Real-time fault detection and isolation in biological wastewater treatment plants
F. Baggiani and S. Marsili-Libelli

ABSTRACT
Automatic fault detection is becoming increasingly important in wastewater treatment plant operation, given the stringent treatment standards and the need to protect the investment costs from the potential damage of an unchecked fault propagating through the plant. This paper describes the development of a real-time Fault Detection and Isolation (FDI) system based on an adaptive Principal Component Analysis (PCA) algorithm, used to compare the current plant operation with a correct performance model based on a reference data set and the output of three ion-specific sensors (Hach-Lange gmbh, Düsseldorf, Germany): two Nitratax® NOx UV sensors, in the denitrification tank and downstream of the oxidation tanks, where an Amtax® ammonium-N sensor was also installed. The algorithm was initially developed in the Matlab environment and then ported into the LabView 8.20 (National Instruments, Austin, TX, USA) platform for real-time operation using a compact Field Point®, a Programmable Automation Controller by National Instruments. The FDI was tested with a large set of operational data with 1 min sampling time from August 2007 through May 2008 from a full-scale plant. After describing the real-time version of the PCA algorithm, this was tested with nine months of operational data which were sequentially processes by the algorithm in order to simulate an on-line operation. The FDI performance was assessed by organizing the sequential data in two differing moving windows: a short-horizon window to test the response to single malfunctions and a longer time-horizon to simulate multiple unrepaired failures. In both cases the algorithm performance was very satisfactory, with a 100% failure detection in the short window case, which decreased to 84% in the long window setting. The short-window performance was very effective in isolating sensor failures and short duration disturbances such as spikes, whereas the long term horizon provided accurate detection of long-term drifts and proved robust enough to allow for some delay in failure recovery. The system robustness is based on the use of multiple statistics which proved instrumental in discriminating among the various causes of malfunctioning.

Key words | fault detection, principal component analysis, process control, programmable automation controllers, wastewater treatment

INTRODUCTION
Real-time monitoring is an increasingly important aspect in the domain of advanced control of Waste Water Treatment Plants (WWTP) (Olsson & Newell 1999; Olsson 2006; Olsson & Jeppsson 2006) in preventing the possible damage caused by a failure propagating through the plant. In particular, Fault Detection and Isolation (FDI) techniques are now emerging as the foremost feature in upgrading SCADA (Supervisory Control And Data Acquisition) systems. Traditionally, many popular FDI methods
relied on model-based approaches, but the present application is based on a data-based approach, due to the complexity of the mechanistic models associated with plant-sensor dynamics (Kettunen et al. 2008; Rosén et al. 2008).

This paper describes the design, implementation and assessment of a real-time fault detection system applied to a full-scale WWTP treating domestic sewage, run by Acque SpA, the water authority managing the integrated water cycle in the middle course of the river Arno, in central Italy. The study was promoted by Acque SpA in order to optimize their investment in sensors by complementing their maintenance programme with an early diagnosis and isolation of the faulty component. After the off-line testing described in this paper, the FDI system will be integrated into an already operational supervisory module (Marsili-Libelli & Maietti 2008) controlling the plant.

Structure of the WWTP and the monitoring system

The Pagnana WWTP

The study was carried out in the Pagnana WWTP, a full-scale activated sludge plant with a capacity of 88,600 PE, located at about 60 km from Florence and treating a mix of domestic sewage and septic tank discharges. The plant is conceived to provide biological nutrient removal through a pre-denitrifying tank followed by three parallel-fed oxidation tanks and three secondary settlers, as shown in Figure 1, which also indicates the sensor positions.

Monitoring and control hardware

To monitor the nitrogenous compounds along the plant, three ion-specific sensors (Hach-Lange gmbh, Düsseldorf, Germany) were installed: two Nitratax® NOx UV sensors, one in the denitrification tank and another at
the mixing tank downstream of the oxidation tanks, where an Amtax® ammonium-N sensor is also installed. The sensor outputs are conditioned by a SC-1000 instrument interface and then transmitted through individual 4–20 mA current loops to a compact Field Point® (cFP) (National Instruments, Austin, TX, USA) performing the typical SCADA functions, in addition to providing server functionalities for internet communication (National Instruments 2002) through a Virtual Private Network (VPN). The cFP and internet connection are shown in Figure 1 and the system architecture is fully described in Marsili-Libelli & Maietti 2008.

Principal components analysis (PCA)

One of the difficulties in implementing the FDI system is the nature of the WWTP data, which are redundant, non stationary, and often auto- and cross-correlated (Schraa et al. 2006). Therefore multivariate methods must be used to process the multivariable data and provide the plant manager with only the relevant information required to take critical decisions. Principal Components Analysis (PCA) is a widely used non-parametric tool for extracting information from a set of redundant and noisy data where the information may be masked by noise and data cross-correlation. Through PCA the data are transformed in order to remove correlations among variables and reduce their dimensionality without significant loss of information. Thus PCA can be viewed as a linear data transformation which removes the three main error sources: inadequate reference, producing cross-correlations, noise, and redundancy (Dillon & Goldstein 1984; Dunia & Qin 1998; Rosén & Olsson 1998; Rosén & Lennox 2001; Jolliffe 2002; Lennox & Rosén 2002). Of these three aspects, the first two are prominent here, whereas redundancy is not a major issue now, with only three sensors being considered, but will become important in the future when many more sensors will be added to the system.

Assuming that \( p \) sensors generate \( n \) data each, the data can be grouped in a matrix \( X \in \mathbb{R}^{p \times n} \), after normalization, i.e. reducing the data to zero mean and unit variance. The variance-covariance matrix \( C_X \in \mathbb{R}^{n \times n} \) is then computed as \( C_X = \frac{1}{p-1}XX^T \). Being a symmetric matrix, \( C_X \) is diagonalized by the orthogonal matrix of its eigenvectors \( T = \begin{bmatrix} t_1 & t_2 & \ldots & t_n \end{bmatrix} \), also called loadings, therefore the principal components of \( X \) are the columns of \( T \) and the corresponding eigenvalues \( \lambda_i, i = 1, \ldots, n \) represent the variance of \( X \) along each principal component \( t_i \) (Dillon & Goldstein 1984; Jolliffe 2002). In this new reference the transformed data, also called scores, are represented by

\[
Z = XT
\]

with a diagonal covariance matrix \( C_Z = \frac{1}{p-1}ZZ^T = \frac{1}{p-1} \text{diag} \{ \lambda_1, \lambda_2, \ldots, \lambda_n \} \). Further, the transformed dimensions can be classified in order of importance by sorting the columns of \( T \) according to the relative magnitude of the eigenvectors, starting with \( t_1 \) corresponding to the largest eigenvalue \( \lambda_1 \) which explains the largest fraction of the data variance and with the other columns corresponding to eigenvectors of decreasing importance. The ranking of \( T \) may lead to the dimension reduction of the data space by retaining a number of components \( a < n \), in which case the original data matrix can be decomposed as

\[
X = Z_aT_a + Z_{n-a}T_{n-a} = Z_aT_a + E,
\]

where \( Z_a \) is formed by the first \( a \) rows of \( Z \) and the \( T_a \) matrix retains only the first \( a \) column of \( T \), corresponding to the first largest \( a \) eigenvalues \( \{ \lambda_1 > \lambda_2 > \ldots > \lambda_a \} \). The residual matrix \( E \) contains the components corresponding to the least significant eigenvalues. The selection of \( a \) will be discussed in the application section. On the contrary the variance reduction feature described earlier was the key to the real-time fault detection and isolation algorithm described in this paper.

Development of a fault detection and isolation (FDI) algorithm based on PCA

A successful fault detection and isolation (FDI) procedure must be capable of detecting malfunctions in real-time, meaning not only that the detection must be timely, but also that the algorithm must keep track of the time-varying condition of the process (Ku et al. 1995; Lee et al. 2004; Lee et al. 2006). The present data-based algorithm differs from the knowledge-based approach in the sense that a set of correct performance data represent the reference model, in place of more sophisticated process description such as
multiway principal component analysis and clustering (Villez et al. 2008a), qualitative representation of trends (Villez et al. 2008b) or dynamical model representation, as in the Benchmark approach (Flores-Alsina et al. 2009), for which an articulated fault classification method was recently proposed (Corominas et al. 2009). The latter approach, however, is conceived mainly for benchmark simulation testing and relies on complex sensor and actuators models (Rosen et al. 2008), which is difficult to apply to real sensors.

In the context of this data-based approach, PCA is used to detect the departure of operational data from the correct performance, previously defined on the basis of data unequivocally representing a well-behaved operating condition. Whenever the data consistently depart from this region, a fault is recognised. Multivariable data may be processed in this way, making the procedure highly parallel. For this reason it has been extensively applied to the monitoring of wastewater processes (Rosen et al. 2003; Li & Rong 2006; Aguado & Rosen 2007; Detroja et al. 2007; Mina & Verde 2007). The algorithm described here is composed of the following steps:

1. Selection of a correct performance reference data set, when the condition of all the sensors and processes is nominal. This data set will then be used as the reference data model;
2. Real-time data updating through a moving window, to obtain a real-time procedure;
3. PCA application to the real-time data set and selection of correct performance thresholds based on appropriate statistics;
4. Real-time comparison of PCA results with the selected statistical thresholds, in order to obtain timely information about the cause and nature of the malfunction.

Selection of the reference data set

Searching the historical data base of the plant, a set of operational data positively corresponding to a nominal condition of sensors and processes is selected and used to define the reference correct performance dataset, based on which the statistical thresholds are computed. Two aspects were considered in selecting the reference data size: a large data set increases the detection reliability, whereas a short data set improves the computational speed. A compromise was found between these two requirements, determining a data set large enough to allow a positive comparison without impairing the real-time computation efficiency. The size of the data set reflected in the length of the moving window selecting the amount of data to be processed in real-time. The evolution of the moving window from its initial position is shown in Figure 2. The window is 19,830 samples long, corresponding to approximately one week of operation at 1 min sampling period. During the real-time operation, the window still maintains the initial length of 19,830 samples and operates as a First-In-First-Out (FIFO) shift-register, discarding old data and including new ones only if they fall in the correct performance set. Since not all the data may be retained in the window, there may be time gaps, as shown in Figure 2.

Error statistics and their limits

Two error statistics largely used in statistical process control (Oakland & Porter 1995; Montgomery 2009) were used: Hotelling $T^2$ and $Q$. The first is a generalization of the $t$-Student statistics and captures the variations in the reference data model. At the generic $k$-th sample, $\mathbf{x}(k)$ is the measurement vector. The PCA analysis yields the eigenvector matrix $\mathbf{T}_k$ and the diagonal matrix of its eigenvalues $\Lambda_k = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)|_k$ from which the $T^2$ statistics is computed

$$T^2(k) = \mathbf{x}^T(k)\mathbf{T}_k\Lambda_k^{-1}\mathbf{T}_k^TX(k).$$

This is normally complemented with the $Q$ statistics, defined as the sum of squared residuals of the active

![Figure 2](image-url)
principal components

\[ Q(k) = x^T(k)(I - TT^T)x(k), \]  

which can be used to gauge the variation in the residual space not accounted for by the PCA. For the \( T^2 \) statistics an upper bound can be defined in relation to multivariable control charts

\[ T_{lim}^2 = \frac{a(p - 1)}{p - a}F_{a,p-a,a}, \]  

where \( a \) is the number of the retained principal components, \( p \) is the window length and \( F_{a,p-a,a} \) is the \( \alpha \)-percentile Fisher \( F \) statistics with \( (a, p - a) \) degrees of freedom. The threshold for the \( Q \) statistics is given by

\[ Q_{lim} = \left( \frac{h_0c_a\sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2h_0(h_0 - 1)}{\theta_1^2} \right)^{(1/h_0)}, \]  

where \( c_a \) is the \((1 - \alpha)\) quantile of the normal distribution and the following quantities

\[ h_0 = 1 - 2 \frac{h_0^{I/a}}{\pi}, \quad \theta_1 = \sum_{i=a+1}^n \lambda_i, \quad \theta_2 = \sum_{i=a+1}^n \lambda_i^2, \quad \theta_3 = \sum_{i=a+1}^n \lambda_i^3 \]  

are defined with respect to the eigenvalues previously determined.

The initial Hotelling \( T^2 \)-test is computed for the correct performance reference data. If \( T^2 \) and \( Q \) exceed the thresholds of Equations (5–6) the data reliability should be checked. Outliers should be eliminated and unusual but possible data should be retained to improve the algorithm future performance (Rosén 1998; Rosén & Lennox 2001).

**Real-time operation**

It is assumed that fresh samples are continuously fed into the algorithm at 1 min intervals. Since the FDI is tested

![Figure 3](image-url)

*Figure 3* | The three main phases of the FDI algorithm. The real-time operations are implemented in the Phase 3 block.
off-line, real-time operation is simulated by feeding the data sequentially one at a time. Each new sample $x(k)$ is projected unto the reference PCA space as

$$z(k) = x(k)T_k$$

and a new PCA is performed. The contribution of each process variable to the $T^2$ and $Q$ statistics is then computed as (Lee et al. 2004)

$$CV(k) = z(k)\Lambda^{-1/2}T_k^T = x(k)T_k\Lambda^{-1/2}T_k^T$$

In order to keep the data dimensions constant, the basic PCA algorithm was made dynamic by updating the data base through a moving window of fixed length $m = 19,830$, as already shown in Figure 2. On-line operation was simulated by sequentially feeding the data into the algorithm, which uses the data only up to the current timeline. In the simulated real-time operation two differing updating modes of the moving window were considered:

(a) **Short window**: the updating process is limited to approximately one week. This window is aimed at assessing the FDI response to a single or a very limited number of malfunctions, assuming that each fault is immediately repaired;

(b) **Long window**: The updating is extended for a longer period, between one and two months running, but still retaining the fixed length of 19,830 samples. This setting is the most demanding for the FDI algorithm because it simulates the occurrence of multiple unrepaired failures which continue to affect the subsequent data.

At each step a new sample is acquired, the whole data matrix is re-normalized and the $T^2$ and $Q$ statistics are re-computed. The new sample is retained if and only if its statistics do not exceed the updating thresholds heuristically defined as a function of the limits defined by Equations (5–6) according to

$$T^2_{\text{update}} = \frac{1}{{10}}T^2_{\text{lim}}Q_{\text{update}} < Q_{\text{lim}}.$$  \hspace{1cm} (10)

If the new sample is below these updating values, it is retained while the oldest in the window is discarded. Otherwise, it is simply dropped and the algorithm moves one step ahead. For this reason there may be gaps in the moving window data, as shown in Figure 2. Table 2 shows that there is a 15% performance degradation going from the short window to the long window mode. This degradation can be regarded as a consequence of the unrepaired faults, which cause the rejection of many samples and decrease the efficiency of the detection algorithm.

### Control charts

The information obtained from the PCA algorithm is graphically presented in the following forms:

- $T^2$ and $Q$ graphs: The time plots of these statistics show the departure from the normal condition caused by one or more reference variables. Warning and alarm

---

**Table 1** | Percentage variance explained by each principal component computed with the reference data of Figure 4

<table>
<thead>
<tr>
<th>Principal component</th>
<th>Eigenvalues</th>
<th>Percent explained variance</th>
<th>Cumulative percent variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.685</td>
<td>56.2</td>
<td>56.2</td>
</tr>
<tr>
<td>2</td>
<td>0.832</td>
<td>27.7</td>
<td>83.9</td>
</tr>
<tr>
<td>3</td>
<td>0.484</td>
<td>16.1</td>
<td>100.0</td>
</tr>
</tbody>
</table>
thresholds can be set to discriminate differing degrees of severity of the malfunction;

- **Score plots**: Each point in the selected principal components space represents the values of the last retained sample. During normal operation, the centre of gravity of the plot is located near the origin, given the initial standardization. Any disturbance or fault will distort the score plot by moving the most recent samples away from the correct performance region, which can be defined in statistical terms by tracing the $i$-th confidence limit as a normal operation threshold, defined as

$$
\delta_{i,a} = \sqrt{\lambda_i I_{p-1,a/2}},
$$

where $\lambda_i$ is the $i$-th eigenvalue, $I_{p-1,a/2}$ is the $\alpha$-percentile of the Student-t distribution and $p$ is the number of samples;

- **Contribution plot**: The contribution of each variable to the malfunction, as defined by Equation (9) can be represented by a set of bars, thus visually indicating the relative contribution of each sensor to the fault.

### Structure of the FDI algorithm

The FDI algorithm was designed to monitor the nitrogen transformations in the wastewater treatment process and alert the operator whenever a malfunction would occur. The algorithm was initially developed and tested in the Matlab® environment (The Mathworks, Natick, MA, USA) and later ported into the N.I. LabView 8.20® platform (National Instruments, Austin, TX, USA) for real-time operation using a National Instruments compact Field Point® as the host hardware, which was also used for plant control. The previously described dynamic PCA was implemented using the N.I. Advanced Signal Processing Toolkit®, a LabView application add-on enhancing its signal processing capabilities.

The FDI algorithm is divided into three main functional blocks shown in Figure 3, which correspond to the following functional phases:

1. **Phase 1**: Data acquisition and processing for the reference period;
2. **Phase 2**: Computation of the initial reference space and of the $T^2$ and $Q$ statistics;

In the first phase the *correct performance* data are loaded and standardized. The $\Lambda^{-1}$ matrix of Equations (3) and (9)...
is computed, to be used in the second phase, where the Advanced Signal Processing toolkit is used to compute the scores, the loadings, and the variances of each principal component in the reference window. With these values the $T^2$ and $Q$ statistics and their thresholds of Equations (5–6) are computed for the initial correct performance data, which are then used to initiate the real-time operation (phase 3) where the moving window is set-up. Each new sample is standardized and projected onto the reference space in order to obtain its scores and the statistics values. Comparing these values with the reference thresholds the occurrence of faults is detected. The reference update depends on the comparison between the last sample and the update thresholds of Equation (10). If the former is below the thresholds, the reference data are updated and a new PCA is performed, otherwise the new sample is analyzed but the database remains unchanged.

**ASSESSMENT OF THE ALGORITHM**

The sensors considered in the FDI algorithm, already described in the previous section and deployed in the plant as shown in Figure 1, are:

1. $\text{NH}_4^+$ at the oxidation tank outlet
2. $\text{NO}_x$ at the outlet of the denitrification tank ($\text{NO}_x^{\text{DEN}}$)
3. $\text{NO}_x$ at the outlet of the oxidation tank ($\text{NO}_x^{\text{OX}}$)

The results presented here assess the performance of the FDI algorithm of Figure 3 in the operational period August 2007 – May 2008. Applying PCA to the initial correct performance reference data shown in Figure 4, the scores of Table 1 were obtained and it was concluded that retaining the first two principal components explained 84% of the observed variance, hence a second order reference data model was adopted throughout. The reference data were used to compute the thresholds for the $T^2$ and $Q$ statistics. Selecting a confidence level $\alpha = 0.05$ resulted in $T^2_{\text{lim}} = 6.0$ and $Q_{\text{lim}} = 3.2$. The real-time operation was simulated by sequential data processing with the two windows previously described.

**Short window**

During a correct performance operation the $T^2$ and $Q$ statistics produce very low values, about one half of the threshold, and remain in this range as long as all the sensors are nominal, as shown in Figure 5. Starting from this normal behaviour some diagnostic episodes are now examined.
Sensor fault

Figure 6 shows a NH$_4^+$ sensor fault in the oxidation tank, observed in December 2007. The FDI algorithm produces a major jump of both $T^2$ and $Q$ statistics when the failure is detected. Though these indicators later decrease, they never return below their thresholds and the failure condition is maintained because the sensor causing the faulty condition is not repaired. Further, the contribution plot, shown in the lower part of the figure, indicates the NH$_4^+$ sensor as the one responsible for the malfunction, being the major contributor to the increased statistics. It could be argued that a simple rule, monitoring the NH$_4^+$ sensor output could do the job equally well, but the algorithm can detect more complex faults and, thanks to the contribution plots, can assist the operator with a centralized information about the state of the plant. This will become important when more sensors are connected to the FDI and concurrent failures are tackled. Further, as shown later in Figure 8, the FDI automatically reverts to normal when the fault is repaired, or more generally when the alarm condition ceases.
This term indicates a sudden sensor output, which is not compatible with its normal output range. Figure 7 shows two spikes generated by the NO\textsubscript{X\textsubscript{DEN}} sensor at the output of the denitrification tank. The negative values produced by the sensor during these events were caused by an abrupt electrical disturbance along the transmission lines. These disturbances are too short to be detected by the \(T^2\) and \(Q\) statistics and the contribution plot is instrumental in discriminating them from other more serious faults and in indicating the responsible sensor.

Process anomaly

The previous examples considered the outright failure of a sensor or the sudden sporadic occurrence of wild data. However there may be other, more subtle reasons for a performance downgrade, even if this does not culminate in a total lack of signal. A nitrification problem (NH\textsubscript{4}\textsuperscript{+} increases at the same time when NO\textsubscript{2} decreases), shown in the left part of Figure 8, is detected as a generic malfunction. The \(T^2\) statistics does not have the capability to discriminate between sensor and process failure, but an alarm is generated anyway. Though some kind of information is transmitted by the sensor during the nitrification problem, the FDI algorithm considers this as a fault and the contribution plot indicates that the NO\textsubscript{X\textsubscript{DEN}} is the primary cause of the anomaly. By contrast, a clear sensor failure is shown in the right part of the same figure for comparison. In this episode the NH\textsubscript{4}\textsuperscript{+} sensor is clearly indicated as the main responsible by contribution plot, with the \(T^2\) statistics returning to normal as soon as the sensor is repaired.

Long window

This analysis is based on a moving window extending over a period of approximately two months and is intended to test the algorithm when the occurrence of repeated failures causes frequent gaps in the updating process. In this case the detection efficiency is somewhat less than with the short window, especially relying only on the \(T^2\) and \(Q\) statistics, as shown in Figure 9, whereas the scores clearly depart from the correct performance operating region when the fault sets in, as shown in Figure 10.
Overall performance assessment

The algorithm performance over a nine months time span is summarized in Table 2. The faults actually occurring on the plant were recorded and compared with those detected by the FDI algorithm with either short (one week) or long (two months) moving window. The short window has the purpose of testing the algorithm on each single fault, which is supposed to be promptly repaired, whereas in the long window the FDI performance can be assessed in the case of multiple unrepaird faults, causing repeated gaps in the PCA updating process and consequent statistics downgrading. In the short window case a perfect performance is obtained, with 100% of the actual faults correctly detected. In the long window case, the performance is somewhat degraded, with a detection rate of 84%. This is due to the long window causing a less efficient updating and the coexistence of multiple faults in the moving data set.

| Table 2 | Overall performance assessment: comparison between observed and detected faults, both in the short and long data window |
|-----------------|-----------------|-----------------|-----------------|
| Aug. 07 Observed faults | 2 | 4 | 1 |
| Detected faults Short window | 2 | 4 | 1 |
| Long window | 2 | 3 | 1 |
| Sept. 07 Observed faults | 4 | 0 | 1 |
| Detected faults Short window | 4 | 0 | 1 |
| Long window | 3 | 0 | 1 |
| Oct. 07 Observed faults | 4 | 6 | 3 |
| Detected faults Short window | 4 | 6 | 3 |
| Long window | 4 | 4 | 2 |
| Nov. 07 Observed faults | 1 | 4 | 0 |
| Detected faults Short window | 1 | 4 | 0 |
| Long window | 1 | 3 | 0 |
| Dec. 07 Observed faults | 2 | 20 | 2 |
| Detected faults Short window | 2 | 20 | 2 |
| Long window | 2 | 15 | 1 |
| Jan. 08 Observed faults | 6 | 20 | 1 |
| Detected faults Short window | 6 | 20 | 1 |
| Long window | 5 | 16 | 1 |
| Mar. 08 Observed faults | 2 | 55 | 1 |
| Detected faults Short window | 2 | 55 | 1 |
| Long window | 2 | 46 | 1 |
| Apr. 08 Observed faults | 3 | 4 | 0 |
| Detected faults Short window | 3 | 4 | 0 |
| Long window | 2 | 3 | 0 |
| May 08 Observed faults | 5 | 0 | 0 |
| Detected faults Short window | 5 | 0 | 0 |
| Long window | 4 | 0 | 0 |
| Total percentage of detected faults (Short window) | 100% |
| Total percentage of detected faults (Long window) | 84% |
CONCLUSION

This paper has described a real-time Fault Detection and Isolation (FDI) algorithm based on the Principal Component Analysis (PCA) and related statistics. After briefly recalling the underlying theory and describing the plant producing the data used to test the algorithm, the paper has described the real-time features of the algorithm where a block of time-varying process data, organized in a moving window of prescribed length, are used to detect the fault occurrence and adapt the detection thresholds, provided that certain safety bounds are satisfied. The organization of the real-time algorithm is described in Figure 3: the first and second phases initialize the algorithm by selecting a reference correct performance behaviour, whereas the third describes the real-time operation proper, with the arrangement of data in a moving window, the statistics updating and the check of the alarm thresholds.

The algorithm was tested by processing the data sampled at 1 min intervals from three nitrogen sensors in a conventional pre-denitrifying wastewater treatment plant with a capacity of 88,600 PE, during a period of nine months from August 2007 to May 2008. In order to simulate the real-time operation, the data were sequentially fed into the algorithm, which kept in memory the last 19,830 representative samples, organized in two kinds of moving windows: a short one-week window, suitable to check the algorithm reaction to a single malfunctioning, and a long two-months window, to test the robustness of the algorithm to repeated, unrepaired faults, causing substantial breaks in the FDI updating process. While in the first case all the faults actually occurring in the plant were correctly detected, in the long window case the detection percentage decreased to 84% due to diminished updating caused by more faulty data in the moving window. The algorithm proved also capable of dealing with three possible kinds of faults: outright sensor failure, e.g. when the data transmission was interrupted by a power loss or a disconnected cable, spikes, defined as a short burst of data outside the reasonable sensor operating range, and process anomalies, when a sensor produced an unexplained drift. In the latter case the basic error statistics could only indicate the occurrence of a fault, but the contribution plots were instrumental in indicating which sensor was responsible for which kind of fault and proved very useful in the long-term operation.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the cooperation of Acque SpA and Acque Engineering srl for their support during this research, for providing technical assistance on the plant and for granting access to the company information system throughout the testing of the algorithm. The invaluable collaboration of National Instruments Italy is also acknowledged for providing a complimentary compact Field Point unit and competent engineering assistance during this research.

REFERENCES

National Instruments 2002 Taking your measurements to the web with LabVIEW. White Paper 2218.