

EM ALGORITHM FOR ELECTRICITY POOL PRICE PREDICTION AND
ERRORS-IN-VARIABLES PROCESS IDENTIFICATION

by

Ouyang Wu

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

©Ouyang Wu, 2015

Abstract

In this thesis, under the EM algorithm framework, a multiple model approach is developed towards electricity price prediction, and the identification problem for errors-in-variables (EIV) systems is studied.

Alberta's electricity price, which shows high volatility and erratic nature, is considered as an example of a nonlinear process. A Markov regime-switching model is applied to predict the price using the local models through the investigations of characteristics of the pool price sequence. The expectation maximization (EM) algorithm is applied to solve the maximum likelihood (ML) estimation problem for model parameters, and several initialization methods are proposed to generate the initial values for the EM algorithm. The validations are presented to verify the proposed approach, which demonstrate an improvement on the existing price prediction for the range of high electricity prices.

A dynamic system that has both input and output measurement errors is considered as an errors-in-variables (EIV) system. Employment of traditional identification strategies for EIV systems will result in biased estimates and inaccurate estimation of system parameters. EIV approaches such as the subspace EIV method has been proposed, but the subspace approach does not possess the optimality such as ML estimation. However, the direct application of ML approach for EIV model parameter estimation can lead to intractable solutions. In this work, we assume a dynamic model for noise-free input and propose to solve the ML problem using the EM algorithm.

To identify industrial nonlinear EIV processes that operate along an operating trajectory, a linear parameter varying (LPV) EIV model is proposed to approximate the global models. The EM algorithm is used to solve the ML estimation for LPV EIV model parameters. Various numerical simulations and pilot-scale experiments are used to demonstrate the effectiveness of the proposed approach.

Preface

Chapter 2 of this thesis is the extension of the conference paper published as O. Wu, T. Liu, B. Huang, F. Forbes, *Predicting Electricity Pool Prices Using Hidden Markov Models*, IFAC-PapersOnLine (2015). In the paper, I was responsible for the extension of the theory and algorithm developments based on the work of Tianbo Liu, validation studies, and manuscript composition. Dr. Biao Huang and Dr. Fraser Forbes were the supervisory authors and were involved with research discussions and manuscript composition. The extension in Chapter 2 includes the solution of the time-varying problem, improving the initialization for the EM algorithm, and formulating the two-hour ahead prediction with a number of validation studies.

Acknowledgements

Another Edmonton's winter comes kindly, and I am looking back to the journey of my master program. It is a remarkable one to me that I not only gained scientific research experience, but also a professional attitude and a responsive heart.

It gives me great pleasure in acknowledging the support from my supervisors Dr. Biao Huang and Dr. Fraser Forbes. They are always helpful and patient to me, and I am really appreciated and grateful to their guidance during my graduate study.

It's my honor to be one in the Computer Process Control (CPC) group where we are free to ask questions, share ideas and benefit from this excellent group. Among this group, I would like to thank Hariprasad Kodamana specially for his suggestions and discussions about the errors-in-variables problem. The help from Fangwei Xu, Hao Chen and Yi Zheng in the on-going project is greatly appreciated. I would give my sincere thanks to my colleagues who helped me, they are: Tianbo Liu, Kangkang Zhang, Lei Chen, Yaojie Lu, Ming Ma, Yuan Yuan, Yujia Zhao, Swanand Khare, Shima Khatibisepehr, Nima Sammknejad, Alireza Fatehi, Mohammad Rashedi, Elham Naghoosi, Anahita Sadeghian, Fadi Ibrahim, Rishik Ranjan and many others from CPC group. I would also like to thank my spiritual group friends for helping me seek the meaning of life, they are: Boon Chew, Lang Liu, Thomas Xi and Cherry Zhang. Moreover, I appreciate the friendships with those who are: Chen Li, Ran Li, Xiaodong Xu, Shunyi Zhao, Bin Xia, Yang Zhou, Jian Guo, Yuyu Yao, Jing Zhang, Ruomu Tan, Haoluan Wang, Wenda Xiao, Yipei Styles, Su Liu, Xunyuan Yin, Wenhan Shen, Tianrui An, Wenhai Zhang and many else.

This thesis would not have been possible without the financial support from NSERC of Canada and Alberta Innovates Technology Futures.

Finally, I owe my deepest gratitude to my parents for their unconditional love and support.

Contents

1	Introduction	1
1.1	Background	1
1.2	Motivation	1
1.3	Thesis contributions	2
1.4	Thesis outline	3
2	Peak pool price prediction	4
2.1	Introduction	4
2.2	Price prediction algorithm	7
2.2.1	Input variables selection and data preprocessing	7
2.2.2	Identifying pool price model using the EM algorithm	10
2.2.3	Initialization for the EM algorithm	13
2.3	Validation studies	18
2.3.1	One-hour ahead pool price prediction	19
2.3.2	Two-hour ahead pool price prediction	22
2.4	Conclusions	23
3	Identification of linear dynamic errors-in-variables systems with a dynamic process for noise-free inputs using EM algorithm	30
3.1	Introduction	30
3.2	Problem formulation	33
3.3	EM algorithm for linear dynamic EIV model identification	34
3.3.1	ML method for the EIV system using the EM algorithm	35
3.3.2	Smoothing for state space model with colored stochastic inputs	39

3.3.3	Computation of the conditional expectation terms in the Q function	43
3.3.4	Algorithm	45
3.4	Validations	45
3.4.1	Simulation: Example 1	47
3.4.2	Simulation: Example 2	48
3.4.3	Experiment: Tank system	51
3.5	Conclusions	53
4	Identification of LPV Errors-In-Variables systems with a dynamic process for noise-free inputs using EM algorithm	55
4.1	Introduction	55
4.2	Problem formulation	57
4.3	The LPV EIV system identification	59
4.3.1	Revisit of the EM algorithm	59
4.3.2	ML estimation of the LPV EIV model using the EM algorithm	60
4.3.3	Smoother for the LPV state space model	64
4.3.4	Computation of the conditional expectation terms in the Q function	69
4.3.5	Algorithm	71
4.4	Validations	71
4.4.1	Numerical example 1: a first-order continuous LPV system . .	72
4.4.2	Numerical example 2: CSTR	74
4.4.3	Experimental example	76
4.5	Conclusions	80
5	Conclusions	82
5.1	Summary and conclusions of the thesis	82
5.2	Directions for future work	83
	Bibliography	85

A	Variance estimation for pool price prediction	90
A.1	Supplemented EM algorithm	90
A.2	Uncertainty estimation for pool price prediction	91
B	Detailed derivations for section 3.3.3	93
B.1	Detailed derivation of equation (3.43)	93
B.2	Detailed derivation of equation (3.45)	95
B.3	Detailed derivation of equation (3.47)	97
C	Smoothing step for the lag-one covariance of LPV EIV state space model	100

List of Tables

2.1	Training data fitting in August 2013	20
2.2	Prediction in August 2014 by training data August 2013	21
2.3	Training data fitting in January 2013	22
2.4	Prediction in January 2014 by training data January 2013	23
2.5	High-price range prediction in January 2014 by training data January 2013	24
2.6	High-price range prediction in August 2014 by training data August 2013	24
2.7	Prediction in January 2014 by different training data	25
2.8	Prediction in August 2014 by different training data	25
2.9	High-price range prediction in January 2014 by different training data	26
2.10	High-price range prediction in August 2014 by different training data	27
2.11	High-price range prediction in different months using ‘similar month’ rule	28
2.12	High-price range two-hour ahead prediction in different months using ‘similar month’ rule	29
3.1	Mean prediction costs of the simulation results	51
3.2	Average MSEs of the simulation results	53
4.1	Model parameters of the CSTR process	75
4.2	Notations of multi-tank system	78

List of Figures

2.1	Actual pool price and AESO's pool price forecast (Dec 2013)	5
2.2	Actual demand and AESO's demand forecast (Dec 2013)	6
2.3	Preprocessed time sequence (Dec 2013)	8
2.4	Typical hourly offer curve	9
2.5	Preprocessed forecast demand (Dec 2013)	9
2.6	Preprocessed actual demand (Dec 2013)	10
2.7	The hidden Markov model diagram	13
2.8	Triangle representation components	15
2.9	Continuous hidden Markov model with hidden cluster m_k	16
3.1	The errors-in-variables process diagram	31
3.2	The errors-in-variables process with input generation diagram	32
3.3	The process variables relationship diagram	39
3.4	Prediction error cost for first order EIV system (input variance: 5%) .	48
3.5	Prediction error cost for first order EIV system (input variance: 10%)	48
3.6	State estimation for the first-order EIV system (input variance: 10%)	49
3.7	Prediction error cost for second order EIV system (input variance: 5%)	49
3.8	Prediction error cost for second order EIV system (input variance: 10%)	50
3.9	State estimation for the second-order EIV system (input variance: 10%)	50
3.10	Schematic Diagram of the tank system	52
3.11	Input and output data	52
3.12	Simulated noise-corrupted input: (1) 5% of input variance; (2) 10% of input variance	53
3.13	Validation results: 5% input noise	53

3.14	Validation results: 10% input noise	54
4.1	(1) The noise-corrupted input u_k and the PF estimations; (2) The operating trajectory for the numerical example 1	73
4.2	(1) The self validation result for the numerical example 1; (2) The model identity probabilities of the local models	74
4.3	(1) The cross validation result for the numerical example 1; (2) The operation trajectory for the cross validation data	74
4.4	(1) The operating trajectory for the numerical example 2; (2) The PF estimation of the nonlinear process output	76
4.5	(1) The self validation results for the numerical example 2; (2) The model identity probabilities of the local models	77
4.6	(1) The cross validation results for the numerical example 2; (2) The operating trajectory for the cross validation data	77
4.7	Diagram of multi-tank system	78
4.8	(1) The noise-corrupted input and the PF estimations; (2) The operation trajectory for the experimental example	79
4.9	(1) The self validation results for the experimental example; (2) The model identity probabilities of the local models	80
4.10	(1) The cross validation results for the experimental example; (2) The operating trajectory for the cross validation data	80

Chapter 1

Introduction

1.1 Background

Predictive model development is always an attractive topic in research since the model predictions help to develop better strategies and make the right decision for the future. In daily life, the fluctuating prices of commodities are always drawing customers' attention. For example, the electricity price in Alberta, also called pool price, fluctuates from 0 \$/MWh to 1000 \$/MWh depending on the market. A good pool price prediction could help the suppliers and consumers make appropriate strategies to gain profits or reduce costs.

With the development of the advanced control strategies, model-based control methods are commonly applied in complex industrial processes, which require models of good predictive capability for the key physical variables of the processes. On the other hand, the physical variables in practical processes are always noise-corrupted due to the measurement error, which leads to the so-called errors-in-variables (EIV) problem. The errors-in-variables behavior, where both the input and output measurements are corrupted with noise, presents difficulties and challenges in system identification.

1.2 Motivation

The nonlinearity and noise uncertainty problems are the main challenges in system identification. In Alberta's electricity market, the pool price shows high volatility and erratic nature, and the price rating mechanism presents the time-varying and non-

linear feature according to the real-time power demands [1]. A statistical approach, namely, the Markov regime-switching model has been proposed to approximate the pool price process [2], and the model parameters are calibrated using the expectation maximization (EM) algorithm for the maximum likelihood estimation (MLE) [3]. The proposed Markov regime switching model in a multiple-model framework outperforms the existing predictions from Alberta Electric System Operator (AESO).

To predict the pool price, the forecast demands from AESO are used as the input. However, the variable that decides the price is the real-time demand, and it exhibits some differences from the forecast demands. In industrial processes, this phenomenon also happens when the input variable is measured with noise, and the noise-free input is unobservable. This class of system identification problems are referred as EIV problems in literature. Identifying an EIV system with traditional methods, which only consider the noise-corrupted output, would result in biased estimations of the model parameters due to the non-inclusion of the input measurement uncertainties [4]. The existing EIV methods also have some limitations. For example, though the subspace EIV methods have the advantage of numerical simplicity compared to iterative methods, the estimation accuracy has much scope for improvement [5].

To overcome nonlinearity and to have a smooth operation, most of the industrial processes will have an operating trajectory, designed in a structured way, depending on some known variables called scheduling variables. In the pool price prediction model, some characteristics of the pool price such as periodic patterns are employed as the input to mitigate the effect of nonlinearities. Since nonlinear identification is computationally complex, a feasible option in modeling the nonlinear EIV process is to develop a linear parameter varying (LPV) model over the operating trajectory, where the parameter variation will depend on the scheduling variables.

1.3 Thesis contributions

The main contributions of this thesis are the development of identification approaches using a linear EIV state space model and an LPV EIV state space model to solve the difficulties in modeling EIV behaviors, nonlinearity, and an application of multiple model approach on Alberta's pool price prediction. Specifically, the contributions are

summarized as follows:

1. Modeling of Alberta's pool price process using a Markov regime-switching model and providing prediction to catch spike behaviors of pool price.
2. Proposing two specific initialization methods when applying EM algorithm to estimate parameters of a pool price predictive model, which helps to the convergence of EM algorithm.
3. Proposing a 'similar month' training data selection strategy to tackle pool price time-varying parameter problems, and validating improved peak pool price predictions on different periods throughout a year.
4. Development of a linear dynamic EIV system identification through EM algorithm and proposing a smoother for state space models with colored inputs in E-steps of EM algorithm.
5. Identifying LPV EIV state space models using EM algorithm and proposing a smoother for LPV state space models.

1.4 Thesis outline

This thesis is organized in a paper format, and literature reviews are distributed in each chapter. In Chapter 2, a peak electricity price prediction approach for Alberta's electricity market is developed, and EM algorithm is employed for model calibrations. In Chapter 3, a linear EIV state space model with input dynamics is identified using EM algorithm. In Chapter 4, a linear EIV state space model is extended to an LPV EIV state space model for identifying nonlinear EIV processes, and the identification problem is formulated and solved using EM algorithm.

Chapter 2

Peak pool price prediction

2.1 Introduction

In recent decades, electricity market deregulation has become a world-wide trend with promising application in North America and Europe. By introducing competition, it is expected that the efficiency of the electricity market is improved, thus providing opportunities and also presenting challenges to both the generators and the consumers. As a result, electricity price prediction has become an important problem in deregulated markets.

In 1996, Alberta's electricity market began to evolve as a deregulated market and full deregulation was established in 2001 [1]. According to the law, wholesale electrical energy generated in Alberta, which is not consumed on site, must flow through a power pool which is operated by Alberta's independent system operator called Alberta Electric System Operator (AESO). Thus, the power pool is Alberta's wholesale spot electricity market, and the hourly electricity price for the power pool is called pool price. The power pool pricing characteristics are summarized in [1] and [6]. Apparently, there are on-peak and off-peak electricity price patterns, and the on-peak period is often from 8:00 to 21:00 during weekdays. Also, there are pronounced periodic effects for the pool price, such as daily, intra-daily and weekly repeating patterns, or even monthly repeating patterns. For example, prices vary with demands in a day, which presents an hourly pattern. Another characteristic of the power pool is price spikes. Since spikes and dips in demand and supply are common, the pool prices may be quite volatile in certain periods. For example, unplanned outages along transmission lines can drive the pool price to a high level, such as

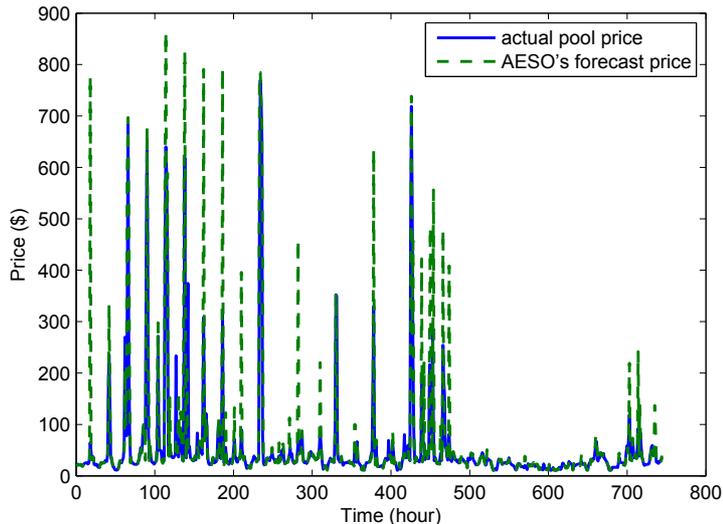


Figure 2.1: Actual pool price and AESO's pool price forecast (Dec 2013)

500 \$/MWh or more. AESO provides a two-hour ahead pool price forecast and a day-ahead demand forecast as Figure 2.1 and Figure 2.2 presented. As the historical data shows, the two-hour ahead pool price forecast can have large prediction errors compared to the actual pool price, especially when prices spike. One reason for these errors is that generators are free to modify their supply offers two hours ahead, which could result in the dispatch level in the next two hour period being quite different from the current dispatch level. In this case, a prediction model for Alberta's pool price with an improved forecasting performance would benefit customers with their decisions on electricity consumption.

There are a number of challenges in the electricity price prediction due to their high volatility and erratic nature. According to Weron [7], approaches such as statistical methods, quantitative models and nonlinear techniques have been used in electricity price modeling. Among these approaches, statistical methods have advantages of simplicity and analytical tractability. To deal with the price volatility, regime-switching models have been proposed to model switches between different states, such as normal price and spike price regimes. In [8], a Markov regime-switching model is applied to electricity prices in the United States and Australian markets, and the results confirmed the existence of two states with different means and variances. Huisman

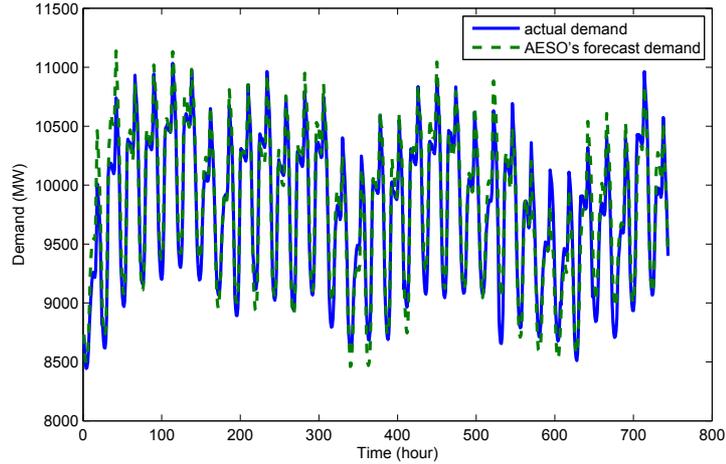


Figure 2.2: Actual demand and AESO's demand forecast (Dec 2013)

and Mahieu [2] proposed to use Markov regime-switching for modeling price spikes in European electricity markets which have three states, including normal electricity price dynamics, a jump state describing sudden increases or decreases, and a state describing a recovery from a jump state to a normal state. Markov regime-switching models have also been used in modeling of nonlinear processes, and the expectation maximization (EM) algorithm is employed for the parameter estimations [9], [10], [3]. The EM algorithm is an efficient tool to apply maximum likelihood estimation when there are missing data. However, good initial values are required for results to accurately converge. The Hidden Markov Model (HMM) approach has been widely applied to model systems with switched dynamics, for example, biological and financial fields [11], [12]. In this study, the HMM approach is used to produce the initial values for the Markov regime-switching approach for electricity price prediction.

In this chapter, a Markov regime-switching model is applied for predicting the pool price in Alberta's electricity market. Model calibration based on the EM algorithm with the proposed initialization methods, is presented in Section 2.2; the validation studies are presented in Section 2.3.

2.2 Price prediction algorithm

In this section, a Markov regime-switching model for Alberta's pool price prediction is formulated. Parameter estimation for the switching model is developed based on the EM algorithm, and the initialization problem for the EM algorithm is solved through hidden Markov model approaches.

2.2.1 Input variables selection and data preprocessing

We consider the autoregressive exogenous (ARX) model as the local model structure due to its simplicity and low computational requirements. To select the input variables, the following characteristics of the pool price are considered [3], [6], [7]: 1) the pool price in Alberta shows a strong periodic behavior; 2) the AESO's forecast pool price reflects the fluctuation of future electricity pool price; 3) the day ahead forecast demand by AESO affects the bidding results of the generators; 4) the actual demand is correlated with the pool price at the same time instant; 5) the actual demand is unavailable at the time of prediction, so the historical data for the immediate past are considered as the best available alternative for prediction. Therefore, the time sequence, the real-time forecast pool price by AESO, the historical data for actual system demand, the real-time day ahead forecast demand by AESO, and a dummy variable denoting weekday or weekend are chosen as the input variables to predict the real time pool price.

The time sequence is preprocessed to build a linear correlation with the actual pool price. First, the time sequence is transformed to be periodic with respect to a 24 hour time clock to appropriately reflect the periodic pattern of the pool price. Then the weights for on-peak and off-peak hours are calculated based on the following weighting expressions [3]:

$$\begin{aligned} F(k) &= K(k) \cdot \exp\left(\frac{(k - k_p)^2}{2\sigma_p^2}\right) \\ k_p &\sim PMF \\ K(k) &= f(P(I_k|C_{obs}, \Theta^{old})) \end{aligned} \tag{2.1}$$

where $F(k)$ is the preprocessed time sequence, weighted by peak-price magnitude $K(k)$ and Gaussian function $\exp(\frac{(k-k_p)^2}{2\sigma_p^2})$; σ_p^2 is the tuning parameter; k_p is the hourly

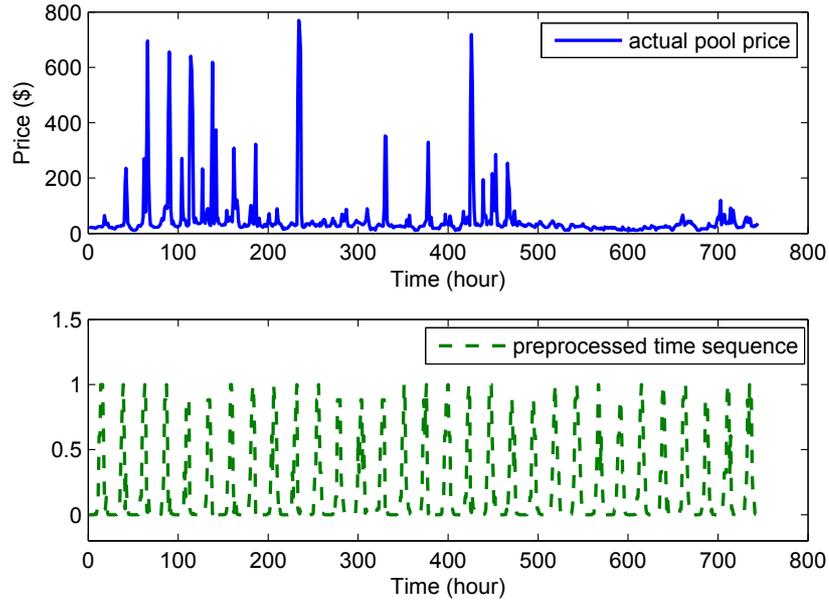


Figure 2.3: Preprocessed time sequence (Dec 2013)

time instant with the peak price in a day and is a random variable with a probability mass function (PMF) based on historical data; the peak-price magnitude $K(k)$ is a function of the posterior state probability at time instant k and relates the possibility of the price that is governed by a hidden price state. The preprocessed time sequence is presented in Figure 2.3.

Preprocessing of system demands is carried out considering the electricity market mechanism. All the electricity generators in Alberta submit their offers to the power pool with their available capacity and desired prices. These offers are ranked from lowest to highest in price to meet the system demands and the high-price surplus capacities will be dispatched. The hourly supply offer curve can be drawn as a piecewise function as in Figure 2.4 [1]. One of supply offer curve feature is that there is a high price triggered at the upper end of the curve, and this is because these offers are from some emergency capacity of plants which requires higher maintenance costs. Therefore, the demand time series can be processed in such a way that the portions over certain demand are emphasized, while the low demand portions are flattened. The preprocessed curves are presented in Figure 2.5 and Figure 2.6.

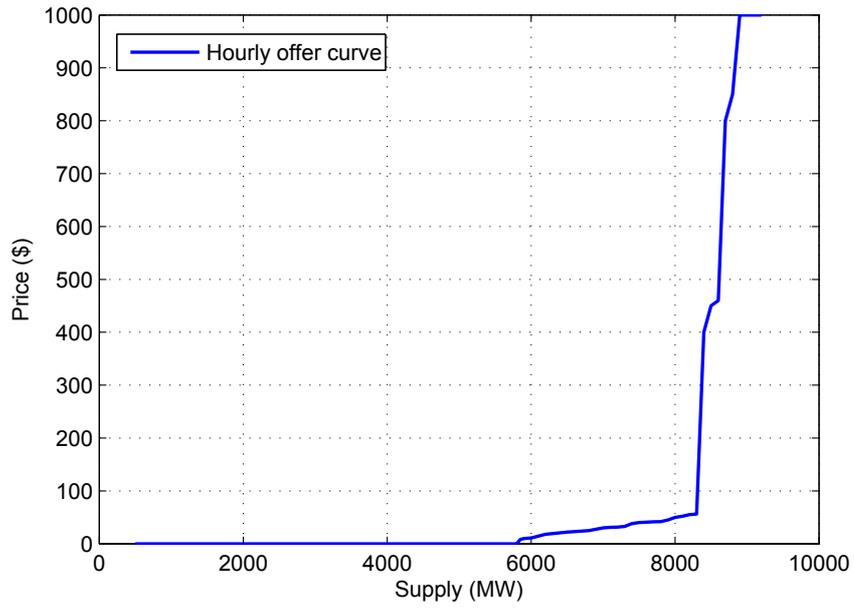


Figure 2.4: Typical hourly offer curve

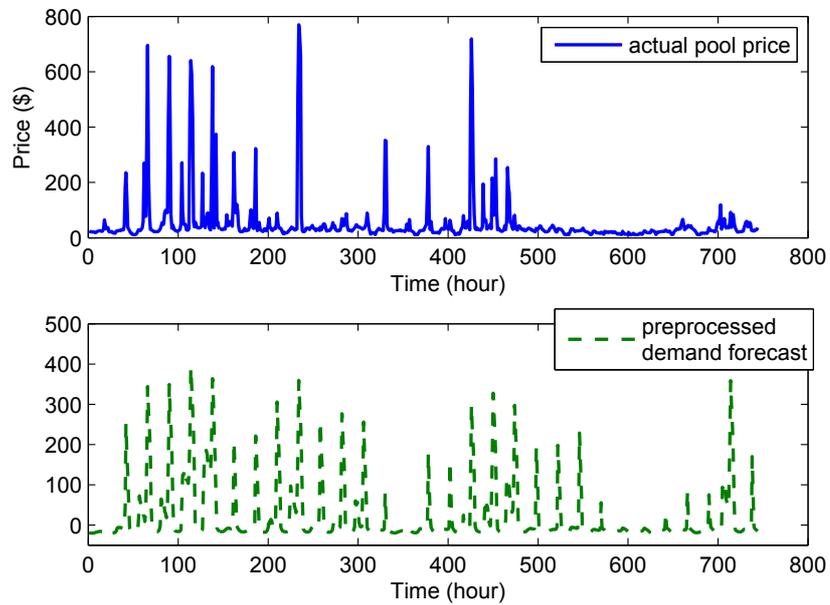


Figure 2.5: Preprocessed forecast demand (Dec 2013)

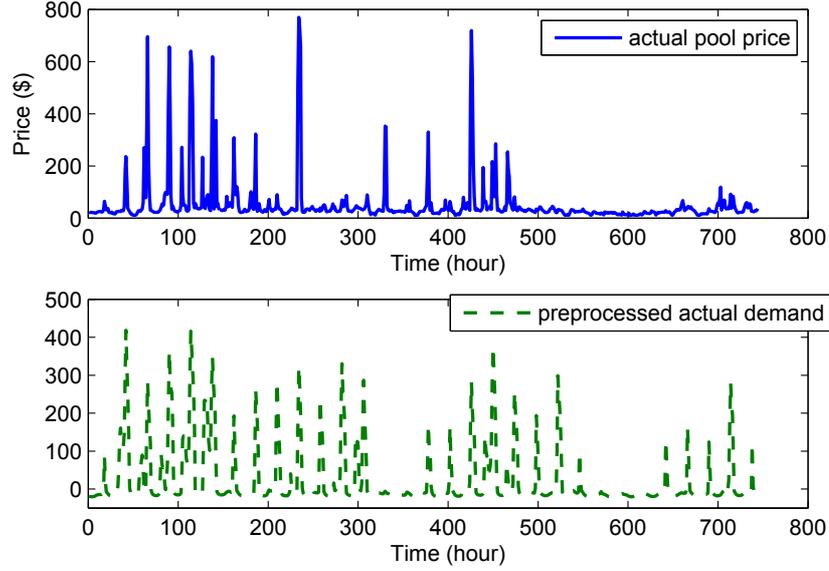


Figure 2.6: Preprocessed actual demand (Dec 2013)

2.2.2 Identifying pool price model using the EM algorithm

For each certain hidden state or regime I_k , it is assumed that the input-output relationship follows the local autoregressive exogenous (ARX) model, as below:

$$y_k = \phi_k^T \theta_{I_k} + v_k, \quad \forall k = 1, 2, \dots, N \quad (2.2)$$

where, $I_{1:k-1}$ denotes $\{I_1, I_2, \dots, I_{k-1}\}$; N is the data length; the regime sequence $\{I_k\}$ is assumed to follow a first-order Markov chain $P(I_k | I_{1:k-1}) = P(I_k | I_{k-1})$; ϕ_k is the regressor and can be expressed as:

$$\phi_k = [1, y_{k-1}, y_{k-2}, \dots, y_{k-n_a}, u_{k-1}^T, u_{k-2}^T, \dots, u_{k-n_b}^T]^T$$

n_a and n_b are orders of denominator and numerator for the ARX model in transfer function form; θ_{I_k} are the parameters for the local sub-model with hidden state $I_k \in \{1, 2, \dots, M\}$ that indicates the model identity, and M is the number of local models; v_k is the output noise assumed to be Gaussian distributed with zero mean and variance σ^2 ; y_k is the actual pool price at time instant k ; $\{u_k\}$ is the time series vector for inputs such as preprocessed demands, AESO's price forecast, preprocessed time sequence and dummy variable.

To estimate the model parameters from input and output data with unknown switching state $\{I_k\}$, the Expectation Maximization algorithm is used for solving the maximum likelihood estimation problem with hidden variables. By calculating the conditional expectation of the complete-data log likelihood function and maximizing the so-called Q function iteratively, the likelihood keeps increasing until it reaches a maximum [13]. The complete-data log likelihood function is given by:

$$\begin{aligned} \log P(y_{1:N}, u_{1:N}, I_{1:N}|\Theta) &= \log \prod_{k=1}^N P(y_k|y_{1:k-1}, u_{1:k}, I_{1:k}, \Theta) P(I_k|y_{1:k-1}, u_{1:k}, I_{1:k-1}, \Theta) \cdot \\ &\quad P(u_k|y_{1:k-1}, u_{1:k-1}, I_{1:k-1}, \Theta) \\ &= \sum_{k=1}^N [\log P(y_k|\phi_k, \theta_{I_k}) + \log P(u_k)] + \sum_{k=2}^N \log P(I_k|I_{k-1}) + \log P(I_1) \end{aligned} \quad (2.3)$$

As the constant terms $\log P(u_k)$ do not affect the maximization, the Q function, with constant terms removed, is given as:

$$\begin{aligned} Q(\Theta|\Theta^{old}) &= E_{I_{1:N}|C_{obs}, \Theta^{old}} [\log P(y_{1:N}, u_{1:N}, I_{1:N}|\Theta)] \\ &= \sum_{k=1}^N \sum_{i=1}^M P(I_k = i|C_{obs}, \Theta^{old}) \log P(y_k|\phi_k, \theta_{I_k}) + \\ &\quad \sum_{k=2}^N \sum_{i=1}^M \sum_{j=1}^M P(I_k = i, I_{k-1} = j|C_{obs}, \Theta^{old}) \log P(I_k|I_{k-1}) + \\ &\quad \sum_{i=1}^M P(I_1 = i|C_{obs}, \Theta^{old}) \log P(I_1) \end{aligned} \quad (2.4)$$

where $C_{obs} = \{y_{1:N}, u_{1:N}\}$ is the observed data. For each local sub-model, the noise is assumed to follow the zero-mean Gaussian distribution with variance σ^2 .

To maximize the conditional expectation of the likelihood function over the parameters Θ , derivatives on Q function are taken with respect to each local sub-model parameter θ_i , transition probability a_{ij} , and noise parameter σ^2 , respectively. Equating the derivatives to zero yields the parameter update expressions:

$$\begin{aligned}
\theta_i &= \left[\sum_{k=1}^N P(I_k = i | C_{obs}, \Theta^{old}) \phi_k \phi_k^T \right]^{-1} \cdot \left[\sum_{k=1}^N P(I_k = i | C_{obs}, \Theta^{old}) \phi_k y_k \right], \\
&\quad \forall i \in \{1, 2, \dots, M\} \\
a_{ij} &= \frac{\sum_{k=2}^N P(I_k = j, I_{k-1} = i | C_{obs}, \Theta^{old})}{\sum_{k=2}^N P(I_{k-1} = i | C_{obs}, \Theta^{old})}, \quad \forall i, j \in \{1, 2, \dots, M\} \\
\sigma^2 &= \frac{\sum_{k=1}^N \sum_{i=1}^M P(I_k = i | C_{obs}, \Theta^{old}) (y_k - \phi_k^T \theta_i)^2}{\sum_{k=1}^N \sum_{i=1}^M P(I_k = i | C_{obs}, \Theta^{old})} \\
\pi_i &= P(I_1 = i | C_{obs}, \Theta^{old}), \quad \forall i \in \{1, 2, \dots, M\}
\end{aligned} \tag{2.5}$$

The calculation of posterior probabilities is using the backward and forward algorithms [12] as Algorithm 1 and Algorithm 2 illustrated, and the notations are given as $\alpha_k(i) \triangleq P(I_k = i, y_{1:k} | \Theta^{old})$ and $\beta_k(i) \triangleq P(y_{k+1:N} | I_k = i, \Theta^{old})$.

Algorithm 1: Forward algorithm

Data: $y_{1:N}, u_{1:N}$
Result: forward probabilities $\{\alpha_k\}$
Initialization: $\alpha_0(1) = 1$, and $\alpha_0(i) = 0, \forall i \neq 1$;
for $k = 1, 2, \dots, N$ **do**
 | $\alpha_k(i) = b_i(y_k) \sum_{j=1}^N \alpha_{k-1}(j) a_{ji}, \quad \forall i \in \{1, 2, \dots, M\}$
end

Algorithm 2: Backward algorithm

Data: $y_{1:N}, u_{1:N}$
Result: Backward probabilities $\{\beta_k\}$
Initialization: $\beta_N(i) = a_{i1}, \forall i \in \{1, 2, \dots, M\}$;
for $k = N - 1, N - 2, \dots, 1$ **do**
 | $\beta_k(i) = \sum_{j=1}^N a_{ij} b_j(y_{k+1}) \beta_{k+1}(j), \quad \forall i \in \{1, 2, \dots, M\}$
end

Note that a_{ij} refers to the transition probability $P(I_k = j | I_{k-1} = i)$, and $b_i(y_k)$ refers to the probability of the observation given state i , and is defined as $\frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(y_k - \phi_k^T \theta_i)^2}{2\sigma^2}}$. The posterior hidden state probabilities $P(I_k = i | C_{obs}, \Theta^{old})$ and $P(I_k =$

$j, I_{k-1} = i | C_{obs}, \Theta^{old}$ are derived as:

$$\begin{aligned}
P(I_k = i | C_{obs}, \Theta^{old}) &= \frac{P(y_{1:k}, I_k = i) P(y_{k+1:N} | I_k = i, \Theta^{old})}{P(y_{1:N} | \Theta^{old})} \\
&= \frac{\alpha_k(i) \beta_k(i)}{\sum_{i=1}^N \alpha_k(i) \beta_k(i)} \\
P(I_k = j, I_{k-1} = i | C_{obs}, \Theta^{old}) &= \frac{P(y_{1:N}, I_k = j, I_{k-1} = i | C_{obs}, \Theta^{old})}{P(y_{1:N} | \Theta^{old})}
\end{aligned} \tag{2.6}$$

where,

$$\begin{aligned}
&P(y_{1:N}, I_k = j, I_{k-1} = i | C_{obs}, \Theta^{old}) \\
&= P(y_{k+1:N} | I_k = j, \Theta^{old}) P(y_k | I_k = j, \Theta^{old}) P(I_k = j | I_{k-1} = i) P(I_{k-1} = i, y_{1:k-1} | \Theta^{old}) \\
&= \beta_k(j) b_j(y_k) a_{ij} \alpha_{k-1}(i)
\end{aligned} \tag{2.7}$$

2.2.3 Initialization for the EM algorithm

The EM algorithm is used for the calibration of the regime-switching model, and a good initialization would improve the computation and performance of the EM algorithm when applied to identify the pool price predictive model. It is assumed that various mechanisms that govern the changes of pool price, follow the Markov chain as defined by a discrete state sequence $\{I_{1:N}\}$ with three regimes (i.e., low, middle and high volatility). Therefore, a hidden Markov model (HMM) presented in Figure 2.7, is considered for the modeling of hidden regimes switching on the trend of pool price, and the modeling result is used as the initialization for the EM algorithm. In this section, both discrete and continuous HMMs are considered and trained to obtain the initial parameter values, respectively.

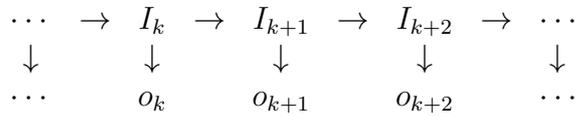


Figure 2.7: The hidden Markov model diagram

Discrete HMM approach

The discrete HMM approach is considering the both the regime I_k and the observation o_k to be discrete. In the HMM initialization approach for the pool price predictive model, the observation sequence is considered as the trend of pool price. In [3], a feature extraction method, called neighboring difference grouping, is used to obtain a discretized trend sequence of pool price. On the other hand, a graphic trend capturing method called triangle representation was proposed [14], [15], [16], which can also be used in pool price prediction process.

For the grouping representation based on neighboring difference, the pool prices are divided into five groups based on their value with group index from low to high represented as:

$$Index = \begin{cases} 1, & \forall \text{ Price} < 30 \text{ \$/MWh} \\ 2, & \forall 30 \text{ \$/MWh} \leq \text{Price} < 100 \text{ \$/MWh} \\ 3, & \forall 100 \text{ \$/MWh} \leq \text{Price} < 300 \text{ \$/MWh} \\ 4, & \forall 300 \text{ \$/MWh} \leq \text{Price} < 500 \text{ \$/MWh} \\ 5, & \forall \text{ Price} \geq 500 \text{ \$/MWh} \end{cases} \quad (2.8)$$

Then, a new processed sequence $\{S_k\}$ is developed as a discretized trend of price by calculating the group index differences between every two neighboring pool price as follows:

$$S_k = \begin{cases} 1, & \forall d_k > 2 \\ 2, & \forall d_k < -2 \\ 3, & \forall -2 \leq d_k \leq 2 \end{cases} \quad (2.9)$$

where, S_k is the element of the feature extracted pool price symbol sequence at time k and d_k is the group index differences between each two neighboring electricity prices, which is one kind of discretized trend sequence of pool price.

Instead of grouping the pool price neighboring differences into a discretized trend sequence, a triangular representation for trend capturing can also be employed as another feature extraction method for the pool price sequence. The triangle representation is proposed by Cheung and Stephanopoulos [14] to calculate trend representation graphically by using the first and second order derivatives. By searching and connecting neighboring extrema or inflection points, the discretized trend sequence is generated with the elements classified into different basic triangular shapes in Figure 2.8, where (a) refers to concave downward with monotonic increase; (b) refers

to concave downward with monotonic decrease; (c) refers to concave upward with monotonic increase; (d) refers to concave upward with monotonic decrease; (e) refers to linear increase; (f) refers to linear decrease; (g) refers to the shape of constant. Moreover, two dimensional data of time duration and magnitude are obtained for each neighboring extrema or inflection points as follows:

$$d_n = t_n - t_{n-1} \quad (2.10)$$

$$m_n = y_{t_n} - y_{t_{n-1}}$$

where, y_{t_n} refers to the value of the extrema or inflection point, the pool price in our case, at time t_n with index subscript n ; n refers to the series number for extrema and inflection points. Based on the durations and magnitudes, the basic triangle shapes are divided into different ranges for the slope. Then, the pool price sequence is processed into triangle representations as a discretized trend sequence.

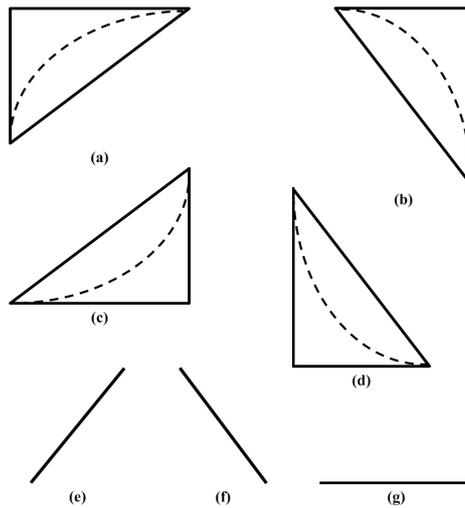


Figure 2.8: Triangle representation components

To estimate the parameters of discrete HMM, the Baum-Welch algorithm [12] is applied by using the EM algorithm to find the maximum likelihood estimate. The posterior state probability $P(I_k = i | o_{1:N}, \Theta^{old})$ using Bayes' theorem is given as:

$$P(I_k = i | o_{1:N}, \Theta^{old}) = \frac{P(o_{k+1:N} | I_k = i, o_{1:k}) P(I_k = i, o_{1:k})}{P(o_{1:N})} \quad (2.11)$$

where, $P(o_{k+1:N} | I_k = i, o_{1:k}) = P(o_{k+1:N} | I_k = i)$ is the backward probability denoted by $\beta_k(i)$; $P(I_k = i, o_{1:k})$ is the forward probability denoted by $\alpha_k(i)$; the backward

and forward probabilities can be calculated from the backward-forward algorithm [12] where $b_i(o_k)$ is state to discrete observation emission probability in discrete HMM case and a_{ij} as transition probability from state i to state j .

Continuous HMM approach

The pool price $\{y_k\}$ is real valued from 0 to 1000, and the neighboring difference $\{e_k|e_k = y_k - y_{k-1}\}$ is also real valued from -1000 to 1000, therefore a continuous observation hidden Markov model can be employed based on the pool price trend using the continuous neighboring difference sequence. Unlike the discrete case, the state $\{I_k\}$ to continuous observation $\{e_k\}$ emission probabilities are replaced with a continuous distribution. The probability density follow a mixture of Gaussian with hidden clusters $\{m_k\}$, and we have the following model representation as Figure 2.9 illustrated:

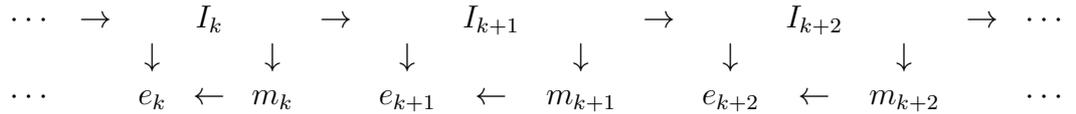


Figure 2.9: Continuous hidden Markov model with hidden cluster m_k

At each time step, the hidden state $\{I_k\}$ is generated by following Markov chain rule with transition probability $a_{I_{k-1}I_k}$, which determines the hidden cluster m_k according to states to clusters emission probabilities $g_{I_k m_k}$. The probability density of continuous observation difference e_k with known state and cluster at time k is modeled by Gaussian function as follows:

$$P(e_k|I_k = i, m_k = l, \Theta) = \phi(e_k|\mu_{il}, \sigma_{il}^2) \quad (2.12)$$

where $\phi(\cdot)$ is Gaussian function with mean μ_{ij} and variance σ_{ij}^2 .

The overall probability density of an observation generated by a state I_k is given by a mixture of Gaussian functions:

$$\begin{aligned} P(e_k|I_k = i, \Theta) &= \sum_l P(m_k = l|I_k = i, \Theta)P(e_k|I_k = i, m_k = l, \Theta) \\ &= \sum_l g_{il} \cdot \phi(e_k|\mu_{il}, \sigma_{il}^2) \end{aligned} \quad (2.13)$$

The maximum likelihood estimation is employed to estimate the unknown parameters for the continuous HMM using the EM algorithm. The derivations of the EM algorithm are as follows:

First the complete-data log likelihood function or the Q function is derived as:

$$\begin{aligned}
Q(\Theta|\Theta^{old}) &= E_{I_{1:N}, m_{1:N}|e_{1:N}, \Theta^{old}} \{ \log P(e_{1:N}, I_{1:N}, m_{1:N}|\Theta) \} \\
&= \sum_{k=1}^N \sum_i \sum_j P(I_k = i, m_k = j|e_{1:N}, \Theta^{old}) \log P(e_k|I_k, m_k, \Theta) + \\
&\quad \sum_{k=1}^N \sum_i \sum_j P(I_k = i, m_k = j|e_{1:N}, \Theta^{old}) \log P(m_k|I_k, \Theta) + \\
&\quad \sum_{k=2}^N \sum_i \sum_l P(I_k = i, I_{k-1} = l|e_{1:N}, \Theta^{old}) \log P(I_k|I_{k-1}, \Theta) + \\
&\quad \sum_i P(I_1 = i|e_{1:N}, \Theta^{old}) \log P(I_1|\Theta)
\end{aligned} \tag{2.14}$$

Taking derivatives of the Q function with respect to parameters and equating the derivatives to zero yields the following parameter update equations:

$$\begin{aligned}
\pi_i &\triangleq P(I_1 = i) = P(I_1 = i|e_{1:N}, \Theta^{old}) \\
a_{ij} &\triangleq P(I_k = j|I_{k-1} = i) = \frac{\sum_{k=2}^N P(I_k = j, I_{k-1} = i|e_{1:N}, \Theta^{old})}{\sum_{k=2}^N P(I_{k-1} = i|e_{1:N}, \Theta^{old})} \\
g_{il} &\triangleq P(m_k = l|I_k = i) = \frac{\sum_{k=2}^N P(m_k = l, I_k = i|e_{1:N}, \Theta^{old})}{\sum_{k=2}^N P(I_k = i|e_{1:N}, \Theta^{old})} \\
\mu_{il} &= \frac{\sum_{k=1}^N P(m_k = l, I_k = i|e_{1:N}, \Theta^{old}) e_k}{\sum_{k=1}^N P(m_k = l, I_k = i|e_{1:N}, \Theta^{old})} \\
\sigma_{il}^2 &= \frac{\sum_{k=1}^N P(m_k = l, I_k = i|e_{1:N}, \Theta^{old}) (e_k - \mu_{il})^2}{\sum_{k=1}^N P(m_k = l, I_k = i|e_{1:N}, \Theta^{old})}
\end{aligned} \tag{2.15}$$

where, above equations are given for $i, j = 1, 2, \dots, N_1$ and $l = 1, 2, \dots, N_2$, and N_1 and N_2 is the number of hidden states and hidden clusters, respectively.

For the E-step, the posterior hidden variable probabilities are calculated using the results of the forward and backward algorithm [12]:

$$\begin{aligned}
P(I_k = i|e_{1:N}, \Theta^{old}) &= \frac{\alpha_k(i)\beta_k(i)}{\sum_i \alpha_k(i)\beta_k(i)} \\
P(I_k = i, I_{k+1} = j|e_{1:N}, \Theta^{old}) &= \frac{\beta_{k+1}(j)b_j(e_{k+1})a_{ij}\alpha_k(i)}{\sum_i \sum_j \beta_{k+1}(j)b_j(e_{k+1})a_{ij}\alpha_k(i)} \\
P(I_k = i, m_k = l|e_{1:N}, \Theta^{old}) &= P(I_k = i|e_{1:N}, \Theta^{old}) \cdot \frac{g_{il} \cdot \phi(e_k|\mu_{il}, \sigma_{il}^2)}{b_i(e_k)}
\end{aligned} \tag{2.16}$$

where, $b_i(e_k) \triangleq P(e_k|I_k = i, \Theta^{old}) = \sum_l g_{il} \cdot \phi(e_k|\mu_{il}, \sigma_{il}^2)$; $\alpha_k(i)$ and $\beta_k(i)$ are forward and backward probabilities at time step k with state index as i , respectively, and a_{ij} refers to transition probability from state i to state j .

2.3 Validation studies

Validation studies for monthly predictions are presented using the proposed approach, and the data is from the AESO's website*. Hourly predictions within a month (around 750 data points) are chosen for validation studies to ensure sufficiency of data including price spikes and periodic behavior. For training purposes, a batch of historical data is selected over a month (around 750 data points).

To evaluate the prediction performance, monthly mean absolute error (MMAE), monthly root mean squared error (MRMSE), monthly correlation coefficient, monthly fitting rate and monthly mean absolute percentage error (MMAPE) are used as validation metrics, which are defined below:

$$\text{MMAE} = \frac{1}{N_{month}} \sum_{k=1}^{N_{month}} |f_k - y_k| = \frac{1}{N_{month}} \sum_{k=1}^{N_{month}} |e_k| \quad (2.17)$$

$$\text{MRMSE} = \sqrt{\frac{1}{N_{month}} \sum_{k=1}^{N_{month}} e_k^2} \quad (2.18)$$

$$\text{MMAPE} = \frac{1}{N_{month}} \sum_{k=1}^{N_{month}} \left| \frac{f_k - y_k}{y_k} \right| \quad (2.19)$$

$$\text{Fitting rate} = 1 - \frac{\text{norm}(F - Y)}{\text{norm}(Y - \text{mean}(Y))} \quad (2.20)$$

$$\text{Correlation coefficient} = \frac{\sum_{k=1}^{N_{month}} (f_k - \text{mean}(F))(y_k - \text{mean}(Y))}{\sqrt{\sum_{i=1}^{N_{month}} (f_i - \text{mean}(F))^2 (y_i - \text{mean}(Y))^2}} \quad (2.21)$$

where f_k refers to the price prediction at time step k ; F is the vector form of f_k ; y_k is the actual price at time step k with vector form Y ; correlation coefficient measures the linear relationship between two data sets, and 1 implies perfect positive linear correlation; fitting rate measures the variation of the output in percentage, and higher fitting rate means better prediction performance.

*<http://ets.aeso.ca/>

2.3.1 One-hour ahead pool price prediction

For one-hour ahead pool price prediction, the prediction dynamics is represented as:

$$\begin{aligned}
 E(k+1) &= \sum_{i=1}^M \sum_{j=1}^M P(I_k = i | C_k, \Theta) \cdot a_{ij} \cdot E^j(k+1) \\
 E^j(k+1) &= \phi_{k+1}^T \theta_j
 \end{aligned} \tag{2.22}$$

where $E^j(k+1)$ is the prediction from the j^{th} local model; ϕ_k is the regressor at time step k ; θ_j is the parameter for the j^{th} local sub-model; $P(I_k | C_k, \Theta)$ is the filtered hidden regime probability at time step k , which is multiplied with transition probability a_{ij} in order to get a prediction of the state probability $P(I_{k+1} | C_k, \Theta)$ at time step $k+1$; C_k refers to the available observations at time k : $\{y_{1:k}, u_{1:k}\}$.

Initialization methods and model orders

The inputs for the local model could be rearranged as pool price, demand, weekly dummy variable and hourly processed time sequence. Among them, the pool prices or demands including the AESO's prediction and historical actual data can be combined with the prediction on the next hour by AESO and historical actual data, and n refers to the order of the rearranged input term including the historical data for the pool price and demands.

DHMM refers to discrete HMM of feature extracted sequence based on neighboring difference; DHMMTR refers to discrete HMM of feature extracted sequence based on triangle representations; CHMM refers to continuous HMM method based on the neighboring difference. Two examples are provided, namely January 2013 and August 2013 months' data, and the proposed method using different initialization methods is applied to estimate the model parameters, where the self validation results are presented in Table 2.3 and Table 2.1. This shows that CHMM and DHMMTR tend to have a better fitting than DHMM, i.e., CHMM and DHMMTR provide the initial values which helps the EM algorithm reach a better fitting result at the cost of more computation. Using the model estimated from previous examples as the training data to predict another future month, the cross validation results are exhibited in Table 2.4 and Table 2.2. The performance metrics do not seem to trend the same direction. The monthly mean absolute error shows that the prediction using the proposed approach

is not as good as AESO’s results, while monthly root mean squared error shows the opposite results. The reason is that AESO’s prediction is more accurate in the low-range price but has large prediction errors when it is peak price. Therefore, we divide the prediction results into high-range price by setting the threshold for the high pool price as 100 \$/MWh. The prediction results for high pool price are presented in Table 2.5 and Table 2.6. Based on the two validation results, DHMM and DHMMTR tend to have a better performance when model order is 2, and CHMM tend to have a better performance when model order is 1.

Table 2.1: Training data fitting in August 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO’s	N/A	21.00	69.59	93.64	59.69
DHMM	n=1	20.86	55.43	95.30	67.83
	n=2	19.99	52.33	95.76	69.65
	n=3	21.06	56.71	94.61	67.13
	n=4	20.28	53.51	95.25	69.00
DHMMTR	n=1	13.50	35.73	97.84	79.26
	n=2	19.91	52.71	95.57	69.43
	n=3	19.80	52.84	95.45	69.37
	n=4	19.31	52.51	95.46	69.58
CHMM	n=1	14.76	38.32	97.51	77.76
	n=2	16.56	43.50	96.77	74.77
	n=3	16.52	45.29	96.50	73.75
	n=4	16.39	42.51	96.93	75.37

Training data selection

Pool price sequence exhibits strong periodic characteristics due to the customers’ power-consuming behaviors. The generators scheduling could also contribute to pool pricing due to introduction of new generators or some shutdowns. These intrinsic mechanisms result in time-varying parameters for the switching model applying in pool price prediction, that is to say, the predicted pool price for a certain future month has different performance from different training data. To capture the appropriate local model, we use an entire month’s data for training purpose, and compare the prediction results on the same future month with the ones using different months as training data. The use of the past year monthly data to model for a future year

Table 2.2: Prediction in August 2014 by training data August 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	N/A	6.61	35.51	94.76	87.16
DHMM	n=1	9.46	22.53	97.66	91.85
	n=2	9.00	23.53	97.40	91.49
	n=3	10.12	26.06	96.85	90.58
	n=4	10.91	24.11	97.46	91.28
DHMMTR	n=1	7.78	22.67	97.62	91.80
	n=2	8.36	25.20	97.01	90.89
	n=3	8.83	28.22	96.24	89.80
	n=4	9.16	28.34	96.22	89.75
CHMM	n=1	7.48	24.49	97.24	91.14
	n=2	8.14	24.85	97.13	91.01
	n=3	9.62	41.73	92.23	84.91
	n=4	9.61	40.32	92.62	85.42

same or similar month prediction is called ‘similar month’ rule in the sequel. Two examples are presented in Table 2.7 and Table 2.8, where the initialization method is DHMMTR and the model order is chosen as one. January 2014 and August 2014 month data are chosen as the prediction examples, since these months represent two typical different seasons, and have dissimilar prediction errors for AESO’s pool price forecast. The metrics MMAE and MRMSE in Table 2.7 and Table 2.8 show some disagreement since MRMSE exaggerates the prediction errors in peaks, which also calls for the investigation of high-price range prediction. In Table 2.10, in past two years monthly data are taken as the training data separately, and the results in Table 2.9 indicate that a better high-price range prediction obtained using the training data from the same month in the previous year.

The predictions using the ‘similar month’ training data selection rule are presented in Table 2.11, which validate the proposed method by applying in the high-price range pool price prediction. The month rows denoted as *N/A* in Table 2.11 indicates that there is no peak price. Different initialization methods for the EM algorithm are compared. By taking the average of MRMSE of different months’ predictions in a year, the DHMMTR and CHMM have errors 133.57 and 133.62, respectively, smaller than that of DHMM, which is 135.87.

Table 2.3: Training data fitting in January 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	N/A	10.92	40.44	94.85	62.72
DHMM	n=1	11.81	31.81	95.73	70.64
	n=2	10.21	31.67	95.64	70.79
	n=3	9.20	25.61	97.21	76.39
	n=4	9.10	24.20	97.55	77.70
DHMMTR	n=1	10.49	34.62	94.81	68.05
	n=2	9.63	28.78	96.45	73.45
	n=3	10.19	29.04	96.72	73.23
	n=4	10.38	29.37	96.69	72.94
CHMM	n=1	10.70	31.64	95.66	70.80
	n=2	10.75	32.82	95.35	69.73
	n=3	9.65	28.15	96.60	74.05
	n=4	9.90	28.17	96.62	74.04

2.3.2 Two-hour ahead pool price prediction

The prediction model can also be extended for two-hour ahead prediction as follows:

$$E(k+2) = \sum_{i=1}^M \sum_{j=1}^M \sum_{h=1}^M P(I_k = i | C_k, \Theta) \cdot a_{ij} \cdot a_{jh} \cdot E^h(k+2) \quad (2.23)$$

$$E^h(k+2) = \theta_h^T \phi_{k+2}^n$$

where $E^h(k+2)$ is the two-hour ahead prediction from the h^{th} local sub-model; ϕ_k^n is the model regressor for two-hour ahead prediction at time step k , and the difference compared to the one-hour ahead prediction is that historical data of one hour ahead are replaced with the proposed prediction; θ_h is the parameter for the h^{th} local sub-model; $P(I_k | C_k, \Theta)$ is the filtered hidden regime probability at time step k , which is multiplied with transition probability a_{ij} and a_{jh} in order to get a prediction of the state probability $P(I_{k+2} | C_k, \Theta)$ at time step $k+2$.

The proposed approach is applied and the ‘similar Month’ training data selection rule is used, the prediction results for high pool price regions are presented in Table 2.12. CHMM and DHMMTR are used as the initialization methods to generate the two-hour ahead pool price prediction, which are compared with AESO’s prediction. CHMM exhibits average MRMSE of 140.63, smaller than that of DHMMTR, which is 149.72. However, the two-hour ahead pool price prediction faces more process uncer-

Table 2.4: Prediction in January 2014 by training data January 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	N/A	6.87	32.01	90.16	72.27
DHMM	n=1	9.49	34.59	87.32	70.04
	n=2	9.44	40.68	81.17	64.76
	n=3	8.95	34.64	86.52	69.99
	n=4	8.78	33.60	87.55	70.90
DHMMTR	n=1	8.07	33.02	87.92	71.39
	n=2	7.45	30.17	90.14	73.86
	n=3	8.39	36.56	84.91	68.33
	n=4	8.10	31.81	88.84	72.44
CHMM	n=1	8.32	30.98	89.54	73.16
	n=2	8.78	34.71	86.58	69.93
	n=3	8.41	35.15	86.04	69.55
	n=4	8.21	32.75	88.03	71.63

tainties than the one-hour ahead pool price prediction, which results in deteriorated performance.

2.4 Conclusions

A pool price prediction approach is developed and applied in Alberta's electricity market. The Markov regime-switching model is well defined for pool price, and the model calibration is based on the EM algorithm with the hidden Markov model is proposed for initialization methods. Through validations studies, the time-varying problem for parameter estimations is solved by using the 'similar month' training data selection rule and the results illustrate good performance for prediction in different months, particularly in high pool price regions. Different initialization methods are compared and we conclude that using the proposed initialization methods CHMM and DHMMTR lead to a superior prediction than using DHMM.

Table 2.5: High-price range prediction in January 2014 by training data January 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	N/A	77.18	139.54	81.43	37.03
DHMM	n=1	85.00	149.51	77.76	32.54
	n=2	79.21	134.52	82.27	39.30
	n=3	89.99	144.24	77.61	34.91
	n=4	92.41	145.05	77.51	34.55
DHMMTR	n=1	85.17	143.82	77.47	35.10
	n=2	73.01	130.48	82.23	41.12
	n=3	74.55	129.06	82.73	41.76
	n=4	75.13	130.19	82.51	41.25
CHMM	n=1	75.21	133.84	81.51	39.61
	n=2	87.77	150.42	76.45	32.12
	n=3	81.34	141.89	78.65	35.97
	n=4	83.40	141.80	78.66	36.01

Table 2.6: High-price range prediction in August 2014 by training data August 2013

Initialization	Order	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	N/A	127.41	192.15	88.48	45.95
DHMM	n=1	77.79	110.96	95.02	68.79
	n=2	67.06	105.73	95.84	70.26
	n=3	69.16	106.44	95.69	70.06
	n=4	81.89	112.34	95.42	68.40
DHMMTR	n=1	78.58	117.17	94.55	67.04
	n=2	71.11	106.77	95.41	69.97
	n=3	73.67	109.50	95.20	69.20
	n=4	75.48	109.48	95.22	69.20
CHMM	n=1	59.91	97.94	96.14	72.45
	n=2	85.16	127.91	93.84	64.02
	n=3	66.44	102.71	95.75	71.11
	n=4	64.11	101.45	95.89	71.46

Table 2.7: Prediction in January 2014 by different training data

Month(Tr)	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	6.87	32.01	90.16	72.27
Dec 2013	9.45	35.51	85.80	69.24
Nov 2013	9.36	37.21	84.40	67.77
Oct 2013	9.49	32.71	88.69	71.67
Sept 2013	9.76	34.93	86.79	69.74
Aug 2013	10.11	35.54	85.84	69.21
Jul 2013	9.08	34.76	86.52	69.89
Jun 2013	9.45	33.43	87.58	71.04
May 2013	9.37	33.24	88.50	71.21
Apr 2013	7.62	31.87	89.11	72.39
Mar 2013	11.64	34.92	86.61	69.76
Feb 2013	7.20	30.85	89.51	73.28
Jan 2013	7.45	30.17	90.14	73.86

Table 2.8: Prediction in August 2014 by different training data

Month(Tr)	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	6.61	35.51	94.76	87.16
Jul 2014	7.96	32.14	95.14	88.38
Jun 2014	5.60	27.03	96.64	90.22
May 2014	6.40	26.69	96.64	90.35
Apr 2014	7.04	37.12	93.52	86.58
Mar 2014	13.12	71.40	76.98	74.18
Feb 2014	8.53	26.06	96.91	90.57
Jan 2014	6.44	26.56	96.64	90.39
Dec 2013	8.06	25.47	96.99	90.79
Nov 2013	11.38	54.17	90.56	80.41
Oct 2013	9.05	27.93	96.59	89.90
Sept 2013	8.19	29.68	95.93	89.26
Aug 2013	8.36	25.20	97.01	90.89

Table 2.9: High-price range prediction in January 2014 by different training data

Month(Tr)	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	77.18	139.54	81.43	37.03
Dec 2013	94.85	154.26	74.24	30.39
Nov 2013	104.87	161.41	70.76	27.17
Oct 2013	83.42	140.97	80.08	36.39
Sept 2013	87.85	147.95	75.56	33.24
Aug 2013	94.58	153.28	74.06	30.83
Jul 2013	89.47	151.02	74.79	31.85
Jun 2013	86.75	141.78	77.31	36.02
May 2013	84.55	142.52	78.44	35.69
Apr 2013	80.42	138.46	79.72	37.52
Mar 2013	87.17	145.85	75.99	34.19
Feb 2013	79.50	134.38	80.02	39.36
<u>Jan 2013</u>	73.01	130.48	82.23	41.12

Table 2.10: High-price range prediction in August 2014 by different training data

Month(Tr)	MMAE	MRMSE	Corr(%)	Fit(%)
AESO's	127.41	192.15	88.48	45.95
Jul 2014	99.03	145.12	93.05	59.18
Jun 2014	76.89	134.04	93.26	62.29
May 2014	86.57	131.20	93.08	63.09
Apr 2014	107.90	150.06	91.78	57.79
Mar 2014	299.24	387.19	24.46	-8.92
Feb 2014	75.46	116.35	94.81	67.27
Jan 2014	92.07	136.65	92.82	61.56
Dec 2013	77.72	109.12	95.77	69.30
Nov 2013	219.31	281.51	85.98	20.81
Oct 2013	92.10	121.50	95.66	65.82
Sept 2013	96.62	145.24	93.20	59.14
Aug 2013	71.11	106.77	95.41	69.97
Jul 2013	67.12	107.89	95.30	69.65
Jun 2013	76.20	118.50	94.72	66.67
May 2013	101.51	150.03	92.09	57.80
Apr 2013	85.57	129.80	93.79	63.49
Mar 2013	105.73	137.58	93.92	61.30
Feb 2013	172.82	204.01	86.11	42.61
Jan 2013	77.74	107.85	95.68	69.66
Dec 2012	91.31	119.23	95.03	66.46
Nov 2012	91.55	117.60	95.02	66.92
Oct 2012	91.53	129.62	93.52	63.54
Sept 2012	100.96	142.77	92.22	59.84
Aug 2012	77.13	116.10	94.64	67.34

Table 2.11: High-price range prediction in different months using ‘similar month’ rule

Month(Val)	Methods	MMAE	MRMSE	Corr(%)	Fit(%)
Sept 2014				N/A	
Aug 2014	AESO's	127.41	192.15	88.48	45.95
	DHMMTR	71.11	106.77	95.41	69.97
	DHMM	67.06	105.73	95.84	70.26
	CHMM	62.13	98.53	96.09	72.28
Jul 2014	AESO's	154.12	253.46	71.53	22.35
	DHMMTR	141.78	215.21	78.82	34.07
	DHMM	158.35	238.62	75.55	26.90
	CHMM	154.36	233.76	74.08	28.39
Jun 2014	AESO's	160.79	268.46	63.42	1.85
	DHMMTR	131.86	196.18	72.13	28.27
	DHMM	108.83	186.03	77.27	31.99
	CHMM	95.66	156.79	82.89	42.68
May 2014	AESO's	115.82	199.93	76.84	29.47
	DHMMTR	118.33	184.65	77.40	34.86
	DHMM	104.90	163.82	83.01	42.21
	CHMM	115.17	173.90	80.31	38.66
Apr 2014				N/A	
Mar 2014	AESO's	45.96	80.92	81.31	-5.73
	DHMMTR	39.46	60.40	74.42	21.09
	DHMM	43.34	57.61	77.34	24.73
	CHMM	40.56	58.71	78.24	23.29
Feb 2014	AESO's	51.55	94.32	94.31	61.92
	DHMMTR	60.71	89.64	94.16	63.81
	DHMM	53.75	80.97	94.70	67.31
	CHMM	64.10	102.94	91.05	58.44
Jan 2014	AESO's	77.18	139.54	81.43	37.03
	DHMMTR	73.01	130.48	82.23	41.12
	DHMM	79.21	134.52	82.27	39.30
	CHMM	75.21	133.84	81.51	39.61
Dec 2013	AESO's	142.84	200.04	68.53	0.98
	DHMMTR	105.82	141.43	73.40	29.99
	DHMM	111.31	149.32	69.70	26.09
	CHMM	110.63	149.52	71.13	25.99
Nov 2013	AESO's	118.36	168.04	82.70	-24.27
	DHMMTR	58.55	75.27	85.93	44.34
	DHMM	62.42	86.18	78.18	36.27
	CHMM	65.89	83.70	79.33	38.10
Oct 2013	AESO's	134.00	200.43	72.71	5.58
	DHMMTR	104.06	144.40	76.41	31.97
	DHMM	94.82	147.23	75.99	30.64
	CHMM	99.39	144.56	75.22	31.90

Table 2.12: High-price range two-hour ahead prediction in different months using 'similar month' rule

Month(Val)	Methods	MMAE	MRMSE	Corr(%)	Fit(%)
Sept 2014				N/A	
Aug 2014	AESO's	127.41	192.15	88.48	45.95
	CHMM	109.79	137.52	93.02	61.32
	DHMMTR	141.07	171.38	93.12	51.79
Jul 2014	AESO's	154.12	253.46	71.53	22.35
	CHMM	180.93	232.27	73.48	28.84
	DHMMTR	194.08	265.65	70.89	18.62
Jun 2014	AESO's	160.79	268.46	63.42	1.85
	CHMM	128.48	182.54	78.04	33.26
	DHMMTR	145.95	200.35	73.71	26.75
May 2014	AESO's	115.82	199.93	76.84	29.47
	CHMM	110.14	160.92	82.86	43.23
	DHMMTR	130.87	186.42	80.80	34.24
Apr 2014				N/A	
Mar 2014	AESO's	45.96	80.92	81.31	-5.73
	CHMM	39.90	57.71	82.50	24.59
	DHMMTR	36.18	50.59	81.12	33.90
Feb 2014	AESO's	51.55	94.32	94.31	61.92
	CHMM	64.98	98.43	92.45	60.26
	DHMMTR	62.81	89.50	94.38	63.87
Jan 2014	AESO's	77.18	139.54	81.43	37.03
	CHMM	76.95	137.07	79.50	38.15
	DHMMTR	98.80	154.57	81.79	30.25
Dec 2013	AESO's	142.84	200.04	68.53	0.98
	CHMM	127.32	171.08	68.67	15.32
	DHMMTR	113.95	144.70	70.65	28.37
Nov 2013	AESO's	118.36	168.04	82.70	-24.27
	CHMM	61.13	85.55	82.59	36.73
	DHMMTR	69.30	89.94	78.95	33.49
Oct 2013	AESO's	134.00	200.43	72.71	5.58
	CHMM	101.44	143.27	75.92	32.50
	DHMMTR	107.91	144.17	73.65	32.08

Chapter 3

Identification of linear dynamic errors-in-variables systems with a dynamic process for noise-free inputs using EM algorithm

3.1 Introduction

An errors-in-variables (EIV) system refers to a process where both observed inputs and outputs are noise-corrupted. A single-input single-output EIV process is depicted in Figure 3.1, where \hat{u}_k is the noise-free process input; \tilde{u}_k is the input disturbance and u_k is the measured input; \tilde{y}_k is the output disturbance; \hat{y}_k is the noise-free process output and y_k is the measurement. Both the noise-free variables \hat{u}_k and \hat{y}_k are unknown. In practice, the input measurements will also be corrupted with noises, hence, the EIV model is a more practical representation of the system dynamics in contrast to the models that assume noise-free inputs. For example, consider an electricity pool prediction model with the electricity demand forecast as one of the input variables. However, the actual mechanism for deciding the pool price is the real demand which is not available. The demand forecast variable is therefore equivalent to a noise-corrupted input for the actual pool price model, and if we can find the uncertainty of the demand forecast and develop a more realistic EIV model for the pool price mechanism, then the pool price prediction could be improved. Another example is the model development of tray temperature profiles of a distillation column using flow-rate data. The flow rate measurement is corrupted with noise, so are

the temperature measurements. Hence, if the input noise was accounted for, it would provide more fidelity to the model.

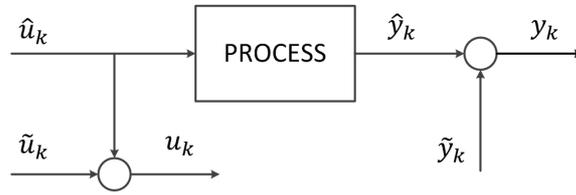


Figure 3.1: The errors-in-variables process diagram

There are many different ways for identifying linear dynamic systems with noise-corrupted output measurements. However, using these traditional methods to identify EIV systems would lead to biased estimations of parameters due to the uncertainties in the input measurements, making the parameter estimation for EIV systems a difficult problem. For more comprehensive treatment on this topic, readers are referred to [4].

In literature, the EIV identification problem has already been studied extensively and many different approaches have been proposed as in reviews [4] and [17]. Among the existing EIV identification approaches, several subspace methods for identifying EIV systems have been investigated in [18], [19] and [5]. In [18], unbiased estimations using subspace framework for EIV systems are derived using the instrumental variables (IV) methods. These subspace EIV methods have the advantages of numerical simplicity and stability by relying on computational tools such as the QR factorization and the singular value decomposition (SVD). Furthermore, being a non-iterative method, it avoids local maxima and convergence problems [20]. Also, subspace methods are in state space form and therefore can be easily adapted to multi-input multi-output (MIMO) processes. The results presented in [18], [19] and [5] show that adoption of the subspace EIV methods for linear dynamic EIV systems identification provide good estimation of poles but less accurate estimation of zeros. Other EIV methods like the bias-eliminated least squares (BELS), proposed to correct the bias when applying the ordinary least squares to estimate parameters of dynamic EIV processes, also present a good parameter estimations in a non state-space form [21].

In this chapter, a feasible alternative is to use the Maximum Likelihood (ML)

approach which has the advantage of high accuracy on estimations of parameters. But the caveat here is that, the large number of unknown variables compared with the number of observations would lead to the failure of the MLE [22]. To overcome this issue, one strategy is to reduce the number of unknowns by choosing reasonable assumptions and parameterizations. For example, in [23], the frequency domain ML approach is applied in EIV systems under the assumption that the noise variance ratio is known; another strategy mentioned in [17] imposes a parametric model for the noise-free input instead of assuming the noise-free input as an arbitrary sequence following an unknown distribution.

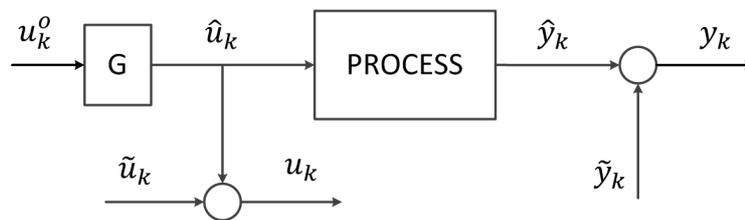


Figure 3.2: The errors-in-variables process with input generation diagram

As a variant of the ML method, the EM algorithm has been applied widely in system identification and shown the effectiveness, such as in identification of switching ARX models [24], linear parameter varying (LPV) models [10], and nonlinear models with missing observations [25]. For the identification of the state space model with noise-free inputs, the ML approach has been used in [26] and [27], where the EM algorithm is used to estimate the parameters with unknown states. Following a similar approach, we are using the time domain ML identification through the EM algorithm to identify EIV model parameters, due to its ability in solving the estimation problems with hidden variables. The linear dynamic EIV model structure is chosen in the state space form. The novelties of the current work are as follows: (i) it is proposed to use the EM algorithm for the ML estimation of linear EIV state space model parameters with input dynamics assumption; (ii) in conjunction with that, the development of a smoother for state space models with stochastic inputs; (iii) extensive validations of the proposed method including an experimental study.

In this chapter, it is considered that, the noise-free input is driven by a dynamic state space model, in accordance with more practical situations. Section 3.2 of the

paper provides problem formulation. The main contributions of the work, necessary derivations for ML method using the EM algorithm, are presented in Section 3.3. Section 3.4 shows the estimation performance of the proposed approach using various examples including an experimental study. Section 3.5 draws conclusions from this study.

3.2 Problem formulation

The EIV system with a dynamic input process is presented in Figure 3.2, where u_k^o is the source of the input and is known. For example, it can be a controller output for a flow control valve in a cascaded control loop; \hat{u}_k can be the actual flow rate and u_k is the measured flow rate that is corrupted by noise. G is the model describing the input dynamics, while other notations are the same as in Figure 3.1.

The linear dynamic EIV model in a state-space form for the system represented in Figure 3.2 is given by:

$$\begin{aligned} x_{k+1} &= Ax_k + B\hat{u}_k + w_k \\ \hat{y}_k &= Cx_k \\ y_k &= \hat{y}_k + \tilde{y}_k \\ u_k &= \hat{u}_k + \tilde{u}_k \end{aligned} \tag{3.1}$$

where, $x_k \in \mathbb{R}^{n \times 1}$ is the hidden state; $\hat{u}_k \in \mathbb{R}^{m \times 1}$ is the unknown noise-free input; $u_k \in \mathbb{R}^{m \times 1}$ is the measurement of \hat{u}_k ; $w_k \in \mathbb{R}^{n \times 1}$ is the additive process noise in the state; $\tilde{u}_k \in \mathbb{R}^{m \times 1}$ and $\tilde{y}_k \in \mathbb{R}^{q \times 1}$ are the measurement noises of the input and the output, respectively; and $y_k \in \mathbb{R}^{q \times 1}$ is the output measurement.

We consider a parametric dynamic model for the noise-free input $\hat{u}(k)$, as follows:

$$\begin{aligned} z_{k+1} &= A_o z_k + B_o u_k^o + w_k^o \\ \hat{u}_k &= C_o z_k \end{aligned} \tag{3.2}$$

where, $z_k \in \mathbb{R}^{n_o \times 1}$ is the hidden state of the input dynamics and $u_k^o \in \mathbb{R}^{m_o \times 1}$ is the known input source; $w_k^o \in \mathbb{R}^{n_o \times 1}$ is the process noise of the input process. The measurement noises \tilde{u}_k , \tilde{y}_k , and the process noises w_k , w_k^o are assumed to follow i.i.d Gaussian distributions with zero means and unknown covariance parameters $\Sigma_{\tilde{u}}$, $\Sigma_{\tilde{y}}$, Σ_w , Σ_{w^o} , respectively, and the noise-free input \hat{u}_k is uncorrelated with the noises.

Further, the four white-noise sequences \tilde{u} , \tilde{y} , w , w^o are assumed to be uncorrelated, that is:

$$E \left[\begin{pmatrix} \tilde{u}_k \\ \tilde{y}_k \\ w_k \\ w_k^o \end{pmatrix} \begin{pmatrix} \tilde{u}_i^T & \tilde{y}_i^T & w_i^T & w_i^{oT} \end{pmatrix} \right] = \begin{bmatrix} \Sigma_{\tilde{u}} & 0 & 0 & 0 \\ 0 & \Sigma_{\tilde{y}} & 0 & 0 \\ 0 & 0 & \Sigma_w & 0 \\ 0 & 0 & 0 & \Sigma_{w^o} \end{bmatrix} \cdot \delta_{ki} \geq 0 \quad (3.3)$$

where, δ_{ki} refers to the Kronecker's delta. It is worth noting that the estimation of \hat{u}_k is a colored signal due to the dynamics in (3.2).

The model parameters $\{A, B, C\}$ are to be estimated using the measurements $\{y_{1:N}, u_{1:N}\}$ and the known input source $\{u_{1:N}^o\}$, for which we propose to use the ML estimation method through the EM algorithm to identify model parameters, by treating states $x_{1:N}$ and $\hat{u}_{1:N}$ as hidden variables, where the notation $y_{1:N} \triangleq \{y_1, y_2, \dots, y_N\}$ and similarly for other variables.

As the estimation of the state z_k ultimately yields the estimation of the noise-free input \hat{u}_k , from now onwards we represent $\hat{u}_{1:N} \triangleq \{\hat{u}_1, \hat{u}_2, \dots, \hat{u}_N\}$ as the hidden variable to avoid confusion and for ease of representation. The two processes shown by (3.1) and (3.2) can also be augmented to yield a more complex, higher order model. But the identification of a higher order model will increase the computational demand on the EM algorithm due to its iterative nature and may lead to non-unique identification of parameters. To avoid this situation, the identification is processed separately for the linear dynamic EIV system and the linear dynamic model for the noise-free input. In the next section, we present the application of the ML method using the EM algorithm, which is the main contribution of this study.

3.3 EM algorithm for linear dynamic EIV model identification

The EM algorithm provides iterative procedure for the ML estimation of both the parameters and unknown or hidden variables. It is a two-step iterative algorithm containing expectation (E) and maximization (M) steps which are repeated till convergence. In the E-step, the conditional expectation of the complete-data likelihood function, called Q function, is calculated, where the hidden variables are estimated

based on model parameters estimated from the previous iteration:

$$Q(\Theta|\Theta^{old}) = E_{C_{mis}|\Theta^{old}, C_{obs}} \log L(C_{obs}, C_{mis}|\Theta) \quad (3.4)$$

where Θ is the new set of parameters; Θ^{old} is the old set of parameters; C_{mis} refers to the missing data or hidden variables; C_{obs} refers to the observations; $L(\cdot)$ is the likelihood function.

In the M-step, the new parameters Θ^{new} are computed by maximizing the Q function as:

$$\Theta^{new} = \arg \max_{\Theta} Q(\Theta|\Theta^{old}) \quad (3.5)$$

When applying E-step for estimating the unknown input \hat{u}_k of the process (3.2), one way is to use the point estimation schemes by means of a filter or a smoother. However, this method would completely ignore the interplay between the uncertainty in the estimation of the noise-free input variable and the estimation of EIV system parameters. To account for this, we also develop a smoother for the estimation of \hat{u}_k through model (3.1) that includes the uncertainty of the input estimation from the process (3.2). In brief, this section will present the derivations of the ML method using the EM algorithm for the linear EIV state space model, the Q-function, the smoother for state space model with colored inputs, and other terms emanating from the Q-function.

3.3.1 ML method for the EIV system using the EM algorithm

To apply EM algorithm for parameter estimation of the EIV model (3.1), the Q function is presented as follows:

$$Q = E_{x_{1:N}, \hat{u}_{1:N} | y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o} [\log P(y_{1:N}, x_{1:N}, \hat{u}_{1:N}, u_{1:N}, u_{1:N}^o | \Theta, \Theta_o)] \quad (3.6)$$

where, $\{x_{1:N}, \hat{u}_{1:N}\}$ is the missing data denoted as C_{mis} ; $\{y_{1:N}, u_{1:N}, u_{1:N}^o\}$ is the observed data denoted as C_{obs} ; Θ includes the process parameters $\{A, B, C\}$, noises variance parameters $\{\Sigma_{\tilde{y}}, \Sigma_w\}$ and the hidden state initial parameters $\{\mu_{x_1}, \Sigma_{x_1}\}$. The hidden variables $x_{1:N}$ are estimated from the historical data $\{y_{1:N}\}$ and the estimates of the noise-free input $\{\hat{u}_{1:N}\}$ using the parameters of the previous iterations

Θ^{old} . The noise-free input $\{\hat{u}_{1:N}\}$ is estimated from the historical data $\{u_{1:N}, u_{1:N}^o\}$, and $\Theta_o \triangleq \{A_o, B_o, C_o, \Sigma_{\tilde{u}}, \Sigma_{w^o}\}$, for notational simplicity. We also assume that the parameter set Θ_o is known or could be estimated based on the available information about the input. For example, Θ_o can be the model of a control valve when u^o is the signal to the valve and \hat{u} is the flow rate. Next we present the derivations of E-step and M-step briefly.

E-step

The complete-data log likelihood function is derived as:

$$\begin{aligned}
& \log P(y_{1:N}, x_{1:N}, \hat{u}_{1:N}, u_{1:N}, u_{1:N}^o | \Theta, \Theta_o) \\
&= \log \prod_{k=2}^N P(y_k, x_k, \hat{u}_k, u_k, u_k^o | y_{1:k-1}, x_{1:k-1}, \hat{u}_{1:k-1}, u_{1:k-1}, u_{1:k-1}^o, \Theta, \Theta_o) \cdot \\
& \quad P(y_1, x_1, \hat{u}_1, u_1, u_1^o | \Theta, \Theta_o) \\
&= \log \prod_{k=2}^N P(y_k | x_k, \Theta) P(x_k | x_{k-1}, \hat{u}_{k-1}, \Theta) P(y_1 | x_1, \Theta) P(x_1 | \Theta). \\
& \quad \underbrace{\prod_{k=2}^N P(u_k | \hat{u}_k, \Theta_o) P(\hat{u}_k | \hat{u}_{k-1}, u_{k-1}^o, \Theta_o) P(u_k^o) P(\hat{u}_1, u_1, u_1^o | \Theta_o)}_{C_1}
\end{aligned} \tag{3.7}$$

where C_1 is independent of Θ and plays no role in the following maximization step.

Hence, disposing the term C_1 , the log likelihood function is presented as follows:

$$= \sum_{k=1}^N \log P(y_k | x_k, \Theta) + \sum_{k=1}^{N-1} \log P(x_{k+1} | x_k, \hat{u}_k, \Theta) + \log P(x_1 | \Theta) \tag{3.8}$$

Using the Gaussian assumptions for state and output profiles:

$$\begin{aligned}
P(y_k | x_k, \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{\tilde{y}})}} \exp\{-(y_k - Cx_k)^T \Sigma_{\tilde{y}}^{-1} (y_k - Cx_k) / 2\} \\
P(x_{k+1} | x_k, \hat{u}_k, \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{\tilde{w}})}} \exp\{-(x_{k+1} - Ax_k - B\hat{u}_k)^T \Sigma_{\tilde{w}}^{-1} (x_{k+1} - \\
& \quad Ax_k - B\hat{u}_k) / 2\} \\
P(x_1 | \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{x_1})}} \exp\{-(x_1 - \mu_{x_1})^T \Sigma_{x_1}^{-1} (x_1 - \mu_{x_1}) / 2\}
\end{aligned} \tag{3.9}$$

Substituting (3.9) in (3.8), the Q function is now written as:

$$\begin{aligned}
Q &= E_{x_{1:N}, \hat{u}_{1:N} | y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o} [\log P(y_{1:N}, x_{1:N}, \hat{u}_{1:N}, u_{1:N}, u_{1:N}^o | \Theta, \Theta_o)] \\
&= -\frac{N}{2} \log 2\pi \det(\Sigma_{\tilde{y}}) - \underbrace{\frac{1}{2} \sum_{k=1}^N E[(y_k - Cx_k)^T \Sigma_{\tilde{y}}^{-1} (y_k - Cx_k)]}_{t_1} \\
&\quad - \frac{N-1}{2} \cdot \log 2\pi \det(\Sigma_w) - \underbrace{\frac{1}{2} \sum_{k=1}^{N-1} E[(x_{k+1} - Ax_k - B\hat{u}_k)^T \Sigma_w^{-1} (x_{k+1} - Ax_k - B\hat{u}_k)]}_{t_2} \\
&\quad - \frac{1}{2} \log 2\pi \det(\Sigma_{x_1}) - \underbrace{\frac{1}{2} E[(x_1 - \mu_{x_1})^T \Sigma_{x_1}^{-1} (x_1 - \mu_{x_1})]}_{t_3} + E(\log C_1)
\end{aligned} \tag{3.10}$$

where the conditional expectation notations are simplified as $E(\cdot)$ for ease of representation. The conditional expectation terms arising in Q function are computed in subsequent subsections. Next we proceed to M-step for maximization.

M-step

In the M-step, the Q function is maximized using unconstrained first order optimality conditions, as we do not assume any constraints on the parameters Θ . Before taking derivatives of the Q function, the following transformations are made to the quadratic terms represented by underbraces in equation (3.10):

$$\begin{aligned}
t_1 &= \frac{1}{2} \sum_{k=1}^N E\{\text{Tr}[\Sigma_{\tilde{y}}^{-1} (y_k - Cx_k)(y_k - Cx_k)^T]\} \\
t_2 &= \frac{1}{2} \sum_{k=1}^{N-1} E\{\text{Tr}[\Sigma_w^{-1} (x_{k+1} - [A \ B] \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix})(x_{k+1} - [A \ B] \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix})^T]\} \tag{3.11} \\
t_3 &= \frac{1}{2} E\{\text{Tr}[\Sigma_{x_1}^{-1} (x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T]\}
\end{aligned}$$

Derivatives of the Q function are taken over parameters Θ for maximization, as follows:

$$\frac{dQ}{d\bar{A}} = \frac{d\left\{-\frac{1}{2} \sum_{k=1}^{N-1} E\{\text{Tr}[\Sigma_w^{-1} (x_{k+1} - \bar{A}\bar{x}_k)(x_{k+1} - \bar{A}\bar{x}_k)^T]\}\right\}}{d\bar{A}} \tag{3.12}$$

$$\frac{dQ}{dC} = \frac{d\left\{-\frac{1}{2} \sum_{k=1}^N E\{\text{Tr}[\Sigma_{\tilde{y}}^{-1} (y_k - Cx_k)(y_k - Cx_k)^T]\}\right\}}{dC} \tag{3.13}$$

$$\frac{dQ}{d\Sigma_{\tilde{y}}} = \frac{d \left\{ -\frac{N}{2} \cdot \log 2\pi \det(\Sigma_{\tilde{y}}) - \frac{1}{2} \sum_{k=1}^N E\{\text{Tr}[\Sigma_{\tilde{y}}^{-1}(y_k - Cx_k)(y_k - Cx_k)^T]\} \right\}}{d\Sigma_{\tilde{y}}} \quad (3.14)$$

$$\frac{dQ}{d\Sigma_w} = \frac{d \left\{ -\frac{N-1}{2} \cdot \log 2\pi \det(\Sigma_w) - \frac{1}{2} \sum_{k=1}^{N-1} E\{\text{Tr}[\Sigma_w^{-1}(x_{k+1} - \bar{A}\bar{x}_k)(x_{k+1} - \bar{A}\bar{x}_k)^T]\} \right\}}{d\Sigma_w} \quad (3.15)$$

$$\frac{dQ}{d\mu_{x_1}} = \frac{d \left\{ -\frac{1}{2} E\{\text{Tr}[\Sigma_{x_1}^{-1}(x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T]\} \right\}}{d\mu_{x_1}} \quad (3.16)$$

$$\frac{dQ}{d\Sigma_{x_1}} = \frac{d \left\{ -\frac{1}{2} \log 2\pi \det(\Sigma_{x_1}) - \frac{1}{2} E\{\text{Tr}[\Sigma_{x_1}^{-1}(x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T]\} \right\}}{d\Sigma_{x_1}} \quad (3.17)$$

where \bar{A} denotes $\begin{bmatrix} A & B \end{bmatrix}$; \bar{x}_k denotes $\begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}$.

The detailed steps are omitted here for brevity. Doing algebraic manipulations using matrix calculus and by setting the derivatives to zero, the parameter updating equations are obtained as follows:

$$\begin{bmatrix} A & B \end{bmatrix} = \left[\sum_{k=1}^{N-1} E(x_{k+1} \begin{bmatrix} x_k^T & \hat{u}_k^T \end{bmatrix}) \right] \cdot \left[\sum_{k=1}^{N-1} E\left(\begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}^T \right) \right]^{-1} \quad (3.18)$$

$$C = \left[\sum_{k=1}^N E(y_k x_k^T) \right] \cdot \left[\sum_{k=1}^N E(x_k x_k^T) \right]^{-1} \quad (3.19)$$

$$\begin{aligned} \Sigma_w &= \frac{1}{N-1} E\left\{ \sum_{k=1}^{N-1} (x_{k+1} - \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix})(x_{k+1} - \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix})^T \right\} \\ &= \frac{1}{N-1} \left\{ \sum_{k=1}^{N-1} E(x_{k+1} x_{k+1}^T) - \left[\sum_{k=1}^{N-1} E(x_{k+1} \begin{bmatrix} x_k^T & \hat{u}_k^T \end{bmatrix}) \right] \cdot \right. \\ &\quad \left. \left[\sum_{k=1}^{N-1} E\left(\begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}^T \right) \right]^{-1} \cdot \left[\sum_{k=1}^{N-1} E(x_{k+1} \begin{bmatrix} x_k^T & \hat{u}_k^T \end{bmatrix}) \right]^T \right\} \end{aligned} \quad (3.20)$$

where the parameters $\{A, B\}$ in equation (3.20) are updated from equation (3.18).

$$\begin{aligned} \Sigma_{\tilde{y}} &= \frac{1}{N} E\left\{ \sum_{k=1}^N (y_k - Cx_k)(y_k - Cx_k)^T \right\} \\ &= \frac{1}{N} \left\{ \sum_{k=1}^N E(y_k y_k^T) - \left[\sum_{k=1}^N E(y_k x_k^T) \right] \cdot \left[\sum_{k=1}^N E(x_k x_k^T) \right]^{-1} \cdot \left[\sum_{k=1}^N E(y_k x_k^T) \right]^T \right\} \end{aligned} \quad (3.21)$$

where parameter C in equation (3.21) is updated from equation (3.19).

For simplicity, the process noise $w(t)$ is assumed to have the covariance as $\Sigma_w = \sigma_w^2 \cdot I$, therefore, σ_w^2 is calculated as:

$$\sigma_w^2 = \frac{1}{(N-1) \cdot n} \text{Tr} \left\{ \sum_{k=1}^{N-1} E(x_{k+1}x_{k+1}^T) - \left[\sum_{k=1}^{N-1} E(x_{k+1} \begin{bmatrix} x_k^T & \hat{u}_k^T \end{bmatrix}) \right] \cdot \left[\sum_{k=1}^{N-1} E \left(\begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix} \begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}^T \right) \right]^{-1} \cdot \left[\sum_{k=1}^{N-1} E(x_{k+1} \begin{bmatrix} x_k^T & \hat{u}_k^T \end{bmatrix}) \right]^T \right\} \quad (3.22)$$

The initial state estimation parameters are updated as:

$$\mu_{x_1} = E(x_1) \quad (3.23)$$

$$\Sigma_{x_1} = E[(x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T] \quad (3.24)$$

To compute the parameters using the parameter updating equations (3.18-3.22), we need to evaluate conditional expectation terms like $E(x_k \hat{u}_k^T)$, $E(x_k x_k^T)$, $E(x_{k+1} \hat{u}_k^T)$, $E(x_{k+1} x_k^T)$, etc. In this situation, the hidden states x_k as well as unknown noise-free inputs \hat{u}_k should also be estimated. Methods for the same are presented in the next subsections.

3.3.2 Smoothing for state space model with colored stochastic inputs

This section derives a smoother for the estimation of the state x_k from the noise-corrupted output y_k and colored input \hat{u}_k for the state space model (3.1). The relationship between variables in (3.1) and (3.2) can be illustrated by Figure 3.3.

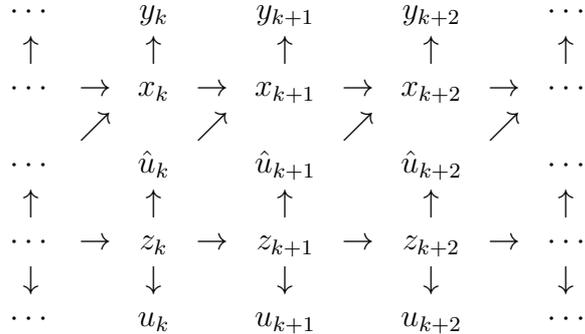


Figure 3.3: The process variables relationship diagram

To calculate the conditional expectation of Q function, we bring back the hidden variable z_k due to unavailability of the accurate value of noise-free input \hat{u}_k , where the missing data posterior probability $P(x_{1:N}, \hat{u}_{1:N}|y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o)$ is yielded from $P(x_{1:N}, z_{1:N}|y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o)$ since $\hat{u}_k = C_o z_k$. As stated previously, for the ease of computation, we apply the smoother for both $x_{1:N}$ and $z_{1:N}$ in a separated way as:

$$\begin{aligned} P(x_{1:N}, z_{1:N}|y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o) &= P(x_{1:N}|z_{1:N}, y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o) \cdot \\ &P(z_{1:N}|y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o) \\ &\approx P(x_{1:N}|z_{1:N}, y_{1:N}, \Theta^{old})P(z_{1:N}|u_{1:N}, u_{1:N}^o, \Theta_o) \end{aligned} \quad (3.25)$$

Here we have used the following facts for derivation: (i) the conditional probability equation $P(x_{1:N}|z_{1:N}, y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o) = P(x_{1:N}|z_{1:N}, y_{1:N}, \Theta^{old})$ holds according to Figure 3.3; (ii) u_k is the more direct observation for z_k according to Figure 3.3, therefore we do the following approximation $P(z_{1:N}|y_{1:N}, u_{1:N}, u_{1:N}^o, \Theta^{old}, \Theta_o) \approx P(z_{1:N}|u_{1:N}, u_{1:N}^o, \Theta_o)$. The Kalman filter and smoother are used to estimate distribution parameters of $P(z_{1:N}|u_{1:N}, u_{1:N}^o, \Theta_o)$ [28] [29]. For the calculation of the distribution parameters of $P(x_{1:N}|z_{1:N}, y_{1:N}, \Theta^{old})$, a smoother for $x_{1:N}$ is proposed, employing fixed period smoothing method [28], as follows:

$$x_{k|N} \triangleq E[X_k|y_{1:N}, u_{1:N}, u_{1:N}^o] = x_{k|k} + J_k^x(x_{k+1|N} - x_{k+1|k}) \quad (3.26)$$

$$P_{k|N}^x \triangleq V[X_k|y_{1:N}, u_{1:N}, u_{1:N}^o] = P_{k|k}^x + J_k^x(P_{k+1|N}^x - P_{k+1|k}^x)J_k^{xT} \quad (3.27)$$

where $E(\cdot)$ denotes the expectation operation; $V(\cdot)$ refers to the variance for scalar-variable or covariance for vector-variable; $x_{k|k}$, $x_{k+1|k}$ are the filtered and predicted states, and $P_{k|k}^x$, $P_{k+1|k}^x$ are their corresponding covariances, respectively; the uppercase X_k denotes the variable and x_k is the corresponding realization, and similarly for other variables like Y_k , Z_k . The smoother gain J_k^x is derived as:

$$\begin{aligned} J_k^x &= Cov[X_k, X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o](V[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o])^{-1} \\ &= Cov[X_k, AX_k + BC_o Z_k + w_k|y_{1:k}, u_{1:k}, u_{1:k}^o](V[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o])^{-1} \\ &= (P_{k|k}^x A^T + Cov[X_k, Z_k|y_{1:k}, u_{1:k}, u_{1:k}^o]C_o^T B^T) \cdot (P_{k+1|k}^x)^{-1} \end{aligned} \quad (3.28)$$

Each of the terms in the equation (3.26, 3.27) is derived separately, starting with

filtering step.

Filtering step

Below, we present the derivation of the filter for state space model with the colored input using the conditional expectation theorem[30]. Considering the process dynamics (3.1), the filtered state expression can be written as:

$$\begin{aligned}
x_{k+1|k+1} &\triangleq E[X_{k+1}|y_{k+1}, y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= E[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] + K_{k+1}^x \{y_{k+1} - E[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o]\} \\
&= E[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] + K_{k+1}^x \{y_{k+1} - E[CX_{k+1} + \tilde{y}_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o]\} \\
&= x_{k+1|k} + K_{k+1}^x (y_{k+1} - Cx_{k+1|k})
\end{aligned} \tag{3.29}$$

with the covariance:

$$\begin{aligned}
P_{k+1|k+1}^x &\triangleq V[X_{k+1}|y_{k+1}, y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= V[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] - K_{k+1}^x V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] K_{k+1}^{xT} \\
&= P_{k+1|k}^x - K_{k+1}^x V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] K_{k+1}^{xT}
\end{aligned} \tag{3.30}$$

where the variance of output $V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o]$ is calculated as:

$$\begin{aligned}
V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] &= V[CX_{k+1} + \tilde{y}_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= CP_{k+1|k}^x C^T + R
\end{aligned} \tag{3.31}$$

and the filter gain K_{k+1}^x is determined as:

$$\begin{aligned}
K_{k+1}^x &= Cov[X_{k+1}, Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] \cdot (V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o])^{-1} \\
&= Cov[X_{k+1}, CX_{k+1} + \tilde{y}_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] \cdot (V[Y_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o])^{-1} \\
&= P_{k+1|k}^x C^T (CP_{k+1|k}^x C^T + R)^{-1}
\end{aligned} \tag{3.32}$$

Thus:

$$x_{k+1|k+1} = x_{k+1|k} + P_{k+1|k}^x C^T (CP_{k+1|k}^x C^T + R)^{-1} (y_{k+1} - Cx_{k+1|k}) \tag{3.33}$$

$$P_{k+1|k+1}^x = P_{k+1|k}^x - P_{k+1|k}^x C^T (CP_{k+1|k}^x C^T + R)^{-1} CP_{k+1|k}^x \tag{3.34}$$

Prediction step

Next we present steps for the state prediction. For the process (3.1), the state prediction is derived as follows:

$$\begin{aligned}
x_{k+1|k} &\triangleq E[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= E[AX_k + BC_o Z_k + w_k | y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= E[AX_k | y_{1:k}, u_{1:k}, u_{1:k}^o] + E[BC_o Z_k | u_{1:k}, u_{1:k}^o] + E[w_k] \\
&= Ax_{k|k} + BC_o z_{k|k}
\end{aligned} \tag{3.35}$$

where $z_{k|k}$ is the Kalman filter for z_k in process (3.2). The covariance of the prediction is presented as:

$$\begin{aligned}
P_{k+1|k}^x &\triangleq V[X_{k+1}|y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= V[AX_k + BC_o Z_k + w_k | y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= V[AX_k | y_{1:k}, u_{1:k}, u_{1:k}^o] + V[BC_o Z_k | u_{1:k}, u_{1:k}^o] + V[w_k] + \\
&\quad Cov[AX_k, BC_o Z_k | y_{1:k}, u_{1:k}, u_{1:k}^o] + Cov[BC_o Z_k, AX_k | y_{1:k}, u_{1:k}, u_{1:k}^o] \\
&= AP_{k|k}^x A^T + BC_o P_{k|k}^z C_o^T B^T + Q + A \cdot \underbrace{Cov[X_k, Z_k | y_{1:k}, u_{1:k}, u_{1:k}^o]}_{P_{k|k}^{xz}} \cdot (BC_o)^T + \\
&\quad BC_o \cdot Cov[Z_k, X_k | y_{1:k}, u_{1:k}, u_{1:k}^o] \cdot A^T
\end{aligned} \tag{3.36}$$

where $P_{k|k}^z$ is the filtered covariance for z_k in process (3.2).

To estimate $P_{k+1|k}^x$ using (3.36), we need to compute the covariance of the estimation denoted as $P_{k|k}^{xz}$. The covariance for X_k and Z_k , $P_{k|k}^{xz}$, is derived using equations (3.1), (3.2), (3.33), (3.34), (3.35) and the Kalman filter for Z_k as well as the initial-

ization assumption that X_1 and Z_1 are uncorrelated, as follows:

$$\begin{aligned}
P_{k|k}^{xz} &= E[(X_k - x_{k|k})(Z_t - z_{k|k})^T] \\
&= E\{[(A - K_k CA)(X_{k-1} - x_{k-1|k-1}) + (BC_o - K_k CBC_o) \cdot \\
&\quad (Z_{k-1} - z_{k-1|k-1}) - K_k^x \tilde{y}_k + (I - K_t C)w_{k-1}] \cdot [(A_o - K_k^z C_o A_o) \cdot \\
&\quad (Z_{k-1} - z_{k-1|k-1}) - K_k^z \tilde{u}_k + (I - K_k^z C_o)w_{k-1}^o]^T\} \\
&= E\{[(A - K_k CA)(X_{k-1} - X_{k-1|k-1})] \cdot [(A_o - K_k^z C_o A_o) \cdot \\
&\quad (Z_{k-1} - Z_{k-1|k-1})]^T\} + (BC_o - K_k CBC_o) \cdot \\
&\quad \underbrace{E[(Z_{k-1} - z_{k-1|k-1})(Z_{k-1} - z_{k-1|k-1})^T]}_{P_{k-1|k-1}^z} \cdot (A_o - K_k^z C_o A_o)^T \\
&= (A - K_k CA) \cdot P_{k-1|k-1}^{xz} \cdot (A_o - K_k^z C_o A_o)^T + \\
&\quad (BC_o - K_k CBC_o) P_{k-1|k-1}^z (A_o - K_k^z C_o A_o)^T
\end{aligned} \tag{3.37}$$

where K_k^z is Kalman filter gain for z_k in process (3.2).

Integration of the filtering (3.33, 3.34) and prediction (3.35), (3.36) steps in conjunction with the smoother expressions (3.26, 3.27) provides the smoother of X_k for state space model (3.1) with inputs \hat{u}_k .

3.3.3 Computation of the conditional expectation terms in the Q function

In this subsection we evaluate the various conditional expectation terms appearing in equations (3.18-3.24).

We begin with the expression for $E(X_k X_k^T | C_{obs})$. Consider the smoothed covariance of state x_k :

$$\begin{aligned}
P_{k|N}^x &= E[(X_k - x_{k|N})(X_k - x_{k|N}) | C_{obs}] \\
&= E(X_k X_k^T | C_{obs}) - E(X_k x_{k|N}^T | C_{obs}) - \\
&\quad E(x_{k|N} X_k^T | C_{obs}) + E(x_{k|N} x_{k|N}^T | C_{obs}) \\
&= E(X_k X_k^T | C_{obs}) - x_{k|N} x_{k|N}^T \\
&\Rightarrow E(X_k X_k^T | C_{obs}) = x_{k|N} x_{k|N}^T + P_{k|N}^x
\end{aligned} \tag{3.38}$$

Next we consider $E(\hat{u}_k \hat{u}_k^T | C_{obs})$:

$$\begin{aligned}
P_{k|N}^z &\triangleq E[(Z_k - z_{k|N})(Z_k - z_{k|N}) | C_{obs}] \\
&= E(Z_k Z_k^T | C_{obs}) - E(Z_k z_{k|N}^T | C_{obs}) - \\
&\quad E(z_{k|N} Z_k^T | C_{obs}) + E(z_{k|N} z_{k|N}^T | C_{obs}) \\
&= E(Z_k Z_k^T | C_{obs}) - z_{k|N} z_{k|N}^T \\
\Rightarrow E(\hat{u}_k \hat{u}_k^T | C_{obs}) &= E(C_o Z_k Z_k^T C_o^T | C_{obs}) \\
&= C_o z_{k|N} z_{k|N}^T C_o^T + C_o P_{k|N}^z C_o^T
\end{aligned} \tag{3.39}$$

Following a similar approach mentioned above, the other conditional expectation terms are derived below:

$$E(X_{k+1} X_k^T | C_{obs}) = x_{k+1|N} x_{k|N}^T + M_{k+1|N}^x \tag{3.40}$$

where $M_{k+1|N}^x \triangleq E[(X_{k+1} - x_{k+1|N})(X_k - x_{k|N})^T | C_{obs}]$.

$$E(X_k \hat{u}_k^T | C_{obs}) = x_{k|N} z_{k|N}^T C_o^T + P_{k|N}^{xz} C_o^T \tag{3.41}$$

where $P_{k|N}^{xz} \triangleq E[(X_k - x_{k|N})(Z_k - z_{k|N})^T | C_{obs}]$.

$$E(X_{k+1} \hat{u}_k^T | C_{obs}) = x_{k+1|N} z_{k|N}^T C_o^T + M_{k+1|N}^{xz} C_o^T \tag{3.42}$$

where $M_{k+1|N}^{xz} \triangleq E[(X_{k+1} - x_{k+1|N})(Z_k - z_{k|N})^T | C_{obs}]$.

The covariance term $P_{k|N}$ is derived using equation (3.27), and the covariance $P_{k|N}^z$ is derived by applying the Kalman smoother on (3.2). The expressions for the lag-one covariance terms $M_{k|N}$ and $M_{k|N}^{xz}$ and the cross covariance term $P_{k|N}^{xz}$ are presented as follows, while detailed derivations are provided in Appendix B.

$$P_{k|N}^{xz} = P_{k|k}^{xz} + J_k^x (P_{k+1|N}^{xz} - A P_{k|k}^{xz} A_o^T - B C_o P_{k|k}^z A_o^T) J_k^{zT} \tag{3.43}$$

$$M_{k+1|N}^{xz} \triangleq E[(X_{k+1} - x_{k+1|N})(Z_k - z_{k|N})^T | C_{obs}] \tag{3.44}$$

$$\begin{aligned}
&= (I - K_{k+1}^x + J_{k+1}^x A K_{k+1}^x C) (A P_{k|k}^{xz} + B C_o P_{k|k}^z) + J_{k+1}^x B C_o \cdot \\
&\quad K_{k+1}^z C_o A_o P_{k|k}^z + J_{k+1}^x (M_{k+2|N}^{xz} - A P_{k+1|k}^{xz} - B C_o P_{k+1|k}^z) J_k^{zT}
\end{aligned} \tag{3.45}$$

$$M_{k+1|N}^x \triangleq E[(X_{k+1} - x_{k+1|N})(X_k - x_{k|N})^T | C_{obs}] \tag{3.46}$$

$$= (I - K_{k+1}^x C + J_{k+1}^x A K_{k+1}^x C) (A P_{k|k}^x + B C_o P_{k|k}^{xzT}) +$$

$$\begin{aligned}
& J_{k+1}^x BC_o K_{k+1}^z A_o P_{k|k}^{xzT} + J_{k+1}^x [M_{k+2|N}^x - AP_{k+1|k}^x - \\
& BC_o (AP_{k|k}^{xz} A_o^T + BC_o P_{k|k}^z A_o^T)^T] J_k^{xT}
\end{aligned} \tag{3.47}$$

where J_k^z is the Kalman smoother gain for Z_k ; $P_{k+1|k}^z$ is the predictor for the covariance of Z_k . The initialization for the recursive computations of the lag-one covariance smoother using equations (3.45, 3.47), $M_{N|N}^{xz}$ and $M_{N|N}^x$, are also presented as follows:

$$\begin{aligned}
M_{N|N}^x = & (I - K_N^x C) [(AP_{N-1|N-1}^{xz} + BC_o P_{N-1|N-1}^z) - P_{N-1|N-1}^x (J_{N-1}^x K_N^x C)^T] - \\
& K_N^x R (J_{N-1}^x K_N^x)^T
\end{aligned} \tag{3.48}$$

$$M_{N|N}^{xz} = (I - K_N^x C) (AP_{N-1|N-1}^{xz} + BC_o P_{N-1|N-1}^z) (I - J_{N-1}^z K_N^z C_o A_o)^T \tag{3.49}$$

while $P_{N|N}^{xz}$ is computed using equation (3.37). Having presented the necessary derivations, the next subsection elicits the key steps of the proposed method for identification, denoted as Algorithm 3.

3.3.4 Algorithm

The algorithmic representation of the complete steps of the ML estimation of linear EIV system parameters using the EM algorithm is concisely presented as Algorithm 3.

3.4 Validations

In this section, the proposed approach is compared with the subspace EIV method [18] as well as the bias-eliminated least squares method (BELS) [21] using two different simulation examples as well as through an experimental study. The proposed smoother for state space model with colored inputs is also validated through the numerical examples. The prediction error cost is used to compare different methods, given by:

$$E_N(\hat{\theta}_l) \triangleq \frac{1}{pN} \sum_{k=1}^N (y_k - \hat{y}_{k|k-1}(\hat{\theta}_k))^T (y_k - \hat{y}_{k|k-1}(\hat{\theta}_k)) \tag{3.50}$$

where, $\hat{\theta}_l$ is estimated parameter for the l th simulation data; p is the dimension of the output; N is the data length for each estimation experiments; $\hat{y}_{k|k-1}$ is the one-step ahead output prediction.

Algorithm 3: Proposed EM-based method

Data: $y_{1:N}, u_{1:N}, u_{1:N}^o$

Result: Θ

Initialization :

- (i) Initial estimation of Θ based on subspace EIV method using data $y_{1:N}, u_{1:N}$;
- (ii) Identify input dynamic model (Θ_o) using data $u_{1:N}^o, u_{1:N}$ if the input dynamic model is not known;
- (iii) Estimation of the distribution parameters of $Z_{1:N}$ is obtained by applying Kalman smoother on (3.2);
- (iiii) Initial guesses of noise distribution parameters and initial states.

while compare newly updated parameters with the previous ones, till the desired convergence metric is reached, **do**

E – step:

for $k = 2, 3, \dots, N$ **do**

 Calculate the predictor of X_k (3.35, 3.36), the filter of X_k (3.33, 3.34) and the cross covariance of X_k and Z_k (3.37).

end

 Calculate the initialization of smoother computation for $k = N$.

for $k = N - 1, 2, \dots, 1$ **do**

 Calculate the smoother of X_k (3.26,3.27), the covariance smoother of X_k and Z_k (3.43), the lag-one covariance smoother of X_k and Z_k (3.45) and the lag-one covariance smoother of X_k (3.47).

end

 Calculate the conditional expectation quadratic terms for hidden variables based on equation (3.38), (3.39), (3.40), (3.41) and (3.42).

M – step:

 Calculate the updated model parameters Θ using equation (3.18), (3.19), (3.20), (3.21) and (3.24).

end

3.4.1 Simulation: Example 1

The first example we have chosen is a simple first order EIV state space model given as:

$$\begin{aligned}x_{k+1} &= 0.8x_k + 0.8\hat{u}_k + w_k \\y_k &= x_k + \tilde{y}_k\end{aligned}\tag{3.51}$$

where, the variances of w_k and \tilde{y}_k are set to be 2% of the state variance and 5% of the output variance ($\Sigma_w = 1$, $\Sigma_{\tilde{y}} = 5$), respectively, with the input-output discrete transfer function in zeros and poles form:

$$G(z) = \frac{0.8}{z - 0.8}\tag{3.52}$$

The input parametric model of the first order state space model is:

$$\begin{aligned}z_{k+1} &= 0.8z_k + 0.8u_k^o + w_k^o \\u_k &= z_k + \tilde{u}_k\end{aligned}\tag{3.53}$$

where, the variances of w_k^o is set to be around 2% of the state variance ($\Sigma_{w^o} = 0.1$), and input noise \tilde{u}_k variance is set at two different levels as around 5% ($\Sigma_{\tilde{u}} = 0.5$) and 10% ($\Sigma_{\tilde{u}} = 1$) of the noise-free input \hat{u}_k variance for representing two different scenarios; the input u_k^o is generated from the random binary sequence with shifts between -1 and 1.

The parameter for the input process is estimated first through the data $\{u_{1:N}^o, u_{1:N}\}$. Then using the results of the subspace EIV method as the initial value, the proposed EM-based method is applied on the process described in (3.51). 50 Monte-Carlo runs are considered, and the prediction error results of the different simulations are presented as blue asterisks in Figure 3.4 and Figure 3.5 for $\Sigma_{\tilde{u}} = 0.5$ and $\Sigma_{\tilde{u}} = 1$, respectively. It can be seen that the proposed EM-based method has smaller prediction error cost than the subspace EIV method and the BELS method for both different input noise levels. The smoother result is also presented in Figure 3.6, with blue solid line indicating true state values and red cycle line is the smoothed estimates. This result also shows the efficacy of the proposed smoothing method.

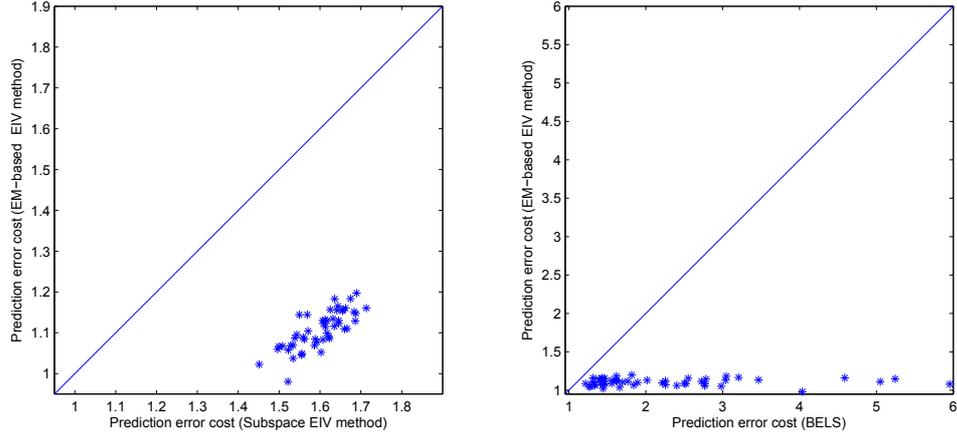


Figure 3.4: Prediction error cost for first order EIV system (input variance: 5%)

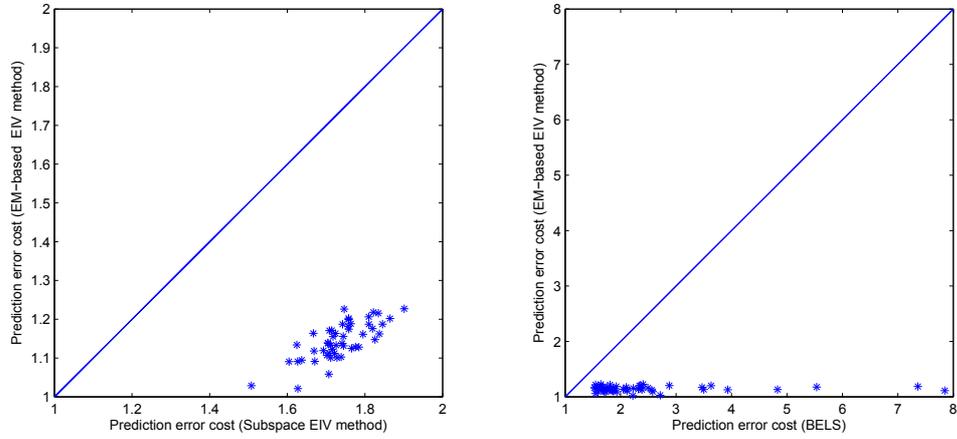


Figure 3.5: Prediction error cost for first order EIV system (input variance: 10%)

3.4.2 Simulation: Example 2

The second example is a second order state space model:

$$A = \begin{bmatrix} 0.4 & 0.4472 \\ 0 & 0.8 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = [0.3578 \quad 0.8] \quad (3.54)$$

with the input-output discrete transfer function in zeros and poles form:

$$G(z) = \frac{0.8(z - 0.2)}{(z - 0.4)(z - 0.8)} \quad (3.55)$$

where, the variances of w_k and \tilde{y}_k are set to be around 2% of the state variance and 2% of the output variance ($\Sigma_w = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\Sigma_{\tilde{y}} = 1.8$); the input measurement noise variance $\sigma_{\tilde{u}}^2$ is set at two different levels as around 5% ($\Sigma_{\tilde{u}} = 0.5$) and 10% ($\Sigma_{\tilde{u}} = 1$)

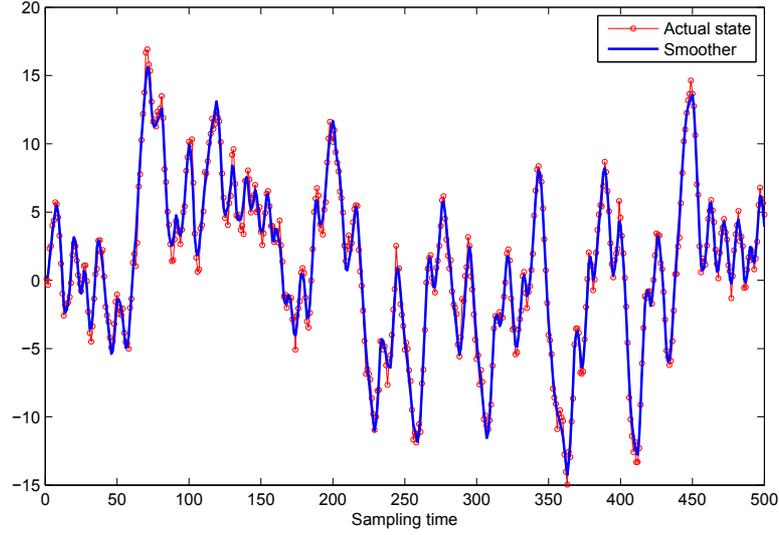


Figure 3.6: State estimation for the first-order EIV system (input variance: 10%)

of the noise-free input \hat{u}_k variance for representing two different scenarios, and the input dynamic model is the same as that presented in (3.53).

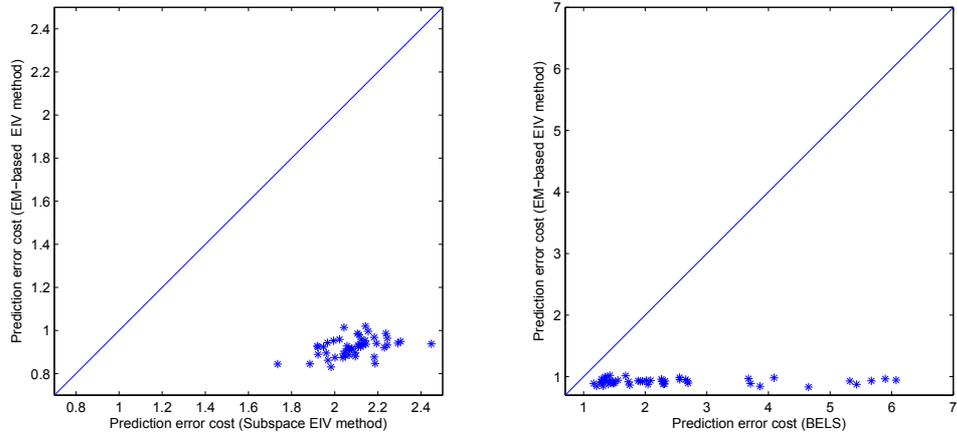


Figure 3.7: Prediction error cost for second order EIV system (input variance: 5%)

The prediction error costs of 50 numerical simulations are presented as blue asterisks in Figure 3.7 and Figure 3.8 when $\Sigma_{\bar{u}} = 0.5$ and $\Sigma_{\bar{u}} = 1$, respectively. The results show that the EM-based method has smaller prediction errors than the subspace EIV method and the BELS method. For comparison purposes, the statistical values of the prediction cost for the two examples are also tabulated in Table 3.1, which fur-

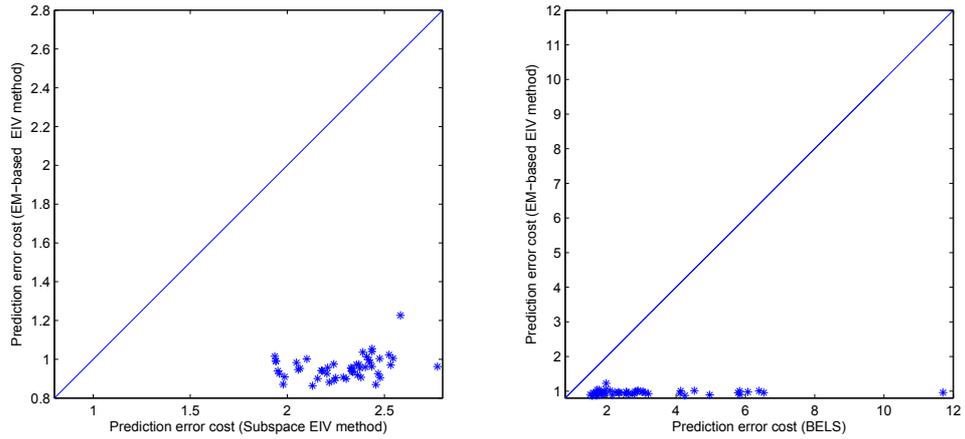


Figure 3.8: Prediction error cost for second order EIV system (input variance: 10%)

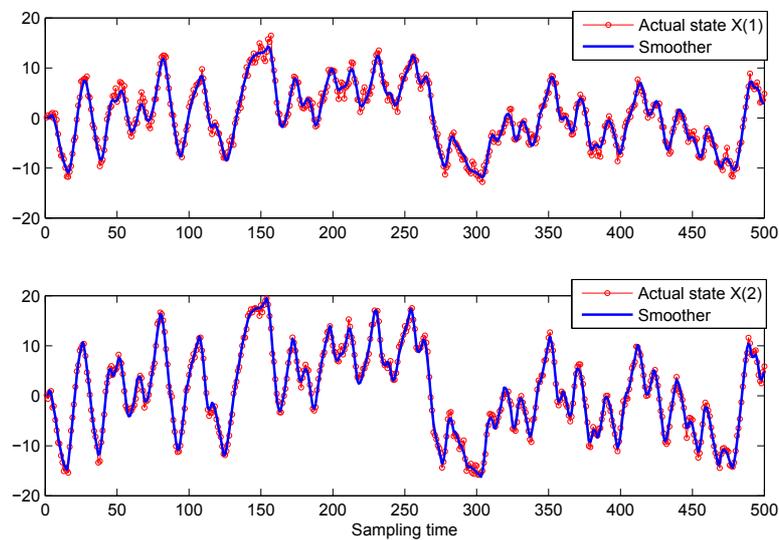


Figure 3.9: State estimation for the second-order EIV system (input variance: 10%)

ther underlines that the proposed EM-based method outperforms the subspace EIV method. The smoother result for the second order EIV system is presented in Figure 3.9, where the blue solid line shows the true state profile and red circle line indicates the smoothed estimate. The result also demonstrates the effectiveness of the proposed smoothing technique.

Table 3.1: Mean prediction costs of the simulation results

Order	Input noise level	EM-based method	Subspace method	BELS
2	10%	0.958	2.2798	3.0877
2	5%	0.9229	2.093	2.6545
1	10%	1.1446	1.7356	2.4676
1	5%	1.1074	1.6007	2.2521

3.4.3 Experiment: Tank system

To illustrate the effectiveness of the proposed EM-based method, an EIV system identification experiment is designed and performed on a tank system. The schematic diagram of the process is presented in Figure 3.10. The tank system is a nonlinear process and the linearization approximation is applied around an appropriate working point to obtain linear models. In the experiment, Tank2 is considered as the EIV process with the water level L_2 measured by the sensor $LT2$ as the output and the water level L_3 measured by the sensor $LT3$ as the noise-corrupted input, with Tank3 being the input generator. The referred input for Tank3 is the water flow rate F_2 measured by the sensor $FC2$. In the configuration, valves $V1$, $V2$, $V6$ and $V7$ are closed, while valves $V3$, $V4$, $V5$, $V8$ and $V9$ are open, making Tank1 (shaded in Figure 3.10) isolated from the experiment. The experiment is conducted and 1500 samples of data are collected and the input source, the noise-corrupted EIV system input and output are presented in Figure 3.11, where the L_3 is added with the simulated white noise with two different noise level as 5% and 10% of the input variance as Figure 3.12 to demonstrate the tolerability of the proposed EIV identification method to different input noises.

The model parameters for the input generation process are estimated firstly. For the identification of the EIV system, the first 1000 sampling data are used as the training data to estimate the parameters Θ using the proposed EM-based method. The cross validation results on the remaining data set are illustrated along with the self validations in Figure 3.13 and Figure 3.14 by plotting the predictions from different approaches with the plant data L_2 . By running 10 Monte Carlo simulations on the simulated-noise added data with different noise level, the average mean square errors (MSE) for different methods are presented in Table 3.2, where one can see

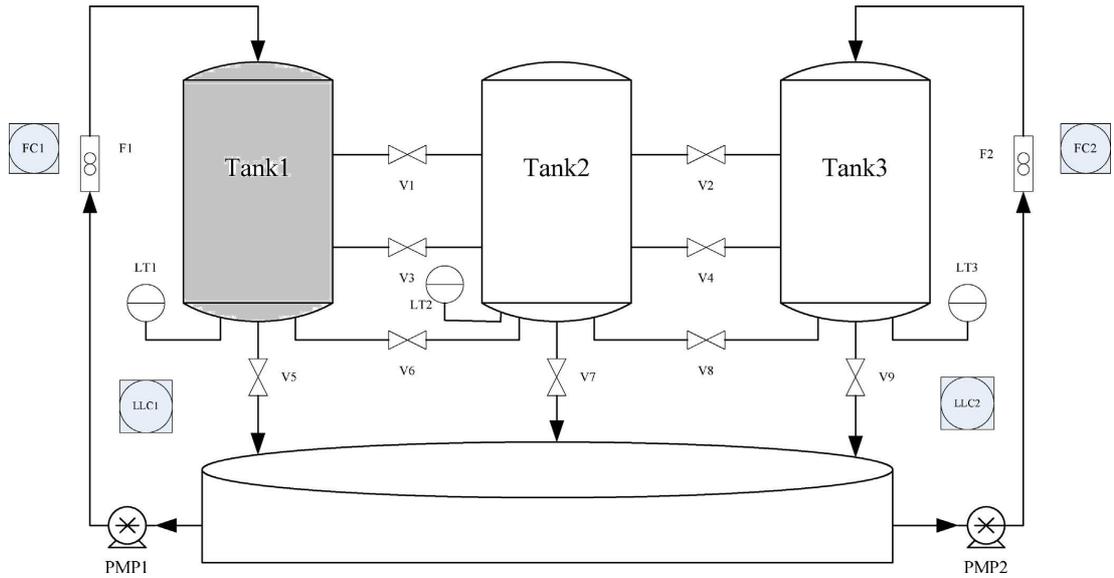


Figure 3.10: Schematic Diagram of the tank system

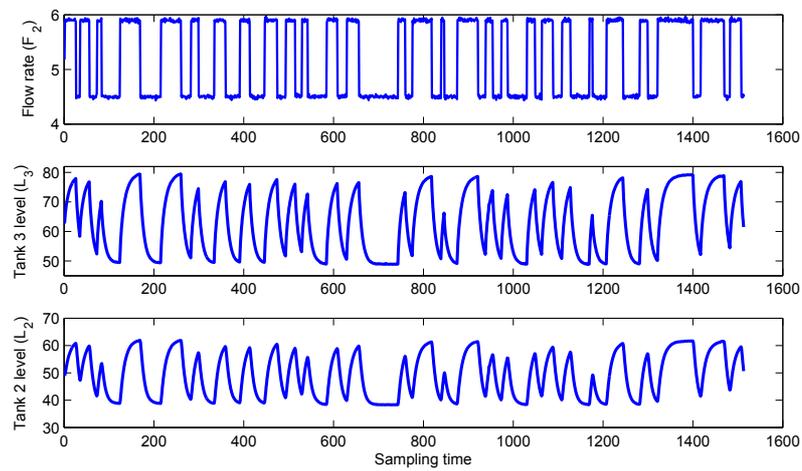


Figure 3.11: Input and output data

that the predictions using the EM-based method has better performance in matching with the plant data than the predictions using the subspace method and the BELS method. This experiment validates that the proposed EM-based method has superior performance over both the subspace-based method and the BELS method.

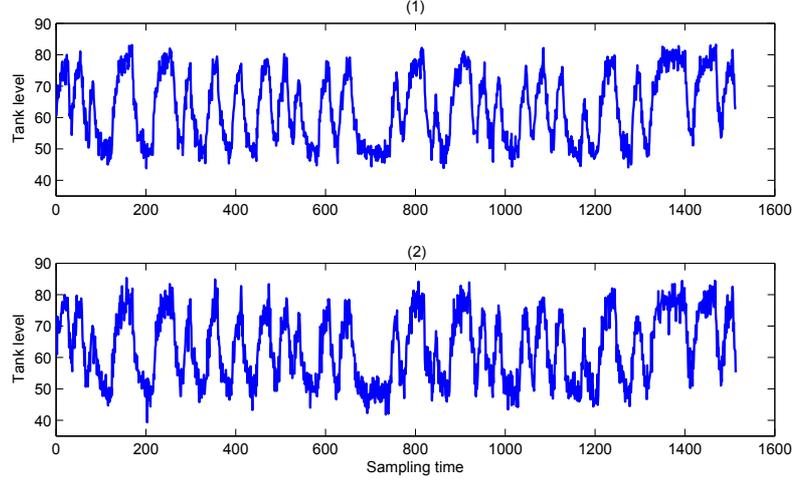


Figure 3.12: Simulated noise-corrupted input: (1) 5% of input variance; (2) 10% of input variance

Table 3.2: Average MSEs of the simulation results

Validation	Input noise level	EM-based	Subspace	BELS
Self	10%	0.3513	0.6351	1.9536
Cross	10%	0.3346	0.5995	2.6545
Self	5%	0.2908	0.4571	1.0813
Cross	5%	0.2728	0.4305	1.0021

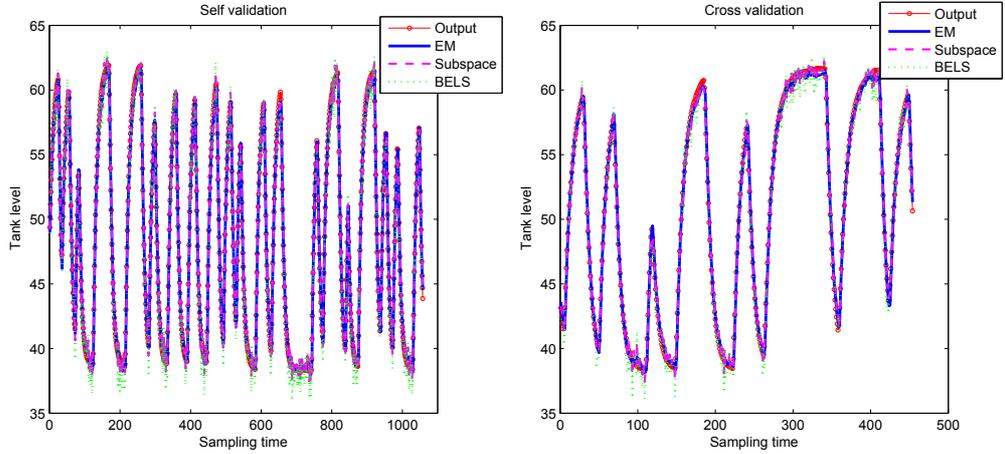


Figure 3.13: Validation results: 5% input noise

3.5 Conclusions

In this chapter, the EM algorithm was adopted in a ML framework for the identification of dynamic linear EIV state space model with a linear input dynamics. To take

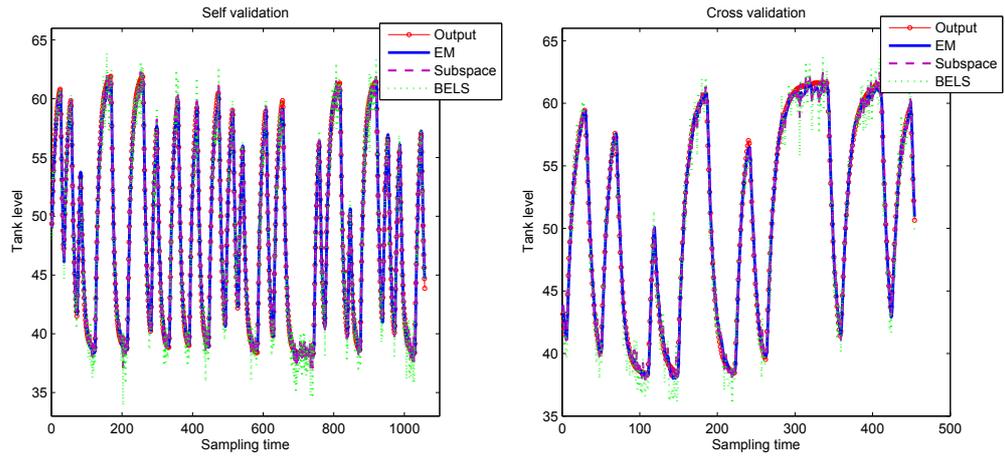


Figure 3.14: Validation results: 10% input noise

computation advantage, the proposed method avoids using the augmented model approach, and proposes a smoother for the state space models with colored inputs in the E-step of the EM algorithm.

To validate the proposed EM-based method, two simulation examples are used along with an experiment on the tank system. The results show that the EM-based method has achieved better performance than the existing subspace EIV method and the BELS method.

Chapter 4

Identification of LPV Errors-In-Variables systems with a dynamic process for noise-free inputs using EM algorithm

4.1 Introduction

In the previous chapter, the identification of linear dynamic errors-in-variables system is solved as a maximum likelihood (ML) estimation problem using EM algorithm. In this chapter, we would be extending the same approach for the identification of nonlinear EIV process.

Identification of nonlinear processes is always a challenging problem due to the difficulties posed by complex nonlinearities and stochastic nature of the disturbances. Intensive researches have been carried out in this topic for the past decades. There have been two major directions in the research of nonlinear system identification: (i) completely black box, data-driven, nonlinear model identification like NARX models, Hammerstein and Wiener models and models based on Artificial Neural Networks or combinations of those [31], [32]; (ii) grey box identification techniques, where the nonlinear first principles model structure, derived from physical principles such as mass and momentum balances, are retained and the parameters as well as noise characteristics are estimated from the process data [31], [33]. Both of these approaches broadly adopt Prediction Error Estimation (PEM) method, Nonlinear Least Square (NLS) method or MLE method in order to estimate the model parameters [34], [35],

[25]. Direct nonlinear black box identification is more computationally demanding in contrast to the grey box identification of nonlinear processes. However, grey box modeling requires more thorough understanding of the physical process. Other relevant methods include Just-in-time modeling (JIT) approaches where a number of local linear models are used to approximate the nonlinear process [36], [37]. However, the limitation here is that it is suitable only for static process as with mild nonlinearities.

To circumvent the problem, linear parameter varying (LPV) approaches have been proposed to identify nonlinear industrial processes using the cognizance that the industrial processes are designed to operate in certain structured ways, and it can be expressed as operating trajectory [38] where local models can be identified. In [38], a global LPV model for a class of nonlinear processes is proposed, and the local models are identified based on input excitation around certain chosen operating points, while cubic spline functions are employed to represent the validity of each local model. In [10], an LPV model identification method with a less error is proposed, and exponential functions are used to weight each local model, and showed a superior performance compared to [38]. The local model structure in [38], [10] is chosen as ARX model due to its capability in approximating any arbitrary linear dynamic system with reduced complexity [39]. Considering the local model structure as ARX model, extensions of LPV identification are also reported in literature, for example, LPV robust identification [40], stochastic scheduling variables problem in LPV identification [41], [42].

Comparing to ARX models, the state-space representation is a more general and flexible form for linear local input-output models. Further the Markov property of the state space model has advantages on estimation and control problems [43]. Among multiple model approaches with state space model structure, Markov switched state space models are discussed in [26], [44], [45] and [46], where local models' switching behavior is assumed to follow a Markov chain. LPV state space models are different compared to the former in that the local models are governed by operating points which can be referred from scheduling variables.

This work concerns with identification of LPV EIV systems. EIV systems refer to processes where both inputs and outputs are noise-corrupted. In practice, both the measured inputs and outputs can be noise-corrupted. Ignorance of noise corruption in the input can deteriorate the performance of traditional identification techniques.

Direct EIV identification of the process could pose computational difficulty and may lead to intractable problem due to the arbitrary structure of nonlinear functions. As a result, we propose to identify LPV EIV model to approximate the nonlinear EIV process. None of the previous works reported in literature has attempted to address this problem cohesively, and bridging of this gap forms the novelty of the present work.

In this chapter, an LPV EIV identification approach is developed to approximate model nonlinear EIV processes. The identification problem is treated as a MLE problem and the expectation maximization (EM) algorithm is employed to identify the model parameters. Necessary smoothers and filters are also derived to estimate the posterior probabilities of states and model identities as well as noise-free inputs. Two simulation examples and an experimental verification is presented to demonstrate the effectiveness of the proposed approach. The chapter is organized as follows: Section 4.2 presents the problem formulation. Section 4.3 provides the details of the LPV EIV identification using MLE method through the EM algorithm. Section 4.4 presents validation examples including experimental case study. Section 4.5 draws conclusive remarks from the study.

4.2 Problem formulation

A nonlinear dynamic EIV process in state-space structure can be represented as:

$$\begin{aligned}x_{k+1} &= f(x_k, \hat{u}_k) + w_k \\ y_k &= g(x_k, \hat{u}_k) + \tilde{y}_k \\ u_k &= \hat{u}_k + \tilde{u}_k\end{aligned}\tag{4.1}$$

where, $x_k \in \mathbb{R}^{n \times 1}$ is the hidden state; $\hat{u}_k \in \mathbb{R}^{m \times 1}$ is the unknown noise-free input; $u_k \in \mathbb{R}^{m \times 1}$ is the measurement of \hat{u}_k ; $w_k \in \mathbb{R}^{n \times 1}$ is the additive process noise in the state; $\tilde{u}_k \in \mathbb{R}^{m \times 1}$ and $\tilde{y}_k \in \mathbb{R}^{q \times 1}$ are the measurement noises of the input and the output, respectively; and $y_k \in \mathbb{R}^{q \times 1}$ is the output measurement; $g(\cdot)$ and $f(\cdot)$ are nonlinear functions for the state and output dynamics.

An LPV EIV model is employed to approximate the above nonlinear dynamic

EIV system (4.1), which in state-space form, is presented as follows:

$$\begin{aligned}
x_k &= A_{I_k} x_{k-1} + B_{I_k} \hat{u}_{k-1} + w_{k-1} \\
y_k &= C_{I_k} x_k + D_{I_k} \hat{u}_k + \tilde{y}_k \\
u_k &= \hat{u}_k + \tilde{u}_k
\end{aligned} \tag{4.2}$$

where, $I_k \in \{1, 2, \dots, M\}$ are the local model identities, determined by the scheduling variable L_k ; the state space matrices $\{A_i, B_i, C_i, D_i\}$ $i = 1, \dots, M$ represent the parameters for each local linear model developed along with the operating trajectory. A combination of estimations given by the local models is used to approximate the output in different operating conditions, and a normalized exponential function is employed as the local model weight [10]:

$$\begin{aligned}
P(I_k = i | L_k, T_{1:M}, o_{1:M}) &= \frac{w_{ki}}{\sum_{j=1}^M w_{kj}} \\
w_{ki} &= \exp\left(\frac{-(L_k - T_i)^2}{2o_i^2}\right)
\end{aligned} \tag{4.3}$$

where, $T_{1:M} = \{T_1, T_2, \dots, T_M\}$ are the M different operating points; $o_{1:M} = \{o_1, o_2, \dots, o_M\}$ represent the validity width of the different local models. The probability of output y_k , given all the past input-output data and the current scheduling data, can be derived as:

$$\begin{aligned}
P(y_k | y_{1:k-1}, \hat{u}_{1:k-1}, L_k, \Theta) &= \sum_{i=1}^M P(I_k = i, y_k | y_{1:k-1}, \hat{u}_{1:k-1}, L_k, \Theta) \\
&= \sum_{i=1}^M P(I_k = i | L_k, T_{1:M}, o_{1:M}) P(y_k | y_{1:k-1}, \hat{u}_{1:k-1}, I_k = i, \Theta)
\end{aligned} \tag{4.4}$$

where, Θ represents the collection of all the parameters to be estimated.

We assume a parametric dynamic model for the noise-free input \hat{u}_k in the state-space form, given by:

$$\begin{aligned}
z_k &= f_o(z_{k-1}, u_{k-1}^o) + w_{k-1}^o \\
\hat{u}_k &= g_o(z_k, u_k^o)
\end{aligned} \tag{4.5}$$

where, $z_k \in \mathbb{R}^{n_o \times 1}$ is the hidden state of the input process and $u_k^o \in \mathbb{R}^{m_o \times 1}$ is the known source of input; $w_k^o \in \mathbb{R}^{n_o \times 1}$ is the process noise of the input process. The measurement noises \tilde{u}_k, \tilde{y}_k , and the process noises w_k, w_k^o are assumed to follow i.i.d Gaussian distributions with zero mean and unknown covariance parameters $\Sigma_{\tilde{u}}, \Sigma_{\tilde{y}}$,

Σ_w, Σ_{w^o} , respectively, and the noise-free input \hat{u}_k is assumed to be uncorrelated with the noises. Further, the four white-noise sequences $\tilde{u}, \tilde{y}, w, w^o$ are assumed to be uncorrelated, that is:

$$E \left[\begin{pmatrix} \tilde{u}_k \\ \tilde{y}_k \\ w_k \\ w_k^o \end{pmatrix} \begin{pmatrix} \tilde{u}_i^T & \tilde{y}_i^T & w_i^T & w_i^{oT} \end{pmatrix} \right] = \begin{bmatrix} \Sigma_{\tilde{u}} & 0 & 0 & 0 \\ 0 & \Sigma_{\tilde{y}} & 0 & 0 \\ 0 & 0 & \Sigma_w & 0 \\ 0 & 0 & 0 & \Sigma_{w^o} \end{bmatrix} \cdot \delta_{ki} \geq 0 \quad (4.6)$$

where, δ_{ki} refers to the Kronecker's delta.

The LPV EIV model identification task is to estimate model parameters Θ , which includes the state-space matrices $\{A_i, B_i, C_i, D_i\}$ $i = 1, \dots, M$, distribution parameters for noises w, \tilde{u}, \tilde{y} and parameters for the validity width of the local models o_i $i = 1, \dots, M$, from the input and output data set $\{y_{1:N}, u_{1:N}\}$. For simplicity, in the present work, it is assumed that nonlinear functions $f_o(\cdot)$ and $g_o(\cdot)$ in Equation (4.5) are known.

To identify the LPV EIV model parameters, the maximum likelihood (ML) method is applied in order to leverage its good estimation accuracy. To tackle the hidden variables, the expectation maximization (EM) algorithm is used to compute the maximum likelihood estimation for the parameters of the LPV EIV model.

4.3 The LPV EIV system identification

In this section, the application of the EM algorithm to identify the LPV EIV model parameters is presented as the main contribution of this study. Before presenting the derivations for parameter estimation of the LPV EIV system using the EM algorithm, a brief revisit of the EM algorithm is given.

4.3.1 Revisit of the EM algorithm

The EM algorithm provides an iterative procedure for the ML estimation of both parameters and unknown variables. It is a two-step iterative algorithm containing expectation (E) and maximization (M) steps which are repeated till convergence. In the E-step, the conditional expectation of the complete-data likelihood function, called Q function, is calculated, where the hidden variables are estimated based on

the old parameters:

$$Q(\Theta|\Theta^{old}) = E_{C_{mis}|\Theta^{old}, C_{obs}} \log L(C_{obs}, C_{mis}|\Theta) \quad (4.7)$$

where Θ is the new set of parameters; Θ^{old} is the old set of parameters; C_{mis} refers to the missing data or hidden variables; C_{obs} refers to the observations; $L(\cdot)$ is the complete-data likelihood.

In the M-step, the new parameters Θ^{new} are computed by maximizing the Q function as follows:

$$\Theta^{new} = \arg \max_{\Theta} Q(\Theta|\Theta^{old}) \quad (4.8)$$

4.3.2 ML estimation of the LPV EIV model using the EM algorithm

The Q function for parameter estimations of the LPV EIV state space model (4.2) is derived as:

$$Q = E_{x_{1:N}, \hat{u}_{1:N}, I_{1:N} | y_{1:N}, u_{1:N}, u_{1:N}^o, T_{1:M}^o, L_{1:N}, \Theta^{old}, \Theta_o} \{ \log P(y_{1:N}, u_{1:N}, u_{1:N}^o, T_{1:M}^o, L_{1:N}, x_{1:N}, \hat{u}_{1:N}, I_{1:N} | \Theta, \Theta_o) \} \quad (4.9)$$

where, $\{x_{1:N}, \hat{u}_{1:N}, I_{1:N}\}$ are the missing data denoted as C_{mis} ; $\{y_{1:N}, u_{1:N}, u_{1:N}^o, L_{1:N}\}$ are the observed data denoted as C_{obs} ; Θ includes the local models parameters $\{A_i, B_i, C_i, D_i\}$ for $i = 1, \dots, M$, noises variance parameters $\{\Sigma_{\tilde{y}}, \Sigma_{\tilde{u}}, \Sigma_w, \Sigma_{w^o}\}$ and the initial parameters of the hidden state $\{\mu_{x_1}, \Sigma_{x_1}\}$. The hidden variables $x_{1:N}$ are estimated from the historical data $\{y_{1:N}\}$ and the estimates of the noise-free input $\{\hat{u}_{1:N}\}$, using the parameters of the previous iterations Θ^{old} . The noise-free input $\{\hat{u}_{1:N}\}$ is estimated from the historical data $\{u_{1:N}, u_{1:N}^o\}$, and $\Theta_o \triangleq \{f_o, g_o, \Sigma_{\tilde{u}}, \Sigma_{w^o}\}$, for notational simplicity. For the sake of simplicity, in this study, we assume that Θ_o is known. In Θ_o is unknown, a separate identification procedure using data $\{u_{1:N}, u_{1:N}^o\}$ can be conducted to estimate Θ_o .

E-step

Using chain rule of probability, the complete-data log likelihood function is derived as:

$$\begin{aligned}
& \log L(x_{1:N}, \hat{u}_{1:N}, I_{1:N}, y_{1:N}, u_{1:N}, u_{1:N}^o, L_{1:N}, T_{1:M}^o | \Theta) \\
= & \log \prod_{k=2}^N P(y_k | x_k, \hat{u}_k, I_k, \Theta) P(x_k | x_{k-1}, \hat{u}_{k-1}, I_k, \Theta) P(I_k | L_k, T_{1:M}^o, \Theta) P(y_1 | x_1, \hat{u}_1, I_1, \Theta) \cdot \\
& P(x_1 | \Theta) P(I_1 | L_1, T_{1:M}^o, \Theta) \underbrace{\prod_{k=1}^N P(u_k | u_k^o, \hat{u}_k) P(u_k^o) P(L_k) \prod_{k=2}^N P(\hat{u}_k | \hat{u}_{k-1}, u_{k-1}^o) P(\hat{u}_1) P(T_{1:M}^o)}_C \\
= & \sum_{k=1}^N \log P(y_k | x_k, \hat{u}_k, I_k, \Theta) + \sum_{k=2}^N \log P(x_k | x_{k-1}, \hat{u}_{k-1}, I_k, \Theta) + \\
& \sum_{k=1}^N \log P(I_k | L_k, T_{1:M}^o, \Theta) + \log P(x_1 | \Theta) + \log C
\end{aligned} \tag{4.10}$$

where, C is the term which is independent of Θ and plays no role in the maximization step, hence the subsequent steps will dispense with the term. Based on the Gaussian assumptions of the noises, the individual likelihood terms in equation (4.10) can be derived as:

$$\begin{aligned}
P(y_k | x_k, \hat{u}_k, I_k, \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{\tilde{y}})}} \exp\left\{-\frac{(y_k - C_{I_k} x_k - D_{I_k} \hat{u}_k)^T \Sigma_{\tilde{y}}^{-1} (y_k - C_{I_k} x_k - D_{I_k} \hat{u}_k)}{2}\right\} \\
P(x_{k+1} | x_k, \hat{u}_k, I_k, \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{\tilde{w}})}} \exp\left\{-\frac{(x_{k+1} - A_{I_k} x_k - B_{I_k} \hat{u}_k)^T \Sigma_{\tilde{w}}^{-1} (x_{k+1} - A_{I_k} x_k - B_{I_k} \hat{u}_k)}{2}\right\} \\
P(x_1 | \Theta) &= \frac{1}{\sqrt{2\pi \det(\Sigma_{x_1})}} \exp\left\{-\frac{(x_1 - \mu_{x_1})^T \Sigma_{x_1}^{-1} (x_1 - \mu_{x_1})}{2}\right\}
\end{aligned} \tag{4.11}$$

Therefore, the Q function (4.9) is represented as:

$$\begin{aligned}
Q = & -\frac{1}{2} \sum_{k=1}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \cdot \\
& \int_{x_k} P(x_k | I_k = i, C_{obs}, \Theta^{old}) (y_k - C_{I_k} x_k - D_{I_k} \hat{u}_k^l)^T \Sigma_{\bar{y}}^{-1} (y_k - C_{I_k} x_k - \\
& D_{I_k} \hat{u}_k^l) dx_k - \frac{N}{2} \log 2\pi \det(\Sigma_{\bar{y}}) - \frac{1}{2} \sum_{k=2}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) \cdot \\
& P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \cdot \int_{x_{k-1:k}} P(x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}) \cdot \\
& (x_{k+1} - A_{I_k} x_k - B_{I_k} \hat{u}_k)^T \Sigma_{\bar{w}}^{-1} (x_{k+1} - A_{I_k} x_k - B_{I_k} \hat{u}_k) dx_k dx_{k-1} - \\
& \frac{N-1}{2} \cdot \log 2\pi \det(\Sigma_w) + \sum_{k=1}^N \sum_{i=1}^M P(I_k = i | C_{obs}, \Theta^{old}) \log P(I_k | L_k, T_{1:M}^o, \Theta) \\
& - \frac{1}{2} \sum_{i=1}^M P(I_1 = i | C_{obs}, \Theta^{old}) \int_{x_1} P(x_1 | I_1 = i, C_{obs}, \Theta^{old}) \cdot \\
& (x_1 - \mu_{x_1})^T \Sigma_{x_1}^{-1} (x_1 - \mu_{x_1}) dx_1 - \frac{1}{2} \log 2\pi \det(\Sigma_{x_1})
\end{aligned} \tag{4.12}$$

where, the particle filter and smoother are applied to obtain the estimates of the noise-free input $\hat{u}_{1:N}$ through the input-output data $\{u_{1:N}, u_{1:N}^o\}$. The smoother density function $P(\hat{u}_k | u_{1:N}, u_{1:N}^o)$ can be numerically calculated as [25]:

$$P(\hat{u}_k | u_{1:N}, u_{1:N}^o) = \sum_{l=1}^L w_{k|N}^l \delta(\hat{u}_k - \hat{u}_k^l) \tag{4.13}$$

where $w_{k|N}$ is the smoothed weight of the particle \hat{u}_k^l , computed as $P(\hat{u}_k = \hat{u}_k^l | u_{1:N}, u_{1:N}^o)$, and L is the number of the particles.

M-step

In M-step, the Q function (4.12) is maximized to yield the parameter updating expressions. Unconstrained optimization provides parameter update expressions for parameters $\{A_{1:M}, B_{1:M}, C_{1:M}, D_{1:M}, \Sigma_w, \Sigma_{\bar{y}}, \mu_{x_1}, \sigma_{x_1}^2\}$, where derivatives are taken directly. Explicit update expression for local model width o_i is difficult to obtain in maximization due to the presence of exponential functions and physical constraints. To deal with such a case, we do a constrained nonlinear optimization to obtain the optimal validity width of each local model.

Next, we take the derivatives of Q function with respect to parameters,

$$\frac{dQ}{d\bar{A}_i} = d \left\{ -\frac{1}{2} \sum_{k=2}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_w^{-1} (x_k - \bar{A}_i \bar{x}_{k-1})(x_k - \bar{A}_i \bar{x}_{k-1})^T] \} \right\} / d\bar{A}_i \quad (4.14)$$

$$\frac{dQ}{d\bar{C}_i} = d \left\{ -\frac{1}{2} \sum_{k=1}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_k | I_k = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_{\bar{y}}^{-1} (y_k - \bar{C}_i \bar{x}_k)(y_k - \bar{C}_i \bar{x}_k)^T] \} \right\} / d\bar{C}_i \quad (4.15)$$

$$\frac{dQ}{d\Sigma_w} = d \left\{ -\frac{N-1}{2} \log 2\pi \det \Sigma_w - \frac{1}{2} \sum_{k=2}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) \right. \\ \left. P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_w^{-1} (x_k - \bar{A}_i \bar{x}_{k-1})(x_k - \bar{A}_i \bar{x}_{k-1})^T] \} \right\} / d\Sigma_w \quad (4.16)$$

$$\frac{dQ}{d\Sigma_{\bar{y}}} = d \left\{ -\frac{N}{2} \log 2\pi \det \Sigma_{\bar{y}} - \frac{1}{2} \sum_{k=1}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_k | I_k = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_{\bar{y}}^{-1} (x_k - \bar{C}_i \bar{x}_k)(x_k - \bar{C}_i \bar{x}_k)^T] \} \right\} / d\Sigma_{\bar{y}} \quad (4.17)$$

$$\frac{dQ}{d\mu_{x_1}} = d \left\{ -\frac{1}{2} \sum_{i=1}^M P(I_1 = i | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_1 | I_1 = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_{x_1}^{-1} (x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T] \} \right\} / d\mu_{x_1} \quad (4.18)$$

$$\frac{dQ}{d\Sigma_{x_1}} = d \left\{ -\frac{1}{2} \log 2\pi \det(\Sigma_{x_1}) - \frac{1}{2} \sum_{i=1}^M P(I_1 = i | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_1 | I_1 = i, C_{obs}, \Theta^{old}} \{ \text{Tr}[\Sigma_{x_1}^{-1} (x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T] \} \right\} / d\Sigma_{x_1} \quad (4.19)$$

where \bar{A}_i denotes $[A_i \ B_i]$; \bar{C}_i denotes $[C_i \ D_i]$; \bar{x}_k denotes $\begin{bmatrix} x_k \\ \hat{u}_k \end{bmatrix}$.

The parameter update expressions are obtained by equating the derivatives (4.14 - 4.19) to zero, as follows:

$$\bar{A}_i = \left[\sum_{k=2}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \right. \\ \left. E_{x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}} (x_k \bar{x}_{k-1}^T) \right] \cdot \left[\sum_{k=2}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) \right. \\ \left. P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \cdot E_{x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}} (\bar{x}_{k-1} \bar{x}_{k-1}^T) \right]^{-1} \quad (4.20)$$

$$\begin{aligned} \bar{C}_i &= \left[\sum_{k=1}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \cdot \right. \\ &\quad \left. E_{x_k | I_k=i, C_{obs}, \Theta^{old}}(y_k \bar{x}_k^T) \right] \cdot \left[\sum_{k=1}^N \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) \cdot \right. \\ &\quad \left. P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \cdot E_{x_k | I_k=i, C_{obs}, \Theta^{old}}(\bar{x}_k \bar{x}_k^T) \right]^{-1} \end{aligned} \quad (4.21)$$

$$\begin{aligned} \Sigma_w &= \left\{ \sum_{k=2}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_{k-1} = \hat{u}_{k-1}^l | C_{obs}, \Theta^{old}) \cdot \right. \\ &\quad \left. E_{x_k, x_{k-1} | I_k=i, C_{obs}, \Theta^{old}}[(x_k - \bar{A}_i \bar{x}_{k-1})(x_k - \bar{A}_i \bar{x}_{k-1})^T] \right\} / (N-1) \end{aligned} \quad (4.22)$$

$$\begin{aligned} \Sigma_{\bar{y}} &= \left\{ \sum_{k=1}^N \sum_{i=1}^M \sum_{l=1}^L P(I_k = i | C_{obs}, \Theta^{old}) P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}) \cdot \right. \\ &\quad \left. E_{x_k | I_k=i, C_{obs}, \Theta^{old}}[(x_k - \bar{C}_i \bar{x}_k)(x_k - \bar{C}_i \bar{x}_k)^T] \right\} / N \end{aligned} \quad (4.23)$$

$$\mu_{x_1} = \sum_{i=1}^M P(I_1 = i | C_{obs}, \Theta^{old}) E_{x_1 | I_1=i, C_{obs}, \Theta^{old}}(x_1) \quad (4.24)$$

$$\Sigma_{x_1} = \sum_{i=1}^M P(I_1 = i | C_{obs}, \Theta^{old}) E_{x_1 | I_1=i, C_{obs}, \Theta^{old}}[(x_1 - \mu_{x_1})(x_1 - \mu_{x_1})^T] \quad (4.25)$$

Further, the optimization formulation for obtaining the optimal o_i can be posed as:

$$\begin{aligned} &\max_{o_{1:M}} \sum_{k=1}^N \sum_{i=1}^M \log P(I_k = i | L_k, T_{1:M}, o_{1:M}) \cdot P(I_k = i | C_{obs}, \Theta^{old}) \\ &S.T. \quad o_{i,ld} \leq o_i \leq o_{i,ub}, \quad \forall i \in \{1, 2, \dots, M\} \end{aligned} \quad (4.26)$$

where, $o_{i,ld}$ and $o_{i,ub}$ represent lower and upper bounds of the variable o_i .

In this work, the above problem (4.26) is solved using the constrained nonlinear optimization function ‘fmincon’ provided in MATLAB[10][41].

4.3.3 Smoother for the LPV state space model

In the M-step, the following posterior probabilities are required to calculate the conditional expectation terms:

$$\begin{aligned} &P(I_k = i | C_{obs}, \Theta^{old}), \quad \forall k = 1, 2, \dots, N; i = 1, 2, \dots, M \\ &P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old}), \quad \forall k = 1, 2, \dots, N; l = 1, 2, \dots, L \\ &P(X_k, X_{k-1} | I_k = i, C_{obs}, \Theta^{old}), \quad \forall k = 2, 3, \dots, N; i = 1, 2, \dots, M \\ &P(X_k | I_k = i, C_{obs}, \Theta^{old}), \quad \forall k = 1, 2, \dots, N; i = 1, 2, \dots, M \end{aligned} \quad (4.27)$$

Among them, the posterior probability $P(\hat{u}_k = \hat{u}_k^l | C_{obs}, \Theta^{old})$ and the generation of the particles \hat{u}_k^l are computed using the particle smoother [25], [47], while the rest of the terms are computed by deriving a smoother of the LPV state space model as follows:

$$\begin{aligned}
P(x_k | I_k = i, C_{obs}, \Theta^{old}) &\sim \mathcal{N}(x_{k|N}^i, P_{k|N}^i) \\
P(x_{k-1} | I_k = i, C_{obs}, \Theta^{old}) &\sim \mathcal{N}(x_{k-1|N}^{(*,i)}, P_{k-1|N}^{(*,i)}) \\
P(x_k, x_{k-1} | I_k = i, C_{obs}, \Theta^{old}) &\sim \mathcal{N} \left[\begin{pmatrix} x_{k|N}^i \\ x_{k-1|N}^{(*,i)} \end{pmatrix}, \begin{pmatrix} P_{k|N}^i & M_{k|N}^i \\ M_{k|N}^{iT} & P_{k-1|N}^{(*,i)} \end{pmatrix} \right]
\end{aligned} \tag{4.28}$$

where, $x_{k|N}^i$ and $P_{k|N}^i$ are the mean and the variance for the smoother of x_k given model identity $I_k = i$; $x_{k-1|N}^{(*,i)}$ and $P_{k-1|N}^{(*,i)}$ are the mean and the variance for the smoother of x_{k-1} given model identity $I_k = i$; and $M_{k|N}^i$ is the lag-one covariance smoother given $I_k = i$.

In this section, the estimation of states is carried out using output data $y_{1:N}$ and the particle filter is used for the point estimation of the inputs $\hat{u}_{1:N}$ and is represented as $\hat{u}_{1:N}^p$:

$$\hat{u}_k^p = \sum_{l=1}^L \hat{u}_k^l / L \tag{4.29}$$

where, \hat{u}_k^l is the re-sampled particle with a weight of $1/L$.

The posterior probabilities are computed based on the Gaussian noise assumption and are presented as:

$$\begin{aligned}
P(x_k | \phi_N, I_k = j) &\sim \mathcal{N}(x_{k|N}^j, P_{k|N}^j) \\
P(x_k | \phi_N, I_{k+1} = h) &\sim \mathcal{N}(x_{k|N}^{(*,h)}, P_{k|N}^{(*,h)}) \\
P(x_k | \phi_N, I_k = j, I_{k+1} = h) &\sim \mathcal{N}(x_{k|N}^{(*,j,h)}, P_{k|N}^{(*,j,h)})
\end{aligned} \tag{4.30}$$

where $\phi_N \triangleq \{y_{1:N}, \hat{u}_{1:N}^p\}$; $x_{k|N}^{(*,j,h)}$ and $P_{k|N}^{(*,j,h)}$, the smoothed estimate and variance of the state, respectively, are derived as next.

Smoothing step

The smoothers for X_k given different state identities are derived using the expressions for the conditional distribution of a subset of Gaussian random variables given another

subset of Gaussian random variables [28], as follows:

$$\begin{aligned}
x_{k|N}^{(*,j,h)} &\triangleq E(X_k|\phi_N, I_k = j, I_{k+1} = h) \\
&= E[E(X_k|X_{k+1}, \phi_k, I_k = j, I_{k+1} = h)|\phi_N, I_k = j, I_{k+1} = h] \\
&= E[x_{k|k}^j + J_k^{(*,j,h)}(X_{k+1} - x_{k+1|k}^{(j,h)})|\phi_N, I_k = j, I_{k+1} = h] \\
&= x_{k|k}^j + J_k^{(*,j,h)} \underbrace{[E(X_{k+1}|\phi_N, I_k = j, I_{k+1} = h) - x_{k+1|k}^{(j,h)}]}_{x_{k+1|N}^{(j,h,*)}}
\end{aligned} \tag{4.31}$$

$$\begin{aligned}
P_{k|N}^{(*,j,h)} &\triangleq V(X_k|\phi_N, I_k = j, I_{k+1} = h) \\
&= E[V(X_k|X_{k+1}, \phi_k, I_k = j, I_{k+1} = h)|\phi_N, I_k = j, I_{k+1} = h] + \\
&\quad V[E(X_k|X_{k+1}, \phi_k, I_k = j, I_{k+1} = h)|\phi_N, I_k = j, I_{k+1} = h] \\
&= E(P_{k|k}^j - J_k^{(*,j,h)} P_{k+1|k}^{(j,h)} J_k^{(*,j,h)T}|\phi_N, I_k = j, I_{k+1} = h) + \\
&\quad V[x_{k|k}^j + J_k^{(*,j,h)}(X_{k+1} - x_{k+1|k}^{(j,h)})|\phi_N, I_k = j, I_{k+1} = h] \\
&= P_{k|k}^j + J_k^{(*,j,h)} \underbrace{(V(X_{k+1}|\phi_N, I_k = j, I_{k+1} = h) - P_{k+1|k}^{(j,h)})}_{P_{k+1|N}^{(j,h,*)}} J_k^{(*,j,h)T}
\end{aligned} \tag{4.32}$$

where, $E(\cdot)$ denotes the expectation operation; $V(\cdot)$ denotes the variance operation; the upper case X_k denotes the variable for the state, x_k is the corresponding realization, and similarly for other variables like Y_k ; $x_{k|k}^j$, $P_{k|k}^j$, $x_{k+1|k}^{(j,h)}$ and $P_{k+1|k}^{(j,h)}$ are the filtered and predicted states, variances of the state X_k , respectively, derived in the next sub-section. The smoother gain $J_k^{(*,j,h)}$ is calculated as:

$$\begin{aligned}
J_k^{(*,j,h)} &= Cov[X_k, X_{k+1}|\phi_k, I_k = j, I_{k+1} = h] \cdot \\
&\quad [V(X_{k+1}|\phi_k, I_k = j, I_{k+1} = h)]^{-1} \\
&= P_{k|k}^j A_h^T (P_{k+1|k}^{(j,h)})^{-1}
\end{aligned} \tag{4.33}$$

To keep the backwards calculation running, following approximations are adopted in equation (4.31) and (4.32) as:

$$\begin{aligned}
x_{k+1|N}^{(j,h,*)} &\approx x_{k+1|N}^h = E(X_{k+1}|\phi_N, I_{k+1} = h) \\
P_{k+1|N}^{(j,h,*)} &\approx P_{k+1|N}^h = V(X_{k+1}|\phi_N, I_{k+1} = h)
\end{aligned} \tag{4.34}$$

Thus, the expression for smoother of X_k when $I_k = j$, $I_{k+1} = h$, is as follows:

$$\begin{aligned}
x_{k|N}^{(*,j,h)} &= x_{k|k}^j + J_k^{(*,j,h)} [x_{k+1|N}^h - x_{k+1|k}^{(j,h)}] \\
P_{k|N}^{(*,j,h)} &= P_{k|k}^j + J_k^{(*,j,h)} (P_{k+1|N}^h - P_{k+1|k}^{(j,h)}) J_k^{(*,j,h)T}
\end{aligned} \tag{4.35}$$

where, $x_{k+1|N}^h$ and $P_{k+1|N}^h$ are evaluated later using the principles of distribution collapsing [48].

Prediction and filtering steps

The filtered and the predicted probabilities, which are required in the smoothing step are given as:

$$\begin{aligned}
P(x_k|\phi_k, I_k = j) &\sim \mathcal{N}(x_{k|k}^j, P_{k|k}^j) \\
P(x_k|\phi_k, I_k = j, I_{k-1} = i) &\sim \mathcal{N}(x_{k|k}^{(i,j)}, P_{k|k}^{(i,j)}) \\
P(x_k|\phi_{k-1}, I_k = j, I_{k-1} = i) &\sim \mathcal{N}(x_{k|k-1}^{(i,j)}, P_{k|k-1}^{(i,j)})
\end{aligned} \tag{4.36}$$

For the prediction step, the expressions are as follows:

$$\begin{aligned}
x_{k|k-1}^{(i,j)} &\triangleq E(X_k|\phi_{k-1}, I_k = j, I_{k-1} = i) \\
&= A_j x_{k-1|k-1}^i + B_j \hat{u}_{k-1}^p \\
P_{k|k-1}^{(i,j)} &\triangleq V(X_k|\phi_{k-1}, I_k = j, I_{k-1} = i) \\
&= A_j P_{k-1|k-1}^i A_j^T + Q
\end{aligned} \tag{4.37}$$

where, the filter parameters $x_{k-1|k-1}^i$ and $P_{k-1|k-1}^i$ can not be calculated directly using the filtering step based on the above predictions. They are derived in the collapsing step instead and explained in the next subsection.

The filtering step for $x_{k|k}^{(i,j)}$ and $P_{k|k}^{(i,j)}$ is derived based on the results of the prediction step, as below:

$$\begin{aligned}
x_{k|k}^{(i,j)} &\triangleq E(X_k|\phi_k, I_k = j, I_{k-1} = i) \\
&= x_{k|k-1}^{(i,j)} + K_k^{(i,j)} (y_k - y_{k|k-1}^{(i,j)}) \\
P_{k|k}^{(i,j)} &\triangleq V(X_k|\phi_k, I_k = j, I_{k-1} = i) \\
&= (I - K_k^{(i,j)} C_j) P_{k|k-1}^{(i,j)}
\end{aligned} \tag{4.38}$$

where, $x_{k|k}^{(i,j)}$ is the filtered estimate of X_k given model identities $I_k = j, I_{k-1} = j$, and $P_{k|k}^{(i,j)}$ is the corresponding variance. Further, $K_k^{(i,j)}$ is the filter gain, and $y_{k|k-1}^{(i,j)}$ is the predicted output, given by:

$$\begin{aligned}
y_{k|k-1}^{(i,j)} &\triangleq E(Y_k|\phi_{k-1}, I_k = j, I_{k-1} = i) \\
&= C_j x_{k|k-1}^{(i,j)} + D_j \hat{u}_k^p \\
K_k^{(i,j)} &= Cov[X_k, Y_k|\phi_k, I_k = j, I_{k-1} = i] \cdot \\
&\quad (V(Y_k|\phi_{k-1}, I_k = j, I_{k-1} = i))^{-1} \\
&= P_{k|k-1}^{(i,j)} C_j^T (C_j P_{k|k-1}^{(i,j)} C_j^T + R)^{-1}
\end{aligned} \tag{4.39}$$

A method for local model identity distribution approximation using distribution collapsing

To keep the predicting and the filtering steps working, we have assumed that both $P(x_k|\phi_k, I_k = j)$ and $P(x_k|\phi_k, I_k = j, I_{k-1} = i)$ are normally distributed. However it is contradictory that:

$$\begin{aligned} P(x_k|\phi_k, I_k = j) &= \sum_{i=1}^M P(x_k, I_{k-1} = i|\phi_k, I_k = j) \\ &= \sum_{i=1}^M P(I_{k-1} = i|\phi_k)P(x_k|I_{k-1} = i, I_k = j, \phi_k) \end{aligned} \quad (4.40)$$

where $P(I_{k-1} = i|\phi_k)$ is the weight of $P(x_k|\phi_k, I_k = j, I_{k-1} = i)$. This indicates that $P(x_k|\phi_k, I_k = j)$ is a Gaussian mixture distribution under the assumption that $P(x_k|\phi_k, I_k = j, I_{k-1} = i)$ is Gaussian distributed. This property of the multiple state space model makes the number of the Gaussian mode posteriors which are to be computed, to increase exponentially with M^2 . In this work, to reduce the number of calculated posteriors from M^2 to M , we propose to approximate the Gaussian mixture by distribution collapsing method proposed by [48], as follows,

$$\begin{aligned} x_{k|k}^j &= \sum_{i=1}^M P(I_{k-1} = i|\phi_k)x_{k|k}^{(i,j)} \\ P_{k|k}^j &= \sum_{i=1}^M P(I_{k-1} = i|\phi_k)[P_{k|k}^{(i,j)} + (x_{k|k}^j - x_{k|k}^{(i,j)})(x_{k|k}^j - x_{k|k}^{(i,j)})^T] \end{aligned} \quad (4.41)$$

and applying the same 'collapsing' process for the smoothing steps $P(x_k|\phi_N, I_k = j)$, $P(x_k|\phi_N, I_{k+1} = h)$ and $P(x_k|\phi_N, I_k = j, I_{k+1} = h)$, leads to,

$$\begin{aligned} x_{k|N}^j &= \sum_{h=1}^M P(I_{k+1} = h|\phi_N)x_{k|N}^{(*,j,h)} \\ P_{k|N}^j &= \sum_{h=1}^M P(I_{k+1} = h|\phi_N)[P_{k|N}^{(*,j,h)} + (x_{k|N}^j - X_{k|N}^{(*,j,h)})(x_{k|N}^j - X_{k|N}^{(*,j,h)})^T] \end{aligned} \quad (4.42)$$

$$\begin{aligned} x_{k|N}^{(*,h)} &= \sum_{j=1}^M P(I_k = j|\phi_N)x_{k|N}^{(*,j,h)} \\ P_{k|N}^{(*,h)} &= \sum_{j=1}^M P(I_k = j|\phi_N)[P_{k|N}^{(*,j,h)} + (x_{k|N}^{(*,h)} - x_{k|N}^{(*,j,h)})(x_{k|N}^{(*,h)} - x_{k|N}^{(*,j,h)})^T] \end{aligned} \quad (4.43)$$

where, for computational tractability the posterior probability of the local model identity is approximated from smoothing to filtering as follows:

$$\begin{aligned}
P(I_{k-1} = i | \phi_k) &\approx P(I_{k-1} = i | \phi_{k-1}) \\
P(I_{k+1} = h | \phi_N) &\approx P(I_{k+1} = h | \phi_{k+1}) \\
P(I_k = j | \phi_N) &\approx P(I_k = j | \phi_k)
\end{aligned} \tag{4.44}$$

The filtered probability of I_k is derived as follows:

$$P(I_k = i | \phi_k) = \frac{P(y_k | \phi_{k-1}, I_k = i) P(I_k = i | \phi_{k-1})}{P(y_k | \phi_{k-1})} \tag{4.45}$$

where, $P(y_k | \phi_{k-1}, I_k = i) \sim \mathcal{N}(y_{k|k-1}^i, H_{k|k-1}^i)$; $P(I_k = i | \phi_{k-1}) = P(I_k = i | L_k, T_i)$.

$$\begin{aligned}
y_{k|k-1}^i &= E(C_i X_k + D_i \hat{u}_k + \tilde{y}_k | I_k = i, \phi_{k-1}) \\
&= C_i x_{k|k-1}^i + D_i \hat{u}_k^p \\
H_{k|k-1}^i &= V(C_i X_k + D_i \hat{u}_k + \tilde{y}_k | I_k = i, \phi_{k-1}) \\
&= C_i P_{k|k-1}^i C_i^T + \Sigma_{\tilde{y}}
\end{aligned} \tag{4.46}$$

and $x_{k|k-1}^i$ and $P_{k|k-1}^i$ are evaluated using the collapsing step from $x_{k|k-1}^{(l,i)}$ and $P_{k|k-1}^{(l,i)}$:

$$\begin{aligned}
x_{k|k-1}^i &= \sum_{l=1}^M P(I_{k-1} = l | \phi_{k-1}) x_{k|k-1}^{(l,i)} \\
P_{k|k-1}^i &= \sum_{l=1}^M P(I_{k-1} = l | \phi_{k-1}) [P_{k|k-1}^{(l,i)} + (x_{k|k-1}^i - x_{k|k-1}^{(l,i)})(x_{k|k-1}^i - x_{k|k-1}^{(l,i)})^T]
\end{aligned} \tag{4.47}$$

4.3.4 Computation of the conditional expectation terms in the Q function

In this section, the conditional expectation terms appearing in equation (4.20-4.23) are derived.

We start with the expression for $E(X_k X_k^T | C_{obs}, I_k = i)$. The smoother covariance of state X_k given local model identity $I_k = i$ is given as:

$$\begin{aligned}
P_{k|N}^i &= E[(X_k - x_{k|N}^i)(X_k - x_{k|N}^i)^T | C_{obs}, I_k = i] \\
&= E(X_k X_k^T | C_{obs}, I_k = i) - E(X_k x_{k|N}^i{}^T | C_{obs}, I_k = i) - \\
&\quad E(x_{k|N}^i X_k^T | C_{obs}, I_k = i) + x_{k|N}^i x_{k|N}^i{}^T \\
\Rightarrow E(X_k X_k^T | C_{obs}, I_k = i) &= x_{k|N}^i x_{k|N}^i{}^T + P_{k|N}^i
\end{aligned} \tag{4.48}$$

Following a similar approach as above, the conditional expectation term $E(X_{k-1}X_{k-1}^T|C_{obs}, I_k = i)$ is derived as:

$$E(X_{k-1}X_{k-1}^T|C_{obs}, I_k = i) = x_{k-1|N}^{(*,i)}x_{k-1|N}^{(*,i)T} + P_{k-1|N}^{(*,i)} \quad (4.49)$$

and the conditional expectation term $E(X_kX_{k-1}^T|C_{obs}, I_k = i)$ is also derived in a similar fashion as:

$$E(X_kX_{k-1}^T|C_{obs}, I_k = i) = x_{k|N}^i x_{k-1|N}^{(*,i)T} + M_{k|N}^i \quad (4.50)$$

where, $M_{k|N}^i$ is the lag-one covariance smoother given model identity $I_k = i$, defined as $M_{k|N}^i \triangleq E[(X_k - x_{k|N}^i)(X_{k-1} - x_{k-1|N}^{(*,i)})^T|C_{obs}, I_k = i]$. $M_{k|N}^i$ can not be derived directly due to the multi-mode nature of the LPV state space model. Hence, the lag-one covariance smoother given model identity $I_{k-1} = i, I_k = j$ and $I_{k+1} = h$, denoted as $M_{k|N}^{(i,j,h)}$, is derived first and detailed derivation is presented in Appendix C,

$$\begin{aligned} M_{k|N}^{(i,j,h)} &= E[(X_k - X_{k|N}^{(i,j,h)})(X_k - X_{k-1|N}^{(*,i,j,h)})^T] \\ &= (A_j + K_k^{(i,j)}C_jA_j + J_k^{(i,j,h)}A_hK_k^{(i,j)}C_jA_j)P_{k-1|k-1}^i + \\ &\quad J_k^{(i,j,h)}(M_{k+1|N}^{(j,h,*)} - A_hP_{k|k-1}^{(i,j)})J_{k-1}^{(*,i,j)T} \end{aligned} \quad (4.51)$$

where, $M_{k|N}^{(i,j,*)} = E[(X_k - X_{k|N}^{(i,j,*)})(X_{k-1} - X_{k-1|N}^{(*,i,j)})^T]$, and a similar collapsing step is adopted for the lag-one covariance smoothing as:

$$\begin{aligned} M_{k|N}^{(i,j,*)} &= \sum_{h=1}^M P(I_{k+1} = h|C_{obs})[M_{k|N}^{(i,j,h)} + (x_{k|N}^{(i,j,*)} - x_{k|N}^{(i,j,h)})(x_{k-1|N}^{(*,i,j)} - x_{k-1|N}^{(*,i,j,h)})^T] \\ M_{k|N}^j &= \sum_{i=1}^M P(I_{k-1} = i|C_{obs})[M_{k|N}^{(i,j,*)} + (x_{k|N}^j - x_{k|N}^{(i,j,*)})(x_{k-1|N}^{(*,j)} - x_{k-1|N}^{(*,i,j)})^T] \end{aligned} \quad (4.52)$$

where, $x_{k|N}^{(i,j,*)}$, $x_{k|N}^{(i,j,h)}$ and $x_{k-1|N}^{(*,i,j,h)}$ are the auxiliary state smoothers whose derivations are presented in Appendix C.

The initialization for the backwards calculation of the lag-one covariance smoother is also presented as follows:

$$\begin{aligned} M_{N|N}^{(i,j,*)} &= E[(X_N - X_{N|N}^{(i,j,*)})(X_{N-1} - X_{N-1|N}^{(*,i,j)})^T] \\ &= (A_j - K_N^{(i,j)}C_jA_j)P_{N-1|N-1}^i(I - J_{N-1}^{(i,j)}K_N^{(i,j)}C_jA_j)^T - \\ &\quad (I - K_N^{(i,j)}C_j)Q(J_{N-1}^{(i,j)}K_N^{(i,j)}C_j)^T + K_N^{(i,j)}R(J_{N-1}^{(i,j)}K_N^{(i,j)})^T \end{aligned} \quad (4.53)$$

4.3.5 Algorithm

The algorithmic representation of the complete steps of the ML estimation of LPV EIV system parameters using the EM algorithm is concisely presented as Algorithm 4 shown in the following page.

Algorithm 4: Proposed LPV EIV approach

Data: $y_{1:N}, u_{1:N}, u_{1:N}^o$

Result: Θ

Initialization :

(i) Estimate noise-free input $\hat{u}_{1:N}$ using the particle filter and smoother for the nonlinear input dynamics or Kalman filter and smoother for the linear input dynamics.

(ii) Obtain initial guess of Θ by conducting a number of initial trials with random values as initial guesses and selecting the best one as the final initial guess of parameters Θ .

while *compare the value of the new likelihood function based on the updated parameters with the one using the old parameters, till the desired convergence metric is reached*, **do**

E – step:

for $k = 2, 3, \dots, N$ **do**

 | Calculate the predicted state (4.37), the filtered state (4.38, 4.41) and the filtered model identities (4.47, 4.45).

end

 Calculate the initialization of smoother computation for $k = N$.

for $k = N - 1, 2, \dots, 1$ **do**

 | Calculate the smoothed state (4.31, 4.32, 4.42, 4.43) and the lag-one covariance smoother (4.51, 4.52).

end

 Calculate the conditional expectation quadratic terms for hidden variables using equations (4.48), (4.49), (4.50).

M – step:

 Calculate the updated model parameters Θ using equations (4.20), (4.21), (4.22), (4.23), (4.24) and (4.25).

end

4.4 Validations

In this section, two numerical examples including a first-order continuous LPV system and a CSTR process with a nonlinear dynamic model of the noise-free input are

depicted to illustrate the proposed method. Furthermore, an experimental verification of the multi-tank system is also presented to validate the proposed method.

4.4.1 Numerical example 1: a first-order continuous LPV system

A first-order continuous LPV model example [38] [40] is considered for the noise-free input-output process with the following transfer function form:

$$G(s, p) = \frac{K(p)}{\tau(p)s + 1} \quad (4.54)$$

where, the process time constant $\tau(p)$ and gain $K(p)$ change nonlinearly in operation range, $p \in [1, 4]$, as:

$$K(p) = 0.6 + p^2, \quad \tau(p) = 3 + 0.5p^3 \quad (4.55)$$

By transforming the process (4.54) into discrete state space model with sample time 1 and assuming state and input measurement uncertainties, it yields,

$$\begin{aligned} x_{k+1} &= A_d(p)x_k + B_d(p)\hat{u}_k + w_k \\ y_k &= C_d(p)x_k + v_k \\ u_k &= \hat{u}_k + \tilde{u}_k \end{aligned} \quad (4.56)$$

where w_k and v_k are the process noise and the measurement noise, respectively; w_k is designed as white noise with variance 0.02; v_k is set as white noise with variance of about 5% of the variance of the noise-free output ($\Sigma_v = 0.08$); \hat{u}_k is the unknown noise-free input, which is generated by a nonlinear process as follows:

$$\begin{aligned} z_{k+1} &= 0.5z_k + \frac{0.1z_k}{1 + z_k^2} + u_k^o + w_k^o \\ \hat{u}_k &= 0.05z_k^3 \end{aligned} \quad (4.57)$$

where u_k^o is the input source, and it is generated by a random binary signal with magnitude -1 and 1; z_k is the hidden state of the input process; the process noise w_k^o is set as Gaussian distribution with zero mean and variance 0.02; the input measurement noise \tilde{u}_k is white noise with variance of about 5% of the variance of the noise-free input \hat{u}_k ($\Sigma_{\tilde{u}} = 0.006$). The particle filter is applied for the state estimation of the nonlinear process, while three operating points for the numerical example are selected as $\{1, 2.25, 4\}$, and the corresponding operating trajectory P is presented in Figure

4.1 (2). In Figure 4.1 (1), particle filter state estimation results are presented, where the red circle line indicates noise corrupted input and blue solid line indicates particle filter estimation performance, from which it can be seen that the particle filter is able to estimate the input dynamics with reasonable accuracy.

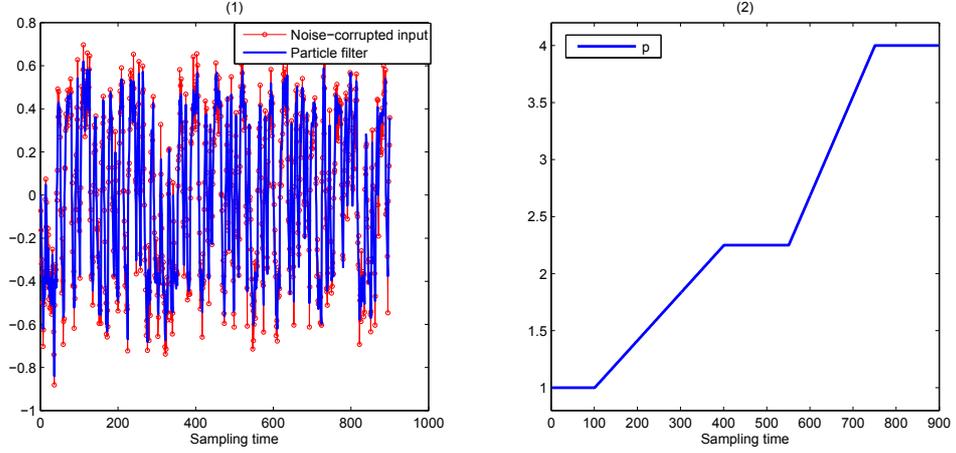


Figure 4.1: (1) The noise-corrupted input u_k and the PF estimations; (2) The operating trajectory for the numerical example 1

The self validation result of applying the proposed algorithm on the input-output data, are presented in Figure 4.2 (1), where the one-step ahead prediction (blue solid line) and the infinite step ahead prediction (green dash line) are compared to the plant output (red circle line) with mean square error (MSE) as 0.0739 and 0.4878, respectively, and the normalized weights of local models are also presented in Figure 4.2 (2). For cross validation, a new operating trajectory is used and the one-step ahead prediction and the infinite step ahead prediction are compared with the output measurement. The MSE for one-step ahead prediction is 0.0677 and MSE for infinite step ahead prediction is 0.1337. The results are presented in Figure 4.3, where blue solid line indicates one-step ahead prediction, green dash line presents the infinite step ahead prediction, and red circle line indicate the plant output measurement. This illustrates that the identified LPV EIV model is able to satisfactorily predict the nonlinear EIV process for the first order LPV system.

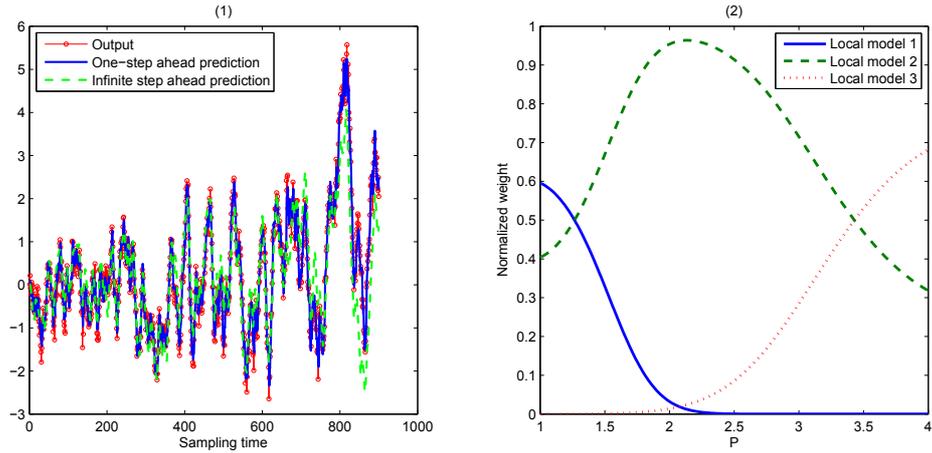


Figure 4.2: (1) The self validation result for the numerical example 1; (2) The model identity probabilities of the local models

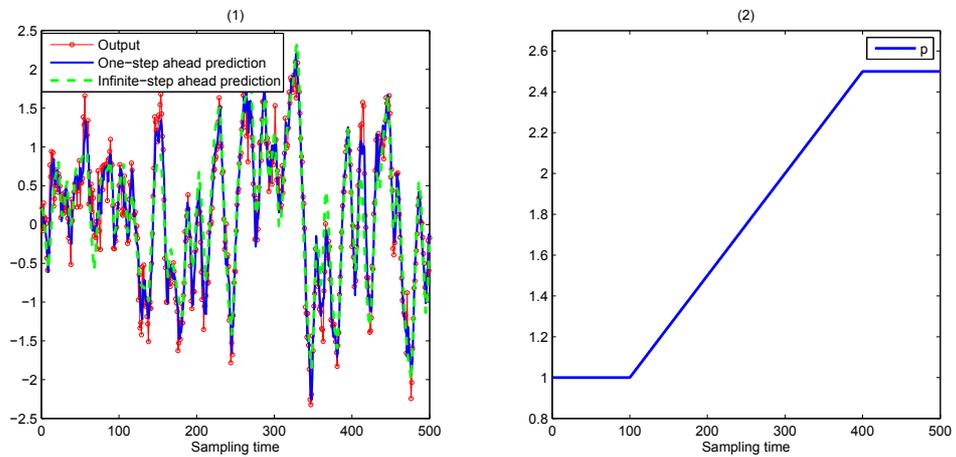


Figure 4.3: (1) The cross validation result for the numerical example 1; (2) The operation trajectory for the cross validation data

4.4.2 Numerical example 2: CSTR

In this sub-section, a continuous stirred tank reactor (CSTR) simulation example is considered to validate the proposed method. CSTR is a nonlinear chemical process, which has been commonly used for nonlinear system state estimation, modeling and control [10],[40],[49],[50] as a benchmark. The first principles model of CSTR is

derived based on the mass and heat balance, as [10]:

$$\begin{aligned}\frac{dC_A(t)}{dt} &= \frac{q(t)}{V}(C_{A0} - C_A(t)) - k_0 C_A(t) \exp\left(\frac{-E}{RT(t)}\right) \\ \frac{dT(t)}{dt} &= \frac{q(t)}{V}(T_0(t) - T(t)) - \frac{(-\Delta H)k_0 C_A(t)}{\rho C_p} \exp\left(\frac{-E}{RT(t)}\right) + \\ &\quad \frac{\rho_c C_{pc}}{\rho C_p V} q_c(t) \left\{ 1 - \exp\left(\frac{-hA}{q_c(t)\rho C_p}\right) \right\} (T_{c0}(t) - T(t))\end{aligned}\quad (4.58)$$

where the model parameters are listed in Table 4.1. The output variable is the product concentration $C_A(t)$, and the input variable is the coolant flow rate $q_c(t)$, which is also chosen as the scheduling variable since it influences the model significantly.

Table 4.1: Model parameters of the CSTR process

Parameters	Value	Unit
Process flow rate (q)	100	L/min
Feed concentration of component A (C_{A0})	1	mol/L
Feed temperature (T_0)	350	K
Inlet coolant temperature (T_{c0})	350	K
Reactor volume (V)	100	L
Heat transfer term (hA)	7×10^5	$cal/(min \cdot K)$
Specific heats (C_p, C_{pc})	1	$cal/(g \cdot K)$
Liquid density (ρ, ρ_c)	1×10^3	g/L
Reaction rate constant (k_0)	7.2×10^{10}	min^{-1}
Activation energy term (E/R)	1×10^4	K
Heat of reaction ($-\Delta H$)	-2×10^5	cal/mol

Three operating points of the coolant flow considered are [97, 103, 109], as the coolant flow rate varies from 97 L/min to 109 L/min . Since the coolant flow rate acts as both the input variable and the scheduling variable, its dynamics is assumed to be composed and divided into two parts: 1) is the scheduling variable which changes in a large scale that denotes the operating points; 2) is the noise-free input around the operating points, which is generated from the same model shown in (4.57). The source of input u_k^o is generated by a random binary sequence ranging between -1 and 1. The measurement u_k is corrupted by the input noise \tilde{u}_k which is the white noise with variance at 5% of the variance of noise-free input. Particle filter is applied to estimate the noise-free input on (4.57). The operating trajectory and the noise-corrupted input, and the particle filter results are presented in Figure 4.4 (1) and

Figure 4.4 (2), respectively, where the noise-corrupted input is represented by red circle line and particle filter estimation is by blue solid line. The simulated output measurement is corrupted by the white noise with variance at 5% of the variance of the noise-free output. The proposed algorithm is applied, and the self validation results are presented in Figure 4.5 (1), where the MSEs for one-step ahead prediction and infinite step ahead prediction are 6.623×10^{-7} and 1.8×10^{-6} , respectively. Corresponding model identity probability profiles are presented in Figure 4.5 (2). A new operating trajectory is used to generate the data for cross validation, and the results are presented in Figure 4.6 (2). In Figure 4.6 (1), the one-step ahead prediction (blue solid line) and the infinite step ahead prediction (green dash line) are compared with the plant output measurement (red circle line), for which the MSEs are 8.07×10^{-7} and 1.116×10^{-6} , respectively, thereby showing the effectiveness of the proposed identification.

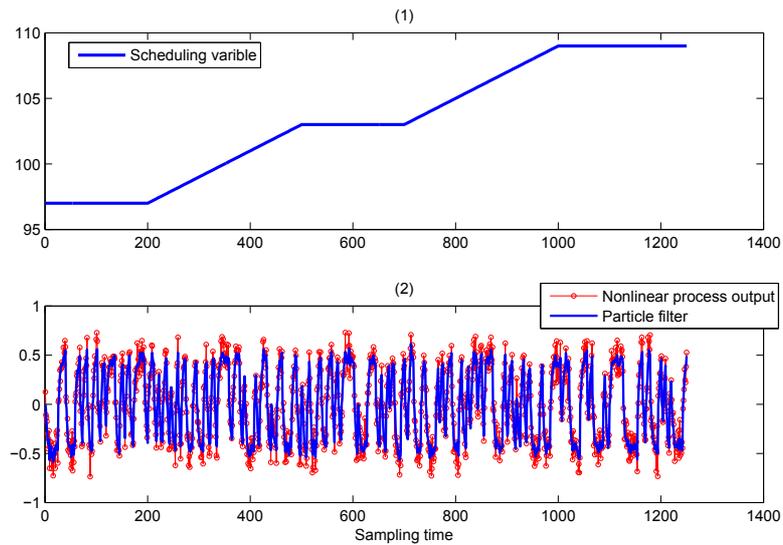


Figure 4.4: (1) The operating trajectory for the numerical example 2; (2) The PF estimation of the nonlinear process output

4.4.3 Experimental example

In this sub-section, an experiment on the multi-tank system is conducted to validate the proposed method. The schematic diagram of the multi-tank system is presented in Figure 4.7, and the corresponding nonlinear model equations [51],[52],[42] are given

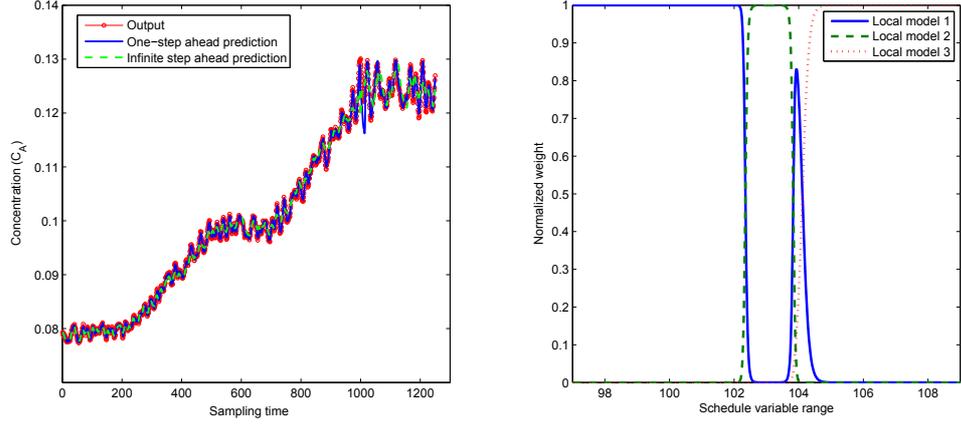


Figure 4.5: (1) The self validation results for the numerical example 2; (2) The model identity probabilities of the local models

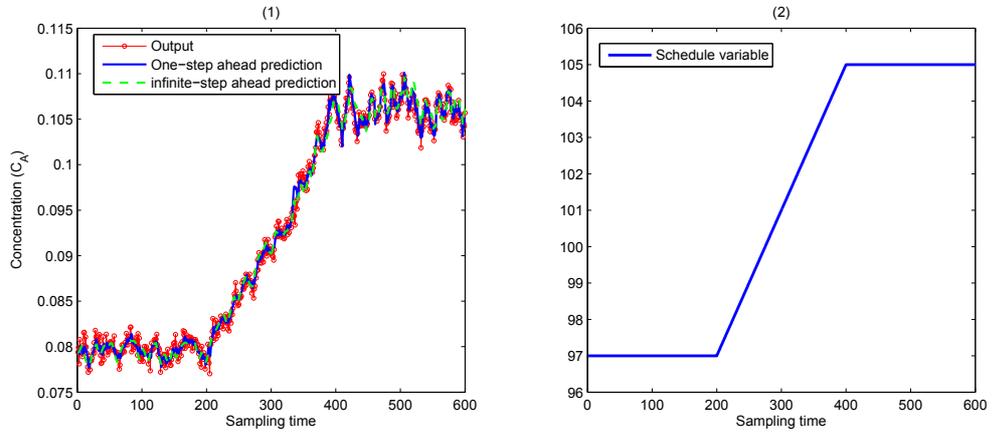


Figure 4.6: (1) The cross validation results for the numerical example 2; (2) The operating trajectory for the cross validation data

below:

$$\begin{aligned}
 \frac{dH_1}{dt} &= \frac{1}{\beta_1(H_1)}q - \frac{1}{\beta_1(H_1)}D_1H_1^{\alpha_1} \\
 \frac{dH_2}{dt} &= \frac{1}{\beta_2(H_2)}D_1H_1^{\alpha_1} - \frac{1}{\beta_2(H_2)}D_2H_2^{\alpha_2} \\
 \frac{dH_3}{dt} &= \frac{1}{\beta_3(H_3)}D_2H_2^{\alpha_2} - \frac{1}{\beta_3(H_3)}D_3H_3^{\alpha_3}
 \end{aligned} \tag{4.59}$$

where the notations are listed in Table 4.2, and $i \in \{1, 2, 3\}$.

In this work, we only use the middle tank as the EIV process, and flow valves have been adjusted in such a way that the inflow q is directly channeled into the middle

Table 4.2: Notations of multi-tank system

Notation	Description
q	the inflow to the upper tank
H_i	the water levels in the i th tank
D_i	the resistance of the output orifice of i th tank
β_i	the cross sectional area of the i th tank
α_i	the flow coefficient for the i th tank

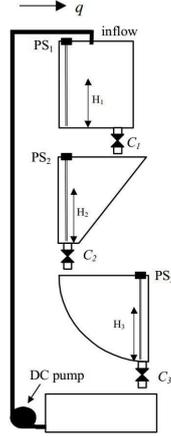


Figure 4.7: Diagram of multi-tank system

tank. Therefore, the nonlinear model for the EIV process considered is reduced to:

$$\frac{dH_2}{dt} = \frac{1}{\beta_2(H_2)}q - \frac{1}{\beta_2(H_2)}D_2H_2^{\alpha_2} \quad (4.60)$$

where, the EIV output variable is chosen as the middle tank water level H_2 , and the EIV input variable is chosen as the inflow q . The nonlinearity of the process is arisen from the shape of the tank where the cross sectional area changing along with the water level. By forcing the inflow q around a working point with certain small flow rate range, an approximated linear model can be built for the corresponding working point. Since the valve position can be used to change the working point for different water levels, the scheduling variable is chosen as the valve position of the middle tank C_2 . The operating points of the valve position are per-selected as $\{1, 0.85, 0.8\}$, and the operating trajectory is presented in Figure 4.8 (2).

To generate the input q , we considered the following nonlinear dynamics:

$$\begin{aligned} z_{k+1} &= 0.5z_k + \frac{0.1z_k}{1 + z_k^2} + u_k^o + w_k^o \\ \hat{u}_k &= 0.0025z_k^3 \end{aligned} \quad (4.61)$$

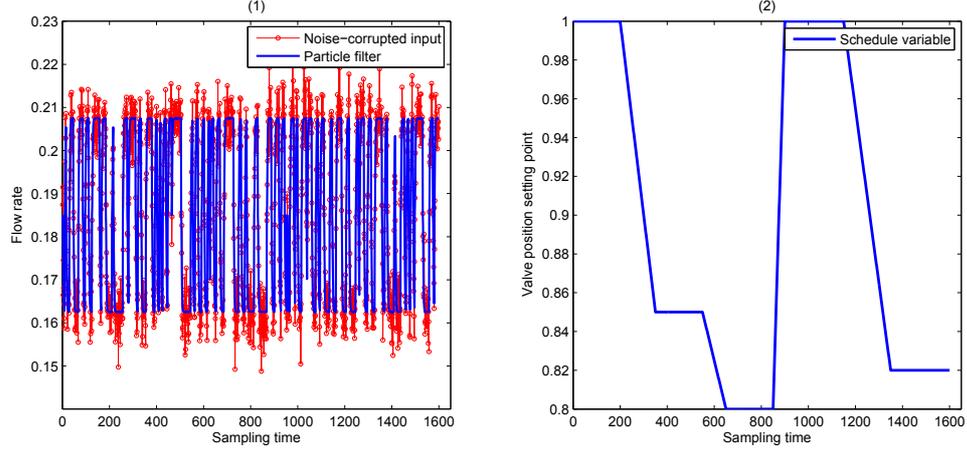


Figure 4.8: (1) The noise-corrupted input and the PF estimations; (2) The operation trajectory for the experimental example

where z_k is the hidden state for the input process; u_k^o is the source of input, which is a known random binary sequence; process noise w_k^o is white noise with variance 0.01. Simulated noise is added into the noise-free input \hat{u}_k with a noise variance at 5% of the input variance. Particle filter is used to estimate the noise-free input on (4.61), and the estimation result are presented in Figure 4.8 (1).

The self validation results and the normalized weights for local models are presented in Figure 4.9 (1) and Figure 4.9 (2), respectively. The one-step ahead prediction (blue solid line) and infinite step ahead prediction (green dash line) are compared with the plant output measurement with MSEs of 4.95×10^{-6} and 3.11×10^{-5} , respectively. The cross validation results are presented in Figure 4.10 (1), where one step ahead prediction (blue solid line) and infinite step ahead prediction (green dash line) are compared with the true plant output (red circle line), and the corresponding operating trajectory is shown in Figure 4.10 (2). Although one step ahead prediction shows good performance, the infinite step predictions have more significant mismatch at certain points. This is attributed to the possible valve stiction presented in this process, where the scheduling variable (valve opening) would not physically operate in the same trajectory as it is designed for the experiment.

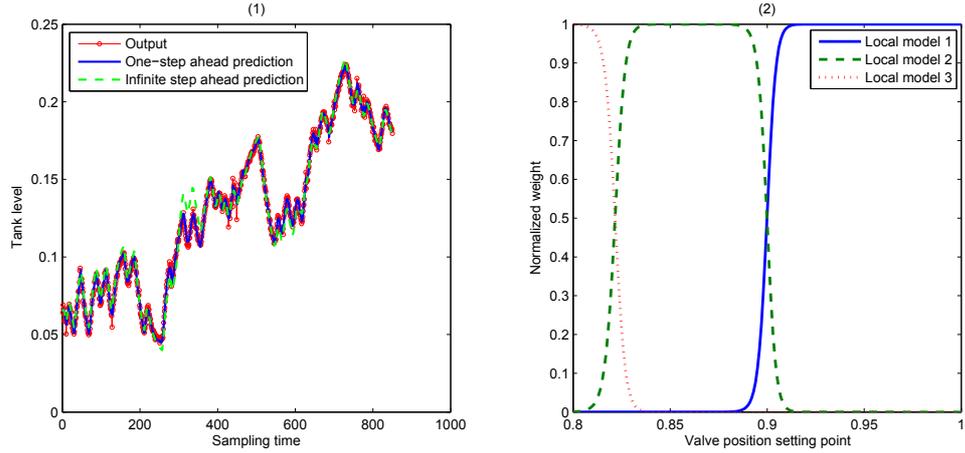


Figure 4.9: (1) The self validation results for the experimental example; (2) The model identity probabilities of the local models

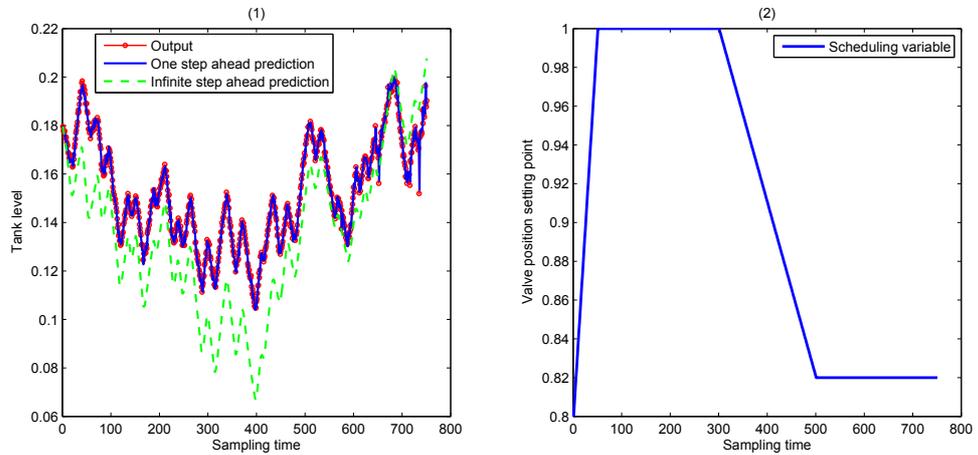


Figure 4.10: (1) The cross validation results for the experimental example; (2) The operating trajectory for the cross validation data

4.5 Conclusions

In this study, a multiple model approach is presented for the identification of nonlinear EIV system as an LPV EIV system. By selecting the scheduling variables and operating points based on the process information, the parameters of local state space models and the exponential weighting functions are estimated using the EM algorithm. A filtering and smoothing method for LPV state space model is proposed to compute the posterior probabilities of the hidden variables in the E-step of the EM algorithm. A nonlinear state space process is assumed for generating inputs and

particle filter is employed to estimate the noise-free input from noise-corrupted measurements. Two numerical simulation examples, a first-order continuous LPV model and a nonlinear CSTR, as well as an experimental study on the multi-tank system are used to validate the effectiveness of the proposed approach.

Chapter 5

Conclusions

In this chapter, the work in the thesis is summarized, conclusions are drawn and future work directions are discussed.

5.1 Summary and conclusions of the thesis

This thesis mainly focused on the EIV problem identification and multiple model approaches for solving the nonlinear and time-varying problem. As one of the multiple model approaches, the Markov regime-switching model approach is applied to model the highly erratic pool price process and improve the existing pool price forecast.

In the Chapter 1, the background and motivations are demonstrated for the peak price prediction and the EIV system identification.

In the Chapter 2, the Alberta's electricity market is introduced, and the Markov regime-switching model is applied to model the pool price process. The EM algorithm is used to solve the MLE problem of the model parameters, and several hidden Markov model (HMM) approaches are proposed to generate the initial values for the EM algorithm. To solve the time-varying behavior problem of pool price, the 'similar' month selection rule is proposed based on the periodic patterns of pool price. Validation studies are provided using the initialization methods and the 'similar' month selection rule on price data in different time periods, and the results for one-hour ahead predictions and two-hour ahead predictions are presented. The results show that the prediction using the proposed approach can improve the existing price forecast in the high-price region. Also, the proposed two initialization methods help to improve the predictions in contrast to the existing initialization method.

In the Chapter 3, the identification problem for the EIV state space model is formulated. By introducing the input dynamics, the MLE of the EIV model parameter is solved under the EM framework. To estimate the hidden state of the EIV state space model, a filtering and smoothing approach for the state space model with colored stochastic inputs is proposed and applied in the E-step of the EM algorithm. Two numerical examples and one experimental example are presented to compare the proposed method with the existing EIV methods, which demonstrate that the proposed method can outperform the subspace EIV method and the BELS method by comparing the prediction errors.

In the Chapter 4, an LPV EIV state space model is formulated to approximate the nonlinear EIV system. By introducing the input dynamics, the EIV system identification problem is solved using the EM algorithm. To estimate the hidden state and local model identities of the LPV EIV state space model, a filtering and smoothing approach for LPV state space model is proposed and applied in the E-step of the EM algorithm. In the validation step, two numerical examples namely, a the continuous LPV process and the nonlinear CSTR process are presented. Furthermore, an experimental example the nonlinear tank system is used to validate the proposed method.

5.2 Directions for future work

In this thesis, the EIV problem and nonlinear problem are investigated and solved under the EM framework, including the identification of linear EIV state space modes and LPV EIV state space models. Besides, an application using the multiple model approach is presented in pool price prediction. To further extend the work in this thesis, following aspects can be considered:

1. The proposed pool price prediction approach providing the point forecast only, and incorporating uncertainty like confidence intervals into the price prediction would help to the related decision-making. The variance estimation procedure are presented in Appendix A.
2. The identification of linear EIV state space model with input dynamics is based on a separated model estimation framework. A efficient combination of the

estimations of EIV model and input dynamics would be helpful to generate more accurate estimation with acceptable computation increase.

3. The state estimation in an LPV EIV state space model has employed the proposed smoothing approach. However, there are some approximation steps called the ‘collapsing process’ due to the structure of the LPV state space model, which use a Gaussian mode to approximate a mixture of Gaussian. In such a case, the particle approach would be helpful in this non-Gaussian case. On the other hand, a new multiple state space model structure may be investigated, which can reduce the approximations when doing filtering and smoothing.

Bibliography

- [1] Market Surveillance Administrator. Alberta wholesale electricity market. 2010.
- [2] Ronald Huisman and Ronald Mahieu. Regime jumps in electricity prices. *Energy economics*, 25(5):425–434, 2003.
- [3] Tianbo Liu. Economic optimization of steam operation. M.sc. thesis, University of Alberta, 2013.
- [4] Torsten Söderström. Errors-in-variables methods in system identification. *Automatica*, 43(6):939–958, 2007.
- [5] Jin Wang and S Joe Qin. A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12(8):841–855, 2002.
- [6] Lei Xiong. *Stochastic models for electricity prices in Alberta*. PhD thesis, University of Calgary, 2004.
- [7] Rafal Weron. *Modeling and forecasting electricity loads and prices: A statistical approach*, volume 403. John Wiley & Sons, 2007.
- [8] Robert Ethier and Timothy Mount. Estimating the volatility of spot prices in restructured electricity markets and the implications for option values. *Cornell University, Ithaca New York*, 1998.
- [9] Xing Jin and Biao Huang. Robust identification of piecewise/switching autoregressive exogenous process. *AIChE journal*, 56(7):1829–1844, 2010.
- [10] Xing Jin, Biao Huang, and David S Shook. Multiple model lpv approach to nonlinear process identification with em algorithm. *Journal of Process Control*, 21(1):182–193, 2011.
- [11] Rogemar S Mamon and Robert J Elliott. *Hidden markov models in finance*, volume 4. Springer, 2007.
- [12] Richard Durbin. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.
- [13] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

- [14] JT-Y Cheung and George Stephanopoulos. Representation of process trends part i. a formal representation framework. *Computers & Chemical Engineering*, 14(4):495–510, 1990.
- [15] BR Bakshi and G Stephanopoulos. Representation of process trends iii. multi-scale extraction of trends from process data. *Computers & chemical engineering*, 18(4):267–302, 1994.
- [16] Nima Sammaknejad, Biao Huang, Alireza Fatehi, Yu Miao, Fangwei Xu, and Aris Espejo. Adaptive monitoring of the process operation based on symbolic episode representation and hidden markov models with application toward an oil sand primary separation. *Computers & Chemical Engineering*, 71:281–297, 2014.
- [17] Torsten Söderström. System identification for the errors-in-variables problem. *Transactions of the Institute of Measurement and Control*, page 0142331211414616, 2011.
- [18] Chun Tung Chou and Michel Verhaegen. Subspace algorithms for the identification of multivariable dynamic errors-in-variables models. *Automatica*, 33(10):1857–1869, 1997.
- [19] Tony Gustafsson. Subspace identification using instrumental variable techniques. *Automatica*, 37(12):2005–2010, 2001.
- [20] Biao Huang and Ramesh Kadali. *Dynamic modeling, predictive control and performance monitoring: a data-driven subspace approach*. Springer, 2008.
- [21] Wei Xing Zheng. A bias correction method for identification of linear dynamic errors-in-variables models. *Automatic Control, IEEE Transactions on*, 47(7):1142–1147, 2002.
- [22] Petre Stoica and Jian Li. On nonexistence of the maximum likelihood estimate in blind multichannel identification. *Signal Processing Magazine, IEEE*, 22(4):99–101, 2005.
- [23] Umberto Soverini and Torsten Söderström. Frequency domain maximum likelihood identification of noisy input–output models. In *IFAC 19th World Congress*, pages 24–29, 2014.
- [24] X Jin and B Huang. Identification of switched markov autoregressive exogenous systems with hidden switching state. *Automatica*, 48(2):436–441, 2012.
- [25] RB Gopaluni. A particle filter approach to identification of nonlinear processes under missing observations. *The Canadian Journal of Chemical Engineering*, 86(6):1081–1092, 2008.

- [26] Robert H Shumway and David S Stoffer. An approach to time series smoothing and forecasting using the em algorithm. *Journal of time series analysis*, 3(4):253–264, 1982.
- [27] Stuart Gibson and Brett Ninness. Robust maximum-likelihood estimation of multivariable dynamic systems. *Automatica*, 41(10):1667–1682, 2005.
- [28] Movellan J. R. Discrete time kalman filters and smoothers. *MPLab Tutorials*, 2011.
- [29] Robert H Shumway and David S Stoffer. *Time series analysis and its applications: with R examples*. Springer Science & Business Media, 2010.
- [30] Dan Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.
- [31] Holger Kantz and Thomas Schreiber. *Nonlinear time series analysis*, volume 7. Cambridge university press, 2004.
- [32] R Bhushan Gopaluni. Nonlinear system identification under missing observations: The case of unknown model structure. *Journal of Process Control*, 20(3):314–324, 2010.
- [33] Jing Deng, Li Xie, Lei Chen, Shima Khatibisepehr, Biao Huang, Fangwei Xu, and Aris Espejo. Development and industrial application of soft sensors with on-line bayesian model updating strategy. *Journal of Process Control*, 23(3):317–325, 2013.
- [34] Lennart Ljung. Prediction error estimation methods. *Circuits, Systems and Signal Processing*, 21(1):11–21, 2002.
- [35] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial & Applied Mathematics*, 11(2):431–441, 1963.
- [36] Mulang Chen, Swanand Khare, Biao Huang, Haitao Zhang, Eric Lau, and Enbo Feng. Recursive wavelength-selection strategy to update near-infrared spectroscopy model with an industrial application. *Industrial & Engineering Chemistry Research*, 52(23):7886–7895, 2013.
- [37] Ming Ma, Shima Khatibisepehr, and Biao Huang. A bayesian framework for real-time identification of locally weighted partial least squares. *AIChE Journal*, 61(2):518–529, 2015.
- [38] YC Zhu and Zuhua Xu. A method of lpv model identification for control. In *Preprint of IFAC World Congress*, volume 11, pages 5018–5023, 2008.
- [39] Lennart Ljung. System identification: Theory for the user. *PTR Prentice Hall Information and System Sciences Series*, 198, 1987.

- [40] Yaojie Lu and Biao Huang. Robust multiple-model lpv approach to nonlinear process identification using mixture t distributions. *Journal of Process Control*, 24(9):1472–1488, 2014.
- [41] Lei Chen, Aditya Tulsyan, Biao Huang, and Fei Liu. Multiple model approach to nonlinear system identification with an uncertain scheduling variable using em algorithm. *Journal of Process Control*, 23(10):1480–1496, 2013.
- [42] Lei Chen, Shima Khatibisepehr, Biao Huang, Fei Liu, and Yongsheng Ding. Nonlinear process identification in the presence of multiple correlated hidden scheduling variables with missing data. *AIChE Journal*, 2015.
- [43] Panqanamala Ramana Kumar and Pravin Varaiya. *Stochastic systems: estimation, identification and adaptive control*. Prentice-Hall, Inc., 1986.
- [44] Yaakov Bar-Shalom and Xiao-Rong Li. Estimation and tracking- principles, techniques, and software. *Norwood, MA: Artech House, Inc, 1993.*, 1993.
- [45] Chang-Jin Kim. Dynamic linear models with markov-switching. *Journal of Econometrics*, 60(1–2):1–22, 1994.
- [46] Zoubin Ghahramani and Geoffrey E Hinton. Variational learning for switching state-space models. *Neural computation*, 12(4):831–864, 2000.
- [47] Nicholas Kantas, Arnaud Doucet, Sumeetpal Sindhu Singh, and Jan Marian Maciejowski. An overview of sequential monte carlo methods for parameter estimation in general state-space models. In *15th IFAC Symposium on System Identification (SYSID), Saint-Malo, France.(invited paper)*, volume 102, page 117, 2009.
- [48] P Jeffrey Harrison and Colin F Stevens. Bayesian forecasting. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 205–247, 1976.
- [49] R Senthil, K Janarthanan, and J Prakash. Nonlinear state estimation using fuzzy kalman filter. *Industrial & engineering chemistry research*, 45(25):8678–8688, 2006.
- [50] Zuhua Xu, Jun Zhao, Jixin Qian, and Yucai Zhu. Nonlinear mpc using an identified lpv model. *Industrial & Engineering Chemistry Research*, 48(6):3043–3051, 2009.
- [51] Shima Khatibisepehr and Biao Huang. Dealing with irregular data in soft sensors: Bayesian method and comparative study. *Industrial & Engineering Chemistry Research*, 47(22):8713–8723, 2008.
- [52] Jing Deng and Biao Huang. Identification of nonlinear parameter varying systems with missing output data. *AIChE Journal*, 58(11):3454–3467, 2012.

- [53] Xiao-Li Meng and Donald B Rubin. Using em to obtain asymptotic variance-covariance matrices: The sem algorithm. *Journal of the American Statistical Association*, 86(416):899–909, 1991.
- [54] Brian DO Anderson and John B Moore. *Optimal filtering*. Courier Corporation, 2012.

Appendix A

Variance estimation for pool price prediction

A.1 Supplemented EM algorithm

The expectation maximization (EM) algorithm does not produce the asymptotic variance-covariance matrices for parameters automatically. Therefore, we need to search for other methods to compute the parameter uncertainties when we are using EM algorithm. The large-sample variance-covariance matrix of $(\Theta - \Theta^*)$ based on observation C_{obs} , V , is used to compute the uncertainties in the parameter estimation, which is derived as:

$$V = I_o(\Theta^*|C_{obs})^{-1} \quad (\text{A.1})$$

where, Θ^* is the final estimation of Θ ; $I_o(\Theta|C_{obs})$ is the observed information matrix, defined as:

$$I_o(\Theta|C_{obs}) = -\frac{\partial^2 \log P(C_{obs}|\Theta)}{\partial \Theta \cdot \partial \Theta} \quad (\text{A.2})$$

The observed information matrix cannot be computed directly due to the existence of missing data C_{mis} . To solve this problem, a supplemented expectation maximization (SEM) algorithm is proposed in [53], where the large-sample variance-covariance matrix is derived as:

$$V = I_c^{-1} + \Delta V \quad (\text{A.3})$$

where, I_c^{-1} is the conditional expectation of complete-data observed information matrix when $\Theta = \Theta^*$, defined as:

$$I_c = E_{C_{mis}|\Theta^*, C_{obs}} I_o(\Theta^*|C_{obs}, C_{mis}) \quad (\text{A.4})$$

ΔV is the increase in variance due to missing data, defined as:

$$\Delta V = I_c^{-1} DM(I - DM)^{-1} \quad (\text{A.5})$$

DM is the Jacobian matrix of the implicit mapping function $\Theta^{new} = M(\Theta^{old})$ in the EM algorithm at $\Theta = \Theta^*$, which is computed using the iterative SEM algorithm [53].

A.2 Uncertainty estimation for pool price prediction

The pool price point prediction approach is proposed in Chapter 2, however, it would be preferred to predict intervals for future pool price movement than simply point estimation in a risk management view.

Take one-hour ahead prediction as example, the estimation $E(k+1)$ is expressed as a combination of estimation given by each local model, presented as:

$$E(k+1) = \sum_{i=1}^M \sum_{j=1}^M P(I_k = i | C_k, \Theta) \cdot a_{ij} \cdot \phi_{k+1}^T \theta_j \quad (\text{A.6})$$

where, the probability for each local model is predicted by Markov property as $P(I_{k+1} = j) = P(I_k = i) * a_{ij}$, $\forall i, j = 1, \dots, M$. The uncertainties also existed all parameter estimation, and for simplicity we only consider the uncertainties in the local model parameter estimation θ_{I_k} . The covariance expression of one-hour ahead prediction $E(k+1)$ is derived using the uncertainty (covariance) in the local model parameters as:

$$Cov[E(k+1)] = W_{k+1} * Cov(\Theta_l) * W_{k+1}^T \quad (\text{A.7})$$

where,

$$W_{k+1} = \left[\sum_{i=1}^M P(I_k = i | C_k, \Theta) a_{i1} \phi_{k+1}^T \quad \dots \quad \sum_{i=1}^M P(I_k = i | C_k, \Theta) a_{iM} \phi_{k+1}^T \right] \quad (\text{A.8})$$

$$\Theta_l = \left[\theta_1^T, \quad \dots, \quad \theta_M^T \right]^T \quad (\text{A.9})$$

The covariance of local model parameters for $M = 3$ are derived as:

$$Cov(\Theta_l) = \left[\begin{array}{ccc} \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_1 \partial \theta_1} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_1 \partial \theta_2} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_1 \partial \theta_3} \right]^{-1} \\ \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_2 \partial \theta_1} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_2 \partial \theta_2} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_2 \partial \theta_3} \right]^{-1} \\ \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_3 \partial \theta_1} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_3 \partial \theta_2} \right]^{-1} & \left[-\frac{\partial^2 Q(\Theta)}{\partial \theta_3 \partial \theta_3} \right]^{-1} \end{array} \right] \bigg|_{\Theta=\Theta^*} + \Delta V(\Theta_l) \quad (\text{A.10})$$

where, the increase in covariance of Θ_l is computed using the SEM algorithm [53]; the conditional expectation of complete-data observed information matrix are derived as:

$$\begin{aligned}
-\frac{\partial^2 Q(\Theta)}{\partial \theta_i \partial \theta_i} \Big|_{\Theta=\Theta^*} &= \sum_{k=1}^N P(I_k = i | C_{obs}, \Theta^*) \phi_k \phi_k^T / \sigma^{*2}, \quad \forall i = 1, 2, 3 \\
-\frac{\partial^2 Q(\Theta)}{\partial \theta_i \partial \theta_j} \Big|_{\Theta=\Theta^*} &= 0, \quad \forall i \neq j \in \{1, 2, 3\}
\end{aligned} \tag{A.11}$$

Appendix B

Detailed derivations for section 3.3.3

B.1 Detailed derivation of equation (3.43)

To compute the covariance smoother of X_k and Z_k , $M_{k|N}^{xz} = E[(X_k - X_{k|N})(Z_k - Z_{k|N})^T]$, the state smoother equations for both X_k and Z_k are rearranged by subtracting both hand sides of the equations (3.26) and smoothing equation for Z_k [28] from X_k and Z_k , respectively, leading to:

$$(X_k - x_{k|N}) + J_k^x x_{k+1|N} = (X_k - x_{k|k}) + J_k^x x_{k+1|k} \quad (\text{B.1})$$

$$(Z_k - z_{k|N}) + J_k^z z_{k+1|N} = (Z_k - z_{k|k}) + J_k^z z_{k+1|k} \quad (\text{B.2})$$

Then, multiply the left-hand side terms of (B.1) and (B.2), and equate the result to the corresponding result of the right-hand side terms. By taking joint expectation of both sides over all the random variables including $X_{1:N}$ and $Z_{1:N}$ with all possible realizations of $Y_{1:N}$, $U_{1:N}$, $U_{1:N}^o$, we have the new left-hand side as:

$$\text{left-hand side} = E[(X_k - X_{k|N})(Z_k - Z_{k|N})^T] + J_k^x E[X_{k+1|N} Z_{k+1|N}^T] J_k^{zT} \quad (\text{B.3})$$

and the new right-hand side as:

$$\text{right-hand side} = E[(X_k - X_{k|k})(Z_k - Z_{k|k})^T] + J_k^x E[X_{k+1|k} Z_{k+1|k}^T] J_k^{zT} \quad (\text{B.4})$$

Further, the following equations are obtained based on the projection theorem [54], where $Z_{k|N} = E(Z_k | U_{1:N}, U_{1:N}^o)$ is an approximation of $E(Z_k | Y_{1:N}, U_{1:N}, U_{1:N}^o)$

and $Z_{k+1|k} = E(Z_{k+1}|U_{1:k}, U_{1:k}^o)$ is an approximation of $E(Z_{k+1}|Y_{1:k}, U_{1:k}, U_{1:k}^o)$:

$$\begin{aligned}
E[(X_k - X_{k|N})Z_{k+1|N}^T] &= 0 \\
E[(Z_k - Z_{k|N})X_{k+1|N}^T] &\approx 0 \\
E[(X_k - X_{k|k})Z_{k+1|k}^T] &= 0 \\
E[(Z_k - Z_{k|k})X_{k+1|k}^T] &\approx 0
\end{aligned} \tag{B.5}$$

where $X_{k|N}$ denotes the estimator of X_k given variables $Y_{1:N}, U_{1:N}, U_{1:N}^o$ while $x_{k|N}$ denotes the estimate of X_k given one of the realizations as $\{y_{1:N}, u_{1:N}, u_{1:N}^o\}$, and similarly for other variables. It is also easy to derive following equations based on the projection theorem [54]:

$$\begin{aligned}
E[(X_{k+1} - X_{k+1|N})Z_{k+1|N}^T] &= 0 \\
E[(Z_{k+1} - Z_{k+1|N})X_{k+1|N}^T] &\approx 0 \\
\Rightarrow E(Z_{k+1}X_{k+1|N}^T) &\approx E(Z_{k+1|N}X_{k+1|N}^T) \\
E[(X_{k+1} - X_{k+1|k})Z_{k+1|k}^T] &= 0 \\
E[(Z_{k+1} - Z_{k+1|k})X_{k+1|k}^T] &\approx 0 \\
\Rightarrow E(Z_{k+1}X_{k+1|k}^T) &\approx E(Z_{k+1|k}X_{k+1|k}^T)
\end{aligned} \tag{B.6}$$

Therefore:

$$\begin{aligned}
E[(X_{k+1} - X_{k+1|N})(Z_{k+1} - Z_{k+1|N})^T] &= E[X_{k+1}Z_{k+1}^T - X_{k+1|N}Z_{k+1}^T - \\
&\quad (X_{k+1} - X_{k+1|N})Z_{k+1|N}^T] \\
&= E[X_{k+1}Z_{k+1}^T] - E[X_{k+1|N}Z_{k+1|N}^T] \\
E[(X_{k+1} - X_{k+1|k})(Z_{k+1} - Z_{k+1|k})^T] &= E[X_{k+1}Z_{k+1}^T - X_{k+1|k}Z_{k+1}^T - \\
&\quad (X_{k+1} - X_{k+1|k})Z_{k+1|k}^T] \\
&= E(X_{k+1}Z_{k+1}^T) - E(X_{k+1|k}Z_{k+1|k}^T)
\end{aligned} \tag{B.7}$$

By rearranging (B.7), we have:

$$\begin{aligned}
E(X_{k+1|N}Z_{k+1|N}^T) &= E(X_{k+1}Z_{k+1}^T) - E[(X_{k+1} - X_{k+1|N})(Z_{k+1} - Z_{k+1|N})^T] \\
E(X_{k+1|k}Z_{k+1|k}^T) &= E(X_{k+1}Z_{k+1}^T) - E[(X_{k+1} - X_{k+1|k})(Z_{k+1} - Z_{k+1|k})^T]
\end{aligned} \tag{B.8}$$

Using equation (B.8), the joint expectation equation is derived as:

$$\begin{aligned}
E[(X_k - X_{k|N})(Z_k - Z_{k|N})^T] &= E[(X_k - X_{k|k})(Z_k - Z_{k|k})^T] + J_k^x \{E[(X_{k+1} - \\
&\quad X_{k+1|N})(Z_{k+1} - Z_{k+1|N})^T] - E[(X_{k+1} - X_{k+1|k})(Z_{k+1} - Z_{k+1|k})^T]\} J_k^{zT}
\end{aligned} \tag{B.9}$$

Since the unconditional error covariance matrix $E[(X_k - X_{k|N})(Z_k - Z_{k|N})^T]$ equals the conditional error covariance matrix $E[(X_k - X_{k|N})(Z_k - Z_{k|N})^T | y_{1:N}, u_{1:N}, u_{1:N}^o]$ or $P_{k|N}^{xz}$, and similarly for other unconditional covariance matrix [54]. The covariance smoother of X_k and Z_k as $P_{k|N}^{xz}$ is derived as follows:

$$P_{k|N}^{xz} = P_{k|k}^{xz} + J_k^x (P_{k+1|N}^{xz} - AP_{k|k}^{xz} A_o^T - BC_o P_{k|k}^z A_o^T) J_k^{zT} \quad (\text{B.10})$$

B.2 Detailed derivation of equation (3.45)

To compute the lag-one covariance smoother of X_k and Z_k , first the covariance filter with time step $k = N$ is calculated using equation (3.33). This gives the initialization $M_{N|N}^{xz}$ for recursively computing $M_{k|N}^{xz}$:

$$\begin{aligned} M_{N|N}^{xz} &= E[(X_N - X_{N|N})(Z_{N-1} - Z_{N-1|N})^T] \\ &= E\{[(I - K_N^z C)(X_N - X_{N|N-1}) - K_N^z \tilde{y}_N][(Z_{N-1} - Z_{N-1|N-1}) - \\ &\quad J_{N-1}^z K_N^z C_o (Z_N - Z_{N|N-1})]'\} \\ &= (I - K_N^x C) \cdot \underbrace{E[(X_N - X_{N|N-1})(Z_{N-1} - Z_{N-1|N-1})^T]}_{M_{N|N-1}^{xz}} - \\ &\quad (I - K_N^x C) \cdot \underbrace{E[(X_N - X_{N|N-1})(Z_N - Z_{N|N-1})^T]}_{P_{N|N-1}^{xz}} \cdot (J_{N-1}^z K_N^z C_o)^T \end{aligned} \quad (\text{B.11})$$

where the underbraced terms are,

$$\begin{aligned} P_{k|k-1}^{xz} &= E\{(X_k - X_{k|k-1})[A_o(Z_{k-1} - Z_{k-1|k-1}) + w_{k-1}^o]^T\} \\ &= E[(X_k - X_{k|k-1})(Z_{k-1} - Z_{k-1|k-1})^T] A_o^T = M_{k|k-1}^{xz} A_o^T \end{aligned} \quad (\text{B.12})$$

$$\begin{aligned} M_{k|k-1}^{xz} &= E\{[A \cdot (X_{k-1} - X_{k-1|k-1}) + BC_o(Z_{k-1} - \\ &\quad Z_{k-1|k-1}) + w_{k-1}](Z_{k-1} - Z_{k-1|k-1})^T\} \\ &= A \cdot E[(X_{k-1} - X_{k-1|k-1})(Z_{k-1} - Z_{k-1|k-1})^T] + \\ &\quad BC_o \cdot E[(Z_{k-1} - Z_{k-1|k-1})(Z_{k-1} - Z_{k-1|k-1})^T] \\ &= AP_{k-1|k-1}^{xz} + BC_o P_{k-1|k-1}^z \end{aligned} \quad (\text{B.13})$$

Therefore, the equation for $M_{N|N}^{xz}$ is given as:

$$M_{N|N}^{xz} = (I - K_N^x C)(AP_{N-1|N-1}^{xz} + BC_o P_{N-1|N-1}^z)(I - J_{N-1}^z K_N^z C_o A_o)^T \quad (\text{B.14})$$

Then to calculate $M_{k|N}^{xz}$ as $E[(X_k - X_{k|N})(Z_{k-1} - Z_{k-1|N})^T]$, a similar method as computing $P_{k|N}^{xz}$ is used. The state smoother equations for X_k and Z_{k-1} are arranged and presented as follows:

$$(X_k - X_{k|N}) + J_k^x X_{k+1|N} = (X_k - X_{k|k}) + J_k^x X_{k+1|k} \quad (\text{B.15})$$

$$(Z_{k-1} - Z_{k-1|N}) + J_{k-1}^z Z_{k|N} = (Z_{k-1} - Z_{k-1|k-1}) + J_{k-1}^z Z_{k|k-1} \quad (\text{B.16})$$

Then, multiply the left-hand side terms of these two equations, and equate the result to the corresponding result of the right-hand side terms. We also use the following identifies obtained using the projection theorem [54] for simplification:

$$\begin{aligned} E[(X_k - X_{k|N})Z_{k|N}^T] &= 0 \\ E[(Z_{k-1} - Z_{k-1|N})X_{k+1|N}^T] &\approx 0 \\ E[(X_k - X_{k|k-1})Z_{k|k-1}^T] &= 0 \\ E[(Z_{k-1} - Z_{k-1|k-1})Z_{k|k-1}^T] &= 0 \\ E[(Z_k - Z_{k|k-1})Z_{k|k-1}^T] &= 0 \\ E[(Z_{k-1} - Z_{k-1|k-1})X_{k|k-1}^T] &\approx 0 \end{aligned} \quad (\text{B.17})$$

By taking jointly expectation of both sides over all the random variables including $X_{1:N}$ and $Z_{1:N}$ with all possible realizations of $Y_{1:N}$, $U_{1:N}$, $U_{1:N}^o$, we have the left-hand side of the joint expectation equation as:

$$\begin{aligned} \text{left-hand side} &= E[(X_k - X_{k|N})(Z_{k-1} - Z_{k-1|N})^T] + \\ & \quad J_k^z E[X_{k+1|N} Z_{k|N}^T] J_{k-1}^{zT} \end{aligned} \quad (\text{B.18})$$

The right-hand side terms of equation (B.15) are derived as:

$$\begin{aligned} (X_k - x_{k|k}) + J_k^x x_{k+1|k} &= (I - K_k^x C)(X_k - x_{k|k-1}) + J_k^x (Ax_{k|k} + BC_o z_{k|k}) \\ &= (I - K_k^x C)(X_k - X_{k|k-1}) + J_k^x [Ax_{k|k-1} + AK_k^x C(X_k - \\ & \quad X_{k|k-1}) + BC_o z_{k|k-1} + BC_o K_k^z C_o (Z_k - z_{k|k-1})] \end{aligned} \quad (\text{B.19})$$

Thus, we have the right-hand side of the joint expectation equation as:

$$\begin{aligned} \text{right-hand side} &= (I - K_k^x C + J_k^x AK_k^x C) E[(X_k - X_{k|k-1})(Z_{k-1} - Z_{k-1|k-1})^T] + \\ & \quad J_k^x BC_o K_k^z C_o \cdot E[(Z_k - Z_{k|k-1})(Z_{k-1} - Z_{k-1|k-1})^T] + \\ & \quad J_k^x A \cdot E[X_{k|k-1} Z_{k|k-1}^T] J_{k-1}^{zT} + J_k^x BC_o \cdot E[Z_{k|k-1} Z_{k|k-1}^T] J_{k-1}^{zT} \end{aligned} \quad (\text{B.20})$$

Also, we have the following results based on the projection theorem [54]:

$$E[Z_{k|k-1}Z_{k|k-1}^T] = E[Z_kZ_k^T] - E\{(Z_k - Z_{k|k-1})(Z_k - Z_{k|k-1})^T\} \quad (\text{B.21})$$

Using equation (B.8) and (B.21), the smoother equation with lag-one covariance of X_k and Z_{k-1} is derived as follows:

$$\begin{aligned} M_{k|N}^{xz} &= E\{(X_k - X_{k|N})(Z_{k-1} - Z_{k-1|N})^T\} \\ &= (I - K_k C + J_k^x A K_k^x C) \underbrace{E[(X_k - X_{k|k-1})(Z_{k-1} - Z_{k-1|k-1})^T]}_{M_{k|k-1}^{xz}} + \\ &\quad J_k^x B C_o K_k^z C_o \cdot \underbrace{E[(Z_k - Z_{k|k-1})(Z_{k-1} - Z_{k-1|k-1})^T]}_{M_{k|k-1}^z} + \\ &\quad J_k^x \underbrace{\{E[(X_{k+1} - X_{k+1|N})(Z_k - Z_{k|N})^T]\}}_{M_{k+1|N}^{xz}} - \\ &\quad A \cdot \underbrace{E[(X_k - X_{k|k-1})(Z_k - Z_{k|k-1})^T]}_{P_{k|k-1}^{xz}} - \\ &\quad B C_o \cdot \underbrace{E[(Z_k - Z_{k|k-1})(Z_k - Z_{k|k-1})^T]}_{P_{k|k-1}^z} \} J_{k-1}^z{}^T \end{aligned} \quad (\text{B.22})$$

Finally, the equation for $M_{k|N}^{xz}$ as:

$$\begin{aligned} M_{k|N}^{xz} &= (I - K_k^x C + J_k^x A K_k^x C)(A P_{k-1|k-1}^{xz} + B C_o P_{k-1|k-1}^z) + J_k^x B C_o \cdot \\ &\quad K_k^z C_o A_o P_{k-1|k-1}^z + J_k^x (M_{k+1|N}^{xz} - A P_{k|k-1}^{xz} - B C_o P_{k|k-1}^z) J_{k-1}^z{}^T \end{aligned} \quad (\text{B.23})$$

B.3 Detailed derivation of equation (3.47)

To calculate the lag-one covariance smoother for X_k , first, the lag-one covariance filter for time step $k = N$ is derived as:

$$\begin{aligned} M_{N|N}^x &= E[(X_N - X_{N|N})(X_{N-1} - X_{N-1|N})] \\ &= E\{(I - K_N^x C)(X_N - X_{N|N-1}) - K_N^x \tilde{y}_N\} [(X_{N-1} - X_{N-1|N-1}) - \\ &\quad J_{N-1}^x K_N^x C (X_N - X_{N|N-1}) - J_{N-1}^x K_N^x \tilde{y}_N] \} \\ &= (I - K_N^x C)[M_{N|N-1}^x - P_{t|t-1}(J_{N-1}^x K_N^x C)^T] - K_N^x \Sigma_{\tilde{y}} (J_{N-1}^x K_N^x)^T \end{aligned} \quad (\text{B.24})$$

Then, to compute the lag-one covariance smoother $M_{k|N}^x$, the smoother equations for X_k and X_{k-1} are rearranged as follows:

$$(X_k - X_{k|N}) + J_k^x X_{k+1|N} = (X_k - X_{k|k}) + J_k^x X_{k+1|k} \quad (\text{B.25})$$

$$(X_{k-1} - X_{k-1|N}) + J_{k-1}^x X_{k|N} = (X_{k-1} - X_{k-1|k-1}) + J_{k-1}^x X_{k|k-1} \quad (\text{B.26})$$

Then, multiply the left-hand side terms of these two equations, and equate that to the corresponding result of the right-hand side terms. By taking jointly expectation of both sides over all the random variables including $X_{1:N}$ and $Z_{1:N}$ with all possible realizations of $Y_{1:N}$, $U_{1:N}$, $U_{1:N}^o$ as well as using the following equations based on the projection theorem [54]:

$$\begin{aligned} E[(X_k - X_{k|N})X_{k|N}^T] &= 0 \\ E[(X_{k-1} - X_{k-1|N})X_{k+1|N}^T] &= 0 \\ E[(X_{k-1} - X_{k-1|k-1})X_{k|k-1}^T] &= 0 \\ E[(X_{k-1} - X_{k-1|k-1})Z_{k|k-1}^T] &= 0 \\ E[(Z_k - Z_{k|k-1})X_{k|k-1}^T] &\approx 0 \end{aligned} \quad (\text{B.27})$$

we have the left-hand side of the joint expectation equation as:

$$\begin{aligned} \text{left-hand side} &= E[(X_k - X_{k|N})(X_{k-1} - X_{k-1|N})^T] + \\ &J_k^x E[X_{k+1|N} X_{k|N}^T] J_{k-1}^{xT} \end{aligned} \quad (\text{B.28})$$

and right-hand side of the joint expectation equation as:

$$\begin{aligned} \text{right-hand side} &= (I - K_k^x C + J_k^x A K_k^x C) E[(X_k - X_{k|k-1})(X_{k-1} - \\ &X_{k-1|k-1})^T] + J_k^x B C K_k^z C_o \cdot E[(Z_k - Z_{k|k-1})(X_{k-1} - \\ &X_{k-1|k-1})^T] + J_k^x A \cdot E[X_{k|k-1} X_{k|k-1}^T] J_{k-1}^{xT} + \\ &J_k^x B C_o \cdot E[Z_{k|k-1} X_{k|k-1}^T] J_{k-1}^{xT} \end{aligned} \quad (\text{B.29})$$

Using the following equations based on projection theorem [54], we have the following expression:

$$\begin{aligned} E[X_{k+1|N} X_{k|N}^T] &= E[X_{k+1} X_k^T] - E[(X_{k+1} - X_{k+1|N})(X_k - X_{k|N})^T] \\ E[X_{k|k-1} X_{k|k-1}^T] &= E[X_k X_k^T] - E[(X_k - X_{k|k-1})(X_k - X_{k|k-1})^T] \\ E[Z_{k|k-1} X_{k|k-1}^T] &= E[Z_k X_k^T] - E[(Z_k - Z_{k|k-1})(X_k - X_{k|k-1})^T] \end{aligned} \quad (\text{B.30})$$

The smoother equation for lag-one covariance of X_k and X_{k-1} is derived as follows:

$$\begin{aligned}
M_{k|N}^x &= E[(X_k - X_{k|N})(X_{k-1} - X_{k-1|N})^T] \\
&= (I - K_k^x C + J_k^x A K_k^x C) \underbrace{E[(X_k - X_{k|k-1})(X_{k-1} - X_{k-1|k-1})^T]}_{M_{k|k-1}^x} + \\
&\quad J_k^x B C K_k^z C_o \cdot \underbrace{E[(Z_k - Z_{k|k-1})(X_{k-1} - X_{k-1|k-1})^T]}_{M_{k|k-1}^{xz}} + \\
&\quad J_k^x \underbrace{\{E[(X_{k+1} - X_{k+1|N})(X_k - X_{k|N})^T]\}}_{M_{k+1|N}^x} - A P_{k|k-1}^x - B C_o \cdot \\
&\quad \underbrace{E[(Z_k - Z_{k|k-1})(X_k - X_{k|k-1})^T]}_{P_{k|k-1}^{zx}} J_{k-1}^x{}^T
\end{aligned} \tag{B.31}$$

where,

$$\begin{aligned}
M_{k|k-1}^x &= E\{[A(X_{k-1} - X_{k-1|k-1}) + B C_o(Z_{k-1} - Z_{k-1|k-1}) + w_{k-1}] \cdot \\
&\quad (X_{k-1} - X_{k-1|k-1})^T\}
\end{aligned} \tag{B.32}$$

As: $E[w_{k-1}(X_{k-1} - X_{k-1|k-1})^T] = 0$, this will result

$$= A P_{k-1|k-1}^x + B C_o P_{k-1|k-1}^{xz}{}^T$$

$$M_{k|k-1}^{zx} = E\{[A_o(Z_{k-1} - Z_{k-1|k-1}) + \tilde{u}_{k-1}](X_{k-1} - X_{k-1|k-1})^T\} \tag{B.33}$$

As: $E[\tilde{u}_{k-1}(X_{k-1} - X_{k-1|k-1})^T] = 0$, this will result

$$= A_o P_{k-1|k-1}^{zx}{}^T$$

$$P_{k|k-1}^{zx} = P_{k|k-1}^{xz}{}^T \tag{B.34}$$

$$= (A P_{k-1|k-1}^{xz} A_o^T + B C_o P_{k-1|k-1}^z A_o^T) T$$

Finally we have the concise equation for $M_{k|N}^x$ as:

$$\begin{aligned}
M_{k|N}^x &= (I - K_k^x C + J_k^x A K_k^x C) (A P_{k-1|k-1}^x + B C_o P_{k-1|k-1}^{xz}{}^T) + \\
&\quad J_k^x B C K_k^z C_o A_o P_{k-1|k-1}^{xz}{}^T + J_k^x \{M_{k+1|N}^x - A P_{k|k-1}^x - \\
&\quad B C_o (A P_{k-1|k-1}^{xz} A_o^T + B C_o P_{k-1|k-1}^z A_o^T) T\} J_{k-1}^x{}^T
\end{aligned} \tag{B.35}$$

Appendix C

Smoothing step for the lag-one covariance of LPV EIV state space model

The smoothing step for the lag-one covariance

The lag-one covariance smoother for the multiple state space model is defined as:

$$\begin{aligned} M_{k|N}^j &\triangleq Cov[X_k, X_{k-1} | \phi_N, I_k = j] \\ &= E[(X_k - X_{k|N}^j)(X_{k-1} - X_{k-1|N}^{(*,j)})^T] \end{aligned} \quad (C.1)$$

where, $X_{k|N}$ denotes the estimator of X_k given variables $Y_{1:N}, U_{1:N}, U_{1:N}^o$, while $x_{k|N}$ denotes the estimate of X_k given one of the realizations as $\{y_{1:N}, u_{1:N}, u_{1:N}^o\}$, and similarly for other variables.

For recursive calculation, we need to calculate the following expressions first:

$$\begin{aligned} M_{k|N}^{(i,j,h)} &\triangleq Cov[X_k, X_{k-1} | \phi_N, I_{k+1} = h, I_k = j, I_{k-1} = i] \\ &= E[(X_k - X_{k|N}^{(i,j,h)})(X_{k-1} - X_{k-1|N}^{(*,i,j,h)})^T] \\ M_{k|N}^{(i,j,*)} &\triangleq Cov[X_k, X_{k-1} | \phi_N, I_k = j, I_{k-1} = i] \\ &= E[(X_k - X_{k|N}^{(i,j,*)})(X_{k-1} - X_{k-1|N}^{(*,i,j)})^T] \end{aligned} \quad (C.2)$$

where, $x_{k|N}^{(i,j,h)}$ and $x_{k-1|N}^{(*,i,j,h)}$ are introduced for the calculation of lag-one covariance smoother, which are derived as:

$$x_{k|N}^{(i,j,h)} \triangleq x_{k|k}^{(i,j)} + J_k^{(i,j,h)} [x_{k+1|N}^{(j,h,*)} - x_{k+1|k}^{(i,j,h)}] \quad (C.3)$$

$$x_{k|N}^{(*,j,h,g)} \triangleq x_{k|k}^j + J_k^{(*,j,h)} [x_{k+1|N}^{(*,h,g)} - x_{k+1|k}^{(j,h)}] \quad (C.4)$$

where, $x_{k+1|k}^{(i,j,h)} = A_h x_{k|k}^{(i,j)} + B_h u_k$; the smoother gain $J_k^{(i,j,h)}$ is derived as:

$$\begin{aligned} J_k^{(i,j,h)} &= C(X_k, X_{k+1} | \phi_k, I_{k+1} = h, I_k = j, I_{k-1} = i) \cdot \\ &V(X_{k+1} | \phi_k, I_{k+1} = h, I_k = j, I_{k-1} = i) \\ &= P_{k|k}^{(i,j)} A_h^T (A_h P_{k|k}^{(i,j)} A_h^T + Q)^{-1} \end{aligned} \quad (\text{C.5})$$

the state smoother equations for both X_k and X_{k-1} are rearranged by subtracting both the sides of the equations (C.3) and equation (C.4) from X_k and X_{k-1} , respectively, leading to:

$$(X_k - x_{k|N}^{(i,j,h)}) + J_k^{(i,j,h)} x_{k+1|N}^{(j,h,*)} = (X_k - x_{k|k}^{(i,j)}) + J_k^{(i,j,h)} x_{k+1|k}^{(i,j,h)} \quad (\text{C.6})$$

$$(X_{k-1} - x_{k-1|N}^{(*,i,j,h)}) + J_{k-1}^{(*,i,j)} x_{k|N}^{(*,j,h)} = (X_{k-1} - x_{k-1|k-1}^i) + J_{k-1}^{(*,i,j)} x_{k|k-1}^{(i,j)} \quad (\text{C.7})$$

where,

$$\begin{aligned} (X_k - x_{k|k}^{(i,j)}) + J_k^{(i,j,h)} x_{k+1|k}^{(i,j,h)} &= (A_j + K_k^{(i,j)} C_j A_j + J_k^{(i,j,h)} A_h K_k^{(i,j)} C_j A_j) \cdot \\ (X_{k-1} - x_{k-1|k-1}^i) + J_k^{(i,j,h)} x_{k+1|k-1}^{(i,j,h)} &+ (I + K_k^{(i,j)} C_j \\ J_k^{(i,j,h)} K_k^{(i,j)} C_j) w_{k-1} &+ (J_k^{(i,j,h)} K_k^{(i,j)} + K_k^{(i,j)}) \tilde{y}_k \end{aligned} \quad (\text{C.8})$$

By multiplying the left-hand side terms of (C.6) and (C.7), and equating that to the corresponding result of the right-hand side terms, then taking joint expectation for both sides over all the random variables including $X_{1:N}$ with all possible realizations of $Y_{1:N}$, $U_{1:N}$, $U_{1:N}^o$, leads to new expression for left-hand side as:

$$\begin{aligned} \text{Left-hand side} &= E[(X_k - X_{k|N}^{(i,j,h)})(X_{k-1} - X_{k-1|N}^{(*,i,j,h)})^T] + \\ &J_k^{(i,j,h)} E[X_{k+1|N}^{(j,h,*)} X_{k|N}^{(*,j,h)T}] J_{k-1}^{(*,i,j)T} \end{aligned} \quad (\text{C.9})$$

and the new expression for right-hand side is given as,

$$\begin{aligned} \text{Right-hand side} &= (A_j + K_k^{(i,j)} C_j A_j + J_k^{(i,j,h)} A_h K_k^{(i,j)} C_j A_j) E[(X_{k-1} - X_{k-1|k-1}^i) \cdot \\ (X_{k-1} - X_{k-1|k-1}^i)^T] &+ J_k^{(i,j,h)} E[X_{k+1|k-1}^{(i,j,h)} X_{k|k-1}^{(i,j)T}] J_{k-1}^{(*,i,j)T} \end{aligned} \quad (\text{C.10})$$

Further, the following equations are obtained based on the projection theorem [54], yielding:

$$\begin{aligned} E[(X_k - X_{k|N}^{(i,j,h)}) X_{k|N}^{(*,j,h)T}] &= 0 \\ E[(X_{k-1} - X_{k-1|N}^{(*,i,j,h)}) X_{k+1|N}^{(j,h,*)} T] &= 0 \\ E[(X_{k-1} - X_{k-1|k-1}^i) X_{k+1|k-1}^{(i,j,h)T}] &= 0 \\ E[(X_{k-1} - X_{k-1|k-1}^i) X_{k|k-1}^{(i,j)T}] &= 0 \end{aligned} \quad (\text{C.11})$$

It is also easy to derive following equations based on the projection theorem [54]:

$$\begin{aligned}
E[(X_{k+1} - X_{k+1|N}^{(j,h,*)})X_{k|N}^{(*,j,h)T}] &= 0 \\
E[(X_k - X_{k|N}^{(*,j,h)})X_{k+1|N}^{(j,h,*)T}] &= 0 \\
\Rightarrow E[X_k X_{k+1|N}^{(j,h,*)T}] &= E[X_{k|N}^{(*,j,h)} X_{k+1|N}^{(j,h,*)T}] \\
E[(X_k - X_{k|k-1}^{(i,j)})X_{k+1|k-1}^{(i,j,h)T}] &= 0 \\
E[(X_{k+1} - X_{k+1|k-1}^{(i,j,h)})X_{k|k-1}^{(i,j)T}] &\approx 0 \\
\Rightarrow E[X_{k+1} X_{k|k-1}^{(i,j)T}] &\approx E[X_{k+1|k-1}^{(i,j,h)} X_{k|k-1}^{(i,j)T}]
\end{aligned} \tag{C.12}$$

Therefore:

$$\begin{aligned}
E[(X_{k+1} - X_{k+1|N}^{(j,h,*)})(X_k - X_{k|N}^{(*,j,h)})^T] &= E[X_{k+1} X_k^T - X_{k+1|N}^{(j,h,*)} X_k^T - \\
&\quad (X_{k+1} - X_{k+1|N}^{(j,h,*)}) X_{k|N}^{(*,j,h)T}] \\
&= E[X_{k+1} X_k^T] - E[X_{k+1|N}^{(j,h,*)} X_{k|N}^{(*,j,h)T}] \\
E[(X_{k+1} - X_{k+1|k-1}^{(i,j,h)})(X_k - X_{k|k-1}^{(i,j)})^T] &= E[X_{k+1} X_k^T - X_{k+1|k-1}^{(i,j,h)} X_k^T - \\
&\quad (X_{k+1} - X_{k+1|k-1}^{(i,j,h)}) X_{k|k-1}^{(i,j)T}] \\
&= E[X_{k+1} X_k^T] - E[X_{k+1|k-1}^{(i,j,h)} X_{k|k-1}^{(i,j)T}]
\end{aligned} \tag{C.13}$$

By rearranging (C.13), we have:

$$\begin{aligned}
E\{X_{k+1|N}^{(j,h,*)} X_{k|N}^{(*,j,h)T}\} &= E[X_{k+1} X_k^T] - \underbrace{E[(X_{k+1} - X_{k+1|N}^{(j,h,*)})(X_k - X_{k|N}^{(*,j,h)})^T]}_{M_{k+1|N}^{(j,h,*)}} \\
E[X_{k+1|k-1}^{(i,j,h)} X_{k|k-1}^{(i,j)T}] &= E[X_{k+1} X_k^T] - \underbrace{E[(X_{k+1} - X_{k+1|k-1}^{(i,j,h)})(X_k - X_{k|k-1}^{(i,j)})^T]}_{M_{k+1|k-1}^{(i,j,h,*)}}
\end{aligned} \tag{C.14}$$

where,

$$\begin{aligned}
M_{k+1|k-1}^{(i,j,h,*)} &= E\{[A_h(X_k - X_{k|k-1}^{(i,j)}) + w_k](X_k - X_{k|k-1}^{(i,j)})^T\} \\
&= A_h E[(X_k - X_{k|k-1}^{(i,j)})(X_k - X_{k|k-1}^{(i,j)})^T] \\
&= A_h P_{k|k-1}^{(i,j)}
\end{aligned} \tag{C.15}$$

Thus, the smoothing lag-one covariance is derived as:

$$\begin{aligned}
M_{k|N}^{(i,j,h)} &= (A_j + K_k^{(i,j)} C_j A_j + J_k^{(i,j,h)} A_h K_k^{(i,j)} C_j A_j) P_{k-1|k-1}^i + \\
&\quad J_k^{(i,j,h)} (M_{k+1|N}^{(j,h,*)} - A_h P_{k|k-1}^{(i,j)}) J_{k-1}^{(*,i,j)T}
\end{aligned} \tag{C.16}$$

The starting point smoother $M_{N|N}^{(i,j,*)}$ is derived as:

$$\begin{aligned}
M_{N|N}^{(i,j,*)} &= E[(X_N - X_{N|N}^{(i,j)})(X_{N-1} - X_{N-1|N}^{(*,i,j)})^T] \\
&= E\{[(A_j - K_N^{(i,j)}C_jA_j)(X_{N-1} - X_{N-1|N-1}^i) + (I - K_N^{(i,j)}C_j)w_{N-1} - \\
&\quad K_N^{(i,j)}\tilde{y}_N][(I - J_{N-1}^{(i,j)}K_N^{(i,j)}C_jA_j)(X_{N-1} - X_{N-1|N-1}^i) - \\
&\quad J_{N-1}^{(i,j)}K_N^{(i,j)}C_jw_{N-1} - J_{N-1}^{(i,j)}K_N^{(i,j)}\tilde{y}_N]^T\} \\
&= (A_j - K_N^{(i,j)}C_jA_j)P_{N-1|N-1}^i(I - J_{N-1}^{(i,j)}K_N^{(i,j)}C_jA_j)^T - \\
&\quad (I - K_N^{(i,j)}C_j)Q(J_{N-1}^{(i,j)}K_N^{(i,j)}C_j)^T + K_N^{(i,j)}R(J_{N-1}^{(i,j)}K_N^{(i,j)})^T
\end{aligned} \tag{C.17}$$

A similar collapsing form as [48], the collapsed distributions for $M_{k|N}^{(i,j,*)}$ and $M_{k|N}^j$ are given as:

$$\begin{aligned}
M_{k|N}^{(i,j,*)} &= \sum_{h=1}^M P(I_{k+1} = h|C_{obs})[M_{k|N}^{(i,j,h)} + (x_{k|N}^{(i,j,*)} - x_{k|N}^{(i,j,h)})(x_{k-1|N}^{(*,i,j)} - x_{k-1|N}^{(*,i,j,h)})^T] \\
M_{k|N}^j &= \sum_{i=1}^M P(I_{k-1} = i|C_{obs})[M_{k|N}^{(i,j,*)} + (x_{k|N}^j - x_{k|N}^{(i,j,*)})(x_{k-1|N}^{(*,j)} - x_{k-1|N}^{(*,i,j)})^T]
\end{aligned} \tag{C.18}$$