Controller Design for a Large-Scale Ultrasupercritical Once-Through Boiler Power Plant

Kwang Y. Lee, *Fellow, IEEE*, Joel H. Van Sickel, *Student Member, IEEE*, Jason A. Hoffman, Won-Hee Jung, and Sung-Ho Kim

***Abstract*—A large-scale once-through-type ultrasupercritical boiler power plant is investigated for the development of an analyzable model for use in developing an intelligent control system. Using data from the power plant, a model is realized using dynamically recurrent neural networks (NN). This requires the partitioning of multiple subsystems, which are each represented by an individual NN that when combined form the whole plant model. Modified predictive optimal control was used to drive the plant to desired states; however, due to the computational intensity of this approach, it could not be executed quickly enough to satisfy project requirements. As an alternative, a reference governor was implemented along with a PID feedback control system that utilizes intelligent gain tuning, which, while more complicated, satisfied the computational speed required for the controller to be realized.**

***Index Terms*—Gain tuning, intelligent control, modified predictive optimal control (MPOC), ultrasupercritical (USC) power plant.**

I. Introduction

ULTRASUPERCRITICAL (USC) boiler power plants are currently being developed to increase the efficiency of standard fossil fuel power plants. In this paper, the modeling and control of a large-scale once-through-type USC boiler power plant is investigated. Larger more complicated power plants require more sophisticated methods to streamline the modeling process as well as more sophisticated control schemes that can be used to further enhance plant efficiency.

The development of large capacity power plants requires new approaches to analyze plant dynamics for control purposes. In practice, many utility companies utilize simulation programs, such as modular modeling systems [1] or their own simulation tools for modeling. However, it is a challenge to extend current models to model larger capacity plants, and to design new models without component specifications. To design a control system for a power plant, a model must be developed in advance. Recently, the study of neural networks (NN) [2] has become important in designing system identification and control systems in the power systems area [3], [4]. With system data, the NN can be trained to approximate highly nonlinear functions. Since the NN strongly depends on the input/output data, but not on the physical structure of the system, it is flexible and can easily be adapted to different types of power plants.

Accurately modeling such a system with a single NN is theoretically possible, but it was discovered that the training of such a network was not practical. Instead, individual subsystems of the power plant were modeled with separate NNs that were combined to form the power plant model. This type of approach is covered in detail in [5]. Only the higher level details pertinent to this specific application will be covered in this paper.

It was desired to use a modified predictive optimal control (MPOC) scheme [6] with this plant to track unit load demand in order to provide adaptive control that optimized performance of the power plant. This scheme was developed successfully, but turned out to be more computationally intensive than desired for an actual controller. To overcome this difficulty, a reference governor [7] was developed to provide feedforward (ff) controls in conjunction with a simple PID feedback control system that utilizes intelligent gain tuning [8]. Both approaches are presented with a focus on the reference governor and intelligent gain tuning.

II. Large-Scale USC Power Plant

In this paper, the USC boiler power plant consists of four processes, which are air/flue gas, pulverizer, water/steam, and turbine/generator [9]. However, for modeling purposes, the number of detailed subsystems will be 19. Fig. 1shows the USC boiler power plant. Most blocks are subsystems, which will be represented by a NN-based subsystem model. The proposed scheme will be applicable to other types of plants, including nuclear and fuel cell plants.

The power plant under investigation is a coal-pulverized, once-through-type, boiler-turbine-generator unit. There are three economizers used to raise the temperature of water entering the boiler from the feedwater system. Two forced draft fans and two primary air fans provide air to the air preheater. The air preheater in turn provides heated air to the pulverizers, burners, and furnace. The primary air fans also provide cold air to the pulverizers. The fuel is provided to the furnace through the pulverizers and burners. Furnace pressure is maintained at the desired value by controlling two induced draft fans. The waterwall surrounds the furnace vertically and spirally. Flue

Manuscript received May 26, 2009; accepted July 5, 2010. Date of publication October 4, 2010; date of current version November 19, 2010. This work was supported in part by National Science Foundation under Grant ECCS 0801440 and Doosan Heavy Industries and Construction Company Ltd. Paper no. TEC-00180-2009.

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

J. H. Van Sickel is with the Department of Electrical Engineering, Pennsylvania State University, University Park, PA 16802 USA (e-mail: jhv107@psu.edu).

J. A. Hoffman is with Pennsylvania Power and Light Company, Allentown, PA 18101 USA (e-mail: jason.a.hoffman@gmail.com).

W. H. Jung and S. H. Kim are with the Corporate R&D Institute, Doosan Heavy Industries and Construction Company, Taejon 305-811, Korea (e-mail: wonhee.jung@doosan.com, sungho.kim@doosan.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEC.2010.2060488

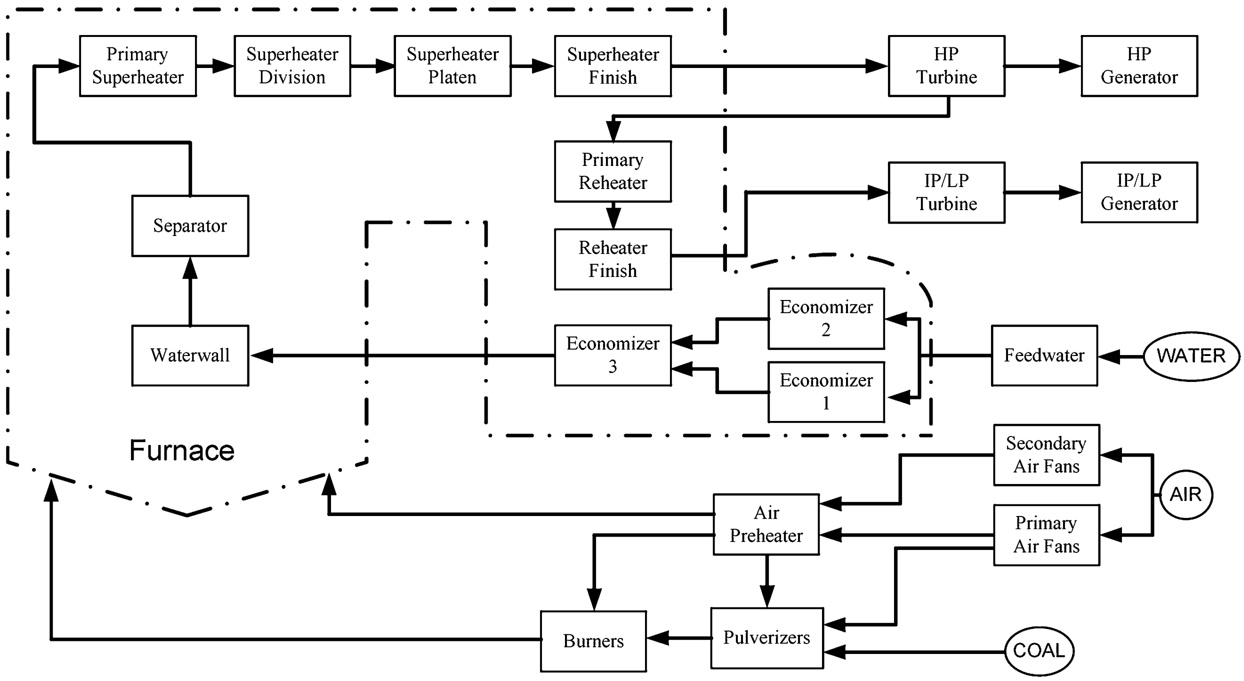

Fig. 1. 1000 MW USC boiler power plant.

gas exiting the furnace travels through the superheaters and reheaters, economizers, and air preheater to raise the temperature of the steam, water, or air, respectively. There is a separator on top of the furnace, which supplies high-pressure (HP) steam to the primary superheater and reduces the impurities in the steam. The superheater consists of four parts: primary, division, platen, and finish. The reheaters reheat the steam after the HP turbine using the primary reheater and the reheater finish. Finally, the turbine generates power from the tandem compound triple turbines, which consist of three parts: a HP turbine, an intermediate-pressure (IP) turbine, and low-pressure (LP) turbine.

The model will be focused on boiler, turbine, and generator parts. Each subsystem has common inputs and outputs: mass flow rate, temperature, pressure, and enthalpy of fluid. In addition to these inputs, there are control variables involved in driving each subsystem to the desired state, which are listed in Table I. To enhance tractability, the subsystems were catergorized into one of four process models: water and steam, air and flue gas, pulverizer, and the turbine and generator. The four process models, which are broken up into subsystems are shown in Table II. With the proposed approach, the utility company is able to investigate the dynamic characteristics of power plants with different capacities.

TABLE I
CONTROL ACTIONS

Control Number	Control Description	Associated Subsystem
u_{c1}	primary air fan	primary air
u_{c2}	forced draft fan	secondary air
u_{c3}	induced draft fan	gas recirculation
u_{c4}	hot primary air damper	pulverizer/burner
u_{c5}	cold primary air damper	pulverizer/burner
u_{c6}	coal feeder	pulverizer/burner
u_{c7}	feedwater pump	feedwater
u_{c8}	superheater division spray	feedwater
u_{c9}	superheater platen spray	feedwater
u_{c10}	high pressure turbine valve	high pressure turbine
u_{c11}	superheater damper	high pressure turbine
u_{c12}	reheater damper	high pressure turbine

III. NN Combined Model

An NN representing each subsystem is trained many times with different number of hidden neurons. The performance of the training, which is measured in the mean squared error (MSE) between the NN output and the target values, is compared with the others for different number of neurons. The number of hidden neurons with the smallest MSE is set as that subsystem's hidden neuron number. The optimal number of neurons depends on the number of inputs and outputs of each subsystem, as well as the input/output data pattern; therefore, some subsystems with few inputs and outputs require more hidden neurons to achieve the best performance. The resulting hidden neurons for each subsystem are shown in Table III. The gas recirculation system was split into two separate networks because there were six inputs and 32 outputs. Each NN of the two gas recirculation networks uses the six inputs to generate half of the outputs. Gas 1 delivers outputs to the division superheater, the platen superheater, the primary superheater, the final superheater, the primary reheater, and the final reheater. Gas 2 delivers outputs to the primary reheater, the economizers 1, 2, and 3, and the air preheater. An NN with six inputs and t32 outputs will cause the

TABLE II
PROCESS MODELS AND SUBSYSTEMS

Water and Steam Model	Air and Flue Gas Model	Pluverizer Model	Turbine and Generator Model
Feedwater Economizer1 Economizer2 Economizer3 Separator Primary Superheater Superheater Division Superheater Platen Superheater Finish Primary Reheater Reheater Finish Waterwall/ Furnace	Primary Air Secondary Air Air Preheater Gas Recirculation	Pulverizer/ Burner	Intermediate/ Low Pressure Turbine High Pressure Turbine

TABLE III
NN PARAMETERS

Subsystems	Inputs	Outputs	Hidden Neurons
Pulverizers/Burners	11	3	19
Primary Air	2	4	17
Secondary Air	2	2	21
Separator	4	4	11
High Pressure Turbine	5	5	21
Intermediate Pressure Turbine	4	4	25
Platen Superheater	10	4	21
Primary Superheater	7	4	23
Primary Reheater	7	4	25
Air Preheater	7	9	17
Division Superheater	10	4	23
Economizer1	7	4	23
Economizer2	7	4	25
Economizer3	11	4	21
Feedwater	5	11	17
Final Reheater	7	4	9
Final Superheater	7	4	9
Furnace	10	7	17
Gas1	6	16	21
Gas2	6	16	15

computer to run out of memory when training. The final result is referred to as the NN combined model (NNCM).

IV. MODIFIED PREDICTIVE OPTIMAL CONTROL

MPOC has already been used successfully in [6], and was the method expected to be used to control this power plant. This particular instance of predictive optimal control uses recurrent NNs (RNNs) [10] to implement an online identifier that models plant behavior. Particle swarm optimization [11] is used in conjunction with this identifier to test the validity of the next control action to see if it moves the power plant to the desired states. This is different from standard predictive control [12], which

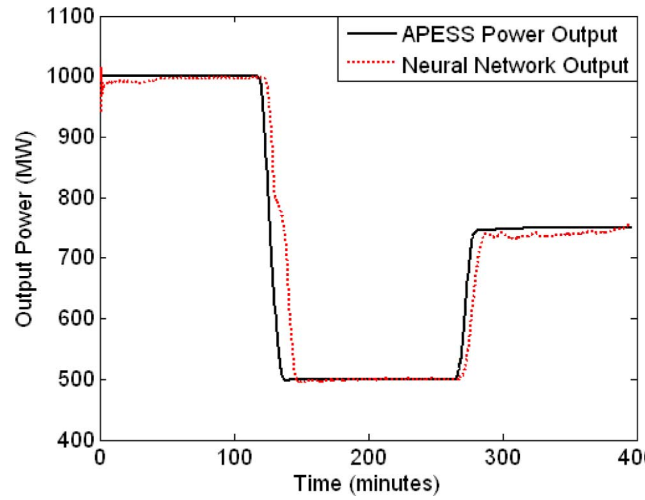

Fig. 2. MPOC (NN output) tracking power demand (APESS power output).

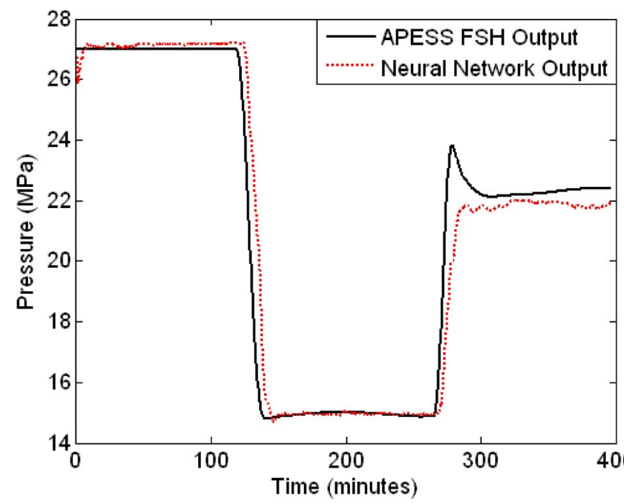

Fig. 3. MPOC (NN output) tracking pressure demand (APESS power output).

evaluates further than just the next time step. This was done to reduce overall calculation time. Unfortunately, the proposed method still did not achieve quick enough results to be used in real time for this application. Figs. 2 and 3 show the designed control system successfully tracking the desired power demand generated by the power plant simulator advanced power and energy system simulator (APESS). An online identifier [13] is updated so that it can accurately model current plant behavior and can be used by the MPOC to search for the next control action. Fig. 4details the operation of the online identifier, which was carried over for use in the second control approach.

The trouble with calculating control actions with MPOC is that the control signal was desired to be updated at least every 0.25 s. It was acceptable, while running in MATLAB, for the algorithm to generate an update every second, but the final speed was closer to 1.5 s. MPOC lends itself to distributed computing, which would allow the controls to be generated easily in the required period of time, however, it would require significantly more sophisticated hardware that is unproven for this type of application. Additionally, reducing look ahead to one-time step

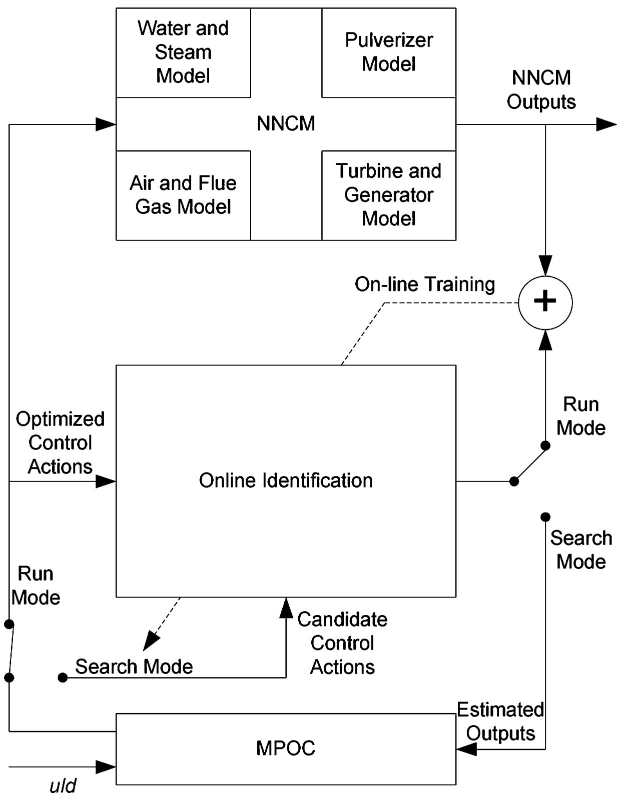

Fig. 4. Scheme for online identifier with MPOC.

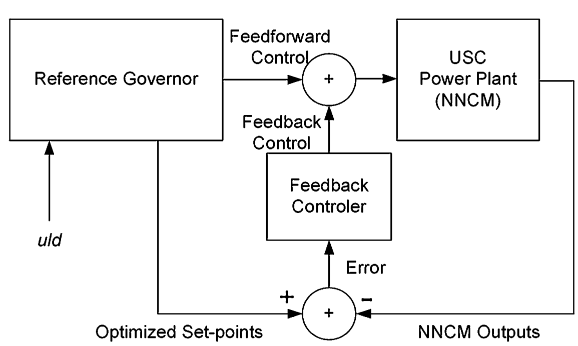

Fig. 5. Overall control scheme for reference governor and feedback control.

TABLE IV
REFERENCE GOVERNOR SETPOINTS

Set Points
Throttle Pressure Demand
Feedwater Demand
Coal Flow Demand
Final Superheater Temperature Demand
Final Reheater Temperature Demand
Furnace Gas Pressure Demand
Pulverizer Temperature Demand
Air Flow Demand
Power Demand

TABLE V
CONTROL ACTIONS AND COUPLED SETPOINTS

Controls	Associated Set Points
Primary Air Fan	Coal Flow Demand
Secondary Air Fan	Air Flow Demand
Feedwater Pump	Feedwater Demand
Spray 2	Final Superheater Temperature Demand
Spray 3	Throttle Pressure Demand
HP Turbine Valve	MW Demand
Induced Draft Fan	Furnace Gas Pressure Demand
Reheater Damper	Air Flow Demand
Superheater Damper	Final Reheater Temperature Demand
Hot Air Damper	Final Reheater Temperature Demand
Cold Air Damper	Pulverizer Temperature Demand
Coal Feeder	Pulverizer Temperature Demand

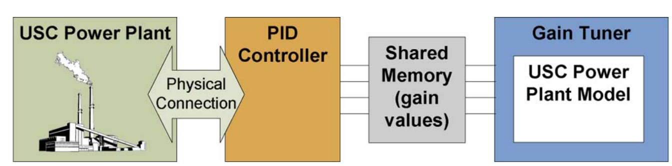

Fig. 6. Physical implementation of gain tuning.

does not guarantee stability. For these reasons, it was decided to investigate an alternative that met speed requirements and guaranteed stability.

V. REFERENCE GOVERNOR AND GAIN TUNING

Since the MPOC did not generate control actions quickly enough, an older method was modified to work with this process. Using a two-stage system, a reference governor can provide ff control actions as well as setpoints for a feedback controller, and the feedback controller provides the actual control actions to the plant, or in this case, the NNCM. This method is visualized in Fig. 5. For this to work, it was required to determine what setpoints would be used and which control actions would be coupled to these setpoints. The results are shown in Tables IV and V.

A. Calculation Time Issue

There are now two separate systems to deal with, the reference governor and the gain tuner, which both must satisfy the time requirements. Since the reference governor uses static NNs, its search times are significantly faster than MPOC, and ff controls and setpoints can be generated faster than 0.25 s using MATLAB, easily satisfying the speed requirements for the reference governor. The gain tuner must address a separate issue, as it does not generate the actual control signal, but tells the feedback control system what gains to use, as shown in Fig 6. This operation happens on a time scale significantly larger than 0.25 s and the only issue is whether the search for optimal gains converges when the gains need to be updated. All control signals are calculated by the PID controller, which could be implemented in standard hardware and meet any calculation speed requirements that are typical in industrial applications. The gain tuning search for this application converged for all tested time windows.

B. Reference Governor

Using a reference governor for providing ff control actions and setpoints has been shown many times, such as in [14]. As done in previous work, a steady-state model of the system was trained using a static NN, and then, a heuristic search method was used to find the ff control actions and corresponding setpoints that would optimize a cost function made of weighted objectives.

For this application, four of the five setpoints are actually held constant regardless of unit load demand, and can therefore be eliminated from the NN, as their values will never change. These setpoints are final superheater temperature demand, final reheater temperature demand, furnace gas pressure demand, and pulverizer temperature demand.

Interestingly, this approach worked very poorly at first. With the high order of this system, the search algorithm was able to find numerous candidate control actions and setpoints that equally satisfied the provided cost function. This was very undesirable as ideally, the cost function should be set up so that a single set of control actions and setpoints provide an optimal solution, or the reference governor will not know which set to choose. Using a scheme where different control actions have the same fitness is very noisy and inefficient. To cope with this problem, the concept of using nominal control actions was introduced. The nominal control actions are simply what the conventional steady-state control actions would be for a given unit load demand if a more sophisticated control scheme was not in place. The cost function was then modified so that it would optimize specific goals, and then, choose the candidate control actions that were closest to the nominal control actions. This modification served to fix the problem and provided good performance. The result was the following cost function:

$$f(u) = \alpha_1 \left|\mathrm{ULD} - \mathrm{PowerOut}\right| + \alpha_2 \left|\mathrm{CoalFlow}\right| + \alpha_3 \left|u - u_{\mathrm{nom}}\right| \qquad (1)$$

where the variables are as follows:

α_1, α_2, and α_3	multiobjective weights;
ULD	unit-load demand;
PowerOut	actual power output;
CoalFlow	control that determines how much coal is used;
u	ff control actions;
u_{nom}	nominal feed forward control actions in Fig. 7.

There is a disadvantage of using this approach because it assumes that the nominal control actions are available. In this case, these nominal control actions were available from earlier in the power plant's design process. If this is not the case, a simple control system would have to be developed to create these control actions, which may be more work than desired to use this particular approach to force convergence to a single solution. In addition to this, having such knowledge of the system allows the search space to be constrained around the nominal control actions, which further enhances the accuracy and timeliness of the results. For this application, the search algorithm was allowed to explore up to 0.1 above or below the normalized nominal control action, with limits of 1 and –1.

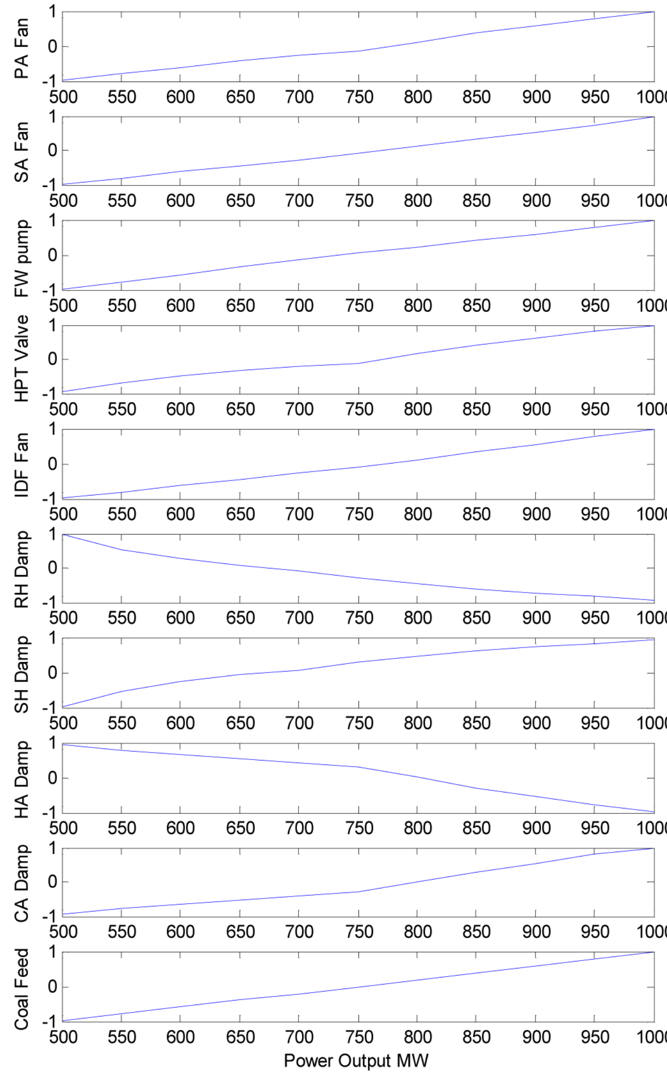

Fig. 7. Nominal control actions.

Another important factor to take into account with this approach is that some control actions may take on a wide range of value for any given unit load demand. This was the case with the spray controls, and better performance was achieved if they were not constrained to a nominal control actions. For this reason, they are not included in Fig. 7.

C. Gain Tuning

Intelligent gain tuning is done using an online identifier and a heuristic search to determine the gains of a PID control system [15]. The online identifier is similar to the one used for MPOC; only it is used by a gain tuner to search for gains instead of control actions. The heuristic search examines different gain values, and then, simulates the system with these gain values and the online identifier. It continues to experiment with different gain values until it finds the set of gains that reduce the error between the setpoints and the plant outputs.

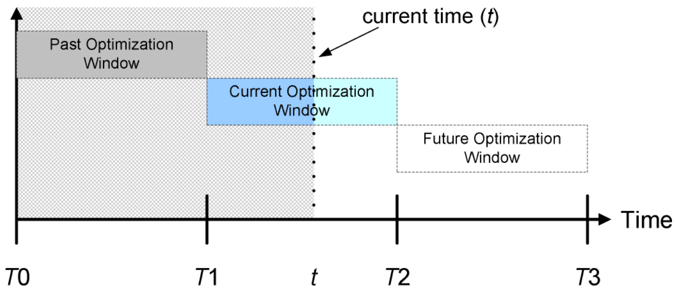

Fig. 8. Window operation.

It is very similar to MPOC except that instead of choosing the control values, it is choosing the gain values. This change is made because the gain values do not have to be updated for each time increment, while control values do. For gain tuning, a large window size can be chosen for which to tune the gains. This window could range from the size of a few minutes to multiple hours, depending on how often it is desired for the gains be tuned and how much trust is placed in the forecasted unit-load demand. The gain tuning takes exactly the time of the search window to search for the next set of gains. Once the window time has passed, the gain tuner reports the best set of gains it has found to the control system, which is then updated with these new gain values. Then, the gain tuner starts searching again for the best set of gains for the next window. This process is repeated indefinitely. It is important that the search has actually converged, or the results would be meaningless.

The window size was chosen to be 20 min. This is not the only window size that can be used, but it was the smallest window size that had smooth operation. Smaller window sizes can change the gains too often, which causes the system to become noisy and if the window size was small enough, could actually lead to unstable operation. Though, this is not the case for a window size of 20 min. The power plant is obviously running for longer than one window size, so its operation must be split into multiple windows. With a window size T, and total operational time of T_f, the operation is split into $N = T_f/T$ windows, with end at $T_0, T_1, \ldots, T_N$. This is shown in Fig. 8. The gains are updated at the end of an optimization window. This means the gains calculated between T_1 and T_2 are used for operation between T_2 and T_3.

The algorithm works by searching three different gain matrices, one for the proportional control, integral control, and derivative control. The range of possible gains is restricted to these values, which maintain stable operation of the power plant. Then, the algorithm takes the possible gain matrices and simulates the system for the next window size with these gain values using forecasted load data, as shown in Fig. 9. It repeats this simulation for different possible combinations of gains, and then, evaluates the gains by choosing which gain has the smallest total error for setpoint tracking, using the cost function

$$y = \sum_{n=1}^{9} \sum_{t=t_o}^{t_f} |\mathrm{setpoint}_n - \mathrm{output}_n| \qquad (2)$$

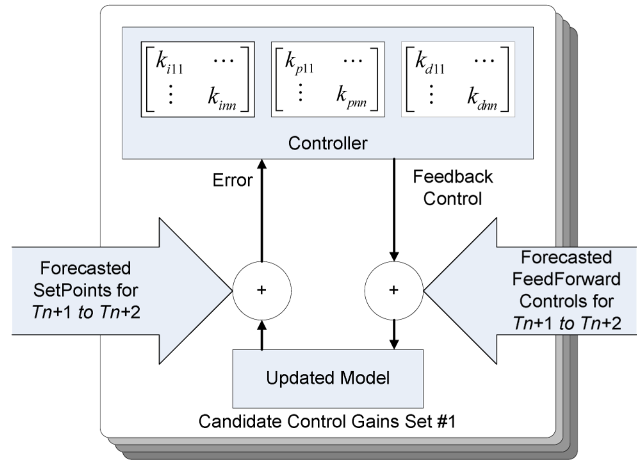

Fig. 9. Simulation process for evaluating candidate control gains.

where setpoint$_n$ is the nth desired setpoint, and output$_n$ is the nth actual plant output of this setpoint. The points t_o and t_f represent the beginning and end times of the current optimization window. This particular application has nine setpoints that need to be included.

Particle swarm optimization was used to implement this search and y from (2) is used to determine the fitness of each set of candidate gains. However, for each particle, to determine its fitness requires an entire simulation of the whole system, so obviously the search process is computationally intensive. Additionally, to be able to appropriately simulate the system, forecasted ff controls and setpoints need to be provided for the simulation. The feedback control system is receiving ff controls and setpoints continuously from the reference governor. It would be a waste of time to forecast the unit load demand for the power plant, and then, include the reference governor for each simulation of the system because the output from the reference governor will be the same every time. Instead, the reference governor only needs to be run once for the forecasted unit load demand and the outputs can be reused through the entire search process for a single window. All of this is shown in Fig. 10 with some of the particle swarm optimization details left out to keep the flowchart tractable.

One danger in this approach is the extra time taken by using a reference governor to forecast the optimized setpoints and ff controls. If the speed of the reference governor is not significantly faster than the gain tuner, this approach would most likely prevent successful convergence of the search for optimized gains. Due to the simple nature of this reference governor, it was not an issue, but problems are a definite possibility to take into account based on the specific application.

An online identifier, as with MPOC, is continually updated and used to provide an updated model for the simulation of the different gain values. It only needs to be updated once every window, so a large window size means the online identifier has to be updated less. In Fig. 11, results are shown for using the reference governor to vary the power plant from 1000 MW to 600 MW, and then to 800 MW. Only the setpoints, which actually change with unit load demand are shown.

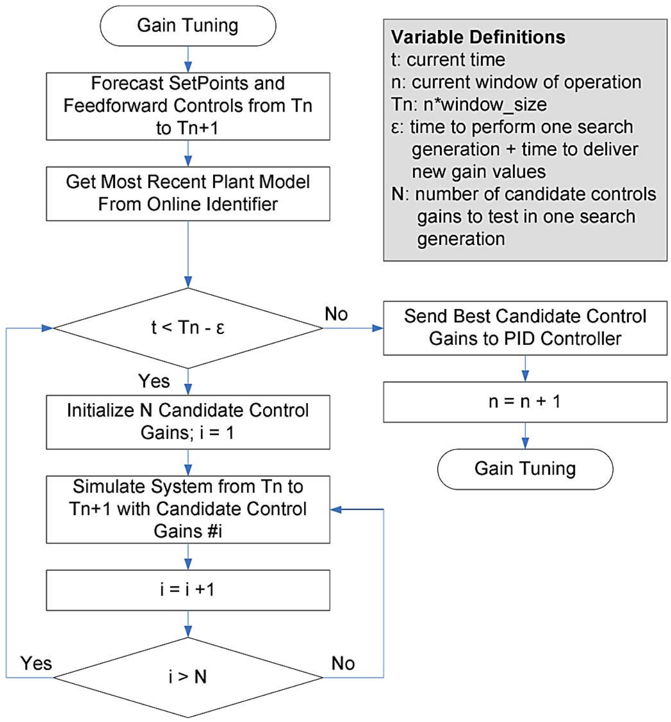

Fig. 10. Flowchart of gain tuning process.

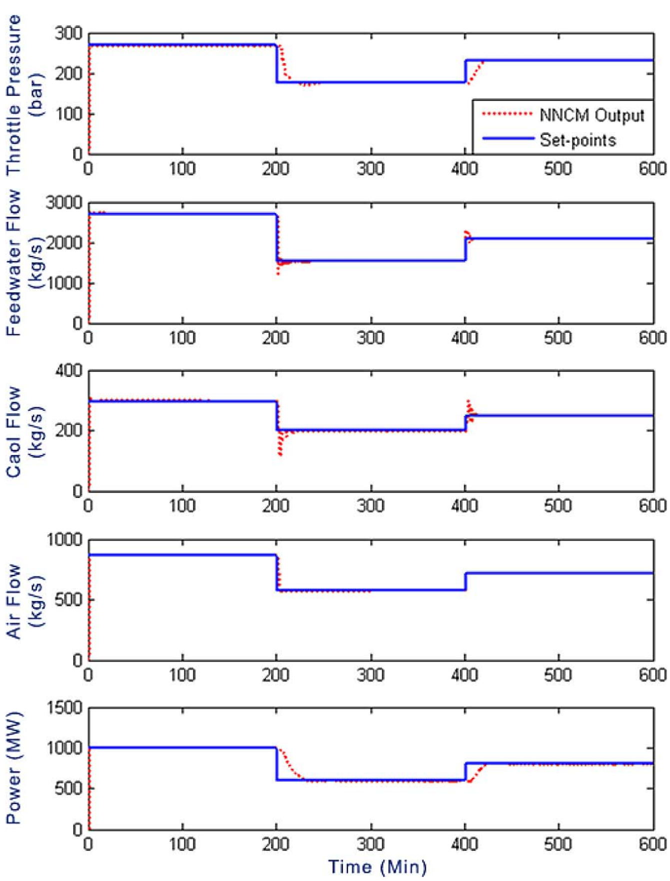

Fig. 11. Setpoint tracking performance of reference governor.

All setpoints were tracked well, with very little overshoot for tracking power, which is very nice as many times it was found that instead of using a unit step, providing a ramped signal helped the power plant respond better to changes in unit-load demand.

VI. Conclusion

Implementation of the reference governor with gain tuning had a few hitches, but overall was very effective for meeting desired performance goals. The MPOC, while simpler, only requiring two processing units instead of three, is not as computationally efficient. Additionally, stability of the gain tuning system can be analyzed with classical methods, making it a much more desirable approach for industry and mission critical applications. It is also an approach that is easier to implement in existing hardware, while MPOC would be more suited for new power plants that do not have classical control systems already in place.

Using the nominal controls in the cost function for the reference governor was sufficient to force convergence of the feedforward controls and setpoints, as well as provide an ideal way to constrict the search space. Unfortunately, the multiple objectives must still be aggregated into a single cost function, meaning that the operator would not be able to make changes online, as the effects would be unknown. It would be ideal to develop a method for generating cost functions that matched specific performance constraints. As of now, the cost function must be optimized by a human conducting a number of experiments to observe proper performance under different conditions.

This type of power plant is going to be used mainly for base line power and will not be seeing a lot of fast changes in the unit-load demand, which allows the gain tuning method to be used with forecasted values. This technique may not be quite as suitable for some of the faster power plants. It is possible that MPOC would be more suited for this type of operation, or that more sophisticated forecasting tools would need to be developed before desired performance could be achieved with the gain tuner.

References

[1] G. H. Leavesley, S. L. Markstrom, M. S. Brewer, and R. J. Viger, "The modular modeling system (MMS)—The physical process modeling component of a database centered decision support system for water and power management," *Water Air Soil Pollut.*, vol. 90, pp. 303–311, 1996.
[2] J. Lawrence, *Introduction to Neural Networks*. Nevada City, CA: California Scientific Software Press, 1994.
[3] C. C. Ku and K. Y. Lee, "Diagonal recurrent neural networks for dynamic system control," *IEEE Trans. Neural Networks*, vol. 6, no. 1, pp. 144–156, Jan. 1995.
[4] M. T. Haque and A. M. Kashtiban, "Application of neural networks in power systems; a review," in *Proc.–World Acad. Sci., Eng., Technol.*, June, 2005, vol. 6, pp. 53–57.
[5] K. Y. Lee, J. S. Heo, J. A. Hoffman, S. H. Kim, and W. H. Jung, "Neural network-based modeling for a large-scale power plant," in *Proc. IEEE PES General Meet.*, 2007, pp. 1–8.
[6] 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," in *Proc. IEEE PES General Meet.*, 2007, 1–8.
[7] R. Garduno-Ramirez and K. Y. Lee, "A multiobjective-optimal neuro-fuzzy extension to power plant co-ordinated control," *Trans. Institute Meas. Control.*, vol. 24, pp. 129–152, 2002.

[8] J. H. Van Sickel, K. Y. Lee, and J. S. Heo, "Differential evolution and its applications to power plant control," *Intell. Syst. Appl. Power Syst., 2007*, pp. 1–6.

[9] S. J. Al-Nasur, "*Simulation of a Fossil-Fuel Power Plant Unit in Matlab Environment,*" Masters dissertation, Dept. Electrical Engineering, Univ. Pennsylvania, State College, 2001.

[10] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Academy Sci. USA*, vol. 79, pp. 2554–2558, Apr. 1982.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, 1995, pp. 1942–1948.

[12] K. Y. Lee and M. A. El-Sharkawi (Eds.), *Modern Heuristic Optimization Techniques with Applications to Power Systems*. New York: Wiley, 2008.

[13] J. S. Heo and K. Y. Lee, "A multi-agent system-based intelligent identification system for power plant control and fault-diagnosis," in *Proc. IEEE PES General Meet.*, 2006, 1–6.

[14] J. S. Heo, K. Y. Lee, and R. Garduno-Ramirez, "Multiobjective control of power plants using particle swarm optimization techniques," *IEEE Trans. Energy Conver.*, vol. 21, no. 2, p. 552-561, Jun. 2006.

[15] B. Liptak, *Instrument Engineers' Handbook: Process Control.* Pennsylvania: Chilton Book Company, 1995, pp. 20–29.

Kwang Y. Lee (F'01) received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1964, the M.S. degree in electrical engineering from North Dakota State University, Fargo, ND, in 1968, and the Ph.D. degree in system science from Michigan State University, East Lansing, MI, in 1971.

He has been with Michigan State, Oregon State, University of Houston, and the Pennsylvania State University. He is currently a Professor and a Chair in the Department of Electrical and Computer Engineering, Baylor University, Waco, TX. His research interests include power system control, operation, planning, and intelligent system applications to power systems.

Dr. Lee is an Editor of IEEE TRANSACTIONS ON ENERGY CONVERSION. He was an Associate Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS. He is also a Registered Professional Engineer.

Joel H. Van Sickel (S'02) received the B.S. degree from Grove City College, Grove City, PA, in 2005, and the Ph.D. degree from the Pennsylvania State University, University Park, PA, in 2009, both in electrical engineering.

He is currently with the Department of Electrical Engineering, Pennsylvania State University. His research interests include distributed artificial intelligence, distributed simulation, intelligent control, power generation, multiagent systems, and optimization techniques.

Jason A. Hoffman received the B.S. degree from Bucknell University, Lewisburg, PA, in 2005, and the M.S. degree in electrical engineering from the Pennsylvania State University, University Park, PA, in 2007.

He is currently with Pennsylvania Power and Light Company, Allentown, PA. His research interests include modeling and control of power plants, neural networks, intelligent control systems, and alternative energy systems.

Won-Hee Jung received the B.S. and M.S. degrees in mechanical engineering from Pukyung National University, Busan, Korea, in 1995 and 1997, respectively.

Since 1997, he has been with Doosan Heavy Industries and Construction Company, Ltd., Taejon, Korea, where he is currently a Senior Engineer of Plant Control System Team in the Corporate R&D Institute. He has been involved in developing advanced control system for boiler unit. His research interests include control algorithm, optimization, simulation technique, and artificial intelligence for fossil power plant.

Sung-Ho Kim received the B.S. degree in mechanical engineering from Yeungnam University, Gyeongsan, Korea, in 1991, and the M.S. degree in mechanical engineering from Kyungpook National University, Daegu, Korea, in 1995.

Since 1995, he has been with Doosan Heavy Industries and Construction Company, Ltd., Taejon, Korea, where he is a Principal Engineer of Plant Control System Team in the Corporate R&D Institute. He has been involved in developing advanced control system for boiler unit. His research interests include control, operation, and modeling of fossil power plant.