

Automatic Predictive Control Algorithm and Emulation Technique Study

Yang Zhongguo, Cai Tianfang

College of Mechanical and Electrical Engineering, Zaozhuang institute, 277160, China

Keywords: Nonlinearity, predictive control, fuzzy neural predictive control, composite particle swarm optimization algorithm.

Abstract. This paper, aiming at the predictive control problem in nonlinear complex system, puts forward fuzzy neural predictive control algorithm based on composite particle swarm optimization algorithm. Under the situation of unknown system model, prediction model is established by combining fuzzy logic with neural network. Meanwhile the effective and feasible predictive control method with well control performance is proposed by making use of composite particle swarm optimization algorithm to complete rolling optimization, so as to provide reliable theoretical foundation for solving practical problems of control system.

Introduction

Modern control theory must be based on precise object parameter model, but in fact, various control systems often have characteristics of nonlinearity, time-dependent nature, strong coupling and indeterminacy, etc [1,2,3], so it is hard to obtain precise parameter models. Therefore, the controller designed by modern control theory often is difficult to reach the expected control effect. Model Predictive Control (MPC, for short predictive control) is emerged under this background. Predictive control is a kind of control thinking, based on practice and by combining with control theory. Predictive control has already obtained extremely successful application, especially for petrochemical engineering region [4], in objects' precise parameter model that is hard to obtain, including petroleum, electric power and aviation, etc. Compared with traditional control algorithm, predictive control algorithm has complicated algorithm, is hard to realize design steps, has larger online calculated amount and is hard to satisfy real-time requirement of control. Therefore, generally speaking, predictive control is applied to the system that doesn't have higher requirement on timeliness. Moreover, it requires professional companies to implement. It is bad for general engineering staff to master.

This paper proposes a kind of fuzzy neural predictive control algorithm based on composite particle swarm optimization algorithm, and makes comparison with standard generalized predictive control algorithm, neural network predictive control algorithm, and fuzzy neural network predictive control algorithm. The result shows that the algorithm proposed in this paper has good control performance, higher predictive precision, excellent convergence, good tracing performance, and stronger robustness. Its control result is satisfactory.

Establishment of Fuzzy Neural Network Predictive Model

In recent years, fuzzy neural network is an effective way to study control problems in uncertain complicated nonlinear system. It doesn't depend on precision of controlled objects' mathematic model in the system. It can confirm structural parameters of fuzzy neural network on line. Meanwhile, it can adapt to fuzzy control rule and related parameters of subordinating degree function with the change of controlled objects' parameters. This paper selects TS fuzzy neural model that is opt to express nonlinearity and can recognize dynamically. Its model structure is shown in Figure 1. The arrow in the figure shows the direction of signal.

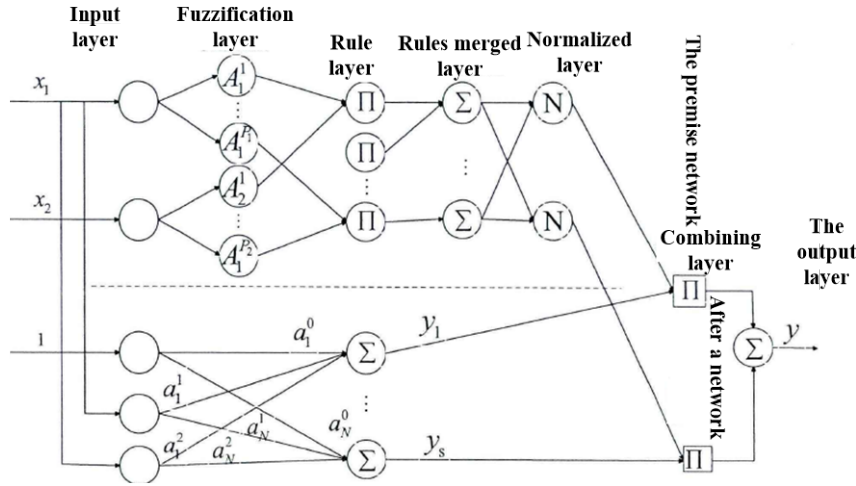


Figure 1-1 FNN Structure Based on T-S Fuzzy Model

This structure is divided into antecedent network and consequent network. Antecedent network can make use of system information to fitness of multiple local linear working areas based on specific characteristics of different working conditions; Consequent network is to recognize parameters of every subsystem. Every sub-network generates an output.

Any linear complex process can be decomposed into multiple local linear regions. Then it is constituted through fuzzy dynamic composition. Variable state in every local linear region can be described by using a fuzzy rule R_i , as follows: here, A_i^n is fuzzy language variable value of input space; a is consequent network; R is fuzzy rule number.

$$R_i : \text{If } x_i \in A_i^1, x_2 \in A_i^2, \dots, x_n \in A_i^n, \text{ then} \quad (1)$$

$$y_i = a_i^0 + a_i^1 x_1 + a_i^2 x_2 + \dots + a_i^n x_n, (i = 1, 2, \dots, n) \quad (2)$$

Input layer: the nodal points in this layer are corresponding to the number of feature vectors. Input of nodes is input variable value. Output of nodes is corresponding to output values. Its function is to transmit input language variable value JC to the sub-layer. The relationship between input and output is:

$$f_{il} = x_{il} = x_i, i = 1, 2, \dots, n \quad (3)$$

Fuzzy layer: the number of nodes is the total number of every input variable language value, being assumed to be M . Every node represents the fuzzy subset to complete the calculation of subordinating degree function and regard it as memory unit. Its subordinating degree function adopts Gaussian function, as follows: the center of subordinating degree function is α_i^j . The width is σ_i^j . θ_i^j is the feedback gain from first-order delayed feedback to the second-floor nodes.

$$u_{A_i^j}(x_i) = \exp\left(-\frac{(x_i - \alpha_i^j)^2}{\sigma_i^j}\right), i = 1, 2, \dots, n, j = 1, 2, \dots, p_i \quad (4)$$

$$u_{y_j}(k) = \exp\left(-\frac{(x_i(k) + u_{y_j}(k-1)\theta_i^j - \alpha_i^j)^2}{\sigma_i^j}\right), i = 1, 2, \dots, n, j = 1, 2, \dots, p_i \quad (5)$$

and

Rule layer: every node in this layer represents a fuzzy rule. The number L in the fuzzy rule is the total number of nodes. Its function is used to match with fuzzy rule. Subordinating degree function is the input of nodes. The multiply imported by this layer is output, namely:

$$u_j = \prod u_{y_j}, i < n, j = 1, 2, \dots, L \quad (6)$$

Selecting parameter σ suitably can make fuzzy neural model fit into input and output data at will. The smaller σ is; the smaller error of fitting will be. However, system nonlinear curve will be tougher. If the curve is not smooth, it will result in reducing generalization ability of the system.

Therefore, the purpose of selecting parameter σ is to ensure the balance between fitting performance and generalization ability. Generally speaking, the smaller σ can make system nonlinear function close to nonlinear degree as much as possible. However, the larger σ can remove the data with interfering noise.

According to the above-mentioned description, use algorithm (7) to conduct error processing. Model expected output is represented by $y_1(k)$. $y(k)$ is the practical output of the model. q is the number of network training data. To obtain prejudiced deviation from neural network connection weight α_i^j , it can obtain its learning algorithm, as shown in Formula (8). In the formula, β is learning rate.

$$E = \sum_{k=1}^q [y_1(k) - y(k)] / 2 \quad (7)$$

$$a_i^n(k+1) = a_i^n(k) - \beta \frac{\partial E}{\partial a_i^n} = a_i^n(k) + \beta [y_1(k) - y(k)] \bar{u}_t y_t \quad (8)$$

Selection of Predictive Controller's Rolling Optimization Algorithm

In the paper, mutation operation in genetic algorithm introduces standard particle swarm optimization to constitute composite particle swarm optimization algorithm (GA-PSO). Particle speed and location update formula are shown in the formula. The specific algorithm is shown in Figure (2).

$$v(k+1) = c_0 v(k) + c_1 [p_{best}(k) - x(k)] + c_2 [(g_{best}(k) - x(k))] \quad (9)$$

$$x(k+1) = x(k) + v(k+1) \quad (10)$$

(1) Particle swarm initialization. To generate a group of particles at random, the number of particles is marked as the initial speed and location of generating particles at will.

(2) To evaluate fitness and update overall extreme value and individual extreme value. According to function of fitness, obtain the fitness value of particles. If it is superior to current individual extreme value, it will adapt to current fitness value and regard as the individual extreme value of this particle. Compared with current overall extreme value, if optimal individual extreme value of all particles is more superior, it will be considered as the overall extreme value g_{best} .

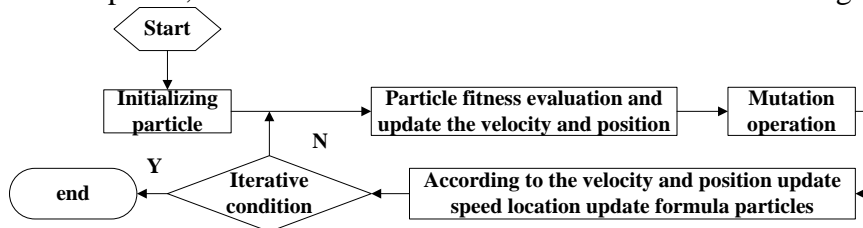


Figure 2 Flow Chart of Composite Particle Swarm Optimization Algorithm

(3) Aiming at random number generated by particle i that is distributed evenly between 0 and 1 is marked as r . If $r < P_m$ mutation probability, local optimum of the particle conducts mutation operation. In other words, first of all, it generates random number η that presents normal distribution and satisfies criteria. Then it increases disturbance on the local optimum. If local optimum is marked as P_{best} , the increased disturbance is presented as $P_{best} = P_{best} * (1 + \eta)$.

(4) According to Formula (9) and Formula (10), update speed and location of every particle.

(5) Check it whether it satisfies terminal condition, if it satisfies condition, it terminates the iteration. Otherwise, return to step (2).

This paper regards objective function of generalized predictive control as rolling optimization objective function of nonlinear intelligent predictive controller, as shown in Formula (11). Consider its reciprocal as the fitness function of composite particle swarm optimization algorithm, control increased sequence $U = [\Delta u(k), \Delta u(k+1), \dots, \Delta u(k+n-1)]$, regard the minimum of fitness function as the evaluation criterion of performance indicator.

$$J = \sum_{j=1}^n [y(k+1) - \omega(k+1)]^2 + \sum_{j=1}^m \lambda(j) [\Delta u(k+j-1)]^2 \quad (11)$$

Design of Fuzzy Neural Network Predictive Control System Based on Composite Optimization Algorithm

In practical control system, external disturbance, uncertainty and time-dependent nature, etc. will result in the error between system's predictive value and actual value. It is an important factor that impacts system performance. Therefore, in the intelligent predictive controller, the paper also designs feedback correction link. The error between system's predictive value and actual value is shown in Formula (12). Formula (12) needs predictive error to modify current prediction, as shown in Formula (13).

$$e(k) = y(k) - y_m(k) \quad (12)$$

$$y_p(k+1) = y_m(k+1) + e(k) \quad (13)$$

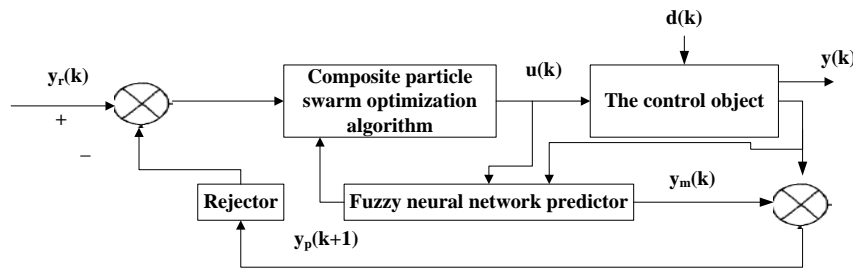


Figure 3 Fuzzy Neural Network Predictive Control System Chart Based on Composite Particle Swarm Optimization Algorithm

Based on the above-mentioned model structure, the intelligent predictive control system structure diagram is shown in Figure 3. The algorithm steps of fuzzy neural network predictive control based on composite particle swarm optimization algorithm are as follows: (1) according to system output and given value, calculate reference trajectory sequence $w(k+1)$; (2) Generate predictive output $y_p(k+j)$ through fuzzy neural network predictive model; (3) According to Formula(13), calculate predictive error; (4) make use of composite particle swarm optimization algorithm to solve the Formula (11) minimum of performance indicator, so as to obtain the optimal control sequence.

Example Verification Simulation

Assumed that system nonlinear model is $y(k+1) = 0.85y^2(k) + \sin u(k) + 1.25u(k-1)$, predictive length $n=10$; control length $m=3$; control weight coefficient $\lambda=0.5$; soft coefficient $\alpha=0.35$; mutation probability $P_m=0.01$. conduct simulation study on the same condition of initial conditions:

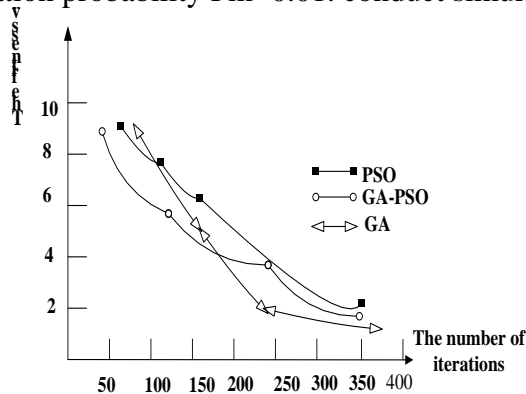


Figure 4 Iterative Optimization Processes of Different Optimization Algorithms

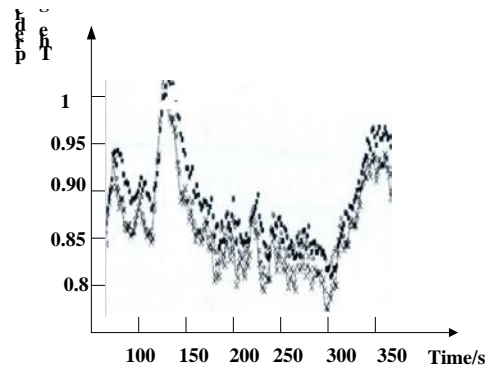


Figure 5 Comparisons between Predictive Value and Given Value Based on Different Predictive Methods

Figure 4 represents iterative optimizing processes of different optimization algorithms. The purpose is to verify superiority of this algorithm proposed in this paper and is used to regard as the effectiveness of rolling optimization algorithm in predictive control process. In the simulation study process, it makes comparisons between this algorithm and traditional optimization algorithm. The simulation result is shown in the figure. Obviously, traditional optimization algorithm has excellent optimization performance. With the increase of iterative times, it also can reach the purpose of optimization. However, by comparing with traditional optimization algorithm, compound optimization algorithm has faster rate of convergence. There is no need for too many iterative times. Thus, it can obtain comparatively satisfactory fitness value.

Select parts of discrete values as comparison objects at random, make comparisons with this algorithm and fuzzy predictive control algorithm and fuzzy neural predictive control algorithm, which have comparatively mature researches. The result is shown in Figure 5. Obviously, the error between predictive value and given value based on composite particle swarm optimization algorithm proposed by this paper is obviously smaller than the other two methods. It indicates that the new algorithm proposed in this paper has more precious prediction, higher predictive precision, stronger adaptive ability and better control performance. In nonlinear system, it can present more satisfactory predictive results.

Conclusions

Practical control objects have many characteristics, such as nonlinearity, time-dependent nature, time lag and multivariable, etc. Meanwhile, because a controlled object is easy to be disturbed by the outside in different environments, it has uncertainty. This indicates that fuzzy neural network model established by fuzzy logic and neural network fusion is regarded as the predictive model of predictive control. By considering composite particle swarm optimization algorithm as rolling optimization algorithm, it establishes a kind of new-type predictive control algorithm and conducts simulation study on new-type algorithm proposed by this paper on the basis of MATLAB platform. Meantime, in order to verify the effectiveness, feasibility and superiority of this algorithm, the author makes comparison with fuzzy neural predictive control algorithm, neural network predictive control algorithm and standard generalized predictive control algorithm. The result is satisfactory.

References

- [1] Fage, Study on Complicated System Engineering [J], Journal of University of Shanghai for Science and Technology, 2011, 33(6):641-5650.
- [2] Zou Tao and Li Haiqiang, Bilayer Structure Predictive Control of Multivariable System with Integration Element Loops [J], Journal of Zhejiang University: Engineering version, 2011, 12(45):2079-2087.
- [3] Zou Tao, Ding Baocang and Zhang Duan, Introductory Theory on Model Predictive Control Engineering Application [M], Beijing: Chemical Industry Press, 2010.
- [4] Zhang Ju, Hao Ping and Wang Wanliang, Expansion Algorithm in Feasible Region of Explicit Model Predictive Model System Design Based on Multiple Parameters' Planning[J], Control Theory and Application, 2009, 26(3): 305-308.
- [5] Li Dewei, Study on Equal Integrated Optimizing Strategies of Predictive Control [J], ACTA AUTOMATICASINICA, 2007, 33(3):302-308.
- [6] Zhu Yanwei and Yang Leping, Robust Constraint Model Predictive Control of Spacecraft Close Range's Relative Movement [J], Control Theory and Application, 2009, 26(11):1273-1277.
- [7] Jingwu Xingsheng, Application of Optimum Control in Rolling Stability of Metamorphic Core's Reentry Vehicle [J], Journal of Astronautics, 2009, 30(1):164-170.

- [8] Guo Wei et al., Predictive Function Excitation Controller Based on Fractional Order of Time Domain [J], Journal of Instrument and Meter, 2012, 32(11):2461-2467.
- [9] Bei-bei L, Yang Chunjie and Jian C, Supply Chain's Operational Optimization and Simulation Scheme Based on Model Predictive Control[J], Journal of System Simulation, 2008, 20(16):4221-4226.
- [10] Gao Xianwen, Zhang Li and Wang Jiesheng et al., Constraint High-Effective Predictive Control [J], Control Theory and Application, 2011, 28(6):855-860.