.

Artificial Neural network as its learning and excellent adaptive capacity is an online control strategy. Receive real-time changes in the characteristics of the controlled object and active learning, the knowledge preserved to the network. Especially Radial Basic Function neural network and BP neural network receives well status. Meantime, even when the controlled object output and the model output exist some system errors, its fault-tolerant features also make the control system owns certain robustness. In addition, the standard structure and neural network and simulation accuracy provide a standard framework to solve the problem of adaptive control of nonlinear systems. If the neural network adopted to uncertain systems will facilitate the effective control of complex manu-

facturing processes thus wins great practical significance. The essential characteristics of neural networks making it is worthy in the application area of nonlinear time-varying characteristics and parameters system.

Genetic algorithm is simulated from Darwinian theory of evolution by natural selection and genetic mechanism of biological evolution as a priority Algorithm, starting from any initial population, through random selection, crossover and mutation operations, resulting in a group of more adapted to the environment body, render the group evolved into the better area from search space so evolving from generation to generation, and finally converge to a group of the most Individuals adapt to the environment, approach to the optimal solution. Genetic algorithm is a global optimization search algorithm, to take intrinsic heuristic random search mechanisms, wins a simple optimization process, parallel processing, not easy to fall into local optimal solution. Genetic algorithms have been widely accepted in pattern recognition, process control, signal Processing, data mining, combinatorial optimization and other fields.

## II. MULTI-INPUT FEEDBACK CONTROL MODEL

A class of typical single-input single-output(SISO) control model can be simplified as:

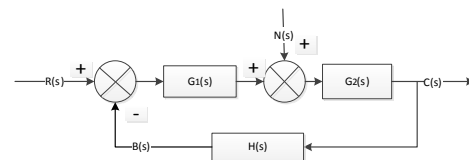


Fig.1. Simple SISO system structure

Feedback control is a control strategy based on the deviation. Compare the desired result with the actual output, then transmit the deviation obtained through the operation back, through zoom adjustment, this parameter influence next output, which is the most typical method of control. It is prevalent in the design of the control system that mathematical model of the system and the actual system there is always a difference, so-called system has unmodeled dynamics; structure and

parameters of the system itself is unknown or time-varying; disturbance on the system is often random and cannot be measured; the control characteristics of the object changes with time or the working environment changes in operation, and hard to know in advance. So it is usually difficult to precise mathematical model to describe the actual engineering systems. Existing theory is to precise finite-dimensional differential equation model based on Lyapunov stability, attributed the disturbance as the initial conditions, to ensure that the system is running when the uncertainty caused by these initial conditions the stability of the response. However, this approach works with noteworthy gap between the real system which is designed not possible to quantitatively grasp the uncertainty to the quality of control performance. Consider a class of uncertain nonlinear systems:

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

where  $y = x, x = (x_1, x_2, \dots, x_n)^T = (x, \dot{x}, \dots, x^{(n-1)})$  is the  $N$ -dimensional state vector,  $u$  is the control input,  $y$  is the system output,  $f$  is unknown continuous function,  $b$  is the unknown constant gain,  $d$  represents dynamic modeling or external interference. Control objectives is to render output  $y$  to track a desired trajectory specified  $y_d$  as well as possible.

### III. TWO KINDS OF BIONIC INTELLIGENT CONTROL TECHNOLOGY

#### A. Back-Propagation Neural Networks

Multi-layer Perception consist of input layer, hidden layer and output layer, the hidden layer which can developed as one or more layers. The number of neurons of input layer dimension depend on input signal, the number of hidden layers and the number of nodes is determined depending on the circumstances, thus make the hidden layer can extract more useful information, complete more complex tasks. Each neuron of multi-layer perception activation function is differentiable sigmoid function, such as:

$$v_i = \frac{1}{1 + e^{-u_i}} \quad (2)$$

wherein  $u_i$  is the input signal of neuron  $i$ ,  $v_i$  is the output signal.

Multi-layer Perception network multiple synapses make it receive more connectivity, computing advantage. BP learning process can be described as follows:

Stage 1: Signals transmit forward: Input signals from the input layer through the hidden layer unit, transmitted to the output layer, produce output. In this process, the network weights remain unchanged; the state of each layer influence its next state, if the output layer cannot get the desired output, then transferred to the error signal back-propagation.

Stage 2: Error signal transmit back: the difference between actual output and the desired output called the error signal, it spread back from the output layer. During this process the network weights revised so that the actual output of the network close to the desired results. We give algorithms below from a specified BP network contains two hidden layers. Let

the input layer is  $M$ , i.e., the  $M$  input signals, where either of the input signal  $m$ . The first layer is  $I$  (neurons), represents any layer. The second layer that is  $J$  (neurons), the output layer is  $P$ .

The input neurons is  $u$ , output is  $v$ , the training sample set of  $X = [X_1, X_2, \dots, X_k, \dots, X_N]$  corresponds to either a training sample  $X_k = [X_{k1}, X_{k2}, \dots, X_{km}]^T, (k = 1, 2, \dots, n)$ , actual output is  $Y_k = [y_{k1}, y_{k2}, \dots, y_{kp}]^T$ , the desired output is  $d_k = [d_{k1}, d_{k2}, \dots, d_{kp}]^T$ , Let  $n$  is the number of iterations, the weights and actual output is function of  $n$ .

Error signal of neuron  $p$  is:

$$e_{kp}(n) = d_{kp}(n) - y_{kp}(n) \quad (3)$$

Defined energy error signal of neuron  $p$  is  $\frac{1}{2}e_{kp}^2$ , then the error energy sum of all neurons in the output layer is  $E(n)$ , which can be expressed as:

$$E(n) = \frac{1}{2} \sum_{p=1}^P e_{kp}^2 \quad (4)$$

The following analysis error signal back-propagation procedure:

Weights correction between hidden layer and output:

$$\Delta w_{jp}(n) \propto \frac{\partial E(n)}{\partial w_{jp}(n)} \quad (5)$$

From its transformation:

$$\frac{\partial E(n)}{\partial e_{kp}(n)} = \frac{\partial E(n)}{\partial e_{kp}(n)} \cdot \frac{\partial e_{kp}(n)}{\partial y_{kp}(n)} \cdot \frac{\partial y_{kp}(n)}{\partial u_p^P(n)} \cdot \frac{\partial u_p^P(n)}{\partial w_{jp}(n)} \quad (6)$$

According to these conclusions:

$$\begin{aligned} \frac{\partial E(n)}{\partial e_{kp}(n)} &= e_{kp}(n), \frac{\partial e_{kp}(n)}{\partial y_{kp}(n)} = -1, \\ \frac{\partial y_{kp}(n)}{\partial u_p^P(n)} &= f' u_p^P(n), \frac{\partial u_p^P(n)}{\partial w_{jp}(n)} = v_j^J(n) \end{aligned} \quad (7)$$

Let the local gradient:

$$\delta_p^P(n) = -\frac{\partial E(n)}{\partial u_p^P(n)} = f' u_p^P(n) \cdot e_{kp}(n) \quad (8)$$

when the activation function is a logic function:

$$f(x) = \frac{1}{1 + e^{-ax}}, a > 0, -\infty < x < \infty \quad (9)$$

According to Delta learning rule, correction amount is:

$$\Delta w_{jp}(n) = -\eta \frac{\partial E(n)}{\partial w_{jp}(n)} = \eta \delta_p^P(n) v_j^J(n) \quad (10)$$

where  $\eta$  is the step size, the weights between next iteration in any one hidden layer and output layer node is:

$$w_{jp}(n+1) = w_{jp}(n) + \Delta w_{jp}(n) \quad (11)$$

According to the same rule, weights between hidden layer  $J$  and the hidden layer  $I$  is:

$$w_{ij}(n+1) = w_{ij}(n) + \Delta w_{ij}(n) \quad (12)$$

weights between the input layer  $M$  and hidden layer  $I$  is:

$$w_{mi}(n+1) = w_{mi}(n) + \Delta w_{mi}(n) \quad (13)$$

## B. Genetic Algorithm Theory

Genetic Algorithm is a mathematical process of biological evolution mode emulation, different from the traditional search algorithms, it is randomly generated from a set called “population” of the initial solution to start the search process.

The genetic algorithm (GA) is an optimization and search technique based on the principles of genetics and natural selection. A GA allows a population composed of many individuals to evolve under specified selection rules to a state that maximizes the fitness (i.e., minimizes the cost function).

Each individual in the population of a solution to the problem referred as “chromosomes”. Chromosome is a string of symbols, such as a binary string, evolving a subsequent iteration, known as hereditary coding by many methods, the mathematical formulas for the binary encoding and decoding of the  $n$ th variable,  $p_n$  are given as follows:

$$p_{norm} = \frac{p_n - p_{low}}{p_{hi} - p_{low}} \quad (14)$$

where  $p_{norm}$  is normalized variable,  $0 < p_{norm} < 1$ ,  $p_{low}$  is the smallest variable value,  $p_{hi}$  is the highest variable value.

Survival of the fittest translates into discarding the chromosomes with the highest cost. In every generation, with “fitness value” to measure the quality of chromosomes to generate the next generation called offspring, through chromosomal crossover or mutation operation.

We can make the following analysis: Set population size, determine chromosomal genes consist of three parts, calculate the fitness function value  $f_i$ , which close to a specified data of system output. Then the same as the total population fitness value  $F$ ,

$$F = \sum_{i=1}^n f_i \quad (15)$$

where  $f_i$  stands for each fitness of each individual. divide  $F$  by a single chromosome fitness value  $f_i$  we get choosing probability  $p_i$ .

$$p_i = \frac{f_i}{F} \quad (16)$$

Select the individual with high survival probability for crossover operator. We usually choose to mutate 0.20 of the population, except for the best chromosome. To continue the next round, repeat indefinitely. Finally take roulette methods produce offspring, utilize new chromosome cross operation set, based on fitness value and procedure of selection, crossover and mutation; produce the third generation of the population. Genetic Algorithm circulates this process and receives the best individual for the best answer.

## IV. MODEL SIMULATION

### A. Model of MISO System

The model was given bellow, also test sample was given in table 1. where  $f(x) = (1 + x_1^{-2} + x_2^{-1.5})^2$ .

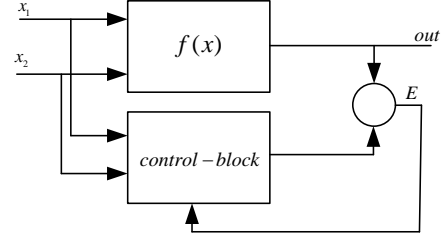


Fig.2. A simple MISO system structure

Table1. Train Sample of MISO system

Group	$x_1$	$x_2$	$y$	Group	$x_1$	$x_2$	$y$
1	1.4	1.8	3.7	13	1.67	2.81	2.47
2	4.28	4.96	1.31	14	2.03	1.88	2.66
3	1.18	4.29	3.35	15	3.62	1.95	2.0
4	1.96	1.90	2.7	16	1.67	2.23	2.75
5	1.85	1.43	3.52	17	3.38	3.70	1.51
6	3.66	1.60	2.46	18	2.83	3.70	1.60
7	4.51	1.52	2.51	19	1.48	4.44	2.44
8	3.77	1.45	2.70	20	1.19	1.53	4.99
9	4.84	4.32	1.33	21	1.67	2.81	2.47
10	1.05	2.55	4.63	22	4.10	1.71	2.27
11	4.51	1.37	2.80	23	1.65	1.38	3.94
12	1.84	4.43	1.97	24	2.00	2.06	2.52

### B. Simulation Results and Comparison

First we use MATLAB for BP program testing, the results were given in figure 3.

From figure 3, it is apparently average error (MSE) soon converged to extremely tiny after about 100 steps, this speed is acceptable and can be put into virtual applications.

GA theory provides an solution for this problem, it performs not so well as BP method. in figure 3 and 4, for series testing data, BP can constrain error to a limited boundary of  $(-0.2, 0.1)$ , while GA can only reach to  $(0.3, 0.3)$ , and vibrate a lot even in many generations. Adjust parameters in this procedure is also invalid. But this result is also pleased, especially in such complex system.

## V. CONCLUSIONS

This paper first gives a brief description of the control system model, introduced on the basis of neural network algorithms and control technologies bionics of the Genetic Algorithm, finally through an example these two methods to be demonstrated.

For complex nonlinear control systems, neural network technology shows its great application value. Nonlinear neural network has a strong adaptability and self-learning ability, can approximate any nonlinear function and system identification. The neural network is used for linear and nonlinear system identification, improve the accuracy of time delay system parameter identification remarkably, so more conducive to the system controller design. The genetic algorithm, in addition to its theoretical mathematical basis for in-depth study,

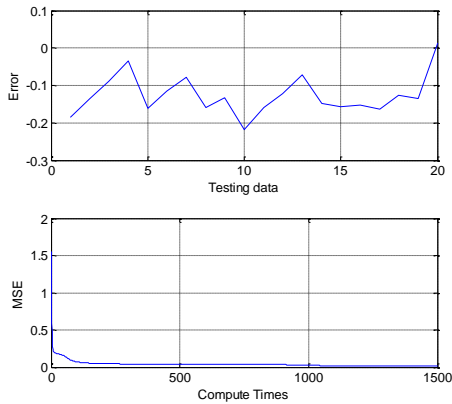


Fig.3. Back-Propagation Networks for simple MISO system

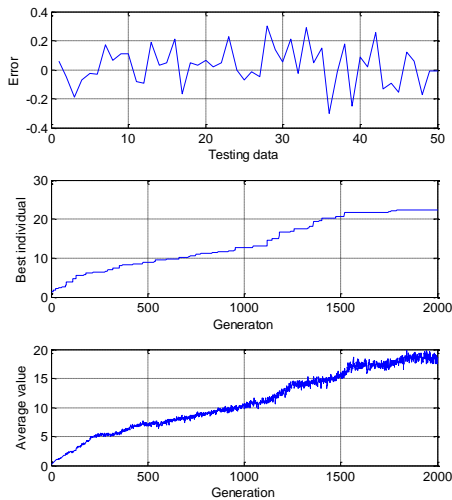


Fig.4. Genetic Algorithm of simple MISO system

should also be in the practical application-specific problems and identify suitable for this application of improved genetic algorithms.

To solve the problem of large population size, should be used in different strategies for different problems, on one hand to optimize the structure of the problem itself improved genetic algorithm, on the other combine traditional structure with advanced intelligent control tactics is necessary.

#### REFERENCES

- [1] Wu Wei, Zhong Sheng, et al. A study on PID intelligent optimization based on radial basis function neural networks[J]. 2013 3rd International Conference on Consumer Electronics, Communications and Networks (CECNet), pp.57-60.(2013)
- [2] Zhang Tianping, Mei Jiandong, Shen Qikun, Jiang Haibo Adaptive fuzzy sliding model control with unknown virtual control coefficients and perturbed uncertainties. Proceedings of the Fourth International Conference on Machine Learning and Cybernetics. Vol.2, pp.857-861.(2005)
- [3] Zhang Tianping, Mei Jiandong, Mao Yuqing, Chen Jing Adaptive neural network control of nonlinear systems with unknown deadzone model- Proceedings of the Fourth International Conference on Machine Learning and Cybernetics. Vol.3, pp.1351-1355.(2005)
- [4] Q. X. Zhu. The multi-step predicting controllers for deterministic networked control systems[J]. Acta Armamentarii, 30(8), pp.1124-1128.(2009)
- [5] Randy L. Haupt, Sue Ellen Haupt. PRACTICAL GENETIC ALGORITHMS (SECOND EDITION), pp.39-53.(2004)
- [6] Seiler P, Sengupta R. An approach to networked control[J]. IEEE Trans Control, 50(3), pp.356-364.(2005)