Contents lists available at ScienceDirect

Chemical Engineering Journal

journal homepage: www.elsevier.com/locate/cej

Industrial batch dryer data mining using intelligent pattern classifiers: Neural network, neuro-fuzzy and Takagi–Sugeno fuzzy models

Levente L. Simon*, Konrad Hungerbuhler

ETH Zurich, Institute of Chemical and Bioengineering, Safety and Environmental Technology Group, W. Pauli Str. 10, 8093 Zurich, Switzerland

ARTICLE INFO

Article history:
Received 2 July 2009
Received in revised form 8 January 2010
Accepted 11 January 2010

Keywords:
Neural networks
Neuro-fuzzy
Pattern recognition
Batch dryer
Data analysis

ABSTRACT

This contribution describes the pattern recognition based data analysis of an existing industrial batch dryer, and the comparison of three artificial intelligence techniques suited to perform classification tasks: neural networks trained using the Levenberg–Marquardt and the Levenberg–Marquardt method with Bayesian regularization, the neuro-fuzzy model based on clustering and grid partition, and the Takagi–Sugeno fuzzy models. The classifiers are used to quantify the dryer batch time and its variation during a certain production period, thus the motivation behind the work is genuine. The presented pattern recognition method implements a supervised learning approach and is based on pressure measurement profiles recorded by the plant data management software—the PI System from OSIsoft.

It is found that the neural networks trained with the Bayesian regularization have shown the most robust classification performance with respect to separation threshold selection. Furthermore, it is concluded that the application of artificial intelligence techniques in real chemical manufacturing facilities is feasible and provides useful information for process performance monitoring purposes. The pattern recognition findings presented in this paper are not case specific and can be directly used for the monitoring of a large variety of drying processes since the pressure profile features – vacuum check, pressure decrease, vacuum break – do not depend on the chemicals which are dried. Since the development of the artificial intelligent classifiers is presented in detail and step by step, this work may be interesting as a pattern recognition tutorial for chemical engineers.

1. Introduction

Plant data analysis provides the opportunity to identify operation improvement potentials. Usually the batch-to-batch variations of parameters such as conversion, selectivity, particle size distribution, batch time are major concerns in the batch industry. In order to maximize the production capacity and product quality it is desired that these variations are identified as fast as possible and action is taken to reduce the variations. The quickest way to identify batch-to-batch variations is to analyze the process data trends which are logged during plant operation.

Data recording in the process industries provides huge amount of information which requires adequate tools to be interpreted and analyzed. Modern computer hardware technology together with intelligent software solutions make it possible today to process the large amount of data at low cost. Some well-known analysis methods and tools that are used for data mining are for instance statistics (regression analysis, discriminant analysis, and principal component analysis), time series analysis, decision trees, cluster analysis, neural networks, fuzzy models and neuro-fuzzy models. These approaches are particularly useful when data are abundant and modeling knowledge is missing. Pattern recognition is a useful approach for interpreting the data generated by chemical and biochemical processes [1,2], and involves inductive reasoning through generalization from a set of learned examples of process behavior. The number of pattern recognition applications is enormous and the literature evaluation in this article focuses on the solutions related to process engineering. These cover a wide range of applications: dynamic disturbance classification [3,4], recognition of chemical reactions with similar chemical end economic characteristics based on an existing reaction database [5], real-time classification of petroleum products using near-infrared spectra [6], fault diagnosis of chemical processes [7–11]. While statistical methods may be based on principal component analysis [12,13] and partial least squares (PLS) [14], the black-box model based pattern recognition methods rely on the neural networks [15], fuzzy models [16,17], and neuro-fuzzy [18] models. From the point of view how the data is processed these methods can be classified as supervised [19], un-supervised [20] and semi-supervised [21]. Dynamic process data usually are processed using the shifting temporal window technique which is well-suited to the processing of the time series data [13,19].

* Corresponding author. Tel.: +41 44 6334486; fax: +41 44 6321189.
E-mail address: levente.simon@chem.ethz.ch (L.L. Simon).

1385-8947/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.cej.2010.01.018

Nomenclature

List of symbols

b	neural network bias
f	fuzzy rule or global model output
k	fuzzy rule
K	number of fuzzy rules
n	number of neurons in the layer
P	pressure
p	fuzzy model consequent parameters
w	neural network weight

Greek symbols

μ	fuzzy set degree of membership
ν	ANFIS model weights
Ω	fuzzy labels
ΔP	pressure gradient

The purpose of the article is twofold: it presents the pattern recognition problem and solution for an industrial batch drying process, and it provides a comparison of neural network, Takagi–Sugeno fuzzy and neuro-fuzzy artificial intelligence methods. The development of the data classifier is motivated by the necessity of the batch time variation quantification. While the neural networks and the neuro-fuzzy models are established techniques for pattern recognition applications, this work shows that the Takagi–Sugeno fuzzy models may be also applicable.

The article is structured as follows: after a discussion on the artificial intelligence methods and applications in the process systems engineering field, a short introduction with emphasis on the employed techniques is given. Subsequently, the development of the artificial intelligent decision maker systems is presented, afterwards the classification and comparison results are discussed, and the conclusions are drawn in the last section.

2. Introduction to artificial intelligence techniques

2.1. Artificial neural networks

Artificial neural networks have emerged as important tools for non-linear data mapping. Artificial neural networks with feed-forward structure can be used for static mapping of the input–output data. Process engineering related applications are found in the form of pattern classification applications and modeling of time series using one-step ahead models. The recursive neural networks are used to implement recursive models of time series data [22] and are used for model based control applications [23]. The combination of the artificial neural networks with first-principles models yields the hybrid models [24,25]. The most common training algorithm of the neural networks is the backpropagation method and applications using metaheuristic optimization methods can be found as well. During the training of neural networks particular care is shown towards the generalization capability of the network. One way to ensure this is to implement the "early stopping" technique using an unseen test set. The other option is to add a regularization term to the objective function. Numerous applications have shown that the regularization using the Bayesian framework [26–28] ensures good generalization features and network training may be carried out without a validation set. Furthermore, to avoid overfitting of the data and achieve the best prediction ability with the simplest structure possible, pruning algorithms are used for topology optimization [29]. The main drawback of artificial neural networks is the difficulty to understand the meaning associated with each neuron and weight. Himmelblau presents a recent review on the application of neural networks in chemical engineering [30].

2.2. Takagi–Sugeno fuzzy models

Non-linear process behavior can be modeled by multiple models with smooth or fuzzy transitions. The data interval is delimited into a set of sub-regions for which a sub-model is identified. The most wide spread fuzzy models fall into two categories depending on the type of the consequent part. The Mamdani fuzzy model [31] is based on IF-THEN rules with output membership functions in the form of fuzzy sets. The second type is the Takagi–Sugeno [32] model which is formed by IF-THEN rules with fuzzy antecedent part and a linear function as the consequent part. This model can be interpreted in terms of a collection of local linear models rules of the process and provides the opportunity to introduce a priori knowledge during process model development; e.g. a pH titration curve may be approximated by 3 linear models with fuzzy transitions; therefore, it is expected that a TS fuzzy model using 3 rules is postulated and identified. These features are valuable for control engineers, e.g. for the pH system mentioned above one may consider to tune 3 PI controllers with fixed P and I parameters (valid on each operation sub-domain). The switching between the fixed PI controllers is handled by the fuzzy system and it results in smooth transitions of the PI controller settings between the operation domains. During the last two decades the TS fuzzy models have found applications in many chemical engineering areas: dynamic process modeling using first-principles Takagi–Sugeno hybrid models [33], time series modeling: one-step ahead [34] and recurrent TS models [35] or model based process control applications.

One approach to build fuzzy models is to express the expert knowledge as fuzzy rules [36]. Alternatively, provided that enough data is available, one may develop measurement based fuzzy models. The identification of the TS fuzzy models using input–output data has two steps: structure identification and parameter estimation. One way to identify the structure of a fuzzy model is to perform input–output data clustering using: fuzzy C-means, Gustafson–Kessel clustering and subtractive clustering. The subtractive clustering method performs clustering of the input–output space and considers the data points as potential candidates for cluster centers. According to this clustering method the computation is proportional to the number of data points and independent of the dimension of the problem. Another option to generate fuzzy structures is the partitioning of the input space by using grid, tree or scatter partition. The grid partitioning method is less suited for problems with many inputs or for cases when the input space is not entirely populated by data. In such cases the fuzzy structure will contain fuzzy rules which do not fire, e.g. identification of pH control related models in process engineering. Nevertheless, non-firing rules may be eliminated or merged using pruning and merging procedures [37]. Furthermore, optimized fuzzy model structures can be constructed using a genetic algorithm based approach with variable chromosome length [38]. The next step during the fuzzy model identification is the determination of the optimal consequent parameters, which is posed as a linear least square optimization problem. For more details about the different type of fuzzy clustering, cluster merging and optimal consequent parameter identification of the fuzzy models the reader is referred to the referenced literature [39].

2.3. Neuro-fuzzy models

Combining the learning capabilities of the neural networks with the knowledge representation of fuzzy logic results in adaptive neuro-fuzzy inference systems ANFIS [40]. This system

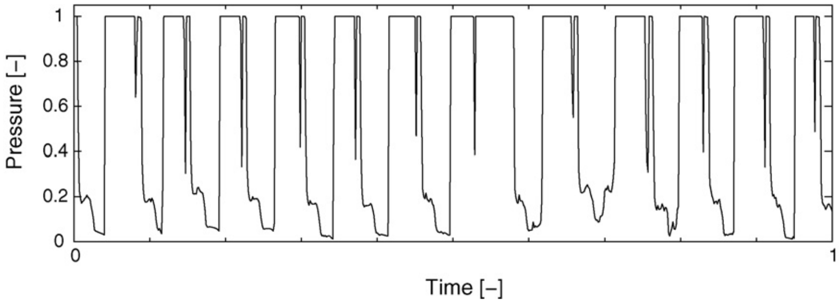

Fig. 1. Dryer pressure profile for selected batches; time and pressure are normalized.

has a five-layer network structure. Initial fuzzy structures can be developed by input space partitioning and by input–output space clustering. In order to perform the supervised training of the neuro-fuzzy systems two methods are common in practice: backpropagation based optimization of all parameters and the hybrid method. The latter implements the backpropagation on the antecedent parameters and later-on performs a linear least-squares estimation of the premise parameters. Fuzzy model structures which were identified based on a clustering algorithm may be further tuned using one of the methods mentioned above.

3. Pattern recognition problem presentation

The aim of this work is to develop an intelligent decision maker which is able to analyze the measurement data of an existing industrial batch dryer. This classifier is needed to quantify the dryer batch time and its variation during a certain production period in order to allow the systematic analysis of the batch plant [41]. The plant data consist of pressure measurements without time tags which could indicate when a new batch is started. In order to calculate the time elapsed between two batches a pattern recognition algorithm is developed using pressure data trends logged by the PI System software [42]. Previous research work has shown that the pattern recognition using data retrieved from the PI System is more difficult [43]. The dryer pressure profile retrieved from the PI System is presented in Fig. 1.

Throughout this work the plant data is normalized due to confidentiality reasons. The presented drying process is characterized by two main operational stages. The first one is the vacuum check, which is performed at the beginning of each batch. The second stage is the drying processes itself during which the pressure has a decreasing trend; after the drying is completed the pressure increases again. The aim of the data analysis is to determine the time elapsed between two vacuum check procedures.

The implemented pattern recognition method uses the concept of the temporal moving window in a simplified form. The size of the temporal window is set to 1 measurement point thus each measurement point is subject to a classification using the decision maker system.

3.1. Inputs and output of the classifier

It is considered that based on the information of the absolute pressure (input 1) and its gradient (input 2) it is possible to uniquely identify the vacuum check stage and its pattern. These two parameters will constitute the input space of the pattern recognition problem, Fig. 2. The size of the data set on which pattern recognition will be performed is about 16,000 measurement points retrieved from the PI System.

3.2. Data pretreatment

This first step during the data pretreatment is the input space data normalization between 0 and 1 for the fuzzy models and −1 and 1 for the neural networks. Data normalization is a necessary step during black-box model development.

The second pretreatment procedure is the input data reduction. The goal of this procedure is to ensure equal data distribution of the input space. The importance of this operation during the development of the Takagi–Sugeno models is also recognized by van Lith et al. [33]. The input data reduction is performed by eliminating data points which are closer to each other than a certain threshold without altering the shape of the patterns. By performing this procedure the consequent part parameter estimation algorithm will learn with equal weights all data points. Due to the input data reduction procedure the data set was reduced to 312 measurement points (vacuum check pattern data together with the other process patterns).

3.3. Pattern membership labeling

The next step is to assign membership values of 1 to the measurements which belong to the vacuum check pattern and value of 0 to the rest of the data, Fig. 3. The data labeling procedure introduces the system specific a priori knowledge and allows the supervised training of models.

3.4. Data classifier calculation

The last step of the classifier development is to find a model which connects the inputs to the output, thus to identify a black-box model. Neural networks and Takagi–Sugeno fuzzy models

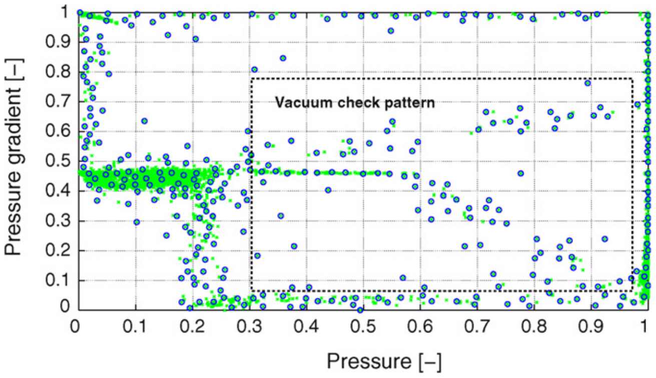

Fig. 2. The input space of the pattern recognition algorithm consists of the normalized absolute pressure and pressure gradient; crosses are original, circles the reduced input data, respectively; the data in the box represents the vacuum check pattern.

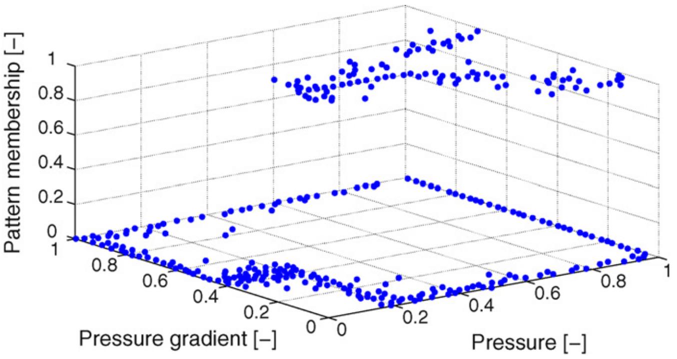

Fig. 3. Normalized and reduced input–output data used for model identification.

are known as universal approximators [44–46]; therefore, it is expected that they are able to match the input–output space, assuming that a model of enough complexity is postulated. Universal approximation theorems state that by using enough model building blocks (layers, neurons in each layer, interconnectivity, fuzzy rules) these models can approximate any real continuous non-linear function to any degree of accuracy. Thus, the challenge is to find the simplest and most robust model, with the least number of parameters, which is suited for the particular application (Occam's razor principle). The classification performance of the model, ideally, should be insensitive towards the value of the classification threshold. The structures of the neural network and ANFIS models developed for this pattern recognition task are presented in Fig. 4. The TS model formulation is presented below:

$$R_k : \text{if } P \text{ is } \Omega_{P,k} \text{ and } \Omega_{\Delta P,k} \text{ then} f_k = p_k^0 + p_k^1 P + p_k^2 \Delta P \qquad (1)$$

where k is the kth fuzzy rule, P and ΔP are inputs, $\Omega_{P,k}$ and $\Omega_{P\Delta,k}$ are the linguistic fuzzy labels for inputs P and ΔP, f_k is the linear output and p_k^0, p_k^1, p_k^2 are the consequent parameters associated with each rule k, respectively. The global output of the fuzzy system f with is the pattern membership value is the weighted average of the individual rule k outputs:

$$f = \frac{\sum_{k=1}^{K} f_k(\mu_k(P) \wedge \mu_k(\Delta P))}{\sum_{k=1}^{K} \mu_k(P) {}^\wedge \mu_k(\Delta P)} \qquad (2)$$

where K is the number of fuzzy rules, μ_k is the degree of membership of the antecedent of rule k and ^ is the minimum operator.

Beside the classification application itself, this article presents the comparison of several artificial intelligence based models, with respect to the pattern recognition performance. The notation of these models, the structure identification and parameter tuning techniques are summarized in Table 1.

In order to analyze the structure size related classification performance we consider two classes of structures (large and small), as presented in Tables 2 and 3. The identified classifiers corresponding to the "large" models are presented in Fig. 5 where it is concluded that the models were able to capture enough information and can be used for classification purposes, as will be shown later. The models show spurious behavior in the range where data is not present—low pressure and low pressure gradient. The modeling residuals for the "large" models are presented in Fig. 6, in which it is observed that for all models the residuals' patterns are similar.

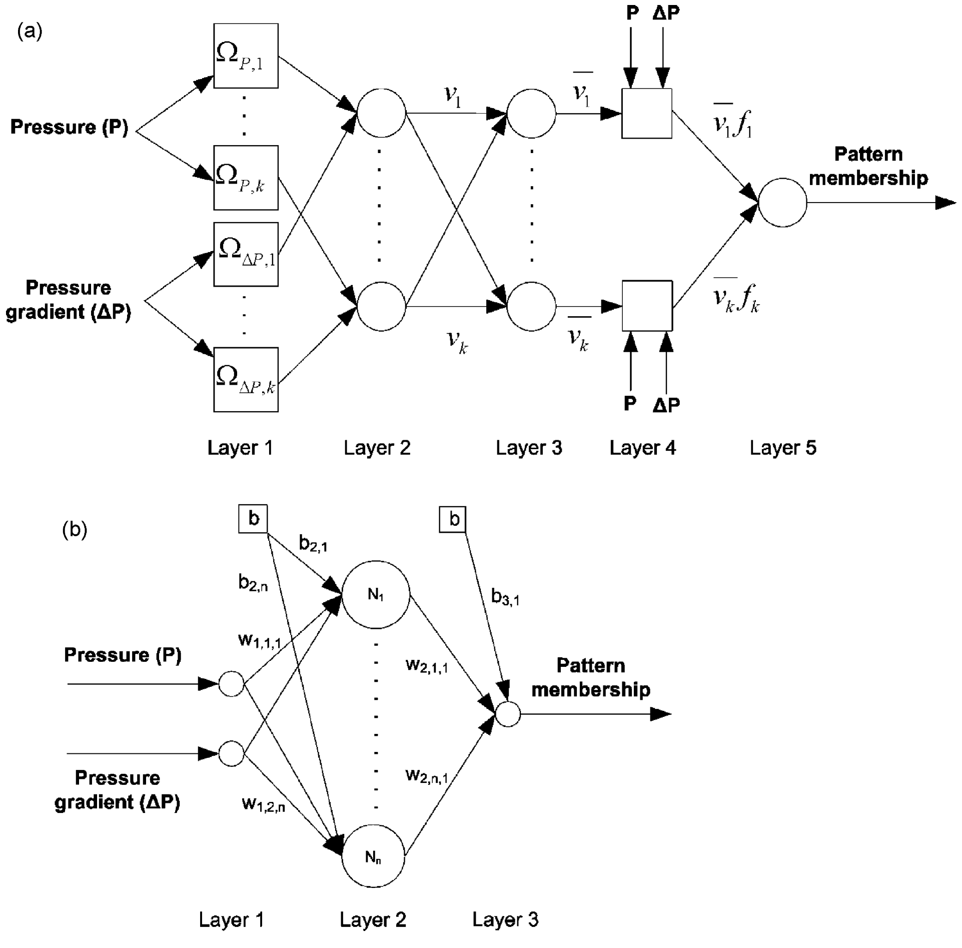

Fig. 4. Structure of the ANFIS (a) and neural network models (b).

Table 1
The artificial intelligence models.

Model type	Takagi–Sugeno fuzzy	Neuro-fuzzy (ANFIS)		Neural network	
Model name	*clustTS*	*gridANFIShyb*	*clustANFIShyb*	*NNlm*	*NNbr*
Structure identification and parameter tuning method	Subtractive clustering and linear least squares	Grid partitioning with backpropagation and linear least squares (hybrid)	Subtractive clustering and linear least squares followed by backpropagation and linear least squares (hybrid)	Levenberg–Marquardt	Levenberg–Marquardt with Bayesian regularization

Table 2
Specifications of the “large” models.

Model	# membership functions for inputs		# rules or neurons	# parameters	RMSE
	P	ΔP			
clustTS	8	8	8	72	0.178
gridANFIShyb	3	2	6	33	0.159
clustANFIShyb	5	5	5	45	0.166
NNbr	–	–	8	33	0.131
NNlm	–	–	8	33	0.109

Table 3
Specifications of the “small” models.

Model	# membership functions for input		# rules or neurons	# parameters	RMSE
	P	ΔP			
clustTS	5	5	5	45	0.278
gridANFIShyb	2	2	4	24	0.234
clustANFIShyb	4	4	4	36	0.183
NNbr	–	–	6	25	0.146
NNlm	–	–	6	25	0.128

Around the normalized pressure values of 0.3 and 0.8 the predicted membership shows significant deviation. The same large deviations are noticed also in the case of the pressure gradient variable. The *NNlm* model was able to match best the data, this is reflected by the RMSE value presented in Table 2.

3.5. The classification procedure

After the development of the classifiers the entire plant data, before input data reduction, was subjected to classification with the calculated models. Most of the data was classified as belonging or not to the vacuum check profile with output 1 or 0. However, a significant number of data had an output value between 0 and 1. This is due to the fact, that the patterns were not defined with perfect accuracy and due to data matching or modeling errors. In order to assign the points with an intermediate membership values to one of the patterns, usually a threshold value is chosen [19]. All the data above the threshold were assigned to the vacuum check, and the data below to another process operation pattern. The membership of each data point to the vacuum check pattern calculated by the proposed classifiers is presented in Fig. 7. It is observed that the classification is successful in the sense that most of the data is clearly segregated into two classes. However, the segregation for the *NNlm* “large” model is less obvious and the selection of the separation threshold is more difficult. The value of the threshold may significantly influence the classification results, as discussed in the next part of the paper.

4. Results and discussion

This section of the paper is composed of three parts: in the first part the classification performances of the “large” and “small” models are assessed, in the second part the different type of black-box models are compared with respect to the root-mean square error (RMSE) value and classification performance. The RMSE value shows discrepancy between the classifier model and plant data. The comparison is focused on the model structure, model size, identification/training algorithm, and on the data classification threshold. The last part of this section deals with the classification performance analysis of several neural networks trained using the Bayesian regularization framework.

4.1. Classification performance

The classification performance for the “large” and “small” models is presented in Table 4. In order to evaluate the classifiers there were two criteria: the number of batches not found and the number of false alarms for the recognized batches. Furthermore, the influence of the separation threshold on the classification performance is also presented. These results show that the “large” models have better performance and the classification is less dependent on the separation threshold. Assuming the same separation threshold most of the models have shown similar pattern recognition performance with the exception of “large” *NNlm* and “small” *clustTS* models.

4.2. Structure size

In this analysis, by the size of the structure we understand the number of the tunable parameters. The classification results presented in Fig. 8 and Table 4 show that the “large” models were able to perform good classification. The only exception is the *NNlm* model using the threshold value of 0.6, which performs poorly. The analysis of the “small” model results shows that the classification performance is completely deteriorated for all models, except the *NNbr* for both threshold values. By comparing the performance of

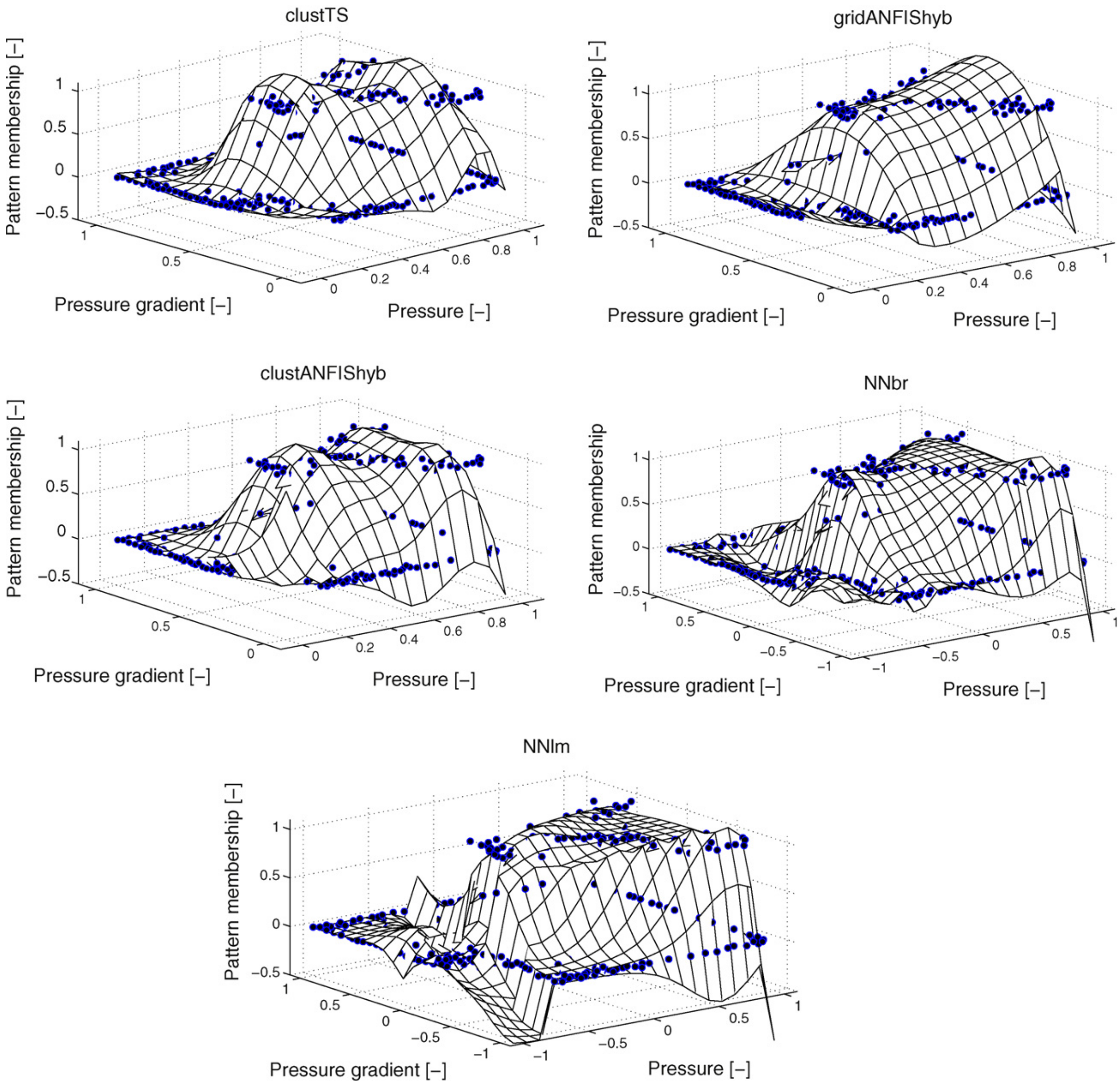

Fig. 5. "Large" classification models (surfaces) and the normalized plant data (dots).

the "large" and "small" models we can clearly recognize that the *NNbr* shows the most robust performance with regard to the model size. Moreover, the classification performance of the *NNbr* model is robust against the value of the separation threshold as well. In Figs. 8 and 9 it is presented that the process upset (lowest figure) could not be classified correctly by any of the models. This observation leads us to believe that this faulty pattern was erroneously included in the set of vacuum check patterns. Another interesting event can be observed with two batches before this faulty event. For this particular batch vacuum check was not carried out and the reason is unknown since the vacuum check is an automated procedure.

Table 4
Classification performance of the models in function of structure complexity and separation threshold.

Model	*clustTS*		*gridANFIShyb*		*clustANFIShyb*		*NNbr*		*NNlm*	
Separation threshold	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8
Large model										
Batch not found	1	1	1	2	1	2	1	2	1	2
False alarm	1	1	3	1	3	1	2	1	15	5
Small model										
Batch not found	2	24	1	4	1	5	1	1	1	0
False alarm	1	1	9	1	2	1	2	1	10	1

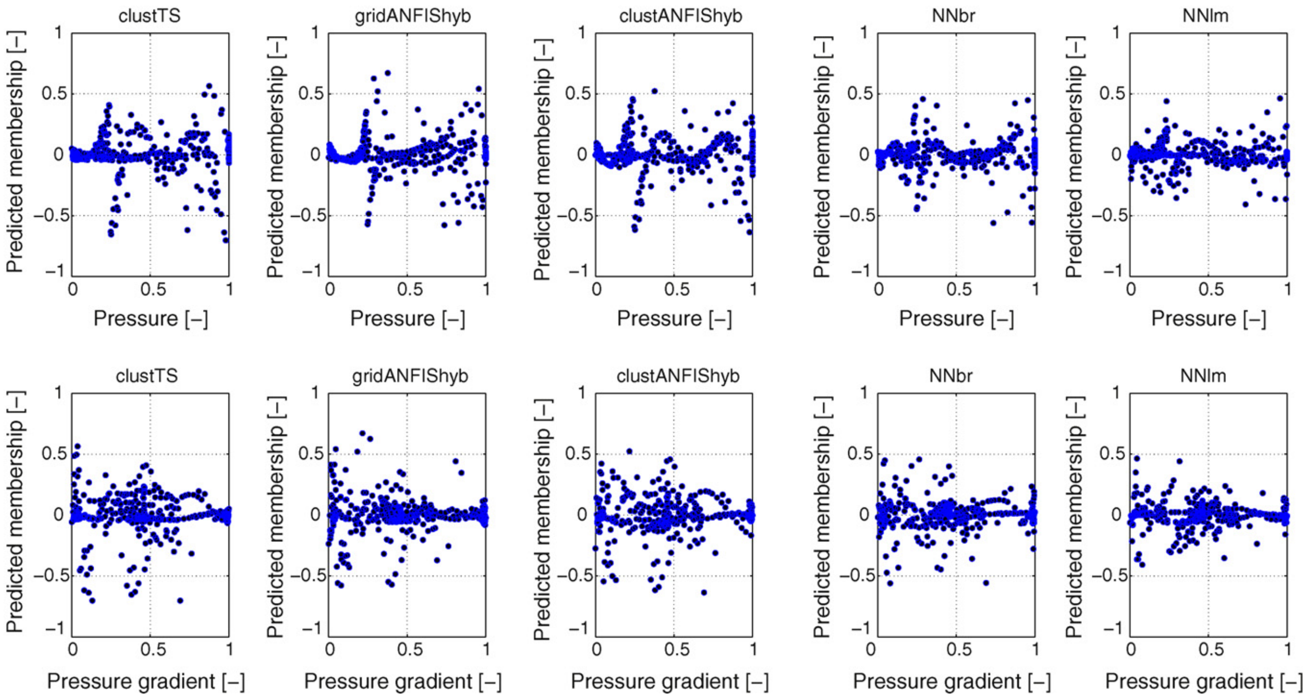

Fig. 6. Modeling residuals corresponding to the “large” models.

4.3. Structure type, parameter tuning method

For the “large” set of models presented in Table 4 it is concluded that the “large” *clustTS* model needed the most parameters in order to capture the non-linearity. Note that the “large” *clustANFIShyb* model (5 rules with 45 parameters) compared to the “large” *clustTS* model (8 rules with 72 parameters) has resulted in a better training error. Hence, it is concluded that the clustering based models can be significantly further improved using the hybrid optimization proposed within the ANFIS framework. The same observation is valid if the “large” *clustANFIShyb* model is compared with the “small” *clustTS* model. The “large” *gridANFIShyb* shows similar modeling and classification performance compared to the “large” *clustANFIShyb* model; however, it has less parameters (33 instead of 45).

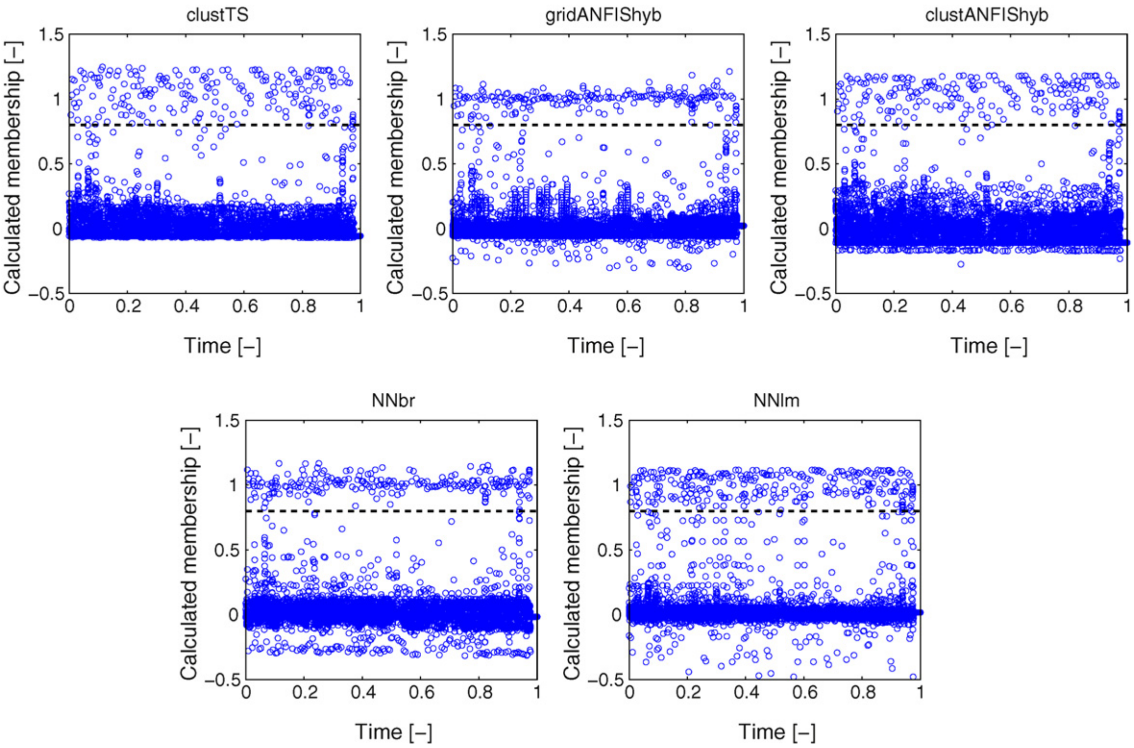

Fig. 7. Calculated pattern membership values using the “large” models; dashed lines show the separation threshold at 0.8.

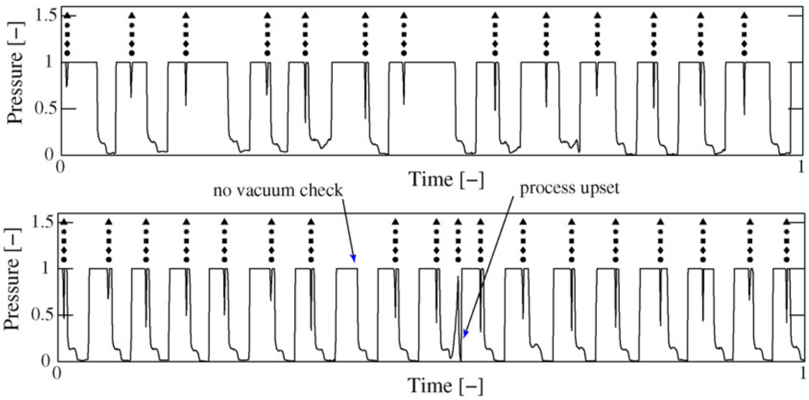

Fig. 8. Vacuum check pattern recognition results using the "large" models with 0.8 separation threshold: lines are the pressure profiles and the markers are the recognized vacuum check patterns ((●) *clustTS*, (♦) *clustANFIShyb*, (■) *gridANFIShyb*, (*) *NNbr* and (▲) *NNlm*).

For this application the grid partitioning based fuzzy structure identification is applicable since the input space is populated almost entirely by data.

The comparison between the *NNbr* and *NNlm* "large" models (these have the same number of parameters) shows that the *NNlm* matches best the data; however, its classification performance is the poorest and depends strongly on the separation threshold value. This result is not surprising, since it is known that the training process has to be controlled in order to avoid data overfitting. In this application the overfitting resulted in an increased number of false alarms. In contrast to this behavior, the *NNbr* model trained using the Bayesian regularization achieved good classification performance; note that no test set is used during the training.

According to the analysis of the classification results for the "small" model, it is concluded that in spite of the high RMSE value, the *clustTS* model is able to perform good classification, Table 4. However, the performance is not robust against the classification threshold and it depends strongly on it. Furthermore, it is observed that the performance of the *gridANFIShyb* and *clustANFIShyb* has deteriorated. Compared to the "large" *NNlm* model, the "small" one shows better performance for the threshold value of 0.8, however it is very sensitive towards it. The RMSE value of the "small" *NNlm* has increased compared to the "large" *NNlm* model; this fact can be explained by the lower number of tunable parameters and implicitly the lower structure flexibility. Note that for the threshold value 0.6 the *NNlm* model shows lower number of false alarms when the overfitting has decreased. The *NNbr* model provides the best classification performance among all "small" models and its classification performance is not sensitive towards the threshold value. The same conclusion is valid for the "large" *NNbr* model.

The performance of the models used in engineering applications can be directly correlated to a certain data matching index, such as the RMSE. However, in the case of this pattern recognition application it is rather difficult to link the RMSE value directly to the classification performance of the models; this is due to the classification performance dependence on the separation threshold.

4.4. Robustness of the NNbr models

In order to examine the robustness of the *NNbr* model with regard to structure complexity given by the number of parameters, several *NNbr* structures for 0.6 and 0.8 separation threshold values were tested as presented in Table 5.

The results show that the RMSE reaches a plateau and it is not dependent anymore on the structure size. In Fig. 10 the pattern recognition results using separation threshold 0.8 are presented and it is observed that the process upset was erroneously classified by all *NNbr* models.

Furthermore, it is concluded that the classification performance does not change with the growing structure size and it does not significantly dependent on the separation threshold, hence the *NNbr* networks deliver robust performance.

It is interesting to compare the worst performing classifier, the "large" *NNlm* model (RMSE: 0.109, 8 neurons, 33 parameters) with the *NNbr* model having similar modeling error (RMSE: 0.113, 16 neurons, 65 parameters). The comparison is presented in Fig. 11 as the contour plot of the distance between the two models. It is observed that these differ mostly in the regions where data is not available (left corner) with a difference as large as 0.6. By comparing the *NNbr* models with 14 (57 parameters, RMSE: 0.107) and 20

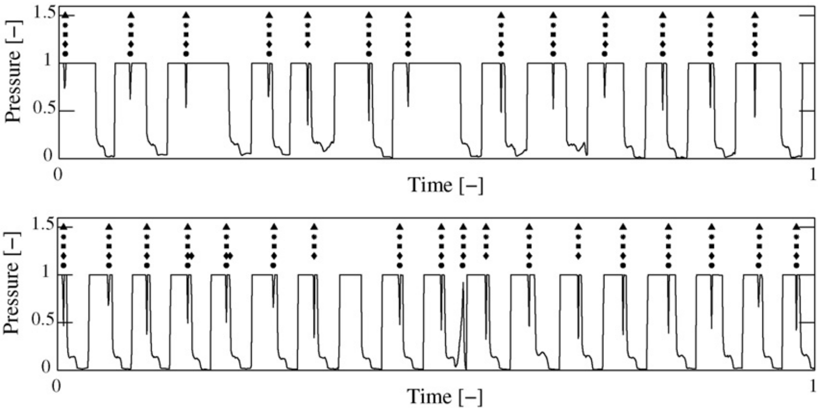

Fig. 9. Vacuum check pattern recognition results using the "small" models with 0.8 separation threshold: lines are the pressure profiles and the markers are the recognized vacuum check patterns ((●) *clustTS*, (♦) *clustANFIShyb*, (■) *gridANFIShyb*, (*) *NNbr* and (▲) *NNlm*).

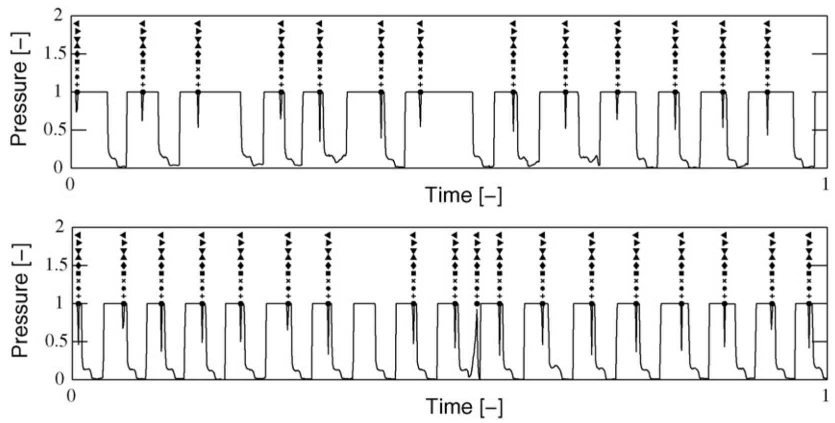

Fig. 10. Vacuum check pattern recognition results using the *NNbr* neural networks – 0.8 separation threshold and 100 training epochs – ((●) 6 neurons, (+) 8 neurons, (*) 10 neurons, (×) 12 neurons, (■) 14 neurons, (♦) 16 neurons, (▲) 18 neurons, (▼) 20 neurons, (►) 22 neurons and (◄) 24 neurons).

neurons (81 parameters, RMSE: 0.105), both trained for 100 epochs, Fig. 12, it is observed that the models are more similar compared to the previous case. Furthermore, it is concluded that the interpolation capabilities of these two models are similar, although the model complexity is significantly different—difference of 6 neurons or 24 tuning parameters.

Finally, it is concluded that any of the presented models are suited for classification purposes. However, these are not equally robust against the separation threshold. In the case of the *clustTS*, *gridANFIShyb* and *clustANFIShyb* models the sensitivity towards the separation threshold decreases as the RMSE value decreases. This is not valid for the *NNlm* model which tends to overfit the data,

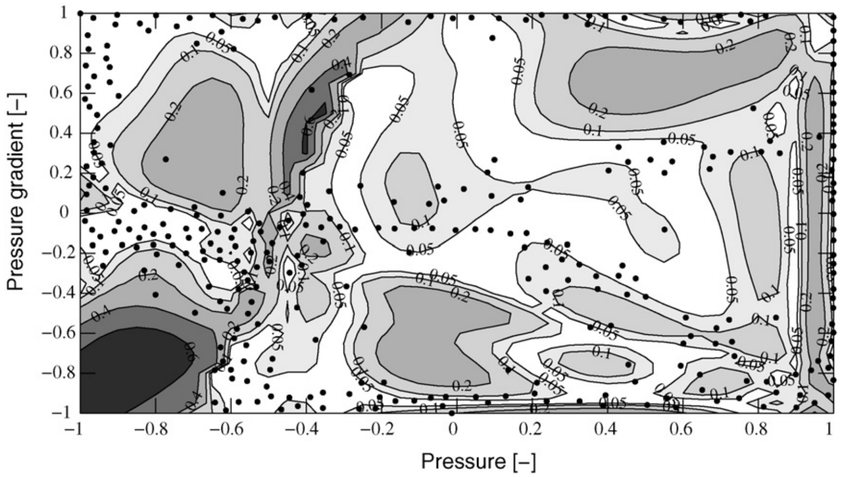

Fig. 11. Contour plot of the distance between the "large" *NNlm* model (8 neurons) and the *NNbr* model (16 neurons), Table 5; both models trained for 100 epochs; dots show input data.

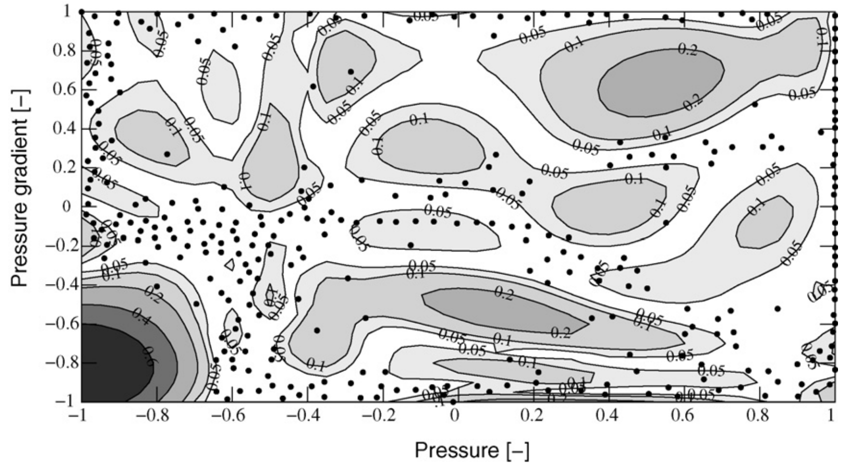

Fig. 12. Contour plot of the distance between the *NNbr* models with 14 and 20 neurons, Table 5; both models trained for 100 epochs; dots show input data.

Table 5
Comparison of the *NNbr* models classification performance based on the separation threshold and training epochs.

Neurons	6		8		10		12		14		16		18		20		22		24	
Parameters	25		33		41		49		57		65		73		81		89		97	
100 training epochs																				
RMSE	0.165		0.133		0.120		0.112		0.107		0.111		0.101		0.105		0.105		0.102	
Threshold	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8
Batches not found	1	2	1	2	1	2	1	2	1	2	1	2	1	1	1	1	1	1	1	1
False alarms	3	1	2	1	4	2	4	2	4	2	4	2	4	3	4	2	4	2	4	2
1000 training epochs																				
RMSE	0.179		0.154		0.127		0.119		0.113		0.113		0.102		0.101		0.091		0.916	
Threshold	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8	0.6	0.8
Batches not found	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	1	1	1	1	1
False alarms	3	3	4	4	3	4	4	4	4	4	4	4	4	4	5	4	5	4	5	4

provided that the neural network structure is complex enough. The best performer among the modeling techniques presented in this work is the *NNbr* model. Its performance is not affected by the structure size, nor by the classification threshold.

4.5. Discussion on the on-line pattern recognition

The on-line implementation of pattern recognition technologies may generate more value than the off-line applications, since the information is processed more frequently and corrective action can be promptly taken. For this case study, it was shown that it is feasible to calculate the batch times on-line (as these are completed) [47]. Usually, on-line systems (process controllers, estimators, classifiers) are retrained as soon as new measurement data is available, thus the models are regularly maintained. However, in case that the process patterns do not significantly change in time model retuning may not be necessary. For this case study, it was found that good pattern recognition is feasible even without retraining the model periodically, hence the maintenance costs of this classification system are minimal. One can eliminate the delay in plant data analysis by using an on-line decision making system, thus the number of off-spec batches can be minimized and productivity maximized.

5. Conclusions

In this contribution the development of an artificial intelligent decision maker using plant data from the PI System data was presented. The developed classifier was used to perform the pressure profile analysis of an industrial batch dryer. Using this classification system it was feasible to quantify the variation in drying time (batch time) which amounts to an average of 43% compared to the minimum drying time; it was a priori known that all the batches were operated with the same amount of material. According to these findings it is concluded that if the variation would be reduced by 50% (by technical, organizational or personnel measures), then a plant productivity improvement of 15% could be achieved. This is particularly important because the dryer is the productivity bottleneck in this batch plant.

While neural networks and neuro-fuzzy models are established techniques for pattern recognition applications, the Takagi–Sugeno fuzzy models were also successfully applied. Furthermore, it was found that by selecting the right separation threshold all models may perform similarly well. Additionally, the assumption that any of these models would suit the pattern recognition purpose has been confirmed. In this study the neural networks trained with the Bayesian regularization have shown the most robust classification performance. The performance of these neural networks was marginally influenced by the network size and data classification threshold. Moreover, these networks exhibit similar interpolation characteristics, although the number of neurons may be significantly different.

In this work it was shown that the main patterns can be captured already with low complexity models using minimal number of parameters, and depending on the requirements, increased model complexity can improve the pattern recognition performance and the robustness. The development of the pattern recognition systems is a complex and iterative procedure. The classification results are influenced by the preprocessing step, e.g. data reduction, normalization, by the classifier structure (model type) and complexity (number of tuning parameters), and last but not least by the training algorithm. The drawback of these pattern recognition systems is that they are problem specific. In order to use these for new classification tasks the decision maker models have to be redesigned and retrained. The findings presented in this paper are not case specific and can be directly used for the monitoring of a large variety of drying processes since the pressure profile features – vacuum check, pressure decrease, vacuum break – do not depend on the chemicals which are dried.

In the context of process analytical technologies (PAT) emerging fields of application of the artificial intelligence methods are for the digital image processing tasks [48,49].

Acknowledgement

The authors are grateful for the financial support from Ciba Specialty Chemicals, Switzerland.

References

[1] G. Stephanopoulos, G. Locher, M.J. Duff, R. Kamimura, G. Stephanopoulos, Fermentation database mining by pattern recognition, Biotechnol. Bioeng. 53 (1997) 443–452.

[2] C. Riverol, J. Cooney, A knowledge-based pattern recognition approach for the prediction of bubble size distribution in Newtonian fluids at high pressure, Chem. Eng. J. 112 (2005) 131–135.

[3] A. Meel, A.N. Venkat, R.D. Gudi, Disturbance classification and rejection using pattern recognition methods, Ind. Eng. Chem. Res. 42 (2003) 3321–3333.

[4] D.J. Cooper, L. Megan, R.F. Hinde, Comparing 2 neural networks for pattern based adaptive process control, AIChE J. 38 (1992) 41–55.

[5] D.S. Tow, D.F. Rudd, Pattern-recognition in computer-aided synthesis of industrial–chemical reactions, Ind. Eng. Chem. Res. 28 (1989) 1619–1628.

[6] M. Kim, Y.H. Lee, C.G. Han, Real-time classification of petroleum products using near-infrared spectra, Comp. Chem. Eng. 24 (2000) 513–517.

[7] S. Dash, R. Rengaswamy, V. Venkatasubramanian, Fuzzy-logic based trend classification for fault diagnosis of chemical processes, Comp. Chem. Eng. 27 (2003) 347–362.

[8] V. Venkatasubramanian, R. Rengaswamy, S.N. Kavuri, K. Yin, A review of process fault detection and diagnosis. Part III. Process history based methods, Comp. Chem. Eng. 27 (2003) 327–346.

[9] B.R. Bakshi, G. Stephanopoulos, Representation of process trends. 4. Induction of real-time patterns from operating data for diagnosis and supervisory control, Comp. Chem. Eng. 18 (1994) 303–332.

[10] S. Yoon, J.F. MacGregor, Principal-component analysis of multiscale data for process monitoring and fault diagnosis, AIChE J. 50 (2004) 2891–2903.

[11] I. Yélamos, M. Graells, L. Puigjaner, G. Escudero, Simultaneous fault diagnosis in chemical plants using a multilabel approach, AIChE J. 53 (2007) 2871–2884.

[12] M.C. Johannesmeyer, A. Singhal, D.E. Seborg, Pattern matching in historical data, AIChE J. 48 (2002) 2022–2038.

[13] A. Singhal, D.E. Seborg, Pattern matching in multivariate time series databases using a moving-window approach, Ind. Eng. Chem. Res. 41 (2002) 3822–3838.

[14] G. Richner, Y.M. Neuhold, S. Papadokonstantakis, K. Hungerbuhler, Temperature oscillation calorimetry for the determination of the heat capacity in a small-scale reactor, Chem. Eng. Sci. 63 (2008) 3755–3765.

[15] L.H. Ungar, B.A. Powell, S.N. Kamens, Adaptive networks for fault-diagnosis and process control, Comp. Chem. Eng. 14 (1990) 561–572.

[16] R. Irizarry, Fuzzy classification with an artificial chemical process, Chem. Eng. Sci. 60 (2005) 399–412.

[17] C. Fu, M. Poch, Fuzzy modeling and pattern recognition for dynamic processes and its application for an activated sludge process, Chem. Eng. Sci. 50 (1995) 3715–3725.

[18] G. Castellano, C. Castiello, A.M. Fanelli, C. Mencar, Knowledge discovery by a neuro-fuzzy modeling framework, Fuzzy Sets Syst. 149 (2005) 187–207.

[19] C. Bordieu, D. Rebiere, J. Pistre, R. Planade, Temporal window system: a new approach for dynamic detection—application to surface acoustic wave gas sensors, Sens. Actuators B: Chem. 35 (1996) 52–59.

[20] K.J. Mo, Y.S. Oh, E.S. Yoon, C.W. Jeong, Development of operation-aided system for chemical processes, Expert Syst. Appl. 12 (1997) 455–464.

[21] A. Klose, R. Kruse, Semi-supervised learning in knowledge discovery, Fuzzy Sets Syst. 149 (2005) 209–233.

[22] C. Faur, A. Cougnaud, G. Dreyfus, R. Le Goirec, Modelling the breakthrough of activated carbon filters by pesticides in surface waters with static and recurrent neural networks, Chem. Eng. J. 145 (2008) 7–15.

[23] J. Zhang, A.J. Morris, E.B. Martin, Long-term prediction models based on mixed order locally recurrent neural networks, Comp. Chem. Eng. 22 (1998) 1051–1063.

[24] M.L. Thompson, M.A. Kramer, Modeling chemical processes using prior knowledge and neural networks, AIChE J. 40 (1994) 1328–1340.

[25] L.L. Simon, U. Fischer, K. Hungerbuhler, Modeling of a three-phase industrial batch reactor using a hybrid first-principles neural-network model, Ind. Eng. Chem. Res. 45 (2006) 7336–7343.

[26] D.J.C. Mackay, Bayesian interpolation, Neural Comput. 4 (1992) 415–447.

[27] D.J.C. Mackay, A practical Bayesian framework for backpropagation networks, Neural Comput. 4 (1992) 448–472.

[28] F. Dan Foresee, M.T. Hagan, Gauss–Newton approximation to Bayesian learning, in: International Conference on Neural Networks, 1997.

[29] Z.K. Nagy, Model based control of a yeast fermentation bioreactor using optimally designed artificial neural networks, Chem. Eng. J. 127 (2007) 95–109.

[30] D.M. Himmelblau, Accounts of experiences in the application of artificial neural networks in chemical engineering, Ind. Eng. Chem. Res. 47 (2008) 5782–5796.

[31] E.H. Mamdani, Advances in linguistic—synthesis of fuzzy controllers, Int. J. Man Mach. Stud. 8 (1976) 669–678.

[32] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Syst. Man Cybern. 15 (1985) 116–132.

[33] P.F. van Lith, B.H.L. Betlem, B. Roffel, A structured modeling approach for dynamic hybrid fuzzy-first principles models, J. Process. Contr. 12 (2002) 605–615.

[34] J. Abonyi, R. Babuska, H.B. Verbruggen, F. Szeifert, Incorporating prior knowledge in fuzzy model identification, Int. J. Syst. Sci. 31 (2000) 657–667.

[35] J. Theocharis, G. Vachtsevanos, Recursive learning algorithms for training fuzzy recurrent models, Int. J. Intell. Syst. 11 (1996) 1059–1098.

[36] B. Roffel, P.A. Chin, Application of fuzzy logic in the control of polymerization reactors, Control Eng. Pract. 1 (1993) 267–272.

[37] K. Gasso, G. Mourot, J. Ragot, Fuzzy rule base optimisation: a pruning and merging approach 2000, in: IEEE International Conference on Systems, Man & Cybernetics, vols. 1–5, 2000.

[38] R. Rajesh, M.R. Kaimal, Gavlc: Ga with variable length chromosome for the simultaneous design and stability analysis of T–S fuzzy controllers, in: Fuzzy Systems Conference, 2008, FUZZ-IEEE 2008, IEEE International, WCCI, Hong Kong, 2008.

[39] R. Babuska, H.B. Verbruggen, An overview of fuzzy modeling for control, Control Eng. Pract. 4 (1996) 1593–1606.

[40] J.S.R. Jang, Anfis—adaptive-network-based fuzzy inference system, IEEE Trans. Syst. Man Cybern. 23 (1993) 665–685.

[41] L.L. Simon, N. Osterwalder, U. Fischer, K. Hungerbuhler, Systematic retrofit method for chemical batch processes using indicators, heuristics, and process models, Ind. Eng. Chem. Res. 47 (2008) 66–80.

[42] http://www.Osisoft.Com, Last Accessed on 29.06.2009.

[43] A. Singhal, D.E. Seborg, Effect of data compression on pattern matching in historical data, Ind. Eng. Chem. Res. 44 (2005) 3203–3212.

[44] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.

[45] H. Ying, General Siso Takagi–Sugeno fuzzy systems with linear rule consequent are universal approximators, IEEE Trans. Fuzzy Syst. 6 (1998) 582–587.

[46] D. Tikk, L.T. Koczy, T.D. Gedeon, A survey on universal approximation and its limits in soft computing techniques, Int. J. Approx. Reason. 33 (2003) 185–202.

[47] L.L. Simon, K. Hungerbuehler, Real time Takagi–Sugeno fuzzy model based pattern recognition in the batch chemical industry, in: Fuzzy Systems Conference, 2008, FUZZ-IEEE 2008, IEEE International, WCCI, Hong Kong, 2008.

[48] L.L. Simon, Z.K. Nagy, K. Hungerbuhler, Comparison of external bulk video imaging with focused beam reflectance and ultra violet–visible spectroscopy for crystallization nucleation detection and metastable zone identification, Chem. Eng. Sci. 64 (2009) 3344–3351.

[49] L.L. Simon, Z.K. Nagy, K. Hungerbuhler, Endoscopy-based in situ bulk video imaging of batch crystallization processes, organic process research & development, Polymorp. Cryst. 13 (2009) 1254–1261 (special issue).