MONITORING OF AN INDUSTRIAL DEAROMATISATION PROCESS

S. Bergman[1], **M. Sourander**[1], **S-L. Jämsä-Jounela**[2]

1 Neste Engineering Oy, P.O. Box 310, FIN-06101 Porvoo, Finland
2 Laboratory of Process Control and Automation, Helsinki University of Technology,
P.O. Box 6100, FIN-02015 HUT, Finland

Abstract: Process monitoring methods have been studied widely in recent years, and several industrial applications have been published. Early detection and identification of abnormal and undesired process states and equipment failures are essential requirements for safe and reliable processes. This helps to reduce the amount of production losses during abnormal events. In this paper, statistical multivariate methods and neural networks applied in monitoring of an industrial dearomatisation process are compared. No appriori process knowledge for the methods were assumed. The data for the comparison were generated with a dynamic simulator model of the process. Special emphasis was put on a case of internal leak in a heat exchanger. *Copyright © 2002 IFAC*

Keywords: Fault diagnosis, Neural networks, Statistical process control, Chemical industry

1. INTRODUCTION

An important purpose of process monitoring systems is to detect abnormal process states and to inform the plant operator accordingly. The objective is to detect even incipient faults. In the case of equipment malfunctions, it is essential to be aware of reduced functionality of the equipment before its complete breakdown, because this often leads to process shutdown.

During recent years increasing interest among researchers to apply different process monitoring and fault diagnosis systems has been recorded. A large number of applications have been reviewed e.g. by Isermann and Ballé (1997) and Patton *et al.* (2000). Both reviews indicate that there is a growing tendency to substitute the use of quantitative model-based methods by neural networks and fuzzy logic.

Dash and Venkatasubramanian (2000) have also published a review of monitoring methods, especially those applied in the field of chemical processes. They classified the methods according to the form of process knowledge used. One category of the methods are those based on process models. These include both qualitative causal models and quantitative methods. The other category is based on process history. This group includes both qualitative (e.g. rule-based) and quantitative methods (neural networks and multivariate statistical methods). The review showed that no single method was able to meet all the prerequisites for a good diagnostic system. Yoon and MacGregor (2000) classified the different methods into three categories: methods based on (i) causal models, (ii) qualitative knowledge and (iii) correlation models, such as neural networks and multivariate statistical methods. They compared statistical methods and causal model-based approaches. They concluded that, while the causal methods can usually only be applied to problems with a small number of variables, the statistical methods are strong in fault detection. However, their ability to identify the causes might be inadequate. Therefore, they suggest that statistical methods should be augmented with some causal information or prior fault knowledge.

The aim of this study was to compare different process monitoring methods applied in an industrial dearomatisation process. Both neural networks and multivariate statistical methods were applied. The methods were applied in real-time environment, with the emphasis on producing the conclusions as early as possible.

2. THE KOHONEN SELF-ORGANISING MAPS

Self-Organising Maps (SOM) are neural networks that form two-dimensional presentations from multi-dimensional data. The SOM algorithm was developed by Professor Teuvo Kohonen (1990). The algorithm preserves the topology of the data in such a way that the data vectors that closely resemble one another are located next to each other on the map.

The training algorithm is an iterative process. Each data vector must be compared with each neuron of the map, in order to find a best matching unit (BMU) for each data vector. The neuron on the map that most closely resembles the current vector is selected as its BMU. When the BMU is found, the weight vector of the neuron and its neighbouring neurons are updated according to formula (1)

$$\mathbf{w}_i(k+1) = \begin{cases} \mathbf{w}_i(k) + \boldsymbol{m}(\mathbf{x}(k) - \mathbf{w}_i(k)), \text{ when } i \in N_C \\ \mathbf{w}_i(k), \text{ when } i \notin N_C \end{cases} \quad (1)$$

where
$\mathbf{w}_i(k+1)$ is the new value of the weight vector of the neuron i,
$\mathbf{w}_i(k)$ is the old value of the weight vector of the neuron i,
$\boldsymbol{m}$ is the learning constant, and
N_C is the neighbourhood of the best matching unit.

The SOM can be interpreted by naming its neurons according to the classified data vectors. If a certain data vector is known to represent an abnormal or undesired process state, then the corresponding BMU can be named as an abnormal neuron. This gives the map a clear physical interpretation. However, the drawback of this method is that a neuron normally receives hits from data vectors representing both normal and abnormal states. Hence, a neuron cannot be explicitly named either normal or abnormal.

Highlighting the neuron corresponding to the latest measurement vector indicates the state of the process. It is often also beneficial to monitor how the state of the process has evolved during the last measurements, and hence a desired length trajectory of the latest neurons representing the latest states can be drawn on the map.

3. STATISTICAL MULTIVARIATE METHODS

3.1. Projection onto Latent Structures

Projection onto latent structures (PLS – also known as Partial Least Squares) is a multivariate statistical method that is based on dividing the measured variables into two blocks, one corresponding to the input-variables and the other to the output. The measurement data that are used for creating the model are respectively divided into two matrices, $\mathbf{X}$ and $\mathbf{Y}$. These matrices are represented in terms of scores and loading vectors. The loading vectors are linear combinations of the original variables, and they are defined such that the correlation between the blocks is maximised. The number of loading vectors (A) can be defined by means of cross validation. The PLS outer model is:

$$\mathbf{X} = \sum_{i=1}^{A} \mathbf{t}_i \mathbf{p}_i^T + \mathbf{E} = \mathbf{T}_A \mathbf{P}_A^T + \mathbf{E} \quad (5)$$

$$\mathbf{Y} = \sum_{i=1}^{A} \mathbf{u}_i \mathbf{q}_i^T + \mathbf{F} = \mathbf{U}_A \mathbf{Q}_A^T + \mathbf{F} \quad (6)$$

where
$\mathbf{t}_i$ and $\mathbf{u}_i$ are the ith score vectors,
$\mathbf{p}_i$ and $\mathbf{q}_i$ are the ith loading vectors also know as latent variables,
$\mathbf{T}$ and $\mathbf{U}$ are the score matrices,
$\mathbf{P}$ and $\mathbf{Q}$ are the loading matrices, and
$\mathbf{E}$ and $\mathbf{F}$ are the residual matrices.

The inner model represents the relationship between the blocks:

$$\mathbf{u}_i = f(\mathbf{t}_i) \quad (7)$$

In the traditional PLS model, the relationship is linear. There are also non-linear extensions of the PLS method. Qin and McAvoy (1992) used a multi-layer preceptor neural network to present the relationship f in formula (7). Wilson $et\ al.$ (1997) applied non-linear PLS in monitoring a distillation process by using a radial-basis function neural network relationship between the PLS input and output blocks.

3.2. Principal Component Analysis

Principal component analysis (PCA) is a method that projects multivariate data onto a smaller dimension. The principal components are linear combinations of the original variables. The principal components are selected in such a way that they explain as much variance as possible. The principal components are orthogonal, which means that transferring from the vector space defined by the original variables to the principal component space removes the correlations between variables.

Generally, most of the common-cause variations can be explained with a small number of principal components. The measurement data are collected in the data matrix $\mathbf{X}$, whose columns correspond to the n different measured variables and the rows correspond the m observations. Before building a principal component model, the data must be scaled in some meaningful way, e.g. mean centred and scaled to unit variance. The principal component decomposition of A principal components is:

$$\mathbf{X} = \sum_{i=1}^{A} \mathbf{t}_i \mathbf{p}_i^T + \mathbf{E} = \mathbf{T}_A \mathbf{P}_A^T + \mathbf{E} \quad (2)$$

where
$\mathbf{t}_i$ is the ith score vector (m×1),

$\mathbf{p}_i$ is the ith principal component also known as loading vector ($n\times1$),
T is the score matrix,
P is the loading matrix, and
E is the residual matrix.

The original variables form a basis to a vector space. The principal components $\mathbf{p}_i$ establish another orthogonal basis to the same *n*-dimensional space. The *A* first principal components span an *A*-dimensional subspace, and the scores of each observation represent the coordinates within the new base.

PCA can be used both as an independent process monitoring method and as a data pre-processing method for the self-organising maps. The strength of PCA as a pre-processing tool is based on elimination of the correlations between the original variables. In practice, the score vectors are used instead of the original measurements.

With PCA, the process can be monitored by projecting the new measurement vectors onto the subspace of the *A* first principal components and calculating the scores. Monitoring is based on observing whether the new scores are in the same domain as those in the in-control model. This can be indicated by the Hotelling T^2 statistics. When the state of the process is abnormal, the T^2 statistics exceeds its control limit. The limit can be calculated based on the F-distribution (Jackson 1991).

3.3. The Squared Prediction Error

The Hotelling T^2 statistics can only detect whether the variation in the subspace of the *A* first principal components is greater than the common-cause variation. When an event occurs that was not present in the in-control PCA model, the data vector can clearly lie outside the PCA subspace even though its projection onto the subspace is in an acceptable area. Such an event can be detected by observing the squared prediction error (SPE).

$$SPE = \left(\mathbf{x}_{new} - \hat{\mathbf{x}}_{new}\right)^T \left(\mathbf{x}_{new} - \hat{\mathbf{x}}_{new}\right) \qquad (4)$$

Once an undesired state is detected either with the T^2 statistics or the SPE exceeding its limit, it can be identified by examining the contributions of the individual variables on the T^2 and SPE. The variables having the largest contributions can indicate the root cause of the fault.

The SPE is used both with PCA and PLS. In conjunction with the PLS, the SPE is the squared error between the measured output variables and those predicted using PLS inner model.

4. DESCRIPTION OF THE DEAROMATISATION PROCESS

The purpose of the process is to remove aromatic compounds in the feedstock by hydrogenating them. The process consists of two continuous reactors filled with a packed bed of catalyst, a distillation column, and several other unit operations. The reactor part of the process is presented in Figure 1.
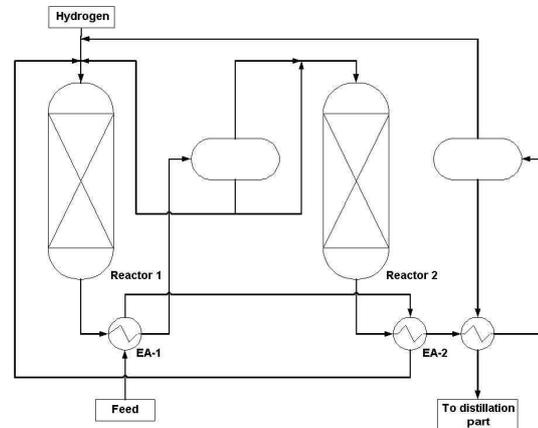
Fig. 1. The dearomatisation process.

The feedstock to the unit is in liquid phase. It is heated and then fed to the first reactor together with gaseous hydrogen. Most of the dearomatisation takes place in the first reactor. The reaction is exothermic. After the rector, heat is exchanged from the reaction product to the feed in the heat exchanger EA-1. The cooled reaction product is then fed to a gas separation drum where the gaseous hydrogen and liquid reaction product are separated. A part of the liquid is recycled back to the first reactor and the rest, together with the gas fraction, is fed to the second reactor. After the second reactor, heat is exchanged from the reaction product to the feed in the heat exchanger EA-2. The cooled reaction product is fed to a gas separation drum. The gas, including unreacted hydrogen, is recycled to the first reactor and the liquid is fed to a distillation unit. The non-aromatic product is drawn as the bottom product of the distillation column.

Several different feedstocks are used. They all are different petroleum oil cuts with clearly different thermodynamical and other properties.

4.1. Dynamic Process Simulator

A dynamic process simulator, PROSimulator, developed by Neste Engineering Oy, was used to study the different process states. PROSimulator is based on modelling the material, component and energy balances of the process, including chemical reactions and V/L relationships. The model of the dearomatisation process consists of three levels of hierarchy. In the highest level, the whole process is described. The second level is the reactor part of the process. In the lowest levels, the dearomatisation reactors are described in detail. The whole simulator

model of the process consists of some 800 differential equations (Lindholm 1996).

Actual plant data from over 60 measurements during a 100-hour period were examined in order to revise the simulator model to correspond to a dearomatisation process at Fortum's petroleum oil refinery in Naantali, Finland. The revision included adjusting the composition of the feed and the parameters of the equipment. The variations in the process measurements were also examined, and variations in both composition, temperature and flow rate of the feed were included in the revised simulator model.

Both normal states and a number of different fault conditions were simulated. The normal data used in constructing the monitoring models described in this paper were generated during a simulation corresponding to 48 hours of real process time. The data contain both long and short time variations similar to the real process.

5. THE PROCESS PHENOMENA MONITORED

The process phenomena that were selected for simulation and further study were (i) an internal leakage in a heat exchanger, (ii) fouling of a heat exchanger, and (iii) channelling of the catalyst. Special emphasis was placed on distinguishing these phenomena from each other, even though they all result in similar changes in certain measurements. In this paper, the monitoring of an internal leakage in a heat exchanger is described in detail.

In order to simulate the appearance of an internal leakage in heat exchanger EA-1, the simulation model was modified according to Figure 2. The heat exchanger was split into two halves, and a part of the cold-side aromatic-rich stream was split and mixed with the hot-side low-aromatic stream. In the simulated fault situation, the leakage increases continuously from starting until it finally reaches 10 % of the feed flow rate.
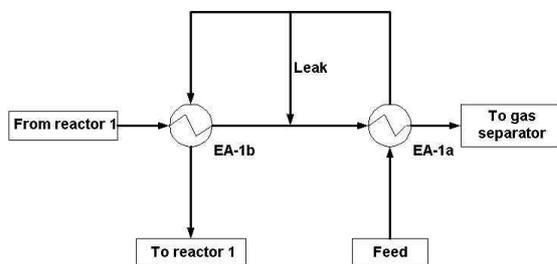
Fig. 2. Modelling an internal leak in the heat exchanger

6. RESULTS AND ANALYSIS

Construction of the process monitoring models began with selecting the variables. The effect of the simulated leakage on the different measurements was

examined. Ten important measurements were identified:
- the temperature of the heated cold feed after heat exchanger EA-1
- the output of the temperature controller of reactor 1 inlet
- three consecutive temperature measurements in reactor 1
- the temperature of the cooled hot stream after heat exchanger EA-1
- the inlet temperature of reactor 2
- one temperature measurement in reactor 2, and the outlet temperature of reactor 2
- the gas flow from the first gas separation drum.

In order to monitor the process, a number of models were constructed. Several self-organising maps were trained: some using directly the process measurements or process-specific computational variables and others with principal component pre-processing. PLS models were also constructed and their abilities to detect the undesired states were evaluated. Some computational quantities that characterise the process states were included in SOMs and PLS models in order to enhance their performance.

The training and analysis of the SOMs were performed with the fault diagnosis system developed by Jämsä-Jounela *et al*. (2001). The PCA pre-processing and construction of the PLS models were made with the Matlab[TM] PLS_Toolbox.

6.1. Self-organising map

A self-organising map was trained with ten process measurements and two additional computational variables, viz. the flow rates through both reactors multiplied with the temperature difference over the reactor. The training data contained both normal data and fault data that consist of four different leak cases.
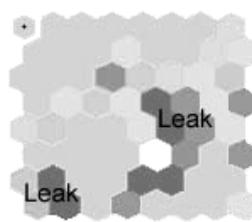
Fig. 3. The SOM trained with process data.

Figure 3 shows the distribution of the internal heat exchanger leakage on the SOM. Observations related to the normal situations are located in the left-hand side and in the upper half of the map and in the lower right-hand side corner. However, the normal and the fault states cannot be distinguished properly. On the SOM, the dark grey neurons correspond to the faults, the light grey neurons correspond to the normal situation, and the white neuron received no hits. On this SOM, there are several neurons that were found as best matching units for both normal and fault observation vectors.

6.2. Self-organising map with principal component pre-processing

Another SOM was trained with the principal component scores. First, a principal component model was constructed based on the normal data. The model of four principal components explained over 99 % of the variance in the 12 variables of the original normal data. Then, scores for the normal data and fault data were calculated. Figure 4 shows the distribution of the internal heat exchanger leakage on the SOM trained with the scores of the four principal components. Observations related to the leakage are located in the lower left-hand side corner. On this SOM, the normal and the fault states are classified more clearly than that in Figure. 3.
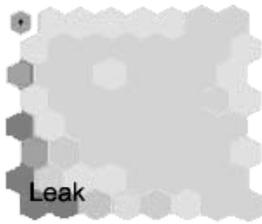
Fig. 4. The SOM trained with PCA pre-processed data.

Figures 3 and 4 show that principal component pre-processing improves the SOMs ability to distinguish between normal and fault states.

6.3. Analysis of the testing of the SOMs

After the maps had been trained, their classification abilities were evaluated using data not used in training the SOMs. The first test data set contained both normal and leakage data that resembled the training data. In the second test data set, the conditions of the feed differed from the training data.

In the tests, a hit represented a situation in which the data vector hits a neuron where the most probable state is the same as the state of the data vector. The test results of the SOM trained directly with the process measurements and the process-specific computational variables are shown in Table 1, and those of the SOM trained with PCA scores in Table 2.

Table 1. Results of detecting the internal leakage in the heat exchanger using the SOM trained directly with the process data

	Normal (set 1)	Leak (set 1)	Leak (set 2)
Number of cases	974	464	170
Number of hits	872	300	16
Hit percentage	89.5	64.7	9.4
Number of false alarms		102	87
Percentage of false alarms		25.4	84.5

Table 2. Results of detecting the internal leakage in the heat exchanger using the SOM trained with PCA pre-processed data

	Normal (set 1)	Leak (set 1)	Leak (set 2)
Number of cases	974	464	170
Number of hits	934	360	82
Hit percentage	95.9	77.6	48.2
Number of false alarms		40	10
Percentage of false alarms		10.0	10.9

Tables 1 and 2 show that the SOM with PCA pre-processed data is clearly better in classifying the internal leak in the heat exchanger than the SOM trained directly with the process measurements and computational variables.

Similar results were achieved with the SOMs trained with the same set of variables in detecting the channelling of the catalyst. The performance in monitoring the fouling of a heat exchanger was considerably lower. Adding process knowledge to the models improves these results.

The influence of the process-specific computational quantities can be examined by comparing the above-described SOMs to ones that were trained without the two computational variables. Without these, the number of false alarms was considerably higher and number of hits lower than those described in Tables 1 and 2.

6.4. Process monitoring with PLS model

In order to monitor the process, several PLS models were constructed. The model that was found best had six measurements or process-specific computational quantities in its **X** block and four in its **Y** block. Based on cross validation, the number of latent variables was selected as three.

The variables of the **X** block are:
- the flow-rate of the feed
- the temperature of the feed
- the outlet temperature of reactor 1
- the flow-rate through reactor 1
- the temperature difference over reactor 1
- the flow-rate through reactor 1 multiplied by the temperature difference over the reactor.

The variables of the **Y** block are:
- the temperature of the heated cold feed after heat exchanger EA-1
- the temperature of the cooled hot stream after heat exchanger EA-1
- the temperature difference over reactor 2
- the flow-rate through reactor 2 multiplied by the temperature difference over the reactor.

The abnormal process states are detected by monitoring the squared prediction error of the **Y**

block. The ability of the PLS model to detect the simulated leakage was evaluated using an independent test set. The SPE of the test data is shown in Figure 5. The limit for the SPE is defined on the basis of its value at the normal state, and it is indicted in Figure 5 with a dashed line. The SPE exceeds its limit when the leakage is approximately seven percent of the feed flow-rate.
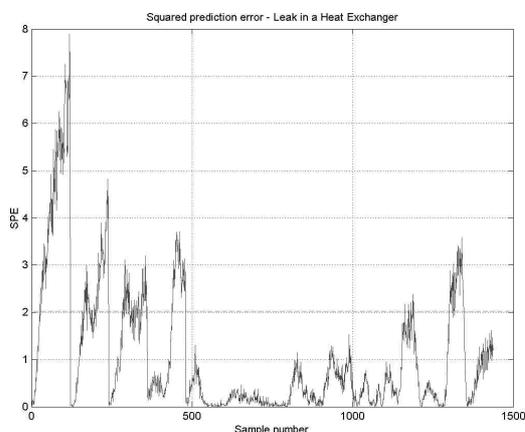
Fig 5. The squared prediction error of the test data.

The test results of the PLS model are shown in Table 3.

Table 3. Results of detecting the internal leakage in the heat exchanger using the PLS model.

	Normal	Leak
Number of cases	974	464
Number of hits	924	248
Hit percentage	94.9	53.4
Percentage of false alarms		16.7

Table 3 shows that the ability of the PLS model to detect the leak is fairly good. The same PLS model also detects over 68 percent of the catalyst channelling cases. Adding original variables to the PLS model does not improve its overall performance to detect abnormal situations.

However, the SPE only can detect whether some abnormal state has occurred or not. The cause of the abnormal state must be identified by monitoring the contributions of the variables to the SPE. The temperature difference over reactor 2 and the flow-rate through reactor 2 multiplied by the temperature difference over the reactor have the greatest contribution on SPE both in the leakage case and in channelling of the catalyst. This makes the identification of the fault more difficult.

7. CONCLUSIONS

Both self-organising maps and projection onto latent structure models gave promising results when applied in monitoring the simulated dearomatisation process. In order to improve the results from this performance level, process-specific knowledge should be added to the monitoring models.

Principal component pre-processing clearly improved the SOMs' ability to classify the process states. The drawback of the PLS model is its inability to identify the different abnormal states. In practice, the use of SOMs for process monitoring might be easier for a process operator, since the map with coloured neurons is very visual. If a PLS model is used in monitoring, both SPE and contribution plots must be observed.

REFERENCES

Dash, S. and Venkatasubramanian, V. (2000). Challenges in the industrial applications of fault diagnostic systems. *Computers & Chemical Engineering*, **24**, 785-791.

Isermann, R. and Ballé P. (1997). Trends in the Application of Model-based Fault Detection and Diagnosis of Technical Processes. *Control Engineering Practise*, **5**, 709-719.

Jackson, J.E. (1991). *A user's guide to principal component analysis*. John Wiley & Sons, New York.

Jämsä-Jounela, S-L., Vermasvuori, M., Haavisto, S. and Enden, P. (2001). Fault Diagnosis System for the Outokumpu Flash Smelting Process. *Preprints of the 10th IFAC Symposium on Automation in Mining, Mineral and Metal Processing*, Tokyo, Japan (pp. 295-300).

Kohonen, T. (1990). The Self-Organizing Map. *Proceedings of the IEEE,* **78**, 1464-1480.

Lindholm, J. (1996). *Fitting of a dynamic simulator to process measurements and development of a real time estimator*. Master of Science (Tech.) Thesis, University of Oulu. (In Finnish)

Patton, R.J., Uppal, F.J. and Lopez-Toribio, C.J. (2000). Soft Computing Approaches to Fault Diagnosis for Dynamic Systems: A Survey. *Proceedings of IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes*, Budapest, Hungary (pp. 298-311).

Qin, S.J. and McAvoy, T.J. (1992). Nonlinear PLS modeling using neural networks. *Computers & Chemical Engineering*, **16**, 379-390.

Wilson, D.J.H., Irwin, G.W. and Lightbody, G. (1997). Nonlinear PLS using Radial Basis Functions. *Transactions of The Institute of Measurement and Control*, **19**, 211-220.

Yoon, S. and MacGregor, J.F. (2000). Statistical and Causal Model-Based Approaches to Fault Detection and Isolation. *AIChE Journal*, **46**, 1813-1824.