Strategy for Dynamic Process Modeling Based on Neural Networks in Macroscopic Balances

Henricus J. L. van Can, Chris Hellinga, Karel Ch. A. M. Luyben, and Joseph J. Heijnen
Kluyver Laboratory for Biotechnology, 2628 BC Delft, The Netherlands

Hubert A. B. te Braake
Control Laboratory, 2628 CD Delft, The Netherlands

Gray box models combine the short development time of data-driven black box models with extrapolation properties of knowledge-driven first principles models (white box), which in (bio)chemical engineering are always based on macroscopic balances. By modeling the inaccurately known terms in a macroscopic balance with a black box model, one naturally obtains a so-called serial gray box model configuration. The identification data must cover only the input-output space of the inaccurately known terms, and the accurately known terms can be used for reliable extrapolation. In this way, the serial gray box configuration results in accurate models with known extrapolation properties with a limited experimental effort. This strategy is demonstrated for the modeling and control of a pressure vessel using real-time experiments. For this case, the strategy is superior to a black box modeling approach that requires much more data and to a parallel gray box approach that results in a model with poor extrapolation properties. Moreover, neural networks are an accurate and convenient modeling tool for the black part in gray box model configurations, because a very fast noniterative training algorithm is used for training neural networks.

Introduction

Neural networks as black box modeling tools have already been used for many applications in industry, business, and science (Widrow et al., 1994). The ability of neural networks to approximate any continuous function to any desired accuracy (Cybenko, 1989) has been the basis for many of these applications. The main advantage of the use of black box modeling techniques like neural networks is that, within a reasonable amount of time, one can obtain a highly accurate mathematical model of a system without detailed knowledge of the system. However, black box approaches are mainly data driven and the resulting models are not believed to have any extrapolation properties. Therefore, data used for identification should cover the whole domain of interest in order to avoid the dangers of extrapolation when using the model. For complex systems with multiple inputs and outputs, this can result in the need for an enormous amount of experimental data. When these data are not available from prior data acquisition, the effort and time required to obtain these data is a serious bottleneck for the application of black box modeling techniques. Especially for chemical and biochemical processes, it can be very time and money consuming to obtain the necessary data.

In white box modeling approaches, the model development is mainly driven by knowledge of the relevant mechanisms and by so-called first principles (macroscopic balances, thermodynamics, and so on). The resulting first principles models are intended to be more or less generally applicable and to have good extrapolation properties. The necessary knowledge for a specific system is usually not directly available. Therefore, most effort in the white box modeling approach is devoted to revealing all relevant mechanisms and quantifying these mechanisms correctly. This usually requires an extensive research program (including experiments), which can also be very time- and money-consuming. Here a compromise must be made in order to save time and money. Therefore, the first principles models that are used in practice often have a limited accuracy for one specific process, because for practi-

Correspondence concerning this article should be addressed to H. J. L. van Can.

cal reasons minor mechanisms are neglected and only the major mechanisms are taken into account.

Given these considerations, arriving at a combination of a black and white box modeling approach, that is, a so-called gray box modeling approach is the challenge. More specifically, such a gray box modeling approach must lead to a short development time for accurate models with good interpolation and extrapolation properties. Two types of extrapolation should be distinguished here. In the first type of extrapolation a variable is applied outside the *range*, in which it was varied during identification. This will be called *range* extrapolation. In the second type of extrapolation the model is applied outside the *dimension* of the identification data, meaning that during the use of the model a variable is changed that was kept constant during identification. This will be called *dimensional* extrapolation. The objective of this article is to show a strategy to develop accurate gray box models using as little data as possible, while maintaining especially the dimensional extrapolation properties.

A gray box approach can be defined as a suitable combination of a black and a white box approach, but at this moment little is known about what is suitable and what is not. Thompson and Kramer (1994) made a systematic inventory of different ways of how prior knowledge and neural networks can be combined for modeling chemical processes. This prior knowledge can be used for the selection of appropriate inputs, the *a-priori* determination of the networks architecture (Mavrovouniotis and Chang, 1992), the definition of constraints during training (Joerding and Meador, 1991), and for development of first principles models. In this article two gray box model configurations are elaborated where a neural network is combined with a first principles model. The first gray box model configuration is the so-called parallel configuration: the neural network is placed parallel with a first principles model. The neural network is in fact an error model, which should model the difference between a first principles model and reality. This configuration was used by Su et al. (1992) in a simulation example. Côte et al. (1995) demonstrated the parallel approach on real-time data of a wastewater treatment plant. Both cases showed improved interpolation in the range of the identification data, but the extrapolation properties of the parallel gray box models were not studied. The second gray box model configuration is the so-called serial configuration: the neural network is placed in series with a first principles model. Psichogios and Ungar (1992) demonstrated the serial configuration in simulation. Thompson and Kramer (1994) used a combination of the serial and parallel configuration in a simulation example. Schubert et al. (1994) combined the serial model configuration with a fuzzy expert system to model a real-time fed-batch baker's yeast production. In these three cases, the used model configurations showed better interpolation and range extrapolation properties than pure black box neural network models; however, the dimensional extrapolation properties were not studied.

In order to fully benefit from the gray box configurations (short development time with a limited amount of identification data for accurate models with reliable extrapolation properties), it is important to be able to relate *a-priori* the application domain of the model to the required domain for the identification data. Only then it is possible to define *a-priori* the minimum domain for the identification data to obtain a model with a predefined application domain. Although range extrapolation was shown in the articles of Psichogios and Ungar (1992), Thompson and Kramer (1994), and Schubert et al. (1994), the extrapolation properties of their model configurations was not studied extensively and the dimensional extrapolation properties were not studied at all. Therefore, it is still not possible to relate *a-priori* the application domain of the model to the required domain for the identification data. The purpose of this article is to show that by focusing on dimensional extrapolation it is very possible to relate the application domain of the model to the required domain for the identification data. As a result, the identified models can be extrapolated reliably outside the domain of the limited set of identification data. Furthermore, the identification of the neural network part still remains an important issue. A fast, simple procedure for the identification of neural networks can stimulate the application of neural networks in gray box model configurations. In addition, experience using gray box models in a real-time situation is still limited.

In this article a serial gray box strategy is presented based on the application of neural networks in macroscopic balances resulting in accurate models with reliable extrapolation properties using only a limited data set for identification. The strategy is demonstrated on the modeling and control of a pressure vessel, for which real-time results are presented. An elaborate procedure is used to test the performance of the candidate model with respect to its interpolation and dimensional extrapolation properties. In addition, the candidate model is tested for its ability to function well in a model based predictive controller (MBPC). The candidate model is compared with pure black box neural network models and with a serial gray box model containing a polynomial, with respect to its interpolation and dimensional extrapolation properties. In order to clarify the origin of the improved dimensional extrapolation of the obtained serial gray box model, it is also compared with a parallel gray box model.

Finally, all neural network structures are identified using the fast noniterative training algorithm developed by te Braake and Van Straten (1995). This way it was possible to train and test many possible neural network configurations within minutes, so that the identification of the neural network structures was less cumbersome than in a situation where iterative training algorithms like back propagation would be involved. Combined with the general function approximation properties of neural networks, this algorithm makes the use of neural networks a very practical modeling tool for the black box part in gray box model configurations.

Serial Gray Box Modeling Strategy Based on Macroscopic Balance

The serial gray box modeling strategy can almost naturally be combined with the general structure of first principles dynamic models in (bio)chemical processes, which are always based on macroscopic balances, e.g., mass, energy, or momentum balances. These balances themselves are correct, but some terms in the balance might be known with a higher level of accuracy than others. For example, terms associated with convective and diffusive transport are usually more accu-

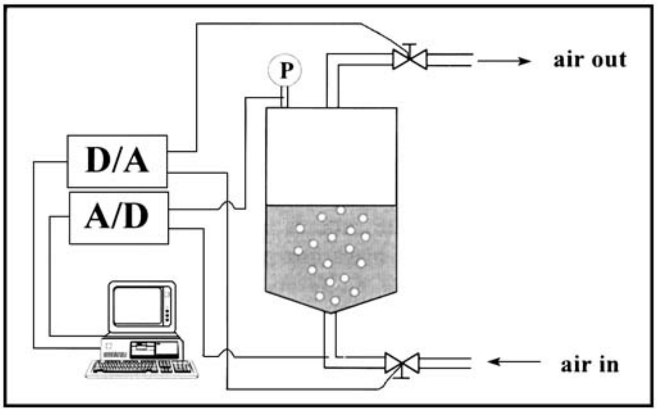

Figure 1. Experimental setup.

rately known than terms associated with transport between phases, kinetic terms related to (bio)chemical conversions, or friction losses in momentum balances. If these inaccurately known terms of the macroscopic balances are modeled by a neural network, one quite naturally obtains the so-called serial gray box model configuration.

In this article it will be shown that in such a configuration the identification data only have to cover the input-output space of the inaccurately known terms instead of covering the whole input-output space of the complete system. As a result, one obtains an accurate model with reliable dimensional extrapolation properties, without the need for many identification experiments and without the need to develop a rigorous first principles model where many more mechanisms would have to be taken into account. The performance of the model outside the domain of the identification data is reliable and can be understood, as long as the extrapolation relies only on the accurately known terms in the macroscopic balance (dimensional extrapolation).

Test Case

The presented test case refers to the modeling and control of a pressure vessel, operating at 30°C. The overall experimental setup is presented in Figure 1. At the bottom of the vessel, air was blown in. The specified flow rate was kept at the required value by local mass-flow controllers, and the valve position in the outlet could be manipulated directly. The nonlinear steady-state characteristic of the system is displayed in Figure 2. The 0.040 m^3 vessel was filled with 0.025 m^3 of water resulting in a head space of 0.015 m^3. When the system was operated in a single-input single-output (SISO) mode, the gas-flow rate was kept constant at 3.75e-4 m^3/s and only the valve position was manipulated. When the system was operated in a multi-input single-output (MISO) mode, both the gas-flow rate and the valve position were manipulated. In both cases the pressure was the only measured output variable. The pressure was measured by a Jumo 4-AP2-40 pressure sensor. In order to reduce system and measurement noise, the pressure signal was always averaged over a period of 0.25 s, while the sample interval was 5 s. All equipment was attached to a Personal Computer with a 486-50 MHz processor via DA/AD converters. All software was custom made in order to allow maximum flexibility.

The model to be developed had to predict the pressure of the above mentioned dynamic system, when both the valve position and the gas-flow rate could change. Furthermore, the model was implemented in a MBPC structure for pressure control. In Van Can et al. (1995a) a complete black box neural network model was already developed for this situation. In that case, the input space of the gas-flow rate and valve position was covered by 750 identification data pairs in order to avoid extrapolation during the use of the model. In this article, only 300 data pairs that were generated by only a varying valve position and constant gas-flow rate (SISO mode) were used for the identification of a serial gray box model for the specified purpose (MISO mode).

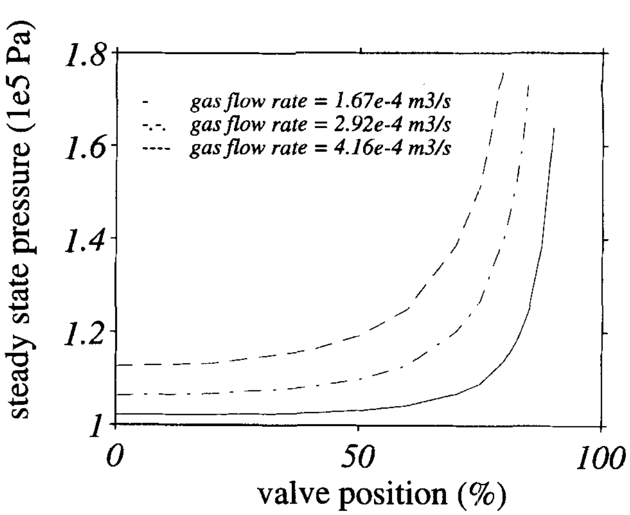

Figure 2. Nonlinear steady-state characteristic of the system.

Model Test and Comparison Procedure

Before the results are presented and discussed in detail, the test and comparison procedure of candidate models followed in this article will be outlined (Figure 3). In Figure 3 two lines of approach can be recognized: one associated with model testing and one associated with model comparison. In the vertical direction of Figure 3 one can see that the different models are tested for their interpolation and dimensional extrapolation properties, and that finally the best model is tested for its ability to function well in MBPC. The interpolation was performed on a data set from the SISO mode containing only a varying valve position as single input and the pressure as single output. The dimensional extrapolation test was performed on a data set from the MISO mode containing a varying valve position and a varying gas-flow rate as multiple inputs, and the pressure as single output. In the controller test, both the valve position and the gas-flow rate could change (MISO mode). In addition, the model was challenged to deal with a head space volume that was three times smaller than the head space volume during identification, so that the model was also tested for its dimensional extrapolation properties in that region.

In the horizontal direction of Figure 3, one can see that the performance of three gray box models is compared: a serial gray box model containing a neural network; a serial gray box model containing a polynomial; and a parallel gray box model containing a neural network. These three gray box

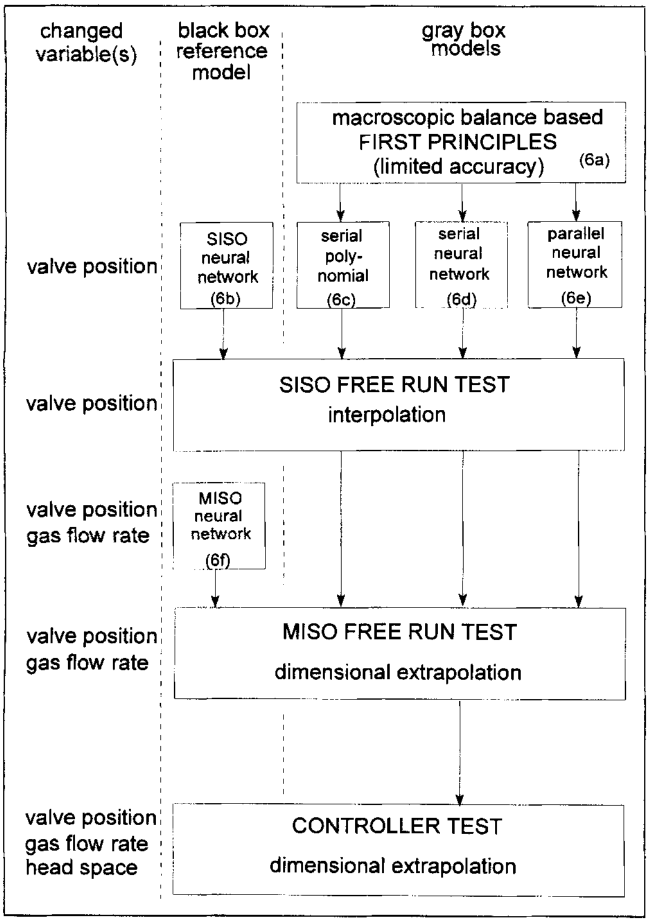
changed variable(s)
black box reference model
gray box models
macroscopic balance based FIRST PRINCIPLES (limited accuracy) (6a)
valve position
SISO neural network (6b)
serial poly-nomial (6c)
serial neural network (6d)
parallel neural network (6e)
valve position
SISO FREE RUN TEST interpolation
valve position gas flow rate
MISO neural network (6f)
valve position gas flow rate
MISO FREE RUN TEST dimensional extrapolation
valve position gas flow rate head space
CONTROLLER TEST dimensional extrapolation

Figure 3. Modeling and test procedure.

models are based on the same first principles model, which typically has a limited accuracy. The idea behind the serial gray box model containing a neural network was explained earlier in the section on serial gray box modeling strategy. As an alternative for a neural network part, a polynomial is also considered in the gray box configuration. The parallel gray box model is considered in order to understand the origin of the extrapolation properties of gray box models. Properly identified black box neural network models served as a reference in the interpolation test (SISO black box neural network) and extrapolation test (MISO black box neural network model).

The three gray box models and the SISO black box neural network reference model were all identified with the same 300 data pairs generated with a varying valve position and constant gas-flow rate (SISO mode). The MISO black box model was identified with 750 data pairs generated with a varying valve position and a varying gas-flow rate (MISO mode).

The performance of the three gray box models was compared in a test for interpolation and a test for dimensional extrapolation. In the interpolation test, the models were confronted with a test set containing data that were not used for the calculation of the model parameters, and that were generated by only a varying valve position and a constant gas-flow rate (SISO mode). The performance of the models was tested by using the models to recursively predict the pressure over the whole horizon of the test set, given only the initial pressure $[y(0)]$ and the input signal $[u(k)]$. This test will be called the SISO-free run test and was a part of the identification procedure as will be explained in the section on identification. The free run test procedure was used, because in this test small errors can accumulate to larger errors, so differences between candidate models will be clearer. As mentioned before, a proper black box neural network model for the SISO mode served as a reference point.

In the extrapolation test the three gray box models identified in the SISO mode were confronted with a test set containing data that were not used for the calculation of the model parameters, and that were generated by a varying valve position as well as by a varying gas-flow rate (MISO mode). Again, the performance of the models was tested in recursive mode. This test will be called the MISO free run test. Obviously, in the MISO free run test the three gray box models were tested for dimensional extrapolation because of the inclusion of a varying gas-flow rate. A complete black box neural network model, identified using data with a varying valve position and a varying gas-flow rate (MISO mode) (Van Can et al., 1995a), served as a reference point in the dimensional extrapolation test for the three gray box models.

Finally, the gray box configuration that had the best performance in the dimensional extrapolation test was used in a MBPC structure. In the controller the model was tested for its ability to guide the system in MISO mode stepwise from one steady state to another steady state. Although the data used for identification and for the free run test were very dynamic, the typical dynamics of a stepwise setpoint change in a controller situation were not explicitly present in these data. In addition, the head space volume of the pressure vessel was reduced with a factor three in order to test the serial gray box model in the controller even more for its dimensional extrapolation properties. The MBPC controller was tuned without severely constraining the change of the manipulated variables, so that the closed-loop stability of the controller was very dependent on the accuracy of the selected serial gray box model.

Controller

Within the MBPC scheme, the model is used to calculate the effects of possible future control signals. The predictions supplied by the model are passed to a numerical optimization routine, which attempts to minimize a specific performance criterion in the calculation of the suitable control signals. The control signal is chosen to minimize a performance criterion (J) of the form (Hunt et al., 1992)

$$J = \sum_{i=1}^{N_o} \sum_{j=1}^{H_p} [y_{i,r}(k+j) - \hat{y}_i(k+j)]^2 + \sum_{i=1}^{N_i} \sum_{j=1}^{H_c} \lambda_{i,j} \cdot [u_i(k+j-1) - u_i(k+j-2)]^2 \quad (1)$$

N_i refers to the number of inputs, N_o to the number of outputs, H_p to the prediction horizon, and H_c to the control horizon. The λs are the control weights and $y_{i,r}$ is the refer-

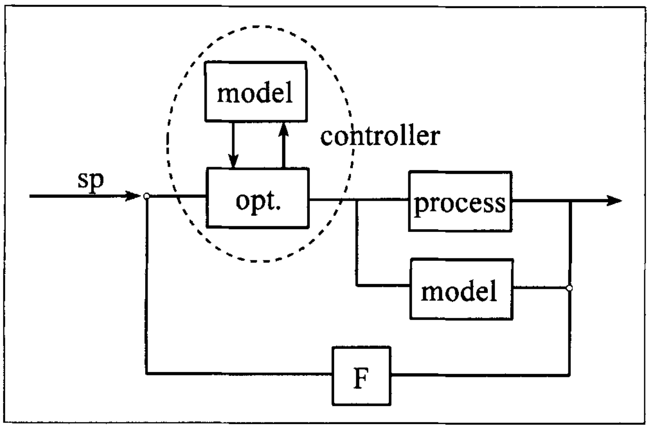

Figure 4. Total scheme of MBPC.

Optimization routine (opt.) is part of the controller.

ence signal for output i. The output of the process and the control signal can be subjected to constraints. Usually, the prediction horizon is larger than one, but only the first of the calculated control signals is implemented. The difference between the predicted output and the measured output is fed back to the reference signal in order to avoid offset due to model errors or disturbances. The MBPC structure is presented in Figure 4. For more details about MBPC, refer to Garcia et al. (1989). The controller that was used as a final test for the best gray box model was a multistep ahead predictive controller with $H_p = 5$, $H_c = 2$, and $\lambda = 1.0\text{e-}5$ (gas-flow rate in L/min and the pressure in bar). The maximum allowed change of the valve position in one sample period was 10%. The maximum allowed change of the gas-flow rate in one sample period was 5.0e-5 m^3/s. Further details about the application of MBPC to the MISO mode of the system under consideration can be found in Van Can et al. (1995a).

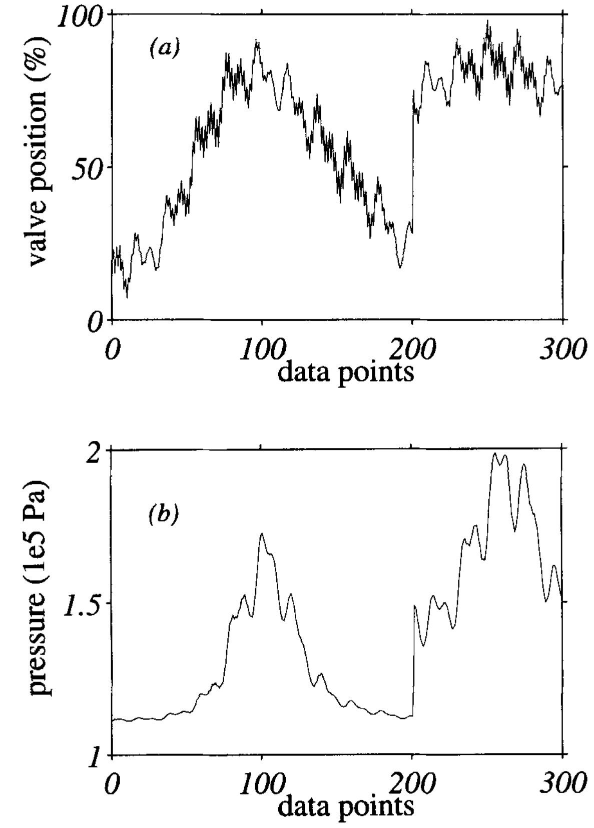

Figure 5. Identification data generated by only a varying valve position.

(a) Input signal; (b) resulting measured output signal.

Results Model Identification

Introduction

The identification data were obtained by changing the valve position according to a predefined wave signal, as can be seen in Figure 5a (input signal) and the Figure 5b (resulting measured output). This resulted in 300 data points for identification. In this article only feedforward artificial neural networks are considered. In all cases the valve position $(u_1(k))$ and the actual pressure $[y(k)]$ served as inputs for the neural network. The identification procedure for neural networks is described in detail by Van Can et al. (1995b). A fast noniterative training algorithm described in Appendix A was used to calculate the parameters of all neural networks. The SISO free run test served as a cross validation test to determine the number of hidden nodes.

First principles model

For the system under consideration the relevant macroscopic balance could directly be written down, as can be found in Appendix B. Using Euler's approximation the resulting equation is given by

$$y(k+1) = y(k) + \Delta t \frac{RT}{V_h} 39.7 \cdot \left(u_2(k) - \sqrt{K[u_1(k), y(k)] \cdot \ln\left(\frac{y(k)}{p_0}\right)} \right) \qquad (2)$$

where

y = pressure, Pa
R = gas constant, J/mol/K
T = temperature, K
V_h = head space of the vessel, m^3
u_1 = valve position, %
u_2 = incoming air flow, m^3/s
p_0 = atmospheric pressure, Pa
K = lumped friction coefficient of the outlet, m^6/s^2
Δt = sample interval, s

The future pressure $y(k+1)$ is predicted on the basis of the actual pressure $y(k)$ and two inputs [valve position $u_1(k)$ and gas flow rate $u_2(k)$]. Possible inaccuracies in the model predictions of $y(k+1)$ are caused by the parameter K, which is associated with the friction in the outlet. The friction will depend on the valve position and on the velocity of the gas in the outlet pipe. This velocity will depend on the difference between the pressure inside the vessel and the atmospheric pressure. Therefore, the friction parameter $K(k)$ is considered to be a function of the valve position $[u_1(k)]$ and the actual pressure $[y(k)]$. Here the compromise is made for the

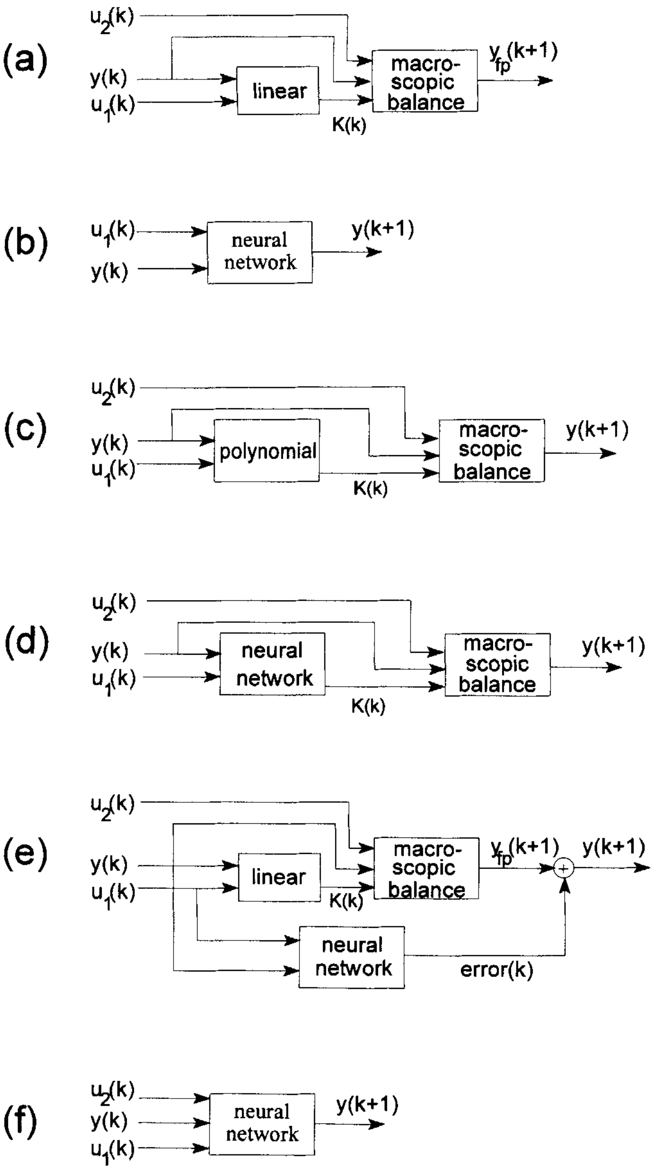

Figure 6. Different model configurations.

(a) First principles model, containing a linear correlation for $K(k)$; (b) black box neural network reference model for the SISO mode; (c) serial gray box model, containing a polynomial for $K(k)$; (d) serial gray box model, containing a neural network for $K(k)$; (e) parallel gray box model, based on the first principles model and a neural network error model; (f) black box neural network reference model for the MISO mode.

first principles model, meaning that the mechanisms associated with the friction in the outlet are not fully taken into account. For the first principles model a simple linear equation was used to approximate $K(k)$:

$$K(k) = a_1 \cdot u_1(k) + a_2 \cdot y(k) + a_0 \qquad (3)$$

The parameters a could be calculated by least-squares estimation, using the data from Figure 5. It is accepted that, due to this simple linear correlation for $K(k)$, this first principles model (Figure 6a) has a limited accuracy, because this is typical for first principles models used in practice. It is especially the purpose of this article to see which gray box configuration leads to efficient improvement of the accuracy of such a first principles model.

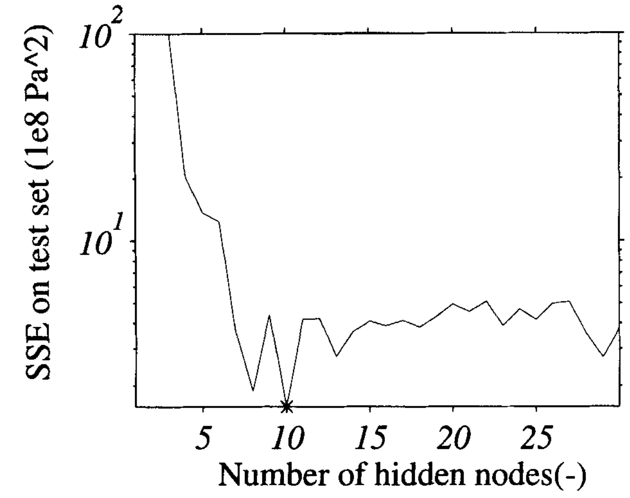

Figure 7. Selection of the number of hidden nodes for the black box neural network reference model (Figure 6b).

SISO black box neural network reference model

In the SISO free run test the performance of a properly identified black box neural network model for the SISO mode was used as a reference for the interpolation properties of the three gray box models. For this neural network the valve position $[u_1(k)]$ and the actual pressure $[y(k)]$ were used as inputs to predict the future pressure $[y(k+1)]$. No additional delay terms such as $[u_1(k-1),\ y(k-1)]$ for the input of the neural network were considered, because this was also not done for the other model configurations. The model configuration of the SISO black box neural network reference model is given in Figure 6b. Due to the speed of the used training algorithm (Appendix A), it was no problem to calculate the parameters of many possible neural networks. Thirty neural networks configurations varying in the number of hidden nodes from 1 to 30 could be calculated within 5–10 min on a PC486-66 MHz. All these candidate black box neural network models were tested for their performance in the SISO-free run test. In Figure 7 it can be seen that a neural network with ten neurons in the hidden layer had the smallest sum of squared errors (SSE). Therefore, the performance of this black box neural network model was chosen to serve as a reference for the performance of the gray box models in the SISO-free run test.

Serial gray box models

In the two serial gray box model configurations a polynomial and a neural network were used to approximate the friction factor $K(k)$ as a function of the valve position $[u_1(k)]$ and the actual pressure $y(k)$. The parameter $K(k)$ used for the identification of the parameters of the functions $K(k) = f[u_1(k), y(k)]$ could be calculated from the identification data (Figure 5) using Eq. 2. The obtained $K(k)$ values, shown in

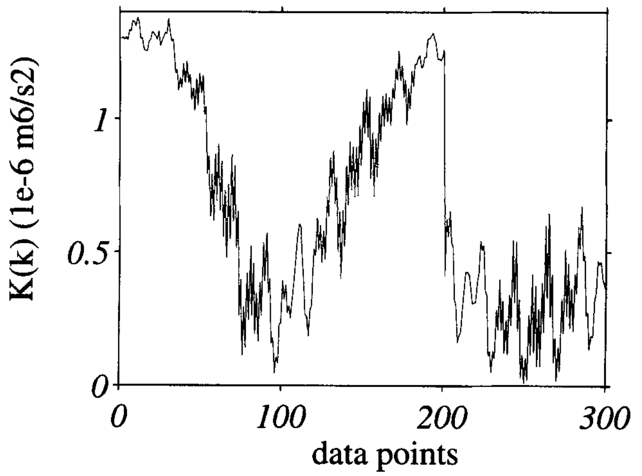

Figure 8. $K(k)$ values for identification of the neural network (Figure 6d) and polynomial (Figure 6c) in the serial configurations.

Table 1. Procedure of Selecting Terms in Polynomial (Eq. 4) of the Serial Gray Box Model (Figure 6c)

Term	SSE	Accepted
$u_1(k)+y(k)+1$	1.00	yes
$u_1(k)^2$	1.03	no
$u_1(k)*y(k)$	0.34	yes
$y(k)^2$	0.50	no
$u_1(k)^3$	1.54	no
$u_1(k)^2*y(k)$	1.37	no
$u_1(k)*y(k)^2$	0.32	yes
$y(k)^3$	0.22	yes
$u_1(k)^4$	2.09	no
$u_1(k)^3*y(k)$	2.14	no
$u_1(k)^2*y(k)^2$	0.78	no
$u_1(k)*y(k)^3$	2.39	no
$y(k)^4$	1.06e4	no

SSE values relative to first one.

Figure 8, were calculated from the original identification data shown in Figure 5b.

First of all, a polynomial model is used to approximate $K(k)$. The resulting serial gray box model configuration is given in Figure 6c. The polynomial function had the following structure.

$$K(k)=a_0+a_1\cdot y(k)+a_2\cdot u_1(k)+a_3\cdot u_1(k)\cdot y(k)$$
$$+a_4\cdot u_1(k)\cdot y(k)^2+a_5\cdot y(k)^3 \quad (4)$$

The parameters a could be calculated by least squares. This structure for the polynomial was determined by using the SISO free run test as a cross validation test. This is comparable with the way the number of hidden nodes of the neural network was determined. Polynomials in ascending order of model complexity were used in the serial configuration and tested in the SISO free run test. In contrast to a neural network, not all possible additional terms are the same, when a more complex polynomial is considered. Therefore, the order in which the terms are considered could matter. To cope with this, Eq. 3 was taken as a first model for the polynomial. After that, higher-order terms were considered one by one, starting with second-order terms, followed by third-order terms, and so on. Within possible terms of a certain order, the term that had the highest correlation with the output was considered first. The term was added to the polynomial, when the SSE in the SISO free run test was smaller than the smallest SSE (of the best model) until that moment. In Table 1 the result of this procedure is displayed. It can be seen all fourth-order terms were rejected; so, no higher-order terms were considered.

Secondly, a feedforward neural network model was considered for the approximation of $K(k)$. In Figure 6d the resulting serial gray box model configuration is displayed. The inputs and outputs for the neural network were directly available (Figure 8), so existing training algorithms could be applied straightforwardly. Neural networks varying in number of hidden nodes from 1 to 30 were trained using the fast noniterative algorithm described in Appendix A. All these configurations were used in the serial configuration and tested for their performance in the SISO free run test. A neural network with 10 hidden nodes was chosen. In Figure 9 it can be seen that only a neural network with 19 neurons in the hidden layer had a slightly smaller SSE. This neural network was not chosen because 36 additional parameters were needed to obtain this small improvement.

Parallel gray box model

In the parallel gray box configuration a neural network was used to model the difference between the measured pressure $y(k+1)$ and the pressure estimated by the first principles model $y_{fp}(k+1)$. The first principles model consisted of the macroscopic balance (Eq. 2) and the linear equation for $K(k)$ (Eq. 3). In Figure 6e the resulting parallel gray box model configuration is displayed. The valve position $u_1(k)$) and the pressure ($y(k)$) were chosen as inputs for the neural network, because the gas-flow rate (3.75e-4 m^3/s) and head space volume (0.015 m^3) were constant during the identification. For training, the output of the neural network could be calculated by subtracting the first principles model prediction

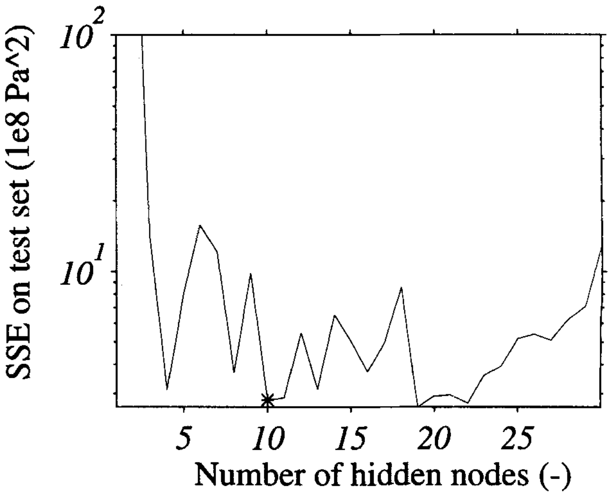

Figure 9. Selection of the number of hidden nodes for the neural network in the serial configuration (Figure 6d).

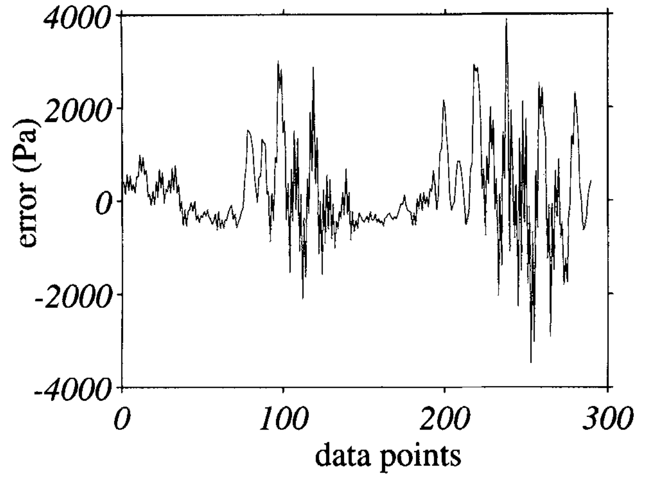

Figure 10. Error(*k*) values for the identification of the neural network in the parallel configuration (Figure 6e).

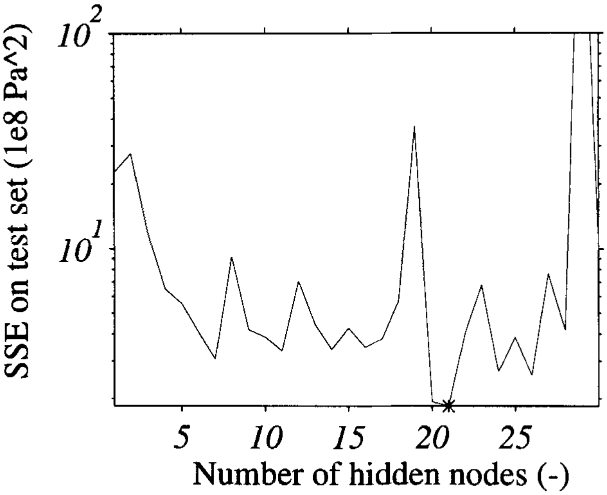

Figure 11. Selection of the number of hidden nodes for the neural network in the parallel configuration (Figure 6e).

(using Eqs. 2 and 3) from the measured pressure (Figure 5b). This way the inputs and outputs of the neural network were directly available and existing training algorithms could be applied straightforwardly. In Figure 10 the signal for identification of the neural network part in the parallel configuration is displayed.

Neural networks varying in number of hidden nodes from 1 to 30 were trained using the training algorithm explained in Appendix A. All trained neural networks were used in the parallel configuration and tested for their performance in the SISO free run test. In Figure 11 it can be seen that a neural network with 21 neurons in the hidden layer had the smallest SSE in the SISO free run test. Therefore, this neural network was chosen as part of the parallel gray box model.

MISO black box neural network reference model

In order to have a proper reference for the dimensional extrapolation properties of the gray box models in the MISO free run test, a black box neural network was used that was especially identified for the MISO mode. It is obvious that the SISO black box neural network reference model could not be used for this purpose, because it was identified at constant gas flow rate while in the MISO mode the gas-flow rate was varied. Therefore, the MISO black box neural network reference model was identified with a data set generated with a varying valve position and a varying gas-flow rate (MISO mode). Details about the identification of this neural network for the MISO mode can be found in Van Can et al. (1995a). The MISO black box neural network model configuration is displayed in Figure 6f. It is again stressed that the gray box models were identified with only 300 data points containing only a varying valve position and constant gas flow rate (SISO mode) and that the MISO black box reference model is identified with 750 data points containing a varying valve position and a varying gas-flow rate (MISO mode). This means that the MISO black box neural network reference model is interpolating in the MISO free run test, but that the gray box models are indeed extrapolating in the MISO-free run test. Therefore, the performance of the MISO black box neural network model is considered to be a reference that should be met as close as possible by the candidate gray box models.

Results Model Comparison

SISO free run test (interpolation)

In the SISO free run test the models were tested for their interpolation properties, because the gas-flow rate was kept constant at the value that was used for generating the identification data (3.75e-4 m^3/s). The input signal and the measured output signal for the SISO free run test are given in Figure 12. The SISO free run test was part of the identification procedure for the different models, as explained above. The SSE values for the SISO free run test of the models are given in Table 2. The SSE values are relative to the SSE for the SISO black box neural network reference model (SSE = 1). As expected, the first principles model had the worst performance. It was already mentioned that this is accepted and that the main point is to see how the gray box models can improve the performance of the first principles model. Therefore, the performance of the SISO black box neural network reference model (Figure 6b) and the performances of the three gray box models (Figures 6c, 6d and 6e) are displayed and compared in Figure 13, in which the predicted values are plotted against the measured values. Kell and Sonnleitner (1995) recently suggested that this is a good way to evaluate the performance of a model. In addition, in Figure 14 the predicted time series for the serial gray box model containing a neural network (Figure 6d) and for the SISO black box neural network reference model (Figure 6b) are compared with the measured pressure time series for the SISO free run test. It is stressed that the predicted values in the free run test (Figures 13 and 14) were obtained by recursive prediction. From Table 2 and Figure 13, it can be concluded that the differences between the different models are not great in this interpolation test. The performance of the serial configuration containing a polynomial model (Figure 6c) deviated the most from the performance of the SISO black box neural network reference model (Figure 6b). Its SSE in the SISO free run test is twice as big as the SSE of serial

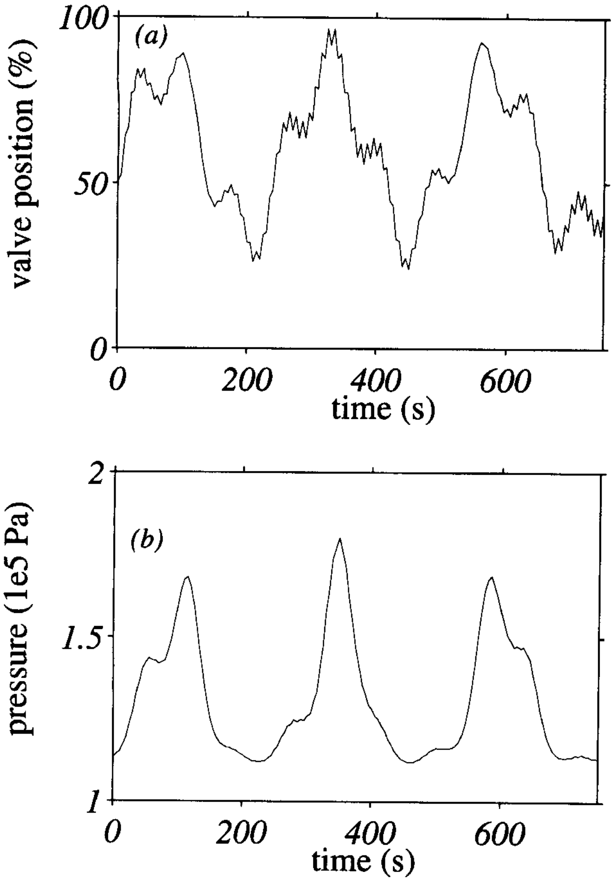

Figure 12. Data used for the SISO free run test.

(a) Input signal; (b) resulting measured output signal.

model containing the neural network model (Figure 6d). The performance of the parallel model (Figure 6e) was comparable very much with the performance of the SISO black box neural network reference model (Figure 6b).

MISO-free run test (dimensional extrapolation)

The performance of the models in the MISO free run test indicates the ability of the models to extrapolate to a situation in which the gas-flow rate deviated from the value applied during model identification (3.75e-4 m^3/s). The input signal and measured output signal for the MISO free run test are displayed in Figure 15. The gas-flow rate reached values between 0.5e-4 m^3/s and 5.0e-4 m^3/s, while for identification of the gray box models (Figures 6c, 6d and 6e) the gas-flow rate was constant at 3.75e-4 m^3/s. In Table 2 the SSE values for the MISO free run test are given. The SSE values are relative to the SSE value of the MISO black box neural network reference model that was identified especially for the MISO mode (Figure 6f). In Figure 16 the predicted values are plotted against the measured values for the three gray models (Figures 6c, 6d and 6e) and the MISO black box neural network reference model (Figure 6f). In addition, in Figure 17 the predicted time series for the serial gray box model containing a neural network (Figure 6d) and for the MISO

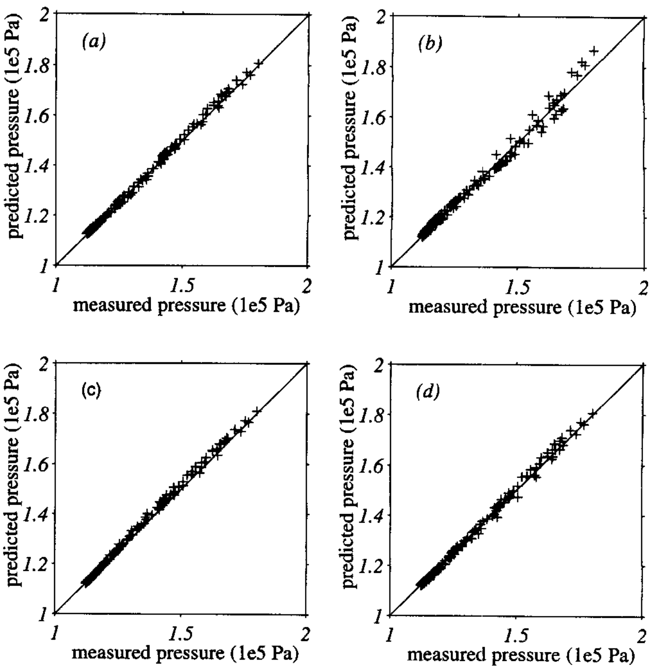

Figure 13. Performance of the different models in the SISO free run test (R^2 = correlation coefficient).

(a) SISO black box neural network reference model (Figure 6b). $R^2 = 0.999$; (b) serial gray box model with polynomial (Figure 6c). $R^2 = 0.994$; (c) serial gray box model with neural network (Figure 6d). $R^2 = 0.999$; (d) parallel gray box model (Figure 6e). $R^2 = 0.997$.

black box neural network reference model (Figure 6f) are compared with the measured pressure time series for the MISO free run test, for which only the last part is displayed. The performance of the serial configuration containing a neural network model for the friction factor (Figure 6d) was closest to the performance of the MISO black box neural network reference model that was especially identified for

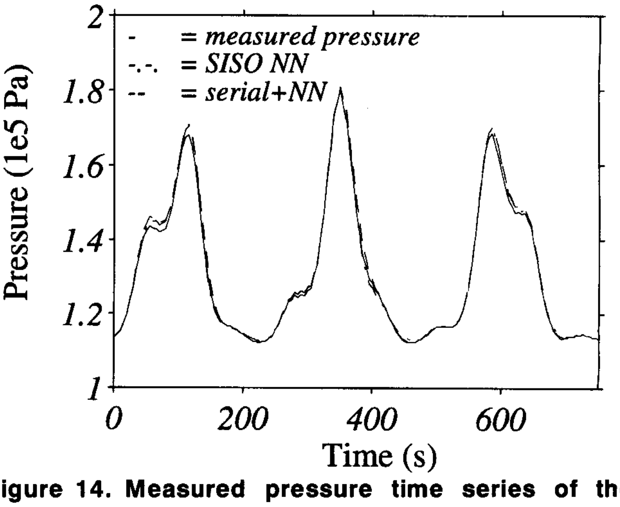

Figure 14. Measured pressure time series of the SISO-free run test vs. predicted time series of the serial gray box model containing a neural network (Figure 6d) and SISO black box neural network reference model (Figure 6b).

Table 2. Performance of Different Models in Free Run

Model	Figure 6	SSE** in SISO Free Run Test (Interpolation)	SSE** in MISO Free Run Test (Dimensional Extrapolation)
First principles model (Eq. 2 + Eq. 3)	(a)	19.3	15.6
SISO black box neural network reference model	(b)	1	–
Serial gray box (Eq. 2 + polynomial)	(c)	4.15	8.02
Serial gray box (Eq. 2 + neural network)	(d)	1.79	1.85
Parallel gray box (Eq. 2 + Eq. 3 + neural network)	(e)	1.17	31.8
MISO black box neural network reference model	(f)	–	1*

*From Van Can et al. (1995a).

**Given numbers are SSE values relative to SSE for the reference models.

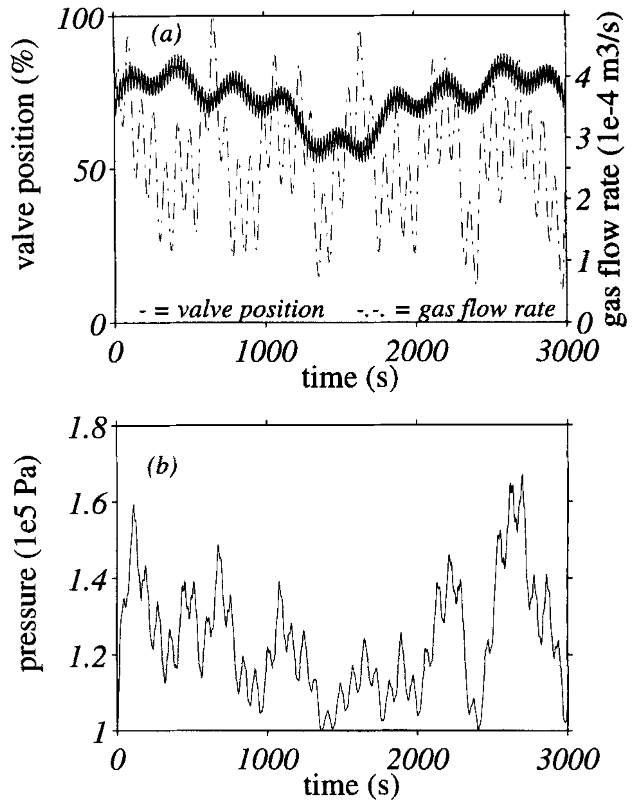

Figure 15. Data used for the MISO free run test.

(a) Input signal; (b) resulting measured output signal.

the MISO mode (Figure 6f). It is again stressed that this serial gray box model (Figure 6d) was identified with only 300 data points, containing only a varying valve position and constant gas-flow rate (SISO mode) and that the MISO black box neural network reference model (Figure 6f) was identified with 750 data points containing a varying valve position and a varying gas-flow rate (MISO mode). The SSE of the serial configuration containing a polynomial model (Figure 6c) was 4.3 times bigger than the SSE of the serial configuration containing a neural network model (Figure 6d). In Figure 16d it can be seen that the ability of the parallel model (Figure 6c) for dimensional extrapolation to the MISO mode is very limited.

Controller test

In the controller test the best model is tested for its ability to guide the system in MISO mode stepwise from one steady state to another steady state. In Table 2, it is seen that the gray box models performed more or less equally well with respect to interpolation and that the serial configuration containing the neural network model (Figure 6d) had the best performance of the gray box models with respect to dimensional extrapolation to the MISO mode. Therefore, this model was used in a MBPC for controlling the system in the MISO mode. In Figure 18 the resulting MISO controller behavior can be seen. The controller was very capable of changing the pressure from 1.2e5. Pa to 1.8e5 Pa and back, using both the valve position and the gas-flow rate as manipulated variables. In order to underline the advantage of the serial gray box approach for this specific application, a disturbance was deliberately added. The vessel was filled with 0.035 m^3 water instead of the 0.025 m^3 that was used during identification, resulting in a threefold reduction of the head space of the vessel from 0.015 m^3 to 0.005 m^3. This meant that the pressure changed much faster due to possible controller actions. When this disturbance was accounted for in the serial gray box model (the head space V_h was changed to 0.005 m^3 in Eq. 2), this resulted in an acceptable performance of the controller, as seen in Figure 19. When this disturbance was not accounted for in the model (the head space V_h was not changed to 0.005 m^3 in Eq. 2), this resulted in a highly unstable controller, as seen in Figure 20. This stresses the advantage of a first principles part in the serial gray box model. Although one would still like to improve the controller behavior (e.g., by a different tuning), the fact remains that the improvement in controller behavior in an extrapolated region is only possible because an adequate first principles model is part of the serial gray box model. If a MISO black box neural network model was used in the controller, this would not have been possible, because in a black box neural network there is no way to account for the changed head space.

Discussion

In the gray box serial configuration a polynomial and a neural network were used to describe the friction factor $K(k)$ as a function of the valve position $[u_1(k)]$ and the actual pressure $[y(k)]$. In the presented case the serial configuration with the neural network had a better performance, as can be seen in Table 2, and by comparing Figures 16b (polynomial) and 16c (neural network). It is pointed out again here that the parameters of the neural network were calculated by a very fast noniterative algorithm. It was found that the effort in-

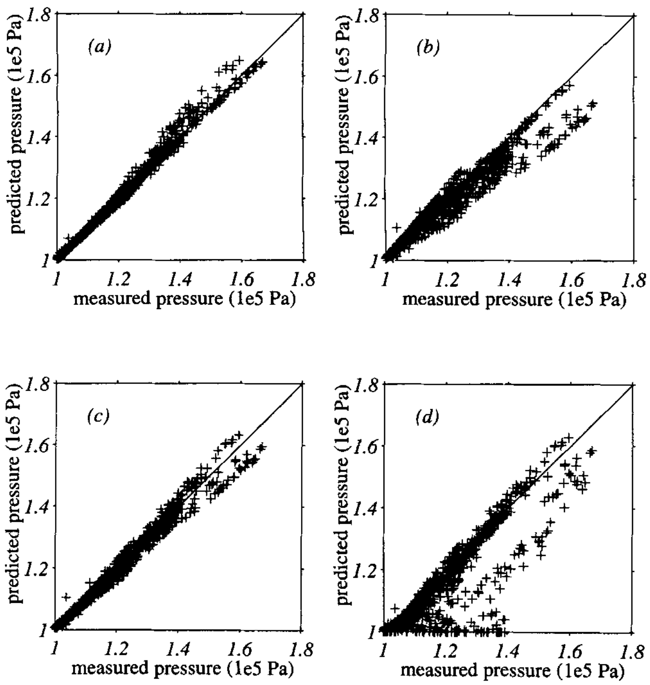

Figure 16. Performance of the different models in the MISO free run test (R^2 = correlation coefficient).

(a) MISO black box neural network reference model (Figure 6f). $R^2 = 0.993$; (b) serial gray box model with polynomial (Figure 6c). $R^2 = 0.962$; (c) serial gray box model with neural network (Figure 6d). $R^2 = 0.987$; (d) parallel gray box model (Figure 6e). $R^2 = 0.838$.

volved in calculating the parameters of a given neural network model is comparable very much with the effort involved in calculation of the parameters of a given polynomial model. In this respect, there is no advantage of one approach over the other. In determining the number of hidden nodes and determining the terms of the polynomial model, there is a difference. Evaluating models in ascending order of complexity is trivial for neural networks, because each time there is only one possible additional term (a hidden node). For a polynomial model this is not trivial, because each time a new term is considered, there is more than one possible additional term. Especially, when the order of the polynomial is high and when there are many inputs, this complicates the selection of the appropriate terms in the polynomial. This was already pointed out by Warnes et al. (1996).

The computational speed of the fast noniterative training algorithm for neural networks was essential in the procedure to select the number of hidden nodes. This made it possible to calculate the parameters of many possible configurations, varying in number of hidden nodes, within a reasonable amount of time (5–10 min). Given the right number of hidden nodes, other training algorithms like back propagation might also give acceptable results, but the time involved in calculating the parameters of various configurations makes the use of iterative algorithms like back propagation less attractive. The used procedure to select the number of hidden nodes is heuristic and does not provide a way beyond any doubt to select the best number of hidden nodes. However, it proved to be a convenient procedure to select a good candidate model. From the presented case, it can, of course, not be concluded that neural networks outperform polynomial models in all cases, but the presented approach for neural networks is very easy to use. Combined with the general

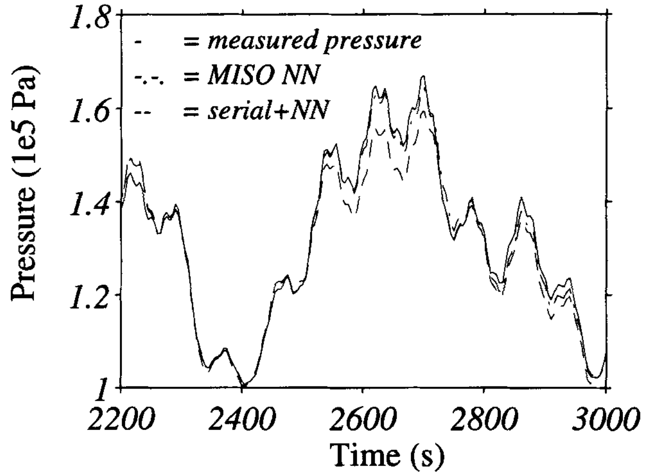

Figure 17. Measured pressure time series of the MISO-free run test vs. predicted time series of the serial gray box model containing a neural network (Figure 6d) and MISO black box neural network reference model (Figure 6f).

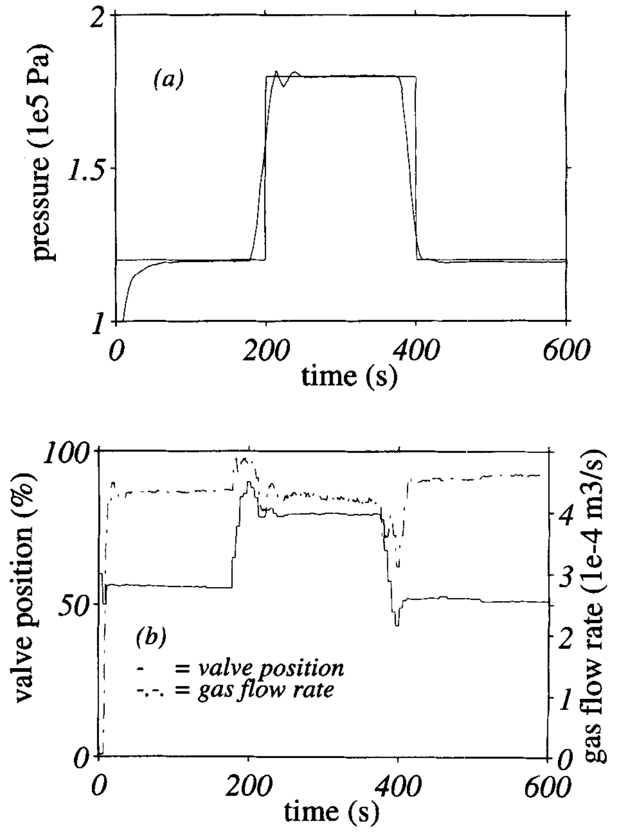

Figure 18. Performance of the controller for the MISO mode.

Controller is based on the serial gray box model with neural network (Figure 6d). Head space not changed. (a) Controlled variable; (b) manipulated variables.

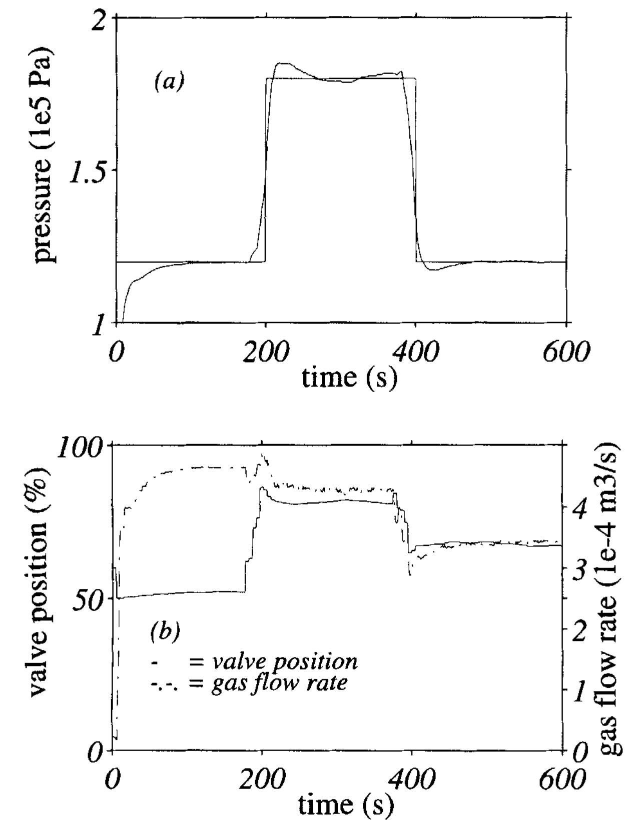

Figure 19. Performance of the controller for the MISO mode.

Controller is based on the serial gray box model with neural network (Figure 6d). Changed head space accounted for in the first principles model. (a) Controlled variable; (b) manipulated variables.

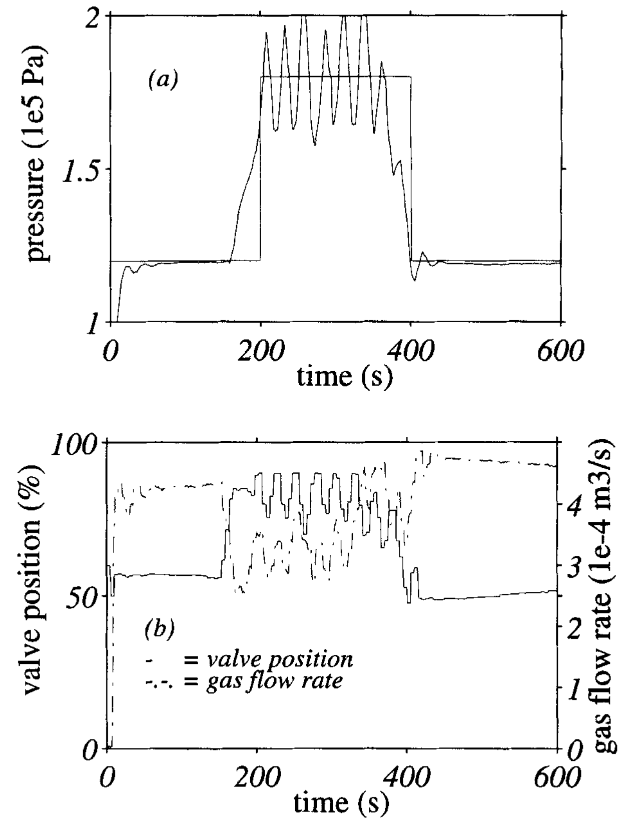

Figure 20. Performance of the controller for the MISO mode.

Controller is based on the serial gray box model with neural network (Figure 6d). Changed head space not accounted for in the first principles model. (a) Controlled variable; (b) manipulated variables.

function approximation properties of neural networks, this makes neural networks a very practical modeling tool for the black box part in a gray box model configuration.

From Table 2 it is seen that the two serial gray box models had much better dimensional extrapolation properties than the parallel gray box model. The difference in performance between the serial and parallel gray box configurations can be understood by looking more closely at the way knowledge is incorporated in the two configurations. The serial and parallel configuration were both based on the same first principles model. In the serial configuration the weak point of the model [friction factor $K(k)$] is recognized as a parametric uncertainty, and the configuration specifically aims at dealing with that problem by modeling $K(k)$ with a neural network. However, in the parallel configuration the knowledge of this weak point of the first principles model is not exploited: the modeling of the error should meet structural as well as parametric uncertainties. Therefore, for the case of the pressure vessel the serial configuration is more suitable, because the model uncertainty can indeed be recognized as a parametric uncertainty.

For the presented case, the parallel configuration showed no advantage over the black box neural network. Su et al. (1992) and Côte et al. (1995) did report improved model predictions of the parallel configuration as compared to the black box configuration. Both cases were concerned with improved interpolation in the identification data. At this moment, no example of improved extrapolation using the parallel configuration is known. For the more general case, the parallel configuration might be more fit to improve interpolation between scarce identification data than providing extrapolation properties outside the domain of the identification data. The performance of the parallel configuration depends very much on the information content of the signal that is used to calculate the parameters of the neural network. This signal should contain all information, preferable to provide predictive power outside the domain of the identification data. Su et al. (1992) already recognized the key importance of the information content of the error signal when using the parallel configuration. In our case the error signal was more than just noise, because for the SISO mode the parallel gray box model was indeed an improvement compared to the first principles model. However, no reasonable prediction is achieved when the model has to perform dimensional extrapolation to the MISO mode (MISO-free run test). This is probably the main drawback of the parallel gray box configuration: there is no strategy for the choice of identification experiments that will yield a training signal with a sufficiently high information content for the identification of the neural network part, especially when extrapolation properties are required.

In our case the serial gray box configuration was successful, even showing reliable predictions far outside the region of the identification data (MISO free run test and controller test). Of course, the correctness of the structure of the first principles model is of key importance, because all model inaccuracies are expected to be parametric. The question for the more general case is whether or not suitable first principles models are available. It was already argued that the serial configuration can be combined very well with the concept of macroscopic balances in (bio)chemical processes. In the macroscopic balance the different terms (such as incoming flows, outgoing flows, and transport between phases and conversion) are combined in a linear way. The inaccurate known terms, which can have a nonlinear relation between the inputs and output like conversion kinetics and friction factors, can be modeled by a neural network. The accurately known terms like convective and diffusive transport can be used directly. The identification data must only cover the input-output space of the neural network part instead of the input-output space of the complete system.

The presented case is a good example for this approach. The used momentum macroscopic balance contained two terms. The term associated with the outgoing air flow was not known accurately due to the friction parameter K, which depended in a nonlinear way on the valve position and the pressure inside the vessel. Therefore, this parameter was modeled by a neural network. The limited set of identification data covered the input-output space of the parameter $K(k)$ sufficiently so that no extrapolation on $K(k)$ was needed when the model was used in the different tests. The term associated with the ingoing air flow was known accurately, because it was a directly manipulated and measured variable. This term could therefore be varied (dimensional extrapolation) when the model was used in the MISO free run test and in the controller test, while it was kept constant during identification. Moreover, the effect of the head space was incorporated in the first principles model, so that the serial gray box models could also be extrapolated in that dimension when the model was used in the controller test. Therefore, based on the structure of the serial gray box model and the available data, one can understand the performance of the model outside the domain of the identification data, because the dimensional extrapolation relies only on the accurately known terms in the macroscopic balance.

In our case it was possible to calculate the parameter $K(k)$ from the original measured identification data. This way the input [$y(k)$ and $u_1(k)$] and output [$K(k)$] at every time instant k were directly available for training the neural network part. Confronted with given identification data, this will not always be possible, especially when more than one inaccurately known term in a macroscopic balance has to be modeled by a black box model. However, if identification experiments can be chosen freely, it is generally possible to design experiments in such a way that the mechanisms associated with the inaccurately known terms are dominantly present in the experiments. Therefore, in contrast to the parallel configuration, the serial configuration does provide a strategy for the design of identification experiments that will yield a training signal with a sufficiently high information content for the identification of the selected neural network part.

Conclusions

In this article a serial gray box modeling strategy was applied successfully for dynamic modeling and control of a real-time pressure vessel. In this strategy a neural network was used to model the inaccurately known term of a macroscopic balance, and the identification data covered only the input-output space of the inaccurately known term. For the presented case, the successful performance of the model far outside the domain of the identification data could be understood, because the needed dimensional extrapolation relied only on the accurately known terms in the macroscopic balance. It is argued that the proposed strategy can be applied for a wide range of (bio)chemical processes, because in (bio)chemical engineering dynamic models can always be based on macroscopic balances and often information is available to choose the inaccurate terms.

The number of hidden nodes of the neural network could be determined easily and quickly, because a noniterative training algorithm was used. Combined with the general function approximation properties of neural networks, this makes neural networks a very practical modeling tool for the black box part in a gray box model configuration.

Acknowledgment

The authors want to thank A. J. Krijgsman, J. M. Schuffelen and H. B. Verbruggen for valuable discussion.

Literature Cited

Bird, R. B., W. E. Steward, and E. D. Lightfoot, *Transport Phenomena*, Wiley, New York (1960).

Côte, M., B. P. A. Grandjean, P. Lessard, and J. Thibault, "Dynamic Modelling of the Activated Sludge Process: Improving Prediction using Neural Network," *Wat. Res.*, **29**, 995 (1995).

Cybenko, G., "Approximating by Superposition of Sigmoidal Function," *Math. Control, Signals Syst.*, **2**, 303 (1989).

Garcia, C. E., D. M. Prett, and M. Morari, "Model Predictive Control: Theory and Practice—a Survey," *Automatica*, **25**, 335 (1989).

Hunt, K. J., D. Sbarbaro, R. Zbikowski, and P. J. Gawthrop, "Neural Networks for Control Systems—A Survey," *Automatica*, **28**, 282 (1992).

Joerding, W. H., and J. L. Meador, "Encoding a priori Information in Feedforward Network," *Neural Networks*, **4**, 847 (1991).

Kell, D. B., and B. Sonnleitner, "GMP—Good Modelling Practice: an Essential Component of Good Manufacturing Practice," *TIBTECH*, **13**, 481 (1995).

Ljung, L., *System Identification: Theory for the User*, Prentice Hall, Englewood Cliffs, NJ (1987).

Mavrovouniotis, M. L., and S. Chang, "Hierarchical Neural Networks," *Computers chem. Eng.*, **16**, 347 (1992).

Psichogios, D. C., and L. H. Ungar, "A Hybrid Neural Network First Principles Approach to Process Modeling," *AIChE J.*, **38**, 1499 (1992).

Schubert, J., R. Simutis, M. Dors, I. Havlik, and A. Lübbert, "Bioprocess Optimization and Control: Application of Hybrid Modelling," *J. Biotechnol.*, **35**, 51 (1994).

Su, H.-T., N. Bath, P. A. Minderman, and T. J. McAvoy, "Integrating Neural Networks with First Principles Models for Dynamic Modeling," *IFAC Symp. on Dynamics and Control of Chemical Reactors*, 327 (1992).

te Braake, H. A. B., and G. van Straten, "Random Activation Weight Neural Net (RAWN) for Fast Non-iterative Training," *Eng. Applic. Artif. Intell.*, **8**, 71 (1995).

te Braake, H. A. B., H. J. L. van Can, G. van Straten, and H. B. Verbruggen, "Two Step Approach in Training of Regulated Activation Weights Neural Networks," *Internal report R95.043*, Control Laboratory, Dept. of Electrical Engineering, Delft Univ. of Technology (1995).

te Braake, H. A. B., H. J. L. van Can, G. van Straten, and H. B. Verbruggen, "Regulated Activation Weight Neural Networks

(RAWN)," *Proc. Euro. Symp. on Artificial Neural Networks*, Brugge, Belgium, p. 19 (1996).

Thompson, M. L., and M. A. Kramer, "Modeling Chemical Processes using Prior Knowledge and Neural Networks," *AIChE J.*, **40**, 1328 (1994).

Van Can, H. J. L., H. A. B. te Braake, C. Hellinga, K. Ch. A. M. Luyben, and J. J. Heijnen, "Neural Models in Predictive Control," Preprints of *IFAC Int. Conf. on Computer Appl. in Biotechnol.*, 95 (1995a).

Van Can, H. J. L., H. A. B. te Braake, C. Hellinga, A. J. Krijgsman, H. B. Verbruggen, K. Ch. A. M. Luyben, and J. J. Heijnen, "Design and Real Time Testing of a Neural Model Predictive Controller for a Nonlinear System," *Chem. Eng. Sci.*, **50**, 2430 (1995b).

Warnes, M. R., J. Glassey, G. A. Montague, and B. Kara, "On Data-Based Modelling Techniques for Fermentation Processes," *Process Biochem.*, **31**, 147 (1996).

Widrow, B., D. E. Rumelhart, and M. A. Lehr, "Neural Networks: Applications in Industry, Business and Science," *Comm. of the ACM*, **37**, 93 (1994).

Appendix A: Noniterative Training Algorithm

Feedforward neural network

In this article only three layer feedforward neural networks are considered. The first layer is the input layer. This layer is in fact a layer without neurons; it only directs the input to every neuron in the next layer, which is called the hidden layer. This hidden layer contains neurons with a nonlinear activation function. The last layer is the output layer, which is built like the hidden layer, but using a linear activation function.

Every input of a neuron in the hidden layer is multiplied with an activation weight W_{ij}^h. These multiplied inputs are added together with a bias input. This is expressed by

$$z_j(k) = \sum_{i=1}^{N_i} W_{ij}^h \cdot x_i(k) + b_j^h \tag{A1}$$

The output of the neuron of the hidden layer is then given by

$$v_j(k) = f(z_j(k)) \tag{A2}$$

The output of the complete neural network is given by

$$y_l(k) = \sum_{j=1}^{N_h} W_{jl}^o \cdot v_j(k) + b_l^o \tag{A3}$$

Subscript j denotes the jth neuron in the hidden layer, subscript i denotes the ith input, and l denotes the lth output. Index k denotes the kth event. N_e is the number of events, N_i the number of inputs, and N_o the number of outputs. N_h is the number of neurons in the hidden layer. The weights W_{ij}^h are called the activation weights and the weights W_{jl}^o the output weights. The activation bias is denoted as b^h and the output bias as b^o. The function $f(.)$ is the activation function. This function can be any sigmoidal or Gaussian function. In this article the tanh-function is considered.

The input vectors and the bias vector can be grouped into a matrix X, with one row for each event and one column for each output (including a column with ones expressing the biases)

$$X(k) = [x_1(k) \dots x_i(k) \dots x_{N_i}(k) \quad 1]$$
$$X = [X(1) \dots X(k) \dots X(N_e)]^T \tag{A4}$$

Similarly, the output vectors can be grouped into a matrix Y

$$Y(k) = [y_1(k) \dots y_l(k) \dots y_{N_o}(k)]$$
$$Y = [Y(1) \dots Y(k) \dots Y(N_e)]^T \tag{A5}$$

Then the neural net can be expressed concisely as

$$Z = X \cdot W^h [N_e \times N_h]$$
$$V = f(Z) \quad [N_e \times N_h]$$
$$Y = V_b \cdot W^o [N_e \times N_o] \tag{A6}$$

With input matrix $X \epsilon R^{Ne \times [Ni+1]}$, and the output matrix $Y \epsilon R^{Ne \times No}$. Matrix Z and V contain intermediate results. Matrix $V_b \epsilon R^{Ne \times (Nh+1)}$ is equal to $V \epsilon R^{Ne \times Nh}$ except that one column with ones is added to express the output bias b^o. The activation weights are grouped together in a matrix $W^h \epsilon R^{[Ni+1] \times Nh}$, and the output weights in a matrix $W^o \epsilon R^{[Nh+1] \times No}$.

The training problem is split into two subproblems which each can be solved optimally: the estimation of W^o and the estimation of W^h. This does not mean that the total estimation problem is solved optimally. The original idea of the training algorithm is described in te Braake and Van Straten (1995). A more thorough treatment of the method can be found in te Braake et al. (1995).

Estimation of W^o

To obtain W^o, first assume that the weights W^h are already known, and therefore V_b is known as well. Suppose that the true output can be modeled by

$$Y = V_b W^o + e \tag{A7}$$

The vector e denotes the modeling error. A parameter estimation remains which is linear in the parameters. By minimizing the sum of squared modeling errors, the well-known least-squares estimation of W^o becomes (Ljung, 1987)

$$W^o = [V_b^T V_b]^{-1} V_b^T Y \tag{A8}$$

The matrix $V_b^T V_b$ must be nonsingular, otherwise $[V_b^T V_b]^{-1}$ would not exist. This implies that V_b must have rank $N_h + 1$.

Estimation of W^h

The estimation of the activation weights W^h is based on the assumption that a nonlinear function can be approximated by a sequence of linear functions based on a linear least-squares estimation of the parameters of the linear part. The parameters in these equations then can be used as the elements of W^h. Thus, each neuron is assigned to a specific subset of the complete data set and the network is built by sequentially assigning a neuron to each subset. The number of neurons is the same as the number of subsets under consideration.

Suppose a feedforward neural network has to fit a certain (static) function $y = f(x)$. Then, a data set Φ can be created with measured input/output pairs, or

$$\Phi = \begin{bmatrix} x_1(0) & \cdots & x_{N_i}(1) & y(1) \\ \vdots & \vdots & \vdots & \vdots \\ x_1(k) & \cdots & x_{N_i}(k) & y(k) \\ \vdots & \vdots & \vdots & \vdots \\ x_1(N_e) & \cdots & x_{N_i}(N_e) & y(N_e) \end{bmatrix} = [X \quad Y] \qquad (A9)$$

The index k denotes the kth event. By partitioning the matrix Φ into N_h subsets, it is possible to construct a linear model for each subset Φ^n. In this article the data pairs were divided straightforwardly over the neurons: the first subset Φ^1 consists of the first N_e/N_h rows of Φ, and so on. The linear model is then given by

$$Y_n = X_n W_n^h + b_n^h \qquad (A10)$$

The weights W_n^h and b_n^h can be calculated with ordinary least squares

$$\begin{bmatrix} W_n^h \\ b_n^h \end{bmatrix} = (X^{n^T} X^n)^{-1} X^n Y^n \qquad (A11)$$

The matrix X^n is supplied with an extra column filled with ones, needed to express the biases b_n^h. Y^n is a scaled matrix in Eq. A11.

Every linear function describes the mapping between input and output for that particular subset n. After calculating N_h linear models, the matrix W^h is built like

$$W^h = \begin{bmatrix} W_1^h & W_2^h & \cdots & W_n^h & \cdots & W_{N_h}^h \\ b_1^h & b_2^h & \cdots & b_n^h & \cdots & b_{N_h}^h \end{bmatrix} \epsilon R^{(N_i+1)\times N_h} \qquad (A12)$$

It should be noted that the output of each linear submodel is transformed by the sigmoid function in the neuron and that the output weights W^o determine the final output of the neural network. Therefore, after the calculation of W^h all data X (Eq. A4) and Y (Eq. A5) are used for the calculation of the output weights W^o (Eq. A8).

Internal scaling of Φ

If the matrix Y is not scaled before it is used in Eq. A11, there is a possibility that the resulting elements of Z (Eq. A6) are so small that the sigmoid function is only operating in the linear range. Alternatively, it is possible that the resulting elements of Z are so large that the function always operates near saturation. In both cases the matrix $V_b^T V_b$ is (close to) singular and W^o cannot be calculated (Eq. A8). Therefore, matrix Y is scaled before it is used in Eq. A11 in order to avoid a badly conditioned $[V_b^T V_b]$ matrix in Eq. A8. In this article the Y matrix in Φ is scaled according to

$$Y_{\text{scaled}} = \frac{\alpha}{y_{\max} - y_{\min}} (Y_{\text{unscaled}} - Y_{\min}) + \beta \qquad (A13)$$

with $y_{\max}$ and $y_{\min}$ being the maximum and the minimum value of Y. $Y_{\min}$ is a vector of the same size as Y and filled with $y_{\min}$ values. In our case $\alpha = 1$ and $\beta = -0.5$. This means that Y in Φ was scaled between -0.5 and 0.5 for the calculation of W^h. It should be noted that the W^h parameters of a neuron are calculated with *only a subset* of the X (and Y) data, but that *all* X data (Eq. A4) are used for the calculation of the elements of Z and V (Eq. A6) before W^o is calculated (Eq. A8). Consequently, the V data (used for the calculation of W^o) are not automatically scaled the same way as Y (used for the calculation of W^h). For example, in the neural network of the serial gray box model ($N_h = 10$ and $N_e = 300$), 450 elements (15%) of the 3,000 V elements were outside the scaled range of Y (-0.5 and 0.5), so that the nonlinear part of the activation function was indeed used. The scaling parameters (α and β) are properly chosen if the matrix $V_b^T V_b$ is nonsingular. The scaling parameters (α and β) can be adjusted in order to optimize the training result, but this was not done in this article. A more thorough treatment of this matter can be found in te Braake et al. (1996).

Appendix B: First Principles Model

The first principles model is based on the ideal gas law

$$p \cdot V = n \cdot R \cdot T \qquad (B1)$$

where p is pressure (Pa), V is volume (m^3), n is number of gas molecules per volume (mol/m^3), R is gas constant (J/mol/K), and T is temperature (K).

The pressure change in the vessel results from the difference in the number of incoming and outgoing gas molecules

$$\frac{dp}{dt} = \frac{RT}{V_h} \cdot \frac{dn}{dt}$$

$$\frac{dp}{dt} = \frac{RT}{V_h} n(\phi_{\text{in}} - \phi_{\text{out}}) \qquad (B2)$$

where V_h is head space of the vessel (m^3), ϕ_{in} is incoming air flow (m^3/s), and ϕ_{out} is outgoing air flow (m^3/s).

The incoming air flow is a manipulated variable and is therefore known. The outgoing air flow is given by

$$\phi_{\text{out}} = Av \qquad (B3)$$

where A is cross-sectional area of the outlet gas pipe (m^2), and v is velocity of outgoing gas (m/s).

The velocity of the outgoing air flow can be calculated from Bernoulli's equation (Bird et al., 1960)

$$\Delta\frac{1}{2}v^2+g\Delta h+\int_{p_0}^{p}\frac{dp}{\rho}+W+\Sigma\frac{1}{2}v^2\frac{L}{R_h}f+\Sigma\frac{1}{2}v^2e_v=0 \quad \text{(B4)}$$

where g is gravitation constant (m/s^2), h is height (m), ρ is density of gas (kg/m^3), L is length of the pipe (m), R_h is hydraulic radius (m), f is friction loss factor due to wall friction, and e_v is friction loss factor due to obstacles, e.g. valve.

For the system under consideration Eq. B4 can be simplified to

$$\int_{p_0}^{p}\frac{dp}{\rho}+\frac{1}{2}v^2e_v=0 \quad \text{(B5)}$$

where

$$\rho=\frac{p}{p_0}\rho_0 \quad \text{(B6)}$$

and p_0 is atmospheric pressure (Pa), and ρ_0 is density of the air at $p=p_0$ (kg/m^3).

This differential Eq. B5 can be solved, resulting in

$$v=\sqrt{2\frac{p_0}{\rho_0e_v}\ln\left(\frac{p}{p_0}\right)} \quad \text{(B7)}$$

A combination of Eqs. B2, B3, and B7 results in

$$\frac{dp}{dt}=\frac{RT}{V_h}39.7\left(\phi_{\text{in}}-\sqrt{2\frac{p_0A^2}{\rho_0e_v}\ln\left(\frac{p}{p_0}\right)}\right) \quad \text{(B8)}$$

The term $(p_0A^2)/(\rho_0e_v)$ can be replaced by an overall unknown parameter K, which is related to the friction. The final model is thus given by

$$\frac{dp}{dt}=\frac{RT}{V_h}39.7\left(\phi_{\text{in}}-\sqrt{K\cdot\ln\left(\frac{p}{p_0}\right)}\right) \quad \text{(B9)}$$

In the article the pressure p will be noted by y (system output) and the incoming gas-flow rate φ_{in} will be denoted by u_2 (system input).

Manuscript received Feb. 14, 1996, and revision received May 20, 1996.