51st IEEE Conference on Decision and Control
December 10-13, 2012. Maui, Hawaii, USA

An Improved Predictive Optimal Controller with Elastic Search Space for Steam Temperature Control of Large-Scale Supercritical Power Unit

Liangyu Ma, *Member, IEEE*, Kwang Y. Lee, *Life Fellow, IEEE,* and Yinping Ge

Abstract— **Predictive optimal control (POC) combined with artificial neural networks (ANNs) modeling and advanced heuristic optimization is a powerful technique for intelligent control. But actual implementation of the POC in complex industrial processes is limited by its known drawbacks, including the oscillation resulting from random search direction, difficulty in meeting the real-time requirement, and unresolved adaptability and generalization ability of the ANN predictive model. In resolving these problems, an improved Intelligent Predictive Optimal Controller (IPOC) with elastic search space is proposed in this paper. A new simpler and high-efficiency Particle Swarm Optimization (PSO) algorithm is adopted to find the optimal solution in fewer epochs to meet the real-time control requirements. The system output error in each control step is fed back to adjust the search space dynamically to prevent control oscillation and also make it easier to find the optimal solution. An improved recurrent neural network with external delayed inputs and outputs is constructed to model the dynamic response of the highly nonlinear system. The proposed IPOC is used to superheater steam temperature control of a 600MW supercritical power unit. Extensive control simulation tests are made to verify the validity of the new control scheme in a full-scope simulator.**

I. Introduction

Superheater Steam Temperature (SST) is one of the key variables closely related to the safety and efficiency of a large-scale coal-fired power generating unit, which must be tightly controlled within design limits during boiler operation. To meet the temperature control requirements, multi-stage water-spray de-superheaters are usually used to control the SST, and several groups of cascaded PID controllers are adopted for temperature control [1]. Since SST has strong nonlinear characteristics under different load levels, several different groups of PID parameters should be found for these controllers to achieve good control performance over a wide range of loading conditions. This often costs much time and effort and is not easy to realize in actual operation. Therefore, it is a logical choice to take advantage of intelligent system techniques in improving the superheater steam temperature control [2-9].

The work is supported by the National Natural Science Foundation of China (Grant No. 61174111) and the U.S. National Science Foundation under grant ECCS 0801440.

L. Y. Ma and Y. P. Ge are with the Automation Department, School of Control and Computer Engineering, North China Electric Power University, Baoding, Hebei 071003, China (e-mail: maliangyu@ncepu.edu.cn, geyinping2010@qq.com).

K. Y. Lee is with the Department of Electrical and Computer Engineering, Baylor University, Waco, TX 76798-7356, USA (e-mail: Kwang_Y_Lee@baylor.edu).

Among different intelligent control strategies, Predictive Optimal Control (POC) is a powerful technique which combines the Artificial Neural Networks (ANNs) modeling capability and advanced heuristic optimization algorithms, such as Particle Swarm Optimization (PSO) or Genetic Algorithm (GA) [5-9]. But the practical implementation of POC in complex industrial processes is seriously curtailed by its apparent drawbacks [10-13]: (1) The optimal solution search with PSO (or GA) is often time-consuming and thus hard to meet the real-time control requirement; (2) The PSO is with certain randomness in its solution space and apt to oscillation; and (3) The POC requires high accuracy of the neural network predictive model since it is used to find the search direction by evaluating the optimization results. However, finding a satisfactory ANN predictive model, with high precision, strong adaptability to a wide-range of operating conditions and strong generalization ability, is still unresolved and needs further investigation.

Aiming in resolving the above-mentioned problems, this paper proposes an Intelligent Predictive Optimal Controller (IPOC) with dynamic elastic search space. A simpler high-efficiency PSO algorithm which cleverly discards the velocity concept is adopted to find the optimal solution with less time and fewer epochs to better meet the real-time control requirement [14]. The system output error in each step between the process variables and their set points are fed back to adjust the search space dynamically to prevent control oscillation and also make it easier to find the optimal solution. An improved recurrent network with external delayed inputs and outputs is constructed to model the dynamic response of the highly nonlinear system based on a wide range of operating data, resulting in faster convergence in fewer training epochs with high precision and good generalization ability to meet the wide-range of loading conditions [19-21].

As a case study, the IPOC is used to improve the SST control of a 600MW supercritical coal-fired power generating unit in a full-scope simulator. The simulator is a commercial-grade product developed by *Baoding Sinosimu Technology Co. Ltd*., a power plant simulator provider in China, who has developed more than 200 power plant simulators for different users, both home and abroad. The simulator has been tested and validated by power plant engineers with its high steady-state accuracy and authentic load-changing dynamic performance and has been put into operation for several years. In this work, the data used for ANN model training are generated by the simulator and the control tests are also performed in the same simulator. The proposed control scheme is implemented in MATLAB, which communicates in a bi-directional way with the simulator. Extensive control simulation tests are made to verify the effectiveness of the IPOC scheme.

II. IPOC WITH ELASTIC SEARCH SPACE

The proposed IPOC scheme with elastic search space is illustrated in Fig. 1.

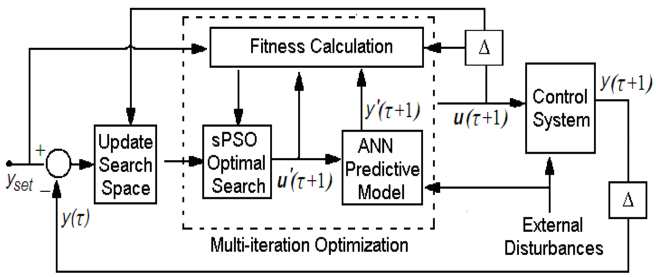


Figure 1. IPOC scheme with elastic search space.

As shown, for a given control system with a control input vector u and a system output vector y, the search space for u is dynamically updated in each step based on the real-time error feedback between the system output y and its setpoint y_{set}. The elastic search space $\Omega=[u_{\min}, u_{\max}]$ is defined by a hypercube in the input space, which can be updated component-wise by:

$$u_{i\max}(\tau+1) = u_i(\tau) + Z_i\left\|y_{set} - y(\tau)\right\| \quad (1)$$

$$u_{i\min}(\tau+1) = u_i(\tau) - Z_i\left\|y_{set} - y(\tau)\right\| \quad (2)$$

Where, u_i is the i-th component of the input vector and Z_i is the corresponding *expansion factor*, which should be predefined through tests. It is easy to see from (1) and (2) that the width of the search space is adjusted automatically according to the real-time control error between the system output y and its setpoint y_{set}. If the output error is small, the search space will be small; if the error is big, the search space will be enlarged. By introducing real-time error feedback to the elastic search space, the oscillation of POC with PSO search can be avoided and also the optimal solution can be found in shorter time.

After the search space is determined, a PSO algorithm will be applied to search for the optimal control demands. Thus a fitness function should be defined to evaluate the control performance and guide the search direction [10-14]. A good fitness function should consider both the system output error and the cost of the actuator moves. Different fitness function can be formed for different applications. For the system shown in Fig. 1, the fitness function is defined as

$$Fit = R_1\left\|y'(\tau+1) - y_{set}\right\| + R_2[\sum S_i\left|u_i'(\tau+1) - u_i(\tau)\right|] \quad (3)$$

where R_1 and R_2 are the weights on the predicted output error and that on the total cost of the actuator moves, respectively; S_i is the respective weight on the cost of the i-th actuator move u_i; $u_i'(\tau+1)$ is the temporary control output during PSO search at each iteration, and $y'(\tau+1)$ is the predicted system output with the neural network nonlinear system model.

III. SIMPLIFIED PARTICLE SWARM OPTIMIZATION (SPSO)

A. Basic Particle Swarm Optimization (bPSO) Algorithm

The Particle Swarm Optimization (PSO) algorithm is based on the simulation of birds flocking in two-dimensional space [10]. The position and velocity of each particle represents the position and velocity of a bird. The position and velocity vectors are represented by *X-Y* coordinates. Each particle (bird) saves its best position so far (called *pbest*) and its current position. Also, each particle knows the best *pbest* so far among the group (called *gbest*). Each particle modifies its position by changing its velocity. The velocity of each particle is updated by

$$v_i^{k+1} = wv_i^k + c_1 rand_1(pbest_i - u_i^k) + c_2 rand_2(gbest - u_i^k) \quad (4)$$

where, v_i^k is the velocity of particle i at the k-th iteration, c_1 and c_2 are weight factors, $rand_1$ and $rand_2$ are random numbers between 0 and 1, u_i^k is the current position of particle i at the k-th iteration, $pbest_i$ is the personal best position of particle i, and *gbest* is the best value so far in the group among the *pbest* of all particles, and w in (4) is a weight function, which is usually adjusted with the Inertia Weights Approach (IWA) by

$$w = w_{\max} - (w_{\max} - w_{\min}) \times iter / iter_{\max} \quad (5)$$

where $w_{\max}$ is the initial weight, $w_{\min}$ is the final minimum weight, $iter_{\max}$ is the maximum iteration number and $iter$ is the current iteration count.

The position at iteration k+1 is updated by

$$u_i^{k+1} = u_i^k + v_i^{k+1} \quad (6)$$

B. A Simplified High-Efficiency PSO Algorithm

The above basic particle swarm optimization (bPSO) has some disadvantages, such as relapsing into local extremum and slow convergence in velocity in later iterations, making the bPSO not favorable for real-time control of a complex industrial process. Therefore, many researchers have suggested different measures for improvement [10-13].

Recently, an effective simplified PSO is proposed to overcome the disadvantages of the bPSO [14]. The simplified PSO (sPSO) cleverly skips the velocity calculation and thus reduces the PSO algorithm from the second-order to the first-order difference equation. Thus, the evolutionary process of the sPSO is controlled only by the particle's position. The position of each particle is updated by

$$u_i^{k+1} = wu_i^k + c_1 rand_1(pbest_i - u_i^k) + c_2 rand_2(gbest - u_i^k) \quad (7)$$

The tests with some typical benchmark functions have shown that the sPSO can greatly improve the convergence speed and precision in the evolutionary optimization [14]. Thus, this sPSO is favorable for real-time control and it will be used to search for the best controls in the proposed IPOC scheme.

IV. Improved Recurrent Neural Network Model

A. Basic Elman Recurrent Neural Network

Based on the direction of information flow, artificial neural networks can be grouped into feedforward networks and recurrent networks. A recurrent neural network differs from the conventional feedforward networks (such as BP or RBF neural networks) in that, it includes recurrent or feedback connections [15-23]. The delay in these connections store values in the previous time-step and use them as inputs in the current step, which makes the network sensitive to the history of input and output data and suitable for dynamic system modeling.

A popular recurrent neural network is the Elman network [17-20]. The basic structure of an Elman network with M inputs and N outputs is shown in Fig. 2, where the recurrent neurons are in the context layer. A special case of the Elman network is the Diagonal Recurrent Neural Network (DRNN), where the context layer is collapsed to the hidden layer, thus eliminating the cross talks and reducing the number of weights between the context layer and the hidden layer [18].

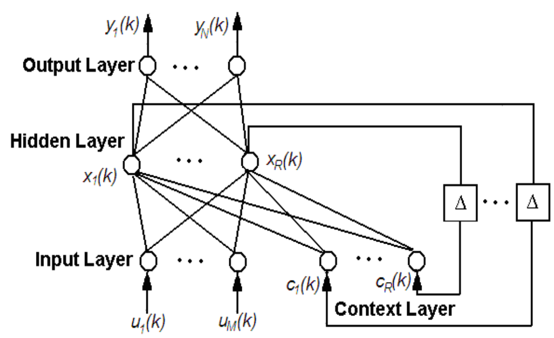


Figure 2. Basic Elman neural network model.

As shown in Fig. 2, the outputs in each layer of an Elman network are given by:

$$x_j(k) = f(\sum_{i=1}^{M} W_{i,j}^1 u_i(k) + \sum_{i=1}^{R} W_{i,j}^3 c_i(k)) \tag{8}$$

$$c_i(k) = x_i(k-1) \tag{9}$$

$$y_j(k) = g(\sum_{i=1}^{R} W_{i,j}^2 x_i(k)) \tag{10}$$

where, $W_{i,j}^1$ is the weight that connects node i in the input layer to node j in the hidden layer; $W_{i,j}^2$ is the weight that connects node i in the hidden layer to node j in the output layer; $W_{i,j}^3$ is the weight that connects node i in the context layer to node j in the hidden layer; and $f(\cdot)$ and $g(\cdot)$ are the transfer functions of the hidden layer and the output layer neurons, respectively, where $f(\cdot)$ mostly takes *logsig* or *tansig* function and $g(\cdot)$ often takes *purelin* function [24].

B. Improved Neural Network Structure

Compared with a feedforward network, an Elman neural network is better suited for nonlinear dynamic system modeling. But due to the existence of the recursive layer a basic Elman network is sensitive to its data sampling time and its real-time prediction accuracy is not very good.

After a series of experiments, the first-order time-delayed values of the input and output variables are added to the input layer of the basic Elman neural network. Thus the input layer of the modified network includes N+2M neurons, as shown in Fig. 3. This modified NN model structure is used for the model development of nonlinear plants in this work.

Compared to the basic Elman network, this modified recurrent network can converge with higher precision in less time and fewer training epochs. It can better adapt to the changes in the sampling time and provide real-time predictions with higher precision.

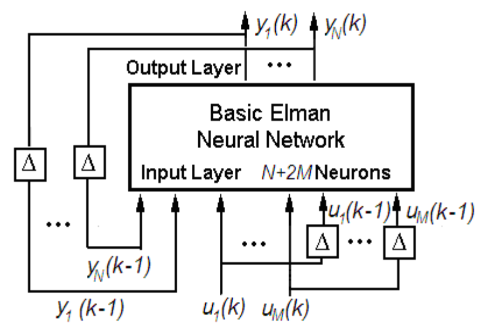


Figure 3. Modified neural network structure.

V. Case Study: Boiler Superheater Steam Temperature Control

A. Boiler Superheater System

The boiler unit investigated is a 600MW supercritical boiler, type DG-1900/25.4-II, manufactured by Dongfang Boiler Co. Ltd., China [4]. The steam flow of the boiler is shown in Fig. 4.

The superheater system of the boiler is composed of four subsystems along the steam flow direction: 1) the roof tubes, walls and midfeather of the vertical flue path in the rear furnace; 2) the primary Low-temperature Superheater (LSH), installed in the rear path; 3) the Platen Superheater (PSH) at the top of the furnace; 4) the Final Superheater (FSH), located above the furnace arch. The whole superheater system has a left-to-right cross before the final superheater to reduce the temperature deviation effect across the width induced by uneven flue gas.

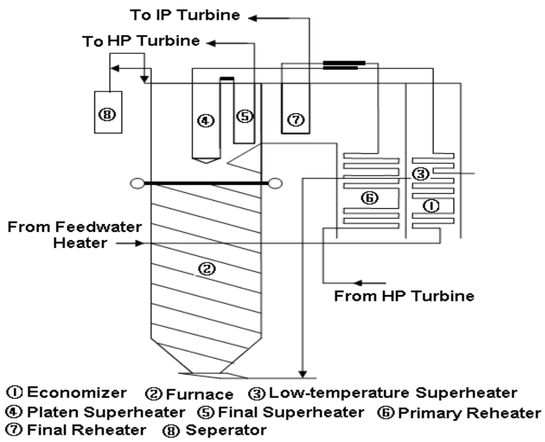


Figure 4. Steam flow chart of a 600MW supercritical boiler.

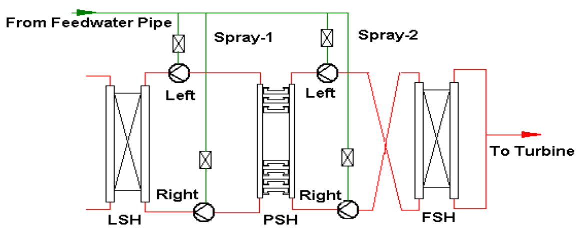


Figure 5. De-superheater system layout.

The SST of the supercritical boiler is controlled by both fuel/feedwater ratio and the two-stage water-spray de-superheating valves. The first-stage de-superheater is introduced to the connecting pipe between the LSH export and the PSH import in order to control the PSH outlet steam temperature. The second-stage de-superheater is introduced between the PSH export and the FSH import to control the final superheater steam temperature. The layout of the de-superheating system is shown in Fig. 5, where each de-superheater (spray) has two control valves, left- and right-sides.

According to the boiler operation manual, the final superheater (FSH) outlet steam temperature should be maintained at its rated value (571°C) for 35% to 100% loading range with allowed deviation of ±5°C. In order to meet the above control requirement, cascaded control scheme is used both for the 1st- and the 2nd-stage de-superheaters, each with two PID controllers. Thus there are totally 4 PID controllers for the superheater system. Two independent temperature control objectives exist, i.e., the PSH outlet temperature control and the FSH outlet temperature control. The whole control system is rather complex and tuning of the parameters of the 4 PID controllers is not easy in ensuring good control quality for the wide-range loading condition.

B. Design of Intelligent Predictive Optimal Controller

Based on the control scheme in Fig. 1, an IPOC is designed for SST control, which is shown in Fig. 6. Different from the original control scheme of the simulator, the new scheme treats the two superheaters (PSH and FSH) as one superheater system, using only one IPOC to control both the 1st- and the 2nd-stage water-spray valves concurrently. The FSH outlet steam temperature is the dominant control objective. The cost of the two actuator movement is considered in the fitness function together with the FSH steam temperature error, as shown in (3). The search space of input is dynamically updated according to (1) and (2). The improved Elman neural network is used for the superheater system nonlinear model development. The sPSO is used for searching the optimal controls and the results are evaluated iteratively by the fitness function (3).

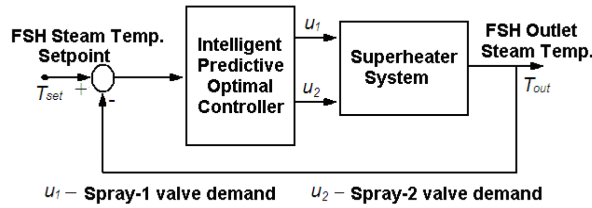


Figure 6. IPOC structure for SST control.

C. Development of NN Prediction Model for Superheater

1) Selection of Input and Output Variables

For control purpose, the input and output variables for the superheater neural network model can be determined by isolating the system from the rest of the boiler unit, and analyzing carefully the most important exogenous variables of the SST. It can be seen that many variables have influence on the SST of a supercritical boiler unit, such as coal flow, air flow, feedwater flow, the 1st- and the 2nd-stage water-spray control valve openings, etc. Moreover, feedwater pressure, temperature and main steam pressure are closely related to the de-superheating water flow and temperature for the 1st- and the 2nd-stage de-superheating valves. All these important variables are included in the superheater system model and the input and output variables of the model are listed in Table 1.

TABLE 1. INPUT/OUTPUT VARIABLES OF THE SUPERHEATER MODEL.

Input Variables (9)	(1) Coal flow (Kg/h)
	(2) Air flow (Km^3/h)
	(3) Feedwater flow to waterwall (Kg/h)
	(4) Feedwater pressure (MPa)
	(5) Feedwater temperature (°C)
	(6) Main steam pressure (MPa)
	(7) LSH out steam temperature. (left) (°C)
	(8) Spray-1 valve opening demand (left) (%)
	(9) Spray-2 valve opening demand (left) (%)
Output Variables (2)	(1) PSH out steam temperature (left) (°C)
	(2) FSH out steam temperature (right) (°C)

2) Training Data Preparation

In order for the model to fully reflect the static and dynamic features of the system, the training data should be as extensive as possible, covering different loading conditions and dynamic transient processes. In our work, 25,314 sets of data are collected from the simulator with sampling period of 1s, including steady-state data for 600MW, 540MW, 480MW and 420MW load levels, and the dynamic transient data between the above four load levels with the load ramping rate of 10MW/min. During data collection, the Coordinated Control System (CCS) of the simulator is all in auto mode, thus the whole unit is controlled by the original control system units.

3) Model Training and Validation

The superheater system model with the modified neural network (as shown in Fig. 3) includes 20 inputs and 2 outputs. It is trained using half of the original data above (taken at every other sampling time) with modified Levenberg-Marquardt algorithm [24]. The optimal number of hidden neurons is fixed to 16 by trial and error. The Mean Squared Error (MSE) of the network reaches 9.6444e-7 after only 9 epochs of training.

To validate the generalization ability of the trained model, data of a different loading condition with the sampling time of 2s are used to test. The validation data is obtained by loading the plant from 600MW to 500, 420, 550, 600, 540, then to 480MW with the load ramping rate of 5MW/min. The NN results are shown in Fig. 7. It is demonstrated that the trained model has very good generalization ability for different loading conditions.

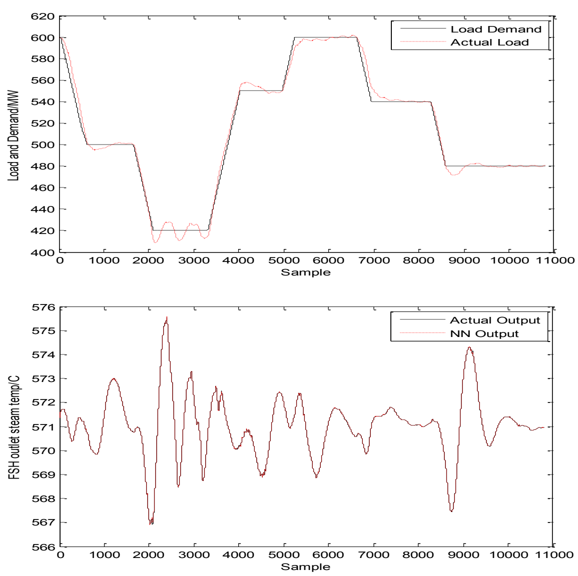


Figure 7. NN model validation test.

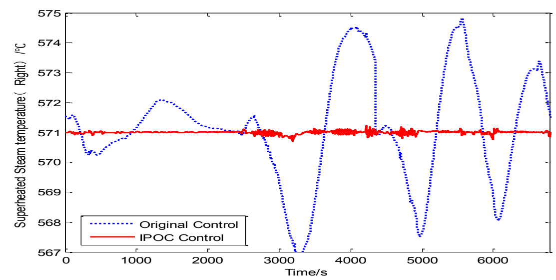
After the neural network model has been trained and validated, it is then used as a system online prediction model to evaluate the performance of the predictive controller, in which the sPSO algorithm is used to search for the optimal controls of the two-stage water-spray control valves.

D. Control Simulation Tests

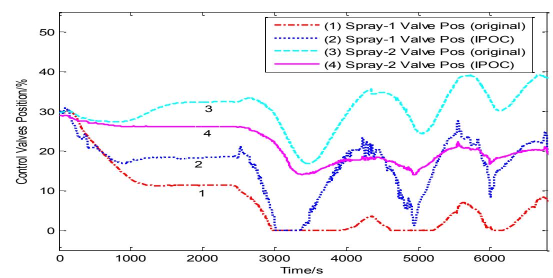
Based on the proposed control scheme, the control program developed in MATLAB is executed real-time by communicating with the full-scope simulator of the 600MW supercritical power generating unit. Then detailed control simulation tests are made.

During following tests, the sPSO parameters in (7) are set as: $c_1=c_2=1.7$, $w_{\max}=0.9$, $w_{\min}=0.4$, $iter_{\max}=5$, and the population size takes 10. The weights in fitness function (3) are: $R_1=10$ and $R_2=0$. The expansion factor Z_i in (1) and (2) takes 5 by trial and error.

Firstly, the control tests are made for loading-down process from 600MW to 420MW with the load changing rate of 10MW/min. The IPOC control results are compared with those of the original cascaded PID controllers in Fig. 8.



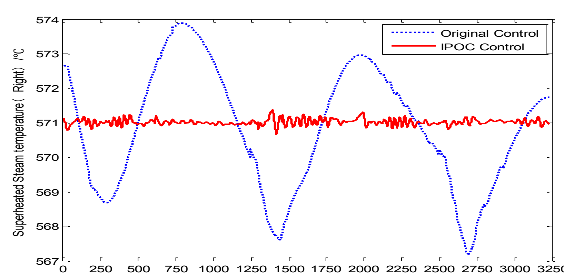
(a) Superheated steam temperature.



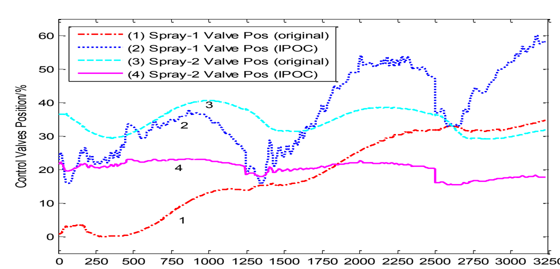
(b) Water-spray control valves action

Figure 8. Control tests during loading down process.

Similarly, control results are made for loading-up process from 420MW to 600MW with a different loading rate of 20MW/min. The IPOC control results are compared with those of the cascaded PID control scheme in Fig. 9.



(a) Superheated steam temperature.



(b) Water-spray control valves action

Figure 9. Control tests during loading up process.

It can be seen from Figs. 8 and 9 that the SST changes between 567 and 575℃ during the whole process with the original cascaded PID controllers. The maximum deviation from the setpoint is about ±4℃. The overshoot is big, the stabilizing time is long and the control quality differs at different loading points. On the other hand, with the new IPOC, the deviation of the SST from its set point is always less than ±1℃ and the steam temperature stabilizes very fast. By analyzing the action of the two-stage control valves in Figs. 8(b) and 9(b), it is easy to see that the good control result of IPOC is achieved by providing faster 1st-stage control valve response and smoother 2nd-stage control valve response compared with the original cascade controllers. The action of the two-stage water-spray control valves with the IPOC is good and acceptable in engineering practice.

VI. CONCLUSIONS

An Intelligent Predictive Optimal Controller (IPOC) with elastic search space is proposed and applied to Superheater Steam Temperature (SST) control of a large-scale power generating unit. To overcome the known drawbacks of the POC scheme based on NN model and PSO algorithm, several important improvements are made in the work. A new simpler and high-efficiency PSO is adopted to find the optimal solution in fewer epochs to meet the real-time control needs. The system output error in each control step is fed back to adjust the elastic search space dynamically to prevent oscillation and make it easier to find the optimal solution. An improved recurrent neural network utilizing external delayed inputs and outputs is constructed to model the dynamic response of the control system.

Extensive control tests with a commercial-grade full-scope simulator for a 600MW supercritical power generating unit show that the IPOC can dramatically improve the SST control effect by giving faster and optimized controls for the 1st- and the 2nd-stage water-spray valves compared with the original cascade PID control scheme.

The work presents a prospect for good engineering application of the IPOC by overcoming the known drawbacks of the POC based on neural network modeling and PSO search. For future research, the on-line adaptation of the neural network identification model with DRNN, the optimal choice of the PSO parameters, the weights in the fitness function and the expansion factors for the dynamic elastic search space will be further investigated. Finally, the concept of the IPOC will be extended to control the entire power plant operation.

REFERENCES

[1] I. Benyo, *Cascade generalized predictive control-applications in power plant control*. Oulu University Press, Finland, 2006.

[2] J. Zhang, G. Hou, and J. Zhang, “Adaptive neuro-control system for superheated steam temperature of power plant over wide range operation," *Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06),* pp. 138-141, 2006.

[3] K. Y. Lee, L. Ma, C. J. Boo, W.-H. Jung, and S.-H. Kim, “Inverse dynamic neuro-controller for superheater steam temperature control of a large-scale ultra supercritical (USC) boiler unit,” *Proc. of the IFAC Symposium on Power Plants and Power Systems Control*, in Tampere, Finland, July 5-8, 2009.

[4] L. Ma and K. Y. Lee, “Neural network based superheater steam temperature control for a large-scale supercritical boiler unit,” *Proc. of the IEEE Power & Energy Society General Meeting*, in Detroit, USA, July 24-28, 2011.

[5] K. Y. Lee, J. S. Heo, J. A. Hoffman, S.-H. Kim, and W.-H. Jung, “Modified predictive optimal control using neural network-based combined model for large-scale power plants,” 2007 *IEEE PES General Meeting*, pp: 1-8, June 2007.

[6] K. Y. Lee, L. Ma, C. J. Boo, W.-H. Jung, and S.-H. Kim, “Intelligent modified predictive optimal control of reheater steam temperature in a large-scale boiler unit,” *Proc. of the IEEE Power Engineering Society General Meeting,* in Calgary, Canada, July 26-30, 2009.

[7] J. S. Heo and K. Y. Lee, “Multi-objective control of power plants using particle swarm optimization techniques,” *IEEE Trans. Energy Convers.,* Vol. 21, No. 2, pp.552-561, June 2006.

[8] J. S. Heo and Kwang Y. Lee, “A multi-agent system-based reference governor for multiobjective power plant operation,” *IEEE Trans. Energy Convers.,* Vol. 23, No. 4, pp. 1082-1092, December 2008.

[9] K. Y. Lee, J. H. Van Sickel, J. A. Hoffman, W.-H. Jung, and S.-H. Kim, “Controller design for a large-scale ultra-supercritical once-through boiler power plant,” *IEEE Trans. Energy Convers.,* Vol. 25, No. 4, pp. 1063-1070, December 2010.

[10] J. Kennedy and R. Eberhart, “Particle swarm optimization,” *Proc. 1995 IEEE Int. Conf. Neural Networks*, Vol. IV, pp. 1942-1948, 1995.

[11] J. B. Park, K. S. Lee, J. R. Shin, and K. Y. Lee. “A particle swarm optimization for economic dispatch with non-smooth cost functions,” *IEEE Trans. Power Syst.,* Vol. 20, No. 1, pp. 34-42, 2005.

[12] J. G. Vlachogiannis and K. Y. Lee, “A comparative study on particle swarm optimization for optimal steady-state performance of power systems,” *IEEE Trans. Power Syst.*, Vol. 21, No. 4, pp. 1718-1728, November 2006.

[13] J. B. Park, Y. W. Jeong, J. R. Shin, and K. Y. Lee, “An improved particle swarm optimization for nonconvex economic dispatch problems,” *IEEE Trans. Power Syst.*, Vol. 25, No. 1, pp. 156- 166, February 2010.

[14] W. Hu and Z. S. Li, “A simpler and more effective particle swarm optimization algorithm,” *Journal of Software*, Vol. 18, No.4, pp.861-868, April 2007 (*In Chinese*).

[15] M. A. Hussain, “Review of the applications of neural networks in chemical process control—simulation and online implementation,” *Artificial Intelligence in Engineering*, No.13, pp. 55-68, 1999.

[16] R. Gencay and T. Liu, “Nonlinear modeling and prediction with feedforward and recurrent networks,” *Physica D,* 108, pp. 119-134, 1997.

[17] J. Elman, “Finding structure in time,” *Cognitive Science*, Vol. 14, pp. 179-211, 1990.

[18] C.-C. Ku and K. Y. Lee, “Diagonal recurrent neural network for dynamic systems control,” *IEEE Trans. Neural Netw.*, vol. 6, no.1, pp. 144-156, 1995.

[19] X. Z. Gao, X. M. Gao, and S. J. Ovaska, “A modified Elman neural network model with application to dynamic systems identification,” *In Proc. IEEE International Conference on Systems, Man and Cybernetics*, Bejing, China, pp.1376-1381. Oct. 1996.

[20] Y. C. Cheng, W. M. Qi, and W. Y. Cai, “Dynamic properties of Elman and modified Elman neural networks,” *Proc. of the 1st International Conference on Machine Learning and Cybernetics*, Beijing, pp. 637-640, Oct. 2002.

[21] C.-C. Ku, K. Y. Lee, and R. M. Edwards, “Improved nuclear reactor temperature control using diagonal recurrent neural networks,” *IEEE Trans. Nuclear Sci.*, vol. 39, no.6, pp. 2298-2308, December, 1992.

[22] R. K. AI Seyab, and Y. Cao. “Differential recurrent neural network based predictive control,” *Computers and Chemical Engineering*, Vol. 32, pp. 1533-1545, 2008.

[23] R. K. AI Seyab, and Y. Cao. “Nonlinear system identification for predictive control using continuous time recurrent neural networks and automatic differentiation,” *Journal of Process Control*, Vol. 18, pp.568-581, 2008.

[24] H. Demuth, M. Beale, and M. Hagan, *Neural Network Toolbox™ 6 User’s Guide*, The Math Works, Inc. 2008.