Preprints of Control Conference Africa 2024
Balaclava, Mauritius, 16-17 September 2024

Particle filtering on hybrid dynamical systems for sensor fault detection

Simone D. Loubser*, Tobias M. Louw, Steven M. Bradshaw*****

Department of Chemical Engineering, University of Stellenbosch, Cape Town, South Africa (e-mail: 20964498@sun.ac.za).

*** Department of Chemical Engineering, University of Stellenbosch, Cape Town, South-Africa (e-mail:tmlouw@sun.ac.za)*

**** Department of Chemical Engineering, University of Stellenbosch, Cape Town, South-Africa (e-mail smb@sun.ac.za}*

Abstract: Sensor faults may be modelled as discrete random variables jointly distributed with process states and measurements, resulting in a hybrid (continuous and discrete) dynamical system. State estimation of such systems is challenging due to the combinatorial explosions following propagation of the discrete states. For this reason, a new algorithm is proposed which leverages the structure of the particle filter to incorporate discrete states representing sensor health. The discrete states characterize either normal or faulty conditions, with unique faults (e.g., stuck, out-of-range, etc.) as individual states. These become additional states to estimate which are subsequently used for fault detection and diagnosis. The method is evaluated using a simulated continuous stirred tank reactor benchmark and diagnoses sensor faults with an average sensitivity > 90% for two fault types and > 70% for three fault types.

Keywords: Fault detection and diagnosis, state estimation, particle filter, hybrid dynamical systems

1. INTRODUCTION

Fault detection and diagnosis (FDD) in the chemical process industries considers the timely detection of faulty conditions and the management of the root cause, aiming to reduce losses in production and profit during abnormal process conditions (Venkatasubramanian et al., 2003). FDD systems should assist operators in rapid and reliable decision making in the event of a fault (Zhang & Zhao, 2017). However, formulating suitable FDD systems is a challenging task, mainly due to the intricacy of the chemical processes which need to be modelled (Yang et al., 2019). Process data may be fed directly to a process monitoring solution, which may offer auto-diagnostic solutions (Wakefield, 2018).

Wakefield (2018), investigated FDD by contextualising the problem within the general framework of a hybrid Dynamic Bayesian Network (DBN) as a probabilistic model suitable to predict process behaviour based on different operating modes and inferring the most likely state based on the measurements received. In this paper we question the applicability of a DBN to a hybrid dynamical system, which is a system that contains continuous variables as well as discrete ones. In this method, and several others like it (Lerner, 2002), the combinatorial explosion of discrete states and transitions to track per timestep, leads to an unmanageble number of complex distributions over a short time period, negatively impacting the fault identification and diagnostic capability of the method. The combinatorial explosion in the number of transitions is also intractable. The problem is addressed by reducing or "collapsing" the number of transitions after each timestep, but this loses the integrity of the discrete combinations over time (Lerner et al., 2000; Wakefield, 2018).

State estimation is a well-established process monitoring method, commonly used in the processing industry and has been successfully applied for FDD. Particle filtering (PF), a form of state estimation, uses Monte Carlo sampling to approximate probablity distributions (Djuric et al., 2003), which is readily extended to a mixture of discrete and continous random variables. No combinatorial explosion occurs as the resampling inherent in the method ensures a fixed number of particles exist after each timestep, and that these particles are representative of the posterior estimate as well as all possible transitions. Typically, state estimation only considers continuous random variables for estimation, however, the presence of sensor faults is most readily modelled as discrete random variables with discrete states, i.e., a sensor may be healthy, failed, stuck, etc. Including discrete states in the state estimation problem requires a solution suitable for a hybrid-dynamical system consisting of both discrete and continuous variables (Li et al., 2022).

A handful of publications have investigated PFs as hybrid dynamical models for fault detection in industrial processes (Duan et al., 2014; Kong et al., 2021; Lerner et al., 2000; Wright & Horowitz, 2017). Kong et al. (2021) studied the use of a 'Salted Kalman Filter' for hybrid dynamical systems, but limited the number of discrete transitions to a single sensor at a time and did not consider the application of the method to FDD (Kong et al., 2021). Wright & Horowitz (2017) investigated a particle filter (PF) for sensor fault detection, but did not explicitly model the various faults: they aimed to

distinguish between normal and abnormal operating conditions only, thereby limiting the possibility of fault diagnosis. Duan et al. (2014) considered a PF for hybrid dynamical systems for mobile robots and proposed a suitable algorithm for state estimation and fault detection for hybrid systems.

This paper contributes to the current field of knowledge by investigating the FDD performance of a PF for a hybrid dynamical system, with explicit fault models that affect the observation of the process states, and evaluates the FDD performance across varying complexities of faults.

2. THEORY

2.1. State estimation

State estimation is typically applied in order to filter noise-corrupted measurements obtained from sensors, or to infer unmeasured state variables (Wright & Horowitz, 2017). State estimation is an iterative process whereby the state variables at any given timestep are first estimated using a process model and then updated based on the observed (measured) values. The conventional process is as follows (Simon, 2006):

1. The *a priori* state estimate x_t^- at timestep t is achieved by propagating the a *posteriori* estimate x_{t-1}^+ from the previous timestep $t-1$ using the process model.
2. The *a posteriori* estimate x_t^+ is obtained by updating the *a priori* estimate x_t^- using measurements y_t^* at t (Simon, 2006).

Classic state estimation techniques (e.g., Kalman filtering), make certain assumptions regarding the *a priori* and *a posteriori* distribution, which limit applications to hybrid dynamical systems. The PF, however, makes no assumptions regarding the underlying distributions, and offers a promising and robust method to handle estimation problems in hybrid dynamical systems (Duan et al., 2014; Thrun et al., 2005).

2.2. Particle filter

The conventional non-linear discrete time state-space equations are given in (1-2):

$$x_t = f_t(x_{t-1}, u_{t-1}) + w_{t-1} \tag{1}$$

$$y_t = g_t(x_t, u_t) + v_t \tag{2}$$

Where x_t, y_t, u_t, $f_t(\cdot)$ and $g_t(\cdot)$ are the state estimates, observations, inputs, state transition model and observation model respectively, all at timestep t. The process- (w_t) and measurement- noise (v_t) terms may assume any readily sampled distribution but are typically assumed normal with zero mean and covariance matrices Q_t and R_t respectively.

The PF begins by sampling *N* particles from the initial state prior probability distribution function (pdf), $p(x_0)$, assuming that it is known. The state of each particle is denoted by $x_{t,i}$ with $i = 1,2,3 \dots N$. At each timestep t, each particle $i = 1,2,3 \dots N$ is propagated from its *a posteriori* state estimate at the previous time step $x_{t-1,i}^+$, to the *a priori* estimate at the current time step $x_{t,i}^-$, using the state transition model $f_t(\cdot)$ as per (3); this function is typically derived from first principles.

$$x_{t,i}^- = f_t\left(x_{t-1,i}^+, u_{t-1}\right) + w_{t-1,i} \tag{3}$$

Each $w_{t-1,i}$ noise vector is sampled from the known pdf of w_{t-1}. Equation (3) amounts to sampling from the conditional distribution depending on the particle state vector at the previous time step, $x_{t,i}^- \sim p\left(x_t | x_{t-1,i}^+\right)$, but we typically abuse notation and refer to $p\left(x_t | x_{t-1,i}^+\right) = p(x_t)$ as the prior. The observation pdf $p\left(y_t | \hat{y}_{t,i}\right)$ is conditioned on the modelled observation $\hat{y}_{t,i}$, which is in turn obtained from the observation model g_t (4),

$$\hat{y}_{t,i} = g_t\left(x_{t,i}^-, u_t\right) \tag{4}$$

In conventional applications, the observation model lacks consideration for the state of the device which is transmitting this observation. The individual particle likelihood, $q_{t,i}$, is determined using (5), where the specific measurements obtained (y_t^*) are compared against the predicted observations ($\hat{y}_{t,i}$) for each particle.

$$\begin{aligned} q_{t,i} &= p\left(y_t = y_t^* | \hat{y}_{t,i}\right) \\ &\propto \exp\left(-\frac{1}{2}\left(y^* - \hat{y}_{t,i}\right)^T R_t^{-1}\left(y^* - \hat{y}_{t,i}\right)\right) \\ &= \tilde{q}_{t,i} \end{aligned} \tag{5}$$

Where we have defined the non-normalized likelihood $\tilde{q}_{t,i}$ which depends on the distribution of the measurement noise v_t according to (2), here assumed to be normal with zero mean and covariance R_t. Each resulting $\tilde{q}_{t,i}$ from (5) is normalized as per (6) to determine the relative likelihoods $q_{t,i}$ of each particle (Simon, 2006).

$$q_{t,i} = \frac{\tilde{q}_{t,i}}{\sum_i^N \tilde{q}_{t,i}} \tag{6}$$

Monte Carlo sampling is used to randomly generate a new set of particles with states $x_{t,i}^+$, based on the weight of the relative likelihoods $q_{t,i}$ (Simon, 2006). Resampling is classically done by sample importance resampling (SIR) but can be done with improved methods, such as low-variance resampling. Once this is complete, the collective pdf of the new set of particles $x_{t,i}^+$ will tend towards the posterior distribution $p(x_t | y_t^*)$ as the number of particles approach infinity (Simon, 2006).

3. METHODOLOGY

The main aim of this work is to determine the likelihood of an explicitly defined sensor fault, at any given timestep. This involves estimating the discrete state of the sensors and providing robust fault detection and diagnosis. The method should accommodate the discrete nature of faulty conditions, while accurately estimating the process state variables under both normal- and faulty conditions. The goals are achieved using a combined Particle Filter with Fault Detection and Diagnosis (PF-FDD) approach.

Noise can typically be separated into process noise (which is derived from the difference between a model and the true process) and measurement noise (which is the sensor or instrument noise attributed to a lack of precision). Typically,

measurement noise can be between 1-5% of the sensor range. Sensor faults are different to noise features, and directly affect the observation of process variables. In some cases, they can indirectly affect actual process variables through control loops. Sensors may experience fixed failures, drift, bias, out-of-range malfunctions or communication failures (Venkatasubramanian et al., 2003). A good FDD technique will detect sensor faults timeously and diagnose them with high sensitivity and specificity: sensitivity is the true positive rate (TPR), which is correctly identifying and diagnosing faults when they occur, while specificity is the true negative rate (TNR) which is correctly identifying normal conditions.

The proposed PF-FDD strategy amounts to sampling particles from an approximation of the joint posterior pdf $p(x_t, s_t|y_t^*)$, where $s_t \in \mathcal{S}$ is a vector containing the discrete sensor states contained in set $\mathcal{S}$ (e.g., $\mathcal{S} = \{\text{healthy, failed, biased, stuck}\}$). Approximating the joint posterior requires the definition of the likelihood function $p(y_t|x_t, s_t)$ as well the joint prior $p(x_t, s_t)$. The joint prior $p(x_t, s_t)$ is estimated using a state transition model (e.g., (3) for a classical PF). In line with similar work, we assume the state- and sensor state transitions to be independent as shown in Figure 1, such that the joint prior is given by $p(x_t, s_t) = p(x_t|x_{t-1}^+)p(s_t|s_{t-1}^+)$ and the state prior $x_{t,i}^- \sim p(x_t|x_{t-1}^+)$ is estimated by (3), excusing the usual abuse of notation.

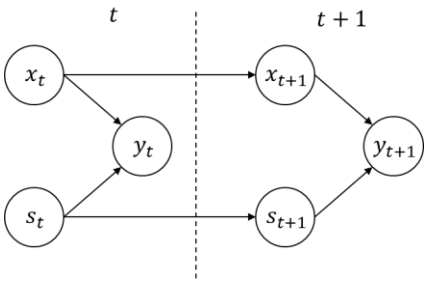

Figure 1: Causal network indicating independence of x_t and s_t

The sensor state transition model in this research is fully specified by the discrete conditional probability distribution $p(s_t|s_{t-1}^+)$. For the application presented, we assume $p(s_t|s_{t-1}^+)$ to be a stationary discrete distribution effectively summarized by a conditional probability table (CPT) with entries S_{kl} representing the probability of a sensor in state k transitioning to state l in the next time step, i.e., $S_{kl} = p(s_t = k|s_{t-1}^+ = l)$ with $k, l \in \mathcal{S}$. Similarly to the state prior $x_{t,i}^-$, the individual particle sensor state prior $s_{t,i}^-$ is sampled directly from this distribution, i.e., $s_{t,i}^- \sim p(s_t|s_{t-1}^+)$.

With the priors in hand, the observation model (4) is augmented with the sensor state to give the modelled observation $\hat{y}_{t,i}$ as per (7),

$$\hat{y}_{t,i} = g_t\left(x_{t,i}^-, s_{t,i}^-, u_t\right) \tag{7}$$

The zero-mean measurement noise covariance Σ is specific to the instrument and may also depend on the sensor state. This defines the likelihood function $p(y_t|x_t, s_t) = p\left(y_t|\hat{y}_t, \Sigma(s_t)\right)$.

The relative particle likelihood $q_{t,i}$ may then be determined using (5, 6) and resampling is performed to model the posterior $p(x_t, s_t|y_t^*)$, to normal. The process is summarized in the algorithm below.

Algorithm: PF-FDD

Set $t = 0$
Generate N particles:
for each particle $i = 1, \ldots, N$ **do**
  sample $x_{0,i}^+ \sim p(x_0)$
  sample $s_{0,i}^+ \sim p(s_0)$
for each timestep t **do**
  Prior estimates from process model:
  for each particle $i = 1, \ldots, N$ **do**
    sample $w_{t-1,i} \sim p(w_{t-1})$
    sample $s_{t,i}^- \sim p(s_t|s_{t-1,i}^+)$
    $x_{t,i}^- = f_t\left(x_{t-1,i}^+, u_{t-1}\right) + w_{t-1,i}$
    $\hat{y}_{t,i} = g_t\left(x_{t,i}^-, s_{t,i}^-, u_t\right)$
    $\tilde{q}_{t,i} = p\left(y_t = y_t^*|\hat{y}_{t,i}, \Sigma\left(s_{t,i}^-\right)\right)$

  Posterior estimates through resampling:
  $q_{t,i} = \tilde{q}_{t,i} / \sum_i^N \tilde{q}_{t,i}$ (normalize)
  for each particle $i = 1, \ldots, N$ **do**
    sample $r \sim$ uniform distribution over [0,1]
    find largest j where $\sum_{k=1}^{j-1} q_{t,k} < r$
    $x_{t,i}^+ \leftarrow x_{t,j}^-$
    $s_{t,i}^+ \leftarrow s_{t,j}^-$

4. CASE STUDY

4.1. Continuous stirred tank reactor model

A simulated non-linear and non-isothermal continuous stirred tank reactor (CSTR), with a cooling jacket to maintain temperature, is used as a case study. Consider the irreversible chemical reaction $A \rightarrow B$. For this system, the measurements of interest are the concentration of reagent A in the outlet (C_A) as well as the temperature of the outlet stream (T). These variables are observed during the simulation. The rate of reaction r is described by (8), with pre-exponential rate constant k_0 = 3.49 x 10^7 h^{-1} and activation energy E = 49.6 kJ/mol.

$$r = k_0 \exp\left(-\frac{E}{RT}\right) \tag{8}$$

The system can be modelled by two continuous-time, non-linear ordinary differential equations given by (9) and (10), where $V = 1\,\text{m}^3$ is the reactor volume, $(-\Delta H) = 2.49 \times 10^4$ kJ/mol is the heat of reaction, UA = 174.5 W/K is the overall heat transfer coefficient and area, $C_p\rho = 2.09 \times 10^3$ kJ/(K.m3) is the volumetric heat capacity of the mixture, and T_c = 300 K is the coolant temperature.

$$\frac{dC_A}{dt} = \left(\frac{F}{V}\right)\left(C_{Af} - C_A\right) - rC_A \tag{9}$$

$$\frac{dT}{dt} = \left(\frac{F}{V}\right)\left(T_f - T\right) + \frac{(-\Delta H)(rC_A)}{C_p\rho} - \frac{(UA)(T - T_c)}{VC_p\rho} \tag{10}$$

The flow rate (F) is assumed to be an error-free known input while C_{Af} and T_f are unmeasured disturbances. The differential equations outline the process model to form the state transition model, $f_t(x_{t-1})$. The continuous state variables and discrete sensor health states are summarized in vectors x and s, respectively (11) and the discrete sensor states may assume any value in set $\mathcal{S}$ (12),

$$x = \begin{bmatrix} C_A \\ T \end{bmatrix}, \qquad s = \begin{bmatrix} s_{C_A} \\ s_T \end{bmatrix} \tag{11}$$

$$s \in \mathcal{S} = \{\text{normal}, \text{failed}, \text{stuck}, \text{bias}\} \tag{12}$$

Fixed failures (stuck sensors), bias and malfunctions are considered in this case study. Stuck sensors are manifested as a sensor holding a certain measurement whilst the true process variable changes (Venkatasubramanian et al., 2003). A biased sensor is one which records a value that has a consistent deviation from its true value. An out-of-range malfunction occurs when the true process variable is outside the calibrated or instrument range of the sensor, and the transmission is beyond the range of the 4-20 mA signal. In some cases, the observed value is saturated at the lower or higher bound of the signal range when this occurs. Communication malfunctions can also be indicated by an out-of-range signal. These malfunctions will be noted as a 'failed' sensor as they both decouple the sensor from the true process dynamics. The sensor state prior probability mass function for both concentration and temperatures sensors in this case study, is given by (13),

$$\Pr(s_0) = \begin{Bmatrix} \Pr(s = \text{normal}) = 0.985 \\ \Pr(s = \text{stuck}) = 0.005 \\ \Pr(s = \text{bias}) = 0.005 \\ \Pr(s = \text{failed}) = 0.005 \end{Bmatrix} \tag{13}$$

The state prior pdf is given by (14),

$$p(x_0) \sim \mathcal{N}(x_0, P) \tag{14}$$

$$x_0 = \begin{bmatrix} 8.55\ \text{kmol/m}^3 \\ 320\ \text{K} \end{bmatrix}, \qquad P = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.7 \end{bmatrix}$$

Where x_0 is the initial state estimate, and P is the initial state estimation error covariance matrix (Simon, 2006).

4.2. Sensor transition model

The relevant sensors require a CPT to describe the probabilities of an individual sensor transitioning from one state to another. Both sensors (for concentration and temperature) are assumed to follow the same sensor transition model, see Table 1.

Table 1: Sensor conditional probability table detailing the probabilities of each transition between the discrete states.

	S_t			
S_{t-1}	Normal	Biased	Stuck	Failed
Normal	0.9994	0.0004	0.0004	0.0004
Biased	0.0002	0.9988	0.0004	0.0004
Stuck	0.0002	0.0004	0.9988	0.0004
Failed	0.0002	0.0004	0.0004	0.9988

4.3. Likelihood function

The likelihood function $p(y_t|x_t, s_t) = p\big(y_t|\hat{y}_t, \Sigma(s_t)\big)$ is necessary to diagnose the discrete state of the sensor. In this research, it defines a quantifiable description of the effects of various sensor faults on the observations expected. If a known sensor fault manifests, the likelihood function should describe the predicted observations. It does, however, rely on all diagnosable faults to be explicitly known and well-modelled. Here, we assume the observations are normally distributed with mean $\hat{y}(x_t, t, s_t)$ following the observation model $g(x_t, t, s_t)$, and sensor state dependent covariance $\Sigma(s_t)$. The likelihood function is given in (15), where we drop the subscript t for convenience.

$$p(y|x, s) = \mathcal{N}\big(\hat{y}(x, t, s), \Sigma(s)\big)$$

$$\hat{y}(x, t, s) = \begin{bmatrix} g_{C_A}(x, t, s_{C_A}) \\ g_T(x, t, s_T) \end{bmatrix}$$

$$\Sigma(s) = \begin{bmatrix} \sigma^2_{C_A}(s_{C_A}) & 0 \\ 0 & \sigma^2_T(s_T) \end{bmatrix}, \qquad s_{C_A}, s_T \in \mathcal{S} \tag{15}$$

The observation model for a healthy sensor ($s = \text{normal}$) is given by (16).

$$g_k(x, t, s_k = \text{normal}) = x_k(t), \qquad k = C_A, T \tag{16}$$

$$\sigma_{C_A}(s_{C_A} = \text{normal}) = 0.05\,\frac{\text{kmol}}{\text{m}^3}$$

$$\sigma_T(s_T = \text{normal}) = 0.5\ \text{K}$$

Failed sensors are assumed to manifest as consistently observing the lower bound of the individual sensor range, without any significant noise (17).

$$g_{C_A}(x, t, s_{C_A} = \text{failed}) = 0\ \text{kmol/m}^3 \tag{17}$$

$$g_T(x, t, s_T = \text{failed}) = 273\ \text{K}$$

$$\sigma_{C_A}(s_{C_A} = \text{failed}) = 10^{-7}\,\frac{\text{kmol}}{\text{m}^3}$$

$$\sigma_T(s_T = \text{failed}) = 5 \times 10^{-5}\ \text{K}$$

The stuck sensors, which also exhibit a consistent value but with a more significant noise component, link the new observation to the previous measurement (18).

$$g_k(x, t, s_k = \text{stuck}) = y_k(t-1), \qquad k = C_A, T \tag{18}$$

$$\sigma_{C_A}(s_{C_A} = \text{stuck}) = 10^{-6}\,\frac{\text{kmol}}{\text{m}^3}$$

$$\sigma_T(s_T = \text{stuck}) = 5 \times 10^{-3}\ \text{K}$$

Biased sensors are denoted by the observation model in (19).

$$g_k(x, \text{t}, s_k = \text{biased}) = x_k(\text{t}) + \beta_k, \qquad k = C_A, T \tag{19}$$

$$\beta_{C_A} = 0.6\,\frac{\text{kmol}}{\text{m}^3}, \qquad \beta_T = 7\ K$$

$$\sigma_{C_A}(s_{C_A} = \text{biased}) = \sigma_{C_A}(s_{C_A} = \text{normal})$$

$$\sigma_T(s_T = \text{biased}) = \sigma_T(s_T = \text{normal})$$

The likelihood function, state and sensor transition model define the key components for the PF-FDD method.

5. RESULTS

Measurements were simulated every minute for 15 days (360 hours) for the variables C_A and T. The PF-FDD method ran over the entire simulation, tracking C_A and T as each were at risk of sensor failure. Various sensor faults were simulated throughout this period, for 12 hours each day. To evaluate the ability of the PF-FDD method against a varying number of faults and degree of complexity, single-fault occurrences were initially used (where only one variable of the pair experienced a sensor fault, at that time) and from 190 hours onward, multi-fault occurrences were simulated (where both the concentration and temperature sensors experienced a fault). The *a posteriori* estimates of the states and sensor states were recorded at each timestep. Additionally, the PF-FDD method was evaluated with only two fault models ('PF-FDD-2F', for stuck and failed sensors) and then for three fault models ('PF-FDD-3F', for stuck, biased and failed sensors) on the same faulty data. The resulting concentration and temperature state estimates are provided in Figure 2 and the FDD performance results are given in Table 2 and 3. The inferred sensor state at each timestep was simply the sensor state with the most particles in that current state. This comparison offers insight into the performance of the PF-FDD method to an increased number of modelled sensor faults. The absolute percentage error is shown at the bottom in Figure 2, which is plotted in relation to a fixed 10% error dotted line. For the PF-FDD-2F simulation the total mean absolute percentage error (MAPE) was 6.6% for concentration and 2.9% for temperature. In comparison, the PF-FDD-3F simulation had an MAPE of 2.6% for concentration and 1.6% for temperature. Comparing these simulations, the PF-FDD method shows that it can provide reasonably accurate state estimates, in the presence of disturbances and sensor faults. The regions of significant deviation from the ground truth were in cases where both sensors experienced faults (multi-fault occurrences). With more modelled faults to consider (in PF-FDD-3F), the estimate is more susceptible to deviations from misdiagnosed faults but does not falter. The FDD performance is outlined by the sensitivity and specificity results in Tables 2 and 3, matching the red-shaded regions in Figure 2.

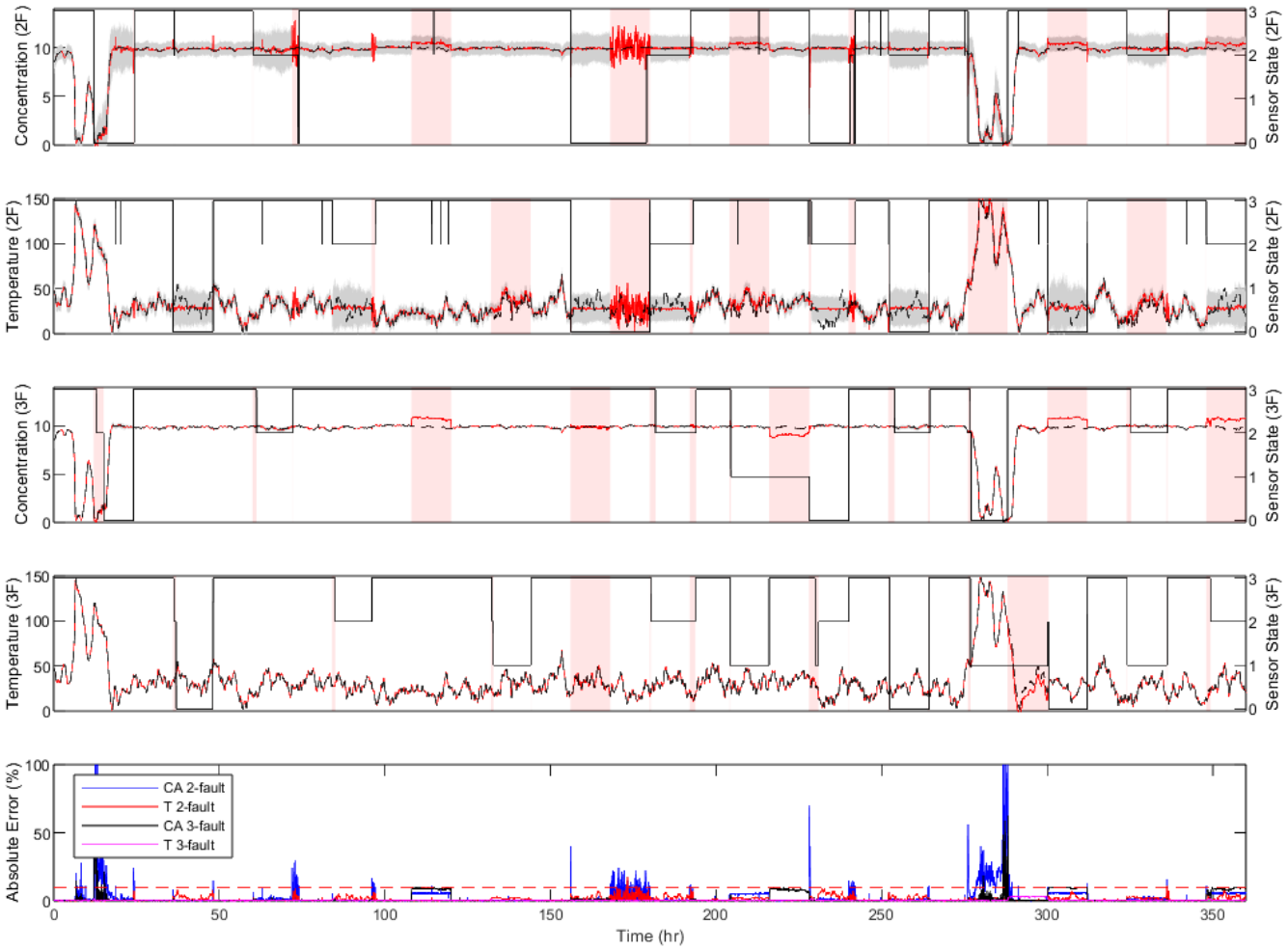

Figure 2: PF-FDD with two fault models (top two, indicated by '2F') and PF-FDD with three fault models (middle two, indicated by '3F') as well as the absolute percentage error between the ground truth and state estimates (bottom). Concentration [kmol/m3] and temperature [°C] profiles are shown for each of the 360-hour simulations. The (top four plots') black lines indicate the ground truth while the grey shaded area shows the region between the 5th and 95th percentile of the PF-FDD state estimates, and the red line is the state estimate. Red shaded regions indicate an incorrectly identified period (normal or faulty) or misdiagnosed fault. In contrast, the white regions showcase the correct identification and diagnosis for the sensor states. Sensor states are denoted as {normal=3; stuck=2; biased=1; failed=0}.

Table 2: Sensitivity and specificity results of the PF-FDD-2F method, over the course of the 360-hour simulation.

		Sensitivity	
Sensor	Specificity	Stuck	Failed
Concentration	92.4 %	82.9 %	98.9 %
Temperature	91.2 %	82.4 %	99.4 %

Table 3: Sensitivity and specificity results of the PF-FDD-3F method, over the course of the 360-hour simulation.

		Sensitivity		
Sensor	Specificity	Stuck	Biased	Failed
Concentration	93.1 %	87.3 %	24.1 %	66.2%
Temperature	93.2 %	88.5 %	96.1 %	71.4 %

Specificity for both simulations, is good: > 90% for each sensor. Sensitivity, however, worsens when going from two to three fault types to detect and diagnose using the PF-FDD method. The method PF-FDD-2F has an exceptionally high (>90%) average sensitivity for the modelled faults. Whereas the PF-FDD-3F had an average sensitivity of only 72%, and albeit containing models of all the true faults imposed on the system, it struggled to diagnose biased, and some failed, sensor faults. In Figure 2, the plots of the 3F method still have a number of misdiagnosed (red-shaded) regions but have much tighter confidence intervals (grey regions) to inform the state estimate. This is likely because the bias fault model remains coupled with the process dynamics and is therefore often mistaken as a normal sensor. Almost none of the biased concentration sensors were detected. This is likely due to the bias constant not being distinguishable from the measurement noise, which could be remedied by decreasing the expected instrument measurement noise, making the bias addition more distinguishable. The stuck and failed sensors are detected with much more ease, and diagnosed with high sensitivity in both simulations. The PF-FDD method is highly dependent on the fault-type and fault models used, as well as sufficiently filtering out the noise features. From Figure 2, it also appears that both methods struggle to correctly identify and diagnose faults when multi-fault occurrences are experienced. The absolute error (%) increases during these periods. It is clear, still, that with increased complexity in the fault models, or number of known and modelled faults, the proposed PF-FDD method can produce highly sensitive FDD results, with accurate state estimates.

6. CONCLUSIONS

A new method was proposed for the state estimation and FDD of hybrid dynamical systems without the challenge of managing the combinatorial explosion of tracked transitions, and ultimately provided sensor health monitoring in the form of fault detection. The proposed Particle Filter with Fault Detection and Diagnosis (PF-FDD) method was able to detect and diagnose sensor faults with an overall fault sensitivity of 61%; namely for the fixed failure, biased and out-of-range faults. Future investigations may consider the convergence guarantees of the proposed method.

REFERENCES

Djuric, P. M., Kotecha, J. H., Zhang, J., Huang, Y., Ghirmai, T., Bugallo, M. F., & Miguez, J. (2003). Particle filtering. *IEEE Signal Processing Magazine*, *20*(5). https://doi.org/10.1109/MSP.2003.1236770

Duan, Z., Cai, Z., & Min, H. (2014). Robust dead reckoning system for mobile robots based on particle filter and raw range scan. *Sensors (Switzerland)*, *14*(9), 16532–16562. https://doi.org/10.3390/s140916532

Kong, N. J., Payne, J. J., Council, G., & Johnson, A. M. (2021). The Salted Kalman Filter: Kalman filtering on hybrid dynamical systems ☆. *Automatica*, *131*, 109752. https://doi.org/10.1016/j.automatica.2021.109752

Lerner, U. (2002). *Hybrid Bayesian Networks for Reasoning* (Issue October). Stanford University.

Lerner, U., Parr, R., Koller, D., & Biswas, G. (2000). Bayesian Fault Detection and Diagnosis in Dynamic Systems. *Seventeenth National Conference on Artificial Intelligence (AAAI-00)*, *August*, 531–537.

Li, T., Sbarufatti, C., Cadini, F., Chen, J., & Yuan, S. (2022). Particle filter-based hybrid damage prognosis considering measurement bias. *Structural Control and Health Monitoring*, *29*(4), 1–18. https://doi.org/10.1002/stc.2914

Simon, D. (2006). *Optimal State Estimation: Kalman, H, and Nonlinear Approaches*. John Wiley & Sons, Ltd.

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics* (Vol. 1). MIT Press. https://doi.org/10.1145/504729.504754

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., & Yin, K. (2003). A review of process fault detection and diagnosis Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, *27*(3), 327–346. https://doi.org/10.1016/s0098-1354(02)00162-x

Wakefield, B. J. (2018). *Applying Dynamic Bayesian Networks to Process Monitoring*. Stellenbosch University.

Wright, M. A., & Horowitz, R. (2017). Particle-filter-enabled real-Time sensor fault detection without a model of faults. *2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017*, *2018-Janua*, 5757–5763. https://doi.org/10.1109/CDC.2017.8264529

Yang, X., Zhou, J., Xie, Z., & Ke, | Gong. (2019). *Chemical process fault diagnosis based on enchanted machine-learning approach*. https://doi.org/10.1002/cjce.23642

Zhang, Z., & Zhao, J. (2017). A deep belief network based fault diagnosis model for complex chemical processes. *Computers and Chemical Engineering*, *107*, 395–407. https://doi.org/10.1016/j.compchemeng.2017.02.041