Fault diagnosis of a benchmark fermentation process. A comparative study of feature extraction and classification techniques

Isaac Monroy,[a*] Kris Villez,[b] Moisès Graells,[a] Venkat Venkatasubramanian[b]

[a]Chemical Engineering Department (DEQ)-CEPIMA. Universitat Politècnica de Catalunya. EUETIB Comte d'Urgell 187, 08036 Barcelona, Spain; telephone: +34 934137275.

[b]Laboratory for Intelligent Process Systems, School of Chemical Engineering, Purdue University, West Lafayette, IN 47907, USA.

[isaac.monroy@upc.edu*, kvillez@purdue.edu, moises.graells@upc.edu, venkat@purdue.edu]

Abstract

This paper investigates fault diagnosis in batch processes and presents a comparative study of feature extraction and classification techniques applied to a specific biotechnological case study: the fermentation process model by Birol, Ündey and Çinar (2002), which is a benchmark for advanced batch process monitoring, diagnosis and control. Fault diagnosis is achieved using four approaches on four different process scenarios based on different levels of noise so as to evaluate their effects on the performance. Each approach combines a feature extraction method, either Multi-way Principal Component Analysis (MPCA) or Multi-way Independent Component Analysis (MICA), with a classification method, either Artificial Neural Network (ANN) or Support Vector Machines (SVM). The performance obtained by the different approaches is assessed and discussed for a set of simulated faults under different scenarios. One of the faults (a loss in mixing power) could not be detecteddue to the minimal effect of mixing on the simulated data. The remaining faults could be easily diagnosed and the subsequent discussion provides practical insight into the selection and use of the available techniques to specific applications. Irrespective the classification algorithm, MPCA shows to render better results than MICA, hence the diagnosis performance proves to be more sensitive to the selection of the feature extraction technique.

Keywords: Fault diagnosis, fermentation processes, MPCA, MICA, ANN, SVM.

1. Introduction

Batch and semi-batch processes are used frequently in biotechnological industries to generate high value-added products in relatively small volumes. Batch processes are characterized by finite duration, unsteady behavior, high conversions and most importantly, a recipe-driven operation. Their monitoring is of utter importance to meet specifications and strict quality requirements, as well as for process optimization and safety, waste reduction, and to enhance general process knowledge.

These processes may suffer a lack of reproducibility from batch to batch due to disturbances and the absence of on-line measurements of quality parameters. Variations among batches may be difficult for an operator to discern and it may be difficult to foresee their adverse effects on the final product quality. Often, disturbances or operational problems, and the subsequent poor quality of the final product, may remain undetected for a long time, until significant expenditure has been incurred.

The penicillin production process has been used by several authors [1,2,3,4,] as a case study to address the problem of batch process monitoring and Fault Detection and Identification (FDD) and it is considered a benchmark for batch processes as the Tennessee Eastman process is for continuous processes.

There is an extensive literature on the modelling of penicillin production but many of the reported models are too simplified or do not consider the effects on biomass growth and penicillin production of important operating variables, such as temperature, pH, agitation power or substrate feed flow rate.

The mechanistic model used in this work [1] considers input variables such as pH, temperature, aeration rate, agitation power and feed flow rate of substrate, introduces the CO_2 evolution term and uses experimental data to improve the simulation of penicillin production by extending the existing mathematical models [1,3]. Some minor modifications to this model were made and explained in the next section.

On the other hand, several techniques have been developed for process monitoring and diagnosis of chemical processes. These techniques can be broadly classified into three categories: model-based techniques, expert systems and data-driven methods [5,6].

The first two categories have been developed earlier in history. However, the advent of an increasing computational capacity and the need for operating ill-understood processes have expanded the attention to the third category, which establishes models on the basis of historical data with minimal input of knowledge. For instance, the complexity of biochemical processes such as the production of antibiotics makes it difficult to

create a detailed and practical model. Instead, empirical models based on process historical data to supplement simple mechanistic models are available [7].

As a result, a growing interest in the use of multivariate (MV) techniques in batch process modelling and FDD has been observed in literature [8,9,10]. A large part of the available literature is focused on the development and application of so called latent variable methods like Principal Component Analysis (PCA) and Partial Least Squares (PLS). These methods can handle highly correlated data sets and allow analysis and visualization which aids in the understanding of process data and possibly of the process itself. Furthermore, these MV techniques are well-suited for on-line Statistical Process Monitoring (SPM) and FDI of batch production and biotechnological processes [4].

Pioneering work in the specific area of fault detection and identification (FDI) for batch processes was performed by Nomikos and MacGregor [11,12] and has been successfully applied to industrial processes on several occasions [13,14]. Here, the basis is to model the common-cause variation present in collected data obtained under Normal Operating Conditions (NOC). This model is subsequently used to determine whether a new batch corresponds to this historically recorded normal operating behavior or not. Therefore, the monitoring performance depends heavily upon this NOC data [15].

Deviations in process variables during the progress of a batch can provide information about product properties and an estimation of the quality of the final product well before the completion of the batch. Process monitoring and fault diagnosis have been very effective in achieving this goal of process supervision [16]. More specifically, Multi-way Principal Component Analysis (MPCA) has been successfully applied to batch processes in order to monitor the process, identify when it shifts to a new operating condition and detect and diagnose abnormalities [12].

In the last two decades a plethora of techniques for FDI have been developed and reported [17] and it has become difficult for practitioners in research and industry to choose a method. Moreover, it is the opinion of the authors that for most commercial processes suitable techniques exist for monitoring and diagnosis. As such, this work is a collaborative effort in view of establishing guidelines for the choice between some techniques, rather than extend the existing, which have been reported to perform successfully in bioprocesses [18,19]. This paper reports the first study in this line of research and the specific purpose is to find out whether the choice for Multi-way Principal Component Analysis (MPCA) or Multi-way Independent Component Analysis (MICA) as feature extraction method and the choice for Artificial Neural Networks (ANN) or Support Vector Machines (SVM) as

classification technique is important for the purpose of finding a suitable fault diagnosis strategy in batch processes.

The comparison among the four combined approaches that results from selecting one technique from the feature extraction methods and one from the classification is done for several process scenarios with different levels of noise in the data so as to assess how noise affects the diagnosis performance of the individual approaches as well as generalize the result of the best approach.

The paper is organized as follows. Materials and methods are summarized in section 2. Section 2.1 describes the Penicillin production process as case study, Section 2.2 presents the statistical techniques that are used as feature extraction methods for comparative purposes, Section 2.3 describes the techniques used as classification algorithms and Section 2.4 summarizes the integration of a feature extraction with a classification technique as the fault diagnosis general approach broken out in two stages, the monitoring step and the fault diagnosis procedure. Section 3 reports the Results, Section 4 exposes some relevant discussions and finally, conclusions of the work are presented in Section 5.

2. Materials and Methods

2.1 Case study

The case study addressed is an extension of the model developed by Birol et al.[1] and is implemented in Matlab. The original nomenclature is followed in Table 1, which provides the data for the case study. The extension of the original model consists of the inclusion of a more practical PI controller for temperature and pH, as well as the related variables: acid flow rate (F_a), the base flow rate (F_b) and the heating/cooling water flow rate (F_c). The static form of the PI control algorithm is used according to the parameters values given in Table 1 and the equations determining the control action U^n_k as a function of the error E^n_k and the integral and proportional signals (I^n_k and P^n_k):

$$U_k^n = I_k^n + P_k^n \tag{1}$$

$$I_k^n = I_{k-1}^n + \left(\frac{K_c^n}{\tau_I^n} \times E_k^n \right) dt \tag{2}$$

$$P_k^n = K_c^n \times E_k^n \tag{3}$$

$$E_k^n = Y_k^n - Y_{SP}^n \tag{4}$$

Set points (Y^n_{SP}) are established at pH=5 and T=298 K, and the derivative time *dt* is set to match the sampling interval (0.02h). Furthermore, the substrate feed rate (*F*) is controlled by an on/off controller, which switches operation to fed-batch when the glucose concentration reaches the 0.3 g/l threshold.

Table 1. Initial conditions, kinetic and controller parameters for normal operation

Parameter symbol	Parameter description	Value	Unit
Initial conditions			
C_{CO2}	Carbon dioxide concentration	0.5	mmol/l
C_H	Hydrogen ion concentration	$10^{-5.1}$	mol/l
C_L	Dissolved oxygen concentration (=C_L^* at saturation)	1.16	mg/l
P	Penicillin concentration	0	g/l
Q_{rxn}	Heat generation	0	cal
S	Substrate concentration	15	g/l
T	Temperature	297	K
V	Culture volume	100	l
X	Biomass concentration	0.1	g/l
F	Feed flow rate of substrate	0.042	l/h
F_a	Acid flow rate	-	l/h
F_b	Base flow rate	-	l/h
F_c	Heating/cooling water flow rate	-	l/h

P_w	Power density	600	W
s_f	Feed substrate concentration	600	g/l
Controller parameters			
K^1_c	Proportional part of Acid in pH control	$1 . 10^{-4}$	-
τ^1_I	Integral proportion of Acid in pH control	8.4	h
K^2_c	Proportional part of Base in pH control	$8 . 10^{-4}$	-
τ^2_I	Integral proportion of Base in pH control	4.2	h
K^3_c	Proportional part of cooling in Temperature control	70	-
τ^3_I	Integral proportion of cooling in Temperature control	0.5	h
K^4_c	Proportional part of heating in Temperature control	5	-

The initial values of some of the input variables (feed substrate concentration, substrate feed temperature, power density and air flow rate) and the set points of the controlled variables (temperature, pH and substrate feed rate in the fed-batch stage) were randomly sampled from independent normal distributions with the mean values reported in Table 1 and standard deviations equal to 1% of their corresponding mean.

Noise was also added to the input and output variables at four different levels (0%, 1%, 5% and 10%) in the model at four different levels. Different nominal values, from which the noise is calculated, are set for each input,controled and output variable.

In order to simulate Abnormal Operating Conditions (AOC), three different faults are considered, namely:

(1) decrease in the agitation power to values between 30 and 200 W,

(2) increase of the saturation constant (K_x) from 0.15 g/l (nominal value) to values ranging from 0.3 to 0.9 g/l and

(3) decrease of the substrate feed rate in the fed-batch stage to values ranging from 0.001 to 0.01 l/h.

It is worthy to notice that faults 1 and 3 are likely caused by faults in the equipment, while fault 2 would likely be generated by human error (the culture contamination or the addition of an impure substrate), which affects the saturation constant value. The sampling time was set to 0.02 h by default, as in the original work [1]. In the process, a fed-batch operation follows the batch operation when the carbon source (glucose) almost depletes. All the simulated batches lasted four-hundred hours.

2.2 Feature extraction methods

Fault diagnosis in batch processes requires previous data arrangement and standardization and a feature processing step. Unfolding, centering and scaling are applied in this paper as part of the data representation before a feature extraction step.

Feature extraction techniques allow reducing the number of process variables to few linear combinations, called components, that represent the major sources of variation in the original variables. Two latent variable techniques, Multi-way Principal Component Analysis (MPCA) and Multi-way Independent Component Analysis (MICA), are applied and compared. A description of these techniques will be presented in the next sub-sections.

2.2.1 Data pre-processing: unfolding, centering and scaling

Multi-way feature extraction methods (MPCA and MICA) have been used as an extension of SPC methods to batch processes. These techniques project the information contained in the process-variable trajectories down into low-dimensional latent variable spaces, allowing summarizing the correlations across different variables and time instants.

In a typical batch run, $j=1,2,...J$ variables are measured at $k=1,2,...K$ time intervals throughout the batch and this data is reproduced on several $i=1,2,...I$ batch runs. The whole data is arranged in a three-dimensional matrix $\overline{X}$ $(IxJxK)$ [11].

Neither PCA nor ICA can be applied directly to such matrix. The so called unfolding procedure re-arranges the data into a two-dimensional matrix. MPCA and MICA are equivalent to unfolding the three-dimensional array $\overline{X}$ into a large two dimensional matrix X and then performing the original PCA and ICA.

$\overline{X}$ is arranged batch-wise into the matrix X ($IxKJ$), the most meaningful way of unfolding matrices for analyzing and monitoring batch processes [11,12], as it is shown in eq. 5.

$$X = \begin{pmatrix} x_{111} & x_{121} & \cdots & x_{1K1} & x_{112} & \cdots & x_{113} & \cdots & x_{1KJ} \\ x_{211} & x_{221} & \cdots & x_{2K1} & x_{212} & \cdots & x_{213} & \cdots & x_{2KJ} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{I11} & x_{I21} & \cdots & x_{IK1} & x_{I12} & \cdots & x_{I13} & \cdots & x_{IKJ} \end{pmatrix} \tag{5}$$

Then the data is mean centered and scaled. Centering is done by subtracting the mean of each column of the matrix X. The time observations in each column of the centered matrix are also scaled to unit variance when divided by their standard deviation so as to give equal weight to each variable at each time interval. This kind of scaling is typically named as auto scaling. X^* matrix is the result of this data standardization step.

$$X^* = \frac{X - mean(X)}{std(X)} \tag{6}$$

2.2.2 Multi-way Principal Component Analysis (MPCA)

MPCA was introduced by Geladi et al., in 1987 [20] to permit the PCA application to three-way data arrays. MacGregor and Nomikos [21] and Nomikos and MacGregor [12] were able to show that MPCA was well suited to handle multi-way batch data in the context of process monitoring.

MPCA is performed on the batch-wise unfolded, mean-centered and scaled data matrix.

MPCA decomposes the standardized data matrix X^* into a summation of R products of score vectors (t_r) and loading matrices (p_r) plus residuals (E), which are as small as possible in a least square sense.

$$X^* = \sum_{r=1}^{R} t_r p_r^T + E = tp^T + E = tp^T + \tilde{t}\tilde{p}^T = \begin{bmatrix} t & \tilde{t} \end{bmatrix} \begin{bmatrix} p & \tilde{p} \end{bmatrix}^T \equiv \overline{t}\overline{p}^T \tag{7}$$

where R is the number of retained principal components. Usually, a few principal components can express most of the variable correlations when the variables are highly correlated. Since the columns of $\overline{t}$ are orthogonal, the covariance matrix of the data is:

$$S \approx \frac{1}{I\text{-}1} X^T X = \overline{p}\overline{\Lambda}\overline{p}^T \tag{8}$$

where

$$\overline{\Lambda} = \frac{1}{I\text{-}1}\overline{t}^T\overline{t} = diag\left\{\lambda_1, \lambda_2, ..., \lambda_{KJ}\right\}$$

and

$$\lambda_i = \frac{1}{I\text{-}1} t_i^T t_i \approx var\left\{t_i\right\}$$

The score vector t_i is the ith column of $\overline{t}$ and λ_i are the eigenvalues of the covariance matrix in descending order [22].

MPCA is very useful for batch process data analysis because (1) it is effective in modelling correlation between variables across the time length of batches [23] and (2) computational efforts are very low compared to explicitly dynamic models.

Some authors have started using Unfold Principal Component Analysis (UPCA) as a better name for the MPCA method, because MPCA is not a multi-way method in the strict sense unlike the so called Tucker models and PARAFAC [24,25].

2.2.3 Multi-way Independent Component Analysis (MICA)

Independent Component Analysis (ICA) has been developed as a technique that extracts statistically independent and non-Gaussian components from multivariate observed data. Whereas PCA finds a set of uncorrelated signals, ICA finds a set of independent source signals.

A generic ICA model for any continuous process is:

$$\begin{pmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_j(k) \end{pmatrix} = A \begin{pmatrix} s_1(k) \\ s_2(k) \\ \vdots \\ s_r(k) \end{pmatrix} \tag{9}$$

where $[x_1(k), x_2(k), \ldots, x_j(k)]$, is a set of J variables observations at each time interval k, A is an unknown mixing matrix that corresponds to the loadings matrix p in PCA and s is the independent component data matrix that corresponds to the scores matrix t. It can be assumed that variables observations are generated as a linear mixture of R ($\leq J$) unknown independent components.

For a batch process, ICA can be applied to the unfolded, mean-centered and scaled data matrix as done for MPCA, thereby resulting in the MICA technique [2,3,26].

When K observations are available and there is data for I batches, the preceding equation can be rearranged as:

$$X = AS \quad \text{being } X \in \Re^{JK \times I}, S \in \Re^{R \times I} \tag{10}$$

where R is the number of independent components. MICA extracts the independent scores for NOC batches and faulty batches as in MPCA.

2.3 Classification methods

Process fault detection with artificial intelligence techniques has been studied by Venkatasubramanian *et al.* [27], Kavuri *et al*. [28], Himmelblau *et al*. [29], Watanabe et al.[30] and others. Most of these techniques are essentially classification methods for assigning classes to faulty conditions. In this work, two classification methods, ANN and SVM, are implemented and evaluated. Both ANN and SVM are black-box methods, suitable for situations where first-principles knowledge is lacking or difficult to achieve.

2.3.1 Artificial Neural Networks (ANNs)

ANNs belong to the most popular pattern recognition methods [31] and are models which capture nonlinear reltionships between variables for otherwise unknown processes. To use them for fault diagnosis, ANNs are trained on historical or simulated process data with the aim of detecting and diagnosing a specified number of faults by differentiating various abnormal patterns from the normal pattern in the output data.

ANNs can be interpreted as a network of massively parallel distributed processing units (neurons) that can store experimental information and make it available for future use [32]. The nodes and information flows are set up in such a way that the resulting network has signal inputs and outputs.

The Multilayer perceptron (MLP) architecture is applied in the ANN method used in this work. A single neuron in the MLP is represented mathematically by the following equations, as reported in [33]:

$$v_k = \sum_{j=0}^{r} w_{kj} x_j \tag{11}$$

$$y_k = \phi(v_k) \tag{12}$$

where r is the total number of inputs to neuron k, w_{kj} represents the input weights to neuron k, x_j represents the output values from the previous layer, v_k is the input to the transfer function of the neuron k, ϕ is the transfer function and y_k is the output from neuron k.

Typical transfer functions include linear functions, the Heaviside function, the logistic function and hyperbolic tangent functions. The values of the synaptic weights are determined by training the network using the Back-Propagation Algorithm (BPA), which consists of two passes through the network layers and is the most widely spread calibration algorithm in ANN [32]. In our study, the Levenberg-Marquardt BPA was used. The input data consist of the scores obtained from MPCA or MICA, while the output data consists of the predictions values to each trained fault.

2.3.2 Support Vector Machines (SVM)

As ANN, SVM are able to classify linear and non-linear cases. SVM method is based on the statistical learning theory and the Structural Risk Minimization (SRM) principle developed by Vapnik [33]. According to the SVM theory, a single global optimum exists for the numeric parameters of the models obtained from SVM so that the calibration stage (data models construction) is straightforward [35].

Practically speaking, the N-dimensional input data set is mapped into a feature space via a selected kernel function (e.g. linear, polynomial and Gaussian functions). In this feature space, the SVM-based classification is posed as the maximization of the model performance and solved as a quadratic optimization problem, identifying optimal separation hyperplanes, each of which exhibits a maximum linear margin (the largest

distance to the nearest training data from any class) and a number of support vectors (such observations close to the optimized hyperplane). The more complex the hyperplane is and the larger the dimension of the data, the more support vectors will be needed.

Yélamos *et al*. [36], Chiang *et al*. [37], Kulkarni *et al*. [38], among others, describe SVM in detail as well as the specific equations. Yungfeng *et al*. [39] apply SVM as classification algorithm for detecting faults using the penicillin production process as case study.

SVM-based diagnosis models are also obtained in this work and used exactly in the same way as the ANNs, i.e. with the scores obtained from MPCA or MICA as inputs and performing the predictions of each sample to the faulty classes as outputs.

2.4 Integration of feature extraction and classification methods

Fault detection is performed using MPCA. Next, this work analyses four different options for diagnosis, which result from combining two feature extraction methods (MPCA and MICA) and two classification methods (ANN and SVM). The resulting two-step procedure is detailed in Figure 1, and the two steps are explained in the following sub-sections.

2.4.1 Monitoring step

One way of monitoring either continuous or batch processes is to use Multivariate Statistical Process Control (MSPC). Specifically, batch process monitoring can be divided into three phases: initial, training and application phases, as described by Ramaker *et al.* [40].

The initial phase consists of collecting a set of historical data from real-time measurements of a NOC batches set. This phase can be a bottleneck because of poorly designed structures of the database. In this sense, the measured data from all NOC batch runs are arranged in a three-way matrix as stated. Moreover, in this phase previous knowledge can be applied to select good batches.

In the training phase, suspicious batches which are not considered to be under NOC are removed from the historical data and then, a representative set of NOC batches is grouped and used for constructing an empirical model. The most common monitoring model to use, produced in this work, is an MPCA model.

The number of principal components needed to construct an MPCA model that describes adequately the normal behavior of a batch operation can be found on the basis of several criteria [11]. One commonly used criterion is

the broken-stick rule [41], which judges whether a principal component adds any structural information about the variance in the data or only explains noise. Therefore, the broken-stick rule is applied to the selection of the number of retained principal components in the MPCA model.

Finally, in the application phase, batches are monitored by means of statistics derived from the training phase. Usually more than 50 batches are required for obtaining a representative sample of sufficient size to correctly estimate confidence limits for the normal operating region [42]. After projecting NOC and AOC batches onto the NOC model, statistical measures are calculated and compared to the corresponding control limits from the reference distribution.

Typically the SPE (or Q statistic) and Hotelling's T^2 statistic are used to represent the variability in the Residual Space (RS) and the Principal Component Space (PCS) respectively [22]. New batches can be projected onto the plane defined by the PCA loading vectors to obtain their scores $\left(t_{i,new} = x_{new} p_i\right)$, and the residuals $\left(E_{new} = x_{new} - \overline{x}_{new}\right)$, where $\overline{x}_{new} = t_{R,new} p_R^T$, $t_{R,new}$ is the $1 \times R$ vector of scores from the model and p_R is the $JK \times R$ matrix of loadings.

The batches can be compared by plotting their t scores and the sum of squared errors, given by the Q statistic [11].

$$Q_i = \sum_{c=1}^{KJ} E(i,c)^2 \tag{13}$$

The Q statistic is calculated through the summation of the squared residuals for a specific batch and represents the deviations that are not captured in the retained PCs. The Q statistic is used to compare the residuals of new batches to an upper control limit (Q_α), defined using a set of residuals from NOC batches.

The control limit of the Q statistic is calculated according to the equation in [43].

$$Q_\alpha = \theta_1 \left[1 - \frac{\theta_2 h_0 (1 - h_0)}{\theta_1^2} + \frac{z_\alpha (2\theta_2 h_0^2)^{1/2}}{\theta_1} \right]^{\frac{1}{h_0}} \tag{14}$$

$$\theta_1 = \lambda_i, \qquad \theta_2 = \lambda_i^2, \qquad \theta_3 = \sum \lambda_i^3, \qquad h_0 = 1 - \frac{2\theta_1 \theta_3}{3\theta_2^2}$$

In addition, the T^2 statistic represents the Mahalanobis distance between new data and the center of the normal operating condition data in the space spanned by the principal components.

At the end of the batch, the T^2 for batch i is calculated as follows [44]:

$$T_i^2 = t_r^T S^{-1} t_r \approx \frac{R\left(I^2 - 1\right)}{I\left(I - R\right)} F_{R,I-R} \tag{15}$$

where I is the number of batches in the reference set, t_r is a vector of R scores, S is the (RxR) covariance matrix of the t-scores calculated during the model development, R is the number of PCs retained in the model and $F_{R,I-R}$ is the F-distribution value with R and I-R-1 degrees of freedom.

For new observations or batches (i.e., not part of the calibration data), the upper control limit for this statistic (T^2_{lim}) is [45]:

$$T_{lim}^2 = \frac{R\left(I+1\right)\left(I-1\right)}{I^2 - IR} F_{\alpha,R,I-R} \tag{16}$$

For a new observation or batch i_{new}, if $T_{new}^2 < T_{lim}^2$ and $Q_{new} < Q_\alpha$, one considers the current batch to be in-control with $100(1-\alpha)\%$ confidence. Otherwise, the batch is identified as out of control.

Since the principal component subspace typically contains normal process variations with large variance and the residual subspace contains mainly noise, if a sample exceeds only the T^2 limit but does not violate the SPE limit, then this can be interpreted as a shift from the usual operating region without breaking this normal correlation structure. This can be due to faults, but also due to desired changes in the process operation. Naturally, the T^2 statistic is thus used to detect faults associated with abnormal variations within the model subspace, whereas the Q statistic is used to detect new events that are not taken into account in the model subspace [44].

In this work, fault diagnosis is addressed at the end of the batch. As such, this would not allow improving the past batch in practice. However, it is noted that several authors have addressed this problem in the past [2,11,15,26,46].

2.4.2 Fault diagnosis procedure

Once new batches are monitored by their projection onto the previous NOC model, the next stage is the diagnosis of the faulty batches. In order to create generalized classification models, NOC and AOC batches are

gathered in the same data set and then centering and scaling operations are applied to this set. The feature extraction techniques are used for dimensionality reduction before applying the classification algorithms.

Furthermore, cross validation is done in this work in order to assess better the classification performance of the different approaches. A 10-cross validation is chosen so that ten models considering both nominal and faulty batches can be tested with their respective validation sets.

The number of principal components in the models is retained by the broken-stick rule and the independent components are determined by a graphical technique similar to the SCREE test of PCA [47]. The scores T_f are used as inputs of the classification algorithms. Regarding the ANN algorithm, the number of inputs is automatically set to the number of components. The remaining architectural parameter to optimize in ANN is the number of hidden nodes, which is optimized based on the obtained cross-validated classification performance. The number of outputs is the same as the number of faults to classify.

Regarding SVM as classification algorithm, the architectural parameter to optimize for each scenario and the selected FE technique is the type of kernel function and its parameter (e.g. order for polynomial kernel, width for Gaussian kernel). An overall measure for both sensitivity and specificity, named the F1 score, is used to assess the diagnosis performance for each class separately and is computed as follows [48,49]

$$F1 = \frac{2 \times Sensitivity \times Specificity}{Sensitivity + Specificity} \tag{17}$$

3. Results

Figures 2 to 4 show sets of 50 simulation runs for three different scenarios: batches under nominal operating conditions (NOC), batches with fault 2 (increase in the saturation constant) and batches with fault 3 (decrease in the substrate feed rate in the fed-batch stage), all of them with 1% of noise level in input and ouput data.

Figure 2 shows the trajectories of the biomass concentration and illustrates how the decrease of the substrate feed rate (fault 3) affects the biomass production: those flow rate below the initial value for normal operation are not enough for the biomass growth and probably just for the cells maintenance. On the other hand, the culture contamination reflected in an increase of the saturation constant (fault 2) results in a slower production of penicillin and more conversion of substrate to biomass.

Figure 3 shows the trajectories of the penicillin production. The decrease in the substrate feed flow rate is shown again to affect the penicillin production. Furthermore, it also shows that an increase in the saturation constant delays the penicillin production.

Figure 4 shows the trajectories of the dissolved oxygen concentration observing that the decrease in the substrate feed flow rate causes higher concentrations of dissolved oxygen in the culture medium because the amount of biomass is smaller and because of that, the oxygen requirements.

As previously reported [42], more than 50 batches are used to estimate the confidence limits in the normal operating region. Also, 50 AOC batches are taken to construct representative diagnosis models for each fault. The NOC model was just built upon as many NOC batches as faulty batches used in the diagnosis step (100). The high ratio of faulty to normal batches is unrealistic from the point of view of industrial practice yet it does not affect our conclusions since a comparitive study is reported, rather than an absolute evaluation of fault diagnostic performance. Thus, UPCA or MPCA was performed on hundred batches under NOC, simulated with 1% of noise level. Figure 5 plots the eigenvalues corresponding to this analysis. According to this plot, two components were selected for the monitoring and detection model, which jointly capture 59.5% of the total variance.

Figure 6 and 7 report the Q statistic and T^2 values for the NOC batches with 1% of noise level in data and one can observe in there that both statistics remain below their control limits for all the batches. Similar results are obtained from analysing NOC batches in the rest of process scenarios (different noise levels), however these are not graphically reported in the paper.

Regarding the batches experiencing abnormal operating conditions (AOC), 50 batches per fault were simulated and monitored. Figure 8 shows the plot of the two first scores obtained from the projection of the NOC batches onto the monitoring model, as well as the scores corresponding to the AOC batches projected onto the same model. One can observe in this biplot that the batches corresponding to Fault 2 and 3 are well separated from the NOC batches and from each other, in contrast to the batches corresponding to Fault 1 (decrease in the agitation power) that are found in the same region as the NOC batches.

As this plot only considers the two first scores from 100, obtained by the broken-stick rule and representing the 59.5% of variability, the other PCs could be expected to allow discerning between NOC and Fault 1 batches. However, an evaluation of some biplots for some left PC's (not shown) indicated that this is not the case. For instance, fault 1 batches cannot be discriminated from the NOC set because (1) it is not a priori true that a faulty

situation also results in a distinguishable symptom and (2) the UPCA model is calibrated only on NOC batches and is therefore not oriented at capturing differences between NOC and AOC batches.

Figures 9 and 10 show the Q statistic and T^2 values in logarithmic scale for the batches experiencing faults 1 to 3. The applied confidence level for the calculation of the UCL is 99%. Q and T^2 statistics for the batches with fault 3 are over the UCL. In the case of the batches with fault 2, the Q statistic value is over the UCL for all of them but in the case of the T^2 statistic, 12 batches are below the UCL and 38 batches over. In this sense, as AOC batches are not taken into account in the model subspace, the Q statistic detects all the faulty batches as new events. Regarding batches with fault 1, both statistics are below the UCL for all of them, which means that this fault is not detected and all these faulty batches are considered as batches under NOC. A closer inspection of the state variables indicated that their patterns were hardly affected by this fault. As such, fault 1 was ignored and excluded for the diagnosis step that follows. The results showed in Figure 9 and 10 for the Q and T^2 statistics are quite similar to those obtained with the other process scenarios but they are omitted in the paper.

The same assumptions for monitoring hold for calibration and validation. Thus a minimum set of batches is also considered for constructing diagnosis models: 100 AOC batches (50 fault 2 and 50 per fault 3) and 100 NOC batches. The 10-fold venetian blind cross-validation results in selecting 10 normal batches and 5 batches of each faulty condition (total = 20) in a single validation set.

The feature extraction techniques retain four components (by means of broken-stick rule as explained before) for the ten projection models, using both MPCA and MICA and a different level of noise in the input and output variables (0%, 1%, 5% and 10%). Table 2 shows the percentage of variance with those retained components within the original variables using both feature extraction techniques and according to the corresponding scenario.

Table 2. Variability percentage with the retained components by broken-stick rule for both MPCA and MICA techniques and all the noise levels in data.

Retained variance (%)				
Feature extraction technique	Noise level (%)			
	0	1	5	10

MPCA	98	86	81	41
MICA	98	86	81	41

Table 3. Number of hidden nodes optimized using both MPCA and MICA with ANN for all the noise scenarios.

Number of hidden nodes				
Feature extraction technique	Noise level (%)			
	0	1	5	10
MPCA	1	3	2	1
MICA	14	9	9	7

The diagnosis or classification step consists of applying properly ANN and SVM as classification algorithms using the scores from MPCA and MICA as inputs. Regarding ANN, the analysis has been restricted to a single hidden layer network since this is capable of mapping all the data [50]. The number of tangent sigmoid nodes which performed the least mean squared normalized error (MSE) and the best classification performance is reported in Table 3 according to the feature extraction technique used and the noise scenario. The networks have two logistic output nodes as the number of faults to classify.

Regarding SVM as classification technique, Table 4 reports the kernel functions that offered the best classification performance when applying previously either MPCA or MICA and for the different scenarios.

Table 4. Kernel function optimized using both MPCA and MICA with SVM for all the noise scenarios.

Kernel function

Feature extraction technique	Noise level (%)			
	0	1	5	10
MPCA	Linear	Poly 2	Poly 2	Poly 2
MICA	Poly 3	Linear	Poly 3	Poly 3

The diagnosis results for each combination (MPCA&ANN, MPCA&SVM, MICA&ANN and MICA&SVM) and for each noise scenario (0, 1, 5 and 10%) are summarized in figures 11 to 14. The performance was evaluated according to the F1 score as a mean from every time observation per test batch and from the whole batches included in the ten validation sets.

Figure 11 shows the mean diagnosis performance considering the three classes of batches and what can be observed and concluded from these results is that the combination of either ANN or SVM with MPCA renders a significantly better diagnosis performance than the combinations with MICA. Therefore, the choice and optimization of the latent or feature extraction method is more important than the selection of the classification technique as can be observed for the four different process scenarios. In the case of the 10% noise scenario, the performance is low in comparison to the rest of scenarios, however it is important to consider that the retained components with the broken-stick rule only explain the 41% of the variance of the process variables, which could be the main reason of such bad performance. A further study considering the percentage of retained variance with the components would corroborate this assumption.

Breaking down the results, Figure 12 shows the diagnosis performance for the nominal class (batches under NOC) for the 16 situations and Figures 13 and 14 do the same for the Faults 2 and 3. These figures reveal that:

- In general, for all the process scenarios and a given classification method, MPCA leads to better results than MICA. The only exception is found for the 10% noise scenario, for which the faults are poorly classified with all the combinations.
- In general, for the four combinations between feature extraction and classification techniques, higher noise levels lead to worse diagnosis performance, which it is not surprising. There is an exception however in the normal class which shows apparently a good performance in the highest noise scenario when using MICA.

- Fault 3 is well diagnosed no matter the combination of feature extraction and classification techniques used. As previously pointed out, the exception is the 10% noise scenario probably because of the high noise in data and low variance in the extracted components.
- MICA performs better when it is combined with ANN rather than with SVM.
- MPCA plus SVM combination seems to be more affected by either high noise levels or low variance percentages explained by the retained principal components.

Both the expected and unexpected results will be discussed in more detail in the next section.

4. Discussion

The presented results point out the benefits of the step-wise procedure (feature extraction step plus fault diagnosis step) in terms of the final classification or diagnosis performance. The application of a feature extraction technique allows dimensionality reduction and obtaining features that summarize the information in the data. Such features retain a given percentage of the variance of the original variables. Moreover, as these feature extraction techniques are multivariate statistical techniques, they allow monitoring new batches in order to know whether they are successful or faulty previously to their diagnosis. Only then, i.e., once the diagnosis models have been constructed with historical batches and the appropriate techniques, the diagnosis of new and current batches becomes an easier task.

The comparison among feature extraction and classification techniques was performed with different noise scenarios (0%, 1%, 5% and 10%). The combinations of both classification techniques used in this work with MICA allow concluding that the best approaches are those in which MPCA is applied as feature extraction no matter the classification technique used for diagnosis. Also the reported results indicate that more effort has to be devoted on selecting the feature extraction technique rather than on the diagnosis algorithm.

The MICA&ANN combination works well when there is no noise or with low level of noise in data (1%), and the performance decreases when dealing with 5% noise. In fact, batches with 1% of noise show the best diagnosis performance, which can be explained by the 86% of the retained variance with the PCs in comparison to the 81% with 5% of noise. However, retaining less variance below a certain high threshold may sometimes, improve the diagnosis, rather than retaining a higher variance, which can explain why the diagnosis performance is better in the case with slight noise in the data (1%) than without noise.

In fault-less batches, MICA&SVM combination seems to be more affected by noise, although the 10% scenario shows an unexpected behavior.The predictions obtained from this combination and this scenario, show that the reason for a high diagnosis performance is that faulty batches are simultaneously diagnosed as both faults, which suggests that for this high level of noise the MICA&SVM approach does not recognize differences among faults.

The sensitivity part in the F1 score accounts for the right diagnosed batches divided by the total number of batches, and this is therefore calculated taking into account the half of a batch when it is double-classified. One third of a batch would be counted in the case of three simultaneous classes and so on. In the same way, specificity counts the right diagnosed batches divided only by the number of batches diagnosed in each class. This is the reason why F1 index reflects such modifications in the final performance.

According to these results, there are no significant differences between ANN and SVM when either of these are combined with MPCA, which indicates that the decision-making should be concentrated on the feature extraction technique. In general, these results can be useful for diagnosing other processes.

Finally, further research issues are described here. For other highly non-linear processes it will be interesting to simulate some different faults as well as to increase the number of NOC historical batches for constructing the diagnosis models. It will be also worthy to investigate other types of scaling such as group scaling before selecting and applying the feature extraction technique and to consider a specific percentage of variance in the retained components.

5. Conclusions

Different available techniques were applied on a benchmark fermentation process with the aim to provide guidelines for the general problem of selecting data-based methods for modelling and diagnosing biotechnological processes that may be ill understood on a mechanistic level. The comparative study presented focuses on the selection of a practical combination of feature extraction and classification techniques that can be of general use for fault diagnosis of highly non-linear batch processes.

Two feature extraction techniques, MPCA and MICA, and two well-known non-linear classifiers, ANN and SVM, were used for this purpose. As such, four approaches were evaluated, respectively tagged as MPCA-

ANN, MPCA-SVM, MICA-ANN and MICA-SVM. The feature extraction techniques (MPCA, MICA) were effective at reducing the dimensionality of the inputs to the classification algorithms (ANN, SVM). This was based on the application of the so called broken-stick rule. The ANN number of hidden nodes and the SVM kernel function were optimized to improve directly the classification performance. All the approaches were tested and assessed for several scenarios in order to generalize the study regarding to the combination between the proposed techniques.

The results obtained allow concluding that regarding fault diagnosis, the selection of the classification method is not as decisive as the choice of the feature extraction technique, which is the issue to be stressed in the design of a fault diagnosis approach with the available techniques. This is demonstrated in all the scenarios tested in the current study, which is applied to a biotechnological benchmark process.

Acknowledgements

Financial support from Generalitat de Catalunya through the FI fellowship program is fully appreciated. Support from the Spanish Ministry of Education through project no. DPI 2006-05673 is also acknowledged. The MPCA Matlab code was developed at modelEAU, Université Laval, Québec.

List of symbols

A: Unknown mixing matrix

C_a, C_b: Acid and Base molarity

C_{CO2}: Carbon dioxide concentration

C_H: Hydrogen ion concentration

C_L: Dissolved oxygen concentration

dt: Derivative time in the control system

E: Residuals

E_N: Error signal at sample N

F: Feed flow rate of substrate

F_a: Acid flow rate

F_b: Base flow rate

F_c: Heating/cooling water flow rate

f_g: Air flow rate

F1: Diagnosis performace index

f: Transfer function in the ANN

I: Number of batches

I_k: Integral signal at sample k

I_{k-1}: Previous integral signal

J: Process variables

k: neuron

K: Time observations

K_c: Proportional gain in PI control

K_x: Contois saturation constant

p: Loading vectors in PCA

P: Penicillin concentration

P_k: Proportional signal at sample k

P_w: Power density

Q_i: Sum of the squared errors or Q statistic

Q_{rxn}: Reaction heat rate

Q_{α}: Control limit of the Q statistic

r: Total number of inputs to the neuron

R: Number of principal and independent components and

S: Substrate concentration, Covariance matrix and Independent component matrix

s_f: Feed substrate concentration

T: Reactor temperature

t: Scores

T_f: Feed temperature of substrate

T_i^2: Hotelling's T squared for each batch

T_{lim}^2: T squared control limit

U: Control signal

V: Culture volume

v_k: Input to the transfer function of neuron

w_{kj}: Input weights to neuron

X: Biomass concentration and Unfolded data matrix

X^*: Centered and scaled data matrix

$\overline{X}$: Three-dimensional data matrix

x_j: Output values from the previous layer in the ANN

Y_k: Current value of the controlled variable at the sampling time

Y_{SP}: Set point of the controlled variable

λ_i: Eigenvalues of the covariance matrix

τ_I: Integral constant in the PI control

References

1.Birol G, Ündey C, Çinar A. (2002) A modular simulation package for fed-batch fermentation: penicillin production. Computers and Chemical Engineering 26:1553-1565.

2.Yoo C, Lee P, Vanrolleghem PA, Lee I (2004) On-line monitoring of batch processes using multiway independent component analysis. Chemometrics and Intelligent Laboratory Systems 71:151-163.

3.Tian X, Zhang X, Deng X, Chen S (2009) Multiway kernel independent component analysis based on feature samples for batch process monitoring. Neurocomputing 72:1584-1596.

4.Nucci ER, Cruz AJG, Giordano RC (2010) Monitoring bioreactors using principal component analysis: production of penicillin G acylase as a case study. Bioprocess Biosyst Eng 33:557-564.

5.Pau L (1981) Failure diagnosis and performance monitoring. Marcel Dekker, New York.

6.Himmelblau DM (1978) Fault detection and diagnosis in chemical and petrochemical processes. Elsevier, Amsterdam.

7.Kourti T (2002) Process analysis and abnormal situation detection: From theory to practice. IEEE Control systems magazine 22:10-25.

8.Cinar A, Parulekar S, Undey C, Birol G (2003) Batch Fermentation: Modeling, Monitoring and Control. Marcel Dekker, New York, NY.

9.Westerhuis J, Kourti T, MacGregor J (1999) Comparing alternative approaches for multivariate statistical analysis of batch process data. Journal of Chemometrics 13:379-396.

10.Dahl S, Piovoso M, Kosanovich K (1999) Translating third-order data analysis methods to chemical batch processes. Chemometrics and Intelligent Systems 46:161-180.

11.Nomikos P, MacGregor J (1995) Multivariate SPC charts for monitoring batch processes. Technometrics 37:41-59.

12.Nomikos P, MacGregor J (1994) Monitoring of batch processes using Multiway Principal Component Analysis. AIChE Journal 40:1361-1375.

13.Neogi D, Schlags CE (1998) Multivariate statistical analysis of an emulsion batch process. Ind Eng Chem Res 37:3971-4979.

14.Tates AA, Louwerse DJ, Smilde AK, Koot GLM, Berndt H (1999) Monitoring a PVC batch process with multivariate statistical process control charts. Ind Eng Chem Res 38:4769-4776.

15.Van Sprang E, Ramaker H, Westerhuis J, Gurden S, Smilde A (2002) Critical evaluation of approaches for on-line batch process monitoring. Chemical Engineering Science 57:3979-3991.

16.Ündey C, Tatara E, Çinar A (2003) Real-time batch process supervision by integrated knowledge-based systems and multivariate statistical methods. Engineering applications of artificial intelligence 16:555-566.

17.Venkatasubramanian V, Rengaswamy R, Yin K, Kavuri SN (2003) A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. Computers and chemical engineering 27:293-311.

18.Pokkinen M, Flores Z, Asama H, Endo I, Aarts R, Linko P (1992) A knowledge based system for diagnosing mocrobial activities during a fermentation process. Bioprocess and Biosystems Engineering 7:331-334.

19. Román RC, Hernández OG, Urtubia UA (2011) Prediction of problematic wine fermentations using artificial neural networks. Bioprocess Biosyst Eng 34.

20.Geladi P, Isaksson H, Lindqvist L, Wold S, Esbensen K (1989) Principal Component Analysis of Multivariate Images. Chemometrics and Intelligent Laboratory Systems 5:209-220.

21.MacGregor J, Nomikos P (1992) “Monitoring Batch Processes” in Batch Processing Systems Engineering: Current status and future directions (NATO ASI Series F) Reklaitis, Rippin, Hortacso and Sunol (eds). Springer-Verlag, Heidelberg.

22.Qin S (2003) Statistical process monitoring: basics and beyond. Journal of Chemometrics 17:480-502.

23.Harington J (1975) Clustering algorithms. Wiley, New York.

24.Aguado D, Ferrer J, Seco A (2007) Multivariate SPC of a sequencing batch reactor for wastewater treatment. Chemometrics and Intelligent Laboratory Systems 85:82–93.

25.Villez K, Steppe K, De Pauw DJW (2009) Use of Unfold PCA for on-line plant stress monitoring and sensor failure detection. Biosystems Engineering 103:23-34.

26.Yoo C, Lee D, Vanrolleghem PA (2004) Application of multiway ICA for on-line process monitoring of a sequencing batch reactor. Water Research 38:1715-1732.

27.Venkatasubramanian V, Chan K (1989) A neural network methodology for process fault diagnosis. AIChE J 35:1993-2002.

28.Kavuri S and Venkatasubramanian V (1993) Representing Bounded Fault Classes Using Neural Networks with Ellipsoidal Activation Functions. Computers and Chemical Engineering 17:139-163.

29.Hoskins JC, Himmelblau DM (1988) Artificial neural network models of knowledge representation in chemical enginnering. Computers and Chemical Engineering 2:881-890.

30.Watanabe K, Matsuura I, Abe M, Kubota M, Himmelblau DM (1989) Incipient fault diagnosis of chemical processes via artificial neural networks. AIChE J 35:1803-1812.

31.Ruiz D, Nougués JM, Calderón Z, Espuña A, Puigjaner L (2000) Neural network based framework for fault diagnosis in batch chemical plants. Computers and Chemical Engineering 24:777-784.

32.Haykin S (1994) Neural Networks, A Comprehensive Foundation. Macmillan College Publishing.

33.Leger R, Garland Wm, Poehlman W (1998) Fault detection and diagnosis using statistical control charts and artificial neural networks. Artificial Intelligence in Engineering 12:35-47.

34.Vapnik V (1999) The nature of Statistical Learning Theory. Springer, New York.

35.Vapnik V (1998) Statistical learning theory. Wiley, New York.

36.Yélamos I, Graells M, Puigjaner L, Escudero G (2007) Simultaneous fault diagnosis in chemical plants using a Multilabel approach. AIChE Journal 53:2871-2884.

37.Chiang L, Kotanchek M, Kordon A (2004) Fault diagnosis based on Fisher discriminant analysis and support vector machines. Computers and Chemical Engineering 28:1389-1401.

38.Kulkarni A, Jayaraman V, Kulkarni B (2005) Knowledge incorporated support vector machines to detect faults in Tennessee Eastman Process. Computers and Chemical Engineering 29:2128-2133.

39.Yunfeng L, Zhifeng W, Jingqi Y (2006) On-line fault detection using SVM-based dynamic MPLS for batch processes. Chinese Journal of Chemical Engineering 14:754-758.

40.Ramaker H, Van Sprang E, Gurden S, Westerhuis J, Smilde A (2002) Improved monitoring of batch processes by incorporating external information. Journal of Process Control 12:569-576.

41.Jolliffe IT (1986) Principal Component Analysis. Springer-Verlag, New York.

.

42.MacGregor JF, Kourti T (1995) Statistical process control of multivariate processes. Control Engineering Practice 3:403-414.

43.Jackson JE, Mudholkar GS (1979) Control Procedures for Residuals Associated with Principal Component Analysis. Technometrics 21:341-349.

44.Yoo C, Villez K, Lee I, Rosén C (2007) Multi-model statistical process monitoring and diagnosis of a sequencing batch reactor. Biotechnology and Bioengineering 96:687-701.

45.Montgomery DC (2005) Introduction to Statistical Quality Control. Wiley International Edition 5, USA.

46.Camacho J, Picó J, Ferrer A (2009) The best approaches in the on-line monitoring of batch processes based on PCA: Does the modelling structure matter?. Analytica Chimica Acta 642:59-68.

47.Jackson JE (1991) A User's Guide to Principal Components. Wiley, USA.

48.Kent A, Berry M, Luehrs F, Perry J (1955) Machine literature searching: VIII. Operational criteria for designing information retrieval systems. American Documentation 6:93-101.

49.Monroy I, Benitez R, Escudero G, Graells M (2010) A semi-supervised approach to fault diagnosis for chemical processes. Computers & Chemical Engineering 34:631-642.

50.Huang S-C, Huang Y-F (1991) Bounds on the number of hidden neurons in multilayer perceptrons. IEEE Trans On Neural Networks 2:47-55.