State and Parameter Estimation of Multi-rate Processes with Variable Measurement Delays

by

Ruijing Han

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Process Control

Department of Chemical and Materials Engineering

University of Alberta

Abstract

Measurements in the process industry can arrive with fast or slow sampling rates. Fast measurements such as flowrate and temperature are sampled frequently and are obtained instantly after sampling. The slow measurements, which are usually related with chemical quality variables such as product composition, are sampled infrequently and have some delay due to the laboratory analysis. Moreover, sample collection for laboratory analysis may extend over a significant time interval, and the slow measurements are actually functions of all the states during the sampling period. Our objective is to develop a multirate state and parameter estimation method for this situation.

In state estimation, the objective is to fuse the two kinds of measurements to provide a more accurate estimation of the system's states. We propose two methods to solve the problem, the exact Bayesian approach and the augmented state approach. In the exact Bayesian approach, the algorithm is developed by implementing Bayes' rule. In the augmented state approach, the system is reformulated by augmenting the current state with past information required for fusing the slow measurements and then apply general state estimation procedures. For system identification, the parameters are estimated along with the time delays using a particle filter (PF) under the framework of the expectation maximization (EM) algorithm. The performance of the proposed methods for both state and parameter estimation are demonstrated though simulation and experimental studies and by comparison with methods that only use the fast measurements.

Finally, the proposed state and parameter estimation methods are applied to the FWKO vessel to demonstrate their effectiveness and applicability.

Acknowledgements

I would like to express my special thanks of gratitude to my supervisors Dr. Biao Huang and Dr. Vinay Prasad for their appreciable inspiration, support and patience throughout my graduation study. I would like to thank Dr. Biao Huang for providing me directions of my research. I would like to thank Dr. Vinay Prasad for giving me important suggestions on my research work. They are great supervisors and friends and it is a great pleasure for me to have the opportunity to work under their supervision. I would like to thank Lei Fan for giving me instructions on the experiment and help me understand the data in application.

Thanks to my friends for helping me when I meet troubles in daily life. Their kindness make me feel happy and grateful. Thanks to my family for supporting and encouraging me to pursue my MSc degree. Finally, a very special appreciation to my boy friend Weite Ni for his accompany and support in the last two years.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

In the process industry, measurements are used to monitor and control the system. Typically, measurements that arrive frequently and online are called fast measurements, while those are infrequent and usually have some delay are named as slow measurements. It is common to see both kinds of measurements in one process system. For example, in a distillation column, the tray temperatures are fast measurements and the top and bottom product concentrations are slow measurements that are sampled irregularly and needed to be sent to laboratory for analysis. The slow measurements or laboratory measurements have three characteristics: (1) they are more accurate than fast measurements; (2) they have delays due to the laboratory analysis; (3) they are actually functions of all the states during the sampling period.

Although systems that contain both fast and slow measurements are universal, there is little research related to the third characteristic mentioned above, and even that focuses on state estimation. In [20], Fatehi and Huang considered multirate and variable delayed measurements for a linear system. In another paper [21], they proposed a state estimation method based on the Kalman filter for a linear state space model which consists of fast and slow integrated measurements. Guo and Huang [30] proposed a variable dimension unscented Kalman filter (VD-UKF) to estimate the states with infrequent, delayed and integral measurements for a nonlinear system. Motivated by the importance of the multirate measurements problem and these researches, in this thesis, we present two different state estimation algorithms for nonlinear multirate measurements and a parameter algorithm under the same assumptions.

The following three sections in this chapter will briefly introduce the general state-space model formulation, nonlinear filters used in the second chapter, and the expectation maximization (EM) algorithm applied in the third chapter. In the final section, the thesis outline and contributions are illustrated.

1.2 Dynamic state space model

The nonlinear state space model is a general class of models to represent nonlinear dynamic systems. The general nonlinear and non-Gaussian state space model can be expressed as follows,

$$x_k = f\left(x_{k-1}, u_{k-1}, \omega_{k-1}\right) \tag{1.1}$$

$$y_k = h\left(x_k, v_k\right) \tag{1.2}$$

where $x_k \in \mathbb{R}^{n_x}$ is the system state vector; $u_k \in \mathbb{R}^{n_u}$ is the input vector; $y_k \in \mathbb{R}^{n_y}$ is the measurement; f and h are nonlinear functions; $\omega_k \in \mathbb{R}^{n_x}$ and $v_k \in \mathbb{R}^{n_y}$ are process noise and measurement noise, respectively. It can also be represented as a state transition probability density $p\left(x_k \mid x_{k-1}, u_{k-1}\right)$ and a measurement probability density $p\left(y_k \mid x_k\right)$. It can be seen that the states follow a first order Markov process and the measurements are assumed to be independent given the states [40].

In this thesis, we will assume that the process and measurement noises follow Gaussian distributions with zero means. And the state space model is specified by moving the noise term out from the nonlinear function as an addend,

$$x_k = f\left(x_{k-1}, u_{k-1}\right) + \omega_{k-1} \tag{1.3}$$

$$y_k = h\left(x_k\right) + v_k \tag{1.4}$$

Then the state transition and measurement probability densities are

$$p\left(x_k \mid x_{k-1}, u_{k-1}\right) = \mathcal{N}\left(f\left(x_{k-1}, u_{k-1}\right), Q\right) \tag{1.5}$$

$$p\left(y_k \mid x_k\right) = \mathcal{N}\left(h\left(x_k\right), R\right) \tag{1.6}$$

where Q and R are covariances of ω_k and v_k, respectively.

1.3 State estimation

In state estimation, the goal is to compute the posterior density $p\left(x_k \mid y_{1:k}\right)$, where $y_{1:k} = \{y_1, y_2, \cdots, y_k\}$. If we know the posterior density, we can easily derive various estimates of

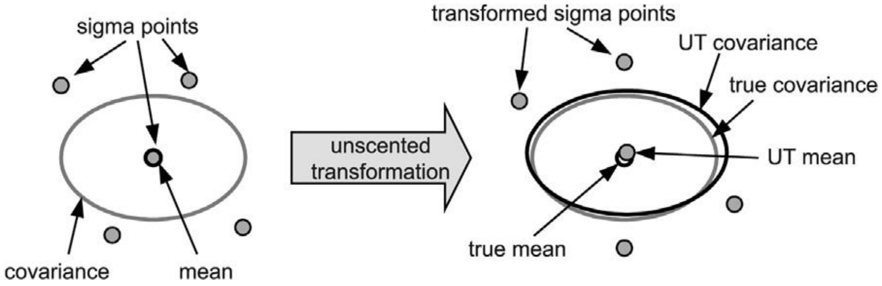

Figure 1.1: Schematic of the unscented transformation [46]

the system's states including means, modes, medians and confidence intervals [47]. Bayesian methods provide a rigorous general framework for state estimation problems [29]. In Chapter 2, we will develop a state estimation algorithm named the exact Bayesian method by applying Bayesian rules. For the general state space model in Section 1.2, the optimal solution is called recursive Bayesian estimation, also known as a Bayes filter. The Bayes filter contains two stages: prediction and update. The prediction stage applies the Chapman–Kolmogorov equation [1]:

$$p\left(x_k \mid y_{1:k-1}\right) = \int p\left(x_k \mid x_{k-1}, u_{k-1}\right) p\left(x_{k-1} \mid y_{1:k-1}\right) dx_{k-1} \tag{1.7}$$

Then, the Bayesian rule is used to provide the basis for the update stage,

$$p\left(x_k \mid y_{1:k}\right) = \frac{p\left(y_k \mid x_k\right) p\left(x_k \mid y_{1:k-1}\right)}{p\left(y_k \mid y_{1:k-1}\right)} \tag{1.8}$$

where $p\left(y_k \mid y_{1:k-1}\right)$ is the normalizing constant, and is obtained as,

$$p\left(y_k \mid y_{1:k-1}\right) = \int p\left(y_k \mid x_k\right) p\left(x_k \mid y_{1:k-1}\right) dx_k \tag{1.9}$$

When the system model is linear with white Gaussian noise, the Bayes filter becomes equivalent to the Kalman filter [45]. Next, we introduce two nonlinear filters, the unscented Kalman filter (UKF) and the particle filter (PF), which are built based on the Bayes filter framework.

1.3.1 Unscented Kalman filter

The unscented Kalman filter (UKF) provides a Gaussian approximation to the posterior density $p\left(x_k \mid y_{1:k}\right)$. In the UKF, the state distribution is represented by a set of sigma points selected using a deterministic sampling approach called the unscented transform.

These sigma points completely capture the true mean and covariance of the state for the Gaussian case, and when propagated through the true nonlinear functions, they can capture the posterior mean and covariance accurately to the second order for any nonlinearity, with errors only introduced in the third and higher orders [47]. Another advantage of the UKF is that it removes the requirement to explicitly calculate Jacobians or partial derivatives [34], [48]. Figure 1.1 shows the schematic of the unscented transformation used in the UKF.

To formulate the UKF, assume the density $p\left(x_k \mid y_{1:k}\right)$ is approximated as a Gaussian distribution with a mean μ_k^x and a covariance Σ_k^x. Then, the UKF is recursive and formulated in the prediction and update steps as in [48].

1. The prediction step

Calculate the sigma points as follows:

$$\mathcal{X}_k^0 = \mu_k^x \tag{1.10}$$

$$\mathcal{X}_k^i = \mu_k^x + \left(\sqrt{\left(n_x + \kappa\right) \Sigma_k^x}\right)_i \qquad i = 1, \cdots, n_x \tag{1.11}$$

$$\mathcal{X}_k^i = \mu_k^x - \left(\sqrt{\left(n_x + \kappa\right) \Sigma_k^x}\right)_i \qquad i = n_x + 1, \cdots, 2n_x \tag{1.12}$$

where κ is a scaling parameter and $\left(\sqrt{\left(n_x + \kappa\right) \Sigma_k^x}\right)_i$ is the ith row or column of the matrix square root of $\left(n_x + \kappa\right) \Sigma_k^x$. The first order weights for sigma points are determined as follows:

$$W_0^{(m)} = \kappa / \left(n_x + \kappa\right) \tag{1.13}$$

$$W_i^{(m)} = \kappa / 2\left(n_x + \kappa\right) \qquad i = 1, \cdots, 2n_x \tag{1.14}$$

The second order weights are

$$W_0^{(c)} = \kappa / \left(n_x + \kappa\right) + \left(1 - \alpha^2 + \beta\right) \tag{1.15}$$

$$W_i^{(c)} = W_i^{(m)} \qquad i = 1, \cdots, 2n_x \tag{1.16}$$

where α controls the spread of the sigma points and β is related to the distribution of the state. Note that the summation of the weights for sigma points is equal to 1. The state mean and covariance are predicted by propagating sigma points as follows:

$$\mathcal{X}_{k+1|k}^{i} = f\left(\mathcal{X}_{k}^{i}, u_{k-1}\right) \tag{1.17}$$

$$\mu_{k+1|k}^{x} = \sum_{i=0}^{2n_x} W_i^{(m)} \mathcal{X}_{k+1|k} \tag{1.18}$$

$$\Sigma_{k+1|k}^{x} = \sum_{i=0}^{2n_x} W_i^{(c)} \left(\mathcal{X}_{k+1|k}^{i} - \mu_{k+1|k}^{x}\right)\left(\mathcal{X}_{k+1|k}^{i} - \mu_{k+1|k}^{x}\right)^T + Q \tag{1.19}$$

The measurement prediction is

$$\mathcal{Y}_{k+1|k}^{i} = h\left(\mathcal{X}_{k+1|k}^{i}\right) \tag{1.20}$$

$$\mu_{k+1|k}^{y} = \sum_{i=0}^{2n_x} W_i^{(m)} \mathcal{Y}_{k+1|k} \tag{1.21}$$

2. The update step

Calculate the UKF gain as follows:

$$\Sigma_{k+1|k}^{y} = \sum_{i=0}^{2n_x} W_i^{(c)} \left(\mathcal{Y}_{k+1|k}^{i} - \mu_{k+1|k}^{y}\right)\left(\mathcal{Y}_{k+1|k}^{i} - \mu_{k+1|k}^{y}\right)^T + R \tag{1.22}$$

$$\Sigma_{k+1|k}^{xy} = \sum_{i=0}^{2n_x} W_i^{(c)} \left(\mathcal{X}_{k+1|k}^{i} - \mu_{k+1|k}^{x}\right)\left(\mathcal{Y}_{k+1|k}^{i} - \mu_{k+1|k}^{y}\right)^T \tag{1.23}$$

$$K_{k+1} = \Sigma_{k+1|k}^{xy}\left(\Sigma_{k+1|k}^{y}\right)^{-1} \tag{1.24}$$

Finally, the posterior state mean and covariance are obtained as

$$\mu_{k+1}^{x} = \mu_{k+1|k}^{x} + K_{k+1}\left(y_{k+1} - \mu_{k+1|k}^{y}\right) \tag{1.25}$$

$$\Sigma_{k+1}^{x} = \Sigma_{k+1|k}^{x} - K_{k+1}\Sigma_{k+1|k}^{y}K_{k+1}^{T} \tag{1.26}$$

More details about the UKF are available in [33].

1.3.2 Particle filter

Unlike most other nonlinear Bayes filters, the particle filter (PF) does not rely on linearization or a Gaussian assumption [40]. It is a sequential Monte Carlo method that approximates the posterior density with a set of weighted particles [4], [17], [29], [31], in a discrete summation form:

$$p\left(x_k \mid y_{1:k}\right) \approx \sum_{i=1}^{N} w_k^i \delta\left(x_k - x_k^i\right) \tag{1.27}$$

where x_k^i represents the i^{th} particle, w_k^i is the associated weight, $\delta(\cdot)$ is the Dirac delta function and N is the number of particles.

The ideal case for Monte Carlo sampling is to generate particles directly from the true posterior density $p(x_k \mid y_{1:k})$, which is unavailable [40]. So a technique called importance sampling is utilized to sample particles in PF. The importance sampling density, $q(x_k \mid x_{1:k-1}, y_{1:k})$, is easy to be sample from and approximates the true posterior density. The PF has been shown to be asymptotically unbiased if the support region for $p(x_k \mid y_{1:k})$ is a subset of the region of $q(x_k \mid x_{1:k-1}, y_{1:k})$ [16]. After sampling the particles, the weights are given by

$$w_k^i \propto w_{k-1}^i \frac{p\left(y_k \mid x_k^i\right) p\left(x_k^i \mid x_{k-1}^i\right)}{q\left(x_k^i \mid x_{1:k-1}^i, y_{1:k}\right)} \tag{1.28}$$

For derivation of this equation, readers are referred to [4].

In [36], it is shown that the optimal importance sampling density is

$$q(x_k \mid x_{1:k-1}, y_{1:k}) = p(x_k \mid x_{1:k-1}, y_{1:k}) \tag{1.29}$$

But the optimal importance sampling density is generally unknown except for two cases. The first case is when the state space consists of a finite number of states [18]. The second case is when the state transition function is nonlinear and the measurements are linear [11], [16]. Therefore, in this thesis, we will choose the most common used importance density, i.e. the state transition density $p(x_k \mid x_{k-1})$, since the state space model is known. Then, equation (1.28) reduces to

$$w_k^i \propto w_{k-1}^i p\left(y_k \mid x_k^i\right) \tag{1.30}$$

A general problem in the PF is the degeneracy phenomenon [16]. In order to avoid this problem, a common solution is to apply a resampling step after the weight update. Several resampling strategies are introduced and compared in [15]. In this thesis, we will use the systematic resampling method.

The steps of PF for the state estimation are summarized as follows:

1. *Initialization*: Generate $\{x_0^i\}_{i=1}^N$ from the initial state density $p(x_0)$, set $w_0^i = 1/N$ for $i = 1, \cdots, N$, and $k = 1$.

2. *Prediction*: Predict $\left\{x_{k|k-1}^i\right\}_{i=1}^N$ according to $x_{k|k-1}^i \sim p\left(x_k \mid x_{k-1}^i\right)$ for $i = 1, \cdots, N$.

3. *Update*: Compute the weights as $w_{k|k-1}^i = \dfrac{w_{k-1}^i p\left(y_k \mid x_{k|k-1}^i\right)}{\sum_{j=1}^N w_{k-1}^j p\left(y_k \mid x_{k|k-1}^j\right)}$ for $i = 1, \cdots, N$.

4. *Resampling*: Generate posterior particles $\{x_k^i\}_{i=1}^N$ from particle set $\left\{x_{k|k-1}^i, w_{k|k-1}^i\right\}_{i=1}^N$ according to the resampling strategy, and set $w_k^i = 1/N$ for $i = 1, \cdots, N$.

5. *Output*: Estimate the state as $\hat{x}_k = \sum_{i=1}^N w_k^i x_k^i$, set $k := k + 1$ and go back to step 2.

1.4 Parameter estimation

To conduct the state estimation, we always need an accurate state space model to represent the system. This motivates the parameter estimation problem of the multirate measurements system in Chapter 3. For the parameter estimation of a nonlinear state space model, there are two commonly used solutions. The first method is to combine the parameters that need to be estimated with the system state and to apply filtering approaches to the augmented state [9], [38]. However, the estimated parameters using this method have large covariances [3]. Another method is the maximum likelihood (ML) algorithm to find the parameters that maximize the joint probability of measurements [27], [39]. In this section, we will focus on the expectation maximization (EM) algorithm, which is a well known ML-based method.

1.4.1 EM algorithm

The EM algorithm [12] is recursive and consists of two steps, the expectation step and the maximization step. In the first step, the latent or hidden data Z are estimated given the observed data Y_{obs} and the current estimate of the model parameters. In the second step, the parameters are updated by maximizing the likelihood function (called the Q function) under the assumption that the hidden data are known [7]. Convergence of the algorithm is assured as the Q function is guaranteed to increase at each iteration.

The following is the formulation of the EM algorithm. In the expectation step, the Q function is defined as

$$\begin{aligned} Q\left(\theta \mid \theta_n\right) &= E_{Z|Y_{obs},\theta_n}\left\{\log\left[p\left(Z, Y_{obs} \mid \theta_n\right)\right]\right\} \\ &= \int \log\left[p\left(Z, Y_{obs} \mid \theta_n\right)\right] p\left(Z \mid Y_{obs}, \theta_n\right) dZ \end{aligned} \tag{1.31}$$

where θ_n is the parameter estimate after n iterations. The maximization step maximizes the Q function with respect to θ:

$$\theta_{n+1} = \arg\max_{\theta} Q\left(\theta \mid \theta_n\right) \tag{1.32}$$

By implementing the expectation and maximization steps recursively, a good enough estimates for the parameter θ can be obtained.

1.5 Thesis outline and contributions

The rest of this thesis is organized as follows:

In Chapter 2, two algorithms are proposed to solve the state estimation problem for multirate system with variable measurement delays. The first algorithm is named as the exact Bayesian approach, which is developed by implementing Bayes' rule. The other algorithm is called the augmented state approach. In this approach, the system is reformulated by augmenting the current state with past information required for fusing the slow delayed laboratory measurements. The performance of the proposed methods are demonstrated through simulation and experimental studies by implementing the particle filter and the UKF.

In Chapter 3, a parameter estimation method is presented for the same system. We estimate the parameters along with the time delays using a particle filter under the framework of the EM algorithm. In the evaluation section, the proposed method is compared with the algorithm that only uses the fast measurements.

In Chapter 4, the proposed state and parameter estimation methods are applied to a FWKO vessel. The implementation procedures and results are discussed in this chapter.

Chapter 5 summarizes the main results of this thesis and provides some perspectives for future research.

A note on notation

In the following chapters, three dots between delimiters (parenthesis, brackets, or braces) means that the quantity between the delimiters is the same as the quantity between the previous set of identical delimiters in the same equation. For example,

$$(A + BCD) + (\cdots)^T = (A + BCD) + (A + BCD)^T \tag{1.33a}$$

$$(A + BCD)^T E^{-1} (\cdots) = (A + BCD)^T E^{-1} (A + BCD) \tag{1.33b}$$

Chapter 2

State estimation for multirate measurements in the presence of integral term and variable measurement delay

2.1 Introduction

In the process industry, laboratory analysis is commonly used to provide accurate measurements for chemical quality variables, such as composition of chemicals and so on [20]. These variables are usually not available for fast online measurements, or they are measured on line with large errors. To obtain more accurate estimates for these variables, laboratory measurements can be used to improve the performance of fast online state estimation. However, in many cases, laboratory samples are collected gradually in a specific container for a period of time [21], and the collected time and the sampling time can be irregular. Moreover, the result from laboratory analysis, which can be seen as a measurement of the integrated state, may be delayed variably. All these factors make the fusion of fast online measurements and slow delayed laboratory measurements difficult.

There are many methods to incorporate delayed and infrequent measurements as introduced in [26]. However, there are very few theoretical studies on this subject. Guo and Huang [30] proposed a variable dimension unscented Kalman filter (VD-UKF) to estimate the states with infrequent, delayed and integral measurements. They constructed a variable dimension augmented state space model consisting of the original states and the integral states. By applying UKF on this augmented model, the states can be estimated just as fusing fast measurements and slow delayed integral measurements. This approach is an extension of the sample-state augmentation method which is usually applied to the delayed measurement problem. The VD-UKF proposed by Guo and Huang is useful but difficult to implement. On the other hand, Fatehi and Huang [20], [21] proposed some simpler methods, but they treated this problem separately and only addressed the linear case. In [20], they considered multirate and variable delayed measurements, and proposed a fusion method named the modified delayed track to track fusion (MDTTF) as the extension of MTTF method from [23]. They applied MDTTF on a linear process to fuse the fast rate (delay-free but less accurate) measurements with the slow rate (delayed but more accurate) measurements to improve the state estimation through Kalman filter. In another paper [21], they considered the delay-free integrated measurement problem assuming the integrated time was fixed, and proposed a slow-rate integrated measurement Kalman filter (IMKF) to estimate the states frequently when there were only integrated measurements. In addition, they utilized the general optimal estimation fusion [6] to fuse the estimates of the same state from IMKF and fast measurement Kalman filter (FMKF).

Because most industrial processes are nonlinear and linearizing the system can lead to large errors, we focus on nonlinear systems. The most commonly used algorithms to solve the problem of non-Gaussian, nonlinear state estimation are the extended Kalman filter (EKF), unscented Kalman filter (UKF) and particle filter (PF). The EKF is based upon the principle of linearizing the measurements and evolution models using Taylor series expansions, which may lead to poor representations of the nonlinear functions and probability distributions of interest [47]. The UKF, which uses the unscented transformation, is an extension of the Kalman filter that reduces the linearization errors of the EKF. The UKF can provide significant improvement over the EKF [43]. The PF, which is also called sequential Monte Carlo method, is a completely nonlinear state estimator. The main idea behind the PF is importance sampling. It uses a set of weighted particles to represent the posterior density.

In this chapter, we present two methods: the first is called the exact Bayesian approach, which is an extension method from [52], and the second is the augmented state method. These two methods are implemented on both the PF and UKF. Besides, the proposed methods are compared through simulation and experimental case studies. The remainder of this chapter is organized as follows. The problem statement is presented in Section 2.2. In Sections 2.3 and 2.4, the exact Bayesian approach and augmented state method are introduced, respectively, with their implementations of PF and UKF in each section. In Section 2.5, the proposed algorithms are tested by simulations and experimental studies. Section 2.6 is the conclusion of this chapter.

2.2 Problem statement

Consider the following discrete time system:

$$x_k = f\left(x_{k-1}, u_{k-1}\right) + \omega_{k-1} \tag{2.1a}$$

$$y_k^f = h^f\left(x_k\right) + v_k^f \tag{2.1b}$$

$$y_{k_s(j)}^m = h^m\left(m_{s_j+l_j}\right) + v_{k_s(j)}^m \tag{2.1c}$$

where $x_k \in \mathbb{R}^{n_x}$ is the state to be estimated, and the initial state x_0 follows the distribution $p\left(x_0\right)$; $u_k \in \mathbb{R}^{n_u}$ is the input vector which is known (for the sake of simplicity in presentation, it is omitted in the following derivation); $y_k^f \in \mathbb{R}^{n_y^f}$ is the fast and regular measurement and $y_{k_s(j)}^m \in \mathbb{R}^{n_y^m}$ ($j \in \mathbb{Z}_+$) is the slow and irregular laboratory measurement of a integral term; and f, h^f and h^m are nonlinear functions. The noise terms $\omega_k \in \mathbb{R}^{n_x}$, $v_k^f \in \mathbb{R}^{n_y^f}$ and $v_{k_s(j)}^m \in \mathbb{R}^{n_y^m}$ are *i.i.d.* Gaussian with zero mean and covariance matrices Q, R^f and R^m, respectively. $m_{s_j+l_j} \in \mathbb{R}^{n_m}$ is the integral term which represents the sample collected from time instant s_j to $s_j + l_j$ ($l_j \in \mathbb{Z}$), it can be calculated as

$$m_{s_j+l_j} = \sum_{i=s_j}^{s_j+l_j} c_i x_i \tag{2.2a}$$

Another representation of $m_{s_j+l_j}$ is

$$m_{s_j+l_j} = C_s^l(j) X_s^l(j) \tag{2.2b}$$

where

$$C_s^l(j) = \begin{bmatrix} c_{s_j} & c_{s_j+1} & \ldots & c_{s_j+l_j} \end{bmatrix} \in \mathbb{R}^{n_m \times (l_j+1)n_x}$$
$$X_s^l(j) = \begin{bmatrix} x_{s_j}^T & x_{s_j+1}^T & \ldots & x_{s_j+l_j}^T \end{bmatrix}^T \in \mathbb{R}^{(l_j+1)n_x}$$

As the slow irregular laboratory measurement $y_{k_s(j)}^m$ is usually delayed and much more accurate than the fast measurement, the measurement arrives at time step $k_s(j) = s_j + l_j + d_j$, where $d_j \in \mathbb{Z}$ is the delayed time, and the noise covariance R^m is much smaller than R^f. As illustrated in Figure 2.1, the sampling interval is ρ_j which is assumed $\rho_j \geqslant l_j + d_j + 1$.

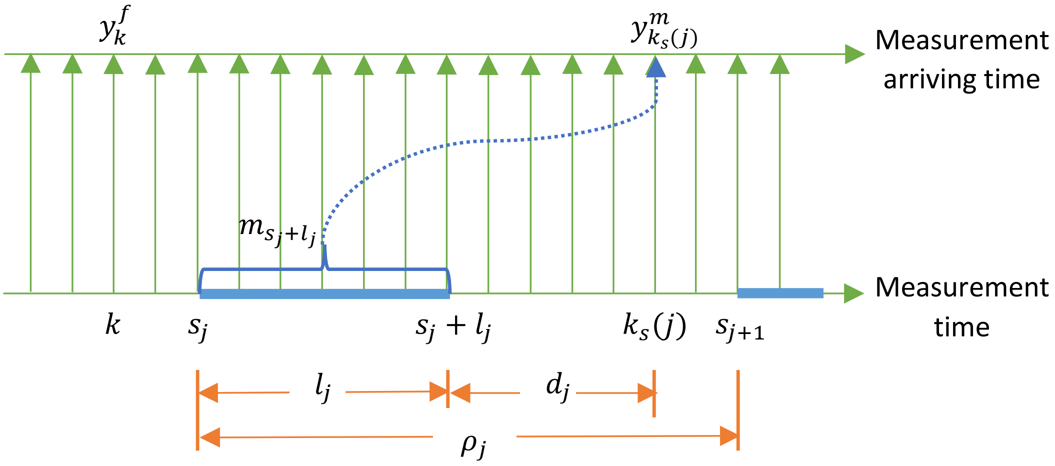

Figure 2.1: System with fast and delayed slow measurements

The goal is to estimate the state by fusing the fast measurements and delayed slow integral laboratory measurements. As the fast measurements are online and frequent, the frequent state estimation can be conducted using fast measurements, while the delayed slow integral laboratory measurements are used to improve the accuracy of the estimation.

The state estimation can also be seen as a filtering problem. In Bayesian filtering, the task is to derive the posterior density, denoted as $p\left(x_k \mid y_{1:k}\right)$. For our system, the measurements $y_{1:k}$ consist of both fast and slow measurements arriving at and before time instant k, i.e. $y_{1:k} \triangleq \left\{y^f_{1:k}, y^m_{k_s(1):k_s(j)}\right\}$ where $k_s(j) \leqslant k$. Once the posterior density is derived, it is easy to obtain a point estimate of x_k, such as the mean, mode or median. The next two sections describe the estimation of the posterior density $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$, which is defined as

$$p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right) \triangleq p\left(x_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}, y^m_{k_s(j)}\right) \tag{2.3}$$

where $y_{1:s_j-1} \triangleq \left\{y^f_{1:s_j-1}, y^m_{k_s(1):k_s(j-1)}\right\}$.

2.3 Exact Bayesian solution

2.3.1 Bayesian approach

According to Bayes' rule, the posterior density in equation (2.3) can be calculated as

$$p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right) = \frac{p\left(y^m_{k_s(j)} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) p\left(x_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)}{\int p\left(y^m_{k_s(j)} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) p\left(x_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) dx_{k_s(j)}} \tag{2.4}$$

where $p\left(x_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ can be obtained from frequent state estimation, the likelihood $p\left(y^m_{k_s(j)} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$ is calculated as follows

$$p\left(y^m_{k_s(j)} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) = \int p\left(y^m_{k_s(j)} \mid m_{s_j+l_j}\right) p\left(m_{s_j+l_j} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) dm_{s_j+l_j} \tag{2.5}$$

In equation (2.5), the first term in the right hand side is determined by the slow measurement equation (2.1c). For the second term, because the relationship between $m_{s_j+l_j}$ and $X^l_s(j)$ is linear, it is equivalent to calculate $p\left(X^l_s(j) \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$. So the remaining task is to obtain this probability density, for which we consider two cases.

Case 1: The delayed time $d_j = 1$, that is $k_s(j) = s_j + l_j + 1$. Using Bayes' rule yields

$$p\left(X^l_s(j) \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) = \frac{p\left(x_{k_s(j)} \mid X^l_s(j)\right) p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)}{\int p\left(x_{k_s(j)} \mid X^l_s(j)\right) p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) dX^l_s(j)} \tag{2.6}$$

In equation (2.6), because of the $x_{s_j+l_j}$ in $X^l_s(j)$, $p\left(x_{k_s(j)} \mid X^l_s(j)\right)$ can be derived directly from the state transition equation (2.1a). $p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$ is a smoothing density of $X^l_s(j)$. Thus, $p\left(X^l_s(j) \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$ can be calculated from equation (2.6).

Case 2: The delayed time $d_j > 1$, that is $k_s(j) > s_j + l_j + 1$. First, rewrite the density as

$$p\left(X^l_s(j) \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) = \int p\left(X^l_s(j) \mid x_{s_j+l_j+1}, y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right) p\left(x_{s_j+l_j+1} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) dx_{s_j+l_j+1} \tag{2.7}$$

which has made use of the Markov property of the state. The first term on the right hand side of equation (2.7) is given from Bayes' rule:

$$p\left(X_s^l(j) \mid x_{s_j+l_j+1}, y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right) = \frac{p\left(x_{s_j+l_j+1} \mid X_s^l(j)\right) p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)}{\int p\left(x_{s_j+l_j+1} \mid X_s^l(j)\right) p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right) dX_s^l(j)} \tag{2.8}$$

which can be fully constructed similar to equation (2.6). The second term on the right hand side of equation (2.7) needs to be obtained recursively. For $n = k_s(j)-2, k_s(j)-3, \cdots, s_j + l_j + 1$,

$$p\left(x_n \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) = \int p\left(x_n \mid x_{n+1}, y_{1:s_j-1}, y^f_{s_j:n}\right) p\left(x_{n+1} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) dx_{n+1} \tag{2.9}$$

$p\left(x_n \mid x_{n+1}, y_{1:s_j-1}, y^f_{s_j:n}\right)$ in equation (2.9) is given by

$$p\left(x_n \mid x_{n+1}, y_{1:s_j-1}, y^f_{s_j:n}\right) = \frac{p\left(x_{n+1} \mid x_n\right) p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right)}{\int p\left(x_{n+1} \mid x_n\right) p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right) dx_n} \tag{2.10}$$

where $p\left(x_{n+1} \mid x_n\right)$ is determined by state transition equation, $p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right)$ can be obtained from frequent state estimation. The initial value of the second term on the right hand side in equation (2.9) is $p\left(x_{k_s(j)-1} \mid x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$. It can be obtained by substituting $n = k_s(j) - 1$ into equation (2.10).

From the derivation above, we can see that the sufficient statistics to get the target posterior density are $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$ and $\left\{p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right)\right\}_{n=s_j+l_j+1}^{k_s(j)}$. Figure 2.2 is the schematic of exact Bayesian method. It should be noted that this method needs to store this information until the delayed slow measurement $y^m_{k_s(j)}$ arrives.

2.3.2 Particle filter implementation

If the PF is implemented, the statistics $\left\{p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right)\right\}_{n=s_j+l_j+1}^{k_s(j)}$ are approximated by particle sets $\left\{x_n^{f(i)}, w_n^{f(i)} : n = s_j + l_j + 1, \cdots, k_s(j)\right\}_{i=1}^{N}$ from frequent state estimation. Since $X_s^l(j)$ is just a combination of system states, $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$ can be derived

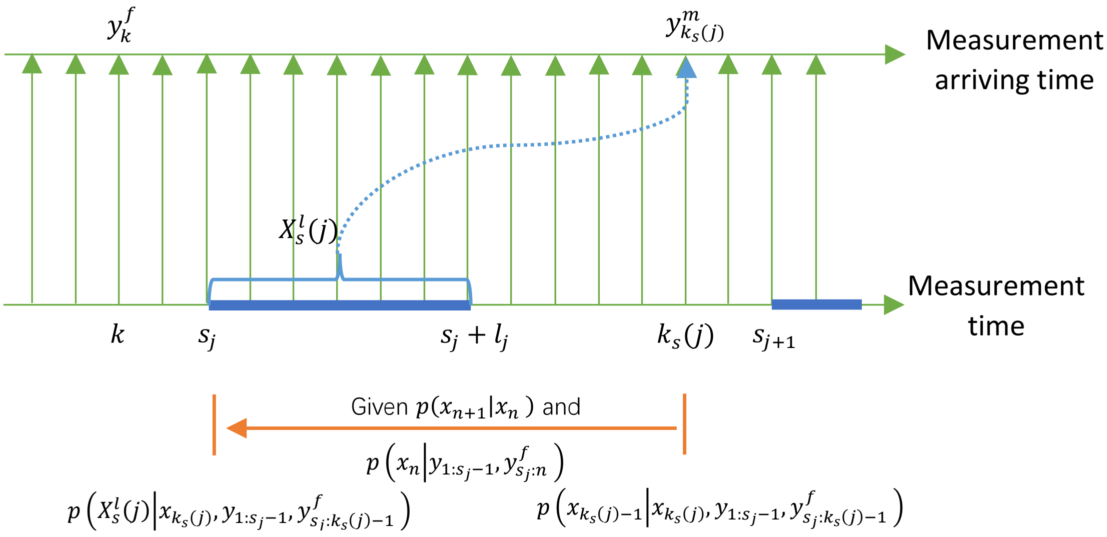
y^f_k
$y^m_{k_s(j)}$
Measurement arriving time
$X^l_s(j)$
Measurement time
k
s_j
$s_j + l_j$
$k_s(j)$
s_{j+1}
Given $p(x_{n+1}|x_n)$ and
$p\left(x_n\middle|y_{1:s_j-1}, y^f_{s_j:n}\right)$
$p\left(X^l_s(j)\middle|x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$
$p\left(x_{k_s(j)-1}\middle|x_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$

Figure 2.2: Schematic of the exact Bayesian method

by storing the particle trajectory from time s_j to $s_j + l_j$ and it is assumed to be described by the particle set $\left\{\{X^l_s(j)\}^{f(i)}_{s_j+l_j}, W^{f(i)}_{s_j+l_j}\right\}^N_{i=1}$. Note that $W^{f(i)}_{s_j+l_j} = w^{f(i)}_{s_j+l_j}$.

Now, we have

$$p\left(x_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) \approx \sum_{i=1}^{N} w^{f(i)}_{k_s(j)} \delta\left(x_{k_s(j)} - x^{f(i)}_{k_s(j)}\right) \tag{2.11}$$

Then, substitute equation (2.11) into equation (2.4) and get

$$p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right) \approx \frac{\sum_{i=1}^{N} w^{f(i)}_{k_s(j)} p\left(y^m_{k_s(j)} \mid x^{f(i)}_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) \delta\left(x_{k_s(j)} - x^{f(i)}_{k_s(j)}\right)}{\sum_{r=1}^{N} w^{f(r)}_{k_s(j)} p\left(y^m_{k_s(j)} \mid x^{f(r)}_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)} \tag{2.12}$$

Consider another representation of $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$ as

$$p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right) \approx \sum_{i=1}^{N} w^{(i)}_{k_s(j)} \delta\left(x_{k_s(j)} - x^{f(i)}_{k_s(j)}\right) \tag{2.13}$$

By comparing equations (2.12) and (2.13), it's easy to find

$$w^{(i)}_{k_s(j)} = \frac{w^{f(i)}_{k_s(j)} p\left(y^m_{k_s(j)} \mid x^{f(i)}_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)}{\sum_{r=1}^{N} w^{f(r)}_{k_s(j)} p\left(y^m_{k_s(j)} \mid x^{f(r)}_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)} \tag{2.14}$$

Thus, the goal is to obtain $p\left(y^m_{k_s(j)} \mid x^{f(i)}_{k_s(j)}, y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)$ in equation (2.14). In the

Bayesian approach (Section 2.3.1), this density can be calculated as equation (2.5). Suppose

$$p\left(X_s^l(j) \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right) \approx \sum_{r=1}^{N} W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) \delta\left(X_s^l(j) - \left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}\right) \tag{2.15}$$

where $W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right)$ is the weight of the particle $\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}$ given the state $x_{k_s(j)}^{f(i)}$. Then $p\left(m_{s_j+l_j} \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right)$ in equation (2.5) can be approximated by the particle set $\left\{C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}, W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right)\right\}_{r=1}^{N}$. Therefore,

$$p\left(y_{k_s(j)}^m \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right) \approx \sum_{r=1}^{N} W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) p\left(y_{k_s(j)}^m \mid C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}\right) \tag{2.16}$$

Substituting equation (2.16) into equation (2.14) yields

$$w_{k_s(j)}^{(i)} = \frac{\sum_{r=1}^{N} w_{k_s(j)}^{f(i)} W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) p\left(y_{k_s(j)}^m \mid C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}\right)}{\sum_{t=1}^{N}\sum_{n=1}^{N} w_{k_s(j)}^{f(t)} W_{k_s(j)-1}^{(n)}\left(x_{k_s(j)}^{f(t)}\right) p\left(y_{k_s(j)}^m \mid C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(n)}\right)} \tag{2.17}$$

Then, the remaining task is to obtain the particle set $\left\{\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}, W_{k_s(j)-1}^{(r)}(x_{k_s(j)}^{f(i)})\right\}_{r=1}^{N}$ which approximates the density $p\left(X_s^l(j) \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right)$ in equation (2.15). There are two cases as discussed in the Bayesian approach (Section 2.3.1).

Case 1: The delayed time $d_j = 1$, that is $k_s(j) = s_j + l_j + 1$. $p\left(X_s^l(j) \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right)$ is calculated through equation (2.6). By substituting the particle representation of $p\left(X_s^l(j) \mid y_{1:s_j-1}, y_{s_j:s_j+l_j}^f\right)$, it can be derived that

$$p\left(X_s^l(j) \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right) \approx \frac{\sum_{r=1}^{N} W_{s_j+l_j}^{f(r)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right) \delta\left(X_s^l(j) - \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^{N} W_{s_j+l_j}^{f(n)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(n)}\right)} \tag{2.18}$$

Therefore, by comparing equations (2.18) and (2.15), one can note

$$\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)} = \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)} \tag{2.19a}$$

$$W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) = \frac{W_{s_j+l_j}^{f(r)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^{N} W_{s_j+l_j}^{f(n)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(n)}\right)} \tag{2.19b}$$

Case 2: The delayed time $d_j > 1$, that is $k_s(j) > s_j+l_j+1$. $p\big(X_s^l(j) \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\big)$ can be calculated through equation (2.7). Substituting $p\big(X_s^l(j) \mid y_{1:s_j-1}, y_{s_j:s_j+l_j}^f\big) \approx \sum_{r=1}^N W_{s_j+l_j}^{f(r)} \delta\left(X_s^l(j) - \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)$ into equation (2.8), the first term on the right hand side of equation (2.7) can be calculated as

$$p\left(X_s^l(j) \mid x_{s_j+l_j+1}, y_{1:s_j-1}, y_{s_j:s_j+l_j}^f\right) \approx \frac{\sum_{r=1}^N W_{s_j+l_j}^{f(r)} p\left(x_{s_j+l_j+1} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right) \delta\left(X_s^l(j) - \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^N W_{s_j+l_j}^{f(n)} p\left(x_{s_j+l_j+1} \mid \{X_s^l(j)\}_{s_j+l_j}^{f(n)}\right)} \quad (2.20)$$

Suppose the second term on the right hand side of equation (2.7) can be written as

$$p\left(x_{s_j+l_j+1} \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right) \approx \sum_{r=1}^N w_{s_j+l_j+1|k_s(j)-1}^{f(r)} \left(x_{k_s(j)}^{f(i)}\right) \delta\left(x_{s_j+l_j+1} - x_{s_j+l_j+1|k_s(j)-1}^{f(r)}\right) \quad (2.21)$$

Substituting equations (2.20) and (2.21) into equation (2.7) and comparing with equation (2.15), we can see that

$$\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)} = \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)} \quad (2.22a)$$

$$W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) = W_{s_j+l_j}^{f(r)} \left[\sum_{t=1}^N \frac{w_{s_j+l_j+1|k_s(j)-1}^{f(t)} \left(x_{k_s(j)}^{f(i)}\right) p\left(x_{s_j+l_j+1|k_s(j)-1}^{f(t)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^N W_{s_j+l_j}^{f(n)} p\left(x_{s_j+l_j+1|k_s(j)-1}^{f(t)} \mid \{X_s^l(j)\}_{s_j+l_j}^{f(n)}\right)}\right] \quad (2.22b)$$

Now, the only unknown information is $\left\{x_{s_j+l_j+1|k_s(j)-1}^{f(r)}, w_{s_j+l_j+1|k_s(j)-1}^{f(r)}\left(x_{k_s(j)}^{f(i)}\right)\right\}_{r=1}^N$ in equation (2.21). In order to obtain this particle set, first using equation (2.10) and replacing $p\left(x_n \mid y_{1:s_j-1}, y_{s_j:n}^f\right)$ by its discrete representation, $p\left(x_n \mid x_{n+1}, y_{1:s_j-1}, y_{s_j:n}^f\right)$ is given by

$$p\left(x_n \mid x_{n+1}, y_{1:s_j-1}, y_{s_j:n}^f\right) \approx \frac{\sum_{i=1}^N w_n^{f(i)} p\left(x_{n+1} \mid x_n^{f(i)}\right) \delta\left(x_n - x_n^{f(i)}\right)}{\sum_{r=1}^N w_n^{f(r)} p\left(x_{n+1} \mid x_n^{f(r)}\right)} \quad (2.23)$$

The density $p\left(x_{n+1} \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right)$ in equation (2.9) is written as

$$p\left(x_{n+1} \mid x_{k_s(j)}^{f(i)}, y_{1:s_j-1}, y_{s_j:k_s(j)-1}^f\right) \approx \sum_{r=1}^N w_{n+1|k_s(j)-1}^{f(r)} \left(x_{k_s(j)}^{f(i)}\right) \delta\left(x_{n+1} - x_{n+1|k_s(j)-1}^{f(r)}\right) \quad (2.24)$$

Then, substituting equations (2.23) and (2.24) into equation (2.9) yields

$$p\left(x_n \mid x^{f(i)}_{k_s(j)}, y_{1:s_j-1}, y^{f}_{s_j:k_s(j)-1}\right) \approx \\ \sum_{r=1}^{N} w_n^{f(r)} \left[\sum_{t=1}^{N} \frac{w^{f(t)}_{n+1|k_s(j)-1}\left(x^{f(i)}_{k_s(j)}\right) p\left(x^{f(t)}_{n+1|k_s(j)-1} \mid x_n^{f(r)}\right)}{\sum_{q=1}^{N} w_n^{f(q)} p\left(x^{f(t)}_{n+1|k_s(j)-1} \mid x_n^{f(q)}\right)}\right] \delta\left(x_n - x_n^{f(r)}\right) \tag{2.25}$$

The above can be written as

$$p\left(x_n \mid x^{f(i)}_{k_s(j)}, y_{1:s_j-1}, y^{f}_{s_j:k_s(j)-1}\right) \approx \sum_{r=1}^{N} w^{f(r)}_{n|k_s(j)-1}\left(x^{f(i)}_{k_s(j)}\right) \delta\left(x_n - x^{f(r)}_{n|k_s(j)-1}\right) \tag{2.26}$$

where

$$x^{f(r)}_{n|k_s(j)-1} = x_n^{f(r)} \tag{2.27a}$$

$$w^{f(r)}_{n|k_s(j)-1}\left(x^{f(i)}_{k_s(j)}\right) = w_n^{f(r)} \left[\sum_{t=1}^{N} \frac{w^{f(t)}_{n+1|k_s(j)-1}\left(x^{f(i)}_{k_s(j)}\right) p\left(x^{f(t)}_{n+1|k_s(j)-1} \mid x_n^{f(r)}\right)}{\sum_{q=1}^{N} w_n^{f(q)} p\left(x^{f(t)}_{n+1|k_s(j)-1} \mid x_n^{f(q)}\right)}\right] \tag{2.27b}$$

The recursion in equation (2.27) is carried out for $n = k_s(j)-2, k_s(j)-3, \cdots, s_j+l_j+1$. At the end of the recursion, $\left\{x^{f(r)}_{s_j+l_j+1|k_s(j)-1}, w^{f(r)}_{s_j+l_j+1|k_s(j)-1}(x^{f(i)}_{k_s(j)})\right\}_{r=1}^{N}$ can be obtained. The initial values of $\left\{x^{f(r)}_{n|k_s(j)-1}, w^{f(r)}_{n|k_s(j)-1}(x^{f(i)}_{k_s(j)})\right\}_{r=1}^{N}$ are $\left\{x^{f(r)}_{k_s(j)-1}, w^{f(r)}_{k_s(j)-1|k_s(j)-1}(x^{f(i)}_{k_s(j)})\right\}_{r=1}^{N}$ which can be calculated using equation (2.23). This completes the PF implementation. We call this algorithm as EB-PF. Algorithm 1 shows the pseudo-code of one fusion loop in EB-PF. The required particle sets $\left\{\{X_s^l(j)\}^{f(i)}_{s_j+l_j}, W^{f(i)}_{s_j+l_j}\right\}_{i=1}^{N}$ and $\left\{x_n^{f(i)}, w_n^{f(i)} : n = s_j + l_j + 1, \cdots, k_s(j)\right\}_{i=1}^{N}$ are available from the previous filtering.

2.3.3 Unscented Kalman filter implementation

The unscented Kalman filter (UKF), like other Gaussian filters, assumes that all filtering density are Gaussian. It approximates the target posterior density $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$ using a Gaussian distribution $\mathcal{N}\left(\mu^x_{k_s(j)|k_s(j)}, \Sigma^x_{k_s(j)|k_s(j)}\right)$. In this assumption, the densities $\left\{p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:n}\right)\right\}_{n=s_j+l_j+1}^{k_s(j)}$ are represented as Gaussian distributions $\left\{\mathcal{N}\left(\mu^{x^f}_{n|n}, \Sigma^{x^f}_{n|n}\right)\right\}_{n=s_j+l_j+1}^{k_s(j)}$ obtained directly from the UKF, while $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$ needs to be calculated additionally.

Prior to derive the target posterior density, two propositions are introduced as the expanded forms of Proposition 1 and 2 in [10].

Algorithm 1 EB-PF for estimating the posterior density $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$

```
Input: $\left\{x_n^{f(i)}, w_n^{f(i)} : n = s_j + l_j + 1, ..., k_s(j)\right\}_{i=1}^N$ and $\left\{\left\{X_s^l(j)\right\}_{s_j+l_j}^{f(i)}, W_{s_j+l_j}^{f(i)}\right\}_{i=1}^N$
Output: Updated weights $\left\{w_{k_s(j)}^{(i)}\right\}_{i=1}^N$ for particles $\left\{x_{k_s(j)}^{f(i)}\right\}_{i=1}^N$
%% Calculate particle set $\left\{\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}, W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right)\right\}_{r=1}^N$
if $k_s(j) = s_j + l_j + 1$ then
    for $r = 1 : N$ do
        $\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)} = \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}$
        $W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) = \frac{W_{s_j+l_j}^{f(r)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^N W_{s_j+l_j}^{f(n)} p\left(x_{k_s(j)}^{f(i)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(n)}\right)}$
    end for
else
    %% Calculate particle set $\left\{x_{s_j+l_j+1|k_s(j)-1}^{f(r)}, w_{s_j+l_j+1|k_s(j)-1}^{f(r)}\left(x_{k_s(j)}^{f(i)}\right)\right\}_{r=1}^N$
    for $r = 1 : N$ do
        $x_{k_s(j)-1|k_s(j)-1}^{f(r)} = x_{k_s(j)-1}^{f(r)}$
        $w_{k_s(j)-1|k_s(j)-1}^{f(r)}\left(x_{k_s(j)}^{f(i)}\right) = \frac{w_{k_s(j)-1}^{f(r)} p\left(x_{k_s(j)}^{f(i)} \mid x_{k_s(j)-1}^{f(r)}\right)}{\sum_{t=1}^N w_{k_s(j)-1}^{f(t)} p\left(x_{k_s(j)}^{f(i)} \mid x_{k_s(j)-1}^{f(t)}\right)}$
    end for
    if $k_s(j) > s_j + l_j + 2$ then
        for $n = k_s(j) - 2 : s_j + l_j + 1$ do
            for $r = 1 : N$ do
                $x_{n|k_s(j)-1}^{f(r)} = x_n^{f(r)}$
                $w_{n|k_s(j)-1}^{f(r)}\left(x_{k_s(j)}^{f(i)}\right) =$
                    $w_n^{f(r)}\left[\sum_{t=1}^N \frac{w_{n+1|k_s(j)-1}^{f(t)}\left(x_{k_s(j)}^{f(i)}\right) p\left(x_{n+1|k_s(j)-1}^{f(t)} \mid x_n^{f(r)}\right)}{\sum_{q=1}^N w_n^{f(q)} p\left(x_{n+1|k_s(j)-1}^{f(t)} \mid x_n^{f(q)}\right)}\right]$
            end for
        end for
    end if
    for $r = 1 : N$ do
        $\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)} = \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}$
        $W_{k_s(j)-1}^{(r)}\left(x_{k_s(j)}^{f(i)}\right) =$
            $W_{s_j+l_j}^{f(r)}\left[\sum_{t=1}^N \frac{w_{s_j+l_j+1|k_s(j)-1}^{f(t)}\left(x_{k_s(j)}^{f(i)}\right) p\left(x_{s_j+l_j+1|k_s(j)-1}^{f(t)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(r)}\right)}{\sum_{n=1}^N W_{s_j+l_j}^{f(n)} p\left(x_{s_j+l_j+1|k_s(j)-1}^{f(t)} \mid \left\{X_s^l(j)\right\}_{s_j+l_j}^{f(n)}\right)}\right]$
    end for
end if
```

```
%% Calculate the output $\left\{w_{k_s(j)}^{(i)}\right\}_{i=1}^{N}$
for $i = 1 : N$ do
    $w_{k_s(j)}^{(i)} = \frac{\sum_{r=1}^{N} w_{k_s(j)}^{f(i)} W_{k_s(j)-1}^{(r)} \left(x_{k_s(j)}^{f(i)}\right) p\left(y_{k_s(j)}^{m} \mid C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(r)}\right)}{\sum_{t=1}^{N}\sum_{n=1}^{N} w_{k_s(j)}^{f(t)} W_{k_s(j)-1}^{(n)} \left(x_{k_s(j)}^{f(t)}\right) p\left(y_{k_s(j)}^{m} \mid C_s^l(j)\left\{X_s^l(j)\right\}_{k_s(j)-1}^{(n)}\right)}$
end for
```

Proposition 1. *A density $p\left(A_\tau \mid C_t, B_{1:T}\right)$ which can be calculated as*

$$p\left(A_\tau \mid C_t, B_{1:T}\right) = \frac{p\left(A_\tau, C_t \mid B_{1:T}\right)}{p\left(C_t \mid B_{1:T}\right)} \tag{2.28}$$

where $\tau \leq t \leq T$, A_k, B_k and C_k are related variables, can be approximated as a Gaussian distribution $\mathcal{N}\left(A_\tau \mid M, S\right)$. The moments of this approximation are in general computed through

$$M = \mu_{\tau|T}^{A} + \Sigma_{\tau,t|T}^{AC}\left(\Sigma_{t|T}^{C}\right)^{-1}\left(C_t - \mu_{t|T}^{C}\right) \tag{2.29a}$$

$$S = \Sigma_{\tau|T}^{A} - \Sigma_{\tau,t|T}^{AC}\left(\Sigma_{t|T}^{C}\right)^{-1}\left(\Sigma_{\tau,t|T}^{AC}\right)^{T} \tag{2.29b}$$

where $\mathcal{N}\left(\mu_{\tau|T}^{A}, \Sigma_{\tau|T}^{A}\right)$ is the Gaussian approximation of $p\left(A_\tau \mid B_{1:T}\right)$, $\mathcal{N}\left(\mu_{t|T}^{C}, \Sigma_{t|T}^{C}\right)$ is the Gaussian approximation of $p\left(C_t \mid B_{1:T}\right)$, and $\Sigma_{\tau,t|T}^{AC}$ is the covariance of A_τ and C_t given $B_{1:T}$.

Proof for Proposition 1. Assume the joint distribution $p\left(A_\tau, C_t \mid B_{1:T}\right)$ in equation (2.28) can be approximated by a Gaussian distribution, that is

$$p\left(A_\tau, C_t \mid B_{1:T}\right) \approx \mathcal{N}\left(\begin{bmatrix}\mu_{\tau|T}^{A} \\ \mu_{t|T}^{C}\end{bmatrix}, \begin{bmatrix}\Sigma_{\tau|T}^{A} & \Sigma_{\tau,t|T}^{AC} \\ \left(\Sigma_{\tau,t|T}^{AC}\right)^{T} & \Sigma_{t|T}^{C}\end{bmatrix}\right) = \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$$

Substituting the equation above into equation (2.28) yields

$$\begin{aligned} p\left(A_\tau \mid C_t, B_{1:T}\right) &\approx \frac{\frac{1}{\sqrt{(2\pi)^{n_A+n_C}|\boldsymbol{\Sigma}|}}\exp\left\{-\frac{1}{2}\left(\begin{bmatrix}A_\tau & C_t\end{bmatrix}^T - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\cdots\right)\right\}}{\frac{1}{\sqrt{(2\pi)^{n_C}|\Sigma_{t|T}^{C}|}}\exp\left\{-\frac{1}{2}\left(C_t - \mu_{t|T}^{C}\right)^T\left(\Sigma_{t|T}^{C}\right)^{-1}\left(\cdots\right)\right\}} \\ &\approx \frac{1}{\sqrt{(2\pi)^{n_A}|S|}}\exp\left\{-\frac{1}{2}\left(A_\tau - M\right)^T S^{-1}\left(\cdots\right)\right\} \end{aligned}$$

where

$$M = \mu_{\tau|T}^{A} + \Sigma_{\tau,t|T}^{AC}\left(\Sigma_{t|T}^{C}\right)^{-1}\left(C_t - \mu_{t|T}^{C}\right)$$

$$S = \Sigma_{\tau|T}^{A} - \Sigma_{\tau,t|T}^{AC}\left(\Sigma_{t|T}^{C}\right)^{-1}\left(\Sigma_{\tau,t|T}^{AC}\right)^{T}$$

This concludes the proof of Proposition 1. □

Proposition 2. *For a density $p\left(A_\tau \mid B_{1:T}\right)$ which can be calculated as*

$$p\left(A_\tau \mid B_{1:T}\right) = \int p\left(A_\tau \mid C_t, B_{1:t-1}\right) p\left(C_t \mid B_{1:T}\right) dC_t \tag{2.30}$$

where $\tau \leq t \leq T$, C_k and B_k satisfy the hidden Markov property and A_k is a related variable, the mean and covariance of a Gaussian approximation to the density $p\left(A_\tau \mid B_{1:T}\right)$ are generally computed as

$$\mu^A_{\tau|T} = \mu^A_{\tau|t-1} + J_\tau\left(\mu^C_{t|T} - \mu^C_{t|t-1}\right) \tag{2.31a}$$

$$\Sigma^A_{\tau|T} = \Sigma^A_{\tau|t-1} + J_\tau\left(\Sigma^C_{t|T} - \Sigma^C_{t|t-1}\right) J^T_\tau \tag{2.31b}$$

$$J_\tau = \Sigma^{AC}_{\tau,t|t-1}\left(\Sigma^C_{t|t-1}\right)^{-1} \tag{2.31c}$$

where $\mathcal{N}\left(\mu^A_{\tau|t-1}, \Sigma^A_{\tau|t-1}\right)$ is the Gaussian approximation of $p\left(A_\tau \mid B_{1:t-1}\right)$, $\mathcal{N}\left(\mu^C_{t|t-1}, \Sigma^C_{t|t-1}\right)$ is the Gaussian approximation of $p\left(C_t \mid B_{1:t-1}\right)$, $\Sigma^{AC}_{\tau,t|t-1}$ is the covariance of A_τ and C_t given $B_{1:t-1}$, and $\mathcal{N}\left(\mu^C_{t|T}, \Sigma^C_{t|T}\right)$ is the Gaussian approximation of $p\left(C_t \mid B_{1:T}\right)$.

Proof for Proposition 2. To compute the density $p\left(A_\tau \mid B_{1:T}\right)$ in equation (2.30), we need to first obtain the conditional density $p\left(A_\tau \mid C_t, B_{1:t-1}\right)$ and formulate it as an unnormalized density in C_t, then multiply it by $p\left(C_t \mid B_{1:T}\right)$, and finally integrate over C_t.

From Proposition 1, we know a density with formulation as $p\left(A_\tau \mid C_t, B_{1:T}\right)$ can be approximated as a Gaussian distribution. In Proposition 2, we add an assumption that C_k and B_k satisfy the hidden Markov property, so $p\left(A_\tau \mid C_t, B_{1:T}\right) = p\left(A_\tau \mid C_t, B_{1:t-1}\right)$. By applying Proposition 1, $p\left(A_\tau \mid C_t, B_{1:t-1}\right)$ can be approximated as $\mathcal{N}\left(A_\tau \mid M, S\right)$, where

$$\begin{aligned} M &= \mu^A_{\tau|t-1} + J_\tau\left(C_t - \mu^C_{t|t-1}\right) \\ S &= \Sigma^A_{\tau|t-1} - J_\tau\left(\Sigma^{AC}_{\tau,t|t-1}\right)^T \\ J_\tau &= \Sigma^{AC}_{\tau,t|t-1}\left(\Sigma^C_{t|t-1}\right)^{-1} \end{aligned}$$

The square root of the exponent of $\mathcal{N}\left(A_\tau \mid M, S\right)$ contains

$$A_\tau - M = \gamma\left(A_\tau\right) - J_\tau C_t$$

with $\gamma\left(A_\tau\right) = A_\tau - \mu^A_{\tau|t-1} + J_\tau \mu^C_{t|t-1}$, which is a linear function of both A_τ and C_t. By reformulating $\mathcal{N}\left(A_\tau \mid M, S\right)$ as a Gaussian of $J_\tau C_t$ with mean $\gamma\left(A_\tau\right)$ and unchanged variance

$\boldsymbol{S}$, we can obtain

$$\begin{aligned}\mathcal{N}\left(A_\tau \mid M, S\right) &= \mathcal{N}\left(J_\tau C_t \mid \gamma\left(A_\tau\right), S\right)\\ &= \frac{1}{\sqrt{\left(2\pi\right)^{n_A}\left|S\right|}} \exp\left\{-\frac{1}{2}\left(J_\tau C_t - \gamma\left(A_\tau\right)\right)^T S^{-1}\left(\cdots\right)\right\}\\ &= \varepsilon_1 \frac{\exp\left\{-\frac{1}{2}\left(C_t - J_\tau^{-1}\gamma\left(A_\tau\right)\right)^T J_\tau^T S^{-1} J_\tau\left(\cdots\right)\right\}}{\sqrt{\left(2\pi\right)^{n_C}\left|\left(J_\tau^T S^{-1} J_\tau\right)^{-1}\right|}}\\ &= \varepsilon_1 \mathcal{N}\left(C_t \mid \mu_1, \Sigma_1\right)\end{aligned}$$

where $\varepsilon_1 = \sqrt{\left(2\pi\right)^{n_C - n_A}\left|\left(J_\tau^T S^{-1} J_\tau\right)^{-1}\right|/\left|S\right|}$, $\mu_1 = J_\tau^{-1}\gamma\left(A_\tau\right)$ and $\Sigma_1 = \left(J_\tau^T S^{-1} J_\tau\right)^{-1}$. Note that the matrix J_τ defined in equation (2.31c) may not be quadratic, so for the non-quadratic case, we can use the pseudo-inverse instead.

Multiplying the new distribution with the Gaussian approximation of $p\left(C_t \mid B_{1:T}\right)$, which is $\mathcal{N}\left(\mu_{t|T}^C, \Sigma_{t|T}^C\right)$, we can derive

$$\varepsilon_1 \mathcal{N}\left(C_t \mid \mu_1, \Sigma_1\right)\mathcal{N}\left(\mu_{t|T}^C, \Sigma_{t|T}^C\right) = \varepsilon_1 \varepsilon_2\left(\mu_1\right)\mathcal{N}\left(C_t \mid \mu_2, \Sigma_2\right)$$

where

$$\begin{aligned}\mu_2 &= \left[\Sigma_1^{-1} + \left(\Sigma_{t|T}^C\right)^{-1}\right]^{-1}\left[\Sigma_1^{-1}\mu_1 + \left(\Sigma_{t|T}^C\right)^{-1}\mu_{t|T}^C\right]\\ \Sigma_2 &= \left[\Sigma_1^{-1} + \left(\Sigma_{t|T}^C\right)^{-1}\right]^{-1}\\ \varepsilon_2\left(\mu_1\right) &= \frac{\sqrt{\left|\Sigma_2\right|}}{\sqrt{\left(2\pi\right)^{n_C}\left|\Sigma_1\right|\left|\Sigma_{t|T}^C\right|}} \exp\left\{-\frac{1}{2}\left[\mu_1^T\Sigma_1^{-1}\mu_1 + \left(\mu_{t|T}^C\right)^T\left(\Sigma_{t|T}^C\right)^{-1}\mu_{t|T}^C - \mu_2^T\Sigma_2^{-1}\mu_2\right]\right\}\end{aligned}$$

Since we integrate over C_t in equation (2.30), we are only interested in the parts which are independent of C_t. They are the constants ε_1 and $\varepsilon_2\left(\mu_1\right)$, where the constant $\varepsilon_2\left(\mu_1\right)$ can be rewritten as $\varepsilon_2\left(A_\tau\right)$ by reversing the step that inverted the matrix J_τ. Then, $\varepsilon_2\left(A_\tau\right)$ is given by

$$\varepsilon_2\left(A_\tau\right) = \varepsilon_1^{-1}\mathcal{N}\left(A_\tau \mid \mu_{\tau|T}^A, \Sigma_{\tau|T}^A\right)$$

where

$$\begin{aligned}\mu_{\tau|T}^A &= \mu_{\tau|t-1}^A + J_\tau\left(\mu_{t|T}^C - \mu_{t|t-1}^C\right)\\ \Sigma_{\tau|T}^A &= \Sigma_{\tau|t-1}^A + J_\tau\left(\Sigma_{t|T}^C - \Sigma_{t|t-1}^C\right)J_\tau^T\end{aligned}$$

As $\varepsilon_1 \varepsilon_1^{-1} = 1$, the desired density is

$$p\left(A_\tau \mid B_{1:T}\right) \approx \mathcal{N}\left(A_\tau \mid \mu^A_{\tau|T}, \Sigma^A_{\tau|T}\right)$$

This concludes the proof of Proposition 2. □

Using Bayes' rule, equation (2.4) can also be written as

$$p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right) = \frac{p\left(x_{k_s(j)}, y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)}{p\left(y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)} \tag{2.32}$$

Assume the joint distribution $p\left(x_{k_s(j)}, y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ in equation (2.32) can be approximated by a Gaussian distribution

$$p\left(x_{k_s(j)}, y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) \approx \mathcal{N}\left(\begin{bmatrix} \mu^{x^f}_{k_s(j)|k_s(j)} \\ \mu^{y^m}_{k_s(j)|k_s(j)} \end{bmatrix}, \begin{bmatrix} \Sigma^{x^f}_{k_s(j)|k_s(j)} & \Sigma^{x^f y^m}_{k_s(j)|k_s(j)} \\ \Sigma^{y^m x^f}_{k_s(j)|k_s(j)} & \Sigma^{y^m}_{k_s(j)|k_s(j)} \end{bmatrix}\right) \tag{2.33}$$

Then, applying Proposition 1, let $\tau = t = T = k_s(j)$, $A_\tau = x_{k_s(j)}$, $B_{1:T} = \left\{y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right\}$ and $C_t = y^m_{k_s(j)}$; the Gaussian approximation of $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$ can be obtained as

$$\mu^x_{k_s(j)|k_s(j)} = \mu^{x^f}_{k_s(j)|k_s(j)} + \Sigma^{x^f y^m}_{k_s(j)|k_s(j)} \left(\Sigma^{y^m}_{k_s(j)|k_s(j)}\right)^{-1} \left(y^m_{k_s(j)} - \mu^{y^m}_{k_s(j)|k_s(j)}\right) \tag{2.34a}$$

$$\Sigma^x_{k_s(j)|k_s(j)} = \Sigma^{x^f}_{k_s(j)|k_s(j)} - \Sigma^{x^f y^m}_{k_s(j)|k_s(j)} \left(\Sigma^{y^m}_{k_s(j)|k_s(j)}\right)^{-1} \Sigma^{y^m x^f}_{k_s(j)|k_s(j)} \tag{2.34b}$$

Thus, the remaining task is to calculate the covariance $\Sigma^{x^f y^m}_{k_s(j)|k_s(j)}$ and the Gaussian approximation of $p\left(y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ in equation (2.33). Once the Gaussian approximation has been obtained, we can use the unscented transformation to calculate the covariance $\Sigma^{x^f y^m}_{k_s(j)|k_s(j)}$. Unless otherwise specified, this method is used to obtain covariances in the following derivation. Generally, $p\left(y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ can be derived as

$$p\left(y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) = \int p\left(y^m_{k_s(j)} \mid m_{s_j+l_j}\right) p\left(m_{s_j+l_j} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) dm_{s_j+l_j} \tag{2.35}$$

where the only unknown statistic is the smoothing density of integral term $m_{s_j+l_j}$. Just as in equation (2.5), since the relationship between $m_{s_j+l_j}$ and $X^l_s(j)$ is linear, we can obtain the smoothing density by calculating $p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$, which can be computed as

$$p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) = \int p\left(X^l_s(j) \mid x_{s_j+l_j+1}, y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right) p\left(x_{s_j+l_j+1} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) dx_{s_j+l_j+1} \tag{2.36}$$

By comparison, it is seen that equation (2.36) is very similar to equation (2.7), but easier to calculate. Using Proposition 2, the approximated moments of $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ are calculated as

$$\mu^{Xf}_{s_j|k_s(j)} = \mu^{Xf}_{s_j|s_j+l_j} + J^{Xf}_{s_j}\left(\mu^{xf}_{s_j+l_j+1|k_s(j)} - \mu^{xf}_{s_j+l_j+1|s_j+l_j}\right) \tag{2.37a}$$

$$\Sigma^{Xf}_{s_j|k_s(j)} = \Sigma^{Xf}_{s_j|s_j+l_j} + J^{Xf}_{s_j}\left(\Sigma^{xf}_{s_j+l_j+1|k_s(j)} - \Sigma^{xf}_{s_j+l_j+1|s_j+l_j}\right)\left(J^{Xf}_{s_j}\right)^T \tag{2.37b}$$

$$J^{Xf}_{s_j} = \Sigma^{Xfxf}_{s_j,s_j+l_j+1|s_j+l_j}\left(\Sigma^{xf}_{s_j+l_j+1|s_j+l_j}\right)^{-1} \tag{2.37c}$$

where $\mathcal{N}\left(\mu^{Xf}_{s_j|s_j+l_j}, \Sigma^{Xf}_{s_j|s_j+l_j}\right)$ is the Gaussian approximation of $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$, and $\mathcal{N}\left(\mu^{xf}_{s_j+l_j+1|k_s(j)}, \Sigma^{xf}_{s_j+l_j+1|k_s(j)}\right)$ is the Gaussian approximation of the smoothing density $p\left(x_{s_j+l_j+1} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ if $k_s(j) > s_j + l_j + 1$. Specifically, the smoothing density $p\left(x_{s_j+l_j+1} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$ can be calculated recursively. For $n = k_s(j), k_s(j)-1, \cdots, s_j + l_j + 2$

$$p\left(x_{n-1} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) = \int p\left(x_{n-1} \mid x_n, y_{1:s_j-1}, y^f_{s_j:n-1}\right) p\left(x_n \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right) dx_n \tag{2.38}$$

which can be described in terms of a Gaussian distribution by applying Proposition 2.

$$\mu^{xf}_{n-1|k_s(j)} = \mu^{xf}_{n-1|n-1} + J_{n-1}\left(\mu^{xf}_{n|k_s(j)} - \mu^{xf}_{n|n-1}\right) \tag{2.39a}$$

$$\Sigma^{xf}_{n-1|k_s(j)} = \Sigma^{xf}_{n-1|n-1} + J_{n-1}\left(\Sigma^{xf}_{n|k_s(j)} - \Sigma^{xf}_{n|n-1}\right) J^T_{n-1} \tag{2.39b}$$

$$J_{n-1} = \Sigma^{xf}_{n-1,n|n-1}\left(\Sigma^{xf}_{n|n-1}\right)^{-1} \tag{2.39c}$$

The final task is to calculate $p\left(X_s^l(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$. As defined in Section 2.2, $X_s^l(j) = \begin{bmatrix} x^T_{s_j} & x^T_{s_j+1} & \dots & x^T_{s_j+l_j}\end{bmatrix}^T$, thus

$$\mu^{Xf}_{s_j|s_j+l_j} = \left[\left(\mu^{xf}_{s_j|s_j+l_j}\right)^T \quad \left(\mu^{xf}_{s_j+1|s_j+l_j}\right)^T \quad \cdots \quad \left(\mu^{xf}_{s_j+l_j|s_j+l_j}\right)^T\right]^T \tag{2.40a}$$

$$\Sigma^{Xf}_{s_j|s_j+l_j} = \begin{bmatrix} \Sigma^{xf}_{s_j|s_j+l_j} & \Sigma^{xf}_{s_j,s_j+1|s_j+l_j} & \cdots & \Sigma^{xf}_{s_j,s_j+l_j|s_j+l_j} \\ \Sigma^{xf}_{s_j+1,s_j|s_j+l_j} & \Sigma^{xf}_{s_j+1|s_j+l_j} & \cdots & \Sigma^{xf}_{s_j+1,s_j+l_j|s_j+l_j} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma^{xf}_{s_j+l_j,s_j|s_j+l_j} & \Sigma^{xf}_{s_j+l_j,s_j+1|s_j+l_j} & \cdots & \Sigma^{xf}_{s_j+l_j|s_j+l_j} \end{bmatrix} \tag{2.40b}$$

These elements in equation (2.40b) can be computed as follows, for $n = s_j + l_j, s_j + l_j -$

$1, \cdots, s_j + 1$

$$\mu^{xf}_{n-1|s_j+l_j} = \mu^{xf}_{n-1|n-1} + J_{n-1}\left(\mu^{xf}_{n|s_j+l_j} - \mu^{xf}_{n|n-1}\right) \tag{2.41a}$$

$$\Sigma^{xf}_{n-1|s_j+l_j} = \Sigma^{xf}_{n-1|n-1} + J_{n-1}\left(\Sigma^{xf}_{n|s_j+l_j} - \Sigma^{xf}_{n|n-1}\right) J^T_{n-1} \tag{2.41b}$$

$$J_{n-1} = \Sigma^{xf}_{n-1,n|n-1}\left(\Sigma^{xf}_{n|n-1}\right)^{-1} \tag{2.41c}$$

This completes the UKF implementation. This method is referred to EB-UKF, which is shown in Algorithm 2. From the derivation above, it can be found that the additional statistics $\left\{\mathcal{N}\left(\mu^{xf}_{n|n}, \Sigma^{xf}_{n|n}\right)\right\}_{n=s_j}^{s_j+l_j}$ are needed to approximate $p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$, and the one-step-ahead predictive distributions $\left\{\mathcal{N}\left(\mu^{xf}_{n|n-1}, \Sigma^{xf}_{n|n-1}\right)\right\}_{n=s_j+1}^{k_s(j)}$ are also used. Although these predictions can be obtained directly from the filtering densities, for computing efficiency, we assume they are saved during previous filtering process.

2.4 Augmented state method

In order to apply this method, the original system model has to be reformulated. By reformulating the model, the state and integral term would be estimated simultaneously when the slow delayed measurement arrives, which means the target posterior density becomes $p\left(x_{k_s(j)}, m_{s_j+l_j} \mid y_{1:k_s(j)}\right)$. Obviously, adding the integral term $m_{s_j+l_j}$ has no impact for us to estimate $x_{k_s(j)}$. Besides, since we consider all information we have until time instant $k_s(j)$ just as in the exact Bayesian solution, these two methods would lead to a similar result.

2.4.1 Reformulation of the system model

The integral term $m_{s_j+l_j}$ can be calculated recursively as

$$\begin{aligned} m_{k+1} &= m_k + c_{k+1}x_{k+1} \\ &= m_k + c_{k+1} f\left(x_k, u_k\right) + c_{k+1}\omega_k \end{aligned} \tag{2.42}$$

for $k = s_j, s_j + 1, ..., s_j + l_j - 1$. The initial value of m_k in equation (2.42) is $m_{s_j} = c_{s_j} x_{s_j}$. Next, let us define an augmented state as,

$$x^a_k = \begin{cases} \left[x^T_k \quad m^T_k\right]^T & \text{for} \quad k = s_j, s_j + 1, ..., s_j + l_j \\ \left[x^T_k \quad m^T_{s_j+l_j}\right]^T & \text{for} \quad k = s_j + l_j + 1, ..., k_s(j) \end{cases} \tag{2.43}$$

One can see that the augmented state x^a_k only exists when $k = s_j, s_j + 1, ..., k_s(j)$, which is the period to collect integral samples and analyze slow measurements. For other

Algorithm 2 EB-UKF for estimating the posterior density $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$

```
Input: $\left\{\mathcal{N}\left(\mu^{x^f}_{n|n}, \Sigma^{x^f}_{n|n}\right)\right\}_{n=s_j}^{k_s(j)}$ and $\left\{\mathcal{N}\left(\mu^{x^f}_{n|n-1}, \Sigma^{x^f}_{n|n-1}\right)\right\}_{n=s_j+1}^{k_s(j)}$
Output: Gaussian approximation $\mathcal{N}\left(\mu^{x}_{k_s(j)|k_s(j)}, \Sigma^{x}_{k_s(j)|k_s(j)}\right) \approx p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$
%% Calculating $\mathcal{N}\left(\mu^{X^f}_{s_j|s_j+l_j}, \Sigma^{X^f}_{s_j|s_j+l_j}\right) \approx p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:s_j+l_j}\right)$
for $n = s_j + l_j : s_j + 1$ do
    $J_{n-1} = \Sigma^{x^f}_{n-1,n|n-1}\left(\Sigma^{x^f}_{n|n-1}\right)^{-1}$
    $\mu^{x^f}_{n-1|s_j+l_j} = \mu^{x^f}_{n-1|n-1} + J_{n-1}\left(\mu^{x^f}_{n|s_j+l_j} - \mu^{x^f}_{n|n-1}\right)$
    $\Sigma^{x^f}_{n-1|s_j+l_j} = \Sigma^{x^f}_{n-1|n-1} + J_{n-1}\left(\Sigma^{x^f}_{n|s_j+l_j} - \Sigma^{x^f}_{n|n-1}\right) J^T_{n-1}$
end for
$\mu^{X^f}_{s_j|s_j+l_j} = \left[\left(\mu^{x^f}_{s_j|s_j+l_j}\right)^T \quad \left(\mu^{x^f}_{s_j+1|s_j+l_j}\right)^T \quad \cdots \quad \left(\mu^{x^f}_{s_j+l_j|s_j+l_j}\right)^T\right]^T$
$\Sigma^{X^f}_{s_j|s_j+l_j} = \begin{bmatrix} \Sigma^{x^f}_{s_j|s_j+l_j} & \Sigma^{x^f}_{s_j,s_j+1|s_j+l_j} & \cdots & \Sigma^{x^f}_{s_j,s_j+l_j|s_j+l_j} \\ \Sigma^{x^f}_{s_j+1,s_j|s_j+l_j} & \Sigma^{x^f}_{s_j+1|s_j+l_j} & \cdots & \Sigma^{x^f}_{s_j+1,s_j+l_j|s_j+l_j} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma^{x^f}_{s_j+l_j,s_j|s_j+l_j} & \Sigma^{x^f}_{s_j+l_j,s_j+1|s_j+l_j} & \cdots & \Sigma^{x^f}_{s_j+l_j|s_j+l_j} \end{bmatrix}$
%% Calculating $\mathcal{N}\left(\mu^{x^f}_{s_j+l_j+1|k_s(j)}, \Sigma^{x^f}_{s_j+l_j+1|k_s(j)}\right) \approx p\left(x_{s_j+l_j+1} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$
if $k_s(j) = s_j + l_j + 1$ then
    $\mu^{x^f}_{s_j+l_j+1|k_s(j)} = \mu^{x^f}_{k_s(j)|k_s(j)}$
    $\Sigma^{x^f}_{s_j+l_j+1|k_s(j)} = \Sigma^{x^f}_{k_s(j)|k_s(j)}$
else
    for $n = k_s(j) : s_j + l_j + 2$ do
        $J_{n-1} = \Sigma^{x^f}_{n-1,n|n-1}\left(\Sigma^{x^f}_{n|n-1}\right)^{-1}$
        $\mu^{x^f}_{n-1|k_s(j)} = \mu^{x^f}_{n-1|n-1} + J_{n-1}\left(\mu^{x^f}_{n|k_s(j)} - \mu^{x^f}_{n|n-1}\right)$
        $\Sigma^{x^f}_{n-1|k_s(j)} = \Sigma^{x^f}_{n-1|n-1} + J_{n-1}\left(\Sigma^{x^f}_{n|k_s(j)} - \Sigma^{x^f}_{n|n-1}\right) J^T_{n-1}$
    end for
end if
%% Calculating $\mathcal{N}\left(\mu^{X^f}_{s_j|k_s(j)}, \Sigma^{X^f}_{s_j|k_s(j)}\right) \approx p\left(X^l_s(j) \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$
$J^{X^f}_{s_j} = \Sigma^{X^f x^f}_{s_j,s_j+l_j+1|s_j+l_j}\left(\Sigma^{x^f}_{s_j+l_j+1|s_j+l_j}\right)^{-1}$
$\mu^{X^f}_{s_j|k_s(j)} = \mu^{X^f}_{s_j|s_j+l_j} + J^{X^f}_{s_j}\left(\mu^{x^f}_{s_j+l_j+1|k_s(j)} - \mu^{x^f}_{s_j+l_j+1|s_j+l_j}\right)$
$\Sigma^{X^f}_{s_j|k_s(j)} = \Sigma^{X^f}_{s_j|s_j+l_j} + J^{X^f}_{s_j}\left(\Sigma^{x^f}_{s_j+l_j+1|k_s(j)} - \Sigma^{x^f}_{s_j+l_j+1|s_j+l_j}\right)\left(J^{X^f}_{s_j}\right)^T$
%% Calculating $\mathcal{N}\left(\mu^{y^m}_{k_s(j)|k_s(j)}, \Sigma^{y^m}_{k_s(j)|k_s(j)}\right) \approx p\left(y^m_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)}\right)$
Use the unscented transformation to get sigma points $\left\{\mathcal{X}^i_{s_j|k_s(j)}\right\}_{i=1}^{2n_X+1}$ of $\mathcal{N}\left(\mu^{X^f}_{s_j|k_s(j)}, \Sigma^{X^f}_{s_j|k_s(j)}\right)$
```

```
for $i = 1 : 2n_X + 1$ do
    $\mathcal{Y}^i_{k_s(j)|k_s(j)} = h^m\left(C^l_s(j)\mathcal{X}^i_{s_j|k_s(j)}\right)$
end for
$\mu^{y^m}_{k_s(j)|k_s(j)} = \sum_{i=1}^{2n_X+1} W^i_s \mathcal{Y}^i_{k_s(j)|k_s(j)}$
$\Sigma^{y^m}_{k_s(j)|k_s(j)} = \sum_{i=1}^{2n_X+1} W^i_c \left(\mathcal{Y}^i_{k_s(j)|k_s(j)} - \mu^{y^m}_{k_s(j)|k_s(j)}\right)(\cdots)^T + R^m$
%% Calculating output $\mathcal{N}\left(\mu^x_{k_s(j)|k_s(j)}, \Sigma^x_{k_s(j)|k_s(j)}\right) \approx p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$
$\mu^x_{k_s(j)|k_s(j)} = \mu^{x^f}_{k_s(j)|k_s(j)} + \Sigma^{x^f y^m}_{k_s(j)|k_s(j)}\left(\Sigma^{y^m}_{k_s(j)|k_s(j)}\right)^{-1}\left(y^m_{k_s(j)} - \mu^{y^m}_{k_s(j)|k_s(j)}\right)$
$\Sigma^x_{k_s(j)|k_s(j)} = \Sigma^{x^f}_{k_s(j)|k_s(j)} - \Sigma^{x^f y^m}_{k_s(j)|k_s(j)}\left(\Sigma^{y^m}_{k_s(j)|k_s(j)}\right)^{-1}\Sigma^{y^m x^f}_{k_s(j)|k_s(j)}$
```

time, there are normal states and fast measurements in the system. So as to estimate $p\left(x_{k_s(j)}, m_{s_j+l_j} \mid y_{1:k_s(j)}\right) = p\left(x^a_{k_s(j)} \mid y_{1:k_s(j)}\right)$, a new model of augmented state is defined for this special period.

The augmented state transition equation is, for $k = s_j, s_j+1, ..., k_s(j)-1$,

$$x^a_{k+1} = \begin{bmatrix} f\left(x_k, u_k\right) \\ m_k + g_{k+1} f\left(x_k, u_k\right) \end{bmatrix} + \begin{bmatrix} \omega_k \\ g_{k+1}\omega_k \end{bmatrix} \tag{2.44}$$

where the initial value of x^a_k is $x^a_{s_j} = \begin{bmatrix} x^T_{s_j} & m^T_{s_j} \end{bmatrix}^T$, the covariance of process noise in equation (2.44) is $Q^a_k = \begin{bmatrix} Q & \left(g_{k+1}Q\right)^T \\ g_{k+1}Q & g_{k+1}Qg^T_{k+1} \end{bmatrix}$, and

$$g_k = \begin{cases} c_k & \text{for} \quad k = s_j, s_j+1, ..., s_j+l_j \\ \mathbf{0} & \text{for} \quad k = s_j+l_j+1, ..., k_s(j) \end{cases}$$

For time from s_j to $k_s(j)-1$, the measurement equation is the same as equation (2.1b). At time $k_s(j)$, the measurement equation becomes

$$y^a_{k_s(j)} = h\left(x^a_{k_s(j)}\right) + v^a_{k_s(j)} \tag{2.45}$$

where

$$y^a_{k_s(j)} = \begin{bmatrix} y^f_{k_s(j)} \\ y^m_{k_s(j)} \end{bmatrix} \quad h\left(x^a_{k_s(j)}\right) = \begin{bmatrix} h^f\left(x_{k_s(j)}\right) \\ h^m\left(m_{s_j+l_j}\right) \end{bmatrix} \quad v^a_{k_s(j)} = \begin{bmatrix} v^f_{k_s(j)} \\ v^m_{k_s(j)} \end{bmatrix}$$

The variance of $v^a_{k_s(j)}$ is $R^a_{k_s} = \begin{bmatrix} R^f & \mathbf{0} \\ \mathbf{0} & R^m \end{bmatrix}$.

After reformulation, it is much easier to fuse two measurements according to Bayes' rule as

$$p\left(x^a_{k_s(j)} \mid y_{1:k_s(j)}\right) = \frac{p\left(y^a_{k_s(j)} \mid x^a_{k_s(j)}\right) p\left(x^a_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right)}{\int p\left(y^a_{k_s(j)} \mid x^a_{k_s(j)}\right) p\left(x^a_{k_s(j)} \mid y_{1:s_j-1}, y^f_{s_j:k_s(j)-1}\right) dx^a_{k_s(j)}} \tag{2.46}$$

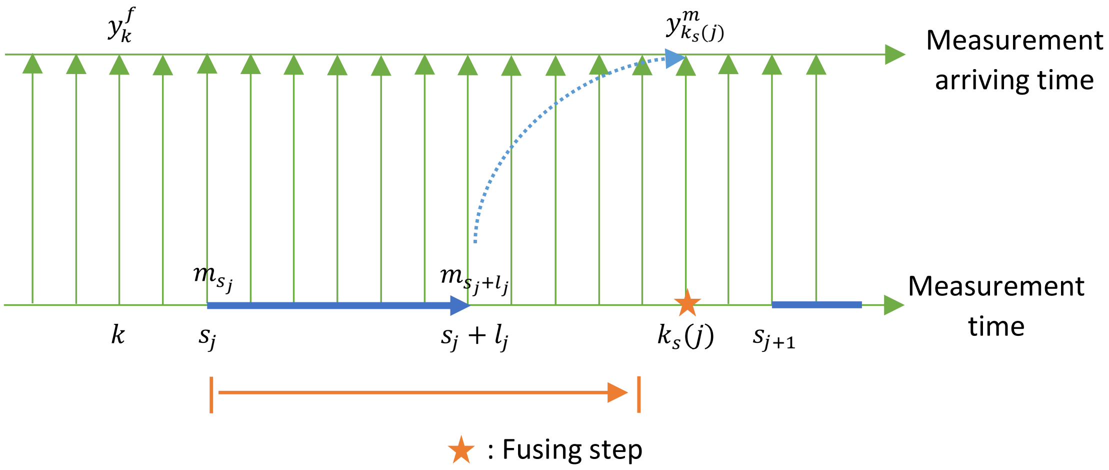

Figure 2.3: Schematic of the augmented state method

In equation (2.46), the first term in the numerator can be derived from equation (2.45), while the second term can be obtained through Bayes filter using the augmented state model. Figure 2.3 is the schematic of the augmented state method. Thus, the only required statistic is the initial value used in the Bayes filter, i.e. $p\left(x^a_{s_j} \mid y_{1:s_j-1}, y^f_{s_j}\right)$. In this section, two Bayes filters, the particle filter and the unscented Kalman filter are both applied just as described in Section 2.3.

2.4.2 Implementations of the PF and UKF

In the particle filter, since $x^a_{s_j} = \left[x^T_{s_j} \quad \left(c_{s_j} x_{s_j}\right)^T\right]^T$, the density $p\left(x^a_{s_j} \mid y_{1:s_j-1}, y^f_{s_j}\right)$ can be approximated by the particle set $\left\{x^{af(i)}_{s_j}, w^{af(i)}_{s_j}\right\}^N_{i=1}$, where $x^{af(i)}_{s_j} = \left[\left(x^{f(i)}_{s_j}\right)^T \quad \left(c_{s_j} x^{f(i)}_{s_j}\right)^T\right]^T$ and $w^{af(i)}_{s_j} = w^{f(i)}_{s_j}$. This method is named as AS-PF. The algorithm of AS-PF is described in Algorithm 3.

Before implementing the UKF, it should be noted that the covariance of $x^a_{s_j}$ is singular which can lead to many problems in calculation process. The solution is to use $p\big(x^a_{s_j+1} \mid y_{1:s_j-1}, y^f_{s_j:s_j+1}\big)$ as the initial value. Assume its approximation is $\mathcal{N}\big(\mu^{x^a}_{s_j+1|s_j+1}, \Sigma^{x^a}_{s_j+1|s_j+1}\big)$, where

$$\mu^{x^a}_{s_j+1|s_j+1} = \begin{bmatrix} \mu^{x^f}_{s_j+1|s_j+1} \\ \mu^{m^f}_{s_j+1|s_j+1} \end{bmatrix} \tag{2.47a}$$

$$\Sigma^{x^a}_{s_j+1|s_j+1} = \begin{bmatrix} \Sigma^{x^f}_{s_j+1|s_j+1} & \Sigma^{x^f m^f}_{s_j+1|s_j+1} \\ \Sigma^{m^f x^f}_{s_j+1|s_j+1} & \Sigma^{m^f}_{s_j+1|s_j+1} \end{bmatrix} \tag{2.47b}$$

Algorithm 3 AS-PF for estimating the posterior density $p\left(x_{k_s(j)} \mid y_{1:k_s(j)}\right)$

```
Input: ${\left\{x_{s_j}^{f(i)}, w_{s_j}^{f(i)}\right\}}_{i=1}^{N}$
Output: ${\left\{x_{k_s(j)}^{(i)}, w_{k_s(j)}^{(i)}\right\}}_{i=1}^{N}$
%% Calculating the particle set ${\left\{x_{s_j}^{af(i)}, w_{s_j}^{af(i)}\right\}}_{i=1}^{N}$
$m_{s_j}^{f(i)} = c_{s_j} x_{s_j}^{f(i)}$
$x_{s_j}^{af(i)} = \begin{bmatrix} x_{s_j}^{f(i)} \\ m_{s_j}^{f(i)} \end{bmatrix}$ and $w_{s_j}^{af(i)} = w_{s_j}^{f(i)}$
%% Calculating the particle set ${\left\{x_{k_s(j)-1}^{af(i)}, w_{k_s(j)-1}^{af(i)}\right\}}_{i=1}^{N}$
for $n = s_j, s_j + 1, ..., k_s(j) - 2$ do
    for $i = 1, 2, ..., N$ do
        $x_{n+1}^{f(i)} = f\left(x_n^{f(i)}, u_n\right) + \omega_n^{(i)}$
        $m_{n+1}^{f(i)} = m_n^{f(i)} + g_{n+1} x_{n+1}^{f(i)}$
        $x_{n+1}^{af(i)} = \begin{bmatrix} x_{n+1}^{f(i)} \\ m_{n+1}^{f(i)} \end{bmatrix}$
        $w_{n+1}^{af(i)} \propto w_n^{af(i)} p\left(y_{n+1}^{f} \mid x_{n+1}^{af(i)}\right)$
    end for
    if $\widehat{Neff} < N_T$ then
        Resample ${\left\{x_{n+1}^{af(i)}, w_{n+1}^{af(i)}\right\}}_{i=1}^{N}$
    end if
end for
%% Calculating the output particle set ${\left\{x_{k_s(j)}^{(i)}, w_{k_s(j)}^{(i)}\right\}}_{i=1}^{N}$
for $i = 1, 2, ..., N$ do
    $x_{k_s(j)}^{f(i)} = f\left(x_{k_s(j)-1}^{f(i)}, u_{k_s(j)-1}\right) + \omega_{k_s(j)-1}^{(i)}$
    $m_{k_s(j)}^{f(i)} = m_{k_s(j)-1}^{f(i)} + g_{k_s(j)} x_{k_s(j)}^{f(i)}$
    $x_{k_s(j)}^{af(i)} = \begin{bmatrix} x_{k_s(j)}^{f(i)} \\ m_{k_s(j)}^{f(i)} \end{bmatrix}$
    $w_{k_s(j)}^{af(i)} \propto w_{k_s(j)-1}^{af(i)} p\left(y_{k_s(j)}^{a} \mid x_{k_s(j)}^{af(i)}\right)$
    $x_{k_s(j)}^{(i)} = x_{k_s(j)}^{f(i)}$ and $w_{k_s(j)}^{(i)} = w_{k_s(j)}^{af(i)}$
end for
if $\widehat{Neff} < N_T$ then
    Resample ${\left\{x_{k_s(j)}^{(i)}, w_{k_s(j)}^{(i)}\right\}}_{i=1}^{N}$
end if
```

In equations (2.47a) and (2.47b), the unknown statistics $\mu^{mf}_{s_j+1|s_j+1}$, $\Sigma^{mf}_{s_j+1|s_j+1}$ and $\Sigma^{m^f x^f}_{s_j+1|s_j+1}$ are calculated as

$$\begin{aligned}
\mu^{mf}_{s_j+1|s_j+1} &= c_{s_j}\mu^{xf}_{s_j|s_j+1} + c_{s_j+1}\mu^{xf}_{s_j+1|s_j+1} \\
\Sigma^{mf}_{s_j+1|s_j+1} &= c_{s_j}\Sigma^{xf}_{s_j|s_j+1}c^T_{s_j} + c_{s_j+1}\Sigma^{xf}_{s_j+1|s_j+1}c^T_{s_j+1} + 2c_{s_j}\Sigma^{xf}_{s_j,s_j+1|s_j+1}c^T_{s_j+1} \\
\Sigma^{m^f x^f}_{s_j+1|s_j+1} &= c_{s_j}\Sigma^{xf}_{s_j,s_j+1|s_j+1} + c_{s_j+1}\Sigma^{xf}_{s_j+1|s_j+1}
\end{aligned}$$

where $\mathcal{N}\left(\mu^{xf}_{s_j|s_j+1}, \Sigma^{xf}_{s_j|s_j+1}\right)$ is the Gaussian approximation of $p\left(x_{s_j} \mid y_{1:s_j-1}, y^f_{s_j:s_j+1}\right)$, and $\Sigma^{xf}_{s_j,s_j+1|s_j+1}$ is the covariance of x_{s_j} and x_{s_j+1} given $\left\{y_{1:s_j-1}, y^f_{s_j:s_j+1}\right\}$.

$$p\left(x_{s_j} \mid y_{1:s_j-1}, y^f_{s_j:s_j+1}\right) = \int p\left(x_{s_j} \mid x_{s_j+1}, y_{1:s_j-1}, y^f_{s_j}\right) p\left(x_{s_j+1} \mid y_{1:s_j-1}, y^f_{s_j:s_j+1}\right) dx_{s_j+1} \tag{2.48}$$

Thus, $\mathcal{N}\left(\mu^{xf}_{s_j|s_j+1}, \Sigma^{xf}_{s_j|s_j+1}\right)$ can be calculated using Proposition 2.

$$\mu^{xf}_{s_j|s_j+1} = \mu^{xf}_{s_j|s_j} + J_{s_j}\left(\mu^{xf}_{s_j+1|s_j+1} - \mu^{xf}_{s_j+1|s_j}\right) \tag{2.49a}$$

$$\Sigma^{xf}_{s_j|s_j+1} = \Sigma^{xf}_{s_j|s_j} + J_{s_j}\left(\Sigma^{xf}_{s_j+1|s_j+1} - \Sigma^{xf}_{s_j+1|s_j}\right)J^T_{s_j} \tag{2.49b}$$

$$J_{s_j} = \Sigma^{xf}_{s_j,s_j+1|s_j}\left(\Sigma^{xf}_{s_j+1|s_j}\right)^{-1} \tag{2.49c}$$

This method is named as AS-UKF.

2.5 Simulation and experimental evaluation

In this section, the proposed methods are evaluated through simulation and experimental studies. In the first example, two cases are tested to see the impact of increasing integral time l_j and delayed time d_j. In the second example, the proposed methods are experimentally studied on a hybrid tank.

2.5.1 Simulation study

The first example is based on the model of a continuous fermenter [30], [32], [35]. The process consists of a constant volume reactor in which a single, rate limiting substrate promotes biomass growth and product formation. The process model is

$$\dot{X} = -DX + \mu(S,P)X \tag{2.50a}$$

$$\dot{S} = D(S_f - S) - \frac{1}{Y_{X/S}}\mu(S,P)X \tag{2.50b}$$

$$\dot{P} = -DP + [\alpha\mu(S,P) + \beta]X \tag{2.50c}$$

where X, S and P are the biomass, substrate, and product concentration, respectively; D is the dilution rate; S_f is the feed substrate concentration; and $Y_{X/S}$, α and β are yield parameters. The specific growth rate μ is modeled as

$$\mu(S,P) = \frac{\mu_m\left(1 - \frac{P}{P_m}\right)S}{K_m + S + \frac{S^2}{K_i}} \tag{2.51}$$

where μ_m is the maximum specific growth rate; P_m, K_m and K_i are constant parameters. Nominal operating conditions are shown in Table 2.1. The state and fast measurement vectors are defined as

$$x = \begin{bmatrix} X & S & P \end{bmatrix}^T \qquad y^f = P \tag{2.52}$$

Table 2.1: Nominal operating conditions for the fermenter model [32]

Variable	Value	Variable	Value
$Y_{X/S}$	0.4 g/g	α	2.2 g/g
β	0.2 h^{-1}	μ_m	0.48 h^{-1}
P_m	50 g/L	K_m	1.2 g/L
K_i	22 g/L	S_f	20 g/L
D	0.202 h^{-1}	X	6.0 g/L
S	5.0 g/L	P	19.14 g/L

The system is discretized with a sampling interval of $\Delta t = 1/60$ h $= 1$ min, and simulated for 700 time steps from the initial condition $x_0 = \begin{bmatrix} 6 & 5 & 19.4 \end{bmatrix}^T$. The covariances of states and fast measurements are $Q = 0.01 I_3$ and $R^f = 1$, respectively. The integral term $m_{s_j+l_j}$ is defined as

$$m_{s_j+l_j} = \frac{1}{l_j}\sum_{i=s_j}^{s_j+l_j} x_i \tag{2.53}$$

The slow measurement equation is

$$y^m_{k_s(j)} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} m_{s_j+l_j} + v^m_{k_s(j)} \tag{2.54}$$

where the covariance of $v^m_{k_s(j)}$ is $R^m = 10^{-6}$. Simulations start from a initial guess $\hat{x}_0 = \begin{bmatrix} 6.03 & 4.98 & 19 \end{bmatrix}$ with the covariance $P_0 = diag\{0.09, 0.06, 0.9\}$. Because the results are influenced by both integral time l_j and delayed time d_j, simulations are conducted by considering these two factors separately. In the first case, different values of l_j are tested with delayed time $d_j = 0$ and the time between two slow measurements $\rho_j = l_j + 1$. For the second case, the integral time is set to be a constant $l_j = 1$, and different values of delayed

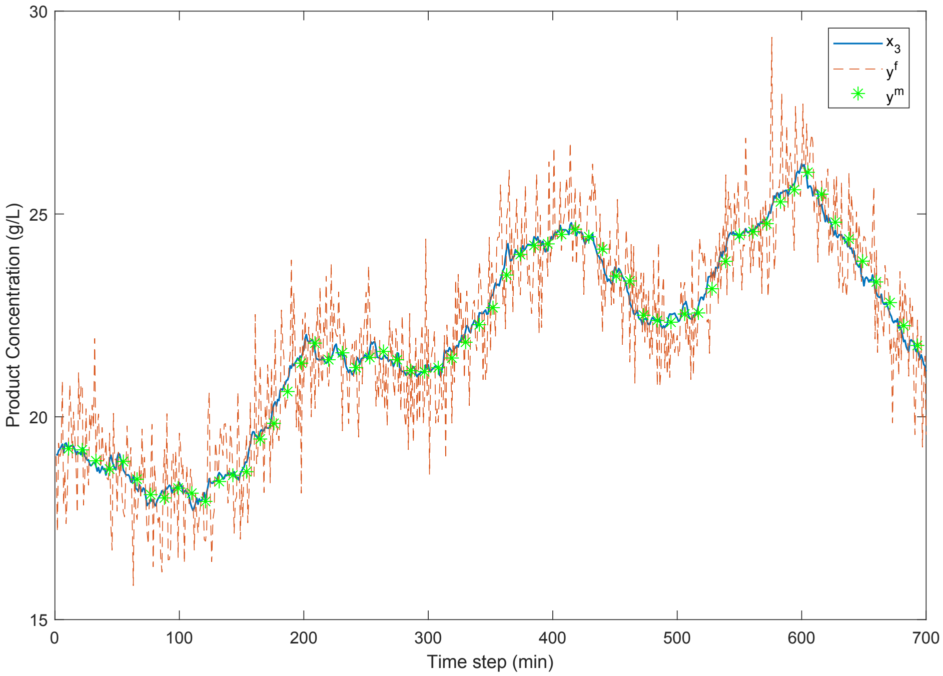

Figure 2.4: True product concentration x_3, fast measurement y^f and integrated laboratory measurement y^m with $l_j = 10, d_j = 0$

time d_j are simulated with $\rho_j = l_j + d_j + 1 = d_j + 2$. Although all three states can be estimated, we compare the estimation of product concentration P, since only P is measured. For comparison, the Average Root Mean Squared Error (ARMSE) is used, which is defined as

$$ARMSE = \frac{1}{M}\sum_{n=1}^{M}\sqrt{\frac{1}{T}\sum_{k=1}^{T}\left(e_k^{(n)}\right)^2} \tag{2.55}$$

where $e_k^{(n)}$ is the estimation error of product concentration P at k time instant for the n^{th} simulation run, T is the simulation horizon which is 700 in our case, and M is the number of simulation runs. The simulation results of two cases are shown in Table 2.2 and 2.3, respectively, where each result comes from a Monte Carlo simulation with 100 runs. The ARMSE of the PF and UKF that only use fast measurements are 0.3280 and 0.3189, respectively. Figure 2.4 is a plot of the state P, its fast measurements with high noise, and integrated no-delay laboratory measurements with integral time $l_j = 10$.

As can be seen from Table 2.2, with the integral time l_j increasing, the estimation errors of all algorithms become larger. Besides, no matter the value of l_j, they perform better than

Table 2.2: Estimation errors (ARMSE) of different algorithms with $d_j = 0$, $\hat{R}^m = 10^{-6}$

Algorithm	EB-PF	AS-PF	EB-UKF	AS-UKF
$l_j = 1$	0.1146	0.1168	0.0860	0.0860
$l_j = 2$	0.1587	0.1611	0.1206	0.1210
$l_j = 3$	0.1986	0.1971	0.1544	0.1549
$l_j = 4$	0.2162	0.2172	0.1684	0.1685
$l_j = 5$	0.2447	0.2444	0.1918	0.1919
$l_j = 6$	0.2612	0.2670	0.2058	0.2067
$l_j = 7$	0.2819	0.2817	0.2192	0.2201
$l_j = 8$	0.2959	0.2966	0.2333	0.2328
$l_j = 9$	0.3066	0.3034	0.2434	0.2426
$l_j = 10$	0.3143	0.3162	0.2563	0.2547

Table 2.3: Estimation errors (ARMSE) of different algorithms with $l_j = 1$, $\hat{R}^m = 10^{-6}$

Algorithm	EB-PF	AS-PF	EB-UKF	AS-UKF
$d_j = 1$	0.1683	0.2113	0.1468	0.1472
$d_j = 2$	0.2119	0.2771	0.1819	0.1823
$d_j = 3$	0.2442	0.3234	0.2199	0.2192
$d_j = 4$	0.2604	0.3334	0.2269	0.2273
$d_j = 5$	0.2791	0.3544	0.2563	0.2563
$d_j = 6$	0.2949	0.3647	0.2683	0.2644

the PF and UKF that only use fast measurements. Figure 2.5 shows that the differences in UKF-based methods and PF-based methods are very small. That is because when $d_j = 0$, the exact Bayesian algorithm and the augmented state algorithm are actually the same. Figure 2.6 illustrates that the estimated variance of x_3 decreases once the laboratory measurement arrives. Although the variance goes back to the normal value gradually as there are only fast measurements, it is still smaller than the minimum variance of the UKF. Therefore, fusing the laboratory measurements can improve the accuracy of estimation.

From Table 2.3, we can see that the estimation errors of all algorithms increase with the delayed time d_j becoming large. The UKF-based methods result in similar performance, while the PF-based methods have large differences. Comparing within Table 2.2 and 2.3, we find that the UKF-based methods always result in more accurate estimation than the PF-based methods. By looking into the estimation process, we can find the reason is sample impoverishment in the PF [4]. Because the noise of the slow laboratory measurements is so small, there are very few particles left after the resampling step, which leads to reduction of particle diversity and hence poor estimation. Furthermore, the large delay time d_j aggravates

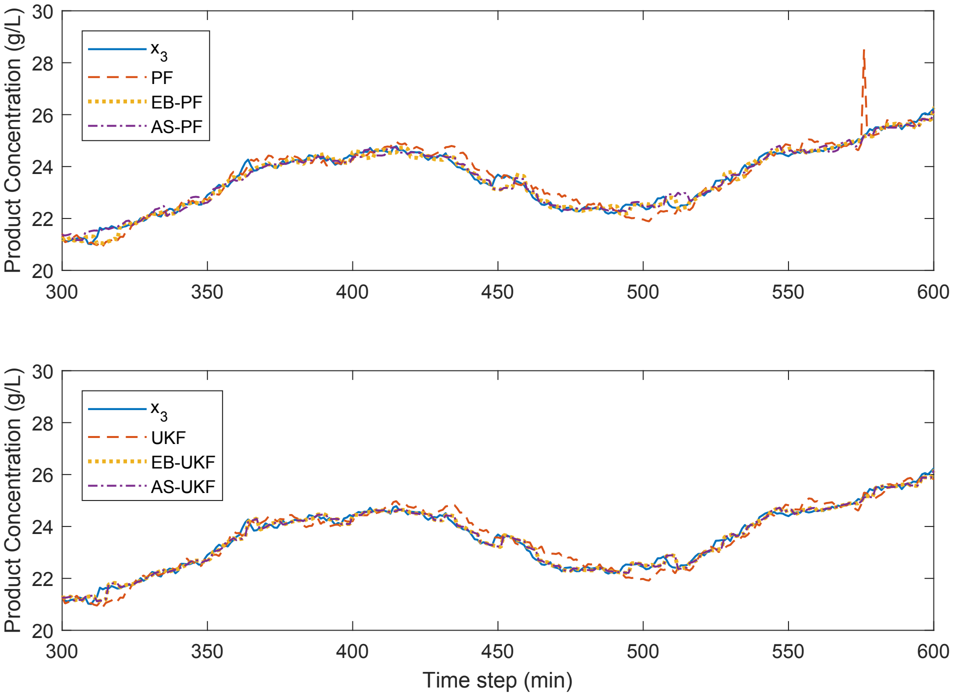

Figure 2.5: Simulation results of PF-based approaches and UKF-based approaches with $l_j = 3, d_j = 0$. Top: Results of PF-based approaches with PF only using fast measurements. Bottom: Results of UKF-based approaches with UKF only using fast measurements.

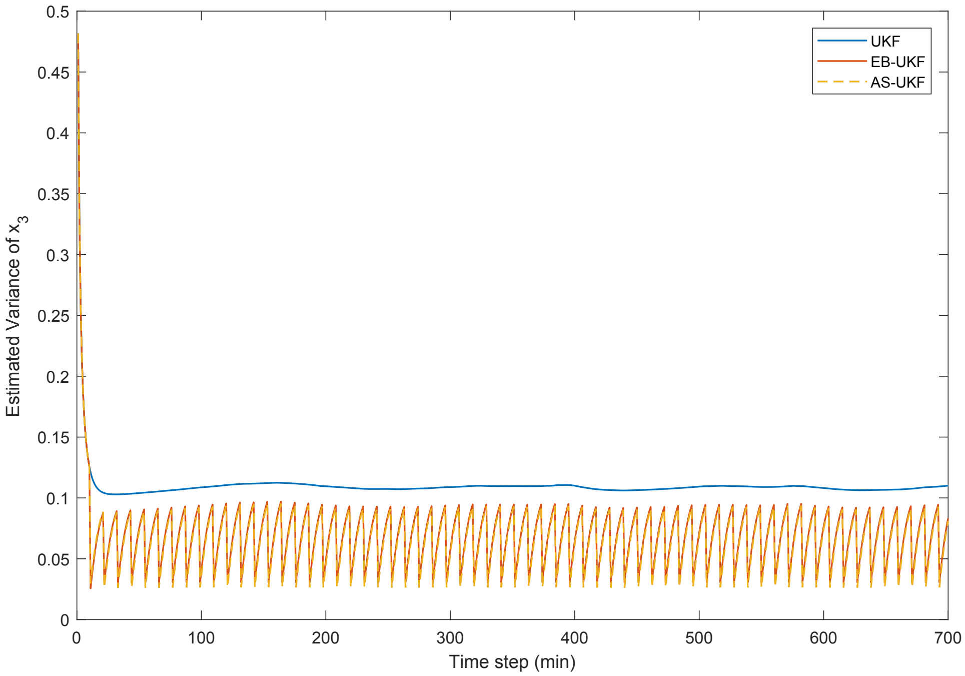

Figure 2.6: Estimated variance of x_3 during a simulation process with $l_j = 10, d_j = 0$

the sample impoverishment problem. But since EB-PF has a smoothing step, the sample impoverishment is relieved a little, which makes it perform better than AS-PF (see in Table 2.3).

A possible solution for the sample impoverishment problem is to use a larger number of particles, but this can greatly increase the computation load. So another solution is used in this section which is to enlarge the estimated noise covariance $\hat{R}^m$. As shown in Table 2.4 and 2.5, increasing $\hat{R}^m$ relieves the sample impoverishment problem and decreases the estimation errors. Figure 2.7 shows the error bounds for different PF-based algorithms. The error bounds of the proposed methods are smaller than PF method.

2.5.2 Experimental evaluation on a hybrid tank

The proposed algorithms are experimentally evaluated on a hybrid tank at the University of Alberta. Figure 2.8 shows the experimental setup. There are three tanks in this system. However, in this experiment, only the left and middle tanks are used. The schematic of the experiment system is shown in Figure 2.9. Water is pumped into the left tank, and can flow to the middle tank through the valve V_1. The water levels of both tanks are measured using

Table 2.4: Estimation errors (ARMSE) of PF-based algorithms with different $\hat{R}^m$ and $d_j = 0$

Algorithm	$\hat{R}^m = 10^{-4}$		$\hat{R}^m = 10^{-2}$	
	EB-PF	AS-PF	EB-PF	AS-PF
$l_j = 1$	0.0904	0.0907	0.0997	0.1002
$l_j = 2$	0.1257	0.1269	0.1335	0.1329
$l_j = 3$	0.1619	0.1616	0.1616	0.1630
$l_j = 4$	0.1806	0.1854	0.1845	0.1841
$l_j = 5$	0.2042	0.2056	0.2021	0.2028
$l_j = 6$	0.2246	0.2228	0.2191	0.2195
$l_j = 7$	0.2327	0.2340	0.2282	0.2303
$l_j = 8$	0.2494	0.2509	0.2424	0.2430
$l_j = 9$	0.2657	0.2641	0.2558	0.2560
$l_j = 10$	0.2840	0.2808	0.2692	0.2666

Table 2.5: Estimation errors (ARMSE) of PF-based algorithms with different $\hat{R}^m$ and $l_j = 1$

Algorithm	$\hat{R}^m = 10^{-4}$		$\hat{R}^m = 10^{-2}$	
	EB-PF	AS-PF	EB-PF	AS-PF
$d_j = 1$	0.1531	0.1564	0.1563	0.1560
$d_j = 2$	0.1912	0.1979	0.1905	0.1896
$d_j = 3$	0.2345	0.2488	0.2322	0.2314
$d_j = 4$	0.2437	0.2622	0.2418	0.2432
$d_j = 5$	0.2729	0.2978	0.2681	0.2691
$d_j = 6$	0.2813	0.3139	0.2763	0.2779

differential pressure (DP) level sensors.

The nonlinear state space model of the experiment plant derived using first principle modelling is as follows [19]–[21]:

$$\dot{h}_l = \frac{-k_1}{s_l}\sqrt{h_l - h_m} + \frac{1}{s_l}q_i \tag{2.56a}$$

$$\dot{h}_m = \frac{k_1}{s_m}\sqrt{h_l - h_m} + \frac{-k_2}{s_m}\sqrt{h_m} \tag{2.56b}$$

where h_l and h_m are the water level of the left and middle tank, respectively, q_i is the flow of the pump, s_l and s_m are the cross-sectional area of the tanks, which are equal to 243.22 cm^2, and $k_1 = 25.62, k_2 = 13.47$ are the coefficients of valves V_1 and V_2.

In the experiment, an RBS signal input around the operating point is used as the flow of the pump. To test the proposed approaches, the nonlinear state space model is discretized with a sampling time $\Delta t = 1$s. The system state is defined as $x = [h_l \quad h_m]^T$ and the fast measurement is middle tank level $y^f = [0 \quad 1]\, x$. It is assumed that process noise ω_k and

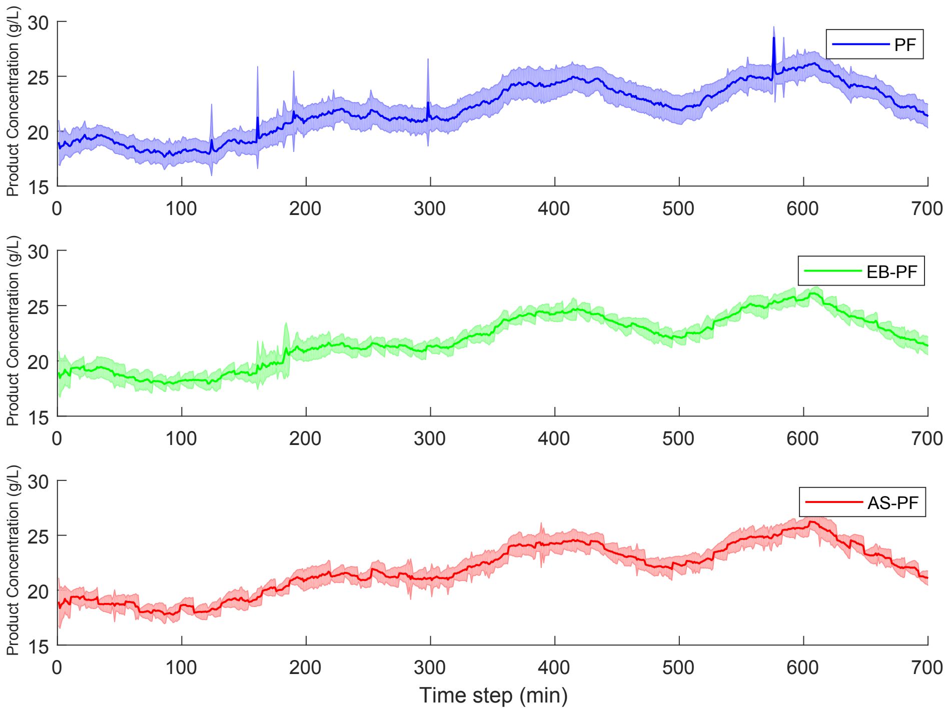

Figure 2.7: Error bounds of PF-based methods with $l_j = 10, d_j = 0$

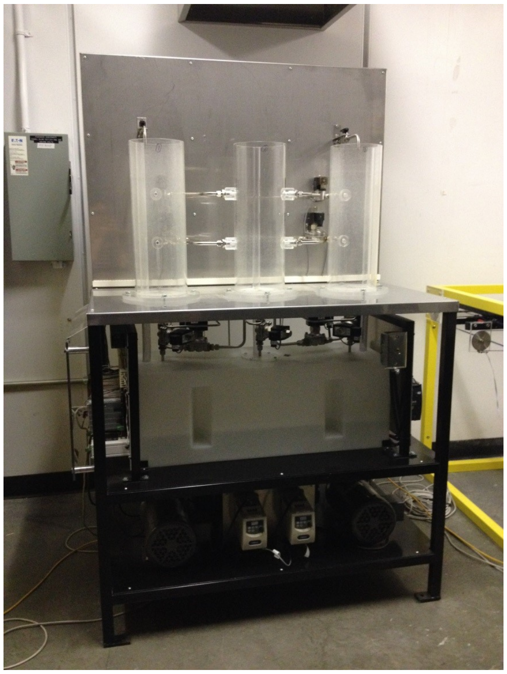

Figure 2.8: Picture of the hybrid tank

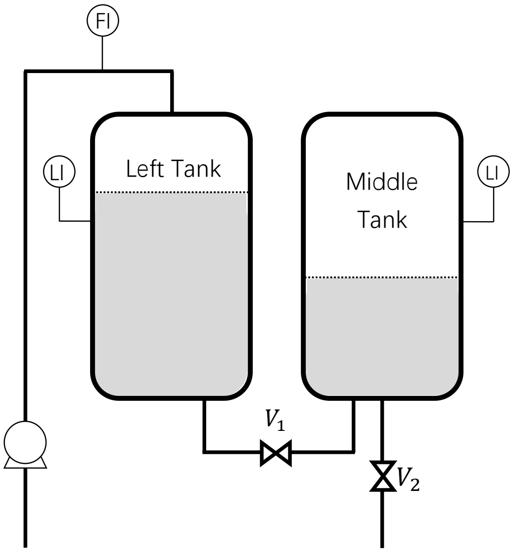

Figure 2.9: Schematic of the experiment plant

fast measurement noise v_k^f can be approximated by zero-mean Gaussian distributions. The covariance matrix of process noise ω_k is estimated as follows, using experiment data:

$$Q = 10^{-4} \times \begin{bmatrix} 0.1486 & -0.0427 \\ -0.0427 & 0.1556 \end{bmatrix} \tag{2.57}$$

The covariance of the fast measurement noise v_k^f is $R^f = 5.0 \times 10^{-3}$, and the initial state is $x_0 = [44.04 \quad 34.58]^T$. The fast measurement data used to conduct state estimation is the middle tank DP sensor output, with artificially added Gaussian noise whose covariance is equal to R^f.

In this example, we conduct two experiments. In the first one, the integrated laboratory measurement is the integration of the left tank level, and in the second experiment, the integrated laboratory measurement is the integration of the middle tank level [21]. Therefore, the slow delayed laboratory measurement is

$$y_{k_s(j)}^m = C_m m_{s_j+l_j} + v_{k_s(j)}^m \tag{2.58}$$

where C_m varies depending on the experiment, $v_{k_s(j)}^m$ is the measurement noise which follows a Gaussian distribution $\mathcal{N}(0, R^m)$ and the covariance R^m according to the variances of DP sensors is

$$R^m = \begin{cases} 6.33 \times 10^{-7} & \text{for the first experiment} \\ 2.42 \times 10^{-7} & \text{for the second experiment} \end{cases}$$

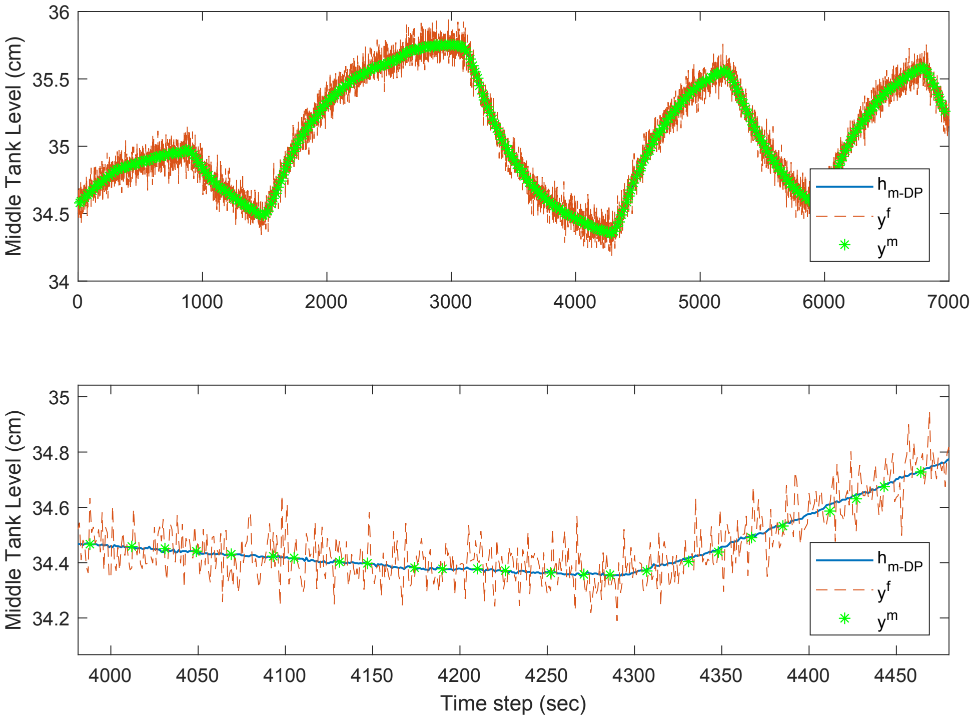

Figure 2.10: Output of middle tank DP level sensor, together with noise contaminated data y^f and slow delayed laboratory measurement y^m. Top: The whole data. Bottom: Zoomed-in-view of data

and the integral term $m_{s_j+l_j}$ is defined as

$$m_{s_j+l_j} = \frac{1}{l_j+1} \sum_{i=s_j}^{s_j+l_j} x_i \tag{2.59}$$

Specifically, the sampling interval ρ_j is determined as 20 seconds, the collection time of samples l_j randomly varies between 1 and 10 seconds following a uniform distribution, and the analysis time (i.e. time delay) d_j also randomly varies between 0 and 5 seconds following a uniform distribution. To construct the slow delayed laboratory measurement, the values of DP sensors are integrated over $l_j + 1$ sampling instants. Figure 2.10 shows the output of DP level sensor of the middle tank, together with fast measurements y^f and slow delayed laboratory measurements y^m in the second experiment.

Table 2.6: Estimation errors (ARMSE) of different algorithms using the integrated measurement of the left tank (Experiment 1)

Algorithm	EB-PF	AS-PF	JFM-PF	EB-UKF	AS-UKF	JFM-UKF
Left tank	0.0650	0.0621	0.1302	0.0361	0.0362	0.1301
Middle tank	0.0325	0.0315	0.0132	0.0311	0.0310	0.0118

Table 2.7: Estimation errors (ARMSE) of different algorithms using the integrated measurement of the middle tank (Experiment 2)

Algorithm	EB-PF	AS-PF	JFM-PF	EB-UKF	AS-UKF	JFM-UKF
Left tank	0.1304	0.1308	0.1302	0.1296	0.1295	0.1301
Middle tank	0.0094	0.0093	0.0132	0.0069	0.0070	0.0118

Tables 2.6 and 2.7 present the estimation errors in each experiment. Figures 2.11 and 2.12 show the estimates of left and middle tank levels in two experiments. The algorithms JFM-PF and JFM-UKF are PF and UKF approaches that only use fast measurements. The difference between the results of EB-PF and AS-PF is very small, as is the difference between EB-UKF and AS-UKF. A comparison of the results reveals that including the slow delayed measurement can improve the estimation of the corresponding variable that is measured directly. In the first experiment, as the slow laboratory measurement is the integrated left tank level, the estimate for the left tank level is much more accurate than that obtained just using fast measurements. For the same reason, in the second experiment, the estimation error of the middle tank level is much smaller. However, the accuracy for the variable which

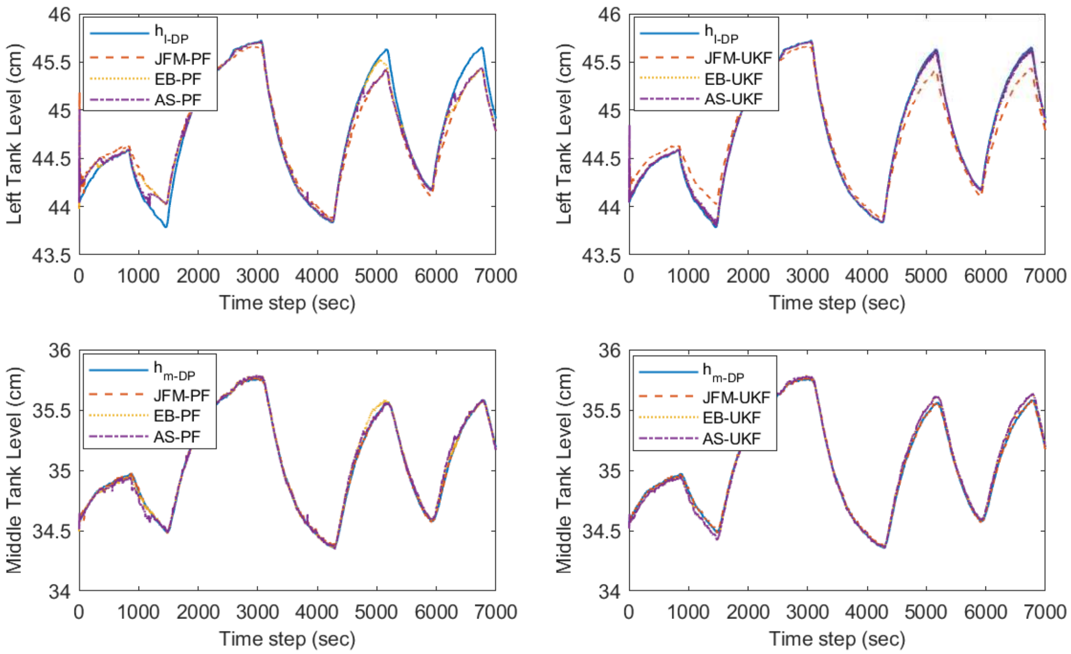

Figure 2.11: Estimates of the left and middle tank levels in the first experiment

is not directly measured does not actually improve, especially in the first experiment, the estimation error of the middle tank level increases. From Table 2.6 and Figure 2.11, we can see the estimation error of the left tank level is much larger than that of the middle tank level when only fast measurements are used, which means the model mismatch of the left tank is significant. Due to this reason, improving the estimate for the left tank level results in bad estimation for the middle tank level.

Although the above results show that the exact Bayesian method and augmented state method lead to about the same result, it is important to notice that the exact Bayesian method has a smoothing step and needs to store the state trajectory before the slow delayed laboratory measurement arrives. This makes the exact Bayesian method have a heavy computation load, especially for the PF-based algorithm, which is not good for online implementation. On the contrary, the augmented state method is not only easy to understand, but also fast in realization and computation.

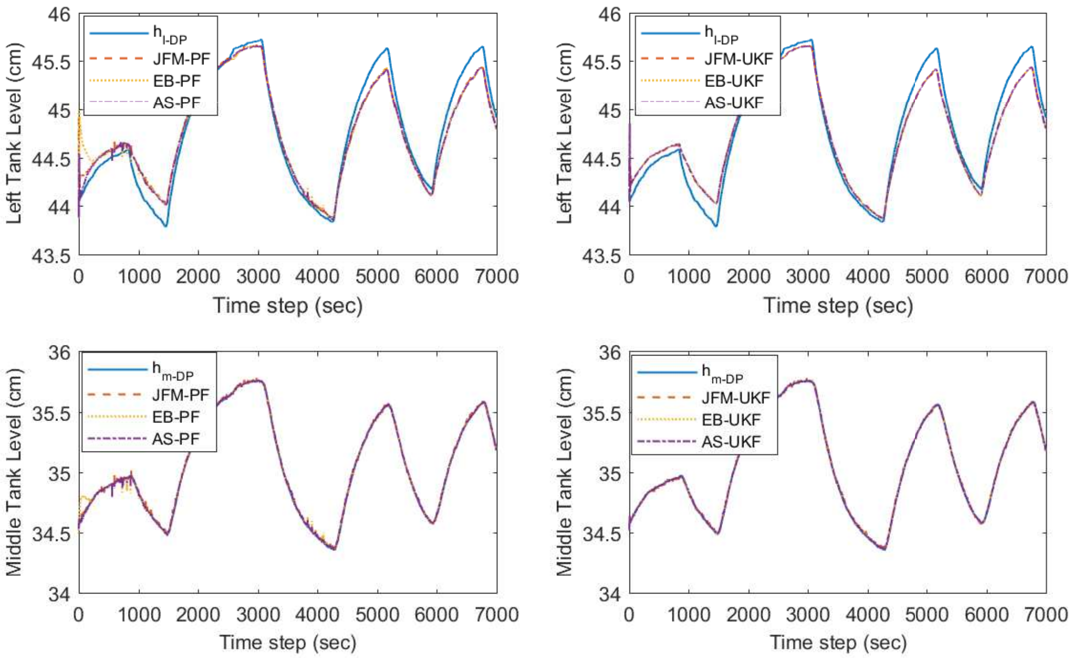

Figure 2.12: Estimates of the left and middle tank levels in the second experiment

2.6 Conclusions

The problem of state estimation for multirate measurements in the presence of integral term and variable measurement delay is studied in this chapter. Two methods are proposed: one is the exact Bayesian approach, and the other is the augmented state method. Both methods are implemented on particle filter and unscented Kalman filter. By simulation and experimental evaluation, we can find these two methods result in similar performances. Fusing the slow delayed integrated laboratory measurements with fast online measurements can give us more accurate estimates than fast state estimation. Since the laboratory measurement noise is very small, the PF-based algorithms have the sample impoverishment problem which can be relieved by increasing the estimated noise covariance artificially.

Chapter 3

Parameter estimation for nonlinear system with multirate measurements and random delays

3.1 Introduction

In the previous chapter, we studied the state estimation problem for multirate nonlinear system with integral term and random delays. In this chapter, we will focus on the parameter estimation problem for the same system.

Expectation maximization (EM) is a standard algorithm for parameter estimation in state space models [42]. There are two steps in the EM algorithm, the expectation step and the maximization step [12]. For linear systems with Gaussian noise, these two steps can be solved analytically, and explicit equations for parameter estimation can be obtained [37], [41]. However, for most nonlinear state space models with Gaussian or non-Gaussian noise, the expectation and maximization cannot be performed explicitly. A number of approximations have been proposed in the literature [28]. For instance, Monte Carlo sampling techniques have been utilized to perform the expectation step [2], [44]. Ghahramani and Roweis [24] used extended Kalman smoothing to estimate the states in the expectation step. In [25], a Taylor's series expansion of the process around a maximum a posteriori estimate of the state is used. Gopaluni [28] used a particle smoother to approximate the smoothing densities. In [13], [14], the smoothing densities in the expectation step are reduced to filtering densities using the particle filter for decreasing the computation cost.

The identification for nonlinear and multirate systems with single measurements and random delays is well studied in literature. In [51], Xie *et al.* proposed an EM-based algorithm to estimate parameters along with the time delays for FIR models by considering the time delays as hidden latent variables. Zhao *et al.* [53] provided a variational Bayesian (VB) approach for ARX models with time varying time delays. In [8], Chen *et al.* extended the algorithm in [53] when the interval of varying time delays is unknown. The problem discussed in this chapter is however more complicated and challenging. The objective is to estimate parameters and delays for nonlinear state space models with regular fast-rate measurements and slow delayed laboratory measurements. By fusing the slow delayed laboratory measurement with the fast-rate measurements, the estimated parameters are supposed to be more accurate than just using fast-rate measurements.

The remainder of this chapter is organized as follows. Section 3.2 states the parameter estimation problem of multirate systems in the presence of uncertain random delays. The next section applies the EM algorithm to solve this as well as a traditional single-rate identification problem as a comparison. Section 3.4 provides the particle filter implementation of the proposed algorithm. Two simulation examples along with a hybrid tank experiment are presented in Section 3.5, followed by the conclusion.

3.2 Problem statement

Consider the following discrete time system:

$$x_k = f\left(x_{k-1}, u_{k-1}, \theta^x\right) + \omega_{k-1} \tag{3.1a}$$

$$y_k^f = h^f\left(x_k, \theta^f\right) + v_k^f \tag{3.1b}$$

$$y_{k_s(j)}^m = h^m\left(m_{s_j+l_j}, \theta^m\right) + v_{k_s(j)}^m \tag{3.1c}$$

where $x_k \in \mathbb{R}^{n_x}$ is the system state, and the initial state x_0 follows the distribution $p\left(x_0\right)$; $u_k \in \mathbb{R}^{n_u}$ is the input vector which is known (for the sake of simplicity in presentation, it is omitted in the following derivation); $y_k^f \in \mathbb{R}^{n_y^f}$ is the fast and regular measurement and $y_{k_s(j)}^m \in \mathbb{R}^{n_y^m}$ $(j \in \mathbb{Z}_+)$ is the slow and irregular laboratory measurement; and f, h^f and h^m are nonlinear functions with parameters θ^x, θ^f and θ^m, respectively. The noise terms $\omega_k \in \mathbb{R}^{n_x}$, $v_k^f \in \mathbb{R}^{n_y^f}$ and $v_{k_s(j)}^m \in \mathbb{R}^{n_y^m}$ are *i.i.d.* Gaussian with zero mean and covariance matrices Q, R^f and R^m, respectively. $m_{s_j+l_j} \in \mathbb{R}^{n_m}$ is the integral term which represents integral of the samples collected from time instant s_j to $s_j + l_j$ ($l_j \in \mathbb{Z}$), and it is defined as

$$m_{s_j+l_j} = \frac{1}{l_j+1} \sum_{i=s_j}^{s_j+l_j} x_i \tag{3.2}$$

As the slow irregular laboratory measurement $y^m_{k_s(j)}$ is usually delayed but much more accurate than the fast measurement, the measurement arrives at time step $k_s(j) = s_j + l_j + d_j$, where $d_j \in \mathbb{Z}$ is the delayed time, and the noise covariance R^m is much smaller than R^f. The delayed time d_j is assumed to follow a uniform distribution between 0 and M, i.e.,

$$p\left(d_j = tr\right) = \frac{1}{M+1} \qquad \text{for} \quad t = 0, 1, \cdots, M \tag{3.3}$$

where $r \in \mathbb{Z}_+$ is the delay ratio. The sampling interval of laboratory measurements is ρ_j with $\rho_j \geqslant l_j + d_j + 1$, so $s_{j+1} = s_j + \rho_j$. The whole system can be represented as shown in Figure 2.1.

In this chapter, we assume that all parameters are unknown except the integral time l_j, as it can be measured directly during the sample collection interval. Therefore, the parameters that need to be estimated are system parameters θ^x, θ^f and θ^m, the noise covariances Q, R^f and R^m, and the delay time $\{d_1, d_2, \cdots, d_{N_m}\}$, where N_m is the number of slow sampled and delayed laboratory measurements.

3.3 Formulation of parameter estimation based on EM algorithm

In this section, two cases are considered. The first case is to estimate parameters θ^x and θ^f using only fast measurements. In the second case, both fast measurements and slow sampled and delayed laboratory measurements are used to estimate θ^x, θ^f and θ^m. The reason to include the first case is to provide a comparison of the accuracy of estimated θ^x between the two cases. The EM algorithm used in this chapter is briefly introduced in Section 1.4.1.

3.3.1 Using only fast measurements

In this case, the observed output data Y_{obs} are $Y^f = \left\{y^f_1, y^f_2, \cdots, y^f_T\right\}$, while the hidden states $X = \{x_0, x_1, \cdots, x_T\}$ can be viewed as the latent data in the EM algorithm. Let $\Theta^f = \left\{\theta^x, \theta^f, Q, R^f\right\}$ represent the overall parameters to be identified. Then, the Q function is

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) &= E_{X \mid Y^f, \Theta_n^f}\left\{\ln\left[p\left(Y^f, X \mid \Theta^f\right)\right]\right\} \\ &= \int \ln\left[p\left(Y^f, X \mid \Theta^f\right)\right] p\left(X \mid Y^f, \Theta_n^f\right) dX \\ &= \int \ln\left[p\left(x_{0:T}, y_{1:T}^f \mid \Theta^f\right)\right] p\left(x_{0:T} \mid y_{1:T}^f, \Theta_n^f\right) dx_{0:T} \end{aligned} \tag{3.4}$$

where Θ_n^f is the estimate of Θ^f after n^{th} iteration.

In equation (3.4), the first term on the right hand side can be derived as follows, according to the hidden Markov property:

$$p\left(x_{0:T}, y_{1:T}^f \mid \Theta^f\right) = p\left(x_0\right) \prod_{k=1}^{T} p\left(x_k \mid x_{k-1}, \Theta^f\right) \prod_{k=1}^{T} p\left(y_k^f \mid x_k, \Theta^f\right) \tag{3.5}$$

where $p\left(x_k \mid x_{k-1}, \Theta^f\right)$ and $p\left(y_k^f \mid x_k, \Theta^f\right)$ for $k = 1, \cdots, T$ are determined by equations (3.1a) and (3.1b), respectively. Substituting equation (3.5) into equation (3.4), the Q function is derived as

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) &= \int \ln\left[p\left(x_{0:T}, y_{1:T}^f \mid \Theta^f\right)\right] p\left(x_{0:T} \mid y_{1:T}^f, \Theta_n^f\right) dx_{0:T} \\ &= \int \ln\left[p\left(x_0\right)\right] p\left(x_{0:T} \mid y_{1:T}^f, \Theta_n^f\right) dx_{0:T} \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(x_k \mid x_{k-1}, \Theta^f\right)\right] p\left(x_{0:T} \mid y_{1:T}^f, \Theta_n^f\right) dx_{0:T} \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(y_k^f \mid x_k, \Theta^f\right)\right] p\left(x_{0:T} \mid y_{1:T}^f, \Theta_n^f\right) dx_{0:T} \end{aligned} \tag{3.6}$$

With further normalization of the states, the Q function can be written as

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) &= \int \ln\left[p\left(x_0\right)\right] p\left(x_0 \mid y_{1:T}^f, \Theta_n^f\right) dx_0 \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(x_k \mid x_{k-1}, \Theta^f\right)\right] p\left(x_{k-1:k} \mid y_{1:T}^f, \Theta_n^f\right) dx_{k-1:k} \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(y_k^f \mid x_k, \Theta^f\right)\right] p\left(x_k \mid y_{1:T}^f, \Theta_n^f\right) dx_k \end{aligned} \tag{3.7}$$

Thus, to evaluate and maximize the Q function, the unknown statistics are these smoothing densities $\left\{p\left(x_{k-1:k} \mid y_{1:T}^f, \Theta_n^f\right)\right\}_{k=1}^{T}$ and $\left\{p\left(x_k \mid y_{1:T}^f, \Theta_n^f\right)\right\}_{k=0}^{T}$. Due to the iterative nature of the EM algorithm, the smoothing step has high computation cost [14]. A practical solution is to replace the smoothing densities with filtering densities which can be easily obtained from Bayes filters. That means the Q function is approximated as follows

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) \approx & \int \ln\left[p\left(x_0\right)\right] p\left(x_0\right) dx_0 \\ & + \sum_{k=1}^{T} \int \ln\left[p\left(x_k \mid x_{k-1}, \Theta^f\right)\right] p\left(x_{k-1:k} \mid y_{1:k}^f, \Theta_n^f\right) dx_{k-1:k} \\ & + \sum_{k=1}^{T} \int \ln\left[p\left(y_k^f \mid x_k, \Theta^f\right)\right] p\left(x_k \mid y_{1:k}^f, \Theta_n^f\right) dx_k \end{aligned} \tag{3.8}$$

The implementation of the particle filter will be introduced in the next section.

3.3.2 Using both fast and slow delayed laboratory measurements

Consider the observed output data are fast measurements $Y^f = \left\{y_1^f, y_2^f, \cdots, y_T^f\right\}$ and slow delayed laboratory measurements $Y^m = \left\{y_{k_s(1)}^m, y_{k_s(2)}^m, \cdots, y_{k_s(N_m)}^m\right\}$. The hidden latent data are states $X = \{x_0, x_1, \cdots, x_T\}$. The parameters to be estimated are delayed time $\Gamma = \{d_1, d_2, \cdots, d_{N_m}\}$ and $\Theta^m = \left\{\theta^x, \theta^f, Q, R^f, \theta^m, R^m\right\}$. Then, the Q function is given by

$$Q_2\left(\Gamma, \Theta^m \mid \Gamma_n, \Theta_n^m\right) = E_{X \mid Y^f, Y^m, \Gamma_n, \Theta_n^m}\left\{\ln\left[p\left(Y^f, Y^m, X \mid \Gamma, \Theta^m\right)\right]\right\} \tag{3.9}$$

where Γ_n and Θ_n^m are the estimated parameters after n^{th} iteration. Specifically,

$$\Gamma_n = \left\{\left(d_1\right)_n, \left(d_2\right)_n, \cdots, \left(d_{N_m}\right)_n\right\} \tag{3.10}$$

$$\Theta_n^m = \left\{\theta_n^x, \theta_n^f, Q, R_n^f, \theta_n^m, R_n^m\right\} \tag{3.11}$$

The joint density $p\left(Y^f, Y^m, X \mid \Gamma, \Theta^m\right)$ in equation (3.9) can be decomposed using the Bayes' rule as

$$\begin{aligned} p\left(Y^f, Y^m, X \mid \Gamma, \Theta^m\right) &= p\left(Y^m \mid Y^f, X, \Gamma, \Theta^m\right) p\left(Y^f, X \mid \Gamma, \Theta^m\right) \\ &= p\left(Y^m \mid Y^f, X, \Gamma, \Theta^m\right) p\left(Y^f \mid X, \Gamma, \Theta^m\right) p\left(X \mid \Gamma, \Theta\right) \end{aligned} \tag{3.12}$$

Using the Markov property and the fact that slow and delayed measurements Y^m are independent of fast measurements Y^f, the first term on the right hand side in equation (3.12) can be further written as

$$\begin{aligned} p\left(Y^m \mid Y^f, X, \Gamma, \Theta^m\right) &= p\left(y_{k_s(1):k_s(N_m)}^m \mid y_{1:T}^f, x_{0:T}, d_{1:N_m}, \Theta^m\right) \\ &= p\left(y_{k_s(1):k_s(N_m)}^m \mid x_{0:T}, d_{1:N_m}, \Theta^m\right) \\ &= \prod_{j=1}^{N_m} p\left(y_{k_s(j)}^m \mid x_{s_j:s_j+l_j}, d_j, \Theta^m\right) \end{aligned} \tag{3.13}$$

where $s_j = k_s(j) - l_j - d_j$. Similarly, because fast measurements Y^m and states X are independent of delay Γ, the second and third term can be simplified as follows:

$$\begin{aligned} p\left(Y^f \mid X, \Gamma, \Theta^m\right) &= p\left(y^f_{1:T} \mid x_{0:T}, d_{1:N_m}, \Theta^m\right) \\ &= p\left(y^f_{1:T} \mid x_{0:T}, \Theta^m\right) \\ &= \prod_{k=1}^{T} p\left(y^f_k \mid x_k, \Theta^m\right) \end{aligned} \tag{3.14}$$

$$\begin{aligned} p\left(X \mid \Gamma, \Theta^m\right) &= p\left(x_{0:T} \mid d_{1:N_m}, \Theta^m\right) \\ &= p\left(x_{0:T} \mid \Theta^m\right) \\ &= \prod_{k=1}^{T} p\left(x_k \mid x_{k-1}, \Theta^m\right) p\left(x_0\right) \end{aligned} \tag{3.15}$$

Thus, substituting equations (3.13)–(3.15) into equation (3.12) yields

$$\begin{aligned} p\left(Y^f, Y^m, X \mid \Gamma, \Theta^m\right) = &\prod_{j=1}^{N_m} p\left(y^m_{k_s(j)} \mid x_{s_j:s_j+l_j}, d_j, \Theta^m\right) \\ &\times \prod_{k=1}^{T} p\left(y^f_k \mid x_k, \Theta^m\right) p\left(x_k \mid x_{k-1}, \Theta^m\right) p\left(x_0\right) \end{aligned} \tag{3.16}$$

Substituting equation (3.16) into equation (3.9) leads to the derivation of the Q function as

$$\begin{aligned} Q_2\left(\Gamma, \Theta^m \mid \Gamma_n, \Theta^m_n\right) = &E_{X|Y^f,Y^m,\Gamma_n,\Theta^m_n}\left\{\sum_{j=1}^{N_m} \ln p\left(y^m_{k_s(j)} \mid x_{s_j:s_j+l_j}, d_j, \Theta^m\right) + \ln p\left(x_0\right)\right\} \\ &+ E_{X|Y^f,Y^m,\Gamma_n,\Theta^m_n}\left\{\sum_{k=1}^{T}\left[\ln p\left(y^f_k \mid x_k, \Theta^m\right) + \ln p\left(x_k \mid x_{k-1}, \Theta^m\right)\right]\right\} \\ &= \sum_{j=1}^{N_m} \int \ln\left[p\left(y^m_{k_s(j)} \mid x_{s_j:s_j+l_j}, d_j, \Theta^m\right)\right] p\left(x_{0:T} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n\right) dx_{0:T} \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(y^f_k \mid x_k, \Theta^m\right)\right] p\left(x_{0:T} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n\right) dx_{0:T} \\ &+ \sum_{k=1}^{T} \int \ln\left[p\left(x_k \mid x_{k-1}, \Theta^m\right)\right] p\left(x_{0:T} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n\right) dx_{0:T} \\ &+ \int \ln\left[p\left(x_0\right)\right] p\left(x_{0:T} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n\right) dx_{0:T} \end{aligned} \tag{3.17}$$

With further marginalization of the states and delay time, the Q function can be rewritten as

$$\begin{aligned} Q_2\left(\Gamma, \Theta^m \mid \Gamma_n, \Theta_n^m\right) = & \int \ln\left[p\left(x_0\right)\right] p\left(x_0 \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) dx_0 \\ & + \sum_{j=1}^{N_m} \int \ln\left[p\left(y_{k_s(j)}^m \mid x_{s_j:s_j+l_j}, d_j, \Theta^m\right)\right] p\left(x_{s_j:s_j+l_j} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) dx_{s_j:s_j+l_j} \\ & + \sum_{k=1}^{T} \int \ln\left[p\left(y_k^f \mid x_k, \Theta^m\right)\right] p\left(x_k \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) dx_k \\ & + \sum_{k=1}^{T} \int \ln\left[p\left(x_k \mid x_{k-1}, \Theta^m\right)\right] p\left(x_{k-1:k} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) dx_{k-1:k} \end{aligned} \tag{3.18}$$

where the calculations of $\left\{p\left(x_{s_j:s_j+l_j} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right)\right\}_{j=1}^{N_m}$, $\left\{p\left(x_k \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right)\right\}_{k=0}^{T}$ and $\left\{p\left(x_{k-1:k} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right)\right\}_{k=1}^{T}$ are smoothing problems, for which the computation cost is high. Therefore, the smoothed densities are approximated as

$$p\left(x_{s_j:s_j+l_j} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) \approx p\left(x_{s_j:s_j+l_j} \mid y_{1:s_j+l_j}, \Gamma_n, \Theta_n^m\right) \tag{3.19}$$

$$p\left(x_k \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) \approx p\left(x_k \mid y_{1:k}, \Gamma_n, \Theta_n^m\right) \tag{3.20}$$

$$p\left(x_{k-1:k} \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) \approx p\left(x_{k-1:k} \mid y_{1:k}, \Gamma_n, \Theta_n^m\right) \tag{3.21}$$

$$p\left(x_0 \mid Y^f, Y^m, \Gamma_n, \Theta_n^m\right) \approx p\left(x_0\right) \tag{3.22}$$

where $y_{1:k}$ is defined as $y_{1:k} \triangleq \left\{y_{1:k}^f, y_{k_s(1):k_s(j)}^m\right\}$, with $k_s(j) - \left(d_j\right)_n \leqslant k$, for notational simplicity. Thus, the remaining task is to calculate the approximated densities in equations (3.19)–(3.21). The particle filtering method is introduced in the next section.

3.4 Parameter estimation using particle filter

3.4.1 Using only fast measurements

Using the particle filter, the densities $\left\{p\left(x_k \mid y_{1:k}^f, \Theta_n^f\right)\right\}_{k=1}^{T}$ in equation (3.8) are approximated as follows:

$$p\left(x_k \mid y_{1:k}^f, \Theta_n^f\right) = \sum_{i=1}^{N} w_k^{f(i)} \delta\left(x_k - x_k^{f(i)}\right) \quad \text{for} \quad k = 1, \cdots, T \tag{3.23}$$

where N is the number of particles, and $w_k^{f(i)}$ is the weight of particle $x_k^{f(i)}$. The joint densities $\left\{p\left(x_{k-1:k} \mid y_{1:k}^f, \Theta_n^f\right)\right\}_{k=1}^T$ are approximated as

$$p\left(x_{k-1:k} \mid y_{1:k}^f, \Theta_n^f\right) = \sum_{i=1}^N w_{k-1|k}^{f(i)} \delta\left(x_{k-1} - x_{k-1}^{f(i)}\right) \delta\left(x_k - x_k^{f(i)}\right) \quad \text{for} \quad k = 1, \cdots, T \tag{3.24}$$

where

$$w_{k-1|k}^{f(i)} = \frac{w_{k-1}^{f(i)} p\left(x_k^{f(i)} \mid x_{k-1}^{f(i)}, \Theta_n^f\right)}{\sum_{j=1}^N w_{k-1}^{f(j)} p\left(x_k^{f(j)} \mid x_{k-1}^{f(j)}, \Theta_n^f\right)} \tag{3.25}$$

The initial state distribution is approximated as

$$p\left(x_0\right) = \sum_{i=1}^N w_0^{f(i)} \delta\left(x_0 - x_0^{f(i)}\right) \tag{3.26}$$

Substituting equations (3.23)–(3.26) into equation (3.8), the Q function is derived as

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) \approx & \sum_{i=1}^N w_0^{f(i)} \ln\left[p\left(x_0^{f(i)}\right)\right] + \sum_{k=1}^T \sum_{i=1}^N w_{k-1|k}^{f(i)} \ln\left[p\left(x_k^{f(i)} \mid x_{k-1}^{f(i)}, \Theta^f\right)\right] \\ & + \sum_{k=1}^T \sum_{i=1}^N w_k^{f(i)} \ln\left[p\left(y_k^f \mid x_k^{f(i)}, \Theta^f\right)\right] \end{aligned} \tag{3.27}$$

where

$$p\left(x_k^{f(i)} \mid x_{k-1}^{f(i)}, \Theta^f\right) = \frac{1}{\sqrt{(2\pi)^{n_x} |Q|}} \exp\left\{-\frac{1}{2}\left[x_k^{f(i)} - f\left(x_{k-1}^{f(i)}, \theta^x\right)\right]^T Q^{-1} [\cdots]\right\} \tag{3.28}$$

$$p\left(y_k^f \mid x_k^{f(i)}, \Theta^f\right) = \frac{1}{\sqrt{(2\pi)^{n_y^f} |R^f|}} \exp\left\{-\frac{1}{2}\left[y_k^f - h^f\left(x_k^{f(i)}, \theta^f\right)\right]^T \left(R^f\right)^{-1} [\cdots]\right\} \tag{3.29}$$

Therefore, the Q function is finally derived as

$$\begin{aligned} Q_1\left(\Theta^f \mid \Theta_n^f\right) \approx & \sum_{i=1}^N w_0^{f(i)} \ln\left[p\left(x_0^{f(i)}\right)\right] \\ & + \sum_{k=1}^T \sum_{i=1}^N w_{k-1|k}^{f(i)} \left\{-\ln\left[\sqrt{(2\pi)^{n_x} |Q|}\right] - \frac{1}{2}\left[x_k^{f(i)} - f\left(x_{k-1}^{f(i)}, \theta^x\right)\right]^T Q^{-1} [\cdots]\right\} \\ & + \sum_{k=1}^T \sum_{i=1}^N w_k^{f(i)} \left\{-\ln\left[\sqrt{(2\pi)^{n_y^f} |R^f|}\right] - \frac{1}{2}\left[y_k^f - h^f\left(x_k^{f(i)}, \theta^f\right)\right]^T \left(R^f\right)^{-1} [\cdots]\right\} \end{aligned} \tag{3.30}$$

The algorithm is denoted as EM1 and can be summarized as:

1. *Initialization*: Initialize the parameter Θ^f to Θ_0^f. Set $n = 0$.
2. *Expectation*: Evaluate the approximate Q function according to equation (3.30) using the current parameter estimate Θ_n^f.
3. *Maximization*: Maximize the approximate Q function with respect to Θ^f and obtain the new parameter estimate Θ_{n+1}^f. Set $n = n + 1$.
4. *Iteration*: Repeat steps 2 and 3 until the convergence condition is satisfied.

3.4.2 Using both fast and slow delayed laboratory measurements

To derive the Q function in Section 3.3.2, probability densities $\{p(x_k \mid y_{1:k}, \Gamma_n, \Theta_n^m)\}_{k=1}^T$, $\{p(x_{k-1:k} \mid y_{1:k}, \Gamma_n, \Theta_n^m)\}_{k=1}^T$, and $\left\{p\left(x_{s_j:s_j+l_j} \mid y_{1:s_j+l_j}, \Gamma_n, \Theta_n^m\right)\right\}_{j=1}^{N_m}$ must be obtained. Since the delay of slow laboratory measurements is estimated and the calculation is off-line, the slow laboratory measurements can be treated as if there is no delay; this makes fusing two measurements simpler.

First, consider an augmented set of states:

$$x^a_{s_j^n+l_j} = \begin{bmatrix} x_{s_j^n+l_j} \\ m_{s_j^n+l_j} \end{bmatrix} \tag{3.31}$$

where

$$s_j^n = k_s(j) - l_j - (d_j)_n \tag{3.32}$$

$$m_{s_j^n+l_j} = m_{s_j^n+l_j-1} + \frac{1}{l_j+1} x_{s_j^n+l_j} \tag{3.33}$$

$$m_{s_j^n+l_j-1} = \frac{1}{l_j+1} \sum_{i=s_j^n}^{s_j^n+l_j-1} x_i \tag{3.34}$$

with $(d_j)_n$ being the estimated delay of slow laboratory measurement $y^m_{k_s(j)}$ after n^{th} EM iteration. An augmented measurement at time step $s_j^n + l_j$ is defined as

$$y^a_{s_j^n+l_j} = \begin{bmatrix} y^f_{s_j^n+l_j} \\ y^m_{k_s(j)} \end{bmatrix} \tag{3.35}$$

The covariance of $y^a_{s_j^n+l_j}$ is $R^a = \begin{bmatrix} R^f & \mathbf{0} \\ \mathbf{0} & R^m \end{bmatrix}$.

Using Bayes' rule, the posterior density of state $p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n\right)$ is calculated as

$$p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) = \frac{p\left(y^a_{s^n_j+l_j} \mid x^a_{s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right)}{\int p\left(y^a_{s^n_j+l_j} \mid x^a_{s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) dx^a_{s^n_j+l_j}} \tag{3.36}$$

The first term in the numerator is obtained as

$$p\left(y^a_{s^n_j+l_j} \mid x^a_{s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) = \frac{1}{\sqrt{(2\pi)^{n^a_y} |R^a_n|}} \exp\left\{-\frac{1}{2}\left[y^a_{s^n_j+l_j} - h^a\left(x^a_{s^n_j+l_j}\right)\right]^T (R^a_n)^{-1} [\cdots]\right\} \tag{3.37}$$

where

$$n^a_y = n^f_y + n^m_y \tag{3.38}$$

$$R^a_n = \begin{bmatrix} R^f_n & \mathbf{0} \\ \mathbf{0} & R^m_n \end{bmatrix} \tag{3.39}$$

$$h^a\left(x^a_{s^n_j+l_j}\right) = \begin{bmatrix} h^f\left(x_{s^n_j+l_j}, \theta^f\right) \\ h^m\left(m_{s^n_j+l_j}, \theta^m\right) \end{bmatrix} \tag{3.40}$$

The second term in the numerator can be computed as

$$p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) = \int p\left(x^a_{s^n_j+l_j} \mid x^a_{s^n_j+l_j-1}, \Theta^m_n\right) p\left(x^a_{s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) dx^a_{s^n_j+l_j-1} \tag{3.41}$$

where

$$x^a_{s^n_j+l_j-1} = \begin{bmatrix} x_{s^n_j+l_j-1} \\ m_{s^n_j+l_j-1} \end{bmatrix} \tag{3.42}$$

The transition density $p\left(x^a_{s^n_j+l_j} \mid x^a_{s^n_j+l_j-1}, \Theta^m_n\right)$ is determined by the following equation (3.43), with consideration of equations (3.1a) and (3.33),

$$x^a_{s^n_j+l_j} = \begin{bmatrix} f\left(x_{s^n_j+l_j-1}, \theta^x_n\right) \\ m_{s^n_j+l_j-1} + \frac{1}{l_j+1} f\left(x_{s^n_j+l_j-1}, \theta^x_n\right) \end{bmatrix} + \begin{bmatrix} \omega_{s^n_j+l_j-1} \\ \frac{1}{l_j+1}\omega_{s^n_j+l_j-1} \end{bmatrix} \tag{3.43}$$

Assume that $p\left(x^a_{s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right)$ in equation (3.41) can be approximated by the particle set $\left\{x^{a(i)}_{s^n_j+l_j-1}, w^{a(i)}_{s^n_j+l_j-1}\right\}_{i=1}^N$ using the particle filter.

$$p\left(x^a_{s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) = \sum_{i=1}^N w^{a(i)}_{s^n_j+l_j-1} \delta\left(x^a_{s^n_j+l_j-1} - x^{a(i)}_{s^n_j+l_j-1}\right) \tag{3.44}$$

To construct $\left\{x^{a(i)}_{s^n_j+l_j-1}, w^{a(i)}_{s^n_j+l_j-1}\right\}^N_{i=1}$, we need the particle set that approximates the density $p\left(x^a_{s^n_j:s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right)$, i.e.

$$p\left(x^a_{s^n_j:s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) = \sum_{i=1}^{N} w^{(i)}_{s^n_j+l_j-1} \delta\left(x_{s^n_j:s^n_j+l_j-1} - x^{(i)}_{s^n_j:s^n_j+l_j-1}\right) \tag{3.45}$$

Using the property of the particle filter, the smoothing particles $\left\{x^{(i)}_{s^n_j:s^n_j+l_j-1}\right\}^N_{i=1}$ can be obtained by storing and resampling the particle trajectory, and $\left\{w^{(i)}_{s^n_j:s^n_j+l_j-1}\right\}^N_{i=1}$ are the particle weights of state posterior density $p\left(x_{s^n_j+l_j-1} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right)$. Therefore, the particle set of the augmented state at time step $k_s(j)-1$ is

$$x^{a(i)}_{s^n_j+l_j-1} = \begin{bmatrix} x^{(i)}_{s^n_j+l_j-1} \\ m^{(i)}_{s^n_j+l_j-1} \end{bmatrix} \tag{3.46}$$

$$w^{a(i)}_{s^n_j+l_j-1} = w^{(i)}_{s^n_j+l_j-1} \tag{3.47}$$

where

$$m^{(i)}_{s^n_j+l_j-1} = \frac{1}{l_j+1} \sum_{k=s^n_j}^{s^n_j+l_j-1} x^{(i)}_k \tag{3.48}$$

Substituting equation (3.44) into equation (3.41) results in

$$p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j-1}, \Gamma_n, \Theta^m_n\right) = \sum_{i=1}^{N} w^{a(i)}_{s^n_j+l_j-1} \delta\left(x^a_{s^n_j+l_j} - x^{a(i)}_{s^n_j+l_j}\right) \tag{3.49}$$

where particles $\left\{x^{a(i)}_{s^n_j+l_j}\right\}^N_{i=1}$ are calculated through equation (3.43).

Next, substituting equation (3.49) into equation (3.36) yields

$$p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) = \sum_{i=1}^{N} \frac{w^{a(i)}_{s^n_j+l_j-1} p\left(y^a_{s^n_j+l_j} \mid x^{a(i)}_{s^n_j+l_j}, \Gamma_n, \Theta^m_n\right)}{\sum_{k=1}^{N} w^{a(k)}_{s^n_j+l_j-1} p\left(y^a_{s^n_j+l_j} \mid x^{a(k)}_{s^n_j+l_j}, \Gamma_n, \Theta^m_n\right)} \delta\left(x^a_{s^n_j+l_j} - x^{a(i)}_{s^n_j+l_j}\right) \tag{3.50}$$

Assume $p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n\right)$ can be represented as

$$p\left(x^a_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n\right) = \sum_{i=1}^{N} w^{a(i)}_{s^n_j+l_j} \delta\left(x^a_{s^n_j+l_j} - x^{a(i)}_{s^n_j+l_j}\right) \tag{3.51}$$

Comparing equation (3.50) with equation (3.51), we can easily find that

$$w_{s_j^n+l_j}^{a(i)} = \frac{w_{s_j^n+l_j-1}^{a(i)} p\left(y_{s_j^n+l_j}^{a} \mid x_{s_j^n+l_j}^{a(i)}, \Gamma_n, \Theta_n^m\right)}{\sum_{k=1}^{N} w_{s_j^n+l_j-1}^{a(k)} p\left(y_{s_j^n+l_j}^{a} \mid x_{s_j^n+l_j}^{a(k)}, \Gamma_n, \Theta_n^m\right)} \tag{3.52}$$

After computing the weights, the posterior density of state $x_{s_j^n+l_j}$ and $m_{s_j^n+l_j}$ can be represented as

$$p\left(x_{s_j^n+l_j} \mid y_{1:s_j^n+l_j}, \Gamma_n, \Theta_n^m\right) = \sum_{i=1}^{N} w_{s_j^n+l_j}^{(i)} \delta\left(x_{s_j^n+l_j} - x_{s_j^n+l_j}^{(i)}\right) \tag{3.53}$$

$$p\left(m_{s_j^n+l_j} \mid y_{1:s_j^n+l_j}, \Gamma_n, \Theta_n^m\right) = \sum_{i=1}^{N} w_{s_j^n+l_j}^{(i)} \delta\left(m_{s_j^n+l_j} - m_{s_j^n+l_j}^{(i)}\right) \tag{3.54}$$

where

$$w_{s_j^n+l_j}^{(i)} = w_{s_j^n+l_j}^{a(i)} \tag{3.55}$$

Thus, the probability densities $p\left(x_k \mid y_{1:k}\right)$ for $k = k_s(1), \cdots, k_s(N_m)$ have been derived, while densities for the other k values can be obtained through the general particle filter using fast measurements and represented as

$$p\left(x_k \mid y_{1:k}\right) = \sum_{i=1}^{N} w_k^{(i)} \delta\left(x_k - x_k^{(i)}\right) \tag{3.56}$$

The joint densities $p\left(x_{k-1:k} \mid y_{1:k}\right)$ for $k = 1, \cdots, T$ are approximated as

$$p\left(x_{k-1:k} \mid y_{1:k}\right) = \sum_{i=1}^{N} w_{k-1|k}^{(i)} \delta\left(x_{k-1} - x_{k-1}^{(i)}\right) \delta\left(x_k - x_k^{(i)}\right) \tag{3.57}$$

where

$$w_{k-1|k}^{(i)} = \frac{w_{k-1}^{(i)} p\left(x_k^{(i)} \mid x_{k-1}^{(i)}\right)}{\sum_{n=1}^{N} w_{k-1}^{(n)} p\left(x_k^{(n)} \mid x_{k-1}^{(n)}\right)} \tag{3.58}$$

and the initial state distribution is approximated as

$$p\left(x_0\right) = \sum_{i=1}^{N} w_0^{(i)} \delta\left(x_0 - x_0^{(i)}\right) \tag{3.59}$$

Therefore, the remaining task is to calculate $\left\{p\left(x_{s_j:s_j+l_j} \mid y_{1:s_j+l_j}, \Gamma_n, \Theta_n^m\right)\right\}_{j=1}^{N_m}$. Since $y_{k_s(j)}^m$ depends on $m_{s_j+l_j}$ and $m_{s_j+l_j}$ is a summation of states $x_{s_j:s_j+l_j}$, the first term on the right hand side of equation (3.18) can be calculated as

$$\sum_{j=1}^{N_m} \int \ln \left[p\left(y^m_{k_s(j)} \mid x_{s_j:s_j+l_j}, d_j, \Theta^m \right) \right] p\left(x_{s_j:s_j+l_j} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n \right) dx_{s_j:s_j+l_j}$$

$$\approx \sum_{j=1}^{N_m} \int \ln \left[p\left(y^m_{k_s(j)} \mid m_{s_j+l_j}, d_j, \Theta^m \right) \right] p\left(m_{s_j+l_j} \mid y_{1:s_j+l_j}, \Gamma_n, \Theta^m_n \right) dm_{s_j+l_j} \quad (3.60)$$

Now, we can see the problem is that there exists no analytical representation of the above equation as the delay d_j is unknown. A practical solution is to assume $d_j = (d_j)_n$, so

$$\sum_{j=1}^{N_m} \int \ln \left[p\left(y^m_{k_s(j)} \mid x_{s_j:s_j+l_j}, d_j, \Theta^m \right) \right] p\left(x_{s_j:s_j+l_j} \mid Y^f, Y^m, \Gamma_n, \Theta^m_n \right) dx_{s_j:s_j+l_j}$$

$$\approx \sum_{j=1}^{N_m} \int \ln \left[p\left(y^m_{k_s(j)} \mid m_{s^n_j+l_j}, (d_j)_n, \Theta^m \right) \right] p\left(m_{s^n_j+l_j} \mid y_{1:s^n_j+l_j}, \Gamma_n, \Theta^m_n \right) dm_{s_j+l_j}$$

$$\approx \sum_{j=1}^{N_m} \sum_{i=1}^{N} w^{(i)}_{s^n_j+l_j} \ln \left[p\left(y^m_{k_s(j)} \mid m^{(i)}_{s^n_j+l_j}, (d_j)_n, \Theta^m \right) \right] \quad (3.61)$$

where the final representation is obtained by substituting equation (3.54) into equation (3.61).

Substituting equations (3.56)–(3.59) and equation (3.61) into equation (3.18), the Q function is derived as

$$Q_2\left(\Gamma, \Theta^m \mid \Gamma_n, \Theta^m_n \right) \approx \sum_{i=1}^{N} w^{(i)}_0 \ln \left[p\left(x^{(i)}_0 \right) \right] + \sum_{j=1}^{N_m} \sum_{i=1}^{N} w^{(i)}_{s^n_j+l_j} \ln \left[p\left(y^m_{k_s(j)} \mid m^{(i)}_{s^n_j+l_j}, (d_j)_n, \Theta^m \right) \right]$$
$$+ \sum_{k=1}^{T} \sum_{i=1}^{N} w^{(i)}_k \ln \left[p\left(y^f_k \mid x^{(i)}_k, \Theta^m \right) \right] + \sum_{k=1}^{T} \sum_{i=1}^{N} w^{(i)}_{k-1|k} \ln \left[p\left(x^{(i)}_k \mid x^{(i)}_{k-1}, \Theta^m \right) \right] \quad (3.62)$$

where

$$p\left(y^m_{k_s(j)} \mid m^{(i)}_{s^n_j+l_j}, (d_j)_n, \Theta^m \right) = \frac{1}{\sqrt{(2\pi)^{n^m_y} |R^m|}} \exp \left\{ -\frac{1}{2} \left[y^m_{k_s(j)} - h^m\left(m^{(i)}_{s^n_j+l_j}, \theta^m \right) \right]^T (R^m)^{-1} [\cdots] \right\} \quad (3.63)$$

$$p\left(x^{(i)}_k \mid x^{(i)}_{k-1}, \Theta^m \right) = \frac{1}{\sqrt{(2\pi)^{n_x} |Q|}} \exp \left\{ -\frac{1}{2} \left[x^{(i)}_k - f\left(x^{(i)}_{k-1}, \theta^x \right) \right]^T Q^{-1} [\cdots] \right\} \quad (3.64)$$

$$p\left(y^f_k \mid x^{(i)}_k, \Theta^m \right) = \frac{1}{\sqrt{(2\pi)^{n^f_y} |R^f|}} \exp \left\{ -\frac{1}{2} \left[y^f_k - h^f\left(x^{(i)}_k, \theta^f \right) \right]^T \left(R^f \right)^{-1} [\cdots] \right\} \quad (3.65)$$

Therefore, the Q function is finally derived as

$$\begin{aligned}
&Q_2\left(\Gamma, \Theta^m \mid \Gamma_n, \Theta^m_n\right) \approx \sum_{i=1}^{N} w_0^{(i)} \ln\left[p\left(x_0^{(i)}\right)\right] \\
&+ \sum_{j=1}^{N_m}\sum_{i=1}^{N} w^{(i)}_{s_j^n+l_j}\left\{-\ln\left[\sqrt{(2\pi)^{n_y^m}\left|R^m\right|}\right] - \frac{1}{2}\left[y^m_{k_s(j)} - h^m\left(m^{(i)}_{s_j^n+l_j}, \theta^m\right)\right]^T \left(R^m\right)^{-1}[\cdots]\right\} \\
&+ \sum_{k=1}^{T}\sum_{i=1}^{N} w^{(i)}_{k}\left\{-\ln\left[\sqrt{(2\pi)^{n_y^f}\left|R^f\right|}\right] - \frac{1}{2}\left[y^f_{k} - h^f\left(x^{(i)}_{k}, \theta^f\right)\right]^T \left(R^f\right)^{-1}[\cdots]\right\} \\
&+ \sum_{k=1}^{T}\sum_{i=1}^{N} w^{(i)}_{k-1|k}\left\{-\ln\left[\sqrt{(2\pi)^{n_x}\left|Q\right|}\right] - \frac{1}{2}\left[x^{(i)}_{k} - f\left(x^{(i)}_{k-1}, \theta^x\right)\right]^T Q^{-1}[\cdots]\right\}
\end{aligned} \tag{3.66}$$

It should be noticed that the delay Γ cannot be estimated by maximizing $Q_2\left(\Theta^m \mid \Theta^m_n\right)$ above. To estimate d_j for each slow laboratory measurements $y^m_{k_s(j)}$, the maximum a posteriori principle can be used:

$$\left(d_j\right)_{n+1} = \arg\max_{d_j} p\left(d_j \mid \hat{X}, Y^m\right) \tag{3.67}$$

where $\hat{X}$ is the estimated state trajectory obtained from the particle filter using parameters Θ^m_n. The posterior density of d_j can be calculated using Bayes' rule as

$$\begin{aligned}
p\left(d_j = tr \mid \hat{X}, Y^m\right) &= p\left(d_j = tr \mid \hat{m}_t, y^m_{k_s(j)}\right) \\
&= \frac{p\left(y^m_{k_s(j)} \mid \hat{X}_t, d_j = tr\right) p\left(d_j = tr \mid \hat{m}_t\right)}{\sum_{n=1}^{M} p\left(y^m_{k_s(j)} \mid \hat{m}_n, d_j = nr\right) p\left(d_j = nr \mid \hat{m}_n\right)} \\
&= \frac{p\left(y^m_{k_s(j)} \mid \hat{m}_t, d_j = tr\right) p\left(d_j = tr\right)}{\sum_{n=0}^{M} p\left(y^m_{k_s(j)} \mid \hat{m}_n, d_j = nr\right) p\left(d_j = nr\right)} \\
&= w_{jt}
\end{aligned} \tag{3.68}$$

where

$$\hat{m}_t = \frac{1}{l_j + 1} \sum_{k=k_s(j)-l_j-tr}^{k_s(j)-l_j} \hat{x}_k \tag{3.69}$$

So the algorithm, denoted as EM2, can be summarized as

1. *Initialization*: Initialize the parameters Θ^m, Γ to Θ^m_0, Γ_0. Set $n = 0$.

2. *Expectation*: Evaluate the approximate Q function according to equation (3.66) using the current parameter estimates Θ^m_n and Γ_n.

3. *Maximization*: Maximize the approximate Q function with respect to Θ^m using $\Gamma = \Gamma_n$, and get the new parameter estimate Θ^m_{n+1}. Obtain Γ_{n+1} by maximizing w_{jt} in equation (3.68) for $j = 1, 2, \cdots, N_m$. Set $n = n + 1$.

4. *Iteration*: Repeat steps 2 and 3 until the convergence condition is satisfied.

3.5 Simulation and experimental study

In this section, the proposed method is evaluated through simulation and experimental studies. In the first simulation example, fast measurements with different noise covariances are used to test if fast measurements will influence the performance of EM2. In the second simulation example, EM2 with known delays is also examined. In the third example, the proposed method is experimentally studied on a hybrid tank with different slow delayed laboratory measurements. The details of EM1 and EM2 implementations are also shown in the third example.

3.5.1 Nonlinear process example

Consider the following nonlinear process taken from [25], [28]:

$$x_{k+1} = ax_k + bu_k + \omega_k \tag{3.70}$$

$$y^f_k = c\cos(x_k) + v^f_k \tag{3.71}$$

$$y^m_{k_s(j)} = m_{s_j+l_j} + v^m_{k_s(j)} \tag{3.72}$$

where $\omega_k \sim \mathcal{N}(0, Q)$, $v_k \sim \mathcal{N}\left(0, R^f\right)$, $v^m_{k_s(j)} \sim \mathcal{N}(0, R^m)$ and $a = 0.9$, $b = c = 1$, $Q = 0.001$, $R^f = 0.1$, $R^m = 5 \times 10^{-6}$. The integral term $m_{s_j+l_j}$ is calculated through equation (3.2), where the sample collection time l_j randomly varies between 1 and 2, and the time delay d_j for laboratory measurement $y^m_{k_s(j)}$ follows the uniform distribution:

$$p(d_j = t) = \frac{1}{3} \qquad \text{for} \quad t = 0, 1, 2 \tag{3.73}$$

The data set collected from the nonlinear process is shown in Figure 3.1. There are 100 fast measurements and 9 slow delayed laboratory measurements in this data set. In the simulation, R^f and R^m are assumed to be known, while a, b, c, Q and d_j are the parameters to be estimated. The initial parameter estimates are $\hat{a} = \hat{b} = \hat{c} = \hat{Q} = 0.5$, and $\hat{d}_j = 0$ for

$j = 1, \cdots, N_m$. In the PF approximation, $N = 150$ particles are used. Figure 3.2 shows the trajectories of the parameter estimates when both fast and delayed laboratory measurements are used. Correspondingly, the estimated delays are plotted in Figure 3.3. Table 3.1 contains the parameter values after 100 EM iterations, where EM1 represents the algorithm using only fast measurements and EM2 uses both measurements. For comparison, the simulation result when $R^f = 0.4$, $R^m = 5 \times 10^{-6}$ is also included.

Table 3.1: Parameter values after 100 iterations

Parameters	$R^f = 0.1$		$R^f = 0.4$	
	EM1	EM2	EM1	EM2
$a = 0.9$	0.9022	0.8993	0.8808	0.8934
$b = 1$	1.0106	1.0050	1.0657	1.0681
$c = 1$	1.0028	1.0044	0.7950	0.7841
$Q = 0.001$	1.3585×10^{-10}	1.0795×10^{-7}	6.5368×10^{-10}	6.2305×10^{-7}
The number of correctly estimated delays ($N_m = 9$)	–	8	–	6

Table 3.2: State estimation errors (ARMSE) of particle filters for different algorithms using the final parameter estimates

Algorithms	$R^f = 0.1$		$R^f = 0.4$	
	EM1	EM2	EM1	EM2
ARMSE	0.0925	0.0913	0.1164	0.1147

As can be seen from Table 3.1, although the estimated Q is much smaller than the true value, parameters a, b, c can converge to the neighbourhood of true values. When the noise variance R^f is increased from 0.1 to 0.4, the errors of estimated parameters are also increased, especially for the estimate of c, since the parameter c is directly related to the fast measurements. It is seen that algorithm EM2, which combines two measurements, performs slightly better in the estimation of a and b than EM1 in this example. Table 3.2 contains the ARMSE over 100 Monte Carlo simulations for state estimation with particle filters that use the final parameter estimates of EM1 and EM2. It shows that EM2 can also result in a better state estimation than EM1.

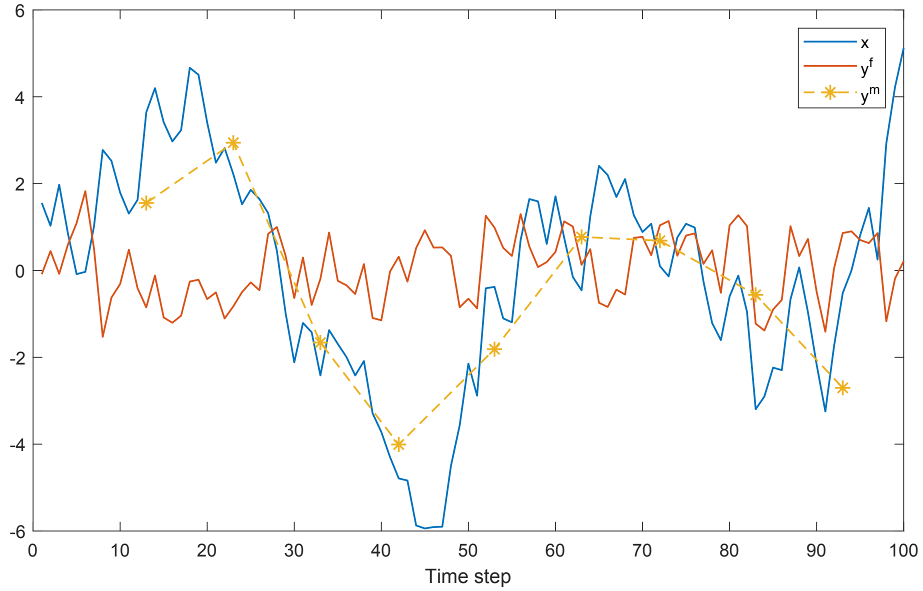

Figure 3.1: True states, fast measurements and slow delayed laboratory measurements with $R^f = 0.1,\ R^m = 5 \times 10^{-6}$

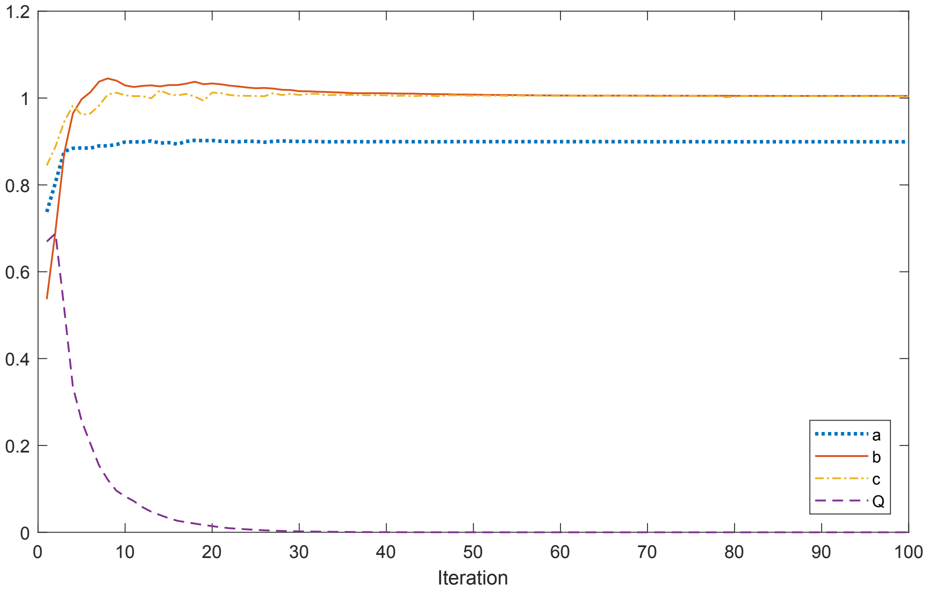

Figure 3.2: Parameter trajectories using both fast and delayed laboratory measurements with $R^f = 0.1,\ R^m = 5 \times 10^{-6}$

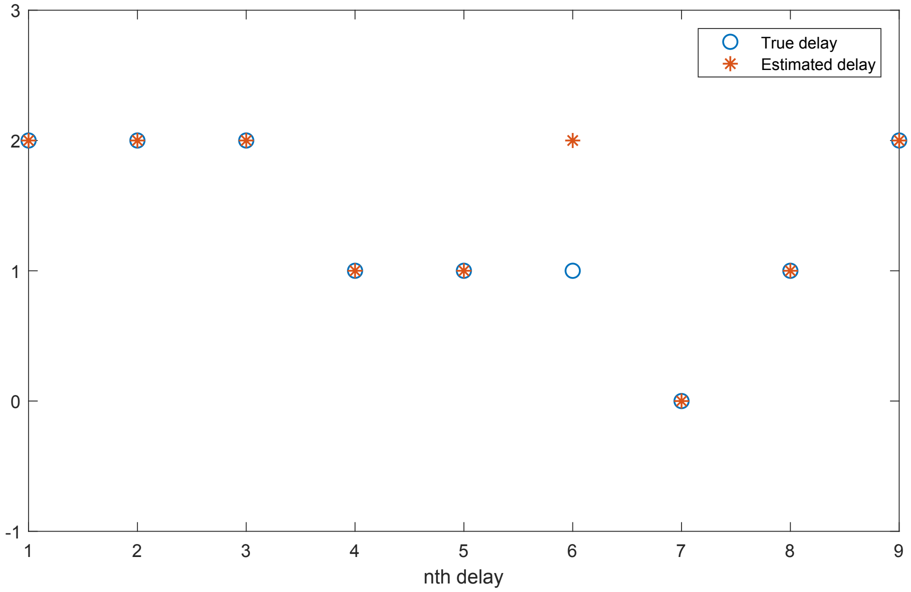

Figure 3.3: The estimations of delays with $R^f = 0.1$, $R^m = 5 \times 10^{-6}$

3.5.2 Semi-continuous fermentation example

A model of the semi-continuous (fed-batch) fermentation of baker's yeast is tested here. Assuming Monod-type kinetics for biomass growth and substrate consumption, the system is described by the following equations [5], [14]:

$$\frac{dx_1}{dt} = \left(\frac{\theta_1 x_2}{\theta_2 + x_2} - u_1 - \theta_4\right) x_1 \tag{3.74}$$

$$\frac{dx_2}{dt} = -\frac{\theta_3 x_1 x_2}{\theta_2 + x_2} + u_1 \left(u_2 - x_2\right) \tag{3.75}$$

where x_1 is the biomass concentration (g/L), x_2 is the substrate concentration (g/L), u_1 is the dilution factor (h^{-1}), and u_2 is the substrate concentration in the feed (g/L). θ_1, θ_2, θ_3 and θ_4 are the system parameters with true values of $\theta_1 = 0.31$, $\theta_2 = 0.18$, $\theta_3 = 0.56$ and $\theta_4 = 0.05$. In the simulation, the system is discretized with a sampling time $\Delta t = 1$ min with state covariance Q equal to $10^{-4} I$. The fast and slow delayed laboratory measurements are

$$y_k^f = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k^f \qquad v_k^f \sim \mathcal{N}\left(0, R^f\right) \tag{3.76}$$

$$y_{k_s(j)}^m = \begin{bmatrix} 1 & 0 \end{bmatrix} m_{s_j + l_j} + v_{k_s(j)}^m \qquad v_{k_s(j)}^m \sim \mathcal{N}\left(0, R^m\right) \tag{3.77}$$

where $R^f = 0.05$, $R^m = 5 \times 10^{-6}$, the sample collection time l_j of integral term $m_{s_j + l_j}$ randomly varies from 1 to 5 minutes following a uniform distribution, and the time delay d_j

also randomly varies between 0, 10 and 20 minutes which means $p\left(d_j = 10t\right) = 1/3$ for $t =$ 0, 1 or 2.

Assume θ_1 and θ_4 in equation (3.74) are unknown. $T = 2000$ fast measurements and $N_m = 65$ slow delayed laboratory measurements are collected from the process simulation, which are shown in Figure 3.4. The parameter estimation algorithms proposed in Section 3.4 are applied to this data set. The initial guesses of two parameters for both algorithms are $\hat{\theta}_1 = 0.1$, $\hat{\theta}_4 = 0.1$. The initial values for all time delays are set to zero. $N = 200$ particles are used for the particle filter approximation. EM1, EM2 and the EM2 version that uses laboratory measurements with known delays are simulated in this example.

After 100 iterations, Figure 3.5 shows the parameter trajectories for each algorithm. The parameters converge to the neighborhood of the true parameters. Although the trajectory trend of each algorithm is similar, Table 3.3 shows that the parameter estimation results of EM2 are closer to the true values. The delay estimation is shown in Figure 3.6 and the delay estimation accuracy of EM2 is about 81.5%. However, EM2 with known delays performs worse than EM2, which seems unreasonable. The possible reason is the particle filter approximation. In Table 3.4, the algorithm EM2 with known delays has the smallest error in estimates of state x_1, which is measured directly.

Table 3.3: Parameter values after 100 iterations for the semi-continuous fermentation process

Parameters	EM1	EM2 (delay unknown)	EM2 (delay known)
$\theta_1 = 0.31$	0.2831	0.2948	0.2901
$\theta_4 = 0.05$	0.0359	0.0439	0.0414
The number of correctly estimated delays ($N_m = 65$)	–	53	–

Table 3.4: State estimation errors (ARMSE) of particle filters for different algorithms using the final parameter estimates for the semi-continuous fermentation process

Algorithms	EM1	EM2 (delay unknown)	EM2 (delay known)
Biomass concentration x_1	0.0474	0.0460	0.0393
Substrate concentration x_2	0.0393	0.0437	0.0406

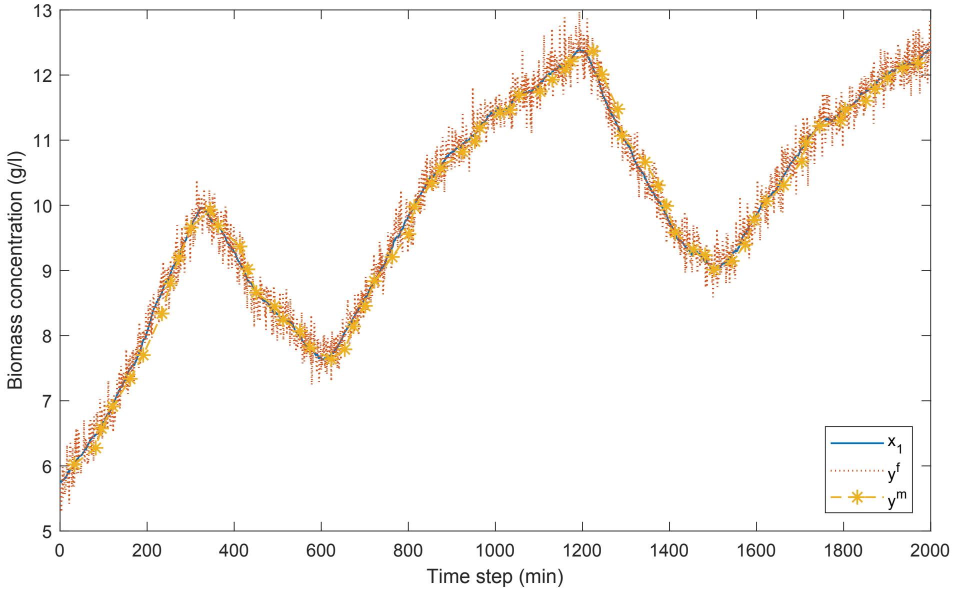

Figure 3.4: True states, fast measurements and slow delayed laboratory measurements for the semi-continuous fermentation process

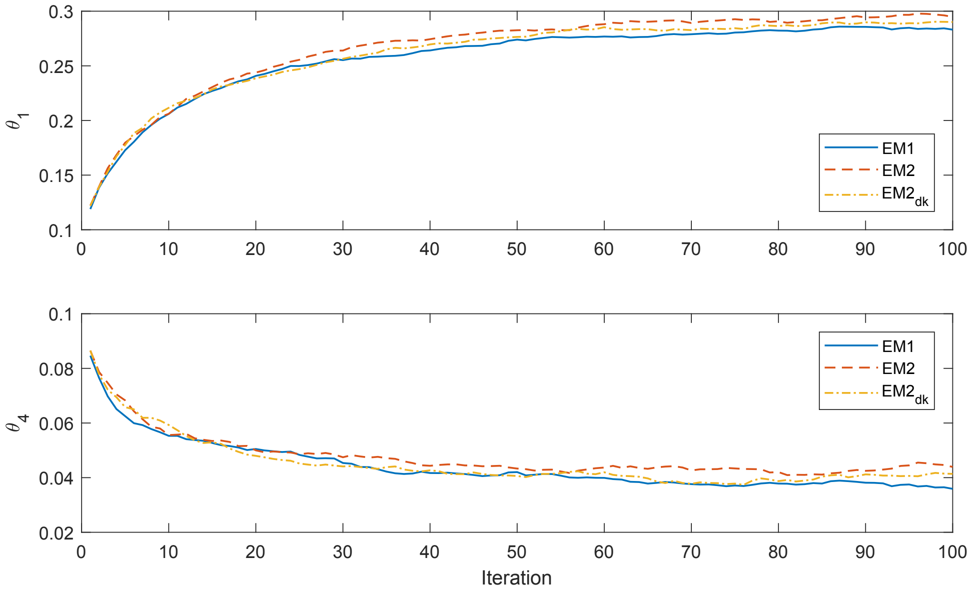

Figure 3.5: Parameter trajectories for the semi-continuous fermentation process ($EM2_{dk}$: EM2 with delays are known)

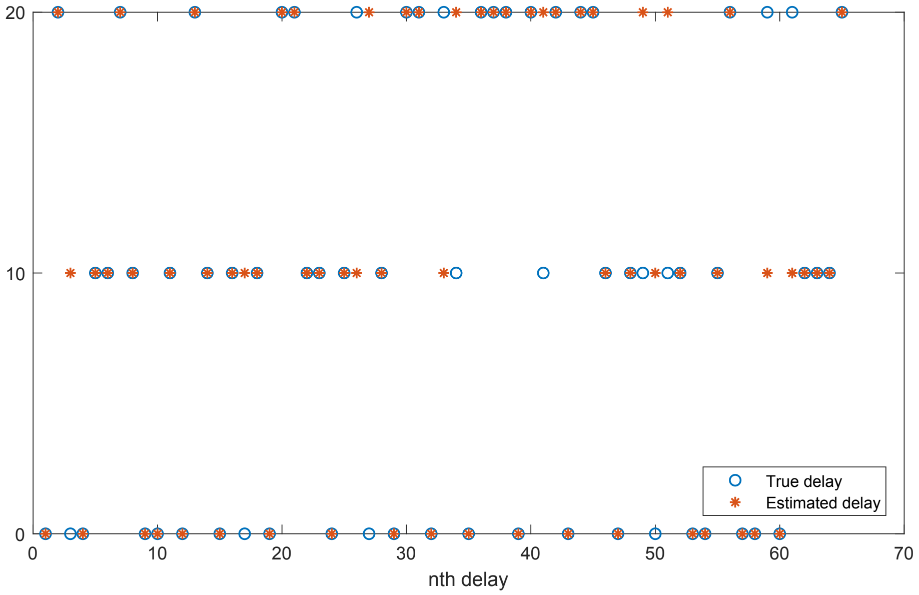

Figure 3.6: The estimations of delays for the semi-continuous fermentation process

3.5.3 Experimental evaluation on a hybrid tank

The proposed algorithms are experimentally evaluated on a hybrid tank at the University of Alberta. The details of the experimental plant are introduced in Section 2.5.2. The nonlinear state space model of the experimental plant derived using first principles modeling is as follows [19]–[21]:

$$\dot{h}_l = \frac{-k_1}{s_l}\sqrt{h_l - h_m} + \frac{1}{s_l}q_i \tag{3.78a}$$

$$\dot{h}_m = \frac{k_1}{s_m}\sqrt{h_l - h_m} + \frac{-k_2}{s_m}\sqrt{h_m} \tag{3.78b}$$

where h_l and h_m are the water level of the left and middle tank, respectively, q_i is the flow of the pump, s_l and s_m are the cross-sectional area of the tanks, which are equal to 243.22 cm^2, and k_1, k_2 are the coefficients of valves V_1 and V_2 that need to be estimated.

In the experiment, a RBS signal input around the operating point is used as the flow rate of the pump. The outputs of DP sensors are shown in Figure 3.7. To test the proposed approaches, the nonlinear state space model is discretized with a sampling time $\Delta t = 1$ s, using rectangular integration. The system state is defined as $x = [h_l \quad h_m]^T$ and the fast measurement is $y^f = x$. Thus,

$$f\left(x_k, \theta^x\right) = \phi_k \theta^x + b_k \tag{3.79}$$

$$h^f\left(x_k\right) = x_k \tag{3.80}$$

where

$$\theta^x = \left[\begin{array}{cc} k_1 & k_2 \end{array}\right]^T \tag{3.81}$$

$$\phi_k = \begin{bmatrix} -\dfrac{\sqrt{x_k(1) - x_k(2)}}{s_l} & 0 \\ \dfrac{\sqrt{x_k(1) - x_k(2)}}{s_m} & -\dfrac{\sqrt{x_k(2)}}{s_m} \end{bmatrix} \tag{3.82}$$

$$b_k = x_k + \begin{bmatrix} \dfrac{1}{s_l} q_i & 0 \end{bmatrix}^T \tag{3.83}$$

It is assumed that process noise ω_k and fast measurement noise v_k^f can be approximated by zero-mean Gaussian distributions with covariances $Q = \sigma^2 I$ and $R^f = 0.05I$, respectively, where σ^2 is set as a small value (0.02^2) to reduce the estimation error of θ^x. The initial guess $\hat{x}_0$ follows a Gaussian distribution $\mathcal{N}\left(\left[44.19 \quad 34.83\right]^T, I\right)$.

In this example, we conduct two experiments. In the first one, the integrated laboratory measurement is the integration of the left tank level, and in the second experiment, the integrated laboratory measurement is the integration of the middle tank level [21]. Therefore, the slow delayed laboratory measurement is

$$y_{k_s(j)}^m = C_m m_{s_j + l_j} + v_{k_s(j)}^m \tag{3.84}$$

where C_m varies depending on the experiment, $v_{k_s(j)}^m$ is the measurement noise which follows a Gaussian distribution $\mathcal{N}\left(0, \sigma_m^2\right)$ and the variance σ_m^2 is equal to 4×10^{-6}. The parameters to be estimated are the delay d_j for each slow delayed laboratory measurement $y_{k_s(j)}^m$.

There are 7000 measurements obtained from DP level sensors as shown in Figure 3.7. In the simulation, we assume the fast measurements are inaccurate and noisy. So the fast measurement data used to conduct parameter estimation is the middle tank DP sensor output, with artificially added Gaussian noise whose covariance is equal to R^f. In order to construct the slow delayed laboratory measurement, the values of DP sensors are integrated over $l_j + 1$ sampling instants as shown in equation (3.2). Specifically, the collection time of samples l_j randomly varies from 1 to 5 seconds following a uniform distribution, and the time

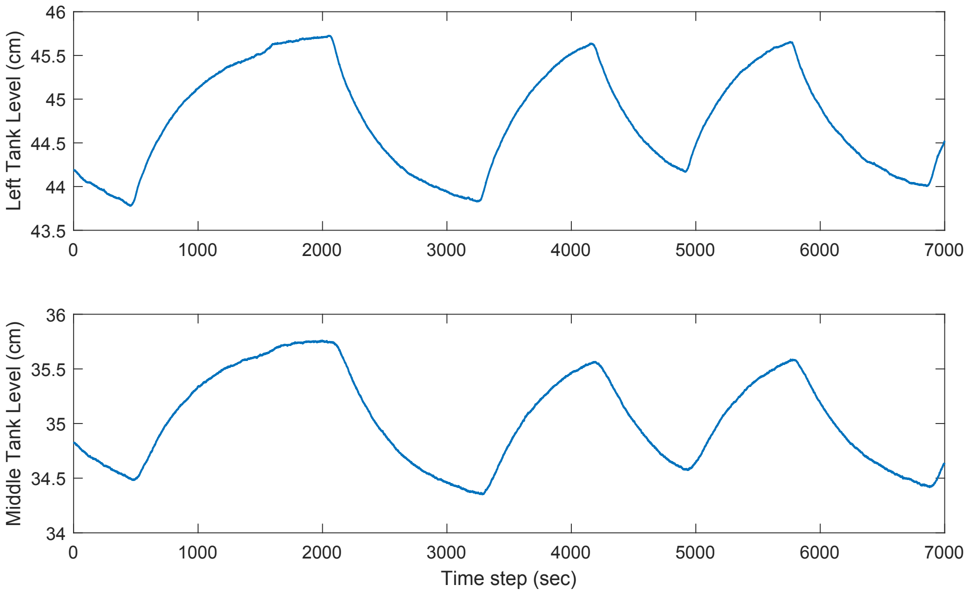

Figure 3.7: Outputs of the left and middle tank DP level sensors

delay d_j also randomly varies between 0, 1, 2 and 3 minutes, which means $p\left(d_j = 60t\right) = 1/4$ for $t = 0$, 1, 2 or 3.

For comparison, EM1 uses the first 5000 fast measurements to estimate θ^x, and EM2 uses the same 5000 fast measurements and $N_m = 25$ slow delayed laboratory measurements to estimate both θ^x and Γ. Note that there are 7000 DP sensors measurements. Hence, the remaining 2000 DP sensors measurements are used for testing the model.

In EM1, the Q function is calculated according to equation (3.30), where the first and last term on the right hand side are independent on θ^x. So, maximizing $Q_1\left(\theta^x \mid \theta_n^x\right)$ is equivalent to maximize the following equation:

$$\begin{aligned} F\left(\theta^x \mid \theta_n^x\right) &= \sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{f(i)} \left\{ -\ln\left[\sqrt{(2\pi)^{n_x}\left|Q\right|}\right] - \frac{1}{2}\left[x_k^{f(i)} - f\left(x_{k-1}^{f(i)}, \theta^x\right)\right]^T Q^{-1}\left[\cdots\right]\right\} \\ &= \sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{f(i)} \left[-\ln\left(2\pi\sigma\right) - \frac{1}{2\sigma^2}\left(x_k^{f(i)} - \phi_{k-1}^{f(i)}\theta^x - b_{k-1}^{f(i)}\right)^T\left(\cdots\right)\right] \end{aligned} \tag{3.85}$$

Then, taking the derivative of $F\left(\theta^x \mid \theta_n^x\right)$ with respect to θ^x yields

$$\frac{\partial F\left(\theta^x \mid \theta_n^x\right)}{\partial \theta^x} = \sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{f(i)} \left[\frac{1}{\sigma^2}\left(\phi_{k-1}^{f(i)}\right)^T\left(x_k^{f(i)} - \phi_{k-1}^{f(i)}\theta^x - b_{k-1}^{f(i)}\right)\right] \tag{3.86}$$

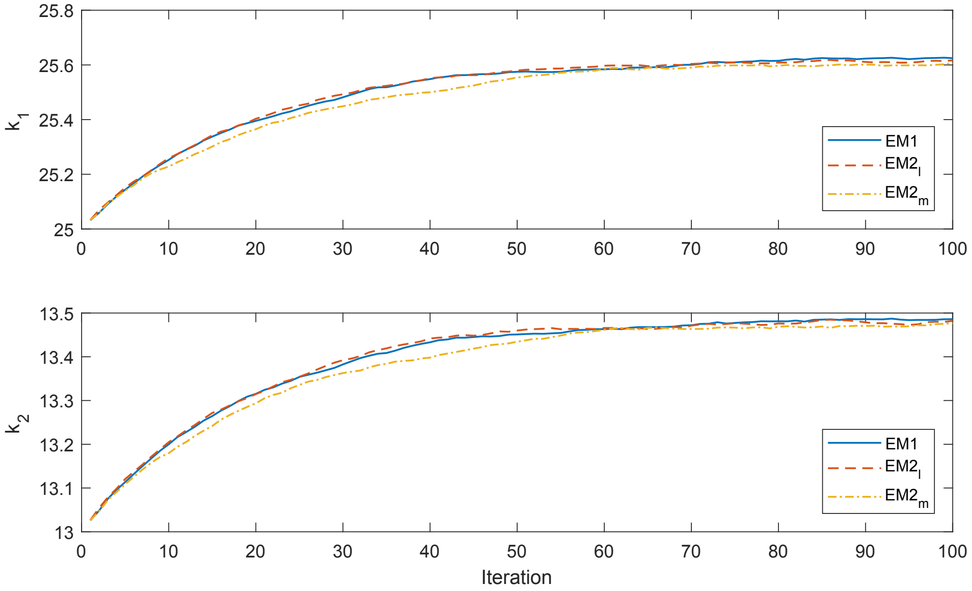

Figure 3.8: Parameter trajectories for the hybrid tank system (EM2$_\mathrm{l}$: EM2 using the integrated left tank level as the laboratory measurement. EM2$_\mathrm{m}$: EM2 using the integrated middle tank level as the laboratory measurement)

To maximize $F\left(\theta^x \mid \theta_n^x\right)$, the right hand side of equation (3.86) is set to zero and the new parameter estimate θ_{n+1}^x can be calculated by

$$\theta_{n+1}^x = \left[\sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{f(i)} \left(\phi_{k-1}^{f(i)}\right)^T \phi_{k-1}^{f(i)}\right]^{-1} \left[\sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{f(i)} \left(\phi_{k-1}^{f(i)}\right)^T \left(x_k^{f(i)} - b_{k-1}^{f(i)}\right)\right] \tag{3.87}$$

In EM2, the parameters to be estimated are the time delay sequence Γ and θ^x. The Q function is calculated according to equation (3.66). Since the parameter θ^m of the slow and delayed laboratory measurements are assumed to be known, the new parameter estimate can be calculated using equation (3.87) by replacing $\left\{w_{k-1|k}^{f(i)}, x_k^{f(i)}, \phi_{k-1}^{f(i)}, b_{k-1}^{f(i)}\right\}_{i=1}^{N}$ with $\left\{w_{k-1|k}^{(i)}, x_k^{(i)}, \phi_{k-1}^{(i)}, b_{k-1}^{(i)}\right\}_{i=1}^{N}$ derived from the particle filter using both fast and delayed slow laboratory measurements.

In simulation, the initial guesses for parameters are $\left\{\hat{k}_1 = 25, \hat{k}_2 = 13\right\}$, and $\hat{d}_j = 0$ for $j = 1, \cdots, N_m$. The number of particles in the particle filtering approximation is $N = 200$. The estimations are terminated after 100 iterations. Figure 3.8 shows the trajectory of the estimates for k_1 and k_2. Tables 3.5 and 3.6 contain the final parameter values and test errors for both levels, respectively. The test errors of EM2s are smaller than EM1, which indicates that fusing the slow delayed laboratory measurements can indeed improve the accuracy of

Table 3.5: Parameter values after 100 iterations for the hybrid tank system

Parameters	EM1	EM2 (left level integrated)	EM2 (middle level integrated)
k_1	25.6248	25.6065	25.6165
k_2	13.4859	13.4770	13.4821
The number of correctly estimated delays ($N_m = 25$)	–	13	16

Table 3.6: Test errors (RMSE) using the final parameter estimates of different algorithms for the hybrid tank system

Test error (RMSE)	EM1	EM2 (left level integrated)	EM2 (middle level integrated)
Left tank level	0.1760	0.1309	0.1560
Middle tank level	0.0587	0.0328	0.0456

estimated parameters k_1 and k_2, even when the estimations of delays may not be sufficiently accurate.

3.6 Conclusions

In this chapter, the problem of parameter estimation for a nonlinear system with multirate measurements and random time delays is studied. The proposed method is based on EM algorithm by incorporating the delays of slow measurements as the parameter to be estimated. In the expectation step, smoothing densities are approximated using the particle filter. In the maximization step, delays are estimated by implementing maximum a posteriori principle. Simulations show the proposed algorithm which uses both fast and slow delayed measurements can provide more accurate parameter and state estimates than that only using fast measurements.

Chapter 4

State and parameter estimation for the FWKO vessel

4.1 Process Description

A Free Water Knockout (FWKO) is a three-phase separator that is used to remove free water held in the vessel and separate brine from crude oil. It is referred to as a three-phase separator because it is capable of segregating oil, gas, and free water [22]. The principle of FWKO is that the oily emulsion will get separated from the water because of the difference in their densities, rate of flow, and residence time. Figure 4.1 is the picture of a FWKO vessel.

In oil sands industries, it is important to control the water content in the emulsion output of the FWKO to ensure the product quality. Therefore, the objective is to develop a model for the FWKO output emulsion water content and estimate the water content. The data set used in this chapter is collected from an industrial site, with the source withheld for reasons of confidentiality. Figure 4.2 shows the process flow diagram of the research object and the measured variable of each sensor is presented in Table 4.1. These online measurements are sampled every 1 minute and can be treated as fast measurements. It is known that the water content measured by sensor AI434 is not accurate and reliable. So the operators also sample the output emulsion for laboratory analysis. The off-line laboratory analysis is much more accurate, but is only available every 6 hours. Figure 4.3 shows the comparison between the measurements from the online sensor and laboratory analysis, where the measurements are resampled as introduced in the next section, and the Y-axis scale is erased for confidentiality. It can be seen that the online measurement has an obvious

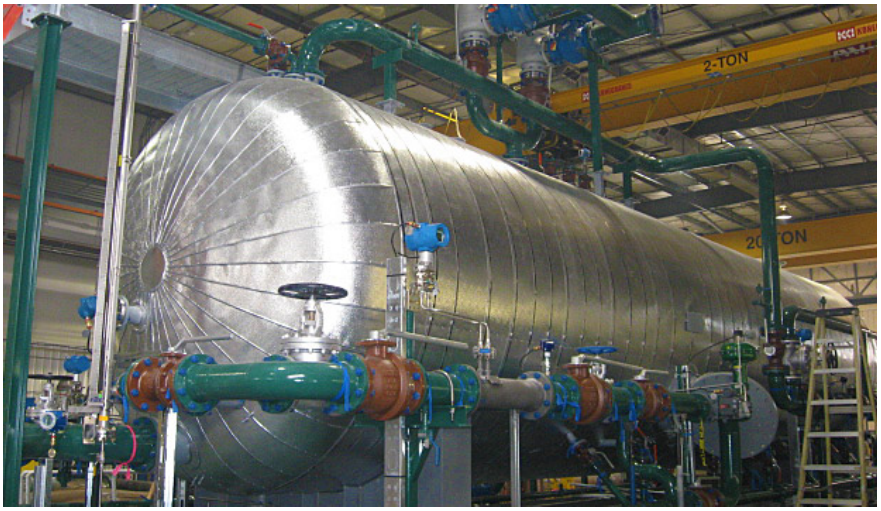

Figure 4.1: Free Water Knockout (FWKO) [49]

bias compared to the laboratory data. Therefore, obtaining a more accurate and real-time measurement of the output emulsion water content is desirable.

Considering this multirate measurements system, the algorithm proposed in Chapter 3 is used to estimate the parameters as well as the states for FWKO vessels. The remainder of this chapter is organized as follows. Section 4.2 introduces the procedure of data preprocessing. Section 4.3 solves the parameter estimation problem for the FWKO vessel. In Section 4.4, we focus on the state estimation for water content when the laboratory analysis delay is unknown. The final section includes the conclusion of this chapter.

Table 4.1: Measured variables of the online sensors

Sensor	Variable
FI451	Emulsion input flowrate (m^3/h)
TI356	Emulsion input temperature (^oC)
FFIC780	Clarifier injection flowrate (mL/min)
FIC435	Produced water flowrate (m^3/h)
LHS433	FWKO level measurement (%)
TI437	FWKO temperature (^oC)
PIT431	FWKO pressure (kPa)
FIC434	Emulsion output flowrate (m^3/h)
AI434	Emulsion output water content (%)

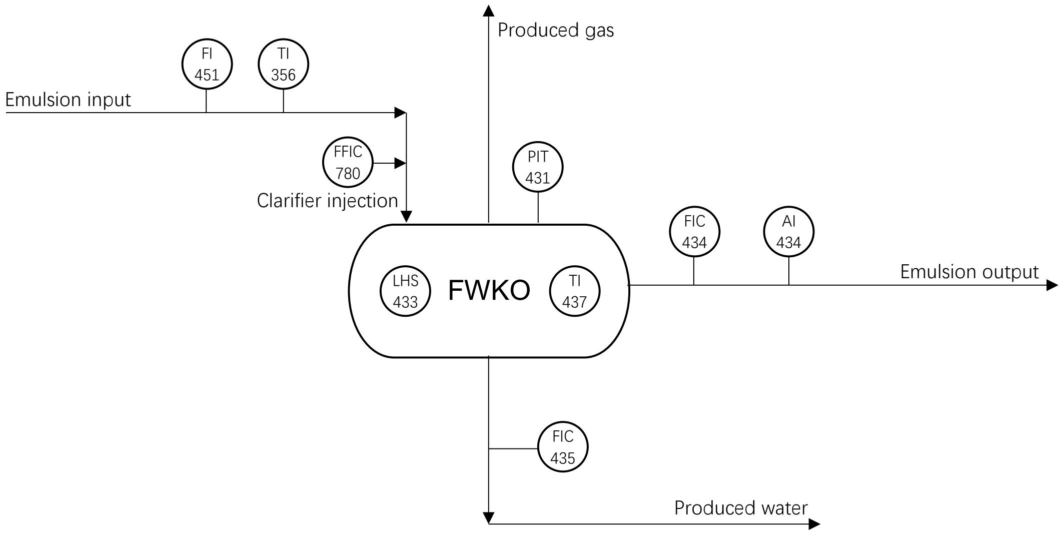

Figure 4.2: Process flow diagram for the Free Water Knockout (FWKO)

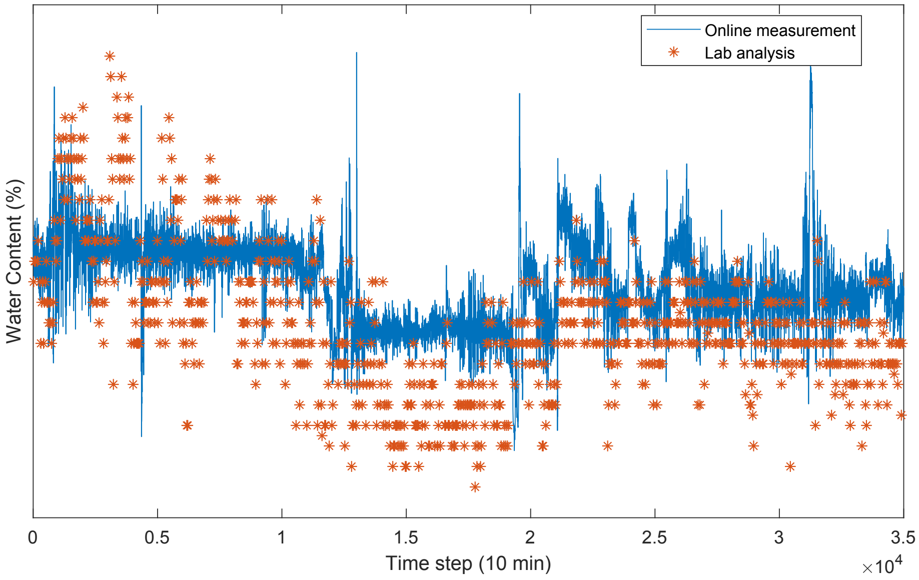

Figure 4.3: Comparison between the online measurements and laboratory analysis

4.2 Data preprocessing

Before modelling the FWKO system, the original data set is needed to be preprocessed due to the problems such as missing data, outliers and so on. The methods applied to deal with these problems are briefly introduced in this section.

Dealing with missing data

Data points are usually missing due to sensors malfunction, process shut down or maintenance activities. The missing value is filled by a random number from a Gaussian distribution with mean μ and variance σ^2. μ and σ^2 are calculated according to the previous and future measurements within a window as 200.

Dealing with outliers

Outliers are detected and replaced dynamically using the moving window Mean Absolute Deviation (MAD) principle. The MAD around of a set $X = \{x_1, x_2, \cdots, x_n\}$ is calculated as follows [50]:

$$MAD\left(X\right) = \frac{1}{n}\sum_{i=1}^{n}\left|x_i - m\left(X\right)\right| \tag{4.1}$$

The choice of measure of central tendency $m\left(X\right)$ has a marked effect on the value of the mean deviation. The two commonly used central points are the mean and median. Since the mean value may be affected by other possible outliers in the chosen window, the median value of the data set is used to calculate the MAD. So the data point z_k is detected as an outlier if

$$\left|z_k - median\left(Z\right)\right| > 2MAD\left(Z\right) \tag{4.2}$$

where the data set $Z = \{z_{k-1000}, \cdots, z_{k-1}, z_{k+1}, \cdots, z_{k+1000}\}$.

Resampling

Since there are a very large number of samples in the data set, the fast measurements are resampled every 10 minutes to reduce the computational cost.

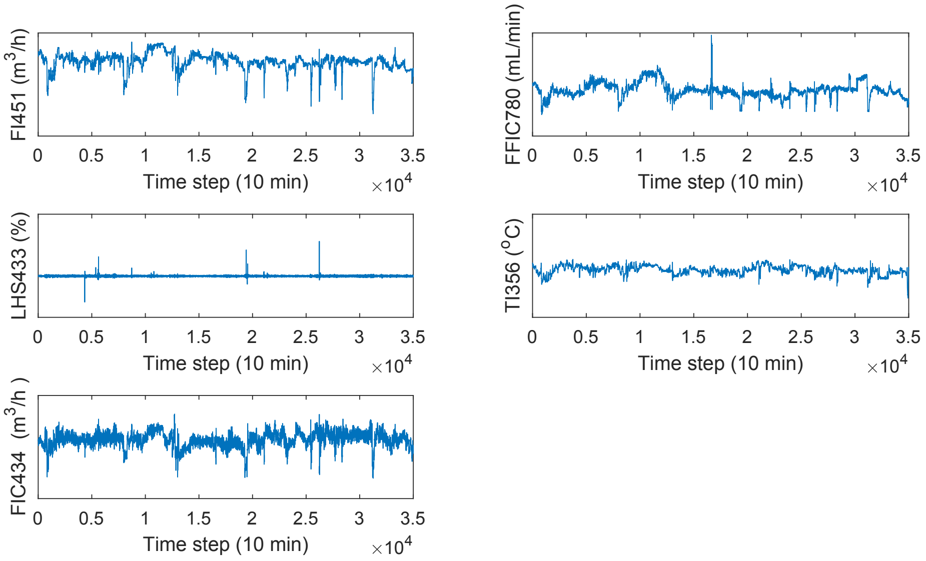

Figure 4.4: Input data for the FWKO vessel

4.3 Parameter estimation for the FWKO vessel

Considering the continuity of the separation process in the FWKO, it is appropriate to use a nonlinear dynamic state space model to describe the system as following:

$$\begin{bmatrix} x_k \\ b_k \end{bmatrix} = \begin{bmatrix} ax_{k-1} + B^T u_{k-1} + c\cos x_{k-1} \\ b_{k-1} \end{bmatrix} + \begin{bmatrix} \omega_{k-1} \\ v^b_{k-1} \end{bmatrix} \tag{4.3}$$

$$y^f_k = ex_k + b_k + v^f_k \tag{4.4}$$

$$y^m_{ks_j} = x_{s_j} + v^m_{ks_j} \tag{4.5}$$

where x_k is the emulsion output water content (%); b_k is the bias; u_k is the model input including five variables: emulsion output flowrate, clarifier injection flowrate, FWKO level, emulsion input flowrate and temperature, and the input data is shown in Figure 4.4; y^f_k is the fast measurement from sensor AI434; $y^m_{ks_j}$ is the jth slow measurement with delay d_j which is sampled at s_j and arrives at time ks_j; ω_{k-1}, v^b_k, v^f_k and $v^m_{s_j}$ are assumed to be independent and identically distributed Gaussian noises with variances $Q = 1$, $Q^b = 0.1$, $R^f = 4$ and $R^m = 1$, respectively. So the parameters to be estimated are $\Theta = \{a, B, c, e, d_{1:N_m}\}$.

The proposed parameter estimation algorithm presented in Chapter 3 is applied to estimate Θ in the state space model. But because the slow measurement in FWKO does not have integral terms, some minor changes are made to the Q function. By eliminating the terms that are not related to Θ, the Q function is given by

$$\begin{aligned} Q\left(\Theta \mid \Theta_n\right) = & \sum_{k=1}^{T}\sum_{i=1}^{N} w_k^{(i)} \left\{ -\ln\left(\sqrt{2\pi |R^f|}\right) - \frac{\left(y_k^f - e x_k^{(i)} - b_k^{(i)}\right)^2}{2R^f} \right\} \\ & + \sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{(i)} \left\{ -\ln\left(\sqrt{2\pi |Q|}\right) - \frac{\left(x_k^{(i)} - a x_{k-1}^{(i)} - B^T u_{k-1} - c\cos x_{k-1}^{(i)}\right)^2}{2Q} \right\} \end{aligned} \tag{4.6}$$

where $\left\{x_k^{(i)}, w_k^{(i)}\right\}_{i=1}^{N}$ is the particle set from particle filter, $w_{k-1|k}^{(i)}$ is calculated through equation (3.58).

By taking the derivative over the Q function with respect to $\{a, B, c, e\}$ and setting it to zero, the parameter estimates of the current iteration are calculated as

$$a_{n+1} = \frac{\sum_{i=1}^{N} w_{k-1|k}^{(i)} x_{k-1}^{(i)} \left(x_k^{(i)} - B_n^T u_{k-1} - c_n \cos x_{k-1}^{(i)}\right)}{\sum_{i=1}^{N} w_{k-1|k}^{(i)} \left(x_{k-1}^{(i)}\right)^2} \tag{4.7}$$

$$B_{n+1} = \left(\sum_{k=1}^{T} u_{k-1} u_{k-1}^T\right)^{-1} \left[\sum_{k=1}^{T}\sum_{i=1}^{N} w_{k-1|k}^{(i)} \left(x_k^{(i)} - a_n x_{k-1}^{(i)} - c_n \cos x_{k-1}^{(i)}\right) u_{k-1}\right] \tag{4.8}$$

$$c_{n+1} = \frac{\sum_{i=1}^{N} w_{k-1|k}^{(i)} \cos x_{k-1}^{(i)} \left(x_k^{(i)} - a_n x_{k-1}^{(i)} - B_n^T u_{k-1}\right)}{\sum_{i=1}^{N} w_{k-1|k}^{(i)} \left(\cos x_{k-1}^{(i)}\right)^2} \tag{4.9}$$

$$e_{n+1} = \frac{\sum_{k=1}^{T}\sum_{i=1}^{N} w_k^{(i)} x_k^{(i)} \left(y_k^f - b_k^{(i)}\right)}{\sum_{k=1}^{T}\sum_{i=1}^{N} w_k^{(i)} \left(x_k^{(i)}\right)^2} \tag{4.10}$$

The delay is assumed to be 0, 1 or 2 hours and is estimated by applying equation (3.67).

In the simulation, the initial guesses for parameters are $\hat{a} = 0.5$, $\hat{c} = 1$, $\hat{e} = 1$, $\hat{B}^T = \begin{bmatrix} 0.01 & 0.01 & 0.01 & 0.01 & 0.01 \end{bmatrix}$ and $\hat{d}_j = 0$ for $j = 1, \cdots, N_m$. The simulation is terminated when the mean absolute error (MAE) between the estimated water contents from particle filter and the true laboratory measurements stops decreasing. In particular, we consider two MAE, for which MAE_1 and MAE_2 are the mean absolute errors after and before fusing the laboratory measurements, respectively. Since we only have the references from laboratory analysis, MAE_2 is a better criterion to evaluate the algorithms. Moreover, two algorithms are simulated for comparison, where EM1 assumes there are no delays in laboratory analysis and EM2 considers variable delays as described in Chapter 3.

Figure 4.5 shows the state estimation performance on the entire training data using EM1. Figure 4.9 is a plot of estimated bias of the online sensor. The bias has a similar trend with

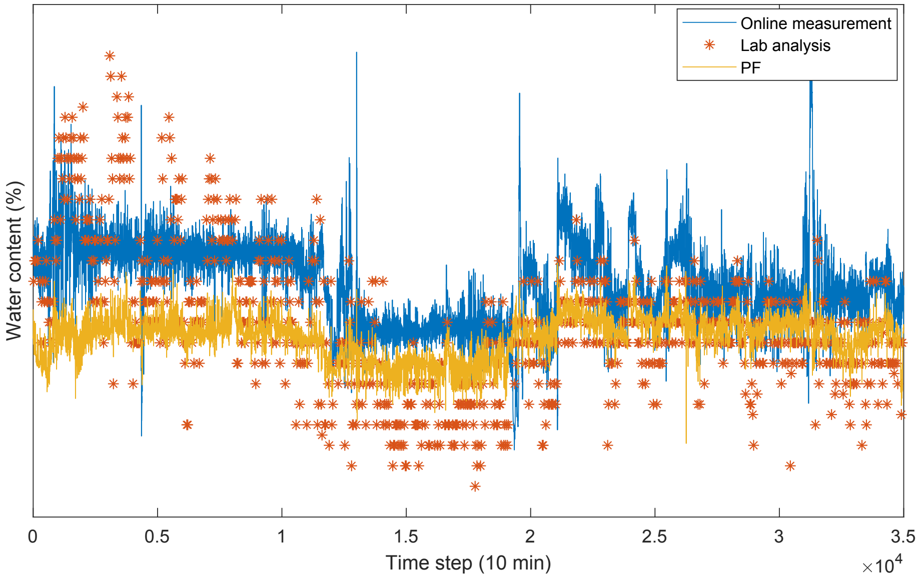

Figure 4.5: State estimation performance of EM1 on the entire training data

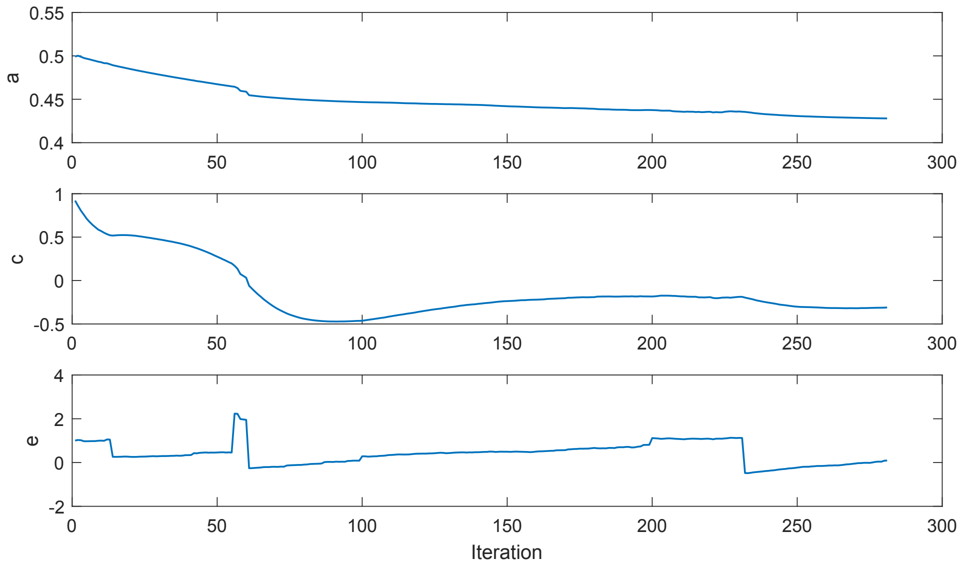

Figure 4.6: Parameter trajectories of EM1 when using the entire training data

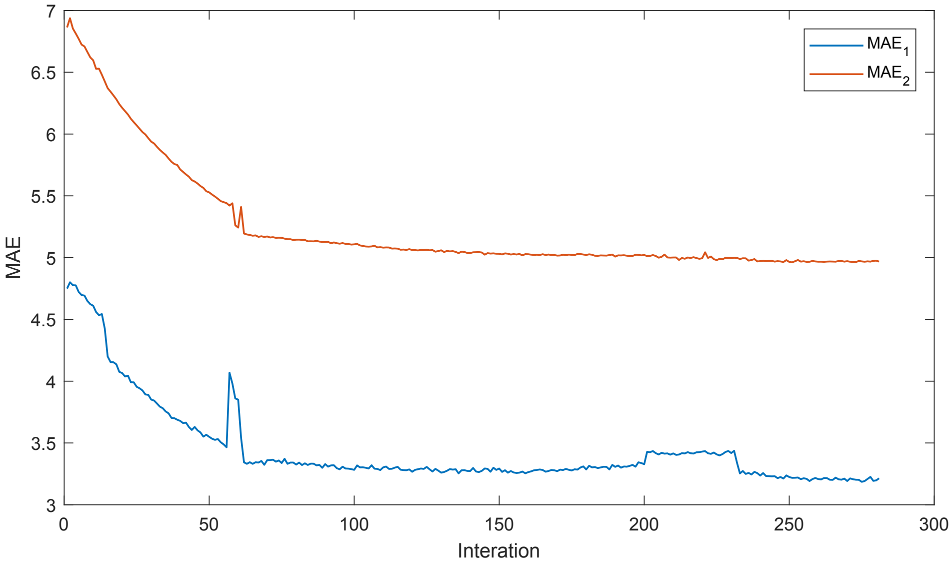

Figure 4.7: MAE trajectories of EM1 when using the entire training data

the online measurements. Figure 4.6 and 4.7 plot the parameter and MAE trajectories, respectively. As we can see, although the MAE trajectories converge, the parameter trajectories are still changing. It is shown that the state estimation is not satisfactory, especially in the time period before 100000 min.

Therefore, considering the instability of the laboratory analysis in this time period, we discard the data points before 150000 min, and train a model on the rest of the data. In the second simulation, the initial guesses are unchanged, but noise covariances are increased to $Q = 6$, $Q^b = 1$, $R^f = 6$ and $R^m = 1$. As seen in Figure 4.8, the state estimation performance are improved. The corresponding parameter and MAE trajectories are shown in Figures 4.10 and 4.11, respectively. The parameter trajectories demonstrate that the parameter estimates for c and e have converged.

Table 4.2 shows the state estimation results in the last iteration for the two simulations. It is shown that the proposed algorithm EM2, which considers the measurement delays, provides best estimates on water content. The algorithm assuming no delays also performs better than the online sensor.

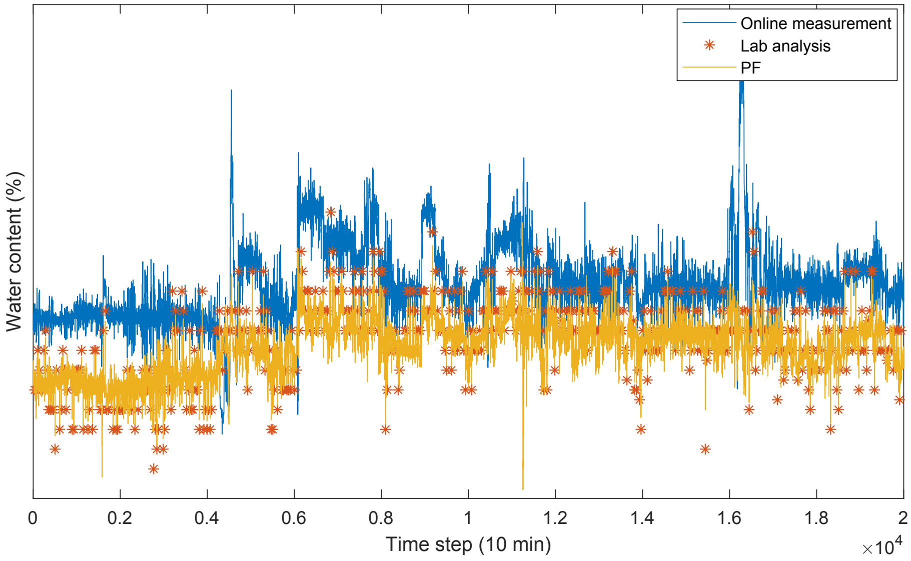

Figure 4.8: State estimation performance of EM1 on a part of the training data

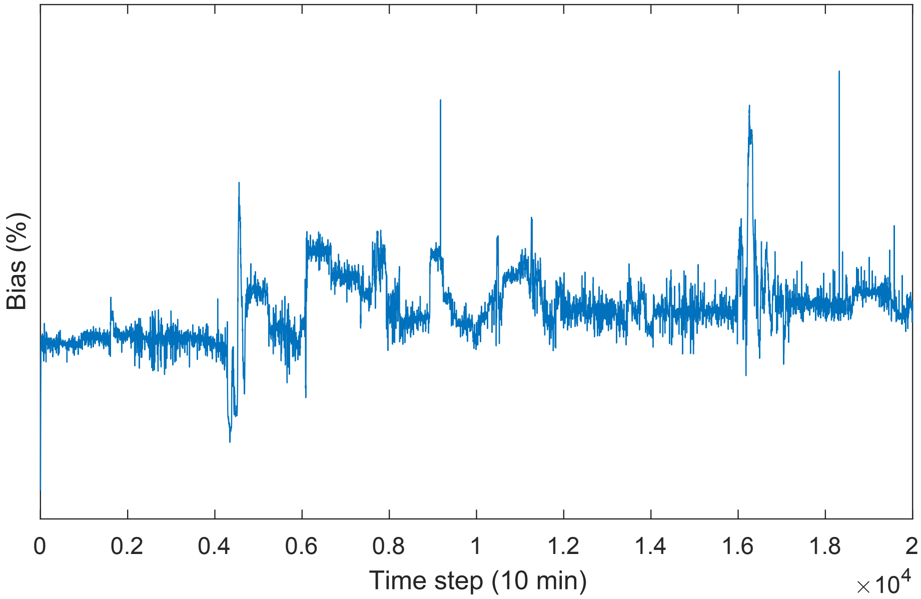

Figure 4.9: Bias estimation of EM1 on a part of the training data

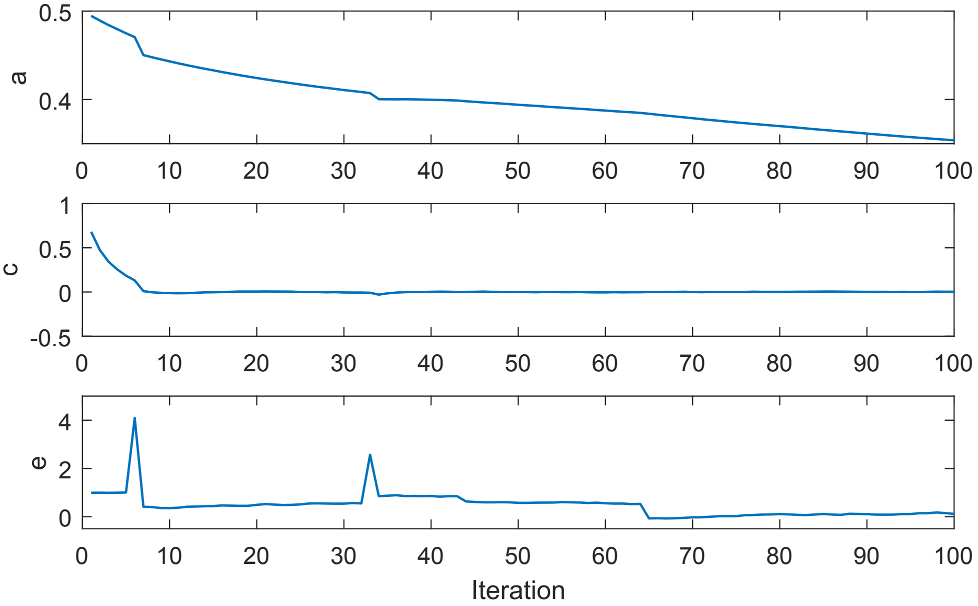

Figure 4.10: Parameter trajectories of EM1 when using a part of the training data

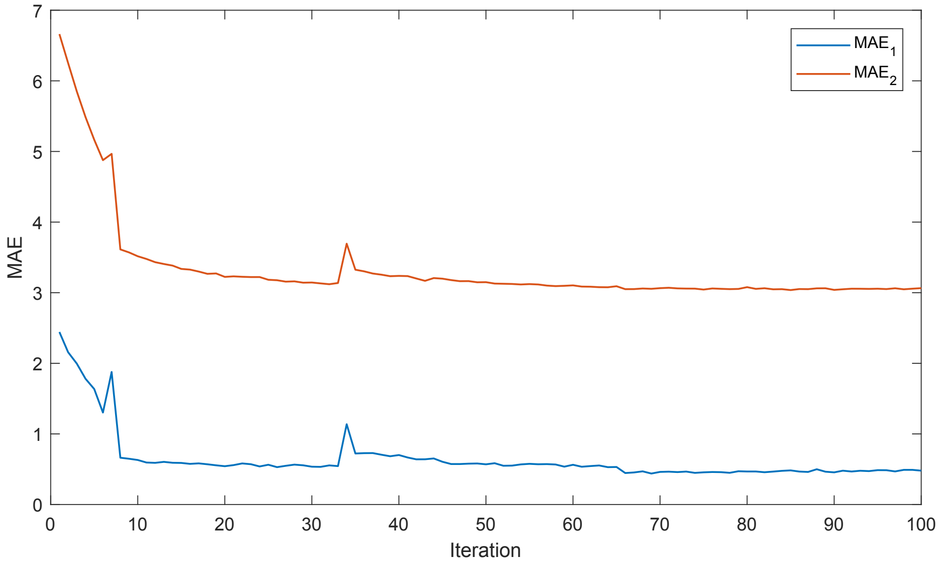

Figure 4.11: MAE trajectories of EM1 when using a part of the training data

Table 4.2: MAE comparison of different algorithms for the water content estimates

	Simulation1			Simulation2		
	Online sensor	EM1	EM2	Online sensor	EM1	EM2
MAE_1	–	3.2104	3.0699	–	0.4799	0.4536
MAE_2	6.1641	4.9642	4.6477	6.0341	3.0642	2.7650

4.4 State estimation for the FWKO vessel

In the previous section, we consider the parameter estimation problem and we also estimate the states using the estimated parameters for the FWKO vessel. But in application, the delay of the laboratory analysis is unknown and the objective is to obtain the estimated water content in real time. Therefore, we focus on the problem of state estimation for the FWKO vessel in this section.

The state space model of the FWKO vessel is described in Section 4.3. The target is to calculate the posterior density $p\left(x_{ks_j} \mid y_{1:s_j-1}, y^f_{s_j:ks_j}, y^m_{ks_j}\right)$ when the laboratory analysis arrives. In Chapter 2, the delay of the slow measurement was assumed to be known. Thus, the first step is to estimate the delay d_j of the current laboratory analysis $y^m_{ks_j}$ in order to apply the state estimation method. Similar to the parameter estimation, the delay can be estimated by applying the MAP approach using the estimated states:

$$\hat{d}_j = \arg\max_{d_j} p\left(d_j \mid \hat{x}_{ks_j-d_j}, y^m_{ks_j}\right) \tag{4.11}$$

where $\hat{x}_{ks_j-d_j}$ is derived from the particle filter using fast online measurements. The posterior density of d_j is derived using Bayes' rule:

$$p\left(d_j = 6t \mid \hat{x}_{ks_j-d_j}, y^m_{ks_j}\right) = \frac{p\left(y^m_{ks_j} \mid \hat{x}_{ks_j-6t}, d_j = 6t\right) p\left(d_j = 6t\right)}{\sum_{n=0}^{2} p\left(y^m_{ks_j} \mid \hat{x}_{ks_j-6n}, d_j = 6n\right) p\left(d_j = 6n\right)} \tag{4.12}$$

After obtaining the estimated delay $\hat{d}_j$, the exact Bayesian method proposed in Chapter 2 can be utilized to estimate the states.

In this section, we use another data set that is sampled in a different time period for testing the performance of the algorithms on data for which they were not trained. The parameters used are obtained from EM2 that considers variable delays. Table 4.3 contains the MAE results for different state estimation methods, where JFM-PF is the particle filter that only uses fast online measurements, EB-PF1 assumes no laboratory analysis delays and

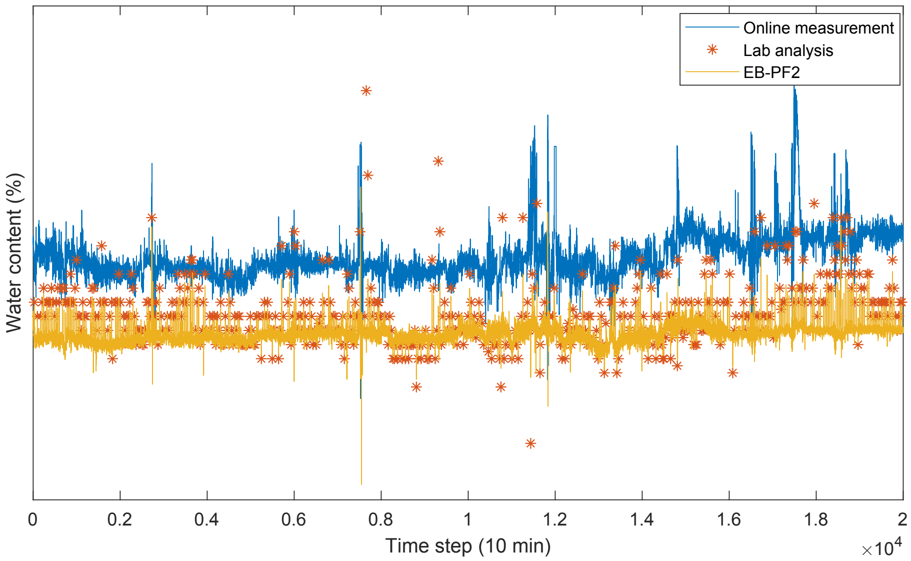

Figure 4.12: State estimation performance on the test data

EB-PF2 assumes variable delays. Furthermore, it is to be noted that the MAE is calculated using the laboratory analysis that is assumed to have the delays as estimated in EB-PF2. As we can see, the performance of the state estimators is much better than the online sensor. The MAE difference between different particle filters are small, which shows the identified model is well fitted to the system. By fusing the laboratory analysis and considering variable delays, we can obtain the most accurate estimates of water contents. Figure 4.12 shows the state estimation result of EB-PF2.

Table 4.3: MAE comparison of various particle filters for the water content estimates

	Online sensor	JFM-PF	EB-PF1 (No delays)	EB-PF2 (Variable delays)
MAE	8.3373	3.8410	2.8644	2.8264

4.5 Conclusions

In this chapter, we solve the state and parameter estimation problem separately for the FWKO vessel. Parameter estimation results show that considering the laboratory analysis delays can improve the estimation accuracy. State estimation performance demonstrates the effectiveness of the identified model and the proposed state estimation approach.

Chapter 5

Conclusions

5.1 Summary of this thesis

In this thesis, we consider the multirate processes with variable measurement delays and integrated states measurements. Specifically, in Chapter 2, we proposed the exact Bayesian approach and the augmented state approach to deal with the state estimation problem for this system. Two approaches are shown to have similar performance by simulation and experiment studies. Moreover, since we fuse fast and slow measurements, the state estimation results are better than those which only use fast measurements. The improvement of state estimation declines as the delay and integral time increase.

In order to estimate the states of the multirate process, an accurate representation of the system is indispensable. Therefore, Chapter 3 studies the parameter estimation problem when the delays of slow measurements are unknown. The parameters are estimated along with the time delays using a particle filter under the framework of expectation maximization algorithm. In the expectation step, smoothing densities in the Q function are approximated using the particle filter. Then, the Q function can be calculated numerically. In the maximization step, the parameters are estimated by differentiating the Q function, while the delays are estimated by applying the maximum a posteriori (MAP) principle. Simulations show that the proposed algorithm can provide more accurate estimates of the parameters as well as the states than that only uses fast measurements.

For further evaluating the proposed methods in Chapter 2 and 3, we discuss the implementations to a FWKO vessel. In this application, the objective is to derive accurate real-time water content estimates. There are online fast measurements with large errors and off-line laboratory analysis for the water content. Thus, the FWKO vessel is a multirate system. It is noted that we do not consider the sampling collection time for laboratory analysis since the time is very small compared to the sample time of the system. The first step is to estimate the parameters and obtain a state space model for the FWKO vessel output emulsion water content in which the algorithm proposed in Chapter 3 is applied. The second step is to estimate the real-time state, which is the water content. However, since the delay is unknown when the laboratory measurement is available, the proposed algorithm in Chapter 2 cannot be directly implemented. Hence, we first estimate the delay using the MAP principle and then apply the exact Bayesian approach to estimate the water content. The results show the effectiveness and applicability of the proposed methods.

5.2 Directions for future work

In this section, we would like to share our perspective on the directions that are worthy of future investigation based on the current results.

- In Chapter 2, we develop two algorithms and implement both the PF and UKF to solve the nonlinear state estimation problem for the multirate system. In fact, the two proposed algorithms are generalizable and many Bayes filters can be applied, such as Kalman filter (KF) for the linear case, and extended Kalman filter (EKF) and unscented particle filter (UPF) for the nonlinear case. By trying different filters, we can find the best state estimates for a particular system.
- We have studied the normal state estimation problem for the multirate system with variable measurement delays. There are still abnormal situations to be considered, for instance, states with multi-modal distributions, states with constraints, etc.
- In Chapter 3, we estimate the parameters for the dynamic state space model using the EM algorithm where we obtain a single point estimate for each parameter. We can also apply the variational Bayesian (VB) algorithm to estimate the distributions of parameters. In addition, other model representations for the multirate system can be discussed.
- In the application example of Chapter 4, the laboratory measurement is quantized and only even integers are recorded. If higher resolution laboratory measurements can be

obtained in the future, we will be able to derive more accurate state and parameter estimations for the FWKO vessel.

References

[1] H. H. Afshari, S. A. Gadsden, and S. Habibi, “Gaussian filters for parameter and state estimation: A general review of theory and recent trends,” *Signal Processing*, vol. 135, pp. 218–238, 2017.

[2] C. Andrieu, A. Doucet, and V. B. Tadić, “Online sampling for parameter estimation in general state space models,” *IFAC Proceedings Volumes*, vol. 36, no. 16, pp. 1275–1280, 2003.

[3] C. Andrieu, A. Doucet, S. S. Singh, and V. B. Tadic, “Particle methods for change detection, system identification, and control,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 423–438, 2004.

[4] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking,” *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[5] S. Asprey and S. Macchietto, “Designing robust optimal dynamic experiments,” *Journal of Process Control*, vol. 12, no. 4, pp. 545–556, 2002.

[6] Y. Bar-Shalom and L. Campo, “The effect of the common process noise on the two-sensor fused-track covariance,” *IEEE Transactions on Aerospace and Electronic Systems*, no. 6, pp. 803–805, 1986.

[7] S. Borman, “The expectation maximization algorithm-a short tutorial,” 2004.

[8] J. Chen, B. Huang, F. Ding, and Y. Gu, “Variational bayesian approach for arx systems with missing observations and varying time-delays,” *Automatica*, vol. 94, pp. 194–204, 2018.

[9] S. B. Chitralekha, J. Prakash, H. Raghavan, R. Gopaluni, and S. L. Shah, “A comparison of simultaneous state and parameter estimation schemes for a continuous fermentor reactor,” *Journal of Process Control*, vol. 20, no. 8, pp. 934–943, 2010.

[10] M. P. Deisenroth and H. Ohlsson, “A general perspective on gaussian filtering and smoothing: Explaining current and deriving new algorithms,” in *American Control Conference (ACC), 2011*, IEEE, 2011, pp. 1807–1812.

[11] P. Del Moral *et al.*, "Measure-valued processes and interacting particle systems. application to nonlinear filtering problems," *The Annals of Applied Probability*, vol. 8, no. 2, pp. 438–495, 1998.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[13] J. Deng and B. Huang, "Identification of nonlinear parameter varying systems with missing output data," *AIChE Journal*, vol. 58, no. 11, pp. 3454–3467, 2012.

[14] J. Deng, L. Xie, L. Chen, S. Khatibisepehr, B. Huang, F. Xu, and A. Espejo, "Development and industrial application of soft sensors with on-line bayesian model updating strategy," *Journal of Process Control*, vol. 23, no. 3, pp. 317–325, 2013.

[15] R. Douc and O. Cappé, "Comparison of resampling schemes for particle filtering," in *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.*, IEEE, 2005, pp. 64–69.

[16] A. Doucet, "On sequential simulation-based methods for bayesian filtering," 1998.

[17] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Statistics and computing*, vol. 10, no. 3, pp. 197–208, 2000.

[18] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump markov linear systems," *IEEE Transactions on signal processing*, vol. 49, no. 3, pp. 613–624, 2001.

[19] A. Fatehi, "Modeling hybrid tank pilot plant using first principle model and experimental data," *Dept. Chem. Mater. Eng., Univ. Alberta, Edmonton, AB, Canada, Tech. Rep*, 2015.

[20] A. Fatehi and B. Huang, "Kalman filtering approach to multi-rate information fusion in the presence of irregular sampling rate and variable measurement delay," *Journal of Process Control*, vol. 53, pp. 15–25, 2017.

[21] ——, "State estimation and fusion in the presence of integrated measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 9, pp. 2490–2499, 2017.

[22] *Free water knockout (FWKO)*. [Online]. Available: `https://www.petropedia.com/definition/1681/free-water-knockout-fwko`.

[23] J. Gao and C. J. Harris, "Some remarks on kalman filters for the multisensor fusion," *Information Fusion*, vol. 3, no. 3, pp. 191–201, 2002.

[24] Z. Ghahramani and S. T. Roweis, "Learning nonlinear dynamical systems using an em algorithm," in *Advances in neural information processing systems*, 1999, pp. 431–437.

[25] G. C. Goodwin and J. C. Aguero, “Approximate em algorithms for parameter and state estimation in nonlinear stochastic models,” in *Proceedings of the 44th IEEE Conference on Decision and Control*, IEEE, 2005, pp. 368–373.

[26] A. Gopalakrishnan, N. S. Kaisare, and S. Narasimhan, “Incorporating delayed and infrequent measurements in extended kalman filter based nonlinear state estimation,” *Journal of Process Control*, vol. 21, no. 1, pp. 119–129, 2011.

[27] R. B. Gopaluni, “Nonlinear system identification under missing observations: The case of unknown model structure,” *Journal of Process Control*, vol. 20, no. 3, pp. 314–324, 2010.

[28] R. Gopaluni, “A particle filter approach to identification of nonlinear processes under missing observations,” *The Canadian Journal of Chemical Engineering*, vol. 86, no. 6, pp. 1081–1092, 2008.

[29] N. J. Gordon, D. J. Salmond, and A. F. Smith, “Novel approach to nonlinear/non-gaussian bayesian state estimation,” in *IEE proceedings F (radar and signal processing)*, IET, vol. 140, 1993, pp. 107–113.

[30] Y. Guo and B. Huang, “State estimation incorporating infrequent, delayed and integral measurements,” *Automatica*, vol. 58, pp. 32–38, 2015.

[31] F. Gustafsson, “Particle filter theory and practice with positioning applications,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.

[32] M. A. Henson and D. E. Seborg, *Nonlinear process control.* Prentice Hall PTR Upper Saddle River, New Jersey, 1997, pp. 198–206.

[33] S. J. Julier and J. K. Uhlmann, “New extension of the kalman filter to nonlinear systems,” in *Signal processing, sensor fusion, and target recognition VI*, International Society for Optics and Photonics, vol. 3068, 1997, pp. 182–194.

[34] ——, “Unscented filtering and nonlinear estimation,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[35] R. Kandepu, B. Huang, L. Imsland, and B. Foss, “Comparative study of state estimation of fuel cell hybrid system using ukf and ekf,” in *2007 IEEE International Conference on Control and Automation*, IEEE, 2007, pp. 1162–1167.

[36] J. S. Liu and R. Chen, “Sequential monte carlo methods for dynamic systems,” *Journal of the American statistical association*, vol. 93, no. 443, pp. 1032–1044, 1998.

[37] B. Ninness and S. Gibson, “Robust and simple algorithms for maximum likelihood estimation of multivariable systems,” *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 343–348, 2002.

[38] L. E. Olivier, B. Huang, and I. K. Craig, "Dual particle filters for state and parameter estimation with application to a run-of-mine ore mill," *Journal of Process Control*, vol. 22, no. 4, pp. 710–717, 2012.

[39] B. L. Pence, H. K. Fathy, and J. L. Stein, "Recursive maximum likelihood parameter estimation for state space systems using polynomial chaos theory," *Automatica*, vol. 47, no. 11, pp. 2420–2424, 2011.

[40] X. Shao, B. Huang, and J. M. Lee, "Constrained bayesian state estimation–a comparative study and a new particle filter based approach," *Journal of Process Control*, vol. 20, no. 2, pp. 143–157, 2010.

[41] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting using the em algorithm," *Journal of time series analysis*, vol. 3, no. 4, pp. 253–264, 1982.

[42] ——, *Time series analysis and its applications: with R examples.* Springer, 2017.

[43] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches.* John Wiley & Sons, 2006, p. 433.

[44] A. Smith, *Sequential Monte Carlo methods in practice.* Springer Science & Business Media, 2013.

[45] H. W. Sorenson, "Least-squares estimation: From gauss to kalman," *IEEE spectrum*, vol. 7, no. 7, pp. 63–68, 1970.

[46] G. Steiner, D. Watzenig, C. Magele, and U. Baumgartner, "Statistical robust design using the unscented transformation," *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, vol. 24, no. 2, pp. 606–619, 2005.

[47] R. Van Der Merwe, A. Doucet, N. De Freitas, and E. A. Wan, "The unscented particle filter," in *Advances in neural information processing systems*, 2001, pp. 584–590.

[48] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*, IEEE, 2000, pp. 153–158.

[49] *What is a free water knockout (FWKO) vessel?* [Online]. Available: `https://www.crimtech.com/what-is-a-free-water-knockout-fwko-vessel/`.

[50] Wikipedia contributors, *Average absolute deviation*, 2019. [Online]. Available: `https://en.wikipedia.org/wiki/Average_absolute_deviation`.

[51] L. Xie, H. Yang, and B. Huang, "Fir model identification of multirate processes with random delays using em algorithm," *AIChE Journal*, vol. 59, no. 11, pp. 4124–4132, 2013.

[52] S. Zhang and Y. Bar-Shalom, "Out-of-sequence measurement processing for particle filter: Exact bayesian solution," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 2818–2831, 2012.

[53] Y. Zhao, A. Fatehi, and B. Huang, "Robust estimation of arx models with time varying time delays using variational bayesian approach," *IEEE transactions on cybernetics*, vol. 48, no. 2, pp. 532–542, 2018.