PREDICTION OF BOILERS EMISSION USING POLYNOMIAL NETWORKS

Moustafa. Elshafei, Systems Engineering Department, King Fahd University of Petroleum and Minerals, Saudi Arabia, elshafei@ccse.kfupm.edu.sa

Mohamed A. Habib, Mechanical Engineering Department, King Fahd University of Petroleum and Minerals, Saudi Arabia, mhabib@kfupm.edu.sa

Mansour Al-Dajani, Systems Engineering Department, King Fahd University of Petroleum and Minerals, Saudi Arabia, dajani@ccse.kfupm.edu.sa

Abstract

In this paper we investigate the problem of NOx pollution using a model of furnace of an industrial boiler, and propose Functional Networks (FunNets) for high performance prediction of NOx as well as O2. The objective is to develop low cost inferential sensing techniques that would help in operating the boiler at the maximum possible efficiency while maintaining the NOx production within a specified limit. The studied boiler is 160 MW, gas fired with natural gas, water-tube boiler, having two vertically aligned burners. The boiler model is a 3D problem that involves turbulence, combustion, radiation in addition to NOx modeling. The 3D computational fluid dynamic model is developed using Fluent simulation package, where the volume of the furnace was divided into 371000 control volumes with more concentration of grids near solid walls and regions of high property gradients. The model provides calculations of the 3D temperature distribution as well as the rate of formation of the NOx pollutant, enabling a better understanding on how and where NOx are produced. The boiler was simulated under various operating conditions. The generated data is then used to train and test the developed neural network softsensors for emission prediction based on the conventional process variable measurements. The softsensors were constructed using Polynomial Networks (PolyNets), which are a special class of the recently introduced Functional Networks. PolyNets compose complex Neural Networks from simple transfer polynomials with weights that are computed efficiently by ordinary least-squares. The performance of the proposed PolyNet softsensor is evaluated in detail in the paper and compared with the traditional MLP neural networks. It is shown that PolyNets achieve better accuracy with simpler structures, and could be trained faster than MLP NN by a factor of 6-8 times.

***Keywords**: FunNets, NOx emission, boilers, NOx prediction.*

1. Introduction

The problems of increasing pollution and environmental degradation have become a major concern for the international community. A number of countries have already introduced new laws requiring a large spectrum of commercial and industrial facilities to reduce and report their emissions. These requirements, together with the economic and operational demands, derive the need for continuing improvement of power plant performance and control.

The predictions of NOx and O2 are essential for efficient operation of boilers while maintaining the NOx pollutant within a tolerable limit. Optimization of the operation of boilers can result in large savings. One of the areas in which the optimization can be performed is through the minimization of excess air. Lowering the excess oxygen (O2) from 1% to 0.5% will increase boiler efficiency by 0.25%, a saving of about one ton of fuel daily in a 160 MW boiler. In the mean time, lowering the excess air leads to higher flame temperature causing more NOx emission. The objective of this work is to investigate the problem of NOx pollution using a detailed simulation model of a furnace of an industrial boiler, and to develop low cost inferential sensing techniques using artificial neural networks (ANN) for emission monitoring of utility boilers and other combustion systems.

Application of ANN for emission monitoring has been recently proposed by a number of investigators. Michael et.al. [1] demonstrated successful application of ANN to predict emission of a 300 HP diesel engine. A multi-layer feedforward ANN of 10 input, 42 hidden neurons, and 3 outputs was developed to predict NOx, CO, and opacity. Emission monitoring using multivariable soft sensors was also proposed in [2,3, and 4]. Al-Duwaish el.al. [5], studied the use of ANN to predict O2 content in a boiler in a petrochemical plant.

The present study aims to investigate numerically the problem of NOx pollution using a model furnace of an industrial boiler, and propose an Artificial Neural Network (ANN) structure for high performance prediction of NOx as well as O2. The study uses a detailed simulation model of the combustion. The simulation data is then used to train polynomial learning networks for predicting emission from a utility boiler. It is shown that the proposed polynomial network outperforms the conventional MLP in speed of training, speed of operation, and accuracy.

In the next section we describe briefly the combustion 3D model, then in Section 3 the polynomial network is introduced. Next, in Section 4 we present and evaluate the PolyNet prediction results.

2. Boiler Combustion Model

NOx formation during the combustion process occurs mainly through the oxidation of nitrogen in the combustion air by two mechanisms known as thermal NOx and prompt NOx.

1. **Thermal NOx** - The concentration of thermal NOx is controlled by the nitrogen and oxygen molar concentrations and the temperature of combustion. Combustion at temperatures well below 1,300 C forms much smaller concentrations of thermal NOx.
2. **Prompt NOx** - Prompt NOx is formed from molecular nitrogen in the air combining with fuel in fuel-rich conditions which exist, to some extent, in all combustion. This nitrogen then oxidizes along with the fuel and becomes NOx during combustion, just like fuel.

The formation of NOx in industrial boilers is a very complicated problem due to many parameters that influence its formation process. These parameters include fuel to air ratio, inlet air temperature and combustion air swirl angle. The rate of thermal NOx formation is directly affected by the combustion zone temperature and the oxygen concentration.

The numerical calculation of the combustion process in industrial boilers is a 3D problem that involves turbulence, combustion, radiation in addition to NOx modeling. Complex 3D Combustion models for equipment design and operational changes are usually based on Computational Fluid Dynamics CFD [6]. CFD models are founded on fundamental physical principles. The set of governing differential equations together with the boundary conditions are to be solved numerically by an iterative, line-by-line procedure [7]. The details of the calculation procedure can be found in previous work such as Habib and Whietelaw [8], Shuja and Habib [9], and Habib et. al. [10]. The 3D model was built using the Fluent 6.1.22 CFD software package. The studied boiler is composed of a furnace (radiation section) and return tube bank (convection section). The boiler is used for production of superheated steam for process industry. The steam flow rate is 240 t/h. Steam pressure and steam temperature are 51 bar and 330°C. The combustion chamber has 12.54 m length in the direction of flame, 4.58 m width of front wall and 7.93 m (distance between drums). The volume of the furnace of the industrial boiler was divided to 371000 control volumes with more concentration of grids near solid walls and regions of high property gradients. The model estimates the 3D temperature distribution as well as the rate of formation of the NO pollutant, enabling a better understanding on how and where NOx are produced as shown in Fig. 1. The figure illustrates the temperature distribution (Fig.1a), and the NOx concentration distribution (Fig.1b) at a vertical plane passing through the two burners. The graphs demonstrate clearly the correlation between the temperature zones and the NO concentration zones.

The boiler simulation is used to generate the training and testing data under various operating conditions, which would have been very difficult, or very expensive to obtain while the boiler is in actual operation. The boiler was simulated under 11 different operating conditions. The data was then generated by simulating the random noise due to the uncertainties of the measurement devices. The measurement noise is considered white and uniformly distributed within the device uncertainty range.

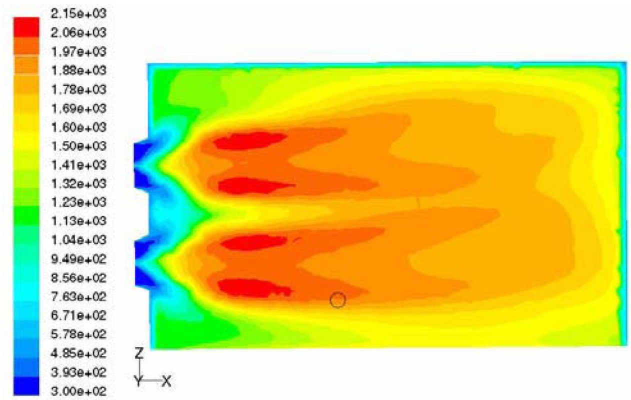

Fig.1a Temperature distributions (K) at a vertical plane passing through the two burner.

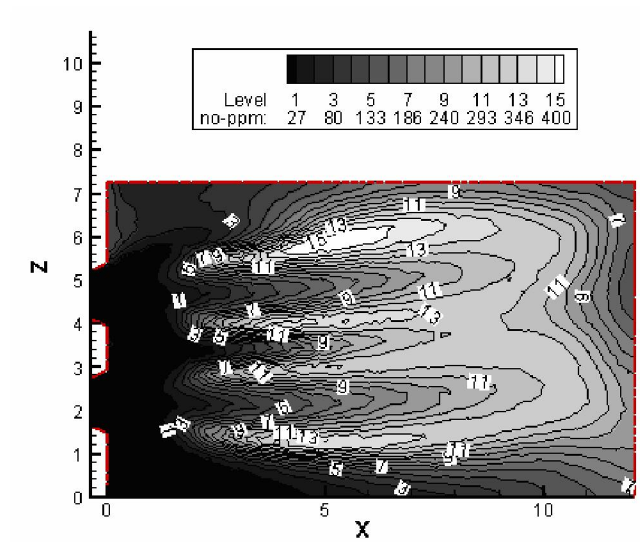

Fig. 1b. Concentration distributions of NO (ppm) at a vertical plane passing through the two burners.

3. ANN Model

One of the strengths of ANN is their capability to learn complex nonlinear relationships by training, which otherwise can hardly be modeled statistically or from first principles [11]. ANNs are composed of signal processing elements called neurons. Each consists basically of a summation node and an activation function. The neurons are interconnected and the strength of the interconnections is denoted by the parameters called synaptic weights. Feed Forward Neural Networks are composed of layers of interconnected neurons. Usually, an input layer, a number of hidden layers and an output layer. For sufficient number of hidden units, feed forward neural networks (FFNN) can approximate any continuous static input-output mapping to any desired degree of approximation [12,13]. Several types of ANNs have also been proposed, such as Radial Basis Function networks (RBFNs), and probabilistic neural networks (PNNs), see for example [14].

Recently, Function Networks (FunNet) has been proposed as a generalization of the standard neural network, Castillo et al. [15] and [16]. FunNet deals with general functional models instead of sigmoid-like ones. In the FunNets, the functions associated with the neurons are not fixed but are learnt from the data. FunNets allow neurons to be multi-argument, multivariate and different learnable functions, instead of fixed functions. A special class of FunNets is Polynomial learning Neural networks (PolyNets), where the functions are multivariate polynomials [17]. The advantage of polynomial learning networks is that they make possible the evaluation of complex, high-order models in acceptable time by composing simple transfer polynomials with weights that are computed efficiently by least-squares method. In the model considered here, the function are selected from the set of bivariate quadratic polynomials. The training algorithm selects the best polynomial for each combination of the input variables using the method of least squares. At each layer, for each pair of input variables we select the best transfer polynomial and optimize its parameters to fit the given data in the least squared error sense as follows

$$J(i,j) = \min_{k,w} \{\sum_{m=1}^{M} (y(m) - f_k(w_k, v_i(m), v_j(m)))^2\} \quad (1)$$

Where $\{v_j^1(m)\}$, for $j = 1,2,...l_1$. Are the outputs of the previous layer, $\{f_k\}$ is a given set of functions, w_k is the vector of the parameters of the function, which in our case is the polynomial coefficients. A typical structure of a polynomial network is shown in Fig. 2. It consists of one or more layers of transfer polynomials and an output layer. The output layer is basically a summation node.

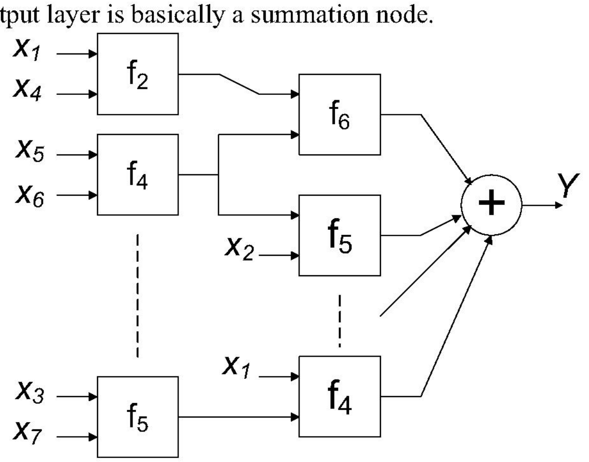

Fig. 2 Structure of polynomial networks.

The training algorithm proceeds as follows:

1- At the input layer, for each pair of input variables we select the best transfer polynomial which fits the given data in the least squared error sense.

$$J(i,j) = \min_{k,w} \{\sum_{m=1}^{M} (y(m) - f_k(w_k, x_i(m), x_j(m)))^2\}$$

$$\text{for } i, j = 1,2,..., N \quad (2)$$

The polynomial is defined by its index k, and its optimized coefficient vector w_k. For each k, we apply the LS technique to find the parameter vector w_k, and compute the residual mean squared error. The best polynomial is the one which achieves the least residual sum of squared error. Clearly, the number of the generated nodes is N^2.

2- Next, we sort the nodes in ascending order of their LSE, and select the best l_1 nodes, where l_1 is the desired maximum number of nodes in the first layer.

3- Compute the outputs $\{v_j^1(m)\}$, for $j = 1,2,...l_1$.

4- The nodes of the subsequent layers are computed by repeating the same steps, however by replacing the input x by v, i.e.

$$J(i,j) = \min_{k,w} \{\sum_{m=1}^{M} (y(m) - f_k(w, v_i(m), v_j(m)))^2\} \quad (3)$$

The residual minimum error is then sorted and the best l_i node are retained. The output from the retained nodes are computed and used in the subsequent layers.

5- Pruning: starting from the last hidden layer, we trace back the nodes, identifying the signal path from each node in the final layer to the input variables. We only keep the nodes along these paths, while the unused nodes are deleted. The network in this case will have at most l_1 node in the firs layer, at most l_2 nodes in the second layer, and so on.

6- Finally, the weights of the output node is computed by minimizing the squared error

$$E = \min_{w} \{\sum_{m=1}^{M} (y(m) - \sum_{i=1}^{l_f} w_{f,i} v_i^f(m))^2\} \quad (4)$$

Where l_f is the number of nodes of the final hidden layer.

4. Simulation Results

The following table gives the concentration of NOx (nitrogen oxides) and Oxygen in the exhaust gas of the simulated boiler under various operating conditions. The concentration is given in ppm. The operating conditions are determined by 6 variables: air/fuel ratio, fuel mass flow rate, air flow rate, flame maximum temperature, average combustion chamber temperature, and the outlet gas temperature. Ten sets of data are given for training, and two sets of data are reserved for testing.

Table 1. Boiler simulation esults for 12 operating conditions.

AFR	Mf	Ma	Maxtmp	Avetmp	Outtmp	Nox	O2
1.05	4.54	80.57	2131.6	1446.1	1171.5	303.5	12568
1.1	4.34	80.5	2132.5	1429.2	1158.6	280.7	18476
1.15	4.14	80.5	2132	1409	1130	256.9	24687
1.2	3.978	80.5	2132.8	1392.2	1111.9	236.5	31208
1.3	3.66	80.5	2133	1354.4	1076.5	200.1	47499
1.1	4.14	77	2128	1420	1144	269.8	18533
1.15	4.14	80.5	2132	1409	1130	256.9	24692
1.2	4.14	84	2136.6	1397.8	1122.9	245.4	31224
1.25	4.14	87.6	2141.2	1385.2	1114.6	235	38137
1.3	4.14	91	2144.7	1372	1108.4	225	48288
TEST DATA							
1.25	3.81	80.5	2133.8	1372.9	1092.1	218	38033
1.05	4.14	73.5	2122.3	1430.9	1142.6	282.1	12775

The data from CFD simulation is not enough to train neural networks. To generate further data from these operating conditions we simulated measurement uncertainty errors. The emission analyzer error is taken to be 2% of the full scale deflection FSD, the IR temperature measurement error is ± 10 C° (above 700 degrees), and is ± 6 degrees below 700 degrees, air flow rate measurement error is 5% of the FSD, and fuel flow rate error is 2% of the FSD. All measurement errors were taken to be uniformly distributed random variables. Six measurements were simulated around each operating condition, for a total of sixty set of measurements for training and 12 sets for testing.

The polynomial network for NOx prediction was built using two hidden layers and an output layer. The first layer was constructed by examining all combinations of input pairs, after sorting and removing redundant cases, only 15 nodes remain in the first hidden layer. The second layer is chosen by try and error to have 5 nodes. The output node is obtained as weighted sum of the outputs of the second layer. Finally, the pruning phase trimmed the number of effective nodes in the first layer to only 6 nodes. Column 2 of Table 2 gives is a summary of the performance of the PolyNet. To demonstrate the performance of the proposed polynet, a conventional feed forward MLP neural network was also constructed. Various FFNN configurations where tested, one hidden layer and two hidden layers, with varying number of neurons in each layer. The best performance was achieved with one hidden layer of 20 Neurons. All neurons were log-sigmoid functions. The training algorithms of MLP may give different results for different initial values of the MLP weights. Several runs were performed and the best result is reported in Table 2. Figure 3 shows the performance of the two networks in predicting NOx under two different operating conditions. These two operating conditions are given by the last two rows in Table 1. From Fig. 3 and Table 2, the superior performance of the proposed polynet is clearly demonstrated.

Similarly, a second polynomial network was constructed for prediction of O2. The network consists of two hidden layers and an output layer. The first hidden layer contains 12 nodes, and the second layer contains 4 nodes. The maximum error came to 2.42 % of FSD. Fig. 4 shows the performance of the polynomial network in O2 prediction under two operating conditions. The predicted O2 is compared with the simulated measurements from the CFD model.

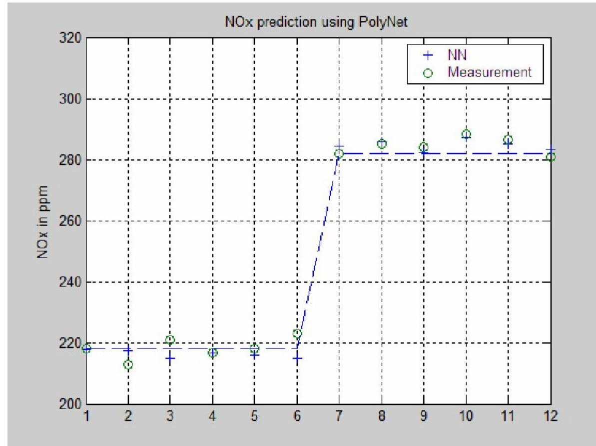

Fig. 3a NOx prediction using PolyNet

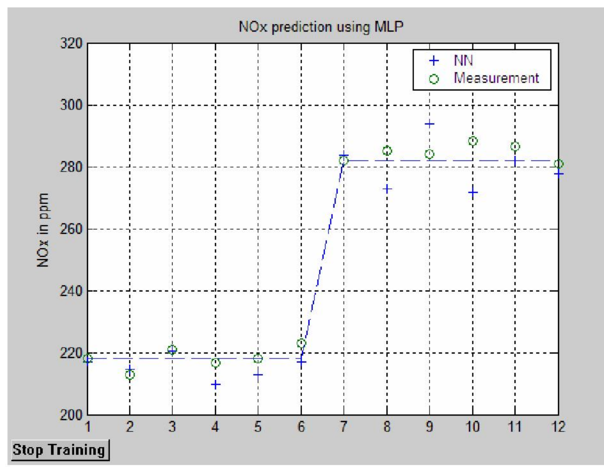

Fig.3b NOx prediction using a FFMLP neural

Table 2. Performance evaluation of the polynet and the conventional feed forward MLP NN.

	PolyNet	MLP
Max absolute error (ppm)	8.14	13.1
MSE	12.36	56.18
Maximum percent relative error	3.65 %	5.6%
Maximum percent error relative to the nominal values	1.83 %	4.26%
Maximum percent error relative to the FSD	3.27%	5.26%
Training time (seconds)	1.26	10.98 average
Complexity: number of weights of the network	71	141

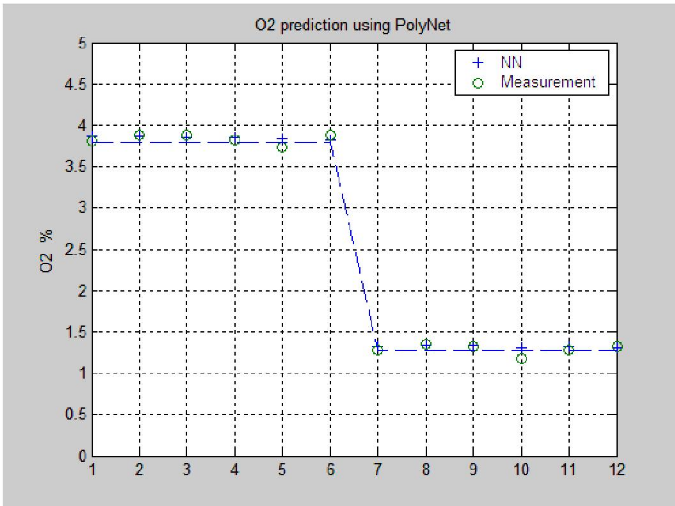

Fig. 4 Prediction of O2 using polynomial network under two operating conditions.

5. Conclusions

The paper provided an efficient polynomial network solution to the problem of on-line monitoring of NOx emission from industrial boilers. The effect of six variables were studied using 3D CFD simulation model and used by polynomial networks for prediction of NOx and O2 in the exhaust flue. The prediction of NOx and O2 are essential for efficient operation of the boiler while maintaining the NOx pollutant within a tolerable limit. The proposed softsensor has a simple modular structure for low cost implementation. The softsensor can be integrated with the boiler control system for optimization of boilers operation.

Acknowledgements

This work has been supported by SABIC (Saudi Arabia Basic Industry Corporation), and King Fahd University of Petroleum and Minerals under contract number SAB 2004/06..

References

[1] Michael L. Traver, Richard J. Atkinson, and Christopher M. Atkinson, "Neural network-based diesel engine emissions prediction using in-cylinder combustion pressure," *International spring Fuels & Lubricants Meeting & Exposition*, Dearborn, Michigan, May 3-6, 1999.

[2] R.H. Habtom, "Softsensing using recurrent neural networks," *Proc. of the 1998 ISIC/CIRA/ISAS Joint Conference,* pp. 342-347, September 1998.

[3] S.J. Qin, H. Yue, and R. Dunia, "A self validating inferential sensor for emission monitoring," *Proceedings of the American Control Conference*, pp. 473-477, 1997.

[4] D. Dong, T.J. McAvoy, and L.J. CHANG, "Emission monitoring using multivariable soft sensors," *Proceedings of the American Control Conference* 1995, 761-765.

[5] H. Al-Duwaish, L. Ghouti, T. Halawani, and M. Mohandes, "Use of artificial neural networks process analyzers: A case study," *Proceedings of the European Symposium on Artificial Neural Networks ESANN'2002,* Bruges (Belgium), pp. 465-470, April 2002.

[6] N. Li and S. Thompson, "A simplified non-linear model of NOx emission in a power station boiler," *Proceeding of the UKACC International Conf. on Control*, pp. 1016-1021, 1996.

[7] S.V. Patankar, *NUMERICAL HEAT TRANSFER AND FLUID FLOW.* First Edition, Taylor and Francis, 1980.

[8] M.A. Habib and J.H. Whitelaw, "The calculation of turbulent flow in wide angle diffusers," J. *Num. Heat Transfer*, Vol. 5, pp. 145-164, 1982

[9] S.Z. Shuja and M.A. Habib, "Fluid flow and heat transfer characteristics in axisymmetric annular diffusers," *Computers and Fluids*, Vol. 25, No. 2, pp.133-150, 1996

[10] M.A. Habib, R. Ben-Mansour, and M.A. Antar, "Flow field and thermal characteristics in a model of a tangentially fired furnace under different conditions of burner tripping," *J. Heat and Mass Transfer,* paper no. 10.1007/s00231-004-0593-6., 2004.

[11] Simon Haykin, *NEURAL NETWORKS; A COMPRE-HENSIVE FOUNDATION.* Macmillan Publishing Company Englewood Cliffs, NJ, 1994.

[12] K. Hornik, M. Stinchcombe and H. White, "Multilayer feed-forward network are universal approximators," *Neural Networks*, v2, p 359-366, 1989.

[13] J.J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. of the National Academy of Sciences of the U.S.A.*, Vol. 79, pp. 2554-2558, 1982.

[14] L.T. Fine, *FEEDFORWARD NEURAL NETWORK METHODOLOGY.* Spring–Verlag, New York, 1999.

[15] E. Castillo, "Functional networks," *Neural Process. Lett.* 7 (3), 151–159, 1998.

[16] E. Castillo, A. Cobo, R. Gómez-Nesterkin, and A. S. Hadi, "A general framework for functional networks," *Networks* 35(1): 70-82, 2000.

[17] H.R. Madala and A.G. Ivakknenko, *INDUCTIVE LEARNING ALGORITHMS FOR COMPLEX SYSTEMS MODELING.* CRC, ISBN: 0-8493-4438-7, 1994.