Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Extended Luenberger Observer-Based Fault Detection for an Activated Sludge Process

F. Nejjari * V. Puig L. Giancristofaro S. Koehler

* *Universitat Politecnica de Catalunya UPC*
Automatic Control Department ESAII, 10, Rambla Sant Nebridi,
08222 Terrassa, Spain
(e-mail: fatiha.nejjari@ upc.edu).

Abstract: In this paper, an observer-based fault detection and isolation (FDI) method for a biological wastewater treatment process (WWTP) is presented. The residual is generated utilizing an Extended Luenberger function observer. The FDI of a set of sensors faults is done by using a bank of Luenberger observers. The implementation of the proposed approach and the results obtained from its application to the WWTP demonstrate its simplicity and effectiveness.

1. INTRODUCTION

The increasing complexity of modern engineering systems has motivated the development of different fault detection and isolation (FDI) approaches for the purpose of supervision. This development has been demonstrated by a large number of publications (Frank (1996); Isermann (1993); Blanke et al. (2003); Gertler (1998) and Chen and Patton (1999)).

Nowadays, the field of model-based FDI is well-studied, there exists a wide variety of model-based FDI approaches for linear and non linear systems, e.g. the observer-based approach, the parity space approach, and the parameter estimation approach. Key references of model-based FDI can be found in Chen and Patton (1999); Gertler (1998); Patton and Chen (1997); Frank (1996); Willsky (1976); Garcia and Frank (1997); Blanke et al. (2003) and Chen and Patton (1999). Especially the observer-based approach has gained a lot of interest recently (Garcia and Frank (1997); Frank et al. (1999); Hammouri et al. (1999); Nijmeijer (1999); DePersis and Isidori (2000)).

The main task of observer-based FDI approach is to design an observer structure that generates structured residuals that enable detection and isolation of the considered faults. In order to isolate faults a set of residuals with different fault sensitivity should be designed using a bank of observers. Each residual is formed through an observer and will be robust to a specific set of faults but sensitive to other faults.

There exist many different observer-based approaches considering linear systems and different classes of nonlinear systems. Most of these methods only handle a specific class of nonlinear FDI problems. The extension of the existing results of linear FDI to the nonlinear case is not an easy task. With the application of nonlinear observer theory some results have been obtained, principally in the detection and, with some restrictions, also in the isolation of faults. Some problems taking into consideration more general models as well as the design of the corresponding nonlinear observers are still opened, because of the difficulties of estimating the state or the measurement vector of a nonlinear system, even if the nonlinearities are known and no disturbances are present.

During the operation of an industrial plant (e.g., a biological wastewater treatment process), many disturbances and faults can occur. The nature of these changes can be either sudden or slow and they can be related to normal or faulty process (and/or instrumental) operation, provoking real or apparent deviations from the normal operation.

Most bio-chemical processes are highly complex systems, with a great number of components interacting to achieve the system's purpose. In these systems, all components are related in a complex manner, which means that a fault in one component can often cause the failure of the entire system. To prevent this event, it is essential to detect faults immediately in order to enable the controlling system to take actions, so that the system can still fulfill its purpose.

In the last decades, biological treatment processes have proven to be an effective way to deal with polluted wastewater.

In particular, activated sludge is one of the most used for this purpose (Henze et al. (1987)). The main feature of this process includes degradation of influent biodegradable pollutants, containing both organic carbon and nitrogen by use of microorganisms. The organisms form flocs which are separated from the treated wastewater by means of gravity settling. A portion of the activated sludge settled is wasted while a large fraction is recirculated back to the reactors to maintain the appropriate substrate-to-biomass ratio (Nejjari et al. (1999)).

This paper focuses on observer-based FDI for nonlinear systems. The observer-based residual generation, using Extended Luenberger observers is applied to an activated sludge process (Nejjari and Quevedo (2002))focussing on detecting and identifying sensor faults.

We consider only additive faults and assume that a fault can occur in one sensor at a time. This means that, if there is a fault, we are lacking just one correct measuring signal. The outline of this paper is as follows. The problem of observer-based fault detection is formulated in section 2. The implementation of the bank of observer-based FDI is presented in section 3. Simulation results are given in section 4 and some concluding remarks end the paper.

2. OBSERVER-BASED FAULT DETECTION

Fault detection has two main objectives: Detection and Isolation. Detection consists of producing a signal from which we can obtain information whether a fault is present or not. Isolation consists of determining in which components the fault is present. The most widely used approach to generate diagnostic signals (residuals) are observers. The basic idea of the observer-based FDI consists in estimating the outputs of the system from the measurement using an observer and then constructing residuals by properly weighted output's estimation errors. One specific diagnostic signal must be generated per each fault to be detected, each diagnostic signal being sensitive only to one particular fault.
In a plant, faults can occur either in the main processing equipments (variation in process parameters) or in the auxiliary equipments (bias or drift in sensors, actuators, controller outputs etc). In the case of actuator faults, we lose the ability to control the system through one of the actuators. Sensor fault reduces the reliable measurement information through a loss of a sensor, making the system less observable while a fault in the process component changes the behavior of all the plant.

2.1 Extended Luenberger-based FDI

Process modeling including process equations that are shown in appendix A and the faults that can occur in the plant can be rewritten in the following form:

$$\dot{x} = f(x,u) + Lv \tag{1}$$

$$y = Cx + Mv + w \tag{2}$$

In these equations the k-dimensional vector v represents in its component all the possible faults and matrices L and M represent the relationship between a fault occurrence and the system dynamics and outputs. We assume that actuators and model faults affect the system dynamics and sensor faults affect the measurement. So the matrices L and M can be partitioned as follows:

$$L = [L_1 \quad L_2] \tag{3}$$

$$M = [M_1 \quad M_2] \tag{4}$$

L_1 has the same number of columns as M_1, L_2 has the same number of columns as M_2. L_2 and M_1 contain only zeros.
We want to realize observer-based fault detection, that is based on the comparison of measured and estimated outputs, using a bank of observers. This bank consists of m observers (in this case m=3), where every observer is fed with p (in this case p=2) different process outputs. To realize observer-based fault detection we need to modify the observers, adding an output that we will call residual:

$$\dot{\hat{x}} = f(\hat{x},u) - K(y - C\hat{x}) \tag{5}$$

$$r = y - C\hat{x} \tag{6}$$

This output is the error between the measurement and its estimation. Based on this new set of signals, we can now start the detection and identification process.
To achieve alarm generation it is better to work with digital signals. Hence, for each residual we add a binary variable, building a fault code vector with elements:

$$e_i(t) = \begin{cases} 1 \text{ if } & |r_i(t)| > k_i \\ 0 \text{ else} \end{cases} \tag{7}$$

(8)

where i takes values from 1 to s if we have s residuals. Of course s will be minor or equal to the product of the number of outputs and the number of observers we have.

2.2 Possible hypothetical faults

In order to test the fault detection system, we need to implement artificial yet realistic faults. In this case we are limiting the type of faults to simple additive faults occurring in the sensors taking the measurements. A more general case (discriminate also actuator and model faults) would be far more complex and this simple case will be enough to illustrate the use of observer-based fault detection techniques. Thus, in this particular case, the above defined L matrix will be all zeros and the faults will affect the outputs but not the dynamic evolution of the process. So the simplified detection scheme should be similar to the one in Fig. 1.

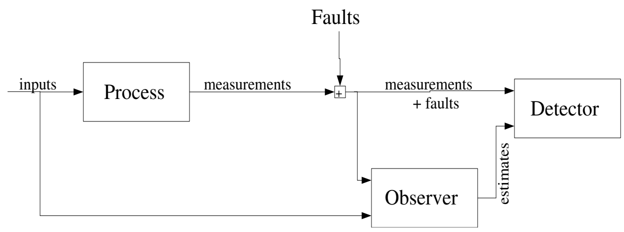


Fig. 1. Fault detection scheme

We additionally simplify the problem supposing that only one sensor at once can present a fault.

In real cases faults can often be represented as an output percentage. So the output with fault will be as follows:

$$y_{f_i}(t) = y_i(t) - f_i y_i(y) = (1 - f_i) y_i(t) \tag{9}$$

Where f_i is a real number such that $0 < f_i \leq 1$. We will test the observer on these particular fault occurrences.

3. IMPLEMENTATION OF OBSERVER-BASED FAULT DETECTION

To detect sensor faults in a system, it is necessary to implement more than just one observer. We chose three measurable signals to implement a bank of observers containing three Extended Luenberger Observers (ELO). Based on the output signal of this bank of observers we then realized the detection of faults. Both implementation steps are further explained in the following subsections.

3.1 Bank of Observers

We simply repeat twice the steps we made to build the Extended Luenberger Observer and calculate the gain matrices imposing the same minimal convergence speed for each observer. Matrix C changes depending on which two of the three measured output signals (S_s, S_o or X_s) the observer is using.

The implementation of the bank of observers was then realized in Simulink and shown in Fig. 2 and Fig. 3.

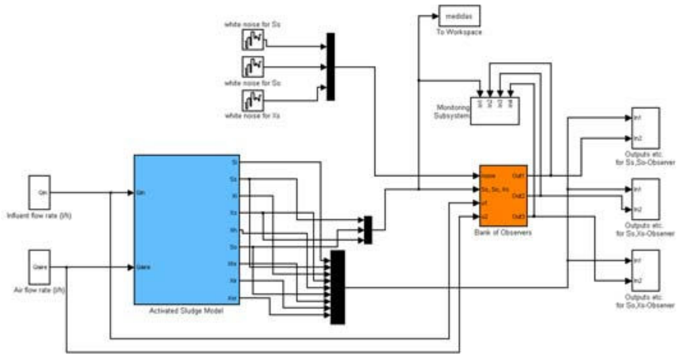


Fig. 2. First layer of Simulink implementation of the bank of observers

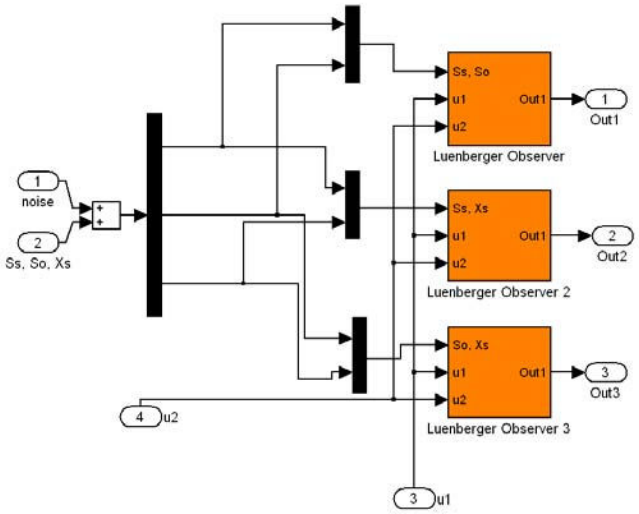


Fig. 3. Simulink implementation of bank of observers

Inside each block (subsystems called *Luenberger Observer 1-3* in Fig. 3) there is an Extended Luenberger Observer. Each Extended Luenberger Observer implemented in the bank of observers uses two of the three measured signals (S_s, S_o and X_s) to observe the system. We call O_1 the observer that uses S_o and X_s, O_2 the one that uses S_s and X_s and O_3 using S_s and S_o.
With these three observers we can generate nine residuals.

3.2 Fault Detection

The Implementation of the fault detection system in Simulink can be seen in Fig. 4. The subsystem in Fig. 4 called *Detection System* is further shown in Fig. 5.

Thanks to the bank of observers implemented, we can now generate the residuals that allow the detection of faults.

We will name $r_{i,j}$ the residuals of the measured system output j and its estimation using the observer i thus calculating each residual

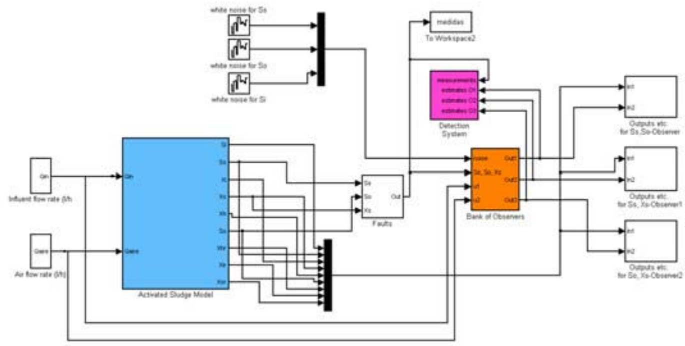


Fig. 4. First layer of Simulink implementation fault detector

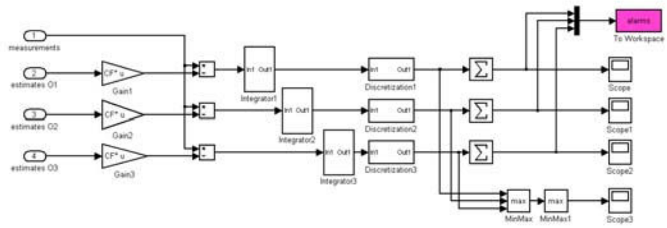


Fig. 5. Simulink implementation of the actual fault detection subsystem

$$r_{i,j} = y_{m,j} - y_{e,i,j}$$

where $y_{m,j}$ is the measured system output and $y_{e,i,j}$ the estimation made by the observer i for measurement j. Once generated, each residual is integrated in order to obtain its average value to use the resulting signal to generate the fault vector. As we did with the residual, we can arrange the fault vector as a matrix $\mathbf{E}$, where

$$E_{i,j}(t) = \begin{cases} 1 \text{ if } & |R_{i,j}(t)| > k_{i,j} \\ 0 \text{ else} & \end{cases} \tag{10}$$

$R_{i,j}(t)$ is the integrated signal. We found the limits k by observing the generated signals when there are no faults in the plant or the sensors and taking their maxima.

Running the system with a fault in output k, we will have an error signal different from zero if the used observer uses that output ($j \neq k$) or if the estimated variable is k. So we will obtain just two zero values in the $\mathbf{E}$ matrix. Note that we can determine in which measured output signal the fault is occurring just by knowing how many zeros there are in each row of this matrix.
Thus we can define another error signal:

$$s_j(t) = \sum_{i=1}^{r} E_{i,j}(t) \qquad j = 1, 2, 3 \tag{11}$$

If a fault occurs, we can determine that it is in component k if

$$s_k(t) = 1 \tag{12}$$

$$\text{and} \qquad s_j(t) = 3 \quad \text{for} \quad j \neq k. \tag{13}$$

In this way we reduce the signal we have to analyze to generate alarms to just 6 bits, 2 for every observer (we only need two bits to represent numbers from 0 to 3), while with the **E** matrix we needed 9 bits, one for each component.

Because of process noise and slow signal convergence these signals are not constant. Therefore, to have a more robust detector, we changed these conditions to:

Decision Rule:

There is a fault in the measured system output k if

$$\max s_j(t) = 2 \tag{14}$$

and

$$s_k(t) = \min s_j(t) \tag{15}$$

for at least one time interval d.

4. SIMULATION RESULTS

First of all we need to calculate the limits $k_{i,j}$ for the residuals. Instead of working with residuals we will calculate these values for the integrated residuals, in order to eliminate noise and take the DC component off the signal. We wrote some lines of code to find these values. $sig1$, $sig2$ and $sig3$ are three-dimensional, time-dependent vectors that contain the integrated residuals from each of the observers:

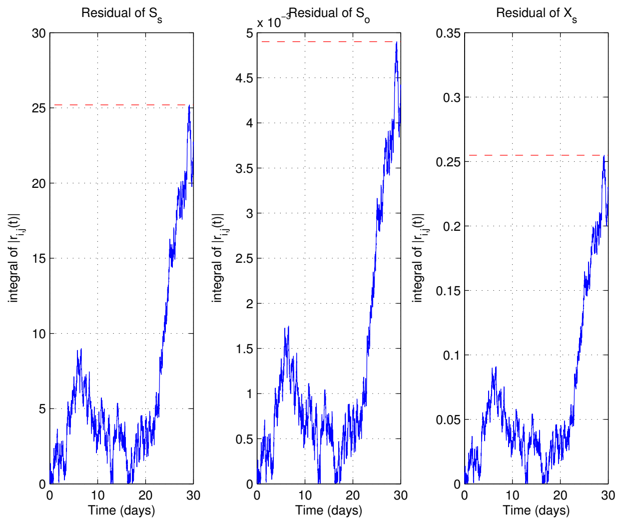
```
limits(1,:)=max(abs(sig1));
limits(2,:)=max(abs(sig2));
limits(3,:)=max(abs(sig3));
```

The result is a 3x3 matrix, whose (i, j) element contains the $k_{i,j}$ limit. In these lines we simulated that no faults were occurring. This process is graphically shown in Fig. 6(a)-(c), where subfigure (a) shows the thresholds for observer O_1, (b) those for O_3 and (c) for O_2.
Note that it is not necessary to take higher values for the limits, because we will just generate the alarm when the alarm status is present for more than one time interval d, so we have a good guarantee that no alarm is produced when there is no fault in the sensors.

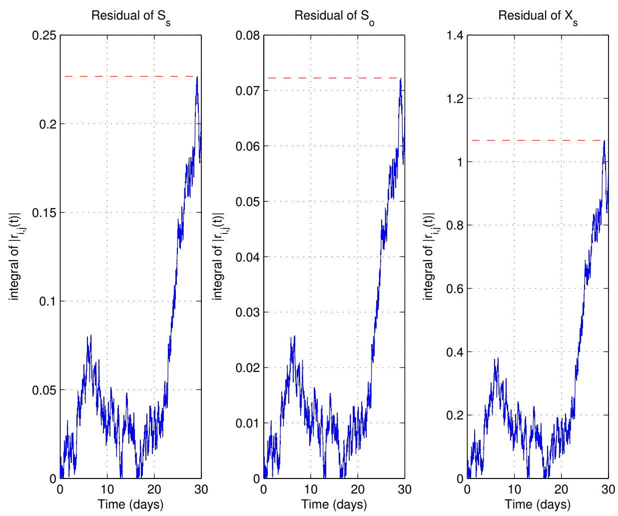
Then we simulated three fault occurrences (one in each of S_s,S_o and X_s) and observed the changes in the graphics. Fig. 7 shows the case of a 67% fault of the S_s sensor occurring at the 15th day. The fault is detected at 16.4 (days), when at least two residuals of the first observer and one of the second have reached the limit. So we can deduce that the fault is in S_s, applying the decision rule defined in (3.2).

The process of alarm generation is done considering digital signals. In Fig. 8 we show the $s_i(t)$ graphics in the case of fault occurrence displayed in Figures 7.

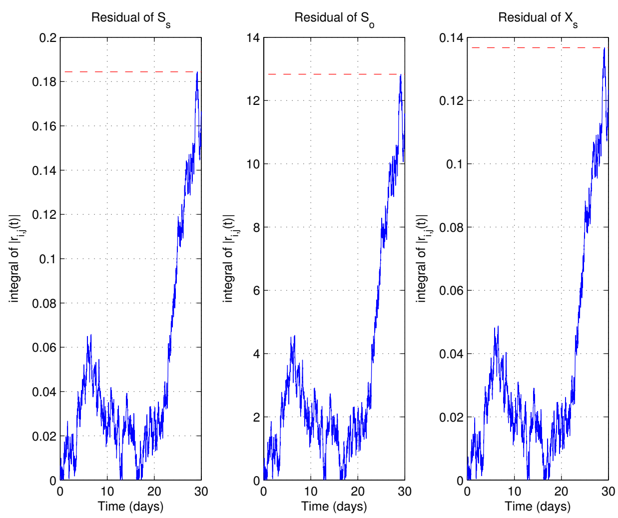
Furthermore, we developed a MATLAB code to randomly generate different faults and detect them. We just focused on the possibility of a 33%, 67% or 100% fault to have a general overview. In our simulations we succeeded in detecting correctly every fault occurrence, with a maximum delay of one day and a half in the case of 33% faults and better performances in the other cases (between 0.2 and 1.5 days). This way we test the efficiency of the detector.



(a) Observer using S_o and X_s

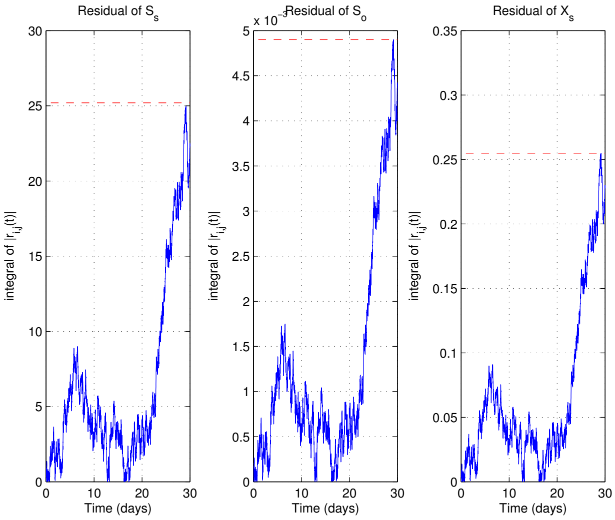


(b) Observer using S_s and S_o

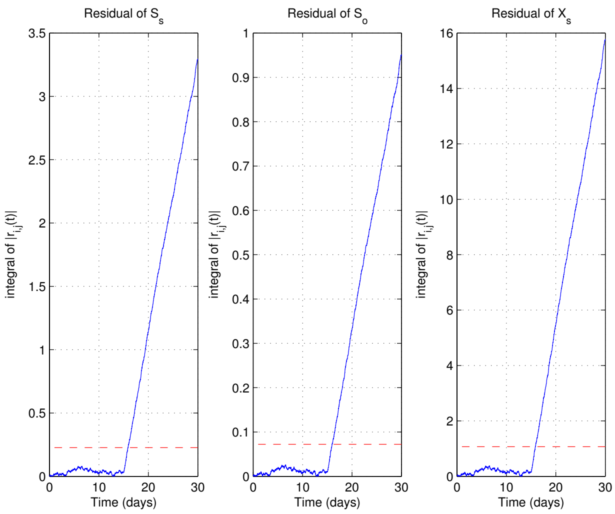


(c) Observer using S_s and X_s

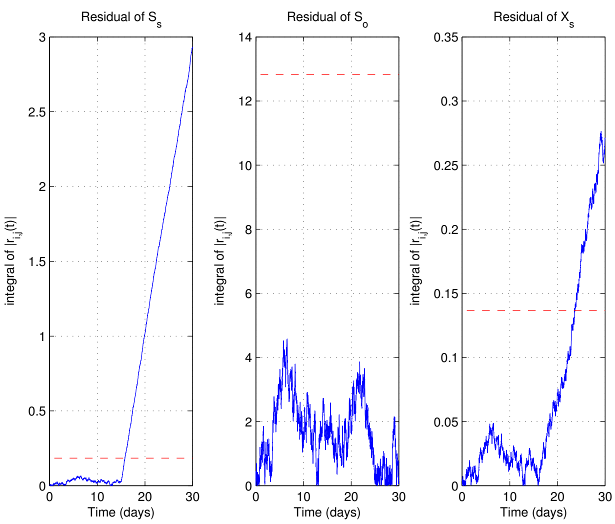
Fig. 6. Thresholds for fault detection in observer using S_o and X_s



(a) Observer using S_o and X_s



(b) Observer using S_s and S_o



(c) Observer using S_s and X_s

Fig. 7. Simulation results for fault occurring in S_s

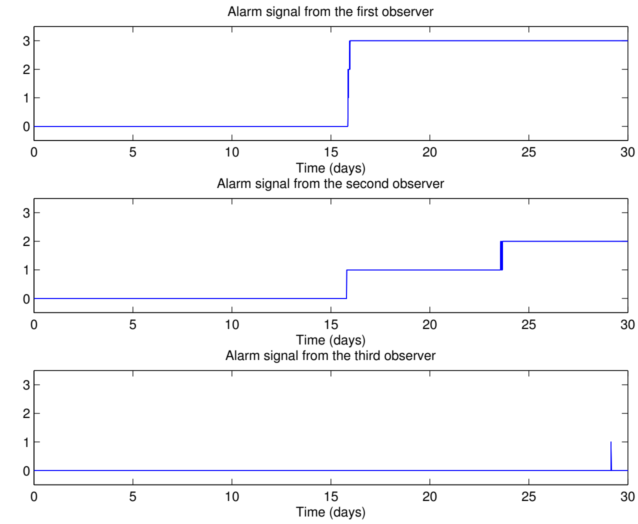


Fig. 8. Alarm Generation with digital signals for fault in S_s

5. CONCLUSION

The article presents an observer-based fault detection and isolation approach for an activated sludge process.
By implementing a bank of three Extended Luenberger Observers, each using different measured signals from the plant, it was possible to calculate three sets of each three residuals, enabling to thus decide with a set of previously defined decision rules, whether there occurred a fault in the sensors, and to identify also, which sensor is in fault.
Although the work only shows the basic principles of observer-based fault detection and identification, the results obtained are quite promising. They show that it is possible, to achieve reliable and relatively fast fault detection and identification with the very common Luenberger Observer and the concept of a bank of these observers.

ACKNOWLEDGEMENTS

The authors wish to thank the support received by the Research Commission of the Generalitat of Catalunya (ref. 2005SGR00537).

REFERENCES

M. Blanke, M. Kinnaert, J. Lunze and M. Staroswiecki. *Diagnosis and Fault-Tolerant Control.* Springer-Verlag,Berlin Heidelberg, 2003.

J. Chen. and R. J. Patton. *Robust Model-based Fault Diagnosis for Dynamic Systems.* Kluwer Academic Publishers, 1999.

C. DePersis and A. Isidori A Geometric Approach to Nonlinear Fault Detection and Isolation. In proc.: IFAC Safeprocess2000. pp. 209-214.

P. M. Frank. Analytical and Qualitative Model-based Fault Diagnosis - A Survey and Some New Results. *European Journal of Control* 2(1), 6-28, 1996.

P.M. Frank, G. Schreier and E.A. Garcia. *New Directions in Nonlinear Observer Design* Springer-Verlag, London, 1999

E.A. Garcia and P.M. Frank. *Deterministic Nonlinear Observer-Based Approaches to Fault Diagnosis: A Survey*, Control Engineering Practice 5(5), p. 663-760, 1997

J.P. Gauthier, H. Hammouri, S. Othman. A Simple Observer for Nonlinear Systems Applications to Bioreactors, *IEEE Transactions on Automatic Control*, Vol.37, 875–880, 1992

J. J. Gertler. *Fault detection and diagnosis in engineering systems* Marcel Dekker, Inc., New York, Basel, Hong Kong, 1998.

H. Hammouri, M. Kinnaert and E. H. Yaagoubi. Observer-Based Approach to Fault Detection and Isolation for Nonlinear Systems, *IEEE Trans. Automat. Contr.* Vol. 44, 1879-1884, 1999.

M. Henze, JR. Leslie Grady, C.P., Gujerm. W,, Maraism, G.V.R. and Matsnom, T. Activated sludge Model no. I, LA.KQ., Scientific and technical Report No. 1, 1987.

R. Isermann. Fault diagnosis of machines via parameter estimation and knowledge processign - tutorial paper, *Automatica*, 29(4): 815–836, 1993.

F. Nejjari, G. Roux, B. Dahhou, A. Benhamou. Non-linear Multivariable Adaptive Control of an Activated Sludge Wastewater Treatment Process *Int. J. Adapt. Control Signal Process.* 13, 347–365, 1999.

F. Nejjari, J. Quevedo. *Control no lineal adaptivo de un proceso de tratamiento de aguas residuales*, Jornadas de Automatica,Tenerife, 2002.

H. Nijmeijer and T. I. Fossen. *New Directions in Nonlinear Observer Design.* Springer-Verlag, London, 1999.

R. J. Patton and J. Chen. Observer-based fault detection and isolation: robustness and applications. *Control Engineering Practice*, 5(5), 671-682, 1997.

A. S. Willsky. A Survey of Design Methods for Failure Detection in Dynamic Systems. *Automatica* 12(6), 601611, 1976.

Appendix A. PROCESS EQUATIONS

$$
\begin{aligned}
\frac{dX_H}{dt} &= \frac{Q_{in}}{V_r}(X_{H,in} - X_H) + \frac{r_1 Q_{in}}{V_r}(X_{H,rec} - X_H) + R_{X_H} \\
\frac{dX_S}{dt} &= \frac{Q_{in}}{V_r}(X_{S,in} - X_S) + \frac{r Q_{in}}{V_r}(X_{S,rec} - X_S) + R_{X_S} \\
\frac{dX_I}{dt} &= \frac{Q_{in}}{V_r}(X_{I,in} - X_I) + \frac{r Q_{in}}{V_r}(X_{I,rec} - X_I) + R_{X_I} \\
\frac{dS_I}{dt} &= \frac{Q_{in}}{V_r}(S_{I,in} - S_I) \\
\frac{dS_S}{dt} &= \frac{Q_{in}}{V_r}(S_{S,in} - S_S) + R_{S_S} \\
\frac{dS_O}{dt} &= \frac{Q_{in}}{V_r}(S_{O,in} - S_O) + K_L a(C_S - S_O) + R_{S_O}
\end{aligned}
$$

$$
\begin{aligned}
\frac{dX_{H,rec}}{dt} &= X_H \frac{Q_{in} + Q_r}{V_{dec}} - X_{H,rec} \frac{Q_r + Q_w}{V_{dec}} \\
\frac{dX_{S,rec}}{dt} &= X_S \frac{Q_{in} + Q_r}{V_{dec}} - X_{S,rec} \frac{Q_r + Q_w}{V_{dec}} \\
\frac{dX_{I,rec}}{dt} &= X_I \frac{Q_{in} + Q_r}{V_{dec}} - X_{I,rec} \frac{Q_r + Q_w}{V_{dec}}
\end{aligned}
$$

$$
\begin{aligned}
R_{X_H} &= \rho_1 - \rho_2 \\
R_{X_S} &= (1 - \phi_{X_I})\rho_2 - \rho_3 \\
R_{X_I} &= \phi_{X_I}\rho_2 \\
R_{S_S} &= -\frac{\rho_1}{Y_H} + \rho_3 \\
R_{S_O} &= -\frac{1 - Y_H}{Y_H}\rho_1
\end{aligned}
$$

$$
\begin{aligned}
\rho_1 &= \mu_{H,max} X_H \frac{S_S}{K_S + S_S} \frac{S_O}{K_O + S_O} \\
\rho_2 &= b_H X_H \\
\rho_3 &= K_h X_H \frac{\frac{X_S}{X_H}}{K_X + \frac{X_S}{X_H}} \frac{S_O}{K_O + S_O}
\end{aligned}
$$

Process parameters and variables	
V_r	aerator's volume
V_{dec}	settler's volume
r	recycling flow
ϕ_{X_I}	fraction of inert COD generated in biomass lysis
b_H	Heterotrophic biomass decay rate
$\mu_{H,max}$	Heterotrophic max. growth rate
K_h	hydrolysis rate constant
K_S	Saturation constant for S_S
K_O	Saturation constant for S_O
$K_L a$	Oxygen mass transfer coefficient
K_X	Hydrolysis saturation constant
Y_H	Yield coefficient
Q_{in}	influent flow rate
X_H	Heterotophic biomass concentration
X_S	Slowly biodegradable substrates
X_I	Inert particulate organics
S_I	Soluble inert organics
S_S	Readily biodegradable substrates
S_O	Dissolved oxygen