COMPARING TWO NONLINEAR STRUCTURES FOR SECONDARY AIR PROCESS MODELLING

K. Leppäkoski, M. Paloranta and E. Ikonen

Fax: +358 8 553 2439, Email: kimmo.leppakoski@oulu.fi, URL: www.oulu.fi
Systems Engineering Laboratory, University of Oulu, Finland

Keywords: identification, nonlinear systems, combustion, power plants, flue-gas emissions

Abstract

The secondary air process, as a part of a flue-gas oxygen-content control system in a fluidized bed combustor, was identified using two Hammerstein types of structures. A tensor product structure developed especially for the process, as well as a sigmoid neural network were examined for modelling the static part of the Hammerstein model. The two approaches were equal in accuracy. The better transparency of the tensor product structure makes it more appealing. Multiple data sets could efficiently be utilised since the structure of the model enables separate identification of the static nonlinear and linear dynamic parts. A more accurate model enables better tuning of the control system, in order to decrease flue-gas oxygen-content and NO\textsubscript{x} emissions.

1 Introduction

The secondary air process is a part of a flue-gas oxygen-content control system in thermal power plants. A part of the total air flow is blown through a secondary air process into the combustion chamber via three or four air feeding levels. The secondary air process is nonlinear and, at least in a practical sense, time-invariant.

Proper function of a secondary air control system is essential, in order to ensure an acceptable control of the combustion process. The timing of fuel and air flow is not easy in industrial cases [5] due to lack of suitable models.

The amount of excess air is one of the most important free control variables effecting NO\textsubscript{x} and CO emissions. The higher flue-gas oxygen-content is, the higher NO\textsubscript{x} emissions are, which was observed also in several experiments [6,7]. In most cases, flue-gas oxygen-content is controlled by using secondary air flow. Consequently, reducing NO\textsubscript{x} emissions requires a secondary air process model.

There are several methods for identification of nonlinear plants [10], for example neural networks [1] and wavelets. Constraints and a priori knowledge can be included in a proper cost function, taking account of the mismatch between model prediction and empirical data, non-smoothness of a model, mismatch between a model and a default model, using soft constraints and hard constraints [4]. Alternatively, assumptions can also be embedded in the model structure. In this case, Wiener/Hammerstein models provide a reasonable and feasible approximation of the general nonlinear time-series model [3, pp. 114].

In this paper, the use of the Hammerstein structure is considered. In Hammerstein models, a static non-linear part precedes a linear dynamic part. A sigmoid neural network and tensor product type structures are considered as the model candidate structures in the static part. A first order linear transfer function is used in the linear part.

The paper is organised as follows. First, the flue-gas oxygen-content control system is introduced, and the secondary air process and its behaviour are presented. Second, the process experiments are described. Finally, two types of secondary air process models are studied and compared.

2 Fluidised bed combustion process

In fluidised bed boilers, fuel flows and air flows are fed into a combustion chamber in which heat sand bed dries and grinds fuel before it burns. Primary air is blown from the bottom of the boiler, in order to fluidise the bed; secondary air is blasted above the bed. The air flows are staged, in order to control better the combustion process.

A process automation system is used to control a combustion process. A power controller maintains energy production on a desired level by changing the fuel feed rate. Its control signal is transmitted to the lower actuator controllers which control fuel feeding screws, air fans and air dampers.

A simulator [6,7] has been developed which (Fig. 1) contains: a power control system, a flue-gas oxygen-content controller system, a fuel feeding system, an air feeding system, and a flue-gas oxygen-content model. The power control system imitates a power controller. The fuel feeding system contains controllers and screw models. The air feeding system contains an air distribution calculation which defines the setpoints for the controllers, a primary air process, and a secondary air process.

The model for the secondary air process is utilised in two ways: in ratio control between fuel feed and air feed, and in flue-gas oxygen-content control. The power controller sends a
signal to calculation block, which distributes the signal to the fuel feed controllers and air feed controllers. Correct timing between the fuel feed controllers and air feed controllers is essential in order to stabilise the flue-gas oxygen-content. The quality of fuel is not homogenous, and especially biomasses may be very heterogenous, resulting in fluctuations in the flue-gas oxygen-content. The disturbances are commonly compensated by using secondary air flow. The flue-gas oxygen-content controller transmits a signal to the air flow controllers.

The purpose, initially, was to use an experimental design method, such as the Taguchi-method [8]. Nonetheless, there were often constraints not known in advance, as all alternatives to change the control signals were not possible. During an experiment series, the bed temperature increased too much since the lower air flow was too strong, resulting that most of fuel burnt in the bed. During the next experiment series, the control signal of the lower air level was slightly decreased, and the bed temperature did not increase.

The control signals correlate during normal operation, and all combinations do not fulfil in the real process. Thus, only the most common cases are necessary to explore. The training (7.5 hours) and validation data (5 hours) were composed of several separate experiments.

4 Model structures

The Hammerstein structure consists of a nonlinear static part followed by a linear dynamic part. Two static types were used in the Hammerstein models: a tensor product and a sigmoid neural network. The tensor product can be considered as a grey-box model made for this specific process, in order to take better the constraints of the process into account. The sigmoid neural network was used as a reference model, due to its flexible function mapping abilities. A first order output error type of time-series model was used in the linear part.

The models were trained iteratively, using the Levenberg-Marquardt method. The values were scaled to unit interval. The static part was first estimated, and then the linear part estimated using the latest static model.

Both models had four inputs and five outputs. Three air damper control signals and a fan control signal were the inputs; three air flow measurements and an air pressure measurement were utilised as the outputs. Both the inputs and the outputs were numbered from one to four, from the upper air flow to air pressure. Since the air fan affected all the air flows, the total air flow was taken as the fifth output.

4.1 Tensor product structure

The tensor product structure was composed of a product of logistic sigmoid growth functions, modified logistic sigmoid growth functions and power functions. The tensor product calculation in the static part can be considered as a global basis function.

Several different functions were studied, such as exponential, power and asymptotic regression functions. A power function was found to be the most suitable to model the effect of the air fan control signal, since the inherent bounds of the process were easy to set. Using a priori knowledge of the behaviour of the air flows, different types of sigmoid growth functions were compared: the Gompertz, logistic, Richards, Morgan-Mercer-Flodin and Weibull [9, pp. 61-63]. The logistic type was found to be the most suitable.
In the tensor product construction, the model input is given by

\[ y_j(k) = f_j(\varphi(k), \beta) = \prod_{i=1}^{I_j} \kappa_m(\varphi_i, \beta_{j,i}) \]  

(1)

where \( J \) is the number of the outputs \((j = 1, \ldots, 5)\), and the column vector \( \varphi \) of the model inputs is \( I \) dimensional \((i = 1, 2, 3, 4)\). \( M \) is the index for the type of a basis function \((m = 1, 2, 3)\).

The logistic sigmoid function is given by

\[ \kappa_1(\varphi, \beta) = \frac{1}{1 + \exp(\beta_{j,1} - \beta_{j,2} \varphi_i)} \]  

(2)

the modified logistic sigmoid function by

\[ \kappa_2(\varphi, \beta) = \beta_{j,1} + \frac{\beta_{j,2}}{1 + \exp(\beta_{j,3} - \beta_{j,4} \varphi_i)} \]  

(3)

and the power function by

\[ \kappa_3(\varphi, \beta) = \beta_{j,1} \varphi_i \beta_{j,2} \]  

(4)

The following models were used:

The top air flow \((j = 1)\):
\[ i = 1, \text{ air damper 1, } \kappa_1 \]
\[ i = 2 \text{ and } 3, \text{ air dampers 2 and } 3, \kappa_2 \]
\[ i = 4, \text{ air fan, } \kappa_3 \]

The middle air flow \((j = 2)\):
\[ i = 1 \text{ and } 3, \text{ air dampers 1 and } 3, \kappa_2 \]
\[ i = 2, \text{ air damper } 2, \kappa_1 \]
\[ i = 4, \text{ air fan, } \kappa_3 \]

The bottom air flow \((j = 3)\):
\[ i = 1 \text{ and } 2, \text{ air dampers 1 and } 2, \kappa_2 \]
\[ i = 3, \text{ air damper } 3, \kappa_1 \]
\[ i = 4, \text{ air fan, } \kappa_3 \]

The air pressure \((j = 4)\):
\[ i = 1-3, \text{ air dampers } 1-3, \kappa_2 \]
\[ i = 4, \text{ air fan, } \kappa_3 \]

The total air flow \((j = 5)\):
\[ i = 1-3, \text{ air dampers } 1-3, \kappa_1 \]
\[ i = 4, \text{ air fan, } \kappa_3 \]

A useful interpretation is to think that a main control signal affects a main output, and other control signals are correcting. A major impact is that the user of the model can easily comprehend the effects of the control signals.

For example, the relation between the lower air control signal and the lower air flow presents one of the five main input/output pairs. The relationship between the main input control signal and the output signal has the shape of the increasing logistic sigmoid function (equation (2)) in this case. The upper air, the middle air and the air fan control signals affect the lower air flow, and thus they are correcting. The correcting effects of the upper and middle air flows have the shapes of the decreasing sigmoid function (equation (3)). \( \beta_{j,1} \) is close to one, and \( \beta_{j,2} \) is a small negative number. The correcting effects of the air flows to the lower (main) air flow are not large. The effect of the air fan control signal is otherwise significant. In equation (4), \( \beta_{j,2} \) is close to 0.5. The relationships above are the same when the upper or middle air flow is the main control signal.

Let us consider another example. The air fan control signal is the main input signal, and the air pressure is the output signal. Then \( \beta_{j,2} \) (in equation (4)) is close to two (compare with Bernoulli’s law). The air flow control signals are correcting. In equation 3, \( \beta_{j,1} \) is close to one and \( \beta_{j,2} \) is a small positive number. The corrections are therefore increasing when the minor control signals are enlarging. The effects of the minor control signals are quite significant.

4.2 Sigmoid neural network

An output of one-hidden-layer sigmoid neural network is given by, [3, pp. 92].

\[ \hat{y} = f(\varphi, \alpha, \beta) = \sum_{h=1}^{H} \alpha_h g_h(\varphi_i, \beta_{h,i}) + \alpha_{H+1} \]  

(5)

where \( H \) is the number of hidden nodes, \( \alpha \) and \( \beta \) are parameters, and

\[ g_h(\varphi, \beta) = \frac{1}{1 + \exp\left(-\sum_{i=1}^{I} \beta_{h,i} \varphi_i - \beta_{h,i+1}\right)} \]  

(6)

Since most of the nonlinearities of the process had inherently the shape of the logistic sigmoid function, in increasing or decreasing direction, the sigmoid neural networks contained ten nodes. Logistic sigmoid growth functions were used as basis functions. There were four inputs and five outputs. The network can be considered as five MISO models.
5 Experimental comparison

The outputs of two Hammerstein models, in which the tensor product structure and the sigmoid neural network were used in the static part, were compared to the training and validation data. A part of the validation is shown in Figures 3 and 4 for air pressure, with which the difference between the identified models and the measurements was the largest.

![Figure 3](image1)

Figure 3. Measured air pressure (solid) and tensor product type based model output (dashed), on a part of the validation data.

![Figure 4](image2)

Figure 4. The changes in the four input signals for the air pressure (the tensor product type), on a part of the validation data.

\[ R^2 = 1 - \frac{SS_E}{SS_T} \]  
(7)

where

\[ SS_E = \sum_{k=1}^{K} (Y_j - \hat{Y}_j) (Y_j - \hat{Y}_j) \]  
(8)

For the \( j \)th output, where \( Y_j \) is measured output and \( \hat{Y}_j \) is the estimated output (\( j = 1, 2, 3, 4, 5 \)).

The root-mean-squared errors (RMSE) are calculated by the following [3, pp. 163]

\[ RMSE = \sqrt{\frac{1}{K} \sum_{k=1}^{K} (Y_j - \hat{Y}_j)^2} \]  
(10)

where \( Y_j \) is measured output, \( \hat{Y}_j \) is estimated output (\( j = 1, 2, \ldots, k \)), and \( K \) is the number of data points.

<table>
<thead>
<tr>
<th></th>
<th>R(^2) -</th>
<th>R(^2) -</th>
<th>RMSE -</th>
<th>RMSE -</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>tp</td>
<td>snn</td>
<td>tp</td>
<td>snn</td>
</tr>
<tr>
<td>Upper air flow</td>
<td>0.9807</td>
<td>0.9821</td>
<td>0.009883</td>
<td>0.009524</td>
</tr>
<tr>
<td>Middle air flow</td>
<td>0.9973</td>
<td>0.9979</td>
<td>0.012083</td>
<td>0.001665</td>
</tr>
<tr>
<td>Lower air flow</td>
<td>0.9915</td>
<td>0.9875</td>
<td>0.014388</td>
<td>0.017420</td>
</tr>
<tr>
<td>Air pressure</td>
<td>0.9831</td>
<td>0.9671</td>
<td>0.009784</td>
<td>0.013631</td>
</tr>
<tr>
<td>Total air flow</td>
<td>0.9751</td>
<td>0.9922</td>
<td>0.014812</td>
<td>0.008134</td>
</tr>
</tbody>
</table>

Table 1: Training results – Hammerstein model: tensor product and sigmoid neural network.

The statistics in Tables 1 and 2 based on RMSE criterion suggest that the tensor product type and the sigmoid neural network had not significant difference. RMSE on the validation data is smallest for the top air flow using the sigmoid neural network model; the contrary holds for the air pressure.

<table>
<thead>
<tr>
<th></th>
<th>R(^2) -</th>
<th>R(^2) -</th>
<th>RMSE -</th>
<th>RMSE -</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>tp</td>
<td>snn</td>
<td>tp</td>
<td>snn</td>
</tr>
<tr>
<td>Upper air flow</td>
<td>0.9083</td>
<td>0.9442</td>
<td>0.016135</td>
<td>0.012587</td>
</tr>
<tr>
<td>Middle air flow</td>
<td>0.9497</td>
<td>0.9489</td>
<td>0.019487</td>
<td>0.019644</td>
</tr>
<tr>
<td>Lower air flow</td>
<td>0.9816</td>
<td>0.9711</td>
<td>0.016338</td>
<td>0.020476</td>
</tr>
<tr>
<td>Air pressure</td>
<td>0.9640</td>
<td>0.8933</td>
<td>0.001065</td>
<td>0.017317</td>
</tr>
<tr>
<td>Total air flow</td>
<td>0.9515</td>
<td>0.9786</td>
<td>0.012382</td>
<td>0.008230</td>
</tr>
</tbody>
</table>

Table 2: Validation results – Hammerstein model: tensor product and sigmoid neural network.

The best validation result among the outputs using \( R^2 \) was obtained for the bottom air flow, for which the data were probably the richest. The worst results were obtained for the...
top air flow (tensor product) and the air pressure (sigmoid neural network). The middle air flow had the most even results in the training and validation data.

There is no particular reason why goodness-of-fit should vary between training and test data sets. Measurement or process noise is not extensive, and all the air flows have similar noise characteristics. Yet, the top air flow model has the largest difference in goodness-of-fits between training and validation data; the difference in the pressure model is also significant. Instead, the difference in goodness-of-fits for the bottom air flow model is minor. Comparison between training and validation data indicates that the richness of the training data could be improved.

6 Conclusions

The Hammerstein structure contains a nonlinear static part and a linear dynamic part. The use of a tensor product structure and a sigmoid neural network in the static part were studied. There were no significant differences in accuracy between the identified models. For this specific process, the tensor product structure is more transparent to use, however.

Measured data can be efficiently utilised by using the Hammerstein approach, since the estimation of the parameters in the static and dynamic parts can be conducted separately. Separate data blocks from different operating conditions, possibly with different sampling frequencies, can be easily used in the identification of the static part. Notice, that for dynamic modelling the conditions for data are stricter. Taking account of the limited resources during normal process operation, this enables modelling in a wider operating area using routinely collected data.

Acknowledgements

The authors wish to thank Foster Wheeler Energia Oy, Air-Ix Tuotantotekniikka Oy and Finnish National Technology Agency. E. Ikonen was financially supported by the Academy of Finland, projects no 48545 and 203231.

References