Integration of a failure monitoring within a hybrid dynamic simulation environment

Nelly Olivier-Maget[a], Gilles Hétreux[a], Jean Marc Le Lann[a], Marie Véronique Le Lann[b]

[a] *Laboratoire de Génie Chimique (PSI – Génie Industriel), UMR-CNRS 5503, INPT-ENSIACET, 118, Route de Narbonne, Toulouse 31077, France.*

[b] *Laboratoire d'Architecture et d'Analyse des Systèmes,UPR-CNRS 8001 , 7, avenue du Colonel Roche, 31077 Toulouse Cedex 4, France*

Abstract :

The complexity and the size of the industrial chemical processes induce the monitoring of a growing number of process variables. Their knowledge is generally based on the measurements of system variables and on the physico-chemical models of the process. Nevertheless this information is imprecise because of process and measurement noise. So the research ways aim at developing new and more powerful techniques for the detection of process fault. In this work, we present a method for the fault detection based on the comparison between the real system and the reference model evolution generated by the extended Kalman filter. The reference model is simulated by the dynamic hybrid simulator, PrODHyS. It is a general object-oriented environment which provides common and reusable components designed for the development and the management of dynamic simulation of industrial systems. The use of this method is illustrated through a didactic example relating to the field of Chemical Process System Engineering.

Keywords: Fault Detection, Extended Kalman Filter, Dynamic Hybrid Simulation, Object Differential Petri Nets

1 Introduction

In a very competitive economic context, the flexibility of the production systems can be a decisive advantage. Generally, this flexibility lies on the search for a greater reactivity to a fluctuating demand, but also to many risks occurring during the manufacture. In this context, a simple failure is considered as prejudicial. This is why the fault detection and diagnosis are the purpose of a particular attention in the scientific and industrial community. The major idea is that the defect must not be undergone but must be controlled.

Nowadays, the functions of detection and diagnosis remain a large research field. The literature quotes as many fault detection and diagnosis methods as many domains of application [3]. A notable number of works has been devoted to fault detection and isolation, including surveys [1], [2], [3].

In the literature, the fault detection techniques are generally classified as:

- Methods without models such as quantitative process history based methods (statistical classifiers [3], [11], neural networks), or qualitative process history based methods (expert systems [3])
- And model-based methods which are composed of quantitative model-based methods (such as analytical redundancy [4], [5], parity space [6], state estimation [3], [7], or fault detection filter [5], [8]) and qualitative model-based methods (such as causal methods: digraphs [9], [10], or fault tree [3]).

The above articles underline the necessity to have precise knowledge of the normal process states during Fault Detection and Isolation. It has been recognized that fault detection using all measurements and models by a single filter is weighty and sometimes inappropriate for real time use. Nevertheless, while model-based techniques require greater computing power and data storage, the amazing progress in computer technology has made them feasible at low costs. Moreover, model deviation, nonlinearities and measurement disturbances are unavoidable in industrial systems. They are often the cause of false detection in fault detection and localisation algorithms. Despite the complexity of such algorithms, most of the time, their robustness is insufficient in industrial context.

In this paper, the proposed approach to fault detection is a model-based approach. The considered systems are batch and semi-continuous processes which are the prevalent mode of production for low volume of high added value products. Such systems are composed of interconnected and shared resources, in which a continuous treatment is carried out. For this reason, they are generally considered as hybrid systems in which discrete aspects mix with continuous ones. Moreover, the recipe is more often described with state events (temperature or composition threshold, etc.) than with fixed processing times.

As a consequence, the simulation of unit operations and physico-chemical evolution of products often necessitates the implementation of phenomenological models. So, the traditional tools such as the discrete events simulators (which require fixed operational durations) or traditional formalisms integrating only the concept of temporal events (timed automata [12], timed Petri nets [13]) or crossing speeds (continuous Petri nets [14], hybrid Petri nets [15]) are not well adapted to these problems. In this context, the use of hybrid dynamic simulators seems to be a better solution [16].

This framework is organized as follows. The first part of this communication focuses on the main fundamental concepts of the simulation library PrODHyS. This is followed by a presentation of a modelling within PrODHyS. Then, Section 4 presents the proposed detection approach. This exploits the extended Kalman Filter to a hybrid dynamic system. The main idea is to reconstruct the outputs of the system from the measurement using observers or Kalman filters and using the residuals for fault detection [3], [17], [18], [19], [20], [21]. The purpose is to detect the presence of a fault and to locate the occurrence time. The estimations are compared to the normal parameter values and so, deviations are interpreted as faults. Section 5 illustrates the proposed model-based fault detection through the simulation of a hydraulic system used as benchmark. Composed of two interconnected tanks, the goal of this system is the regulation of the tank levels. A fault in the functioning of a valve, occurring at an unknown moment, is simulated. The valve stays closed instead of opening. With only the measurements of flow rate, the on-line system identification and damage detection are possible based on the Extended Kalman Filter approach. Finally, the evolution of the information by the state estimation is observed and discussed, as well as the delay in detection according to the decision threshold. Finally, Section 6 summarises the contributions and achievements of the paper and some future research works are suggested.

2 PrODHyS environment

The simulation of hybrid systems has led to the development of several software such as gPROMS [20], BASIP [21], in which the hybrid aspect is described via an imperative language. In parallel, various hybrid formalisms have been defined or obtained by extension of existing discrete or continuous formalisms [14].

In this context, the research works performed for several years within the PSE research department (LGC) on process modelling and simulation have led to the development of PrODHyS [24], [25], [26], [27]. This environment provides a library of classes dedicated to the dynamic hybrid simulation of processes. Currently, this library is made up of more than one thousand classes distributed into three functional layers and nine modules (Figure 1). Based on object concepts, PrODHyS offers extensible and reusable software components allowing a rigorous and systematic modelling of processes. The primal contribution of these works consisted in determining and designing the foundation buildings classes.

The last important evolution of PrODHyS is the integration of the dynamic hybrid simulation kernel [26], [27], [28]. Indeed, the nature of the studied phenomena involves a rigorous description of the continuous and discrete dynamic. The use of Differential and Algebraic Equations (DAE) systems seems obvious for the description of continuous aspects. Moreover the high sequential aspect of the considered systems justifies the use of Petri nets model. This is why the Object Differential Petri Nets (ODPN) formalism is used to describe the simulation model associated with each component. It combines in the same structure a set of DAE systems and high level Petri nets (defining the legal sequences of commutation between states) and has the ability to detect state and time events.

2.1 Petri net/object paradigm

Although scientifically interesting, the mathematical properties of this formalism are not developed here but can be found in [26]. Only its major principles are presented in this paper. The PrODHyS components allow a modular and hierarchical modelling of different processes. In consequence, the object concepts and the Petri nets have been exploited in a combined approach in the ODPN formalism. It consists in making interact these features according to two manners (Figure 2). Firstly it aims at "introducing the objects into Petri nets". The subjacent philosophy is to model a subsystem by a single Petri net, which handles individualised tokens carrying information. The second approach is based on "the introduction of Petri nets into objects" to describe the internal behaviour of the object. The marking of the Petri net indicates the current state of the object.

2.2 Petri net/ DAE paradigm

The ODPN formalism makes collaborate, within the same structure, DAE systems to describe the continuous evolution of the system with a high level Petri net used to specify the legal sequences of commutation between this set of DAE systems. The Figure 3 gives an example of evolution in the ODPN. Each token in a differential place is substituted for formal variables of the DAE model. It involves the continuous evolution of the considered attributes and the estimate of the condition associated to the transition (state or time events). When the condition is checked (occurrence of an event) the action is performed and the transition is crossed. Therefore, a new marking is set up and activates the corresponding DAE system. Of course, several tokens of the same type can mark a place, and each of them activates the computation of the same DAE system from the marking date.

Moreover, the integration of the DAE system of a differential place can require several tokens of the same type and/or of different types. In this case, the consistency must be assured by the modelling or validated by the simulation. Thus, the Petri net can be seen as a DAE monitor. It allows the dynamic creation of a unique simulation model, whose size and structure change between two events (no fixed size of state vector). Besides the resolution of the DAE system (integration based on the Gear method) and of the discrete models (Petri net player specific to this class), the kernel manages other functionalities, such as the exact calculation of the commutation times, the state failing, the checking of the consistency of the new models generated after the commutation, the initialisation of the state variables and their derivatives… These features are detailed in [24].

3 Process modelling with PrODHyS

3.1 General structure of the simulation model

The simulation of a discontinuous process necessitates to model separately the command part (the supervisor) and the operative part (the process).

Concerning the operative part, the specification of any device of PrODHyS is always defined according to two axes: a topological axis and a phenomenological axis. The topological axis defines the structure of the process (system vision): physical connections (material, energy, information) between the different parts of the process and hierarchical decomposition of the devices. The phenomenological axis rests on a set of mathematical models based on mass and energy balances and thermodynamic and physico-chemical laws. Thus, the models of devices are reusable whatever the context. In addition, the combined approach is used to dissociate the model of material from the model of devices which contains the material.

Therefore, object tokens are reusable and reduce the complexity of the devices Petri nets. More details on the modelling of devices and material can be found in previous communications [24]. On the other hand, the model of the command part is specific to the recipe and the process topology.

It consists in describing the procedure of manufacture of each product. So, it specifies the assignments of resources and the sequence of tasks ordered in time necessary to the realization of each batch.

3.2 Connections between « devices » PN and « recipe » PN

The exchanged signals, between the command part and the operative part, are modelled by a discrete place. The state of a signal state is associated to the marking of the corresponding place. In this framework, an entity is either an active device if it has one or more signal places (such as valves, pumps, feeds, column, sensors) or a passive device if there is no direct relation with the recipe (such as simple tanks or reactors). These notions are illustrated in Figure 4. It represents an operative sequence which permits the feed of a tank until a fixed volume is reached. The marking of the signal place of an active entity induces the evolution of its Petri net. This Petri net can itself induce the evolution of active or passive entities in cascade through the net composed with the connection of different material or energy ports

3.3 Modelling of a fault

The reference process (without faults) and the monitored process (with possible faults) have the same recipe: the command part doesn't change. Thus the fault appears in the modelling of the devices. Then, the associated simulation model is build from specific *Device* objects in which the faults are defined in an intrinsic way. In a first time, for a simple analysis of the system, the failures are generated by the simulation monitor thanks to a defect calendar which lists the defect, its occurency time and its duration. The Figure 5 represents the modelling of a faulty valve. In the studied case, we consider only the failures “a” and “b” since the place Normal of the failure generator is only marked by two object tokens $<f_a>$ and $<f_b>$ (type of the objects: the class failure). In a second time, during the test stage, the failures are generated on autonomous and random commutations.

4 The supervision module

Nowadays, for reasons of safety and performance, monitoring and supervision have an important role in process control. The complexity and the size of industrial systems induce an increasing number of process variables and make difficult the work of operators. In this context, a computer aided decision-making tool seems to be wise. Nevertheless the implementation of fault detection and diagnosis for stochastic system remains a challenging task. Various methods have been proposed in different industrial contexts [3].

4.1 Architecture

For this purpose, the simulation model of PrODHyS is used as a reference model to implement the functions of detection and diagnosis. The global principle of this system is shown in the Figure 6. In order to obtain an observer of the physical system, a real-time simulation is done in parallel. So, a complete state of the system will be available at any time. The supervision module must be able to the faults of the physical systems (leak, energy loss, etc.) and the faults of the control/command devices (actuators, sensors, etc.). As defined in

[24], our approach is based on the hypothesis that the reference model is presumed to be correct.

Thus, it is based on the comparison between the predicted behaviour obtained thanks to the simulation of the reference model (values of state variables) and the real observed behaviour (measurements from the process correlated thanks to the Extended Kalman Filter). Detection is realized by comparison with fixed thresholds. For a consistent execution of this task, the measurements must be filtered in order to eliminate the noise. The filter used here is the Extended Kalman Filter. Next the detection is made by a simple threshold.

4.3 Brief description of the Extended Kalman Filter

Generally, the filtering function aims at considering useful information (signal) which is polluted by a noise. The Kalman Filter is one of the most widely used tools for state and parameter estimations in stochastic systems. It estimates optimally the state of the linear system (thus, this state corresponds to useful information). It is a recursive estimator. This means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state.

Within the linear filters, the Extended Kalman filter is a non optimal approach to solve the problem. Among various different model-based methods used in the past, this filter is clearly one of the most popular methods. Originally, it is exploited to approximate both the states and the parameters of chemical engineering processes and then, to identify the causes of abnormal behaviours of the process [29]. This filter implements a Kalman filter for dynamics that results from the linearization of the original non-linear dynamics around the previous state estimates. Of course, the system model is not necessarily linear but it must be differentiable. Consider the following nonlinear system:

$$\dot{X}(t) = f(X(t), U(t)) + v(t) \tag{1}$$

$$Z(t) = h(X(t)) + w(t) \tag{2}$$

Where, X(t) is the state vector, f defines the system model, U(t) is the input vector, v(t) conveys the system noise, Z(t) is the output vector, C defines the output, and w(t) conveys the measurement noise.

The Kalman filter requires a discrete continuous space model. A simplified discrete form for the state space model is generated by using the approximation of Euler. Applying this approximation to the equations (1) and (2), which are first linearized, the equation system becomes:

$$X_k = FX_{k-1} + GU_k + V_k \tag{3}$$

$$Z_k = HX_k + W_k \tag{4}$$

Where,

$X_k \in \Re^n$ is the system state vector,

$F \in \Re^{n \times n}$ defines the system dynamics,

$U_k \in \Re^p$ is the input vector,

$G \in \Re^{n \times p}$ defines the system inputs,

$V_k \in \Re^n$ is the vector representing the system disturbances,

$Z_k \in \Re^m$ is the observation vector,

$H \in \Re^{m \times n}$ defines the measurements,

And $W_k \in \Re^m$ is the vector that represents the measurement error sources,

$\{V_k\}$ and $\{W_k\}$ are white Gaussian, independent random noises or disturbances with zero mean and covariance matrix:

$$E\left[V_k V_k^T\right] = Q_k \tag{5}$$

$$E\left[W_k W_k^T\right] = R_k \tag{6}$$

The Extended Kalman Filter gives an approximation of the optimal estimate. The prediction expression is denoted by:

$$\hat{X}_{k|k-1} = F\hat{X}_{k-1|k-1} + GU_{k-1} \tag{7}$$

The notation k|k-1 means that the calculation of the prediction is made at the instant k knowing only the state at the instant k-1. So, the notation k|k corresponds to the correction of the estimation k|k-1 thanks to the new information (measurements) at the instant k.

The Extended Kalman Filter equations are algebraic and recursive. So, only a little computation is required in order to estimate in real time and to control. In the algorithm, each new estimate is formed as a blend of the old estimate and the current measurement. The following consecutive steps compose of an iteration of the Extended Kalman Filter algorithm (Figure 7).

<u>Step 0:</u> Initialize with the last filtered estimate state $\hat{X}_{k-1|k-1}$

<u>Step 1:</u> Apply the prediction step of the Kalman filter to the linearized system dynamics yielding $\hat{X}_{k|k-1}$

$$\hat{X}_{k|k-1} = F\hat{X}_{k-1|k-1} + GU_{k-1} \tag{8}$$

<u>Step 2:</u> Apply the prediction step of the Kalman filter to the linearized system dynamics yielding the error covariance matrix $P_{k|k-1}$:

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q_{k-1} \tag{9}$$

<u>Step 3:</u> If there are available measurements, then go to step 4, else go to the step 0

<u>Step 4:</u> Calculate the Kalman gain matrix K_k denoted by:

$$K_k = P_{k|k-1} H^T \left[H P_{k|k-1} H^T + R_k \right]^{-1} \quad (10)$$

Step5: Update the state estimate $\hat{X}_{k|k}$:

$$\hat{X}_{k|k} = \hat{X}_{k|k-1} + K_k \left[Z_k - H \hat{X}_{k|k-1} \right] \quad (11)$$

Step 6: Refine the error covariance matrix:

$$P_{k|k} = [I - K_k H] P_{k|k-1} \quad (12)$$

5 Application

In order to illustrate the proposed approach let us consider the hydraulic system depicted in the Figure 8. This system is inspired by the benchmark of the Specific Action on the diagnosis of hybrid systems (AS193) of CNRS[1] and GDRMACS[2] (cf. www.univ-lille1.fr/lail/AS193/).

5.1 Description

This system consists of two cylindrical tanks C1 and C2, connected by two pipes with "on/off" valves V3 and V4. The feed of the tanks is maintained by the "on/off" pumps P1 and P2. The tank C2 can be drained through the "on/off" valve V2. The valve V1 is not used here. The instrumentation of the process is composed (in a maximal configuration) of 6 flow sensors and 2 level sensors.

The goal of the control device consists in maintaining the liquid level h2 in C2 between the heights h_{2min} and h_{2max} by a discrete control of the valve V4. The valve V3 is opened only when the level in C2 is such $h_2 \leq h_{2alarm}$.

[1] Centre National de la Recherche Scientifique
[2] Groupement de Recherche « Modélisation, Analyse et Conduite des Systèmes dynamiques » of CNRS

The implemented command is voluntarily simple. Because the command law does not take into account the level h_1 in the tank C1, the objective can not always be ensured. The Petri net associated with the command level is presented on Figure 9.

5.2 Mathematic model

In this section, a brief description of the complete mathematical model of each device is given.

In this system (Figure 8), a valve may be composed of one input and one output characterized respectively by the data $(F_{input}, h_{input}, x_{input})$ and $(F_{output}, h_{output}, x_{output})$ where F_f, h_f and x_f are respectively the flow, the liquid enthalpy and the liquid composition vector of the flow f (f=input or output). In this case, the model of a valve is composed of:

- The global material balance:

$$F_{input} - F_{output} = 0 \tag{13}$$

When the valve is closed, it becomes:

$$F_{input} = 0 \quad F_{output} = 0 \tag{14}$$

- The constraint on the flow due to the hydraulic phenomena:

$$F_{input} = S_C \cdot \mathrm{sign}(L_{input} - L_{output}) \cdot \sqrt{2 \cdot g \cdot |L_{input} - L_{output}|} \tag{15}$$

Where, S_c is the pipe area, L_f is the liquid level in the tank and g is the gravity constant.

In this system, the model of a pump is composed of:

- The global material balance:

$$F_{input} - F_{output} = 0 \tag{16}$$

- The constraint on the flow:

$$F_{input} = F_{order} \tag{17}$$

Where F_{order} is the order flow of the pump.

Finally a storage tank may be composed of two inputs and two outputs characterized respectively by the data $(F_{input1}, h_{input1}, x_{input1})$, $(F_{input2}, h_{input2}, x_{input2})$, $(F_{output1}, h_{output1}, x_{output1})$ and $(F_{output2}, h_{output2}, x_{output2})$ where F_f, h_f and x_f are respectively the flow, the liquid enthalpy and the liquid composition vector of the flow f (f=input1 or input2 or output1 or output2).The variable U_l represents the liquid holdup in the tank. The variable L is the liquid level calculated according to U_l, the tank area S_t and the molar volume of the liquid phase V_{ml}. T,P are respectively the pressure and the temperature of the system. Consequently, the model of a storage tank is composed of:

- The global material balance:

$$\frac{dU_l}{dt} = F_{input1} + F_{input2} - F_{output1} - F_{output2} \tag{18}$$

- The partial material balances:

$$\begin{aligned}\frac{dU_l \cdot x_i}{dt} &= F_{input1} \cdot x_{input1,i} + F_{input2} \cdot x_{input2,i} \\ &- F_{output1} \cdot x_{output1,i} - F_{output2} \cdot x_{output2,i}\end{aligned} \tag{19}$$

- The energy balance:

$$\begin{aligned}\frac{dU_l \cdot h_i}{dt} &= F_{input1} \cdot h_{input1} + F_{input2} \cdot h_{input2} \\ &- F_{output1} \cdot h_{output1} - F_{output2} \cdot h_{output2}\end{aligned} \tag{20}$$

- The liquid level:

$$L - \frac{U_l \cdot V_{ml}}{S_t} = 0 \tag{21}$$

- The constraint on the liquid enthalpy:

$$h - mh(T, P, x) = 0 \tag{22}$$

- The constraint on the liquid molar volume:

$$V_{ml} - mV_{ml}(T, P, x) = 0 \tag{23}$$

- The constraint on the pressure:

$$P - mP(Z) = 0 \quad (24)$$

Where Z is the set of operative parameters of the system.

Parts of these models are merged only when a simulation model is instantiated. Thus, the size and the structure of the resulting DAE systems change all along the simulation, according to the actual state of the process.

This established model is used by the Extended Kalman filter in order to build both the state and the system output. Then the estimation is compared with the real process and the residual is analysed with the aim of detecting the faults.

5.3 Results

Various scenarios can be simulated by action on the pumps P1 and P2 and the valve V2 (Figure10). For the set of parameters indicated in the Figure 8 and the scenarios shown in the Figure 10, the simulation results are presented in the Figure 11. With only the measurements of flow rates, the fault detection based on the Extended Kalman Filter is possible.

5.4 Discussions

The Extended Kalman Filter requires a linearization of the non linear system. This estimation means the no guarantee success of the filter convergence. The model mistakes are one of the causes the most important in the divergence of the Kalman Filter. This divergence is due to the so important confidence of the filter in the model. This is the case when the model noise is low. As a matter of fact, the terms of the covariance matrix representing the model disturbances and of the gain matrix decrease and so the filter doesn't take into account the observations. So the performance of the filter depends on the knowledge of the covariance matrices representing the system disturbances and the measurement noises. These values were fixed independently..

Furthermore, the value ε, beyond which the difference between measurements and model variables is considered as a defect, remains a delicate point to evaluate. This value is often obtained from a compromise of a series of simulations, in which its value is customized.

Moreover, the evolution of the simulation must be synchronous with the evolution of the real process. However, this feature can not be always ensured. When the reference model is either ahead or late, a retiming of the reference model on the real process is then necessary in order to be able to validate a detection test. Despite these mechanisms, let us underline the difficulty of the decision stage in the detection. Indeed, some particular states of a system can be found where a faulty behaviour seems to be similar to a normal behaviour; in this condition, it is impossible to affirm the absence of fault. This is why the uncertainty on the veracity of the detection tests may lead to unsuccessful diagnosis.

6 Conclusion

In this paper, the feasibility of using the Extended Kalman Filter as a tool for fault detection is described. The method developed in this study uses a hybrid dynamic simulator PrODHyS. This simulator is based on an object oriented approach. It brings many advantages in terms of software quality (extensibility, reutilisability, flexibility), but especially in terms of modelling thanks to a hierarchical and modular description which is both abstracted and close to reality. Then, PrODHyS provides software components intended to model and simulate more specifically the industrial processes. The implementation of a formalism on high level of abstraction associated with powerful numerical methods of integration led to the construction of a robust hybrid dynamic simulator. In this communication, the potentialities of PrODHyS are illustrated through the modelling and the simulation of a hydraulic process. The works in progress aim at integrating this simulation model within a model based diagnosis system.

Different diagnosis approaches mixing model-based and data classification techniques will be studied and compared.

Appendix A. Nomenclature

References

[1] R. Isermann, Process fault detection based on modelling and estimation methods: a survey, Automatica, 20 (1984) 307-404

[2] P.M. Franck, Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy—a survey and some new results, Automatica, 26 3 (1990) 459–474

[3] V. Venkatasubramanian, R. Rengaswamy, K. Yin, S. N. Kavuri, A review of process fault detection and diagnosis, Computers & Chemical Engineering (Elsevier), 27 (2003) 293-346

[4] E.Y. Chow, S. Willsky, Analytical redundancy and the design of robust failure detection systems, IEEE Transactions on Automatic Control, 29 7 (1984) 603-614

[5] P.M. Frank, Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – a survey and some new results, Automatica, 26 (1990) 459-474

[6] J. Gertler, D. Singer, A new structural framework for parity equation-based failure detection and isolation, Automatica, 26 (1990) 381-388

[7] A.S. Willsky , A survey of design methods for failure detection in dynamic systems, Automatica, 12 (1976) 601 – 611

[8] J. Chen, R.J. Patton, H.Y. Zhang, Design of unknown input observer and robust fault detection filters, International Journal Control, 63 1 (1999) 85-105

[9] R. Shih, L. Lee, Use of fuzzy cause-effect digraph for resolution fault diagnosis for process plants I. Fuzzy cause-effect digraph, Industrial and Engineering Chemistry Research, 34 5 (1995) 1688-1702

[10] R. Shih, L. Lee, Use of fuzzy cause-effect digraph for resolution fault diagnosis for process plants II. Diagnostic algorithm and applications, Industrial and Engineering Chemistry Research, 34 5 (1995) 1703-1717

[11] T.W. Anderson, An introduction to multivariate statistical analysis, New York: Wiley, 1984

[12] R. Alur, D.L Dill, A theory of timed automata, Theoretical Computer Science, 126 2 (1994) 183-235

[13] J. Sifakis, Use of Petri Nets for Performance Evaluation, Measuring, Modelling and Evaluating Computer Systems, H. Beilner et E. Gelenbe (dir.), North-Holland Pub. Co, 1977 75-93

[14] R David., H. Alla, Autonomous and Timed Continuous Petri Nets, 11th International Conference on Application and Theory of Petri Nets, Saragosse, Spain, June 1990

[15] R. David, H. Alla, On Hybrid Petri Nets, J. of Discrete Event Dynamic Systems: Theory and Applications, 11 (2001) 9-40

[16] J. Zaytoon, Systèmes dynamiques hybrides, Hermès Sciences publications, 2001

[17] M. Basseville, Model-Based Statistical Signal Processing and Decision Theoretic Approaches to Monitoring, Proceedings of IFAC Safeprocess 2003. Washington, DC, 2003.

[18] R.K. Mehra, J. Peschon, An Introduction approach to fault detection and diagnosis in dynamic systems, Automatica, 5 (1971) 637-640

[19] S. Simani, C. Fantuzzi, Dynamic system identification and model-based fault diagnosis of an industrial gas turbine prototype, Mechatronics, 16 (2006) 341-363

[20] G. Welch., G. Bishop, An introduction to the Kalman filter, Technical Report TR 95-041, University of North Carolina, 1995.

[21] R. Xiong, P.J. Wissman, M.A. Gallivan, An extended Kalman filter for in situ sensing of yttria-stabilized zirconia in chemical vapour deposition, Computers and Chemical Engineering, 30 (2006) 1657-1669

[22] P.I. Barton, C.C. Pantelides, The Modelling of Combined Discrete/Continuous Processes. AIChE Journal, 40 (1994) 966-979

[23] K. Wöllhaf, M. Fritz, C. Schulz, S. Engell, BaSiP — Batch process simulation with dynamically reconfigured process dynamics, Computers & Chemical Engineering, 20 2 (1996) S1281-S1286

[24] L. Jourda, X. Joulia, B. Koehret, Introducing ATOM, the Applied Thermodynamic Object-oriented Model, Computer & Chemical Engineering, 20A (1996) S157-S164

[25] A. Sargousse, Noyau numérique orientée-objet dédié à la simulation des systèmes dynamiques hybrides, Ph.D. thesis at the I.N.P. Toulouse, 1999

[26] J. Perret, G. Hétreux, J.M. LeLann, Integration of an object formalism within a hybrid dynamic simulation environment, Control Engineering Practice (Elsevier), 12 10 (2004) 1211-1223

[27] N. Olivier, G. Hétreux, J.M. LeLann, Formal modelling and simulation for control of batch processes, Conference on Conceptual Modelling and Simulation CMS'05; Marseille, France, 2005

[28] N. Olivier, G. Hétreux, J.M. LeLann, M.V. LeLann, Use of an Object Oriented Dynamic Hybrid Simulator for the Monitoring of Industrial Processes, proceedings of ADHS'06, Alghero, Italia, 2006 235-240

[29] K. Watanabee, D. M. Himmelblau, Incipient fault diagnosis of nonlinear processes with multiple causes of faults, Chemical Engineering Science, 39 3 (1984) 491-508

[30] J. De Kleer, B.C. Williams, Diagnosing multiple faults, Artificial Intelligence, 32 (1987) 97-130

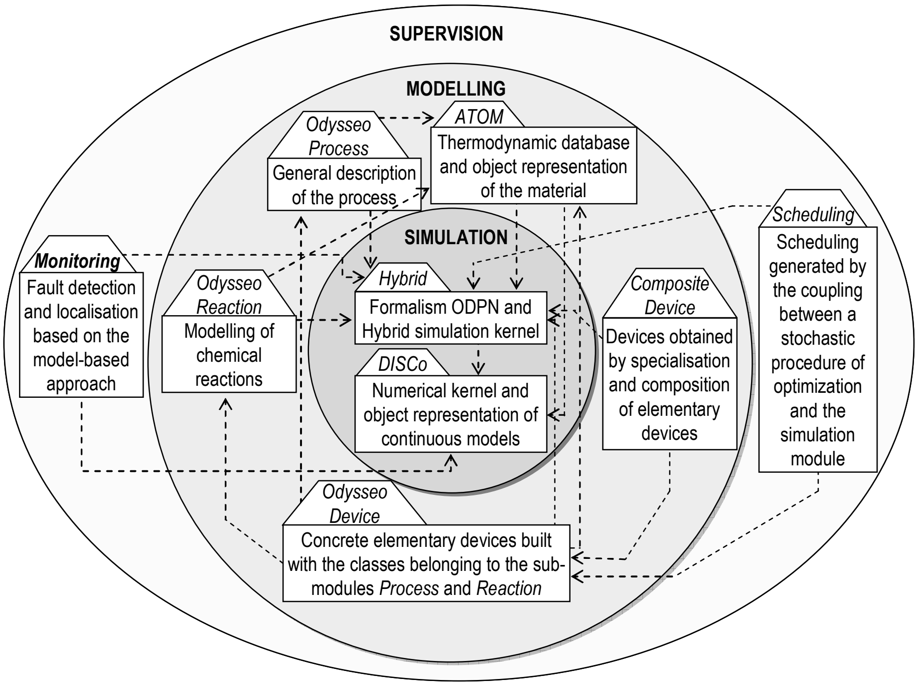


Fig.1. Software architecture of PrODHyS

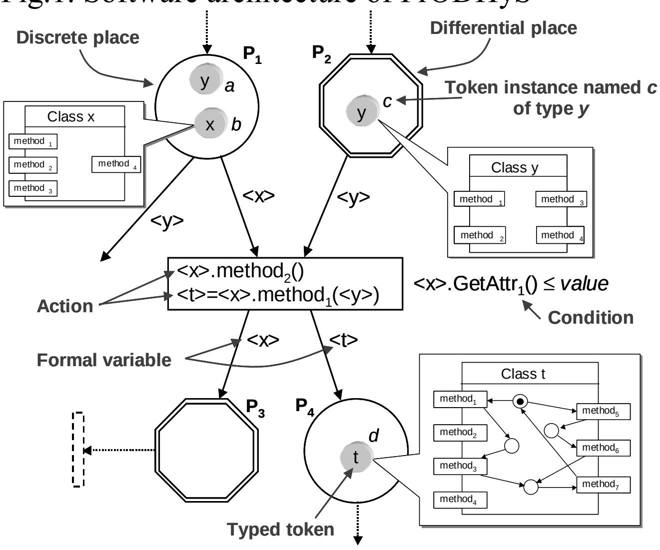


Fig. 2: Elements of ODPN formalism

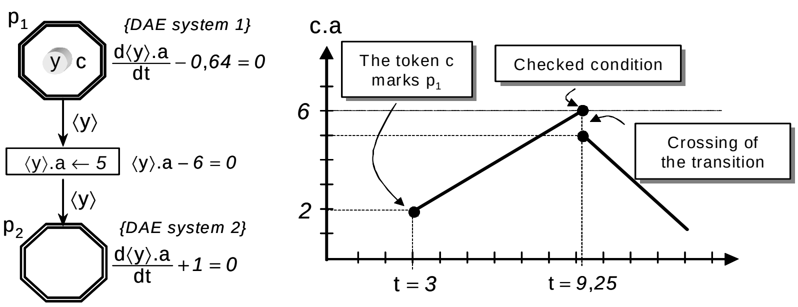


Fig. 3. *ODPN* evolution

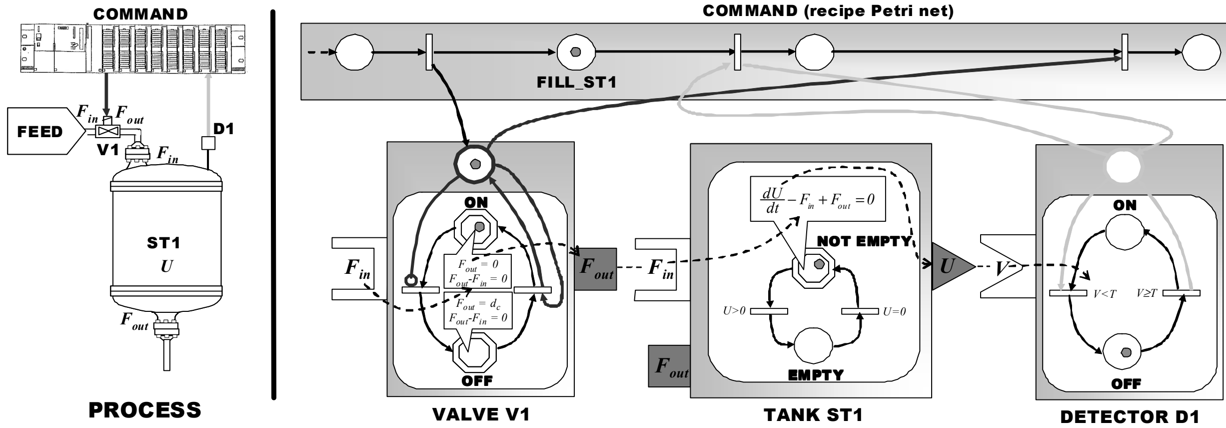
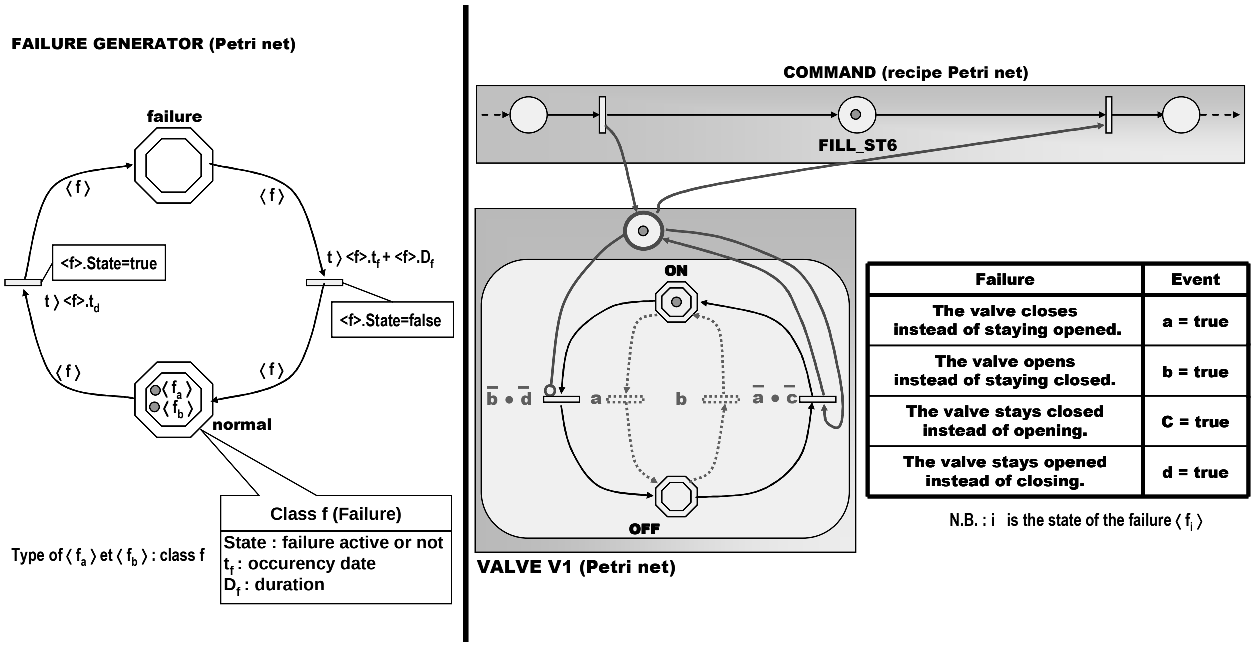


Fig. 4. Interactions between the command level and the process level



Failure	Event
The valve closes instead of staying opened.	a = true
The valve opens instead of staying closed.	b = true
The valve stays closed instead of opening.	C = true
The valve stays opened instead of closing.	d = true

N.B. : i is the state of the failure ⟨ f_i ⟩

Fig. 5. Modelling of a faulty valve

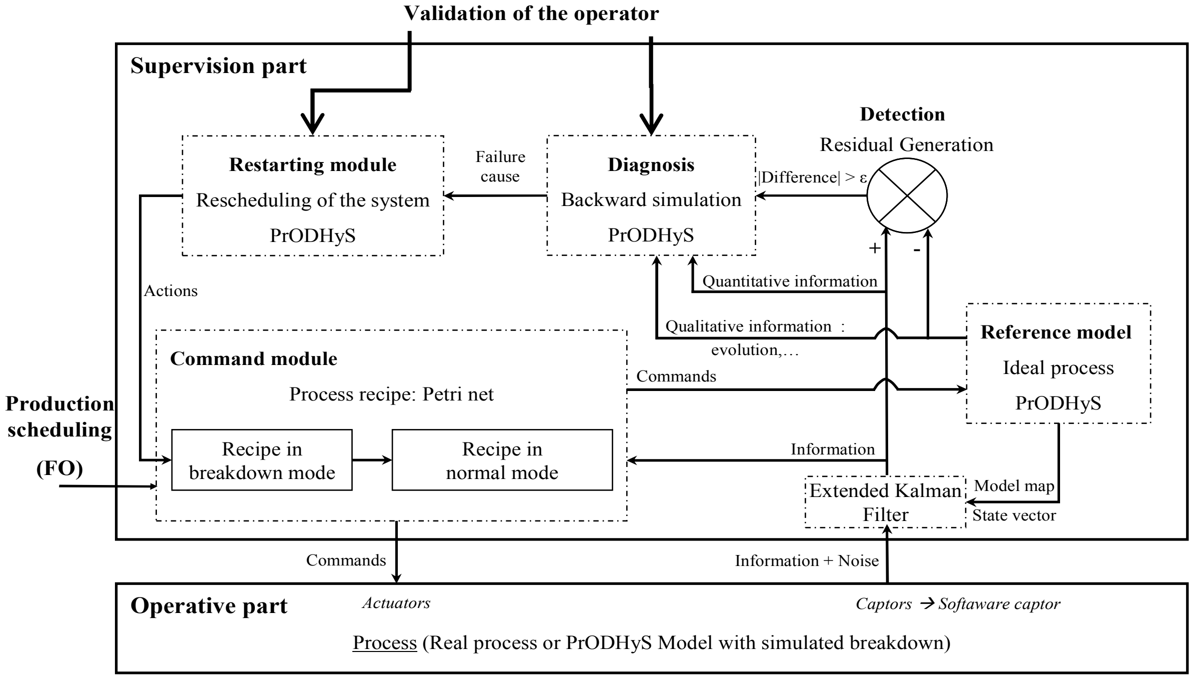


Fig. 6. Supervision Architecture

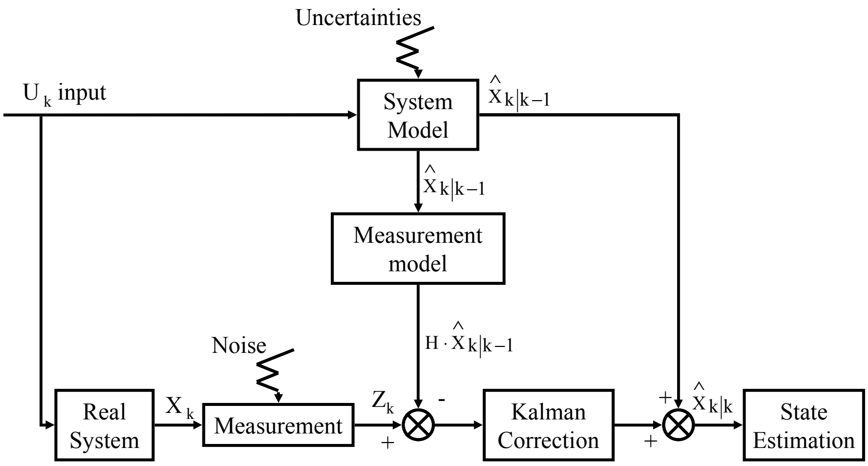


Fig. 7. The Extended Kalman Filter Block Diagram

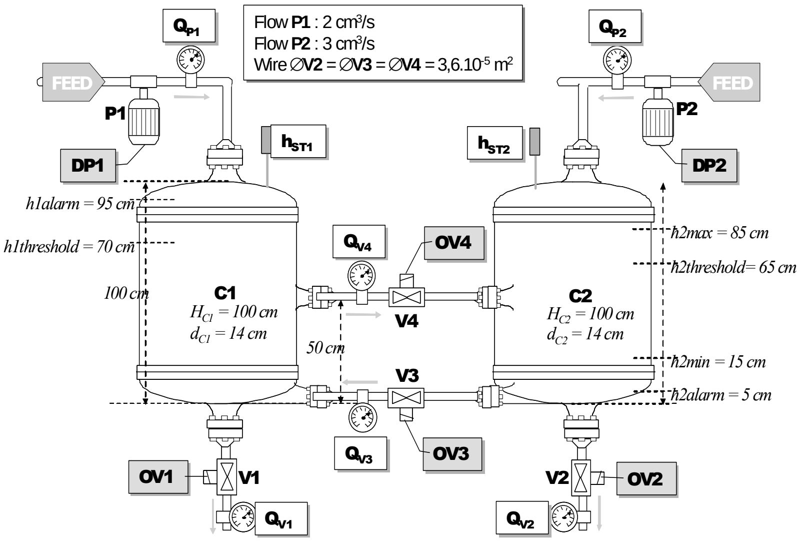


Fig. 8. Flowsheet of the benchmark

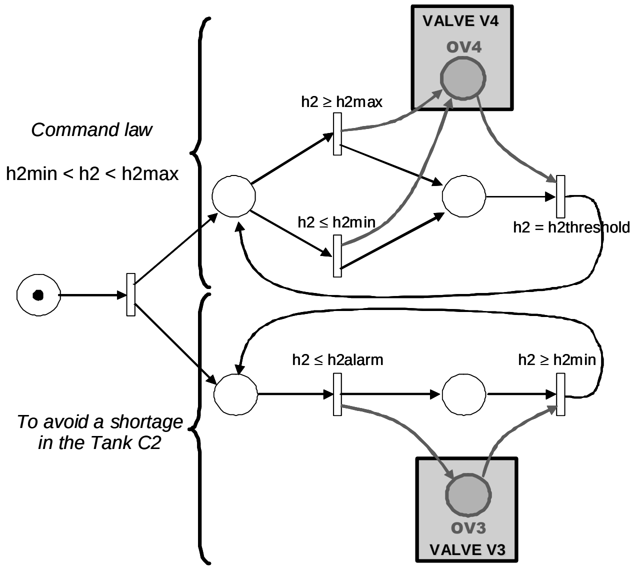


Fig. 9. Command Petri net

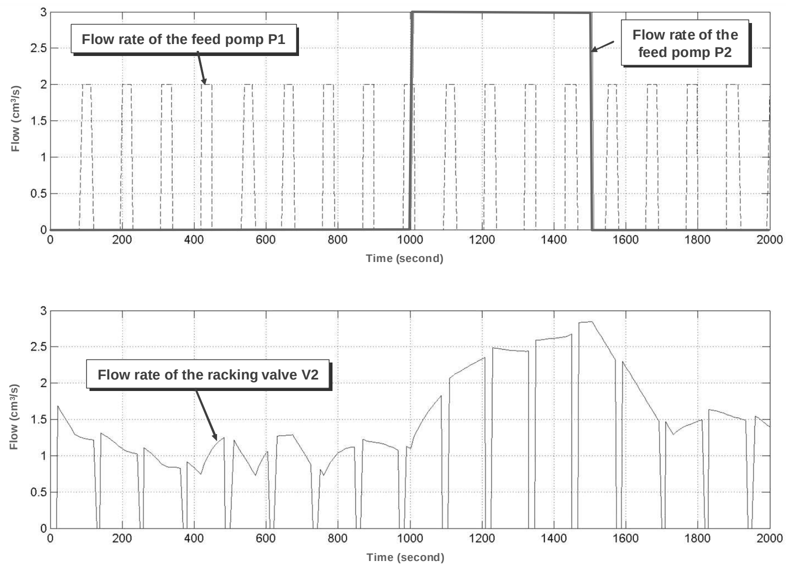


Fig. 10. The scenarios

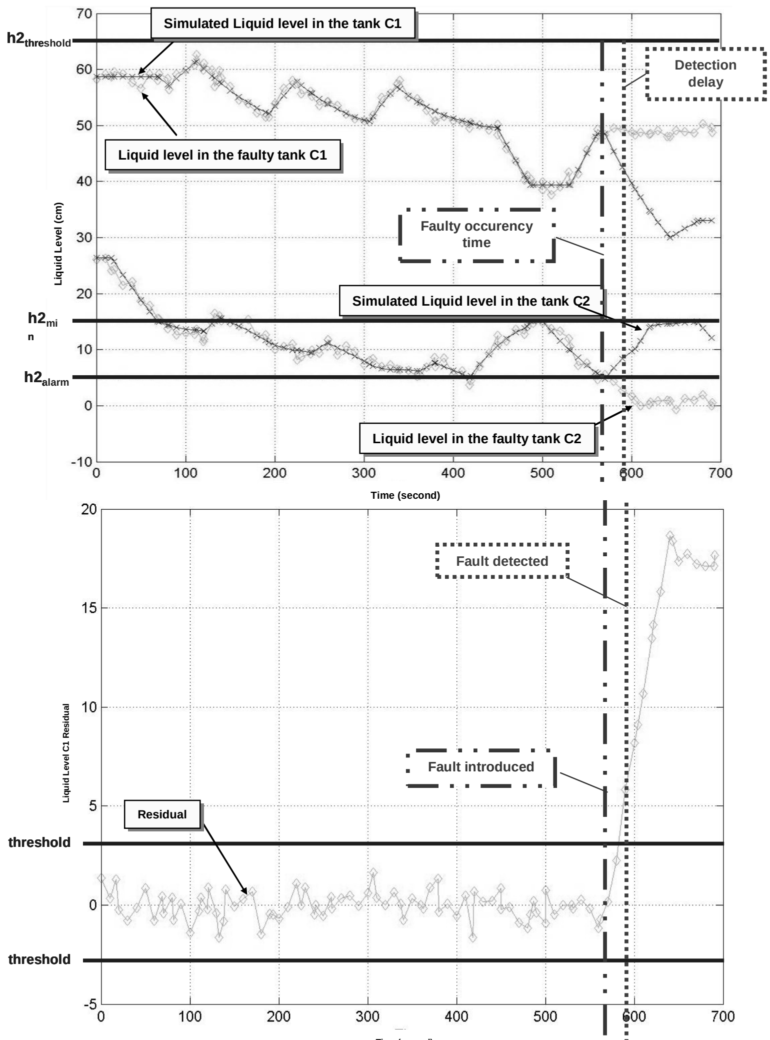


Fig. 11. The results