Computers and Chemical Engineering 35 (2011) 806–816

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Kalman-based strategies for Fault Detection and Identification (FDI): Extensions and critical evaluation for a buffer tank system

Kris Villez[a,*], Babji Srinivasan[b], Raghunathan Rengaswamy[b], Shankar Narasimhan[c], Venkat Venkatasubramanian[a]

[a] Laboratory for Intelligent Process Systems, School of Chem. Eng., Purdue University, West Lafayette, IN 47907, USA
[b] Dept. of Chem. Eng., Texas Tech University, Lubbock, TX 79409, USA
[c] Dept. of Chem. Eng., Indian Institute of Technology, Madras, Chennai 600036, India

ARTICLE INFO

Article history:
Received 1 October 2010
Received in revised form 11 January 2011
Accepted 17 January 2011
Available online 9 March 2011

Keywords:
Kalman filter
Fault Detection and Identification (FDI)
Process safety
Process control
Non-linear systems

ABSTRACT

This paper is concerned with the application of Kalman filter based methods for Fault Detection and Identification (FDI). The original Kalman based method, formulated for bias faults only, is extended for three more fault types, namely the actuator or sensor being stuck, sticky or drifting. To benchmark the proposed method, a nonlinear buffer tank system is simulated as well as its linearized version. This method based on the Kalman filter delivers good results for the linear version of the system and much worse for the nonlinear version, as expected. To alleviate this problem, the Extended Kalman Filter (EKF) is investigated as a better alternative to the Kalman filter. Next to the evaluation of detection and diagnosis performance for several faults, the effect of dynamics on fault identification and diagnosis as well as the effect of including the time of fault occurrence as a parameter in the diagnosis task are investigated.

1. Introduction

Fault Detection and Identification (FDI, Isermann & Ballé, 1997) deals with the timely detection and diagnosis of anomalies in processes or systems and has gained attention since the 1990s. Several philosophies have been adopted in the past, leading to a wide range of available tools (Venkatasubramanian, Rengaswamy, & Kavuri, 2003; Venkatasubramanian, Rengaswamy, Kavuri, & Yin, 2003; Venkatasubramanian, Rengaswamy, Yin, & Kavuri, 2003). A rough classification of methods can be made according to whether the applied methods are deductive or inductive in nature. A typical deductive method will be based on first-principles knowledge while inductive methods are based on recognition of patterns in process data sets, with roots in statistical theory (e.g. Principal Component Analysis) or Artificial Intelligence (e.g. Artificial Neural Networks). Deductive methods, due to their assumption on available first principles knowledge, tend to be more rigorous and accurate in nature. However, the cost of accurate knowledge or models may be prohibitive so that only inductive methods may be achievable in practice. Quite naturally, hybrid approaches are applicable, e.g. when first principles knowledge is available to some extent but not for the whole system. Another way of categorizing FDI methods may be based along the internal representations used. For sure, quantitative representations are the most popular. The Kalman filter adopted for FDI in Prakash, Patwardhan, and Narasimhan (2002) and further extended in this work is a quantitative method in the deductive category. Principal Component Analysis is a quantitative method in the inductive category (Joliffe, 2002). A smaller segment of FDI methods is based on qualitative representations. Examples of such methods in the deductive category are Signed Directed Graphs (SDGs, Maurya, Rengaswamy, & Venkatasubramania, 2004) and qualitative reasoning (Forbus, 1984; Kuipers, 1994). In the inductive category, a large variety of time series trending methods is available (e.g., Akbaryan & Bishnoi, 2001; Bakshi & Stephanopoulos, 1994; Charbonnier, Garcia-Beltan, Cadet, & Gentil, 2005; Dash, Maurya, & Venkatasubramanian, 2004; Flehmig, Watzdorf, & Marquardt, 1998; Villez, 2007), yet little consensus exists on their respective strengths and weaknesses.

The presented work is a result of an ongoing project on state awareness for complex systems. The ultimate aim is to install tools for proper identification of potentially harmful situations in safety-critical systems. This aim fits into a larger vision on design of resilient systems, i.e. systems that only degrade gradually or gracefully when subject to series of harmful events (Rieger, Gertman, & McQueen, 2009). In this contribution, we focus on the extension and critical evaluation of an existing method for Fault Detection and Identification (FDI) which is based on the Kalman filter. This methods finds itself in the deductive-quantitative section of the FDI

* Corresponding author. Tel.: +1 765 494 2430; fax: +1 765 494 0805.
E-mail address: kris.villez@gmail.com (K. Villez).

0098-1354/$ – see front matter
doi:10.1016/j.compchemeng.2011.01.045

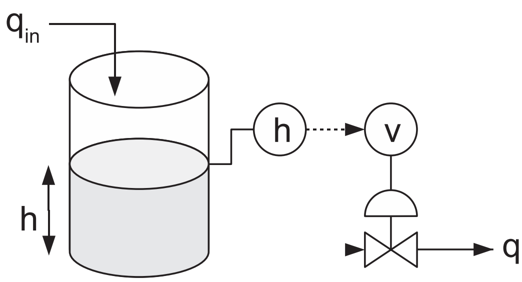

Fig. 1. Scheme of the simulated tank system.

methods. This method has been tested successfully for fault detection and diagnosis (Prakash et al., 2002). In particular, the method has been shown to allow proper detection and diagnosis of biases in different actuator and sensor locations as well as correction of the on-line model predictive control (MPC) scheme for identified faults (Prakash, Narasimhan, & Patwardhan, 2005). However, using only bias faults and the assumption on linearity may be considered limiting. This study therefore concentrates on (1) the extension of the method to allow the separate identification of stuck behavior, stiction, bias and drifts in sensors and actuators and (2) the evaluation of the method on both a non-linear system as well as its linearized version. As such, the simulation study allows to evaluate whether the method is applicable for non-linear systems. In what follows, Materials and Methods will be explained first. Then, results will be shown with broader discussions in separate sections. Finally, the most important conclusions are summarized in the last section.

2. Materials and methods

2.1. Benchmark simulations

Two models are used to benchmark the developed methods for FDI. Both are models of a buffer tank with a pipe connected at the bottom of the tank and with one end open to the atmosphere. The pipe is equipped with a valve. Fig. 1 shows a scheme of the system. A feedback PI controller adjusts the valve position to achieve the setpoint for the tank level based on the measurement of the tank level. One of the models is a non-linear and more realistic version of such a system. The other is the linearized version of this model, obtained by linearization around the nominal operating point. The following paragraphs explain the two models.

2.1.1. Non-linear system

The open-loop system can be written as a Differential Algebraic Equation (DAE) with the tank level (h) as the dynamic state and the outflow rate (q_{out}) as the algebraic state. The valve position (v) is a manipulated input and the inflow rate (q_{in}) a disturbance input. Because the algebraic equation can be solved analytically, one can rewrite the open-loop system model as a single Ordinary Differential Equation (ODE) (Appendix A). The steady-state nominal operation is defined by the tank level ($h_o = 10\,\text{m}$) and valve opening ($v_o = 50\%$), from which the corresponding steady state mean inflow rate can be computed ($q_{in,o} = 5.36\,\text{m}^3/\text{s}$). All simulations are started with this steady state condition. All parameter values of the nonlinear model are listed in Appendix B.

2.1.2. Linear system

To obtain the second benchmark system, the nonlinear model was linearized by means of evaluating the derivatives at the nominal operating point (Appendix C). For this linearized system, the use of the (linear) Kalman filter is theoretically optimal. The results obtained with this linearized system will function as a reference for evaluation of the results obtained in the non-linear case.

2.1.3. Introducing faults and noise

To test the FDI strategy, several fault classes were simulated for both systems. In this paper, we consider a fault type a kind of symptomatic behavior, irrespective of its location. A fault class is defined as the unique combination of fault location and fault type. The simulated fault types are stuck behavior, stiction, bias and drift. These four fault types are introduced in two locations, namely the valve position and the tank level measurement. This leads to 8 different fault classes. Stiction, particularly in valves, has been shown to be relevant in an industrial context (Choudhury, Thornhill, & Shah, 2005; Srinivasan & Rengaswamy, 2008). However, its identification in a context where other faults are possible has not been considered yet. With $u(t)$ the valve position signal delivered by the controller, $u_f(t)$ the corrupted valve position, t_f the time of fault occurrence and δ the fault parameter, the different models for the valve faults are as follows:

$$\begin{array}{llll}
\text{No fault:} & u_f(t) & = u(t) & \\
\text{Stuck:} & u_f(t) & = u_f(t-1) & \\
\text{Stiction:} & u_f(t) & = u(t) & |h(t) - h_f(t-1)| > \delta \\
 & & = u_f(t-1) & |h(t) - h_f(t-1)| \leq \delta \\
\text{Bias:} & u_f(t) & = u(t) + \delta & \\
\text{Drift:} & u_f(t) & = u(t) + \delta \cdot \dfrac{(t - t_f)}{100\,\text{s}} &
\end{array} \tag{1}$$

For the sensor, the equivalent models are obtained by replacing $u(t)$ and $u_f(t)$, with the true tank level, $h(t)$, and the corrupted tank level measurement, $h_f(t)$, respectively. It is noted here that the bias and drift fault types are additive while the stuck and stiction faults are non-additive. This has important implications for fault identification as will shown further.

The resulting corrupted signals ($u_f(t)$, $h_f(t)$) as well as the input flow rate ($q_{in,o}$) are subjected to white noise to obtain the actual tank level measurement, the actual valve position and the actual input flow rate ($y(t)$, $v(t)$ and $q_{in}(t)$) as follows:

$$\begin{aligned}
y(t) &= h_f(t) + e_1, \quad e_1 \sim N(0, \sigma_1^2) \\
v(t) &= u_f(t) + e_2, \quad e_2 \sim N(0, \sigma_2^2) \\
q_{in}(t) &= q_{in,o} + e_3, \quad e_3 \sim N(0, \sigma_3^2)
\end{aligned} \tag{2}$$

Appendix B lists the applied values for the noise standard deviations (σ_1, σ_2 and σ_3).

2.1.4. Simulated scenarios

To make sure that fault detection and diagnosis results are independent of other faults, several short scenarios are simulated rather than a single process history. Both the nonlinear and linear system are simulated repeatedly for 200 s. This is done for combinations of several fault scenarios, which describe the simulation of faults, and setpoint scenarios, which describe the time profile of the tank level setpoint.

Fault scenarios. All faults are introduced at 101 s in the simulation. The stiction, bias and drift faults are introduced with three different parameter values, namely 5, 10 and 15 % of the value at nominal operating point. A faultless scenario is also simulated. A total of 21 fault scenarios thus results. Table 1 summarizes these scenarios and provides a fault class index (0–8) for all the fault classes.

Setpoint scenarios. Each of the above fault scenarios is repeated for two different setpoint scenarios. In the first scenario, SP1, a setpoint change of 10% is introduced at the start of the simulation (1 s). In these cases, the controller dynamics have settled by the time that the fault is introduced. In the other scenario, SP2, the same setpoint change is introduced at 101 s in the simulation, along with the intro-

Table 1
Simulated fault scenarios.

Fault class	Fault type	Fault location	Fault parameter values
0	None	–	–
1	Stuck	Valve	–
2	Stiction	Valve	$5 - 10 - 15\% \cdot v_o$
3	Bias	Valve	$5 - 10 - 15\% \cdot v_o$
4	Drift	Valve	$5 - 10 - 15\% \cdot v_o$
5	Stuck	Sensor	–
6	Stiction	Sensor	$5 - 10 - 15\% \cdot h_o$
7	Bias	Sensor	$5 - 10 - 15\% \cdot h_o$
8	Drift	Sensor	$5 - 10 - 15\% \cdot h_o$

Table 2
Simulated setpoint scenarios.

Setpoint scenario	Magnitude	Time of change
SP1	10%	1 s
SP2	10%	101 s

duced fault (if any). By means of these two setpoint scenarios, a crude analysis of the effect of dynamics on the FDI performance becomes possible. Table 2 lists the different setpoint scenarios.

According to the above description, each fault scenario (21) is combined with each setpoint scenario (2). This leads to a total of 42 simulations for each of the two considered systems (nonlinear and linear).

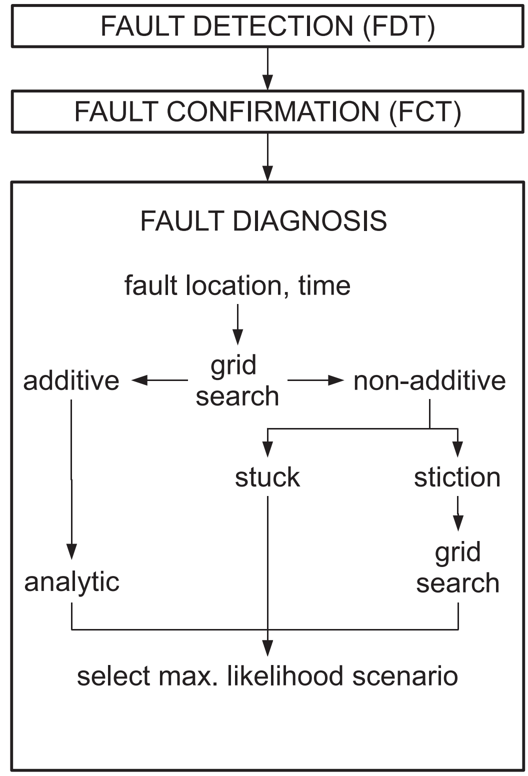

Fig. 2. Scheme of the applied Fault Detection and Identification strategy.

2.2. *Fault Detection and Identification*

2.2.1. *Fault Detection and Identification via the (linear) Kalman filter (KF)*

A Fault Detection and Identification (FDI) strategy based on the Kalman filter is applied for both the linear and the nonlinear system. This FDI strategy is an extension of the method as in Prakash et al. (2002). This original strategy allows to detect and identify biases in multiple sensors and inputs and assumes exact knowledge (structure, parameters) of the linear state-space model with known additive, Gaussian noise in inputs and outputs. The proposed extensions include (1) that four fault types (stuck behavior, stiction, bias and drift) are considered in the fault diagnosis method rather than one (bias) and (2) that the time of fault occurrence is considered a parameter of the fault as opposed to assuming that the time of the fault detection is also the time of fault occurrence. Fig. 2 gives an overview of the whole FDI procedure. The individual steps are described in the next paragraphs.

Fault detection. For detection, one continuously evaluates the residuals, $r(k)$, between predictions by the Kalman filter and the measurements. The Mahalonobis distance of these residuals to the origin is then computed, using the expected covariance matrix of the prediction residuals, $V(k)$, which is delivered by the Kalman filter:

$$M(k) = r(k)^T \cdot V(k)^{-1} \cdot r(k) \quad (3)$$

This distance follows a χ^2 distribution under assumptions of linearity and Gaussian noise distributions and as long as the model matches the system exactly. A fault detection test (FDT) for abnormal behavior is evaluated at each time instant using the 90% confidence limit (Fig. 2 – top box). A fault confirmation test (FCT) integrates the last 10 FDTs each time a new FDT has been evaluated and signals an alarm if 3 out 10 FDTs were positive (Fig. 2 – middle box). This is slightly different from the original approach in Prakash et al. (2002). In that work, the FCT is based on a joint check of all residuals within a time window after a positive FDT by means of one aggregate χ^2 statistic.

Fault Identification (Diagnosis). The original method in Prakash et al. (2002) permits fast identification of biases by means of an extension of the nominal state-space model. Indeed, the introduction of biases in linear systems results in a linear response which can always be computed in advance up to a factor which depends linearly on the magnitude of the fault. This means that an analytic solution for the magnitude of the bias is available directly. In addition, the likelihood of a bias in a given location (actuator or sensor) is easily computed. Diagnosis then simply finishes with the selection of the fault scenario with maximum likelihood. Two critical remarks can be made with respect to this method. The first one is that other, more complex, faults are not considered in the diagnosis step. We therefore extend the method here for the faults described above. A second remark with respect to the original method is that the effect of the start time of the fault is not optimized in the search for the correct fault type and location.

Based on the above remarks, the following strategy is devised. For all considered fault types, a grid search is executed to search for the most likely time of fault occurrence. For each of the considered times, one fits the remaining parameters for each available fault class (location, type). To this end, the considered times of fault occurrence range from 50 samples before the time of positive FCT until time of positive detection in discrete steps (51 start times total). After this optimization, the likelihood for each considered fault scenario (time of occurrence, location, type) is available. From the set of fault scenarios, the one with maximum likelihood is selected as the final result of diagnosis. Except for the fit of the fault parameters, all steps are the same for all fault types (Fig. 2 – bottom box). In the succeeding paragraphs, the applied strategies for additive and non-additive faults are explained.

For the additive fault types (bias, drift), one can find the most likely magnitude parameter values analytically. This is explained in detail for the bias type in Prakash et al. (2002). In essence, the method described the expected profile of the Kalman prediction residuals for a given fault class (type + location) up to a constant which is proportional to the magnitude of the bias. Because of the assumption of linearity, the response of the system to a drift fault is simply the sum of responses to biases of equal magnitude introduced at each time following the start of the drift fault. As a result, the system's response to a drift is also additive and can also be computed upfront up to a factor reflecting the magnitude (speed) of the drift. Furthermore, this means that the drift parameter and the associated likelihood can be computed analytically. As such, the inclusion of drift is the most straightforward of the proposed extensions.

For the non-additive faults (stuck, stiction), the situation is more complicated. Here, no analytical solution exists and one needs to run the Kalman filter over the time series for each considered parameter set for a fault scenario. For both faults, one thus simulates the system under the proposed fault and evaluates the likelihood. For the stuck fault type, this is fairly easy since the time of occurrence is the only parameter. It is computationally more intensive for the stiction fault type as one also needs to optimize the stiction band parameter (Fig. 2 – bottom box – right). In our work, a grid search strategy is used with values ranging for the stiction band parameters ranging from 0 to 25% in steps of 1%. For a given fault start time, the maximum likelihood is used to compare with other fault scenarios.

The above fault identification strategy is compared to two other strategies, which are slight modifications of the above. In the first strategy, the presence and the time of fault occurrence is assumed to be known (*actual time*). Naturally, this is not realistic situation but the results following from this approach form a reference to compare with. A second approach is to assume that the fault started at the time of fault confirmation (*detection time*). The detection time is considered as the first time a fault is confirmed (first positive FCT) after the start of the fault. This can be considered a naive approach. The third and most advanced approach is the complete approach as described above (*optimized time*). The second and third approaches both require a detection for the execution of diagnosis. As a result, when detection fails, diagnosis fails as well. Evaluation of each of these three approaches is executed at the end of the simulation. This makes sure that the same amount of information (samples) is available each time.

2.2.2. Fault Detection and Identification via the Extended Kalman Filter (EKF)

For the nonlinear system, the above strategy is repeated with the Extended Kalman Filter (EKF). In this case, the EKF is used for fault detection and to obtain prediction residuals. To this end, the state prediction equation is replaced by the original nonlinear model. All other equations (covariance prediction, covariance update and Kalman gain) are based on the linearization of the nonlinear model around the state estimate. The diagnosis part is left unchanged except that for each sample in the data history, one uses the corresponding linearized model as obtained by means of the EKF in the fault detection step. Since these linearized models are obtained under the fault-free assumption, one thus assumes that the presence of a fault has little effect on the linearized model.

2.3. Combinations of simulated systems and applied Kalman filters

Three combinations of simulated system and applied Kalman filters are considered. The first combination consists of the application of the (linear) Kalman filter to the linearized system. The obtained results are indicated as *L-KF* results. This is a reference case as the results are theoretically optimal. The second combination consists of the application of the (linear) Kalman filter to the nonlinear system. We use *NL-KF* for short-hand notation. Because of mismatch between the model used by the Kalman filter and the simulated system, one can expect poor results. The third combination consists of the application of the EKF to the nonlinear system, using *NL-EKF* as short-hand notation. This is reduces model mismatch by means of repeated linearization of the nonlinear model around the estimated state. Table 3 gives an overview of the described combinations.

Table 3
Combinations of simulated systems and applied Kalman filter.

Short-hand notation	Simulated system	Applied Kalman filter
L-KF	Linear	(linear) Kalman filter
NL-KF	Nonlinear	(linear) Kalman filter
NL-EKF	Nonlinear	Extended Kalman Filter (EKF)

2.4. Evaluation of FDI performance

To assess the performance of the tested strategies throughout the simulation study, several measures are evaluated. These are explained in the following paragraphs.

Total misclassification rate. The total misclassification rate is the fraction of time over which the obtained outcome is equal to the target outcome. For fault detection, it is assumed that the obtained class is 'normal' before a positive FCT and 'faulty' as from a positive FCT. Since all faults are introduced at 100 s in the simulation, the target outcome is 'normal' for the first 100 samples and 'faulty' for the last 100. For faultless scenarios, all (200) target outcomes are 'normal'. For diagnosis, the total misclassification rate is the fraction of the time during which the fault class is correctly identified after execution of the diagnosis task. Here, the target outcome is '0' (faultless) during the first 100 samples and the simulated fault class index (1–8) in the last 100 samples. The fault diagnosis performance is not evaluated for the faultless scenarios. With T the target outcome and P the obtained class, the formula for the total misclassification rate reads:

$$\text{Total Misclassification Rate} = \frac{1}{K}\sum_{k}^{K}(T(k) \neq P(k)) \tag{4}$$

Type I and Type II error rates. For detection performance, the total misclassification rate can be decomposed in two, namely the Type I error rate and the Type II error rate. The Type I error rate is computed as the time in which the system is truly in a normal state and the obtained outcome is 'faulty' divided by the time in which the system is truly in a normal state. In analogy, the Type II error rate is computed as the time in which the system is in a faulty state and the outcome is 'normal' divided by the time in which the system is in a faulty state. The Type I and Type II error rates are only used for evaluation of detection performance. The respective formulas are as follows:

$$\begin{aligned}\text{Type I Error Rate} &= \frac{\sum_{k}^{K}(T(k) \neq P(k)\,\&\,T(k) = \text{'normal'})}{\sum_{k}^{K}(T(k) = \text{'normal'})}\\ \text{Type II Error Rate} &= \frac{\sum_{k}^{K}(T(k) \neq P(k)\,\&\,T(k) = \text{'faulty'})}{\sum_{k}^{K}(T(k) = \text{'faulty'})}\end{aligned} \tag{5}$$

Each of the above performance measures are set up so that a lower measure means better performance. These measures

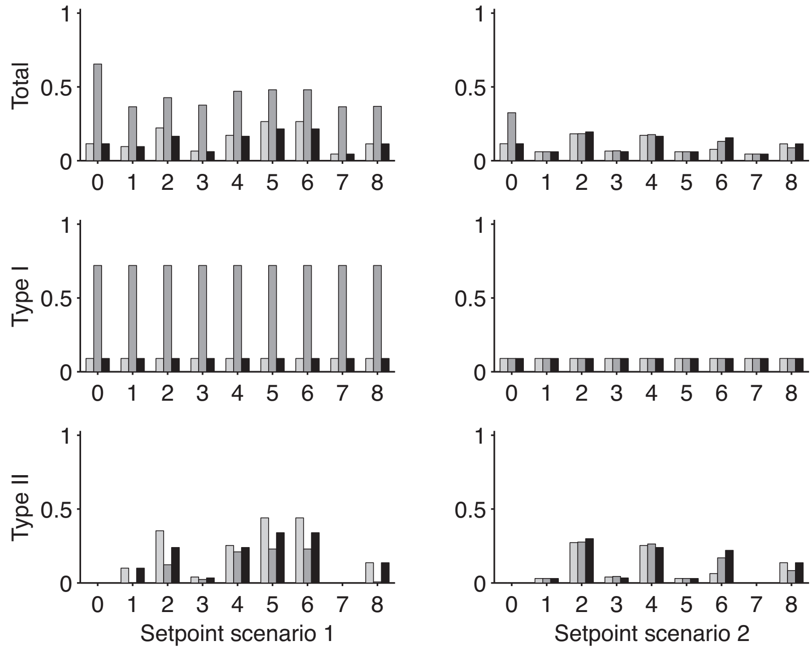

Fig. 3. Fault detection performance. The left panels correspond to SP1 scenarios (setpoint change at 1 s in the simulation) while the right panels correspond to the SP2 scenarios (setpoint change at 101 s in the simulation). The top, middle and bottom panels display the total misclassification rate, Type I error rate and Type II error rate, respectively. In each panel, performance measures are grouped per fault class (0–8 see Table 1). The light gray (left), dark gray (middle) and black (right) bar in each group represent the results for the linear system using the KF (L-KF), the nonlinear system using the KF (NL-KF) and for the nonlinear system using the EKF (NL-EKF), respectively.

may be considered crude as they do not take into account that the impact of a fault may be influenced by process states and the nature and location of the fault. However, they allow sufficiently for a comparative analysis between different methods and strategies.

3. Results

Simulations were executed as explained above for all fault scenarios, for two setpoint scenarios (SP1 and SP2) and for two systems (linear and non-linear). The same Kalman based strategy for FDI was applied in each simulation. For the linear system, this strategy was applied with the (linear) Kalman filter (KF) only (L-KF). For the non-linear system, both the KF and the EKF were applied (NL-KF and NL-EKF). Fault classes 2–4 and 6–8 were executed with different values for the fault parameter (see Table 1). For these classes, the obtained performance measures were averaged. The following paragraphs describe the obtained results.

3.1. Fault detection

As discussed above, an alarm or detection is considered based on the fault confirmation test (FCT). In all that follows, fault detection refers to an alarm generated by means of the Fault Confirmation Test (FCT).

The detection performance measures are displayed for all simulations in Fig. 3. At the top-left, one sees the total misclassification rate for each evaluated combination of simulated system and Kalman filter for the SP1 scenarios (setpoint change introduced at the start). It can be seen that the misclassification rate is always the worst when the KF is applied to the nonlinear system (NL-KF). The larger misclassification rate is largely due to the effect of a large fraction of Type I errors (Fig. 3, middle-left). Clearly, application of the KF to the nonlinear system results in a too sensitive fault detection method. This is due to mismatch of the linearized model and the actual system. The misclassification results for the KF applied to the linear model (L-KF) and the EKF applied to the nonlinear model (NL-EKF) are similar (Fig. 3, top-left). Moreover, the Type I error rates are exactly the same for the L-KF and NL-EKF cases. Type II error rates show more variation (Fig. 3, bottom-left). Here, for each fault class, the NL-KF case results in the best performance. This is due to a high sensitivity of the statistical test. However, it is noted that this also leads to very high Type I error rates, as shown earlier. Overall, fault class 3 (valve bias) and fault class 7 (sensor bias) are the easiest to detect. The worst Type II error rates are found for fault fault class 5 and 6 (sensor stuck, sensor stiction). The Type II error rates for the NL-EKF cases are always equal or lower than those for the L-KF cases. This is explained as a slight increase in sensitivity in the NL-EKF case. This follows from an inexact match of the linearized form of the model which is used in the EKF. Because this only affects the covariance estimates and not the state estimates, this effect is not visible in the observed Type I error rate. Indeed, one would expect a slightly larger Type I error rate for the NL-EKF case if a larger number of repetitions of the same simulations were executed.

At the right hand side of Fig. 3, one finds the equivalent results for the case when the setpoint change is introduced simultaneously with the start of the fault (SP2). In doing so, the simulated system is expected to exhibit dynamic behavior, irrespective of the nature of the fault. This reduces the total misclassification rate considerably for the NL-KF cases (Fig. 3 top-right). This improved performance is explained as follows. In the first, faultless half of each simulation, the system operates in its nominal operating point which is also the point around which the Kalman filter model is linearized. As a result, little mismatch exists between the simulated system and the applied model during this time. As only this first half of the simulation is used for evaluation of the Type I error rate, results are comparable to the other cases (L-KF and NL-EKF, Fig. 3 middle-right). While this may be regarded as a positive result, the advantage of this is minimal since engineered systems seldom operate at a single nominal operating point at all times. In practice, false alarm rates as for the SP1 scenarios should be expected, as severe mismatch exists between the simulated system and the applied model (Fig. 3 middle-left).

When comparing the SP2 results with the SP1 results, one also observes improved performance for fault class 5 and 6, irrespective of the simulated system or the applied filter. To investigate what causes this difference, Fig. 4 displays the measurements and true values of the tank level for four simulated scenarios. The left panels correspond to SP1 simulations (setpoint change at 1 s) with fault class 0 (faultless, top) and fault class 6 (sensor stiction, bottom). The right panels repeat the same simulation yet for the SP2 (setpoint change at 101 s). Clearly, the presence of sensor stiction has no visible influence on the measurements of the tank level measurement in the SP1 simulations (Fig. 4, top-left and bottom-left). Because the tank level is at setpoint and in steady state, the demanded controller action is practically the same in both cases. As a result, the obtained data series are indistinguishable. In contrast, the SP2 simulations are visibly different (Fig. 4, top-right and bottom-right). Indeed, it is due to the introduction of dynamics by means of a setpoint change that one observes symptomatic behavior associated with the fault (oscillations). In contrast to the SP1 scenarios, the controller attempts to displace the valve to achieve the setpoint, leading to different behavior than predicted by the Kalman filter, in turn leading to positive detection. This results in a lower Type II error rate for these classes (Fig. 3 bottom-right). As such, fault class 5 and 6 (sensor stuck, sensor stiction) represent a particular set of fault types for which symptoms do not necessarily appear immediately upon their introduction. One could expect similar results for valve stuck and stiction behavior. In our particular case, this does not occur because of a relatively aggressive tuning of the integral time constant of the PI controller. Because of this, the integral action results in variations in the valve position signal which are large enough to always generate distinguishable time series (not shown).

A last observation made on the basis of the SP2 results is that the Type II error rate is lower for fault class 1, 3, 5 and 7 (valve stuck, valve bias, sensor stuck and sensor bias) than for the other fault classes (valve stiction, valve drift, sensor stiction and sensor drift). For the stiction type of faults, this is because of a relatively aggressive tuning of the PI controller. Because of this aggressive tuning, the changes in valve position demanded by the PI controller immediately after the setpoint change are large enough to surpass the stiction band. Only as the controller actions settle, they become smaller than the stiction band and corresponding symptoms appear. The lower the stiction band, the more pronounced this effect becomes. As such, a lower stiction band is then expected to result in a longer time to detect it. For the drift type of faults (fault class 4 and 8) this is because it takes a while before the drift results in deviations of such magnitude that a positive FCT results.

3.2. *Fault identification (diagnosis)*

In the following paragraphs, the results for fault identification are shown. The overall misclassification rate is used as the measure of performance and is computed only over the second half of the simulation, in which the faults are present. A rate of 1 means then that the assigned fault class is incorrect throughout this time frame. A rate of 0 means that the fault is correctly identified as well as its time of occurrence. Again, the performance is evaluated for both setpoint scenarios, namely SP1 (setpoint change at 1 s) and SP2 (setpoint change at 101 s). This also executed for the three combinations of simulated system and Kalman filter (L-KF, NL-KF, NL-EKF), as before. Results are evaluated for fault classes 1 to 8, thereby excluding the faultless operation class. As described above, three different approaches are evaluated. The first approach is to assume that it is known that a fault is present as well as its time of occurrence (*actual time*). Thus, the results only reflect on the ability to discern between fault types and therefore serve as a reference. The second approach is to execute diagnosis only following a positive Fault Confirmation Test (FCT) while assuming that the time of confirmation is also the time of fault occurrence (*detection time*). The third and most elaborate approach is to perform a grid search for the most likely time of occurrence over a time window ranging from 50 samples before detection to the time of detection (*optimized time*). Naturally this approach is computationally much more expensive than the first two. The diagnosis task is always executed at the end of the simulation.

3.2.1. *Actual time: assuming the actual fault occurrence time to be known*

Fig. 5 shows the results for the fault identification (diagnosis) task. We discuss the results for the linear system and KF first (light gray bars). The top panels in Fig. 5 represent the misclassification rates for the reference cases where the fault presence and time of fault occurrence are assumed known (Actual time). At the top-left (SP1, see Table 2), one observes that the fault identification is perfect for fault class 1 to 4, 7 and 8 (see Table 1) as the misclassification rate is zero. In contrast, fault class 5 and 6 are not classified correctly at any point (misclassification rate is one). When comparing to the the top-right (SP2, see Table 2), one sees in this case perfect classification is obtained for fault class 5 and 6. As such, the presence of dynamics, like the ones induced by the setpoint change in the SP2 scenarios is observed to assist in obtaining a correct fault identification. Still, it is not a complete surprise since the the setpoint scenario also affected the detection performance and most dramatically so for fault class 5 and 6. Moreover, it was seen that the symptoms associated with fault class 6 (oscillations) only appear in the SP2 case. It is noted that since fault presence and time are assumed known this observed effect is not confounded by effects of fault detection performance.

Now we discuss the misclassification rates for NL-KF and NL-EKF cases, still for the case of a known fault occurrence time (Fig. 5, top panels). The NL-KF results are given as dark gray bars; the NL-EKF results as black bars. For the SP1 setpoint scenario, the NL-KF results are the same except for fault class 1 and 8, where the misclassification rate is higher than the L-KF result. Fault class 1 is never identified correctly, resulting in a misclassification rate of 1. Fault class 8 is identified correctly twice (of three instances), resulting in a 33% misclassification rate. In the NL-EKF case, one obtains perfect classification for fault class 8 (sensor drift) as in the L-KF case. Thus, by means of accounting for nonlinearity with the EKF, one regains the theoretically optimal result of the L-KF case. For fault class 1 and 6, no and little improvement is observed compared to the NL-KF case. For these fault classes, it does not help to account for system nonlinearity by means of the EKF. In addition, the misclassification rate for fault class 4 (valve drift) becomes equal to one as opposed to zero in the L-KF and NL-KF case.

The results for SP2 simulations with the NL-KF combination indicate improvement of the misclassification rate for fault class 1, 3, 5 and 7. Note that the detection performance was also better for these classes (Fig. 3). For fault class 6 (sensor stiction), one out of three simulated instances is classified correctly, leading to a 66% misclassification rate. For fault class 2, one instance is not identified correctly, leading to a 33% misclassification rate (as opposed to 0 for SP1 simulations). For the NL-EKF case, SP2 results are the same as SP1 results, except for fault class 6. Here, the misclassification rate is 0 for the SP2 instead of 66% for SP1. Thus, one observes that the presence of dynam-

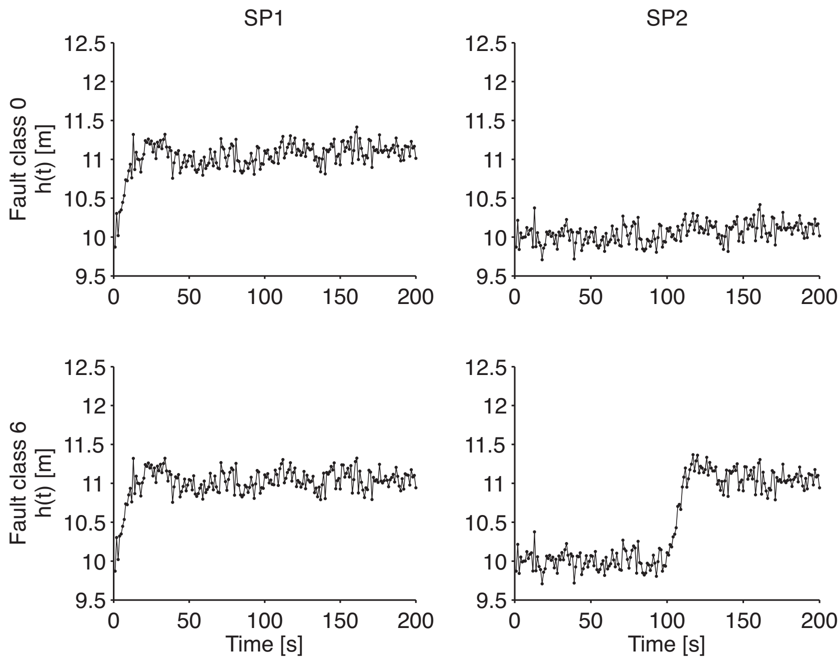

Fig. 4. Tank level measurements for a faultless simulation (top) and sensor stiction fault (bottom), for both SP1 (left) and SP2 (right) scenarios.

ics also generally has a beneficial effect on the fault identification performance. The results for fault class 2 and 6 for the NL-KF combination contrast with this statement. This is attributed to the mismatch of the linear model with the nonlinear system that is simulated.

3.2.2. Detection time: using the detection time as the time of fault occurrence

In the middle of Fig. 5, one finds the panels corresponding to the misclassification rates obtained when the fault start time is assumed to be fault detection time (detection time). Again focusing on the L-KF results first (light gray bars), one observes that

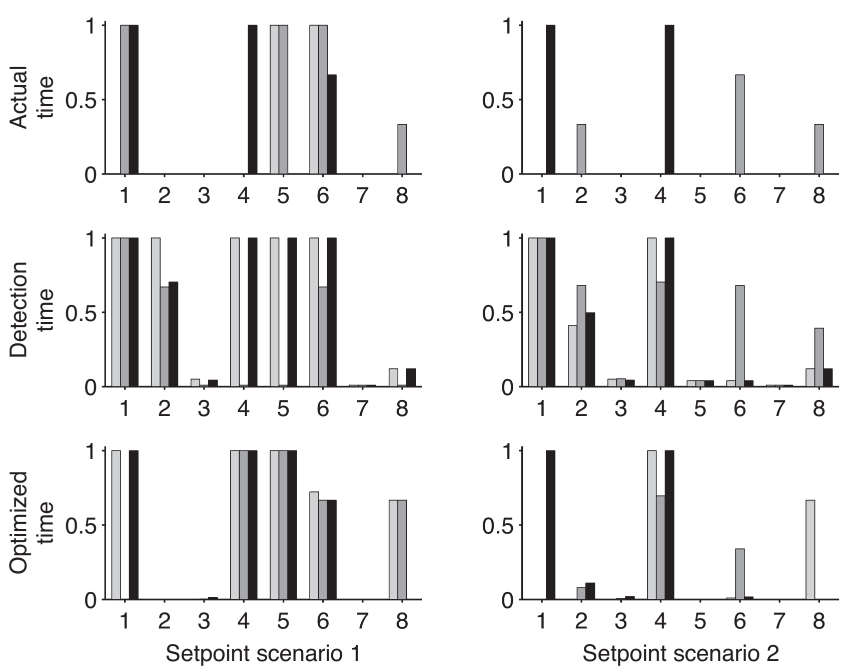

Fig. 5. Total misclassification rates for fault identification. Left panels correspond to SP1 scenarios (setpoint change at 1 s in the simulation); Right panels correspond to the SP2 scenarios (setpoint change at 101 s in the simulation). Top, middle and bottom panels display the total misclassification rate for the time of fault occurrence to be known (actual time), to be assumed at positive FCT (detection time), to be searched (optimized time), respectively. In each panel, performance measures are grouped per fault class (1–8, see Table 1). The light gray (left), dark gray (middle) and black (right) bar in each group represent the results for the linear system using the KF (L-KF), the nonlinear system using the KF (NL-KF) and for the nonlinear system using the EKF (NL-EKF), respectively.

the misclassification rate increases for some SP1 simulations (Fig. 5, middle-left), compared to the results for a known fault occurrence time (Fig. 5, top-left). Indeed, for fault class 1, 2 and 4 (valve stuck, stiction and drift) the misclassification rate is now 1. Thus, assuming the wrong time of fault occurrence can lead to erroneous fault identification results. For fault class 3 and 7 (valve bias, sensor bias) only a slight increase of the misclassification rate results. This is because the detection is very quick, thereby resulting in a detection time which is fairly close to the actual time of the fault. Fault class 5 and 6 remain to be diagnosed wrongly.

For the SP2 scenarios (Fig. 5 middle-right, light gray bars), the misclassification rate also increases when compared with the case when the fault occurrence time is assumed known. However, in comparison to the equivalent SP1 results (Fig. 5, middle-left), the misclassification rate is lower for fault class 2, 5 and 6. Thus, also in this case a beneficial effect of the simultaneous setpoint change and fault is observed. It was discussed above that earlier detection results because demanded control actions make symptoms appear in the case of fault class 5 and 6 in the SP2 simulations. For SP1 simulations, these control actions are not large enough to make the symptoms appear. It is this same effect of introduced dynamics which makes fault identification easier for fault class 5 and 6 in the SP2 simulations. A similar effect results for diagnosis of fault class 2. For the SP1 simulations, the demanded changes in the valve position are so low that the stiction band is never surpassed. As a result, the behavior is practically the same as for a stuck valve, leading to an erroneous classification. For the SP2 simulations, this only happens for the instance with the largest stiction band value (15% of nominal value). Interesting enough, fault identification is thus easier for a smaller stiction band parameter.

In the NL-KF case (Fig. 5 middle-left, dark gray bars), one observes that the misclassification rates for fault class 4 and 5 (valve drift, sensor stuck) improve quite dramatically when compared to those for the L-KF case for some classes. This is as a result of the model mismatch between the simulated system and the applied model for the Kalman filter. As a result, the observed detection time is much closer to the actual time of the fault occurrence which, in turn, improves the fault identification performance. The same effect occurs to a lesser extent for fault class 2, 3, 6 and 8 (valve stiction and bias, sensor stiction and drift). The NL-KF results for the SP2 simulations (Fig. 5, middle-right, dark gray bars) show that the misclassification rate is increased by large for fault class 4 and 8 (valve and sensor drift) and to a lesser extent for fault class 3 and 5 (valve bias, sensor stuck). Here, we presume that this is due to a combined effect of model mismatch and presence of normal process dynamics, in turn due to the setpoint change at the the time of fault occurrence. However, as the Kalman filter model is inappropriate for the nonlinear system, it is hard to pinpoint the exact reason with certainty.

For SP1 simulations in the NL-EKF case (Fig. 5, middle-left, black bars), the positive effect as observed in the L-KF case for the SP1 simulations is not observed. Naturally, this because the model mismatch is significantly reduced in this case. Indeed, the EKF accommodates to a large extent for model nonlinearity. It is therefore not so surprising that the result are very similar to the L-KF case. Only for fault class 2 (valve stiction), an improvement is observed. This is due to a correct identification of the fault class for the instance with the lowest magnitude (5% of nominal value) for the stiction band, in turn due to a detection time closer to the actual time of fault occurrence. Similar observations hold for the SP2 simulations (Fig. 5 middle-right, black bars). Also in this case, the NL-EKF results are practically the same as the L-KF results. This includes fault class 2 now. This is because the presence of dynamics results in earlier detection times which also happen to be closer to each other, as opposed to the results for the corresponding SP1 simulations. Note that these results do not suggest the it would be beneficial to use the (linear) Kalman filter for the nonlinear system. As discussed above already, Type I error rates are so high that such an approach cannot be recommended at all.

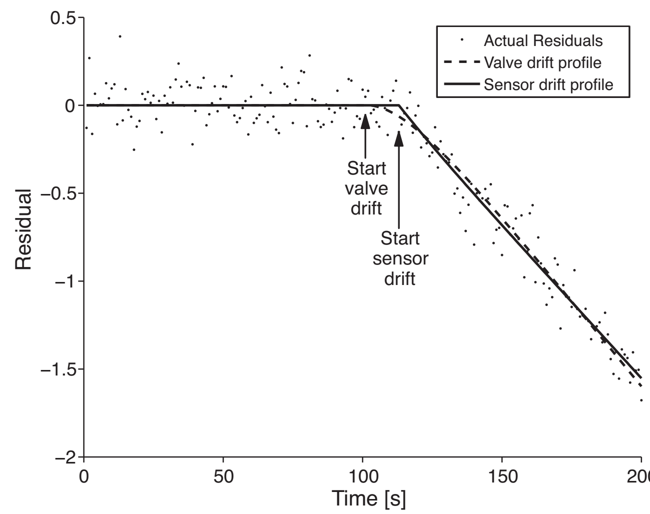

Fig. 6. Kalman prediction residuals for a valve drift and optimized profiles for the valve drift and sensor drift fault classes.

3.2.3. Optimized time: searching the time of fault occurrence as part of the FDI strategy

The results for complete proposed FDI strategy, including optimization of the fault occurrence time, are shown in the bottom panels of Fig. 5. Once more, we discuss the L-KF results for the SP1 scenarios first (Fig. 5, bottom-left, light gray bars). Compared to the corresponding results obtained when assuming the detection time as the actual time of fault occurrence (Fig. 5, middle-left), misclassification rates are lower for fault class 2, 3, 6 and 7. Even more, the original perfect classification performance obtained when assuming the fault occurrence time to be known (see Fig. 5, top-left), is recovered for fault class 2, 3 and 7 (valve stuck, valve bias and sensor bias). For fault class 6 (sensor stiction), the result even improves. For fault class 1, 4 and 5, the inclusion of the fault occurrence time as a parameter does not improve the results in comparison with the case for the fault occurrence time assumed to be the detection time. For fault class 8 (valve drift, sensor drift), the misclassification rate increases when compared to the case for the detection time assumed to be the fault occurrence time.

The situation improves again for the SP2 simulations (Fig. 5 top-right). Now the perfect classification result in the case where the fault occurrence time is assumed to be known is recovered for fault class 1, 2, 3, 5 and 7 (valve stuck, stiction and bias; sensor stuck and drift). A very low misclassification rate results for fault class 6 (sensor stiction). Clearly, the dynamic response due to controller action has a beneficial role on fault diagnosis. However, the misclassification rate remains high for fault class 4 and 8 (valve drift, sensor drift). This is therefore investigated in more detail.

For the SP1 simulations of fault class 4 (valve drift), all three instances are classified as fault class 8 (sensor drift). For fault class 8 (sensor drift), only the instance with the largest drift parameter is classified correctly. The other two instances are classified as fault class 4 (valve drift). Interesting enough, the fault identification procedure thus leads to considerable confusion between fault 4 and 8. Fig. 6 demonstrates why this happens. The Fig. shows the prediction residuals for the valve drift instance with largest drift parameter (accumulated drift attains 15% of nominal value in 100 s). The optimized profiles are shown for a valve drift and a sensor drift. As can be seen, these two profiles are fairly similar and both fit the residuals really well. Because of this similarity between the two profiles, it is generally hard to discriminate these faults in a noisy situation. In our simulation study, this leads to an erroneous result for 5 out of 6 of drift instances.

For the SP1 simulations (Fig. 5, bottom-left), the misclassification rates for the NL-KF and NL-EKF case are very similar to the L-KF case. For instance, perfect or close to perfect performance is achieved for fault class 2, 3 and 7 (valve stiction and bias, sensor bias). Notable differences are that (1) for the NL-KF case, fault class 1 is classified perfectly and (2) for the NL-EKF case, fault class 8 is classified perfectly.

When comparing the SP2 simulation results (Fig. 5 bottom-right) with the SP1 simulation results (Fig. 5 bottom-left), one observes that misclassification rates in the NL-KF case improve for some classes, namely fault class 4, 5, 6 and 8 (valve drift, sensor stuck, stiction and drift). For fault class 1, 5, 7 and 8 (valve stuck, sensor stuck, bias and drift) perfect classification is obtained. In the NL-EKF case, large improvements are seen for fault class 5 and 6 (sensor stuck and stiction). Perfect classification is now obtained for fault class 5, 7 and 8 (sensor stuck, bias and drift). The L-KF and NL-EKF misclassification rates are fairly similar, both for the SP1 and SP2 simulations. As such, it is clear that using the EKF for the nonlinear system enables to obtain a performance similar the L-KF case, which is theoretically optimal. However, this is not true for the SP2 simulation of fault class 1. There, the NL-EKF results in misclassification rate of 1, as opposed to 0 for the L-KF case. However, this does not challenge the general observation above.

4. Discussion

4.1. On non-linearity

By means of the simulation study, the fault detection and identification performance was evaluated for a linear and nonlinear system and for several scenarios. For the linear system, the theoretically optimal Kalman filter was used in the FDI strategy (L-KF case). For the nonlinear system, the (linear) Kalman filter (NL-KF case) and the Extended Kalman Filter (NL-EKF case) were used. However, it is stressed the EKF is only deployed in the fault detection and confirmation step. In the diagnosis step, the strategy assumes a linear yet time-variant model with the time-varying system matrices obtained from the EKF application in the fault detection step. This means that those system matrices are obtained under the assumption of a fault-free condition, which is of course not true when an actual fault is introduced.

Based on the fault detection results, it can be concluded that using the Kalman filter for a nonlinear system should be avoided. Indeed, because of model mismatch the Type I error rate rose to a very high level when the controller setpoint was changed to a point different than the nominal one. This means that a lot of false alarms are expected with this approach. Fortunately, by application of the EKF, the detection performance becomes very similar to the theoretically optimal rates obtained in the L-KF case. As such, it can be concluded that, for the purpose of fault detection, nonlinearity in the nonlinear simulated buffer tank system can be tackled by means of the EKF.

A similar conclusion can be drawn for diagnosis. Also in this case, application of the EKF to the nonlinear system leads to similar results for the theoretically optimal case where the Kalman filter is applied to the linear system. This is true despite the fact that the fault diagnosis step makes use of system matrices which are obtained under fault-free assumptions. As such, we conclude that the introduction of the simulated faults does not change the true dynamics of the system to the point that the diagnosis procedure would break down completely. We reckon that such a positive result may be lost upon introduction of more severe faults.

4.2. Importance of dynamics

At several times during the presentation of results, it was noted that the SP2 scenarios, where faults and setpoint changes are introduced simultaneously, lead to better detection and diagnosis performance compared to the SP1 scenarios, where a setpoint change is introduced earlier and steady-state is achieved again at the time of fault occurrence. This is particularly the case for the stuck and sticky sensor fault classes. This effect was shown to be due to the absence of symptomatic behavior in the SP1 scenarios. Because the controlled process variables are at their setpoint and in steady-state, no control action is warranted as long the setpoint remains the same and in the presence of limited disturbances on the process variable. As a consequence, Kalman prediction residuals never become large enough to induce a fault detection. Moreover, even if fault is detected, the symptoms may not necessarily reflect the true fault well. For instance, a stiction problem in the valve lead to results which are equivalent to a stuck problem, as was observed. In such a case, the controller actions demanded are never large enough to surpass the stiction band. In the SP2 scenarios, this problem does not occur. In this case, the memory function of the stuck and sticky sensor fault alters the behavior of the controlled process variable so that substantial deviations are observed between predicted and measured values, also allowing for proper diagnosis. Such benefit of inducing normal dynamics by means of a setpoint change can be expected for non-additive faults like stuck or stiction behavior. Although this was not observed for stuck valve and valve stiction problems, it can be expected there as well in the general case. In our case, this was not observed due to an aggressive tuning of the controller. We further note that the simulated non-additive fault types belong to a particular class of fault types which can be described as a memory-function. It is our belief that the presence of dynamics only has a positive effect on fault detection and identification for such kind of faults. Other non-additive faults do exist however for which we don't expect such an effect. As an example we indicate multiplicative faults which can be modeled as deviations in the system matrices (of a linear model).

More generally, it can be concluded that for the set of faults considered in this work, dynamic behavior has an important effect on diagnostic performance. Contrary to this statement, all FDI methods available today are passive, irrespective of their classifications as inductive/deductive nature or quantitative/qualitative representation. Indeed, today's state-of-the-art provides no methods which actively modify the process dynamics so to obtain more informative data for the purpose of FDI. Given that our results show that introducing dynamics can be beneficial for proper and fast fault identification, a tremendous opportunity does lie ahead. The idea that one can optimize the information content of on-line produced data for better fault detection and diagnosis in a pro-active manner has indeed not been investigated as of yet. As such, we aim to investigate the on-line application of Optimal Experimental Design techniques (OED, Vanrolleghem & Van Daele, 1994) for FDI in the future which will lead to a broad set of new tools, which we tentatively coin as Active Fault Detection and Identification (ActiveFDI) methods.

4.3. Time of fault occurrence

When evaluating diagnostic performance, we compared three approaches with respect to the time of fault occurrence. A first reference approach assumed that one knows of the presence and time of fault occurrence and therefore served as a reference approach. A second approach consisted of assuming that the time of fault detection is the time of fault occurrence. In a third, the time of fault occurrence was considered a parameter to be estimated as part of the diagnosis task. Generally speaking, the second approach leads to worse results than the first and the third approach leads to better results than the second. It is therefore logical to conclude that the consideration of the time of the fault is important to arrive at correct fault identification results. While this is not a surprise, it is an aspect that has not been investigated thoroughly in previous studies. Proper fault identification becomes difficult however, as the fit objective function is a non-convex function of this time parameter. In this work, we resorted to a grid search for the time of fault occurrence. Unfortunately, this approach is computationally inefficient.

Several strategies may be considered to obtain a more efficient strategy. One may consider a reduced window for the time of fault occurrence, thereby reducing computational demands. This may be based on further analysis of the prediction residuals obtained in the fault detection step. Also, one could optimize fault class and fault time independently in a two-step procedure, rather than jointly. This can be iterated if necessary. As a third alternative, one may seek to an alternative parameterization so that the problem becomes convex or even linear in the parameters. The implementation and evaluation of such alternatives are considered for further research.

5. Conclusions

In this work, a nonlinear buffer tank system was considered to evaluate and benchmark a set of Kalman based techniques for the task Fault Detection and Identification (FDI). The original technique, based on the conventional Kalman Filter was extended for 3 more types of faults, namely stuck, sticky and drifting behavior, in addition to the original bias type of fault. This was shown to work well for a linearized version of the full non-linear model, both for fault detection and diagnosis. An exception to this was observed for for faults which act as a memory function, like stuck and sticky behavior in the sensor. In this case, it was shown that the introduction of dynamics by means of a setpoint change leads to benefits in terms of detection and diagnosis performance. It is based on these observations, that ActiveFDI is considered an interesting area for future research. The same KF strategy as well as an equivalent strategy based on the Extended Kalman Filter (EKF) were applied to the fully non-linear version of the system. In this case, fault detection results turned out much worse when using the KF. Using the EKF was shown to effectively tackle this problem, as the theoretically optimal performance for the linear system was recovered. For fault diagnosis, this was not the case. While it helps to use the EKF-strategy, the fault identification results were not as good as for the theoretically optimal linear case. In our paper, we also investigated the effect of considering the time of fault occurrence as a parameter of the fault models. This leads to increased computational demand but was shown to be more fruitful than assuming the fault detection time as the time of fault occurrence. Following this observation, strategies to decrease the computational burden associated with this time parameter have been proposed for further investigation.

Acknowledgment

The authors wish to thank the ICIS Distinctive Signature at Idaho National Laboratory (INL) for the support of this work.

Appendix A. Nonlinear tank model

Writing the mass balance over the tank system results in an Ordinary Differential Equation (ODE) which relates the tank level (h) with the inflow and outflow rates (q_{in}, q_{out}) as follows:

$$\frac{\mathrm{d}h(t)}{dt} = \frac{1}{A_{tank}} \cdot q_{in}(t) - \frac{1}{A_{tank}} \cdot q_{out}(t) \tag{6}$$

In addition, an algebraic equation follows from the pressure balance, equating the hydrostatic pressure at the bottom of the tank to the pressure losses over the valve and tube. This equation reads as follows:

$$\begin{aligned}
\rho \cdot g \cdot h(t) &= 2 \cdot \frac{f \cdot \rho \cdot L_{tube}}{\pi^2 \cdot D_{tube}^5} \cdot q_{out}(t)^2 + \frac{\rho \cdot g}{C_v(v(t))^2} \cdot q_{out}(t)^2 \\
&= \alpha \cdot q_{out}(t)^2 + \beta \frac{1}{C_v(v(t))^2} \cdot q_{out}(t)^2 \\
\alpha &= 2 \cdot \frac{f \cdot \rho \cdot L_{tube}}{\pi^2 \cdot D_{tube}^5} \\
\beta &= \rho \cdot g \\
C_v(v(t)) &= \frac{C_{v,max}}{\tau} \cdot e^{-v(t) \cdot ln(1/\tau)}
\end{aligned} \tag{7}$$

We assume that the friction factor, f, is constant, corresponding to a turbulent flow regime. As a result, the algebraic equation can be solved analytically for q_{out} as a function of tank level, h, and valve position, v:

$$q_{out}(t) = \sqrt{\frac{\rho \cdot g \cdot h(t)}{\alpha + \beta \cdot C_v(v(t))^{-2}}} \tag{8}$$

This equation can be plugged into Eq. 6 so that the nonlinear system can be written as a single ODE:

$$\frac{\mathrm{d}h(t)}{dt} = \frac{1}{A_{tank}} \cdot q_{in}(t) - \frac{1}{A_{tank}} \cdot \sqrt{\frac{\rho \cdot g \cdot h(t)}{\alpha + \beta \cdot C_v(v(t))^{-2}}} \tag{9}$$

It is this equation which is solved for the nonlinear tank simulations.

Appendix B. Nonlinear model parameters

Symbol	Description	Value	Unit
A_{tank}	Tank cross-sectional area	5	m^2
A_{tube}	Tube cross-sectional area	0.0491	m^2
D_{tube}	Tube diameter	0.1	m
L_{tube}	Tube length	5	m
g	Gravitational constant	9.81	$m\,s^{-2}$
$C_{v,max}$	Valve coefficient	20	$m^4\,N^{-1/2}$
τ_v	Valve rangeability	50	–
f	Friction coefficient	0.0125	–
P	Controller gain	0.3	%/m
τ_I	Controller integral time constant	5	s
h_o	Nominal tank level	10	m
v_o	Nominal valve opening	50	%
$q_{in,o}$	Nominal inflow rate	5.36	m^3/s
σ_1	Standard deviation tank level measurement	0.1	m
σ_2	Standard deviation valve position	0.5	%
σ_3	Standard deviation inflow rate	0.0536	m^3/s

Appendix C. Linearized tank model

The EKF implementation to track the dynamic state variable of the nonlinear system makes use of the linearized version of the model developed in Appendix A. To this end, the model is linearized at each time step around the state estimate at that time step. The same linearized model is also used to simulate the linearized version of the system as well as for implementation of the Kalman filter. In those two cases, the linearization is executed around nominal operating point. To linearize the model, first rewrite the model as follows:

$$\begin{aligned}\frac{dh}{dt} &= \frac{1}{A_{tank}} \cdot q_{in}(t) - \frac{1}{A_{tank}} \cdot \sqrt{\frac{\rho \cdot g \cdot h(t)}{\alpha + \beta \cdot C_v(v)^{-2}}} \\ &= \frac{1}{A_{tank}} \cdot q_{in}(t) - \frac{1}{A_{tank}} \cdot q_{out}(t)\end{aligned} \tag{10}$$

Deviation variables are introduced as follows:

$$\begin{aligned}h^*(t) &= h(t) - h_o \\ v^*(t) &= v(t) - v_o \\ q_{in}^*(t) &= q_{in}(t) - q_{in,o}\end{aligned} \tag{11}$$

The value for $q_{in,o}(t)$ is set equal to the the outflow, $q_{out,o}(t)$, computed from Eq. (8) with h_o and v_o given. This means that the operating point always presents the equilibrium point corresponding to h_o and v_o:

$$\begin{aligned}q_{in,o}(t) &= q_{out,o}(t) \\ &= \sqrt{\frac{\rho \cdot g \cdot h_o(t)}{\alpha(t) + \beta \cdot C_v(v_o(t))^{-2}}}\end{aligned} \tag{12}$$

The model is then linearized around the operating point as follows:

$$\begin{aligned}\frac{dh^*}{dt} &= \frac{1}{A_{tank}} \cdot q_{in}^*(t) \\ &\quad - \frac{1}{A_{tank}} \cdot \left(\left.\frac{\partial q_{out}}{\partial h}\right|_o \cdot h(t)^* + \left.\frac{\partial q_{out}}{\partial v}\right|_o \cdot v(t)^* \right) \\ \left.\frac{\partial q_{out}}{\partial h}\right|_o &= \frac{1}{2} \cdot \sqrt{\frac{\rho \cdot g}{\alpha + \beta \cdot C_v(v_o)^{-2}}} \cdot h_o^{-1/2} \\ \left.\frac{\partial q_{out}}{\partial v}\right|_o &= \sqrt{\rho \cdot g \cdot h_o} \cdot (\alpha + \beta \cdot C_v(v_o)^{-2})^{-3/2} \cdot \beta \cdot C_v(v_o)^{-3} \cdot \left.\frac{\partial C_v(v)}{\partial v}\right|_o \\ \left.\frac{\partial C_v(v)}{\partial v}\right|_o &= -\frac{C_{v,max}}{\tau} \cdot ln\left(\frac{1}{\tau}\right) \cdot e^{-v_o \cdot ln(1/\tau)}\end{aligned} \tag{13}$$

This model can now be rewritten as:

$$\begin{aligned}\frac{dh^*}{dt} &= \Phi \cdot h^*(t) + \Gamma_v \cdot v^*(t) + \Gamma_q \cdot q_{in}^*(t) \\ \Phi &= -\frac{1}{A_{tank}} \cdot \left.\frac{\partial q_{out}}{\partial h}\right|_o \\ \Gamma_v &= -\frac{1}{A_{tank}} \cdot \left.\frac{\partial q_{out}}{\partial v}\right|_o \\ \Gamma_q &= \frac{1}{A_{tank}}\end{aligned} \tag{14}$$

References

Akbaryan, F., & Bishnoi, P. R. (2001). Fault diagnosis of multivariate systems using pattern recognition and multisensor data analysis technique. *Computers & Chemical Engineering*, *25*, 1313–1339.

Bakshi, B. R., & Stephanopoulos, G. (1994). Representation of process trends. Part III. Multiscale extraction of trends from process data. *Computers & Chemical Engineering*, *18*, 267–302.

Charbonnier, S., Garcia-Beltan, C., Cadet, C., & Gentil, S. (2005). Trends extraction and analysis for complex system monitoring and decision support. *Engineering Applications of Artificial Intelligence*, *18*, 21–36.

Choudhury, S. M. A. A., Thornhill, N. F., & Shah, S. L. (2005). Modelling valve stiction. *Control Engineering Practice*, *13*, 641–658.

Dash, S., Maurya, M. R., & Venkatasubramanian, V. (2004). A novel interval-halving framework for automated identification of process trends. *AIChE Journal*, *50*, 149–162.

Flehmig, F., Watzdorf, R. V., & Marquardt, W. (1998). Identification of trends in process measurements using the wavelet transform. *Computers & Chemical Engineering*, *22*, 491–S496.

Forbus, K. D. (1984). Qualitative process theory. *Artificial Intelligence*, *24*, 85–168.

Isermann, R., & Ballé, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, *5*, 709–719.

Joliffe, I. T. (2002). *Principal component analysis* (2nd ed.). New York, USA: Springer.

Kuipers, B. (1994). *Qualitative reasoning: Modeling and simulation with incomplete knowledge*. Cambridge, MA, USA: MIT Press.

Maurya, M. R., Rengaswamy, R., & Venkatasubramanian, V. (2004). Application of signed digraphs-based analysis for fault diagnosis of chemical process flowsheets. *Engineering Applications of Artificial Intelligence*, *17*, 501–518.

Prakash, J., Narasimhan, S., & Patwardhan, S. C. (2005). Integrating model based fault diagnosis with model predictive control. *Industrial & Engineering Chemistry Research*, *44*, 4344–4360.

Prakash, J., Patwardhan, S. C., & Narasimhan, S. (2002). A supervisory approach to fault-tolerant control of linear multivariable systems. *Industrial & Engineering Chemistry Research*, *41*, 2270–2281.

Rieger, C. G., Gertman, D. I., & McQueen, M. A. (2009). Resilient control systems: Next generation design research. In *2nd IEEE conference on human system interaction*.

Srinivasan, R., & Rengaswamy, R. (2008). Approaches for efficient stiction compensation in process control valves. *Computers & Chemical Engineering*, *32*, 218–229.

Vanrolleghem, P. A., & Van Daele, M. (1994). Optimal experimental design for structure characterization of biodegradation models: On-line implementation in a respirographic biosensor. *Water Science and Technology*, *30*(4), 243–253.

Venkatasubramanian, V., Rengaswamy, R., & Kavuri, S. N. (2003). A review of process fault detection and diagnosis. Part I. Quantitative model-based methods. *Computers & Chemical Engineering*, *27*, 293–311.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., & Yin, K. (2003). A review of process fault detection and diagnosis. Part III. Process history based methods. *Computers & Chemical Engineering*, *27*, 327–346.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., & Kavuri, S. N. (2003). A review of process fault detection and diagnosis. Part II. Qualitative models and search strategies. *Computers & Chemical Engineering*, *27*, 313–326.

Villez, K. (2007). *Multivariate and qualitative data analysis for monitoring, diagnosis and control of sequencing batch reactors for wastewater treatment*. PhD Thesis, Ghent University.