Application of Improved PSO and PID Neural Network Controller for Multi-motor Proportion Synchronous Control System

Zhenxin Gao and Jianhong Sun

ABSTRACT

PID neural network (PIDNN) controller was designed, and the adaptive mutation particle swarm optimization (PSO) algorithm is adopted to train and update its weights. The adaptive mutation PSO-PIDNN learning algorithm of motor synchronization control was put forward. According to the need of engineering, the algorithm was simplified. In proportion of multiple motor synchronization control system simulation show that the simplified PSO-PIDNN controller also can effectively realize the motor synchronization control, solve the problem of the parameter setting is not easy for traditional PID control, has good stability and fast convergence speed, no overshoot.

INTRODUCTION

Multi-motor synchronization control is the core problem in printing, paper, binding, textile and other high precision, high speed in the mechanical transmission system. The stand or fall of synchronous control algorithm directly affects the system's reliability and the quality of the products [1].

Because of its simple structure, good robustness, high reliability and other features, traditional PID [2] synchronous control is widely used in the synchronous...
control system. But the key problem of designing PID controller is how to choose the proportion, integral, differential coefficient, and it makes the application of PID controller is restricted. In order to overcome the shortcoming of traditional PID control, control community puts forward some methods of PID setting. Such as the literature [3] setting PID fuzzy control algorithm, but the realization of the fuzzy control is too dependent on human experiences and the scope of its application was very limited. In order to achieve effective control, the neural network was introduced into synchronous control system. Literature [4] uses the combination of neural network and fuzzy algorithm, but its underlying layer is overmuch, increase the complexity of the system and the amount of calculation. Literature [5] uses single neuron PID synchronous control, this kind of single-layer network only has linear classification ability, so this controller in the complex system control is difficult to achieve good performance.

Due to the learning convergence speed of general neural network is slow, easily trapped in local minimum point and there is no guarantee that real-time control system. At the same time, the hidden layer unit number and the connection weights is difficult to determine, thus restricting its extensive applications in control system. Literature [6] proposes a new type of neural network model -- PID neural network (PIDNN) model. The PIDNN is an amalgam of PID control and neural network, thus it has the advantages of neural network and PID control and overcoming the shortcomings of traditional control method and the general neural network. But, PIDNN still adopts the error back propagation algorithm that general neural network used, and there is easy access to local convergence, long training time and other shortcomings.

Particle swarm optimization (PSO) [7] is a kind of new, global optimization algorithm. It obtained the ideal effect in terms of solution to find the optimal solution.

In conclusion, in this paper, the PSO algorithm combined with PIDNN and used multi-group particles search the global optimal position as the network weights in the whole solution space, and updated of particle position and velocity substitute for updates of weights. Then, the standard PSO was improved, and PIDNN was optimized according to the engineering demand. Finally, a simplified adaptive mutation PSO - PIDNN controller was designed and simulated for the parallel multiaxial proportional synchronous control system.

**PSO ALGORITHM AND THE PIDNN**

**Particle Swarm Optimization (PSO)**

The basic idea for PSO is to simulate the process of the flock foraging to search the global optimal point with system under a certain performance index.

According to the literature [7],the particles update their speed $V_i$ and location $X_i$ by following type. Namely, the standard PSO algorithm:
\[ V(k+1) = wV(k) + c_1 \times \text{rand} \times (pbest - X(k)) + c_2 \times \text{rand} \times (gbest - X(k)) \] (1)

\[ X(k+1) = X(k) + V(k) \] (2)

In type: \( i=1,2,3...M \), is populations in the algorithm; \( pbest_i \) is the optimal position of each particle; \( gbest \) is the best location for population; \( k \) is the number of iterations; \( w \) is inertia weight, make \( w = 0.5 + \text{rand}(0)/2 \), in the analysis of dynamic nonlinear problem for synchronous control; \( c_1 \) and \( c_2 \) are two positive acceleration coefficient, general taking \( c_1 = c_2 = 2 \); \( \text{rand}_1 \) and \( \text{rand}_2 \) as a random number between \([0, 1]\).

ADAPTIVE MUTATION PSO ALGORITHM

Due to PSO has some disadvantages, such as premature convergence, low search accuracy and late iterative efficiency. This paper improves the standard particle swarm optimization algorithm, and blends the mutation of genetic algorithm into the particle swarm, that is to initialize some variables again at a certain probability. This can make the particles out of the previous optimal value position and carry out the search in a bigger space, thus improving the ability of algorithm to find the best value.

Introducing the mutation probability \( P_j \). In order to avoid the mutation affects the stability of the algorithm, the mutation probability need to be selected reasonably. According to the experiment, taking \( P_j > 0.9 \) and occurring variation. The MATLAB code is as follows:

```matlab
For k=1:N
    P_j=rand()
    If P_j>0.9,
        X_j(k)=rand
    end
end
```

Figure 1 is the standard particle swarm optimization algorithm and improved particle swarm algorithm simulation results under the same condition. It can be seen that the adaptive PSO algorithm can quickly jump out of local optimum and come into a stable state, get the optimal solution.
Improve PID Neural Network (PIDNN)

Traditional PIDNN is a three layer forward neural network with 2*3*1 structure. The features of PIDNN is that the middle layer defines the proportion, integral and differential three neurons, equivalent to put PID control into the neural network and make it has a good network dynamic characteristics.

THE PIDNN CONTROLLER ALGORITHM AND VALUE FUNCTION

PIDNN hidden layer has three neurons, they are proportion, integral, and differential dollar. The status of each neuron and input/output function in the forward algorithm as shown in literature [6].

Value function is the basis of neural network training, commonly used with the mean square error function [8]. Whole neural network uses the minimum mean square error (MSE) as the training of learning principles and objectives. In this paper, the PSO algorithm combined with PIDNN and using multi-group particles search the global optimal position as the network weights in the whole solution space, instead of traditional error back propagation algorithm, and updates of particle position substitute for updates of weights.

OPTIMIZING FOR PIDNN ALGORITHM

According to the characteristics of the PID control algorithm: e(k)=r(k)-y(k), selecting the input layer to hidden layer connection weights of the initial value and make the tracking error e as an intermediary input neurons. So input layer to hidden layer connection weights of the initial value are

\[ w_{1j} = 1, w_{2j} = -1 \quad (j = 1,2,3) \]  \hspace{1cm} (3)
\[ w'_1 = k_p; \quad w'_2 = k_i; \quad w'_3 = k_d \] (4)

When the PIDNN selection initial value with type (3), and the formula (4) is connection weights from PIDNN hidden layer to output layer, the output of the PIDNN is equivalent to PID controller output. So according to its special relationship with the PID controller and the analysis of the literature [9], the input layer to hidden layer weights with PIDNN can be fixed that maintain the initial values. This can significantly reduce the amount of calculation and the number of cycle, and improve efficiency. Then, the middle layer to output layer weights (kp, ki, kd) are updated and trained by particle swarm.

SYNCHRONIZATION CONTROL SYSTEM AND ITS CONTROLLER DESIGN

Multi-motor Proportion Synchronization Control Strategy

In no axis synchronous motor control system, the running between the drive shafts need to maintain a certain speed. Synchronized proportional relationship commonly is \( w_1y_1 = w_2y_2 = \ldots = w_ny_n \). When the motor synchronous ratio wi is 1, it is the most simple synchronous relationship. But in actual production, running between the motor need according to the different proportion, as:

\[ w_1 : w_2 : \ldots : w_n = \mu_1 : \mu_2 : \ldots : \mu_n \]

Current motor synchronization control strategy have parallel, master-slave control, virtual control total shaft coupling control, cross coupling control and deviation control and so on[10-13].

In parallel synchronous control, each shaft has a good synchronization performance. The system that uses this control structure can extend the number of shaft and has a simple structure. In multi-axis synchronous control system, it is widely used. In this paper, the model uses parallel type synchronous control with 2 motors as actuators. The synchronous control method is shown in figure 2.
Permanent Magnet Synchronous Motor

In the motor shaft less synchronous control system, permanent magnet synchronous motor, with simple structure, reliable operation, high power factor, large torque at low speed, is widely used. But it is a nonlinear, strong coupling, multi-variable system, and needs through some changes to convert to dc motor features and modeling [14]. In engineering design, in order to simplify the analysis of the problem, we have to simplify and get simplified model for permanent magnet synchronous motor [15]

\[
G(s) = \frac{K_m}{JLs^2 + (JR + f_m)s + (Rf_m + K_eK_m)}
\] (5)

\[L, R, J\] are the armature winding inductance, resistance and current; \(K_e\) is counter electromotive force proportion coefficient; \(K_m\) is the moment coefficient of the motor; \(J\) is the moment of inertia of the motor shaft; \(f_m\) is the damping coefficient.

Using the two same motor servo system, and the parameters for: \(J=0.1982013\text{kg} \cdot \text{m}^3\); \(f_m=10^{-4}\text{N} \cdot \text{m/(rad} \cdot \text{s}^{-1})\); \(R=25\Omega\); \(L=0.3744\text{H}\); \(K_m=0.727\text{N} \cdot \text{m/A}\); \(K_e=0.2\text{V/(rad} \cdot \text{s}^{-1})\). We can get:

\[
G(s) = \frac{1}{0.10207231 \ast s^2 + 6.81578 \ast s + 0.2034}
\] (6)

We have the discretization process for formula(6) and use zero order retainer with sampling period \(t_s = 0.01\) s, having

\[
y(k) = 1.513y(k-1) - 0.5129y(k-2) + 0.0003968u(k-1) + 0.0003179u(k-2)
\] (7)

Design for Single Axis Synchronous Controller

THE PIDNN ALGORITHM BASED ON ADAPTIVE MUTATION PSO

Each axis adopts single axis controller to adjust the control the output of control and makes the system output can quickly follow system input values in proportion of multi-motor synchronization control model for figure 2. According to the mentioned algorithm in the second part, single axis synchronous controller is designed. As shown in figure 3, simplify PIDNN algorithm is worked as single axis synchronous controller, adaptive mutation PSO algorithm is used to update and train the PIDNN hidden layer to output layer weights(kp, ki, kd).
Figure 3. PIDNN controller based on adaptive mutation PSO algorithm.

PIDNN ALGORITHM PROCESS BASED ON ADAPTIVE MUTATION PSO

(1) The initialization of each particle search scope, speed, initial position, each group of particle number, group number, the learning efficiency, etc. Setting the PIDNN input layer to hidden layer weights.

(2) Give the step signal for the first iteration. Calculate the value function of each particle, and obtain the initial individual optimal location (pbesti) and the global optimal position (gbest).

(3) Start loop iteration and update the particle's position and speed, according to the formula (1), (2). Make a mutation for the particles that meet the condition.

(4) Forward calculate PIDNN with the updated particles and calculate the fitness of all particles. Then, updating the optimum position of each particle (pbesti) and the optimal position of species (gbest).

(5) To forward calculate the PIDNN with the optimal location of population as PIDN hidden layer to output layer weights, and calculate the system output.

(6) Return(3) and cycle calculation. When reach cycles, terminating calculation and drawing.

SIMULATION AND ANALYSIS

Making input r = 1, scale factor $\mu_1 = 1, \mu_2 = 0.6$; Using MTALAB simulation for parallel type proportional synchronization control system. Population size is 20, particle dimension is 3, input signal sampling points is 200, sampling interval is 0.01s. Particles search range is [-30, 30], maximum speed is [3, 3]. The simulation waveform in motor synchronization control system is shown in figure 4.

The simulation results can be seen in figure 4, the simplified adaptive mutation PSO - PIDNN algorithm enables the system axis 1 and 2 output without overshoot.
phenomenon mainly, and can quickly reach the stable state, has good quickness and adaptability, meets the requirements of synchronous control system in industrial control. Moreover the PID parameters $k_p$, $k_i$, $k_d$ selection does not need to according to the experience of artificial selection and engineering subjects, and has a certain universality.

CONCLUSIONS

For proportion of synchronous control system based on multiple motor, the synchronous controller is designed, and adaptive mutation PSO-PIDNN synchronous control algorithm was presented.

1) The PIDNN weight was trained with adaptive mutation PSO algorithm, and it can solve the shortcomings that learning process is easy to fall into local optimal value and learning rate is slow in traditional learning process for weights.

2) The traditional PSO algorithm is adaptive mutated. It can avoid the disadvantages for the traditional PSO search precision is low and late iterative efficiency is not high, and improve the efficiency of the controller.

3) In the process of PIDNN training, network input layer and output layer weights are set for fixed values according to the actual system. It greatly reduces the computational complexity and improves the practicability of synchronous controller.

Optimization of adaptive mutation PSO - PIDNN controller has good adaptability and quickness, no overshoot by simulation analysis. In addition, it solves the disadvantages of the traditional PID control parameters selection difficultly, and can effectively realize the motor synchronization control performance, can meet the engineering requirements about synchronized control system.
ACKNOWLEDGEMENTS

This work has been supported by the Production Research funding of joint innovation of Jiangsu Province of China (Grant No. BY2014004-08).

REFERENCES

5. Xingang-miao, Su-Wang, Lingpang-Han, etc. Based on the deviation coupling of single neuron synchronous motor control [J], Micro-motor, 2011, 2, 44-47. (In Chinese)