University of Alberta

Identification of Switched Linear Systems

by

Jiadong Wang

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Control

Department of Electrical and Computer Engineering

Fall 2013
Edmonton, Alberta

To my parents and my wife

for their love and support over the years

Abstract

This thesis is concerned with identification of switched linear systems (SLSs), which is an important part in model-based control. There are a large number of physical systems that can be represented or approximated by SLSs. Therefore, the study of SLSs has attracted much attention over the past decades. As input/output data points of SLSs are sampled from a couple of linear modes (or subsystems), conventional methods are not applicable. For this reason, many research results on identification of SLSs have emerged in recent years.

For offline identification of SLSs, many of the existing methods are designed with the assumption that the number of modes is known. This information is, however, not always available in practice. In this thesis, a set membership identification approach is employed to remove this restriction. In its implementation, a major challenge is how to find a maximum feasible subsystem in an efficient way. To achieve this goal, a relaxed heuristic (RH) solution is proposed. Moreover, for SLSs with multiple unknown noise levels, an extended version of the RH solution is subsequently developed.

For online identification, a good mode detection or online data classification procedure is critical to estimation performance. One simple and effective way is to directly run a mode detection function before parameter estimation. However, this creates a problem that there may involve a lot of mode mismatches in the mode detection, which has negative impacts on estimation results. In the thesis, two effective algorithms are developed to overcome this

problem from different perspectives.

In addition to the above aspects, identification of periodically switched linear systems has also been considered in the thesis.

Acknowledgements

I would like to thank all people who helped me and inspired me to progress towards the completion of my PhD thesis. Especially, I want to express my sincere appreciation to my supervisor, Dr. Tongwen Chen, for his significant contributions of careful guidance and generous sharing of time and expertise. He allowed me the freedom to conduct the research that is interesting to me, while providing help to keep me on the right track. With his enormous support, my PhD study became truly enjoyable and rewarding.

I also would like to thank my other committee members, Dr. Alan Lynch, Dr. Qing Zhao, Dr. Vinay Prasad, and Dr. Fangxiang Wu, for their insightful suggestions and comments on my thesis.

My special gratitude goes to all the members in the Advanced Control Group and other staffs in the Department of Electrical and Computer Engineering who gave their kind help to me during my graduate experience.

Finally, I owe a great debt of gratitude to my parents for all of the sacrifices that they have made for me in my life. I must thank my wife for standing beside me throughout these years. To them I dedicate this thesis.

Financial support from Natural Sciences and Engineering Research Council of Canada and Queen Elizabeth II Graduate Scholarship, University of Alberta, is gratefully acknowledged.

Contents

List of Tables

List of Figures

List of Acronyms

ARX	AutoRegresive with eXogenous inputs
ARMAX	AutoRegressive-Moving-Average with eXogenous inputs
BJ	Box-Jenkins
CSTR	Continuous Stirred Tank Reactor
EM	Expectation-Maximization
FS	Forward Search
I/O	Input/Output
HT	Hough Transform
JMLS	Jump Markov Linear System
LP	Linear Programming
LMR	Largest Magnitude of Residual
LTS	Least Trimmed Squares
OE	Output Error
MAX FS	MAXimum Feasible Subsystem/Subset
MIN PFS	Partition an infeasible system into a MINimum number of Feasible Subsystems
MSD	Maximum Standard Deviation
MSE	Mean Squared Error
PSLS	Periodically Switched Linear System
PWARX	Piecewise ARX
QP	Quadratic Programming
RH	Relaxed Heuristic

RLS	Recursive Least Squares
SARX	Switched ARX
SLS	Switched Linear System
SNR	Signal to Noise Ratio
SVD	Singular Value Decomposition
VE	Vertex Enumeration

Chapter 1

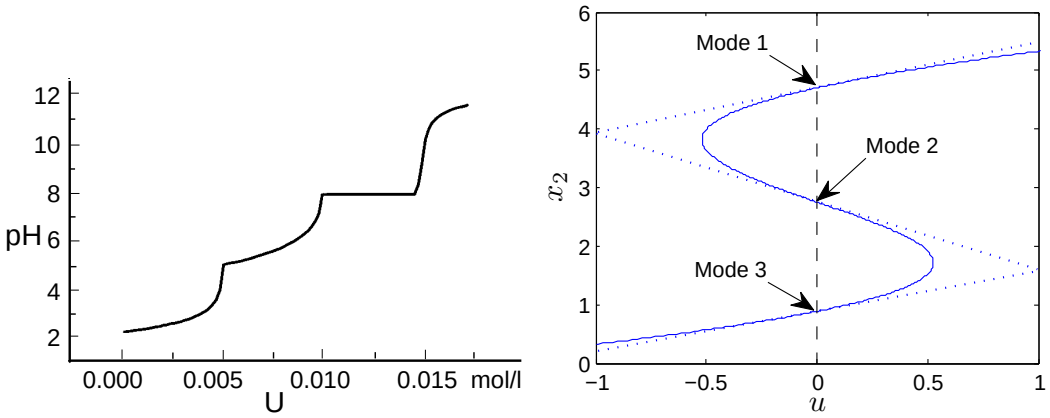
Introduction

1.1 Motivation

In control engineering, system identification is used to build mathematical models for physical systems, which helps researchers to understand and control dynamic behavior of processes systems. The development of system identification began approximately at the middle of 1960's. During the past fifty years, the research on the identification of linear systems has evolved into a relatively mature area. There are a number of well established approaches in the literature. In comparison, the research on the identification of nonlinear systems is still at the developing stage, since most nonlinear systems are complex in nature and a variety of nonlinearities exist. It is nearly impossible to design a general identification method that can be applied to all nonlinear systems. Usually, the research of nonlinear system identification is focused on specific categories of systems.

Switched Linear Systems (SLSs), as a special class of nonlinear systems, typically consist of a group of linear modes (or subsystems) and a switching rule that decides the switching among them. SLSs widely exist in many fields of engineering, such as air traffic systems [30, 54], automotive systems, biological systems [21], robotic systems [16, 60], chemical processes [41] and communication networks [29, 40].

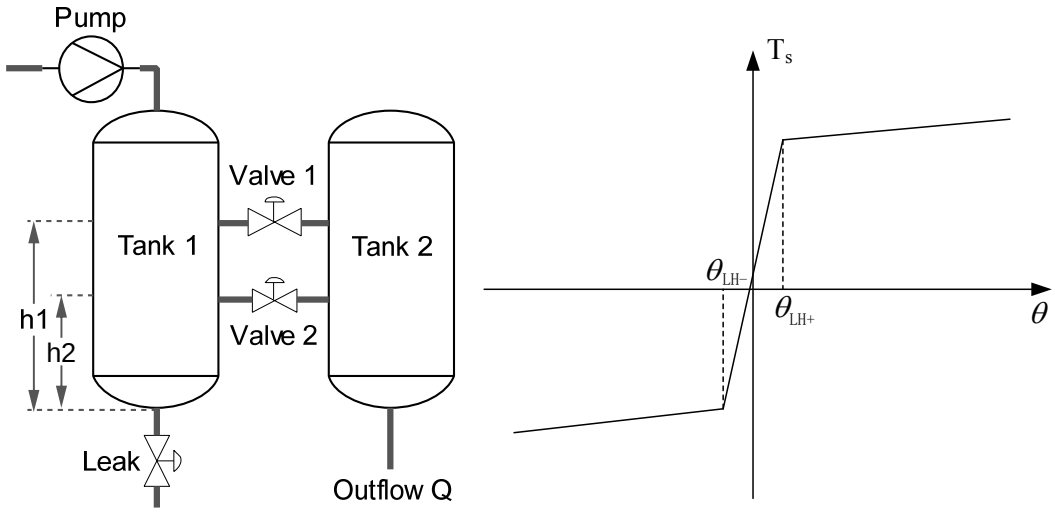
The motivation of studying SLSs comes from both theoretical and practical needs. Firstly, a lot of nonlinear systems can be approximated with multiple linear models and designing multi-model based controllers is relatively easier. In the literature, many applications exist, e.g., [49, 55, 62]. Fig. 1.1 (a) depicts a pH process that is taken from [50]. A piecewise linear function can be



(a) A pH process [50] (b) A CSTR process [64]

Figure 1.1: The steady state maps of two nonlinear processes

used to describe the relation between the pH value and the concentration of constituent U. Fig. 1.1 (b) shows another example, which is a nonlinear CSTR process. It is seen that state x_2 is almost locally linear with the input u. This process will be later studied in Chapter 2.



(a) A two-tank process (b) Spring torque vs. throttle position

Figure 1.2: The examples of systems with switching processes

Secondly, a lot of processes such as a gear shifting in a car, night or day effects on temperature control systems, and random switches in network bandwidth values are intrinsically composed of several working modes. For example, Fig. 1.2 (a) shows two cascaded tanks. The aim of the system is to control

the outflow of tank 2. If we suppose that valves 1 and 2 are both open, then the liquid inflow of tank 2 will experience an abrupt change when the liquid level of tank 1 is equal to h1 or h2. Thus, it belongs to a three-mode SLS. Fig. 1.2 (b) shows return spring characteristics of a throttle valve system [63], where the return spring is used for pulling throttle plate into a limp-home region, that is, [$\theta_{\text{LH-}}$, $\theta_{\text{LH+}}$]. When the throttle plate rotates from a fully closed position to a fully open position, the spring torque (T_s) is a piecewise linear function of its angle (θ). Identification of this system was discussed in [65].

Thirdly, from theoretical point of view, multi-controllers are often used to improve the control performance, see, e.g., [38, 42, 43]. This is another reason that makes a system to be a SLS.

Due to these reasons, there has been increasing interest in the study of SLSs over the past a few decades. To apply model-based control methods, building a faithful model is particularly important. The main difficulty lies in the classification of the observed input/output (I/O) data into different modes. For linear system identification, the I/O data are collected from one dynamic system; however, for the identification of SLSs, the obtained data are actually a mixture of data with respect to (w.r.t.) different modes. If the switching time is known, then the data classification is straightforward—we can identify each mode independently, using the identification approaches for linear systems. However, in most cases, switching time is not available. In such situations, the identification of SLSs becomes a nontrivial task.

1.2 Identification of SLSs

The aim of the PhD thesis is centered around the development of new identification approaches for SLSs. In this section, we present the problem details about the identification of SLSs.

In our research, we focus on the parametric identification approaches, which are based on structural models. The following shows a family of commonly-used model structures in discrete-time transfer function formats:

- ARX (AutoRegresive with eXogenous inputs) model:

$$\begin{aligned} A(z)y_k &= B(z)u_k + v_k \\ A(z) &\triangleq 1 + a_1 z^{-1} + \cdots + a_{n_a} z^{-n_a} \\ B(z) &\triangleq b_1 z^{-1} + \cdots + b_{n_b} z^{-n_b} \end{aligned}$$

where u_k, y_k and v_k are the input, output and measurement noise at time k, respectively; z^{-1} denotes the unit delay operator. The model orders of $A(z)$ and $B(z)$ are n_a and n_b, respectively.

- OE (Output Error) model:

$$y_k = \frac{B(z)}{A(z)}u_k + v_k$$

- ARMAX (AutoRegressive-Moving-Average with eXogenous inputs) model:

$$\begin{aligned} A(z)y_k &= B(z)u_k + C(z)v_k \\ C(z) &\triangleq c_1 z^{-1} + \cdots + c_{n_c} z^{-n_c} \end{aligned}$$

- BJ (Box-Jenkins) model:

$$\begin{aligned} y_k &= \frac{B(z)}{A(z)}u_k + \frac{C(z)}{D(z)}v_k \\ D(z) &\triangleq 1 + d_1 z^{-1} + \cdots + d_{n_d} z^{-n_d} \end{aligned}$$

The complexity of the model structures increases with the above order. As the model structure does not change the difficulty of data classification, for simplicity, we choose the ARX model for our research. (Actually, most existing work in this area was also concerned with the ARX model.)

Now, let us discuss two popular kinds of ARX models that are usually used to model the single-input and single-output (SISO) SLSs: the *Switched ARX* (SARX) model and the *Piecewise ARX* (PWARX) model. They can be described in the following forms:

- SARX model:

$$\begin{aligned} & y_k = \phi_k^T \theta_0^{\sigma_k} + v_k, \quad \sigma_k \in \{1, 2, \ldots, s\}, \\ & \phi_k \triangleq [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n, \\ & \theta_0^{\sigma_k} \triangleq [a_1^{\sigma_k} \ \cdots \ a_{n_a}^{\sigma_k} \ b_1^{\sigma_k} \ \cdots \ b_{n_b}^{\sigma_k}]^T \in \mathbb{R}^n. \end{aligned}$$

where σ_k is the switching signal or mode number; s denotes the number of modes; ϕ_k denotes the regression vector and $\theta_0^{\sigma_k}$ the parameter vector; $n = n_a + n_b$.

- PWARX model:

$$y_k = \phi_k^T \theta_0^{\sigma_k} + v_k, \quad \sigma_k \in \{1, 2, \ldots, s\},$$
$$\sigma_k = i, \text{ iff } \quad \phi_k \in \mathcal{R}_i, \quad i = 1, \ldots, s,$$

  where $\{\mathcal{R}_i\}_{i=1}^s$ is a complete partition of the regressor domain $\mathcal{R} \subseteq \mathbb{R}^n$. $\mathcal{R}_i$ is assumed to be a convex polyhedron.

The difference of the two models is on the switching rule. For the SARX model, it allows arbitrary switching among modes and the switching may be time- or event-triggered. For the PWARX model, we assume that the mode switching is dependent on a polyhedral partition of the regressor domain. To some extent, we may treat the PWARX model as a special class of the SARX model. Fig. 1.3 shows the data planes of a three-mode PWARX model. In

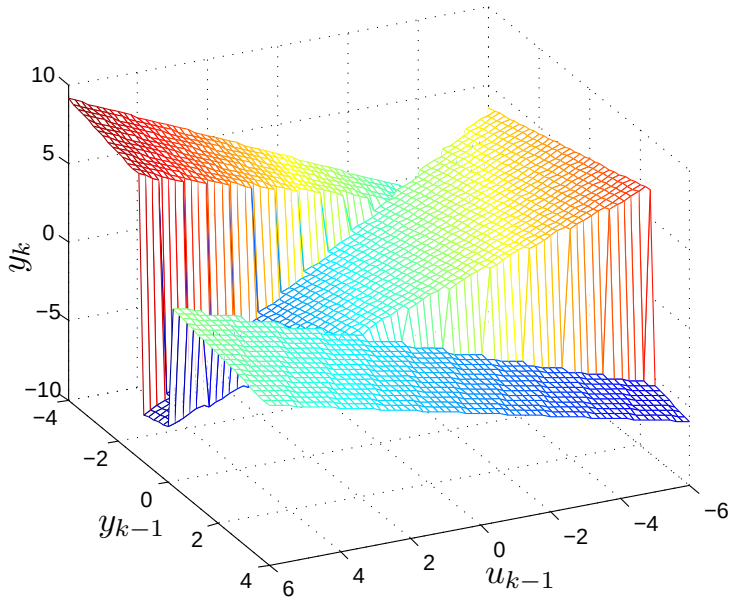


Figure 1.3: The data planes of a three-mode SLS

noise-free case, the input and output data pairs are exactly distributed on the three planes.

Then, regarding to the SARX model, the identification problem can be addressed as follows:

> *Given a collection of N input and output data pairs $\{(\phi_k, y_k)\}_{k=1}^N$, how to estimate the number of modes s, the model parameters $\{\theta_0^i\}_{i=1}^s$, the switching signal σ_k, and the model orders n_a and n_b?*

For the PWARX model, we need to additionally estimate a complete partition of the regressor domain except the above tasks.

This identification problem is highly difficult to solve, because there are too many uncertainties. For example, model orders may be known or unknown. In this thesis, for the sake of simplicity, we assume that model orders are given. In fact, when model orders are unavailable, we are still able to solve the identification problem using methods developed in this thesis by setting an upper bound for the model orders. In reference [67], there is a detailed discussion.

Note that, even without the uncertainty on model orders, the complexity of the identification problem is still high. They are closely related to two ingredients: the number of modes and the switching signal. From the perspective of minimizing prediction errors, here we discuss three different scenarios.

- When s is given and σ_k is known, the parameters of the SARX model can be easily inferred from a series of optimization problems like below:

$$\begin{aligned} \min_{\theta^i} \quad & \sum \ell(y_k - \phi_k^T \theta^i) \\ \text{s.t.} \quad & \sigma_k = i \end{aligned}$$

  where $\ell(\cdot)$ is a penalty function, e.g., $\|\cdot\|_2$.

- When s is given and σ_k is unknown, the identification problem can be posted as a mixed integer program:

$$\begin{aligned} \min_{\theta^i, w_k^i} \quad & \sum_{k=\bar{n}}^{N} \sum_{i=1}^{s} \ell(y_k - \phi_k^T \theta^i) w_k^i \\ \text{s.t.} \quad & \sum_{i=1}^{s} w_k^i = 1 \quad \forall k \\ & w_k^i \in \{0,1\} \quad \forall k, i \end{aligned}$$

  where $\bar{n} = \max\{n_a, n_b\} + 1$. w_k^i is a binary signal that is used to classify data. Compared with the problem in the first scenario, the mixed integer problem is more difficult to solve and it is computationally intractable if N is large.

- When both s and σ_k are unknown, the identification problem is in a more complicated form. We have to trade off the number of modes as well as the sum of prediction errors. Without a constraint on s, an over-fit problem may occur [13]. This thesis will mainly discuss this scenario.

In the next section, a brief review of the existing research work will be given.

1.3 Literature review

With reference to the early work on the identification of SLSs, we have seen that a large amount of research appeared in recent years, see, e.g., [6, 12, 13, 24–26, 36, 37, 48, 51, 52, 57, 66–68, 73]. The existing work can be broadly divided into offline approaches and online approaches.

1.3.1 Offline identification

The tutorial paper [52] gives us an excellent survey of the offline approaches published before the year 2007. The main approaches are the clustering-based approach [26], the bounded-error approach [12, 13], the Bayesian approach [36] and the algebraic approach [46, 68]. The technical details are briefly provided as below.

The idea behind the clustering-based approach [26] is based on the fact that the data in $\{(\phi_k, y_k)\}_{k=1}^N$ are likely to belong to the same mode if they lie close to each other. The implementation is composed of four steps. Firstly, it creates a local data set $\mathcal{C}_k$ for each data (ϕ_k, y_k). Then, a feature vector is constructed based on $\mathcal{C}_k$ and (ϕ_k, y_k). Next, cluster the feature vectors in s clusters $\{\mathcal{D}_i\}_{i=1}^s$ using a K-means like algorithm. With the obtained data set $\mathcal{D}_i$, the parameters of each mode are subsequently computed. This approach requires the knowledge on model orders, the number of modes and the size of clusters.

The bounded-error approach [12, 13] is based on the assumption that $|v_k|$ is upper bounded by $\bar{v}$. With this assumption, the I/O data can be classified by Partitioning the set of N inequalities $|y_k - \phi_k^T\theta| < \bar{v},\ k = 1, \ldots, N$, into a MINimum number of Feasible Subsystems/Subsets (MIN PFS). Since the MIN FS problem is NP-hard and difficult to solve directly, in [13], the authors solved it in iterative steps. Firstly, one obtains the MAXimum Feasible Subsystem/Subset (MAX FS) by applying a thermal relaxation based greedy algorithm. Then, remove this part of data from the whole data set and repeat these steps until no data is left. This approach requires the knowledge on model orders, $\bar{v}$ and a few tuning parameters.

The Bayesian approach [36] was developed from the statistic perspective. The idea is to treat $\{\hat{\theta}_0^i\}_{i=1}^s$ as random variables and pose the data classification as the problem of finding the classification with the highest probability. The procedure is as follows: firstly, compute the most probable mode σ_k of (y_k, ϕ_k), using the available PDFs (probability density function) of the parameter vectors from step $k-1$; then, attribute data to mode σ_k and update a posteriori PDF of $\hat{\theta}^{\sigma_k}$. The required knowledge includes the model orders, number of modes, PDFs of initial parameter vectors.

In the algebraic approach [46, 68], the authors constructed a high-dimensional lifted dynamical model to bypass the data classification problem. It was inspired from the following observation:

$$\prod_{i=1}^{s}(b_i^T\mathbf{z}_k) = \sum h_{s_1,\ldots,s_K} z_1^{s_1}\cdots z_K^{s_K} = \mathbf{h}^T v_s(\mathbf{z}_k) = 0 \quad (\text{when } v_k = 0),$$

where $b_i = \begin{bmatrix}1 & -(\theta_0^i)^T\end{bmatrix}^T$, $\mathbf{z}_k = \begin{bmatrix}-y_k & \phi_k^T\end{bmatrix}^T$, $h_{s_1,\ldots,s_K}$ is the coefficient of the monomial $z_1^{s_1}\cdots z_K^{s_K}$, $K = n_a + n_b + 1$ and $\sum_{i=1}^K s_i = s$. $\mathbf{h}$ is a vector with the dimension $\binom{s+K-1}{s}$, which contains the parameters of the lifted dynamic model. The approach is implemented in three steps: firstly, it creates the vectors $\{v_s(\mathbf{z}_k)\}_{k=1}^N$ using the above equation; then, it computes the parameters in $\mathbf{h}$; finally, it reconstructs $\{b_i\}_{i=1}^s$ from $\mathbf{h}$. The required knowledge is on the number of modes and model orders.

A detailed comparison of the four approaches was conducted in [35], which pointed out the advantages and disadvantages of them. In more recent years, a number of new results have emerged.

In [34], a EM (Expectation-Maximization) algorithm-based approach was developed from the work in [48]. It adopts the contaminated Gaussian distribution to construct the objective function of the maximization step. The advantage is to provide a robust estimation that is less sensitive to outliers or mis-classified data.

In [5, 51], the authors proposed a sparse optimization-based approach, which is similar as the bounded-error approach. The difference lies in the way to solve the MAX FS problem. The idea of this approach is obtained from

the observation on the following vector:

$$\Phi(\theta) \triangleq \begin{bmatrix} y_1 & -\phi_1^T \\ \vdots & \vdots \\ y_N & -\phi_N^T \end{bmatrix} \begin{bmatrix} 1 \\ \theta \end{bmatrix} \in \mathbb{R}^N,$$

which is a sparse vector when $v_k = 0$. The zero entries of $\Phi(\theta)$ correspond to the data generated by mode i. Then, the MAX FS can be formulated as an ℓ_0 minimization problem: $\min_{\theta} \|\Phi(\theta)\|_0$, where $\|\cdot\|_0$ indicates the number of non-zero entries of "$\cdot$".

Other research approaches about the offline identification are mostly like an extension or combination of the above approaches. For example, [24] shows a robust version of the algebraic approach and [39] gives a mixed approach in the algebraic and the bounded-error frameworks. They are omitted here for brevity.

1.3.2 Online identification

Online identification refers to the run of identification approaches in real time, which is very important in adaptive control systems. Compared with the offline identification, data classification in the online manner is more difficult, as the I/O data are collected sequentially and some useful data manipulations, e.g., swap of data order, iterative processing of data, are not applicable. As a result, the approaches of online identification are relatively limited.

In the literature, only a few papers have been found. In [67], the authors proposed a recursive algebraic approach and studied the exponential convergence in difference situations. The approach is well suited for the cases when the measurement noise is low and the physical system can be accurately represented as a SLS. In [6], a forgetting factor based recursive least squares algorithm was employed, where data classification was performed with a distance based mode decision function. This method is effective for the cases where the initial parameters can be well generated with *a priori* system knowledge. Other recent methods are optimization oriented. In [37], a gradient descent algorithm was described, which aimed at minimizing the sum of squared residuals. In [73], a kernel based weighted least squares estimator was proposed from a nonlinear identification perspective.

Note that our research is focused on SISO systems. For the approaches on MIMO systems, they are not reported here. The interested reader is referred

to [7–10, 14, 15, 53, 66] for more information.

1.4 Outline of the thesis

The remaining of the thesis is organized as follows:

- In Chapter 2, we propose a relaxed heuristic (RH) solution to the set membership identification problem of SLSs. By using the new solution, identification procedures become highly efficient and easy-to-implement. Moreover, to guard against data misclassification, we integrate a fast least trimmed squares (LTS) estimator to improve the accuracy of estimation. The performance of the proposed solution is evaluated based on both randomly generated systems and a continuous stirred tank reactor.

- In Chapter 3, we study identification of SLSs when the measurement noise has multiple unknown noise levels. In such situation, the noise level of each mode or subsystem is difficult to be captured by the existing identification methods. Therefore, the performance of parameter estimation may be degraded significantly. To offset this effect, we develop a new set membership identification method by integrating the RH solution with a forward search (FS) approach. Using the RH-FS method, the model parameters can be more accurately estimated.

- In Chapter 4, we discuss the clustering-based online identification approach. In this kind of approaches, a real time mode detection procedure is usually employed to estimate σ_k. During this process, mis-classified data points are inevitably involved. To reduce the negative effects of mis-classified data, we propose a recursive least squares algorithm with a resetting strategy, which is developed from a compensation point of view.

- In Chapter 5, we study the well-known Hough transform (HT) technique and discusses its applicability in the online identification of SLSs. By integrating the HT technique with an online clustering based estimator, we make the proposed algorithm applicable for the cases with unknown number of modes. Moreover, it can also be applied to the identification of time-varying SLSs or some nonlinear systems.

- Chapter 6 is concerned with parameter estimation of a special class of SLSs, namely, periodically switched linear systems (PSLSs). General identification methods that do not explore switching sequence patterns may perform poorly in estimation accuracy and implementation efficiency. In this chapter, we first analyze I/O data sequences and then establish the connection between the periodicity and I/O data sequence. This allows us to obtain an accurate estimation of the switching sequence period. We prove that the correctness of data classification is almost surely guaranteed. In implementation, we propose two efficient strategies, namely, the reverse order search and finite data selection, to improve the computational efficiency. Moreover, we provide both offline and online methods to estimate the period and the parameters.

- Chapter 7 gives a conclusion of the PhD work and presents a number of important research issues for future study.

Chapter 2

Set Membership Identification of SLSs: An RH Solution

Starting from this chapter, we aim at developing new identification methods for SLSs with the number of modes and switching signal being both unknown.

In this chapter, we propose a new solution, namely, the relaxed heuristic (RH) solution, to the set membership identification of SLSs. The RH solution has several advantages: (1) it does not require repeated manipulations of the I/O data; (2) it admits a simple and fast implementation, which makes the computational complexity significantly low; (3) it has a high level of robustness with the help of a fast least trimmed squares (LTS) estimator. All these make the proposed identification method efficient and effective.

The remaining of this chapter is organized as follows. Section 2.1 provides an introduction of the set membership identification and its usage for SLSs. Section 2.2 presents the formulation of the concerned problem. Section 2.3 describes the development of the RH solution. Section 2.4 analyzes the convergence and consistency. Some post-processing procedures are provided in Section 2.5 and the performance of the proposed solution is evaluated and demonstrated in Section 2.6. A summary is given in Section 2.7.

2.1 Introduction

Set membership identification is a well established approach, which has been extensively studied in the work on linear systems identification. It was often used to provide an uncertainty model set for robust control design and it is particularly suitable for the cases when one needs to fit the measured noisy data into an approximate model structure. However, for identification of SLSs, the motivation of applying set membership identification is different. We use it for the purpose of differentiating the data from different modes. The data points in the same membership set are viewed as the data w.r.t. the same mode, while the data points in different membership sets are believed from different modes. The size of the membership set is dependent on the noise level of each mode.

In Chapter 1, we have mentioned that the bounded error (BE) approach in [13] belongs to the set membership identification framework and it requires no *a priori* knowledge on the statistical property of noise except a noise bound. Compared with other offline approaches, it is well suited for identification of SLSs in the presence of noise. The key idea of the approach is to partition the system into a MIN PFS [2] and reformulate it as a sequence of MAX FS problems. To our best knowledge, there are two kinds of successful solutions in the literature. One is an improved Agmon-Motzkin-Schoenberg relaxation solution [12], where a thermal variant implementation was employed to solve the MAX FS problem iteratively. Some refined procedures were later introduced in [13]. The other is a sparse optimization solution based on sparse signal recovery, see [5, 51], which was initially developed in the community of compressive sensing. According to a comparison test of these two solutions in [51], each solution has both advantages and disadvantages: the BE solution is superior in the estimation efficiency; while the sparse optimization solution outperforms the other in the estimation accuracy. Since both solutions rely on a data clustering procedure, one common issue that we should notice is the robustness of estimation. Regarding the robust identification of SLSs, there are a few results available previously, see, e.g., [24, 34].

In the following sections, we develop an efficient and effective solution to the set membership identification of SLSs.

2.2 Problem description

We consider a discrete-time SLS described by the SARX model:

$$\begin{aligned} A^{\sigma_k}(z)y_k &= B^{\sigma_k}(z)u_k + v_k, \\ A^{\sigma_k}(z) &= 1 + a_1^{\sigma_k}z^{-1} + \cdots + a_{n_a}^{\sigma_k}z^{-n_a}, \\ B^{\sigma_k}(z) &= b_1^{\sigma_k}z^{-1} + b_2^{\sigma_k}z^{-2} + \cdots + b_{n_b}^{\sigma_k}z^{-n_b}, \end{aligned} \tag{2.1}$$

where u_k, y_k are respectively the sampled input and output at time k; v_k represents the measurement noise and/or modeling error. Its magnitude is assumed to be bounded by $\bar{v} \in \mathbb{R}^+$; σ_k is the switching signal or mode number that takes value from an index set $\{1, 2, \ldots, s\}$.

With the defined regression vector and the parameter vector,

$$\begin{aligned} \phi_k &\triangleq [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n, \\ \theta_0^{\sigma_k} &\triangleq [a_1^{\sigma_k} \ \cdots \ a_{n_a}^{\sigma_k} \ b_1^{\sigma_k} \ \cdots \ b_{n_b}^{\sigma_k}]^T \in \mathbb{R}^n, \end{aligned}$$

equation (2.1) can be rewritten in a compact form as below,

$$y_k = \phi_k^T\theta_0^{\sigma_k} + v_k. \tag{2.2}$$

In the following context of this chapter, σ_k may simply refer to the mode number without relation to time k. Thus, we will use σ instead of σ_k from now on.

It is seen that the collected I/O data in the whole data set $D \triangleq \{(\phi_k,\ y_k)\}_{k=1}^N$ are sampled from different modes. To classify these data, we denote D^σ as the data set that contains all I/O data points of mode σ, i.e.,

$$D^\sigma \triangleq \{(\phi_k,\ y_k) : y_k - \phi_k^T\theta_0^\sigma = v_k\}.$$

For a given data set D^σ, we can define a membership set S^σ of estimated parameter vectors that satisfy the constraints on the magnitude of residuals:

$$S^\sigma \triangleq \bigcap_{(\phi_k,\ y_k)\in D^\sigma} \{\theta \in \mathbb{R}^n : |y_k - \phi_k^T\theta| \le \bar{v}\}. \tag{2.3}$$

where $\bigcap$ denotes the intersection of a collection of sets. We note that θ_0^σ belongs to the above membership set and it can be estimated by using any appropriate estimator, e.g., ℓ_p-projection estimator with $p = 1, 2$ or ∞ [47].

However, the challenge here is how to obtain S^σ for each mode. In most practical applications, for SLSs, D^σ is not available and the membership set of D is typically empty, i.e.,

$$\bigcap_{(\phi_k,\ y_k)\in D} \{\theta \in \mathbb{R}^n : |y_k - \phi_k^T\theta| \leq \bar{v}\} = \emptyset.$$

Therefore, there does not exist a θ, satisfying

$$|\mathbf{y} - \Phi\theta| \leq \bar{\mathbf{v}},$$

where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \in \mathbb{R}^N,\ \Phi = \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_N^T \end{bmatrix} \in \mathbb{R}^{N\times n},\ \bar{\mathbf{v}} = \begin{bmatrix} \bar{v} \\ \bar{v} \\ \vdots \\ \bar{v} \end{bmatrix} \in \mathbb{R}^N.$$

In other words, the following linear system is infeasible,

$$\Sigma:\ \underbrace{\begin{bmatrix} \Phi \\ -\Phi \end{bmatrix}}_{A}\theta \leq \underbrace{\begin{bmatrix} \mathbf{y}+\bar{\mathbf{v}} \\ -\mathbf{y}+\bar{\mathbf{v}} \end{bmatrix}}_{\mathbf{b}}.$$

To find S^σ, we need an approximation of D^σ that gives a nonempty membership set. A solution is to partition the infeasible system, $\Sigma : \{A\theta \leq \mathbf{b}\}$ with $A \in \mathbb{R}^{2N\times n}$ and $\mathbf{b} \in \mathbb{R}^{2N}$, into a minimum number of feasible subsystems. This is termed as the MIN PFS problem in [2], where the authors also suggested to partition the problem into a sequence of MAX FS problems that deal with one mode at a time and are easier to handle. The MAX FS problem can be stated as follows [19]:

Given an infeasible system, $\Sigma : \{A\theta \leq \mathbf{b}\}$ *with* $A \in \mathbb{R}^{2N\times n}$ *and* $\mathbf{b} \in \mathbb{R}^{2N}$*, find the feasible subsystem with a maximum cardinality.*

When a MAX FS problem is solved, the data set of the feasible subsystem returns an estimation of S^σ. Then, we delete this part of data from the entire data set and repeat this procedure to obtain the remaining membership sets. This is a typical way to solve the MIN PFS problem. Fig. 2.1 shows a sketch of this implementation, where the $\hat{X}$ denotes an estimate of X.

However, the MAX FS problem has been proven to be NP-hard and an exact solution is only possible for small size instances [20]. Therefore, in the present work we shall pursue an approximation-based approach to solve the MAX FS problem.

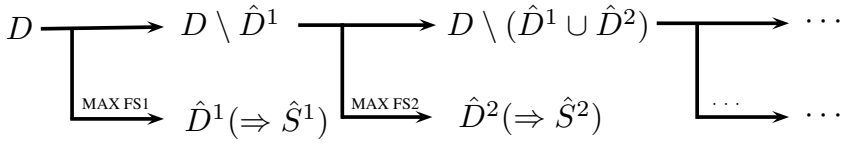


Figure 2.1: A typical way to solve a MIN PFS problem

2.3 The RH solution

Regarding the MAX FS problem, a variety of methods have been studied in the mathematical research community. A detailed description and discussion can be found in a recent book by Chinneck [20]. For this NP-hard problem, a heuristic method proposed in [19] is particularly effective, although it seems intractable for large size systems. Inspired by this method, we develop, in this section, the RH solution to solve the MAX FS problem.

2.3.1 Heuristics

Recalling that our objective is to find a feasible subsystem, so we first convert the infeasible system Σ into a feasible system Σ' by inserting $2N$ nonnegative elastic variables, $s_i \geq 0$ ($i = 1,\ 2,\ \ldots,\ 2N$), into Σ, shown as below:

$$\Sigma : \{A\theta \leq \mathbf{b}\} \longrightarrow \Sigma' : \{A\theta - \mathbf{s} \leq \mathbf{b}\}. \tag{2.4}$$

where $\mathbf{s} = \begin{bmatrix} s_1 & s_2 & \cdots & s_{2N} \end{bmatrix}^T \in \mathbb{R}^{2N}$. To achieve the "maximum" cardinality of $\hat{D}^\sigma$, the sum of the elastic variables are desired to be as small as possible. Therefore, this gives rise to the following constrained linear programming (LP) problem:

$$\begin{aligned} \min_{\theta,\ \mathbf{s}} \quad & J = \sum_{i=1}^{2N} s_i \\ \text{s.t.} \quad & A\theta - \mathbf{s} \leq \mathbf{b}, \\ & \mathbf{s} \geq 0, \end{aligned} \tag{2.5}$$

Notice that the inequalities in Σ' are contained in Σ only when all elastic variables are equal to zero. However, J is initially not zero and so we have to delete a number of inequalities to decrease the value of J. An efficient way, observed in [19], is to remove the inequality from Σ' that yields the largest drop in the objective function J, i.e.,

$$\textbf{Heuristic 1}: \ A \leftarrow [A],\ \mathbf{s} \leftarrow [\mathbf{s}],\ \mathbf{b} \leftarrow [\mathbf{b}],$$

where $[\cdot]$ is an operator that deletes the row of "$\cdot$" corresponding to the largest value in $\mathbf{s}$. After that, we resolve the above LP problem in (2.5) and apply the removal step again. The iteration stops when J is equal to zero. Fig. 2.2 describes the basic procedure of the Heuristic 1.

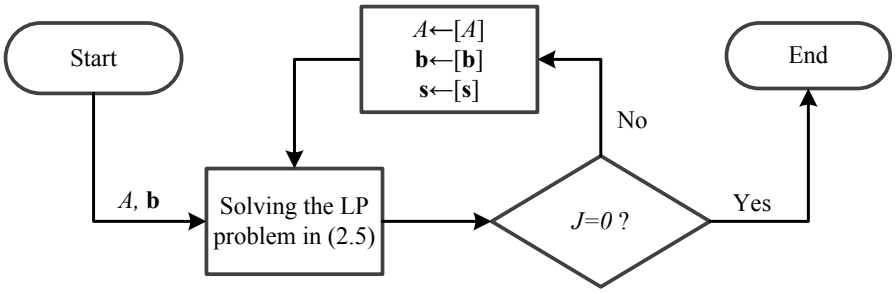


Figure 2.2: Solving the MAX FS problem using Heuristic 1

Remark 2.1. From Fig. 2.2, we observe that the structure of the implementation is very simple; only two steps are involved. Moreover, this heuristic methods requires no other parameters to be tuned except the noise bound. However, a drawback of this implementation lies in the deficiency on computational speed, which can be inspected from the dimension of the constraints in (2.5).

To make it obvious, we substitute A and $\mathbf{b}$. Then, the constraints can be written as

$$\begin{bmatrix} \begin{bmatrix} \Phi \\ -\Phi \end{bmatrix} & -\mathbf{I}_{2N\times 2N} \\ \mathbf{0}_{2N\times n} & -\mathbf{I}_{2N\times 2N} \end{bmatrix} \begin{bmatrix} \theta \\ \mathbf{s} \end{bmatrix} \leq \begin{bmatrix} \mathbf{y}+\bar{\mathbf{v}} \\ -\mathbf{y}+\bar{\mathbf{v}} \\ \mathbf{0}_{2N\times 1} \end{bmatrix}, \tag{2.6}$$

where $\mathbf{0}$ and $\mathbf{I}$ are respectively the zero and identity matrices with compatible dimensions. We see that the number of constraints is four times as large as the number of I/O data. Hence, it would be very expensive to compute the constrained LP problem at each iteration.

In fact, constraints in (2.6) also reflect that every data point introduces two *complementary inequalities*, namely, $|y_k - \phi_k^T\theta| \leq \bar{v}$, w.r.t. the noise bound and two additional inequalities w.r.t. elastic variables. If one data point is deleted, we should delete not only the inequality associated with the largest value in $\mathbf{s}$, but also the complementary inequality and their elastic variables

(say, s_m and s_n). Thus, we have

Heuristic 2 : if $\phi_k^T\theta - s_m \leq y_k + \bar{v}$ & $s_m \geq 0$ are deleted,
then $-\phi_k^T\theta - s_n \leq -y_k + \bar{v}$ & $s_n \geq 0$ should also be deleted.

Taking this into account, we are now able to reduce the problem size of (2.5). Let's start from an equivalent form of the concerned problem, that is

$$\begin{aligned} \min_{\theta,\ \mathbf{s}} \quad & J = \sum_{i=1}^{2N} s_i \\ \text{s.t.} \quad & s_i = \begin{cases} a_i\theta - b_i, & a_i\theta > b_i; \\ 0, & a_i\theta \leq b_i; \end{cases} \end{aligned} \tag{2.7}$$

where a_i and b_i are the i^{th} row of A and $\mathbf{b}$ respectively. Compared with the problem in (2.5), (2.7) has a different form, but it has the same optimal solution.

Property 2.1. *The problems in* (2.5) *and* (2.7) *have the same solution.*

Proof. For a given θ in problem (2.5), the elastic variable s_i can be directly determined by checking the inequality, $a_i\theta - s_i \leq b_i$, since s_i is independent on other elastic variables in both J and constraints. More precisely, if $a_i\theta \leq b_i$, s_i belongs to $[0,\ b_i - a_i\theta]$, for $a_i\theta - s_i \leq b_i$ to be valid; if $a_i\theta > b_i$, s_i belongs to $[a_i\theta - b_i, +\infty)$.

To achieve the minimum sum of s_i, s_i must be set to the representation form in (2.7). Therefore, the LP problems in (2.5) and (2.7) are equivalent and thereby have the same solution. □

Examining the elastic variables in (2.7), we can obtain the following property.

Property 2.2. *For a given θ and the pair of complementary inequalities in Heuristic 2, the sum of the corresponding elastic variables s_m and s_n is 0 if $|y_k - \phi_k^T\theta| \leq \bar{v}$, and is $|y_k - \phi_k^T\theta| - \bar{v}$ otherwise.*

Proof. Scenario 1: when $|y_k - \phi_k^T\theta| \leq \bar{v}$, we have $\phi_k^T\theta \leq y_k + \bar{v}$ and $-\phi_k^T\theta \leq -y_k + \bar{v}$. From (2.7), we see that s_m, s_n are both 0 and hence the sum of them is 0.

Scenario 2: when $|y_k - \phi_k^T\theta| > \bar{v}$, we may have $y_k - \phi_k^T\theta > \bar{v}$ or $y_k - \phi_k^T\theta < -\bar{v}$. When $y_k - \phi_k^T\theta > \bar{v}$, it implies that $\phi_k^T\theta < y_k - \bar{v} < y_k + \bar{v}$ and $-\phi_k^T\theta >$

$-y_k+\bar{v}$. Again, from (2.7), we obtain that $s_m = 0$ and $s_n = |y_k-\phi_k^T\theta|-\bar{v}$, hence the sum is $|y_k - \phi_k^T\theta| - \bar{v}$. When $y_k - \phi_k^T\theta < -\bar{v}$, it implies that $\phi_k^T\theta > y_k + \bar{v}$ and $-\phi_k^T\theta < -y_k - \bar{v} < -y_k + \bar{v}$. Similarly, we get $s_m = |y_k - \phi_k^T\theta| - \bar{v}$ and $s_n = 0$, hence the sum is also $|y_k - \phi_k^T\theta| - \bar{v}$. □

Based on Property 2.2, we may reduce the number of elastic variables by using one elastic variable for each pair of complementary inequalities; see the adapted problem form below:

$$\begin{aligned} &\min_{\theta,\ \mathbf{s}} \quad J = \sum_{i=1}^{N} s_i \\ &\text{s.t.} \quad s_i = \begin{cases} |y_i - \phi_i^T\theta| - \bar{v}, & |y_i - \phi_i^T\theta| > \bar{v}; \\ 0, & |y_i - \phi_i^T\theta| \le \bar{v}. \end{cases} \end{aligned} \tag{2.8}$$

Here, since the complementary inequalities are connected, they can be erased simultaneously.

Remark 2.2. In the above problem, the number of constraints is largely reduced and related constraints can be removed at a time. Therefore, (2.8) can be solved more efficiently than the other LP problems that we posed before. However, if a large data set is in use, the performance on speed may be still unsatisfactory.

2.3.2 Relaxation strategy

To further reduce the computational load, we consider a relaxation strategy that converts the problem in (2.8) into an unconstrained quadratic programming (QP) problem. It can be realized in two steps.

Firstly, we make the problem unconstrained. In (2.8), we separate the differences of $|y_i - \phi_i^T\theta| - \bar{v}$ into a positive part ($|y_i - \phi_i^T\theta| > \bar{v}$) and a negative part ($|y_i - \phi_i^T\theta| \le \bar{v}$); we add up all the positive differences and make the sum of them to be zero. This is a natural idea to retrieve a feasible subsystem. In fact, it can also be implemented in another way without involving any constraints: we consider the positive differences together with the negative ones and focus on the maximum value of $|y_i - \phi_i^T\theta|$; once it is less than $\bar{v}$, the obtained system becomes a feasible system. In this implementation, all elastic variables can be ignored; hence, no constraint is needed.

Secondly, we choose the least squares estimator, namely, ℓ_2-projection estimator, to compute parameters, because it provides analytic solutions.

After the relaxation, we can change problem (2.8) into

$$\min_{\theta} \quad J = \sum_{i=1}^{N} (y_i - \phi_i^T \theta)^2 = \|\mathbf{y} - \Phi\theta\|_2^2. \tag{2.9}$$

The analytic solution is simply the least square fit, namely, $(\Phi^T\Phi)^{-1}\Phi^T\mathbf{y}$. The proposed RH solution is summarized in Algorithm 1. With a bit abuse of notation, we denote by $[\cdot]$ an operator that deletes the row of " $\cdot$ " that corresponds to $\|\mathbf{y} - \Phi\hat{\theta}^\sigma\|_\infty$.

Algorithm 1 The RH solution

procedure RELAXEDHEURISTIC($\mathbf{y},\ \Phi$)
  $\hat{\theta}^\sigma \leftarrow (\Phi^T\Phi)^{-1}\Phi^T\mathbf{y}$
  while $\|\mathbf{y} - \Phi\hat{\theta}^\sigma\|_\infty > \bar{v}$ **do**
    $\mathbf{y} \leftarrow [\mathbf{y}]$
    $\Phi \leftarrow [\Phi]$
    $\hat{\theta}^\sigma \leftarrow (\Phi^T\Phi)^{-1}\Phi^T\mathbf{y}$
  end while
  $\hat{D}^\sigma \leftarrow \{(\Phi,\ \mathbf{y})\}$
  return $\hat{D}^\sigma,\ \hat{\theta}^\sigma$
end procedure

Example 2.1.

To compare the effectiveness of the above discussed heuristic solutions, the following SARX model is considered.

$$\text{Mode1} : A^1(z) = 1 + 0.1z^{-1} + 0.3z^{-2},\ B^1(z) = 4z^{-1} + 1.5z^{-2};$$
$$\text{Mode2} : A^2(z) = 1 - 0.2z^{-1} + 0.5z^{-2},\ B^2(z) = 2z^{-1} + 5z^{-2};$$

where Mode 1 and Mode 2 have 30 and 68 data points, respectively. The signal to noise ratio (SNR) is about 20dB and $\bar{v}$ is set to be 0.95. Table 2.1 shows the estimation results by the following heuristic solutions:

- H1: using Heuristic 1 only, based on problem (2.5);
- H1&2: using Heuristics 1 & 2, based on problem (2.8);
- RH: using the relaxed heuristic solution in Algorithm 1.

Table 2.1 indicates that the introduction of Heuristic 2 can both reduce the computational time and improve the estimation accuracy. It reflects that the

Table 2.1: Performance of estimation using different heuristic solutions

	H1	H1&2	RH
NormErr[a](%)	2.46	0.33	0.33
Time (second)	4.9688	2.5156	≈ 0

[a] NormErr $\triangleq \|\hat{\theta}^\sigma - \theta_0^\sigma\|_2 / \|\theta_0^\sigma\|_2 \times 100\%$, where $\sigma = 2$.

elimination of complementary inequalities is of great importance. Moreover, owing to the use of analytic solutions, we see that RH solution can further improve the computational efficiency to a large extent while keeps the same accuracy as H1&2.

2.3.3 Discussion: steepest descent in $\|\mathbf{y} - \Phi\hat{\theta}^\sigma\|_\infty$

In Algorithm 1, the RH solution is iteratively implemented until the largest magnitude of residual (LMR), i.e., $\|\mathbf{y} - \Phi\hat{\theta}^\sigma\|_\infty$, reaches a noise bound. As the number of iterations directly decides the size of MAX FS, we desire a fast decrease in LMR to reduce the number of data being removed. Deleting the data w.r.t. the LMR, as seen in Algorithm 1 (option I), is one option to do this.

An alternative option is to delete the data that makes the steepest descent in $\|\mathbf{y}-\Phi\hat{\theta}^\sigma\|_\infty$ at each iteration (option II). Let $(\phi_i,\ y_i)$ indicates the candidate of data to be deleted; $\Phi_{(i)},\ \mathbf{y}_{(i)}$ and $\hat{\theta}^\sigma_{(i)}$ indicate the updated terms after deleting $(\phi_i,\ y_i)$. Then, the index of the data to be deleted can be decided by

$$\begin{aligned} i_{sd} &= \arg\min_i \|\mathbf{y}_{(i)} - \Phi_{(i)}\hat{\theta}^\sigma_{(i)}\|_\infty \\ &= \arg\min_i \| \left[\mathbf{I} - \Phi_{(i)}(\Phi_{(i)}^T\Phi_{(i)})^{-1}\Phi_{(i)}^T\right] \mathbf{y}_{(i)}\|_\infty \ , \end{aligned}$$

where $(\Phi_{(i)}^T\Phi_{(i)})^{-1} = (\Phi^T\Phi - \phi_i\phi_i^T)^{-1}$ and can be computed using the matrix inversion lemma in [32]. More precisely, we have

$$(\Phi_{(i)}^T\Phi_{(i)})^{-1} = (\Phi^T\Phi)^{-1} + \frac{(\Phi^T\Phi)^{-1}\phi_i\phi_i^T(\Phi^T\Phi)^{-1}}{1 - \phi_i^T(\Phi^T\Phi)^{-1}\phi_i}.$$

Using this way to delete data, we may keep Algorithm 1 unchanged, but need to redefine $[\cdot]$ as the operator that deletes the i_{sd}^{th} row of " $\cdot$ ".

Remark 2.3. Compared with option I, the second option offers more LMR reduction at each iteration and may give better estimation. However, with

consideration of the computational complexity, it is not recommended when the number of I/O data is large.

2.4 Convergence and consistency

In this section, the convergence and consistency of the proposed RH solution will be analyzed based on option I. For the data set $\hat{D}^\sigma$, we define the mean energy of all residuals as below:

$$E_{\text{mean}} \triangleq \frac{1}{|\hat{D}^\sigma|} \sum_{(\phi_i,\ y_i)\in\hat{D}^\sigma} (y_i - \phi_i^T \hat{\theta}^\sigma)^2, \tag{2.10}$$

where $|\cdot|$ indicates the cardinality of "$\cdot$". According to the above definition, the optimization problem in (2.9) can be viewed as the minimization of mean energy.

Property 2.3. *During the iteration steps in the RH solution, E_{mean} is monotonically decreasing.*

Proof. Let $\hat{D}^\sigma(k)$, $E_{\text{mean}}(k)$ and $\hat{\theta}^\sigma(k)$ denote the estimated data set, mean energy and least squares fit at iteration k, respectively. $e_{\max}(k)$ is the largest squared residual at iteration k, or the energy corresponding to the data being deleted at iteration k. Firstly, we consider the sum of energy at iteration $k+1$. From the knowledge of least squares, we have

$$\sum_{(\phi_i,\ y_i)\in\hat{D}^\sigma(k+1)} [y_i - \phi_i^T \hat{\theta}^\sigma(k)]^2 \geq \sum_{(\phi_i,\ y_i)\in\hat{D}^\sigma(k+1)} [y_i - \phi_i^T \hat{\theta}^\sigma(k+1)]^2.$$

It implies that,

$$|\hat{D}^\sigma(k)| E_{\text{mean}}(k) - e_{\max}(k) \geq |\hat{D}^\sigma(k+1)| E_{\text{mean}}(k+1).$$

Since $|\hat{D}^\sigma(k+1)| = |\hat{D}^\sigma(k)| - 1$, the following inequality holds,

$$E_{\text{mean}}(k) - E_{\text{mean}}(k+1) \geq \frac{e_{\max}(k) - E_{\text{mean}}(k+1)}{|\hat{D}^\sigma(k)|}.$$

Then, from the fact below,

$$e_{\max}(k) \geq \frac{1}{|\hat{D}^\sigma(k+1)|} \sum_{(\phi_i,\ y_i)\in\hat{D}^\sigma(k+1)} [y_i - \phi_i^T \hat{\theta}^\sigma(k)]^2 \geq E_{\text{mean}}(k+1),$$

we conclude that $E_{\text{mean}}(k+1) \leq E_{\text{mean}}(k)$. □

Remark 2.4. As the data points associated with the LMR are deleted iteratively, a smaller LMR usually comes up with a smaller mean energy. In consequence, the RH solution stops in a finite iteration steps and thus the convergence is established.

Next, one should check the consistency of parameter estimation, or whether the obtained $\hat{\theta}^\sigma$ is close to θ_0^σ. Before that, let's consider the geometric meaning of the membership set, S^σ. From the definition in (2.3), we can see that it is actually a convex polytope in the space, which is the intersection of all hyperstrips, represented by

$$\{\theta \in \mathbb{R}^n : |y_k - \phi_k^T \theta| \leq \bar{v},\ (\phi_k,\ y_k) \in D^\sigma\}.$$

Using the proposed RH solution, the estimated parameter vector is possibly located in two kinds of polytopes. One is the polytope w.r.t. S^σ, in which $\|\hat{\theta}^\sigma - \theta_0^\sigma\|_2$ is small. The other is the polytope intersected by the hyperstrips with the data from different modes, which results in consistency failure.

To distinguish these two scenarios, we can examine the number of data in $\hat{D}^\sigma$. For consistency failure, $|\hat{D}^\sigma|$ is considerably less than the one in the first scenario. The reason is: when $\bar{v}$ is sufficiently small or the distance of modes, namely, $\|\theta_0^i - \theta_0^j\|_2$, is sufficiently large, the possibility of a hyperstrip that passes through an existing polytope (not the one w.r.t. S^σ) is nearly zero. We refer the reader to [70] for details.

In case of the consistency failure, we may temporarily remove $\hat{D}^\sigma$ from the D and repeat Algorithm 1. From our simulation study, we observe that it rarely occurs.

2.5 Post-processing

2.5.1 Removal of mis-classified data

Using set membership identification may introduce some mis-classified data, which can be equivalently viewed as outliers for a specific mode. To protect the estimation from corruption of the "outliers", we employ a robust estimator, namely, the least trimmed squares (LTS) estimator, to refine the RH solution. In what follows, we present a fast implementation, similar to [59].

In general, LTS optimizes the following problem,

$$\min_{\theta} \quad Q = \frac{1}{h}\sum_{i=1}^{h}(r^2)_{i:|\hat{D}^\sigma|}\ , \tag{2.11}$$

where h is a positive integer and $|\hat{D}^\sigma|/2 \leq h < |\hat{D}^\sigma|$; r is the residual belonging to $R = \{r : r = y_i - \phi_i^T\theta,\ (\phi_i,\ y_i) \in \hat{D}^\sigma\}$; $(r^2)_{i:|\hat{D}^\sigma|}$ is the ordered squared residual, satisfying

$$(r^2)_{1:|\hat{D}^\sigma|} \leq (r^2)_{2:|\hat{D}^\sigma|} \leq \cdots \leq (r^2)_{h:|\hat{D}^\sigma|} \leq \cdots \leq (r^2)_{|\hat{D}^\sigma|:|\hat{D}^\sigma|}. \tag{2.12}$$

To efficiently solve this problem, one may start from a pre-determined parameter vector, $\hat{\theta}_j^{\text{LTS}}$ with $j = 0$, and then do the following steps iteratively:

2-A1 Use $\hat{\theta}_j^{\text{LTS}}$ to obtain and sort the squared residuals as (2.12);

2-A2 Compute a least squares fit, $\hat{\theta}_{j+1}^{\text{LTS}}$, of the data w.r.t. $(r^2)_{i:|\hat{D}^\sigma|}$, $i = 1,\ 2,\ \ldots,\ h$.

As suggested in [59], the local optimum can be reached within a few iterations of 2-A1 and 2-A2. However, in order to achieve the global optimum, $\hat{\theta}_0^{\text{LTS}}$ has to be close to the true parameter vector, which may not be true if the data to produce $\hat{\theta}_0^{\text{LTS}}$ are not associated with the same mode.

Taking this into account, we need to repeat the preceding procedures a large number of times using different $\hat{\theta}_0^{\text{LTS}}$'s and then decide the optimal solution. For the initial data subsets to generate $\hat{\theta}_0^{\text{LTS}}$'s, they are made up by randomly picking a small amount of data and the data right after them. For example, if the data at positions 1, 5, 12, 28 are selected, then the data at positions 2, 6, 13, 29 are also included. Here, the position order is the same as the sampling order. The reason of doing so is based on the fact that consecutively sampled data points are highly possible to be associated with the same mode.

By adopting the LTS estimation, we can obtain a robust estimation, which is denoted by $\hat{\theta}^{\text{LTS}}$. Then, we apply the following step to remove the portion of mis-classified data.

2-A3 Use $\hat{\theta}^{\text{LTS}}$ to remove any data point that has its residual magnitude greater than $\bar{v}$; set $\hat{\theta}^\sigma \leftarrow \hat{\theta}^{\text{LTS}}$,

2.5.2 Recovery of mis-deleted data

During iterations in the RH solution, some data points associated with the estimated MAX FS are likely to be mis-deleted. To achieve the "maximum" of the feasible subsystem, we need to bring back this portion of data. The following steps can be taken iteratively:

2-B1 From the remaining data set, namely, $D \setminus \bigcup \hat{D}^\sigma$, find any data point (y', ϕ') that has its residual magnitude less than $\bar{v}$;

2-B2 Put the data point(s) (y', ϕ') into $\hat{D}^\sigma$, then update $\hat{\theta}^\sigma$.

Note that the post-processing procedures in this section should be implemented in alphabetic order and follow after the RH solution and the consistency checking.

2.6 Simulation results

This section gives two examples to show the effectiveness of the proposed solution.

Example 2.2.

We now consider a second-order SLS with three modes:

$$A^\sigma(z) = 1 + a_1^\sigma z^{-1} + a_2^\sigma z^{-2}, \quad \sigma \in \{1,\ 2,\ 3\},$$
$$B^\sigma(z) = b_1^\sigma z^{-1} + b_2^\sigma z^{-2},$$

where $A^\sigma(z)$ and $B^\sigma(z)$ are randomly generated and $A^\sigma(z)$ is a monic polynomial with all zeros located inside the unit circle. u_k and v_k are zero-mean white Gaussian noise signals with variance equal to 1 and υ_0^2, respectively.

In the simulation, we collect 900 I/O data for each trial. Modes 1, 2 and 3 have respectively 200, 300, and 400 data. Switching signal is randomly selected from $\{1,\ 2,\ 3\}$ at time k; $\bar{v} = 3\upsilon_0$ and $h = |\hat{D}^\sigma|/2$. The simulation is carried out on a computer with the following specifications: 2.00 GHz Intel Core 2 Duo processor and 1.00 GB 778 MHz RAM.

Firstly, let us see a typical run of the RH solution in Algorithm 1, where υ_0^2 is set to 2 (SNR≈20dB) and correspondingly $\bar{v}$ is equal to 4.243. Fig. 2.3 (a) shows the mode sequence of deleted data. It is seen that most data of modes 1 and 2 were removed from the data set. There are only a few data of mode

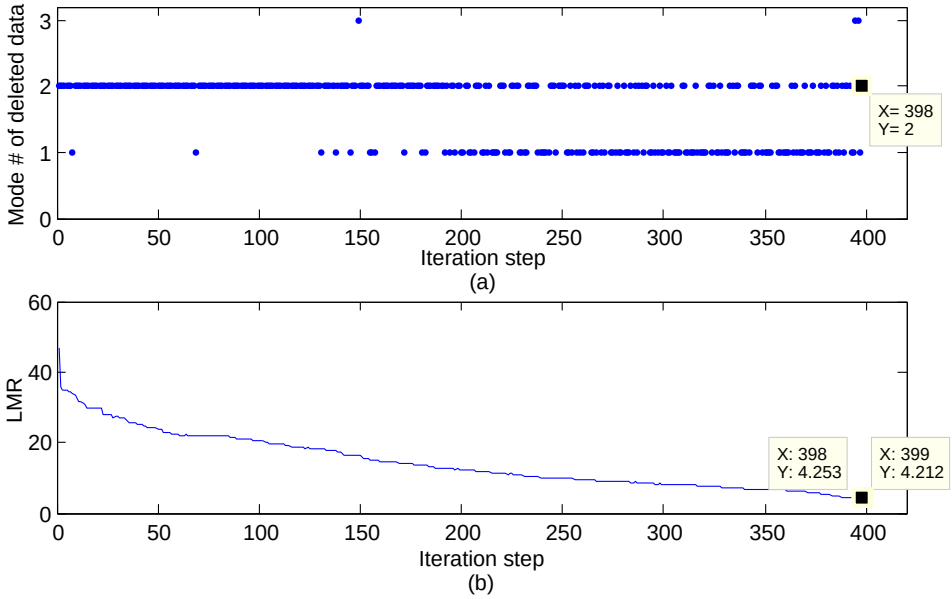


Figure 2.3: A typical run of the RH solution

3 that were mis-deleted. While looking at the LMR of each iteration, in Fig. 2.3 (b), we find that it decreases steadily and stops when the value is less than $\bar{v}$. Using the obtained data set, NormErr is computed, which is as low as 1.98%.

To illustrate the overall performance of estimation accuracy and efficiency, we then test the RH solution over 100 Monte Carlo simulations. That is, for different trials, u_k, v_k, $A^\sigma(z)$ and $B^\sigma(z)$ are independently generated. For each trial, the estimation accuracy is measured by "AveNormErr", defined as $\text{AveNormErr} \triangleq \frac{1}{\hat{s}} \sum_{\sigma=1}^{\hat{s}} \|\hat{\theta}^\sigma - \theta_0^\sigma\|_2 / \|\theta_0^\sigma\|_2 \times 100\%$.

Table 2.2: Estimation accuracy and efficiency of the RH solution

	AveNormErr (%)	Time (second)
$\upsilon_0^2 = 0.02$ ($\approx$ 40dB)	0.18 ± 0.11	0.1552 ± 0.0182
$\upsilon_0^2 = 0.2$ ($\approx$ 30dB)	0.73 ± 0.63	0.1448 ± 0.0193
$\upsilon_0^2 = 2$ ($\approx$ 20dB)	2.70 ± 2.64	0.1230 ± 0.0203

The simulation results, including the mean and standard deviation of AveNormErr and computation time, are given in Table 2.2. It is seen that the RH solution performs very well in both accuracy and efficiency. Moreover, the consistency failure was not observed in all these simulations.

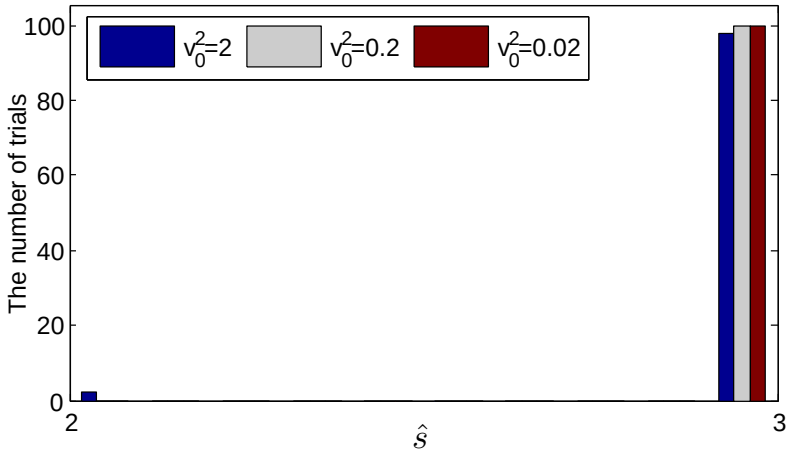


Figure 2.4: Histogram of $\hat{s}$ for the 100 Monte Carlo simulations

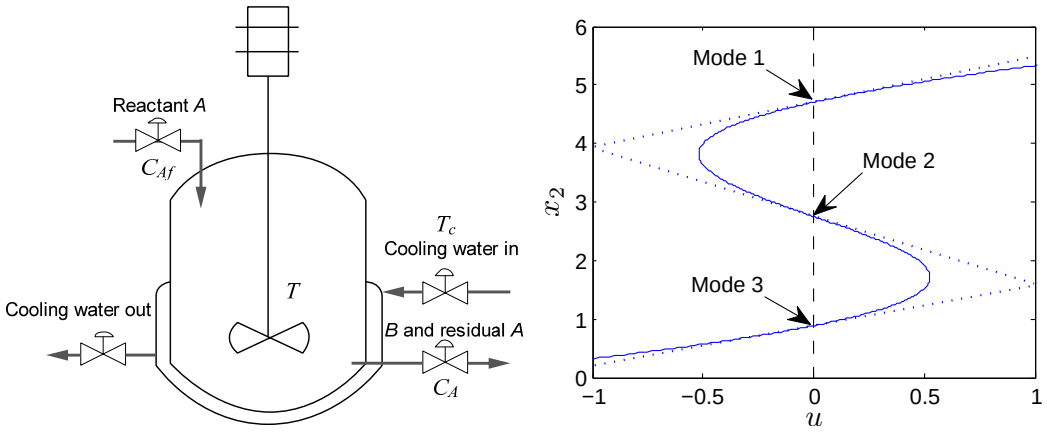
Finally, we check the number of estimated modes, $\hat{s}$. Fig. 2.4 shows the histogram of $\hat{s}$ for the 100 Monte Carlo simulations. In most cases, it matches exactly with its true value and only two trials get failed. It is because that the noise level in these cases (where υ_0^2=2) are large enough to make two modes indistinguishable. In such situations, there exists a nonempty membership set to cover all the data from both modes; therefore these two modes can be combined. However, to some extent, it simplifies the modeling complexity.

Example 2.3.

We consider the identification of a non-isothermal continuous stirred tank reactor (CSTR), which has an exothermic and irreversible reaction: $A \rightarrow B$. Fig. 2.5 (a) depicts a schematic of the CSTR. In this process, reactant A is continuously fed into the reactor at concentration C_{Af}; the reaction takes place inside the vessel at temperature T; the product B and residual A are continuously taken away from the reactor, where the concentration of A reduces to C_A. To remove excess heat from the reaction, cooling water with temperature T_c is flowing through the jacket at the same time. The dynamic model of the system, given in [64], can also be expressed in the following dimensionless form:

$$\begin{cases} \dot{x}_1 &= -x_1 + D_a(1-x_1)e^{\frac{x_2}{1+x_2/\gamma}}, \\ \dot{x}_2 &= -x_2 + BD_a(1-x_1)e^{\frac{x_2}{1+x_2/\gamma}} + \beta(u-x_2), \end{cases} \tag{2.13}$$

where $x_1 = \frac{C_{Af}-C_A}{C_{Af}}$, $x_2 = \frac{20(T-T_f)}{T_f}$, $D_a = 0.072$, $\gamma = 20$, $B = 8$ and $\beta = 0.3$.



(a) (b)

Figure 2.5: (a) a schematic of the CSTR (b) the steady state map

As known in [64], this nonlinear system exhibits the output multiplicity, which is illustrated in the steady state map, see Fig. 2.5 (b). Visually, the system can be roughly approximated by a SARX model with three linear modes (Modes 1-3). Since Mode 2 is unstable and further requires a close-loop approach for identification, we ignore this mode for simplicity and here we only consider the identification of Modes 1 and 3.

To identify the discrete time SARX model, we sample the input and output data from the system in (2.13) with sampling time equal to 20 seconds. The input u is a random signal that is uniformly sampled from $[-1,\ 1]$; the output y takes value of x_2 and is stacked in $\mathbf{y}$. Measurement "noise" is not added, but model mismatch (using linear model to approximate nonlinear system) gives rise to modeling errors, which can be viewed as additional noises. This system is simulated for 900 I/O data points; the first 600 data are used to estimate parameters and the rest are for model validation. When doing the validation test, the following measure is often applied to get the output fit, see, e.g., [5, 13],

$$\text{FIT} \triangleq (1 - \|\hat{\mathbf{y}} - \mathbf{y}_{601:900}\|_2 / \|\mathbf{y}_{601:900} - \text{mean}(\mathbf{y}_{601:900})\|_2) \times 100\%,$$

where $\hat{\mathbf{y}}$ and $\mathbf{y}_{601:900}$ indicate the predicted and true outputs, respectively; mean$(\cdot)$ is the mean value vector of "$\cdot$". Here, the mode number of the data for validation are estimated by: $\sigma_k = \underset{\sigma\in\{1,2,\ldots,\hat{s}\}}{\arg\min} \|\mathbf{y}_{(k)} - \Phi_{(k)}\hat{\theta}^{\sigma}\|_\infty$.

In the simulation, as $\bar{v}$ is unknown, we run the RH solution with different values. Fig. 2.6 shows the output fit and the number of estimated modes

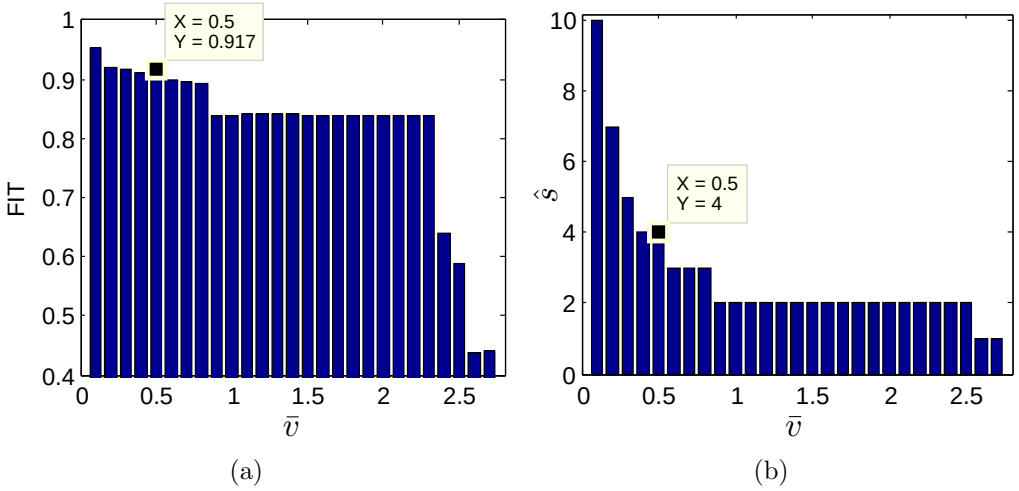


Figure 2.6: FIT and $\hat{s}$ for different $\bar{v}$'s

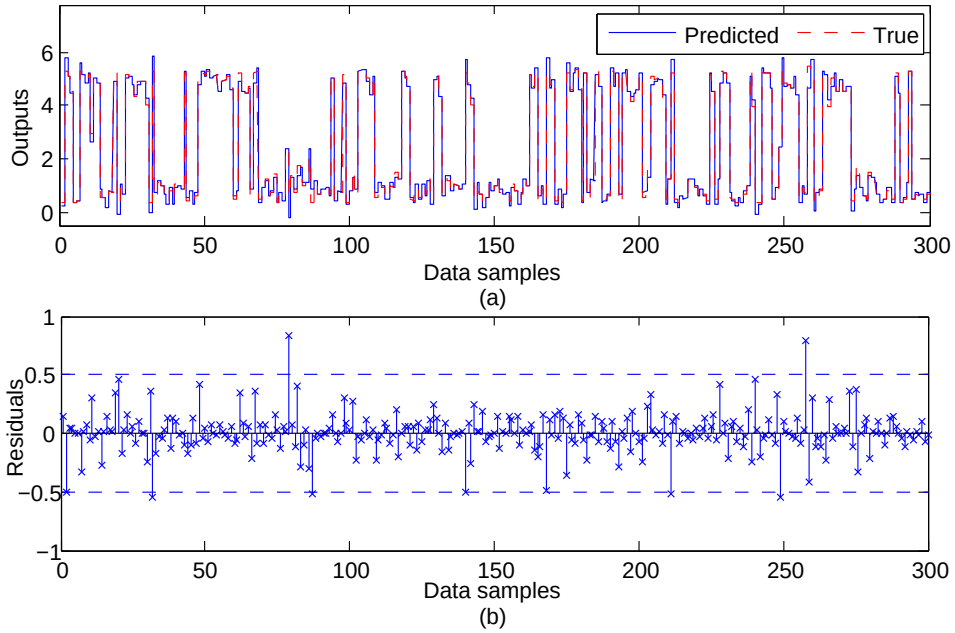


Figure 2.7: Model validation: (a) prediction of outputs (b) residuals of data for validation

w.r.t. different $\bar{v}$'s. We see the output fit is quite poor when $\bar{v} > 2.3$ or when $\hat{s}$ is one; but there is a significant increase in FIT as $\hat{s}$ grows up from 1 to 2. This is because the nonlinear system has two stable equilibrium points as we mentioned. If we further reduce $\bar{v}$, both FIT and $\hat{s}$ will go up. To make a trade-off between the model accuracy and model complexity, we may select $\bar{v}$ at the place where the curvature of the curve in Fig. 2.6 (b) is relatively large.

This noise bound estimation approach was suggested in [13]. In this example, $\bar{v}$ can be chosen to be 0.5, where $\hat{s} = 4$ and FIT = 91.7%. Fig. 2.7 shows the results of model validation. As we expected, the output fit is good and the residuals for the validation data are mostly located in the range of [-$\bar{v}$, $\bar{v}$].

2.7 Summary

In this Chapter, we developed an efficient and effective solution to the set membership identification of SLSs by using heuristics and relaxation techniques. The heuristics can help in solving the MAX FS problem with simply two-step iterations, which require few parameters to be tuned. In addition to being simple, we explored the problem structure and applied the relaxation strategy to reduce the computation complexity of the solution. As for misclassified data, they are treated as outliers and removed by applying the LTS estimator with a fast implementation.

We close by mentioning that the state of the art in set membership identification still stays in the pursuit of the optimal estimation for a specific mode. How to guarantee global optimality in consideration of all data and modes will be an interesting topic to be studied.

Chapter 3

Identification of SLSs with Multiple Unknown Noise Levels*

In Chapter 2, we have studied the set membership identification of SLSs and proposed the RH solution to the MAX FS problem. It was assumed that all linear modes have a common noise level. With this assumption, the noise bound can be readily estimated.

In this chapter, we try to remove this assumption and develop an identification method for SLSs with multiple unknown noise levels. The new method consists of the RH solution and a forward search (FS) method. The RH solution is mainly used for providing an approximate noise bound and an approximate data set for the MAX FS. It exhibits some appealing features and allows us to explore data characteristics more efficiently. The FS method plays the role of robust estimation as well as the selection of a proper noise bound. With the proposed RH-FS method, we can achieve a better estimation performance than the method in [13].

The remaining of this chapter is organized as follows: Section 3.1 provides the motivation of the study. Section 3.2 describes the problem to be solved. Section 3.3 revisits the RH solution and show the importance of a noise bound. Section 3.4 presents the proposed RH-FS method for robust estimation and gives the detailed implementation. The effectiveness of the proposed method is demonstrated in Section 3.5 and a summary is given in Section 3.6.

*A version of Chapter 3 has been submitted for publication in [71].

3.1 Motivation

As shown in Chapter 2, the set membership identification performs quite well for the identification of SLSs with noisy data. The proposed RH solution is a very good option to solve the MAX FS problem, although some practical issues may exist. One major issue that we can think of is on the noise bound, which was assumed to be given in Chapter 2. If the noise bound is not available, the RH solution stops working immediately. Therefore, it is of great importance to study the RH solution when the noise bound is unknown. In Example 2.3, we have provided an approach to estimate the noise bound, but it works well only when all modes of SLSs have the same noise level. In many real applications, different modes of SLSs may have different noise levels. A simple example is the modeling mismatch, where the true model of a mode is different from the user-defined model. Modeling errors will, therefore, be added up to the measurement errors, giving rise to different noise levels for different modes. How to estimate noise bounds in such systems was seldom investigated in the literature, despite its importance. This chapter is dedicated to solving this problem.

3.2 Problem description

Let us consider a compact form of the discrete-time SARX model:

$$\begin{aligned} & y_k = \phi_k^T \theta_0^\sigma + v_k^\sigma, \quad \sigma \in \{1, 2, \ldots, s\}, \\ & \phi_k \triangleq [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n, \\ & \theta_0^\sigma \triangleq [a_1^\sigma \ \cdots \ a_{n_a}^\sigma \ \ b_1^\sigma \ \cdots \ b_{n_b}^\sigma]^T \in \mathbb{R}^n, \end{aligned} \tag{3.1}$$

where ϕ_k and θ_0^σ represent the regression vector and the parameter vector, respectively; σ is the mode number (or switching signal) and it changes with time k; v_k^σ is assumed to be white Gaussian noise. For mode σ, the magnitude of v_k^σ is bounded by $\bar{v}^\sigma \in \mathbb{R}^+$.

Same as in Chapter 2, we denote the whole data set by $D \triangleq \{(\phi_k, \ y_k)\}_{k=1}^N$ and D^σ the data set for mode σ, i.e.,

$$D^\sigma \triangleq \{(\phi_k, \ y_k) : y_k - \phi_k^T \theta_0^\sigma = v_k^\sigma\}. \tag{3.2}$$

Then, the corresponding membership set can be written as

$$S^\sigma = \bigcap_{(y_k, \phi_k) \in D^\sigma} \{\theta \in \mathbb{R}^n : |y_k - \phi_k^T \theta| \leq \bar{v}^\sigma\}. \tag{3.3}$$

Clearly, it is seen that θ_0^σ belongs to the membership set and can be readily estimated if D^σ is available. However, we have no information on the switching signal and even $\bar{v}^\sigma$, which makes it very difficult to estimate D^σ.

Recall that, in Chapter 2, we estimate D^σ by solving the MAX FS problem using the RH solution with the noise bound. Here, although $\bar{v}^\sigma$ is not known, we may replace it with its estimate $\hat{\bar{v}}^\sigma$ and then solve the MAX FS problem. For convenience, we repeat the MAX FS problem as follows: given an infeasible system,

$$\Sigma: \quad \begin{bmatrix} \Phi \\ -\Phi \end{bmatrix} \theta \leq \begin{bmatrix} Y + \bar{V} \\ -Y + \bar{V} \end{bmatrix} \tag{3.4}$$

with

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \in \mathbb{R}^N, \ \Phi = \begin{bmatrix} \phi_1^T \\ \phi_2^T \\ \vdots \\ \phi_N^T \end{bmatrix} \in \mathbb{R}^{N \times n}, \ \bar{V} = \begin{bmatrix} \hat{\bar{v}}^\sigma \\ \hat{\bar{v}}^\sigma \\ \vdots \\ \hat{\bar{v}}^\sigma \end{bmatrix} \in \mathbb{R}^N,$$

find the feasible subsystem with a maximum cardinality.

Now, the key is how to get an appropriate $\hat{\bar{v}}^\sigma$ for each mode. In the following sections, we will show that $\bar{v}^\sigma$ is closely related to the quality of estimation and it can be well estimated by evaluating the residuals with the RH solution.

3.3 Revisit the RH solution

To estimate $\bar{v}^\sigma$, it is necessary to revisit the RH solution, which tells us important effects of the noise bound. From the previous chapter, we know that the RH solution consists of the following steps:

3-A1 Obtain $\hat{\theta}_t^\sigma = \arg\min_\theta \ \|Y_t - \Phi_t\theta\|_2^2$. (Initially, set $t = 1$, $Y_t \leftarrow Y$, $\Phi_t \leftarrow \Phi$.)

3-A2 Check $\|Y_t - \Phi_t\hat{\theta}_t^\sigma\|_\infty > \bar{v}^\sigma$? If yes, go to step 3-A3; otherwise, stop and return $\hat{\theta}_t^\sigma$ and $\hat{D}^\sigma$!

3-A3 Set $Y_t \leftarrow [Y_t]^\dagger$, $\Phi_t \leftarrow [\Phi_t]$ and $t \leftarrow t + 1$; then go to step 3-A1.

† $[\cdot]$ is an operator that deletes the row of "$\cdot$" w.r.t. the largest magnitude of residual (LMR), i.e., $\|Y_t - \Phi_t\hat{\theta}_t^\sigma\|_\infty$.

In the above RH solution, $\bar{v}^\sigma$ plays a very important role in eliminating the unnecessary data from the whole data set. When $\bar{v}^\sigma$ is not properly selected, the estimated data set may be significantly affected.

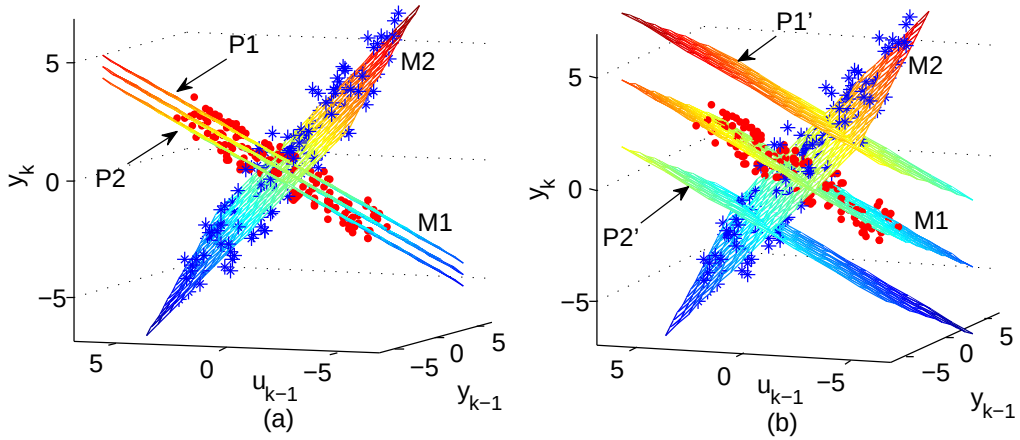


Figure 3.1: The impact of a noise bound: (a) tight bound (b) loose bound

Fig. 3.1 shows the data distribution of a 3-dimension SLS with two modes (M1 and M2), where P1 and P2 are the hyperplanes specified by $y = \phi^T\theta_0^1 \pm \bar{v}^1$; similarly, P1' and P2' are specified by $y = \phi^T\theta_0^1 \pm \bar{v}'^1$. We have $\bar{v}^1 < \bar{v}'^1$. The width of the hyperstrip between P1 and P2 is proportional to the value of $\bar{v}^1$. All data in the hyperstrip are included in $\hat{D}^1$. In Fig. 3.1 (a), as $\bar{v}^1$ is over tight, some data points from M1 are excluded from the hyperstrip. That means the number of data in $\hat{D}^1$ (denoted by $|\hat{D}^1|$) is less than the true number of data from M1 and thus the data information of M1 is not fully used. On the contrary, in Fig. 3.1 (b), $\bar{v}'^1$ is over loose and it causes a lot of mis-classified data (from M2) involved in $\hat{D}^1$, which may render the estimated parameters to be rather far from the true values.

Therefore, we need to be very careful on the selection of $\bar{v}^\sigma$ when it is unknown. In the literature, the authors in [13] provided a feasible method to estimate $\bar{v}^\sigma$ as we did in Example 2.3. The basic idea is to make a trade-off between the number of modes and fitting errors. However, it is restricted to the cases that all modes have a same noise level. If noise levels are different, it may lead to an over loose noise bound for some modes.

3.4 The RH-FS method

In Chapter 2, the RH solution was applied for the identification use. In fact, it can also be used for the estimation of $\bar{v}^\sigma$ with the help of the forward search (FS) method.

3.4.1 The RH solution without $\bar{v}^\sigma$

Since $\bar{v}^\sigma$ is to be estimated, here we only focus on the RH solution with steps 3-A1 and 3-A3. Before we proceed, it is worthwhile to point out some appealing features of using the RH solution (3-A1 and 3-A3) to estimate $\bar{v}^\sigma$. From the above discussion, we have seen that the noise bound is tightly related to the LMR; so we don't need to continuously scale $\bar{v}^\sigma$ and construct a large candidate set. Another advantage is that the LMR is decreasing steadily, so we can find an approximated $\bar{v}^\sigma$ with one-time-through of all data.

These appealing features benefit us to find $\bar{v}^\sigma$'s more efficiently. When iteratively running steps 3-A1 and 3-A3, we could stop at "somewhere" and see whether the LMR is suitable to be a proper noise bound (neither tight nor loose). During this process, there are two key points which need to be clarified. One is the way to check the tightness of a noise bound candidate; the other is the right time to stop and check. For the purpose of finding a good noise bound, we first need to explore the characteristics of the data w.r.t. a single mode.

Definition 3.1. *Given an estimated data set $\hat{D}^\sigma$, we call the data $(y_i, \phi_i) \in \hat{D}^\sigma$ are consistent, if there exists an estimated parameter vector $\hat{\theta}^\sigma$ such that the null hypothesis "$\mathcal{H}_0$: the residual $r_i = y_i - \phi_i^T \hat{\theta}^\sigma$ is normally distributed" can be accepted.*

Theorem 3.1. *Data consistency is a necessary condition to guarantee that all data $(y_i, \phi_i) \in \hat{D}^\sigma$ are w.r.t. a single mode.*

Proof. It follows from the fact that $y_i = \phi_i^T \theta_0^\sigma + v_i$ and v_i is normally distributed. □

Strictly speaking, data consistency is insufficient to conclude that data points, belonging to $\hat{D}^\sigma$, are all corresponding to a single mode. However, data consistency becomes a sufficient condition under some mild assumptions.

Theorem 3.2. *If the inputs in ϕ_k are persistently exciting (PE) of order n and $max\{\bar{v}^{\sigma'},\ \bar{v}^{\sigma}\}/\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2$ is sufficiently small, $\forall\ \sigma\neq\sigma'\in\{1,2,\ldots,s\}$, then the consistent data in $\hat{D}^{\sigma}$ are almost surely w.r.t. a single mode.*

Proof. We prove it by contradiction and we suppose that the data in $\hat{D}^{\sigma}$ are sampled from ≥ 2 modes. Without loss of generality, we consider a 2-mode case, assuming that one data pair, (y',ϕ'), is with mode σ' and the rest are with mode σ. As for mode σ, the equivalent noise term of (y',ϕ') can be written as

$$v_{\text{eq}}=\phi'^{T}(\theta_0^{\sigma'}-\theta_0^{\sigma})+v', \tag{3.5}$$

where $v'=y'-\phi'^{T}\theta_0^{\sigma'}$ and $|v'|<\bar{v}^{\sigma'}$. As we know, if $|v_{\text{eq}}|$ is apparently larger than $\bar{v}^{\sigma}$ (e.g., if $|v_{\text{eq}}|-\bar{v}^{\sigma}\geq\epsilon$), then it destroys the data consistency. Notice that ϵ is positively correlated with $\bar{v}^{\sigma}$, thus $(2\text{max}\{\bar{v}^{\sigma'},\ \bar{v}^{\sigma}\}+\epsilon)/\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2$ can be sufficiently small with the assumption in Theorem 3.2. Together with the PE condition, we have

$$\Pr\{\frac{|\phi'^{T}(\theta_0^{\sigma'}-\theta_0^{\sigma})|}{\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2}<\frac{2\text{max}\{\bar{v}^{\sigma'},\ \bar{v}^{\sigma}\}+\epsilon}{\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2}\}\approx 0, \tag{3.6}$$

where $\Pr\{X\}$ denotes the probability of event X; $\phi'^{T}(\theta_0^{\sigma'}-\theta_0^{\sigma})/\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2$ is the projection of ϕ' onto $\theta_0^{\sigma'}-\theta_0^{\sigma}$. Equation (3.6) implies that

$$\Pr\{|\phi'^{T}(\theta_0^{\sigma'}-\theta_0^{\sigma})|-|v'|<\bar{v}^{\sigma}+\epsilon\}\approx 0. \tag{3.7}$$

As $|v_{\text{eq}}|$ is no less than $|\phi'^{T}(\theta_0^{\sigma'}-\theta_0^{\sigma})|-|v'|$, we get $\Pr\{|v_{\text{eq}}|<\bar{v}^{\sigma}+\epsilon\}\approx 0$. It means the data consistency is violated with probability 1. Hence, Theorem 3.2 holds. □

Remark 3.1. In practical applications, the PE condition can be guaranteed by selecting a proper input sequence and $\text{max}\{\bar{v}^{\sigma'},\ \bar{v}^{\sigma}\}/\|\theta_0^{\sigma}-\theta_0^{\sigma'}\|_2$ is usually small enough for Theorem 3.2 being valid.

Based on the data consistency, we next find a well-selected noise bound that is defined as follows:

Definition 3.2. *If the data of $\hat{D}^{\sigma}$ are consistent and adding any one data will destroy the consistency, then*

$$\bar{v}_{ws}^{\sigma}=\max_{(y_i,\ \phi_i)\in\hat{D}^{\sigma}}|y_i-\phi_i^{T}(\sum\phi_i\phi_i^{T})^{-1}\sum\phi_i y_i| \tag{3.8}$$

is called the well-selected noise bound.

Note that, for a given mode, the proper noise bound has infinitely many different values, whereas the well-selected noise bound is unique. All proper noise bounds, including the well-selected noise bound, should give the same estimate of D^σ. To obtain $\bar{v}^\sigma_{\text{ws}}$, the FS method can be applied, which will be discussed in the subsequent section.

We're now interested in when to stop and check the consistency of data in $\hat{D}^\sigma$. Suggested by Theorems 3.1 and 3.2, the data consistency can be judged by checking whether the remaining data are w.r.t. a single mode. If the latter holds, then the changing rate of $\hat{\theta}$ would become slow. It can be seen from the following result.

Theorem 3.3. [3] *If $\hat{\theta}^\sigma_t$ and $\hat{\theta}^\sigma_{t+1}$ represents, respectively, the least squares fits of the data in $\hat{D}^\sigma$ and $\hat{D}^\sigma \backslash (y_i,\ \phi_i)$, then,*

$$\|\hat{\theta}^\sigma_{t+1} - \hat{\theta}^\sigma_t\|_2^2 = [\frac{r_i}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i}]^2\phi_i^T(\Phi_t^T\Phi_t)^{-2}\phi_i.$$

Proof. It can be found in [3]. But, for clarity, we show it again with the notation used here:

$$\begin{aligned}
\hat{\theta}^\sigma_{t+1} &= (\Phi_{t+1}^T\Phi_{t+1})^{-1}(\Phi_t^T Y_t - \phi_i y_i) \\
&= (\Phi_t^T\Phi_t - \phi_i\phi_i^T)^{-1}(\Phi_t^T Y_t - \phi_i y_i) \\
&= \{(\Phi_t^T\Phi_t)^{-1} + \frac{(\Phi_t^T\Phi_t)^{-1}\phi_i\phi_i^T(\Phi_t^T\Phi_t)^{-1}}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i}\}(\Phi_t^T Y_t - \phi_i y_i) \\
&= \hat{\theta}^\sigma_t + (\Phi_t^T\Phi_t)^{-1}\phi_i\frac{\hat{y}_i - [1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i]y_i - \phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i y_i}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i} \\
&= \hat{\theta}^\sigma_t + (\Phi_t^T\Phi_t)^{-1}\phi_i\frac{\hat{y}_i - y_i}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i} \\
&= \hat{\theta}^\sigma_t - (\Phi_t^T\Phi_t)^{-1}\phi_i\frac{r_i}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i} \\
&\Rightarrow \|\hat{\theta}^\sigma_{t+1} - \hat{\theta}^\sigma_t\|_2^2 = [\frac{r_i}{1-\phi_i^T(\Phi_t^T\Phi_t)^{-1}\phi_i}]^2\phi_i^T(\Phi_t^T\Phi_t)^{-2}\phi_i
\end{aligned}$$

This completes the proof. □

Remark 3.2. Theorem 3.3 reveals that the difference between $\hat{\theta}^\sigma_{t+1}$ and $\hat{\theta}^\sigma_t$ is related to the value of r_i. When $\hat{D}^\sigma$ contains data collected from different modes, $\|\hat{\theta}^\sigma_{t+1} - \hat{\theta}^\sigma_t\|_2$ is likely to be large because LMR would be large, see (3.5).

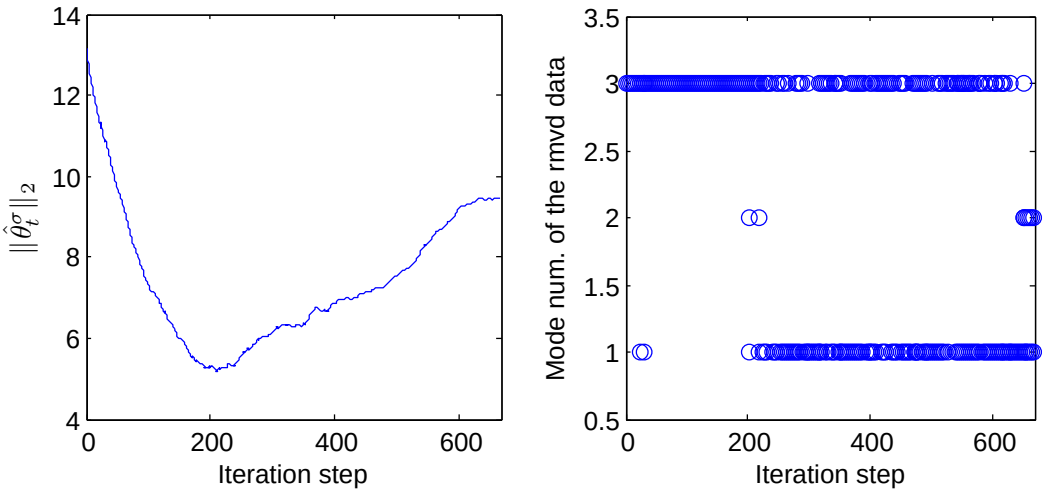


Figure 3.2: The relationship between $\|\hat{\theta}_t^\sigma\|_2$ and the removed data

Fig. 3.2 shows a typical way of change in $\|\hat{\theta}_t^\sigma\|_2$ and the removed data from $\hat{D}^\sigma$. In this case, the remaining data are mostly w.r.t. mode 2 as $\|\hat{\theta}_t^\sigma\|_2$ exhibits the trend of slowing varying. Therefore, the idea here is to activate the consistency checking when the increment in $\|\hat{\theta}_t^\sigma\|_2$ sustains a small value for a period of time.

3.4.2 The FS method

During the search of $\bar{v}_{\text{ws}}^\sigma$, there are two tasks that need to be done by the FS method. The first task is to obtain a $\hat{\theta}^\sigma$ for checking the data consistency (see Definition 3.1); the second one is to bring back any data without destroying the data consistency and calculate the well-selected noise bound.

The FS method employed in this chapter was originally developed in the statistical society for regression and multivariate data analysis [3, 4]. It can be used to reject outliers or mis-classified data in our case by carrying out a hypothesis test on some statistic of a size-increasing subset ($\hat{D}^\sigma$ in our case).

Regarding the first task, $\hat{\theta}^\sigma$ is desired to be close to θ_0^σ, because a precise estimation can help to identify the data w.r.t. mode σ more accurately. We can give a robust estimate of θ_0^σ using the least median of squares (LMS) estimator [58] or least trimmed squares (LTS) estimator [59].

For the second task, it can be solved by testing the normality of the residuals. In this chapter, we calculate the following test statistic of the data not

belonging to $\hat{D}^\sigma$ [3]:

$$r_j^* = \frac{r_j}{\sqrt{s_0^2[1+\phi_j^T(\Phi^T\Phi)^{-1}\phi_j]}},\ \forall(y_j,\phi_j)\notin\hat{D}^\sigma,$$

where $s_0^2 = \sum_{(y_i,\phi_i)\in\hat{D}^\sigma}\frac{r_i^2}{|\hat{D}^\sigma|-n}$. The data consistency or the normality of residuals imposes an envelop on the test statistic, which is specified in terms of order statistics with different quantiles (e.g., 99.99%). We refer the reader to [56] for details. If the minimum value of $|r_j^*|$ is contained in the envelop, then we should add the associated data back to $\hat{D}^\sigma$; otherwise we claim that all the data $(y_j,\phi_j)\notin\hat{D}^\sigma$ are outliers for mode σ, and calculate $\bar{v}_{\text{ws}}^\sigma$ in (3.8).

3.4.3 The implementation

To summarize all the above discussion, the implementation of the proposed RH-FS method consists of the following steps:

3-B1 Run step 3-A1 and 3-A3 iteratively and record $\hat{\theta}_t^\sigma$; if it is changing slowly, go to step 3-B2.

3-B2 Obtain a robust estimate of θ_0^σ using LMS or LTS and calculate all residuals.

3-B3 Add back all data in the ascending order of r_j^*, such that the data consistency is always satisfied.

3-B4 Calculate $\bar{v}_{\text{ws}}^\sigma$ and re-estimate $\hat{\theta}^\sigma$.

Here, we would like to further discuss step 3-B1, where a slow-varying $\hat{\theta}_t^\sigma$ must be declared. In some situations, we may have some prior knowledge on the range of parameters. If so, we could compute the maximum standard deviation (MSD) of parameters instead of checking $\|\hat{\theta}_t^\sigma\|_2$, which is defined as below.

$$\text{MSD}(\hat{\theta}_t^\sigma) \triangleq \max_{i=1,2,\ldots,n}\sqrt{\frac{1}{z_0-1}\sum_{j=1}^{z_0}[\hat{\theta}_{t-j}^\sigma(i)-\bar{\theta}_t^\sigma(i)]^2},$$

where (i) represents the i^{th} entry of the associated vector, $\bar{\theta}_t^\sigma$ is the mean vector of $\{\hat{\theta}_{t-z_0}^\sigma,\hat{\theta}_{t-z_0+1}^\sigma,...,\hat{\theta}_{t-1}^\sigma\}$, and $z_0\in\mathbb{Z}^+$ is a finite time horizon. By this means, we can gain more insight from this measure.

Remark 3.3. The summarized implementation procedures provide parameter estimation of one mode. By repeatedly applying the procedures, we can obtain parameter estimation for all modes, including noise bounds of different levels.

3.5 Simulation results

To demonstrate the effectiveness of the proposed method, we test it using a randomly generated SARX model and compare it with the $\bar{v}^\sigma$ estimation method mentioned in [13].

Example 3.1.

In this example, we consider the following second-order SARX model:

$$y_k = -a_1^\sigma y_{k-1} - a_2^\sigma y_{k-2} + b_1^\sigma u_{k-1} + b_2^\sigma u_{k-2} + v_k,$$

where $\sigma \in \{1, 2, 3\}$ and $a_1^\sigma,\ a_2^\sigma,\ b_1^\sigma,\ b_2^\sigma$ are randomly generated such that each mode is stable. u_k is a white Gaussian noise signal with variance equal to 1. For different modes, the measurement noise, v_k, is also normally distributed but with different noise levels, see Fig. 3.3 (a), the corresponding signal-to-noise ratio (SNR) being approximately equal to 15dB, 20dB and 30dB, respectively. In each simulation trial, 1000 data points are collected for identification and the switching signal is randomly selected from $\{1, 2, 3\}$ at time k.

Fig. 3.3 (b)[‡] shows the evolution of the test statistic in one trial. It gives the idea how r_j^* goes beyond the 99.99% envelope: when it stays outside of the envelope for 3 consecutive data points, we stop adding more data to $\hat{D}^\sigma$. In the case of Fig. 3.3 (b), the first identified mode has about 440 data, which corresponds to Mode 3; see Fig. 3.3 (a).

Below, we define two measures to check the tightness of the estimated noise bound,

$$\text{Cmp} \triangleq \frac{|\hat{D}^\sigma \cap D^\sigma|}{|D^\sigma|} \times 100\%,$$

$$\text{NormErr} \triangleq \frac{\|\hat{\theta}^\sigma - \theta_0^\sigma\|_2}{\|\theta_0^\sigma\|_2} \times 100\%.$$

"Cmp" gives the percentage of data in D^σ that has been successfully identified in $\hat{D}^\sigma$. A higher percentage means a more complete D^σ obtained. "NormErr"

‡ It was generated by using the FSDA toolbox (http://www.riani.it/MATLAB.htm)

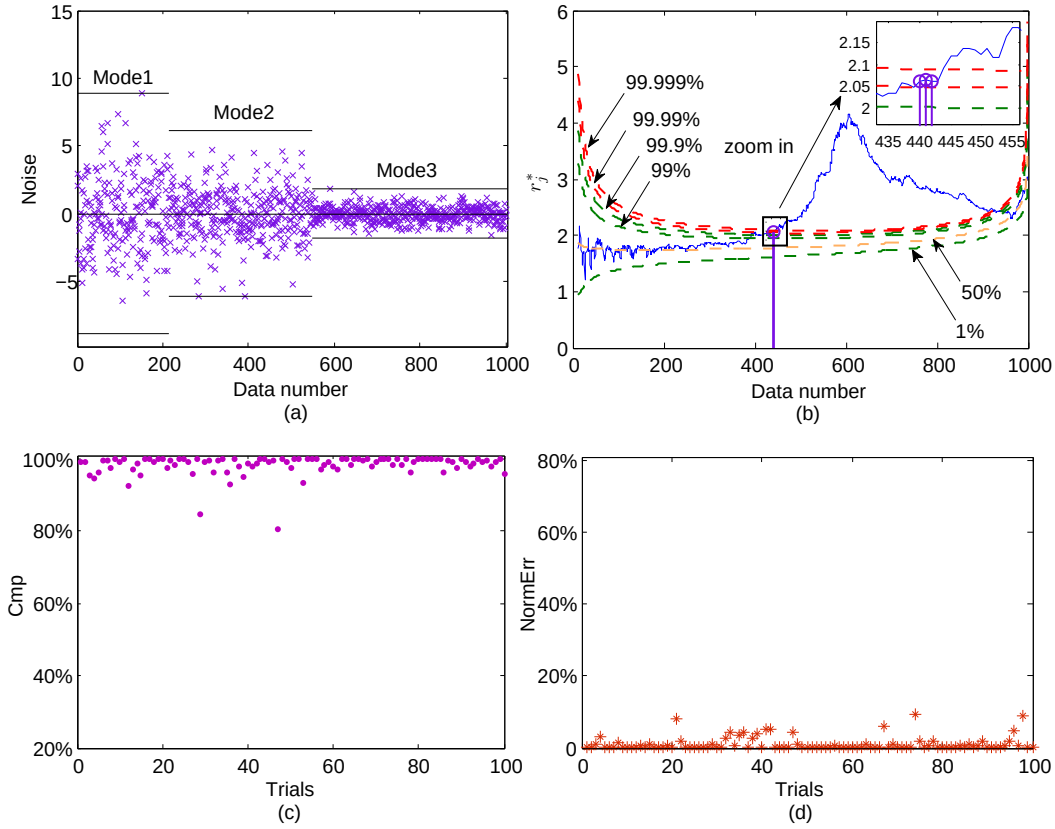


Figure 3.3: Simulation results of Example 3.1

is short for normalized error and it measures the accuracy of the parameter estimation. Ideally, we desire a high value in Cmp (to prevent an over tight $\bar{v}^\sigma$), and a low value in NormErr (to prevent a over loose $\bar{v}^\sigma$).

Fig. 3.3 (c) and (d) are two plots of these measures for 100 trials. We observe that the completeness is sustained at a high level, around 96%, except a few cases. At the same time, the normalized error is mostly less than 3%. Therefore, the estimated noise bounds can characterize the measurement noises very well.

Example 3.2.

The second example is used to compare the proposed method with the work in [13]. As we discussed in Section 3.2, the $\bar{v}^\sigma$ estimation in [13] is based on a trade-off between the number of modes and fitting errors. For convenience, let's call it the "trade-off" method. No matter how many noise levels there exist, the trade-off method estimates a common noise bound for

all modes. Here, we let

$$a_1^1 = 0.1,\ a_2^1 = 0.3,\ b_1^1 = 4,\ b_2^1 = 1.5;$$
$$a_1^2 = -0.2,\ a_2^2 = 0.5,\ b_1^2 = 2,\ b_2^2 = 5;$$
$$a_1^3 = -0.7,\ a_2^3 = 0.4,\ b_1^3 = 1.5,\ b_2^3 = -3;$$

SNR of Mode 2 and Mode 3 are both set to be 30dB; other settings are kept the same as Example 3.1.

Table 3.1: Comparison on the estimation accuracy of $\bar{v}^\sigma$

	Trade-off	RH-FS	True value of $\bar{v}^\sigma$
Mode 1	1.8	2.43	2.76
Mode 2	1.8	0.55	0.54
Mode 3	1.8	0.51	0.51

Table 3.2: Comparison on the estimation accuracy of parameters

NormErr	Trade-off	RH-FS
Mode 1	1.99%	1.41%
Mode 2	0.46%	0.15%
Mode 3	2.18%	0.39%

For the above system, the estimation results are shown in Tables 3.1 and 3.2. It is seen that the noise bounds, obtained by the proposed method, are very close to their true values. Therefore, the estimated parameters using RH-FS are more accurate than those by the trade-off method; see Table 3.2. When there is a large gap in noise bounds for different modes, the advantage of using RH-FS will become more obvious.

3.6 Summary

In this chapter, we have studied the general case of identification of SLSs in the presence of measurement noise. Set membership identification, as a powerful tool, is able to identify the noisy systems by solving a series of MAX FS problems; but it relies on the true noise bound of each mode. With the help of the proposed RH-FS method, we can estimate the noise bounds and

parameters simultaneously for one mode at a time, which enables us to handle SLSs with multiple unknown levels. This is especially important when a few modes of SLSs are highly noisy and others are not.

Chapter 4

The RLS Algorithm with A Resetting Strategy*

In Chapters 2 and 3, offline identification of SLSs has been investigated. The proposed methods are applicable only for a batch of I/O data and cannot be applied for online update of model parameters. For this reason, it is necessary to develop online identification approaches.

In this chapter, we will propose an online identification algorithm, which can handle the SLSs with switching sequences in an arbitrary form. The main results of the work include the following aspects: (i) the mode detection approach suggested in [6] is analyzed and two different types of mode mismatches are explained by examples; (ii) from the compensation point of view, a resetting strategy is proposed based on the analysis on the mode mismatch and characteristics of the recursive least squares (RLS) algorithm.

The remaining of this chapter is organized as follows. Section 4.1 introduces the current research on online identification of SLSs. Section 4.2 describes the concerned system model structure and the identification problem to be solved. Section 4.3 presents the mode detection method and explains the reason of mode mismatches. Section 4.4 analyzes the RLS algorithm and proposes a resetting strategy for online identification. Finally, the simulation results are shown in Section 4.5 and a summary is given in Section 4.6.

*A version of Chapter 4 has been published in [69].

4.1 Introduction

Currently, the research on online identification of SLSs is still limited. There are only a few papers found in the literature. It is not because of the lack of significance, but rather because of the difficulty of the problem itself. Recall that, in offline identification, the main difficulty is on the classification of I/O data. For most (if not all) of existing offline approaches, the success of data classification cannot be 100% guaranteed. In online cases, data classification problem becomes even harder, since data points are sequentially acquired and some useful data manipulations, e.g., swap of data order, iterative processing of data, are not applicable. This has largely prevented the development of online identification methods.

Among the existing online methods, see, e.g., [6, 37, 67, 73], the clustering-based method [6] is relatively easy to implement and also effective for data classification. It consists of two stages: mode detection and parameter estimation. At the mode detection stage, a detection function is usually employed to find the running mode, which can directly affect the estimation results. However, designing a perfect detection function is very difficult; there often exist some mode mismatches, leading to assigning a wrong mode number to a data point. In particular, when initial parameters are not appropriately created, there is a large possibility of mode mismatch, which may lead to poor performance of parameter estimation.

In this chapter, we deal with this issue from the compensation point of view. By introducing a resetting strategy to the RLS algorithm, the negative effects of mode mismatches will be separated into a few resetting intervals, which effectively prevent them from being accumulated.

4.2 Problem description

We consider a single-input and single-output SLS described by the following SARX model:

$$\begin{aligned} &y_k = \phi_k^T \theta_0^\sigma + v_k, \\ &\phi_k \triangleq [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n, \\ &\theta_0^\sigma \triangleq [a_1^\sigma \ \ \cdots \ \ a_{n_a}^\sigma \ \ b_1^\sigma \ \ \cdots \ \ b_{n_b}^\sigma]^T \in \mathbb{R}^n, \end{aligned} \tag{4.1}$$

where ϕ_k and θ_0^σ represent the regression vector and parameter vector, respectively; $n_a \in \mathbb{Z}^+$ and $n_b \in \mathbb{Z}^+$ are the model orders; $\sigma \in \{1, 2, \ldots, s\}$ is

the switching signal or mode number that changes with time k; v_k is a white Gaussian noise.

Then, the online identification problem can be stated as follows:

Given the sequentially acquired I/O data pairs $\{(\phi_k,\ y_k)\}_{k=1}^{N}$, generated by the switched linear system in (4.1), assuming that the order n_a, n_b are known *a priori*, we are interested in estimating the parameters of each mode in real time.

As mentioned in Section 4.1, the data classification is still the key to solve the identification problem. In the clustering-based method [6], the mode detection function is used for detecting the running mode. In the next section, we would like to give a detailed discussion.

4.3 Mode detection

Regarding the mode detection, it is closely related to the mode switching sequence. Typically, there are roughly three types of mode switching:

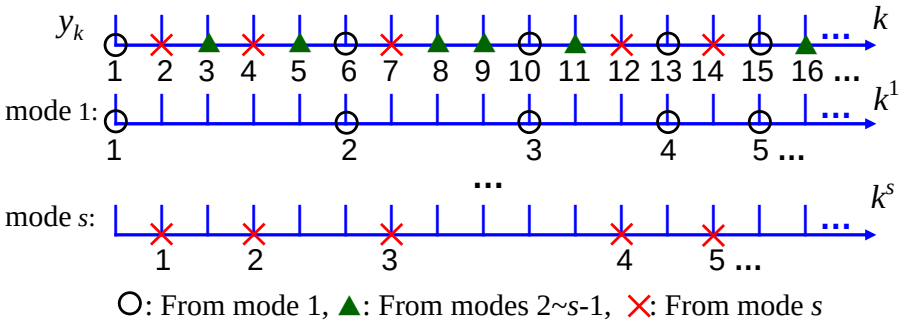


Figure 4.1: A mode switching sequence with arbitrary switching

- Switching periodically: In this case, the difficulties of the identification problem are reduced to the estimation of the mode switching period, which will be later discussed in Chapter 6.

- Switching aperiodically but slowly: If the switch appears aperiodically but the switching frequency is low, that is, the *dwell time* is sufficiently large, then the subspace methods could be used, see, e.g., [7–10, 14, 15, 53, 66].

- Switching arbitrarily: Compared with the above two, this type is more general. As it has no specific form, the running mode number needs to be identified for each I/O data. See an example in Fig. 4.1.

4.3.1 The detection function

In this chapter, the detection function we used is applicable for all types of mode switching sequence. It stems from the work in [6, 9]. At time k, the running mode σ^* is decided by checking a detection function, which is defined as,

$$\begin{aligned} d_{k,\sigma} &\triangleq \frac{|y_k - \phi_k^T \hat{\theta}_{k-1}^{\sigma}|}{\|[1 \quad -(\hat{\theta}_{k-1}^{\sigma})^T]\|_2} \\ &= \frac{|[y_k \quad \phi_k^T][1 \quad -(\hat{\theta}_{k-1}^{\sigma})^T]^T|}{\sqrt{1 + (\hat{\theta}_{k-1}^{\sigma})^T \hat{\theta}_{k-1}^{\sigma}}}, \quad \sigma \in \{1, \ldots, \hat{s}\}, \end{aligned} \tag{4.2}$$

where $\hat{\theta}_{k-1}^{\sigma}$ represents an estimated parameter vector at time $k-1$; $\hat{s}$ is the number of modes at time $k-1$. The minimum of $d_{k,\sigma}$ is denoted by

$$d_{k,\sigma^*} = \min_{\sigma} \; d_{k,\sigma}, \tag{4.3}$$

and σ^* is calculated by,

$$\sigma^* = \arg\min_{\sigma} \; d_{k,\sigma}. \tag{4.4}$$

Here, $d_{k,\sigma}$ geometrically means the distance from the point with position vector $[y_k \quad \phi_k^T]^T$ to the hyperplane with normal vector $[1 \quad -(\hat{\theta}_{k-1}^{\sigma})^T]^T$. All the hyperplanes pass through the origin of regressor domain.

This detection function can help to get a correct running mode for most data. However, mode mismatches still exist in some situations.

4.3.2 Mode mismatch

In the following, two different types of mode mismatch are discussed.

Type I: mode mismatches from ill-initialization

We know that the running mode is detected by comparing the distances between data points and hyperplanes. If there is an initial hyperplane that is always further away from the data points than other hyperplanes, then this hyperplane may not be used at all.

Example 4.1. (a 2-dimensional case):

$$y_k = b^\sigma u_{k-1}, \quad \sigma \in \{1,\ 2\}.$$

Here $b^1 = -0.25$ and $b^2 = 0.25$. Let the initial parameter vectors to be as, $\hat{\theta}_0^1 = \hat{b}^1 = -0.5;\ \hat{\theta}_0^2 = \hat{b}^2 = 4$.

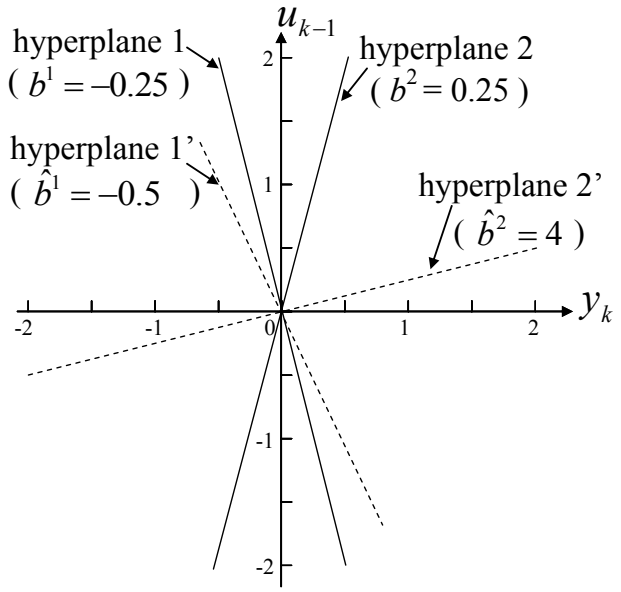


Figure 4.2: An illustration of mode mismatch type I

Fig. 4.2 shows the hyperplane positions of both modes. Hyperplanes 1' and 2' are specified by $\hat{\theta}_0^1$ and $\hat{\theta}_0^2$, respectively. When the standard RLS algorithm in [32] is used, we can see that the I/O data will always be assigned to hyperplane 1' due to a smaller distance.

Remark 4.1. we can detect this type of mode mismatch by checking the existence of such stationary parameter vectors, like hyperplane 2' in the above example.

Type II: mode mismatches from hyperplane bisecting zone

Except the ill-initialization, mode mismatches can also be caused by the data located in a bisecting zone of hyperplanes. Fig. 4.3, as an explanation on this type, shows a three-dimension case with two different modes. Same as Example 4.1, hyperplane 1 is associated with mode 1; hyperplanes 1' and 2' are specified by $\hat{\theta}_k^1$ and $\hat{\theta}_k^2$, respectively. They meet at the origin O.

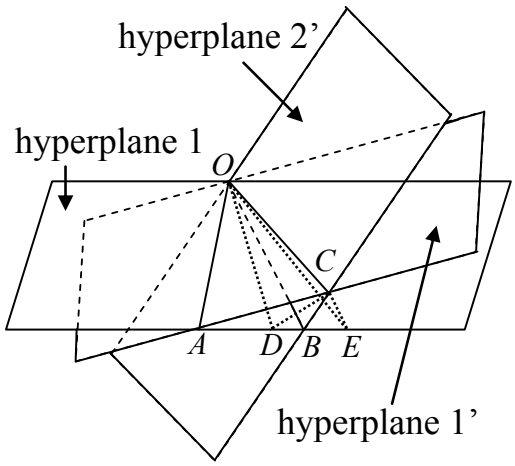


Figure 4.3: An illustration of mode mismatch type II

Now, we draw a bisecting plane OCD that separates the dihedral angle $\angle AOCB$ equally. Thus, the data located in the zone $OCDB$ have closer distance to hyperplane 2' than to hyperplane 1'. Similarly, by drawing another bisecting plane OCE, the data in the zone $OCBE$ has the same property as the data in $OCDB$. Therefore, according to equation (4.4), mode 2 will be mismatched to the data of mode 1 in the bisecting zone $OCDE$.

Remark 4.2. Regarding the second type of mode mismatch in Fig. 4.3, we observe that the potential mode mismatches only occur when the data of mode 1 are located in the bisecting zone $OCDE$. However, the data in $OCDE$ is only a small portion of the data of mode 1 and we may find that the chance of mode mismatch is smaller than the chance of correct mode detection. Moreover, when hyperplane 1' is getting closer to hyperplane 1, the chance of mode mismatch would be even smaller.

Remark 4.3. These two types of mode mismatches appear in both noisy and noise-free systems. In the presence of noise, mode detection is not only dependent on the location of data but also dependent on the properties of noise. A large noise will bring with more mode mismatches in the mode detection stage.

4.4 The modified RLS algorithm

In this section, we first analyze the influence of mode mismatches on the standard RLS algorithm and then present a modified recursive least squares

(MRLS) algorithm in details.

To avoid mode mismatches, we may want to design a more powerful method of mode detection. In fact, due to the lack of sufficient data features, it is rather difficult to create a perfect detection function that can match the correct mode with the I/O data, especially in the presence of measurement noise. In this work, we try to achieve this goal by using a compensation-based approach, which is based on the modification of RLS algorithm to reduce the negative effects brought by mode mismatches.

4.4.1 Analysis on the RLS algorithm

To proceed further, it is necessary to check the commonly used RLS algorithm from (4.5) to (4.7):

$$e_k = y_k - \phi_k^T \hat{\theta}_{k-1}^{\sigma^*}, \tag{4.5}$$

$$(P_k^{\sigma^*})^{-1} = (P_{k-1}^{\sigma^*})^{-1} + \phi_k \phi_k^T, \tag{4.6}$$

$$\hat{\theta}_k^{\sigma^*} = \hat{\theta}_{k-1}^{\sigma^*} + P_k^{\sigma^*} \phi_k e_k. \tag{4.7}$$

The estimated $\hat{\theta}_k^{\sigma^*}$ is obtained by equivalently solving the optimization problem shown as below [18],

$$\min_{\theta} \sum_{i=1}^{k} [y_i - \phi_i^T \theta]^2 + (\theta - \hat{\theta}_0^{\sigma^*})^T (P_0^{\sigma^*})^{-1} (\theta - \hat{\theta}_0^{\sigma^*}). \tag{4.8}$$

To reduce the effect of the second term, $P_0^{\sigma^*}$ is usually selected to be a large identity matrix, $P_0^{\sigma^*} = pI$, with $p \gg 1$. The initial parameter vectors are randomly generated or created by *a priori* knowledge on the processes. It has been proved that the initial parameter vector $\hat{\theta}_0^{\sigma^*}$ can converge to $\theta_0^{\sigma^*}$ in many references.

However, when the data are not all collected from the same mode, the estimated parameters will be far away from their true values. Let's go back to check equation (4.8) and we may find that this optimization problem could be equivalently written in the form of (4.9),

$$\min_{\theta} \sum_{i=t+1}^{k} [y_i - \phi_i^T \theta]^2 + (\theta - \hat{\theta}_t^{\sigma^*})^T (P_t^{\sigma^*})^{-1} (\theta - \hat{\theta}_t^{\sigma^*}). \tag{4.9}$$

For the second term, $P_t^{\sigma^*}$ is large only at the starting iterations. As k increases, $(P_t^{\sigma^*})^{-1}$ would increase steadily and dramatically [32]. It also can be seen from,

$$(P_k^{\sigma^*})^{-1} \geq (P_{k-1}^{\sigma^*})^{-1} \geq ... \geq (P_t^{\sigma^*})^{-1} \geq ... \geq (P_0^{\sigma^*})^{-1} > \mathbf{0}. \tag{4.10}$$

On the other hand, when mode mismatches exist before time t, it is reasonable to say that $\hat{\theta}_t^{\sigma^*}$ is not close to $\theta_0^{\sigma^*}$. Therefore, if k is finite, then at some starting time t, say $t = k_0$, the second term in (4.9) would take significant weight in this optimization problem and hence can not be ignored. In this situation, even if the rest data (from $k_0 + 1$ to k) are all exactly from mode σ^*, the result $\hat{\theta}_k^{\sigma^*}$ is still not a good estimate.

Remark 4.4. Recall that the detection function d_{k,σ^*} is dependent on the estimated $\hat{\theta}_{k-1}^{\sigma^*}$. So poorly estimated parameter vectors may further result in more mode mismatches in the following iterations and make the estimates even worse.

4.4.2 Modification of RLS algorithm

From the previous analysis, it is seen that the commonly used RLS algorithm may result in an unsatisfactory $\hat{\theta}_k^{\sigma^*}$ when mode mismatches are involved. The reason is ascribed to a small $P_k^{\sigma^*}$ matrix.

To fix this problem, we embed a resetting step in the RLS algorithm, *i.e.*, forcing $P_k^{\sigma^*} = P_0^{\sigma^*}$. For the sake of clarity, we denote the resetting time as $t_s^{\sigma^*}$ and call the time interval $[t_s^{\sigma^*},\ t_{s+1}^{\sigma^*} - 1]$ as resetting interval. The main idea of resetting is to prevent the negative effects of mode mismatches from spreading, because for a new resetting interval, the only thing passed over is the estimated parameters at time $t_s^{\sigma^*} - 1$. When $P_{t_s^{\sigma^*}}^{\sigma^*}$ is reset to be $P_0^{\sigma^*}$, the estimate will be obtained mainly on the first term of (4.9) again. Moreover, from Section 4.3, we know the chance of mode mismatch is generally small. When $\hat{\theta}_k^{\sigma^*} \to \theta_0^{\sigma^*}$, the chance of incorrect mode detection would be even smaller. These reasons explain why the estimation results get improved. We're now interested in how to activate the resetting step.

• Resetting condition

Since a small $P_k^{\sigma^*}$ usually lead to a small increment in $\hat{\theta}_k^{\sigma^*}$, see (4.7), we detect a small covariance matrix by checking the maximum standard deviation of parameters in $\hat{\theta}_k^{\sigma^*}$, like we did in Chapter 3. It is denoted by $\theta_{\text{MSD}}^{\sigma^*}$ and obtained in a finite time horizon $z_0 \in \mathbb{Z}^+$,

$$\theta_{\text{MSD}}^{\sigma^*} = \max_i \sqrt{\frac{1}{z_0 - 1} \sum_{t=1}^{z_0} [\hat{\theta}_{k-t}^{\sigma^*}(i) - \bar{\theta}^{\sigma^*}(i)]^2}, \tag{4.11}$$

where (i) represents the i^{th} entry of the associated vector and $\bar{\theta}^{\sigma^*}$ is the mean vector of $\{\hat{\theta}^{\sigma^*}_{k-z_0}, \hat{\theta}^{\sigma^*}_{k-z_0+1}, ..., \hat{\theta}^{\sigma^*}_{k-1}\}$. Define an indicator function $I^{\sigma^*}(\theta^{\sigma^*}_{\text{MSD}})$ as the resetting condition:

$$I^{\sigma^*}(\theta^{\sigma^*}_{\text{MSD}}) = \begin{cases} 1, & \theta^{\sigma^*}_{\text{MSD}} \leq \tau_1 \\ 0, & \theta^{\sigma^*}_{\text{MSD}} > \tau_1 \end{cases} \tag{4.12}$$

where $\tau_1 \in \mathbb{R}^+$ is a prescribed tolerance. If $I^{\sigma^*}(\theta^{\sigma^*}_k) = 1$, then it indicates the convergence rate is already slow and we are ready to reset the covariance matrix.

Remark 4.5. In fact, we only need to take the resetting step for the unsatisfactory or poorly estimated $\hat{\theta}^{\sigma^*}_k$. For good estimates, as mentioned, the chance of mode mismatch would be small and there is no need to reset $P^{\sigma^*}_k$.

• **Evaluation of $\hat{\theta}^i_k$**

The following shows how we evaluate the quality of $\hat{\theta}^{\sigma^*}_k$. We say an estimate is a good estimate when $\hat{\theta}^{\sigma^*}_k \approx \theta^{\sigma^*}_0$. For $v_k \sim \mathcal{N}(0,\ \upsilon_0^2)$, it can be proved that,

$$\Pr\{\ |v_k| > 6\upsilon_0\ \} \approx 2 \times 10^{-9}, \tag{4.13}$$

where the notation $\Pr\{X\}$ denotes the probability of event X. It means almost all data points are sufficiently close to their real hyperplanes. Therefore, when the resetting condition holds, we can evaluate $\hat{\theta}^{\sigma^*}_k$ by checking,

$$\|[1 \quad -(\hat{\theta}^{\sigma^*}_{k-1})^T]\|_2 d_{k,\sigma^*} > 6\upsilon_0. \tag{4.14}$$

If conditions (4.12) and (4.14) hold simultaneously, we take a resetting step to that mode.

Remark 4.6. The resetting strategy is an efficient way to avoid the second type of mode mismatch, but it is not enough to get rid of the first type of mode mismatch.

From Section 4.3, we know this problem can be fixed by replacing all stationary parameter vectors with new $\hat{\theta}_0$. However, some of them are not necessary to be replaced, since they might be corresponding to the modes that are not visited in a long time interval. Therefore, it is difficult to say which stationary mode is associated with the first type of mode mismatch.

For this reason, a new method is employed. We create the initial parameter vectors one by one. A new parameter vector is created when any unsatisfactory $\hat{\theta}_k^{\sigma^*}$ demonstrates a low convergence rate. Meanwhile, we set the new parameter vector to be the same as $\hat{\theta}_k^{\sigma^*}$. By doing so, the first class of mode mismatch is not likely to happen.

4.4.3 The MRLS algorithm

Based on the above modifications, the modified recursive least squares algorithm is summarized in Algorithm 2 (on the next page).

Algorithm 2 The RH solution

- Step 4-A: Initialization

4-A1 Randomly generate a $\hat{\theta}_0^1$ s.t. the model in (4.1) is stable and let $\hat{s} = 1$.
4-A2 Set $P_0^1 = pI$ with $p \gg 1$ and $I^1 = 0$.

- Step 4-B: Iteration

for $k = 1 : N$ **do**
4-B1 Obtain (ϕ_k, y_k) and decide σ^* by (4.2)-(4.4);
4-B2
if $I^{\sigma^*} = 1$ and $d_{k,\sigma^*} > 6\upsilon_0 / \|[1 \quad -(\hat{\theta}_{k-1}^{\sigma^*})^T]\|_2$ **then**
if $\hat{s} < s$ **then**
$\hat{s} = \hat{s} + 1$;
$\hat{\theta}_{k-1}^{\hat{s}} = \hat{\theta}_{k-1}^{\sigma^*}$;
$P_{k-1}^{\hat{s}} = pI$;
$I^{\hat{s}} = 0$;
end if
$P_{k-1}^{\sigma^*} = pI$ and $I^{\sigma^*} = 0$;
end if
4-B3 Update e_k, $P_k^{\sigma^*}$ and $\hat{\theta}_k^{\sigma^*}$ by the following equations:

$$
\begin{aligned}
e_k &= y_k - \phi_k^T \hat{\theta}_{k-1}^{\sigma^*} \\
P_k^{\sigma^*} &= P_{k-1}^{\sigma^*} \left[I - \frac{\phi_k \phi_k^T P_{k-1}^{\sigma^*}}{1 + \phi_k^T P_{k-1}^{\sigma^*} \phi_k} \right] \\
\hat{\theta}_k^{\sigma^*} &= \hat{\theta}_{k-1}^{\sigma^*} + P_k^{\sigma^*} \phi_k^{\sigma^*} e_k
\end{aligned}
$$

4-B4 Calculate $\theta_{\mathrm{MSD}}^{\sigma^*}$ and I^{σ^*} by (4.11) and (4.12);
end for

4.4.4 Practical issues

• **Selection of z_0 and τ_1**

Intuitively, z_0 is expected to be large so that it reflects high accuracy on the variety of parameters; but it is not necessary for RLS based algorithms, since the increment in $\hat{\theta}_k^{\sigma^*}$ decreases at a fast rate, see (4.10). From practical experience, this parameter can be easily tuned and the final estimates are not very sensitive on its value. As for τ_1, it affects the number of times to reset the covariance matrix. A small τ_1 is necessary to guarantee good performance of the MRLS algorithm. However, a too small value would bring a great number of resettings, which should be taken into account.

• **Convergence**

For the proposed MRLS algorithm, it is necessary to show whether the embedded resetting step will lead to divergence and whether the parameters can still converge to the true values when mode mismatch exists. The first problem can be answered by checking the work in [23] and [61] with the assumption that no mode mismatch is involved, while the second problem is more important and its theoretical analysis is still under investigation.

4.5 Simulation results

In order to show the effectiveness of the proposed algorithm, the following SARX model is used:

$$
\begin{aligned}
&A^\sigma(z)y_k = B^\sigma(z)u_k + v_k,\ \sigma \in \{1,2\} \\
&\text{Mode 1}: \begin{cases} A^1(z) &= 1 + 0.1z^{-1} + 0.3z^{-2}, \\ B^1(z) &= 4z^{-1} + 1.5z^{-2}, \end{cases} \\
&\text{Mode 2}: \begin{cases} A^2(z) &= 1 - 0.2z^{-1} + 0.5z^{-2}, \\ B^2(z) &= 2z^{-1} + 5z^{-2}. \end{cases}
\end{aligned}
$$

In the simulations, the input u_k and noise v_k are zero-mean white Gaussian noise signals with variance equal to 1 and 0.2, respectively. The signal to noise ratio (SNR) is about 20 dB. Modes 1 and 2 are randomly switched with a probability of 0.5 for each. The process models are simulated for 2000 I/O data pairs. In addition, the initial covariance matrix $P_0 = 1000I$.

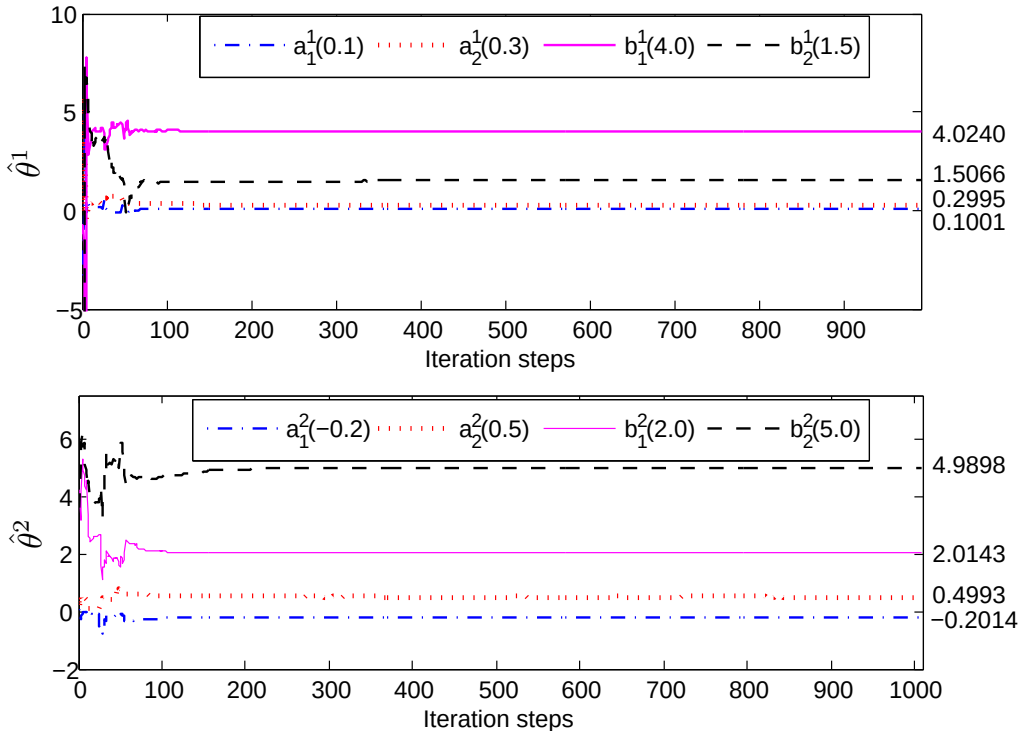


Figure 4.4: The parameters evolution for a single run

Fig. 4.4 illustrates the parameter evolution from a single run of the MRLS algorithm, where the numbers in the legend are the true values of the corresponding parameters. In this simulation, both modes take the resetting step three times. Table 4.1 shows the associated resetting intervals and the number of mode mismatches. We observe that all parameter estimates are able to converge to theirs true values and the chance of mode mismatches decreases when a resetting step is taken.

Table 4.1: The number of mode mismatches for all resetting intervals

	$[t_s^1,\ t_{s+1}^1 - 1]$	No. (or %)	$[t_s^2,\ t_{s+1}^2 - 1]$	No. (or %)
1	[1, 25]	11 (44.00%)	[1, 25]	13 (52.00%)
2	[26, 47]	7 (31.82%)	[26, 46]	5 (23.81%)
3	[48, 990]	47 (4.98%)	[47, 1008]	58 (6.03%)

Table 4.2 reports the estimation results by Monte Carlo simulation with 100 runs. The results are compared with the forgetting factor based RLS method (shorthand for FRLS) by [6], where the detection function $d_{k,\sigma}$ and

I/O data pairs are the same. The forgetting factor equals 0.9 as used in [6]. It shows that the MRLS algorithm has better performance.

Table 4.2: Comparison results in 100 Monte Carlo simulations

SNR=20dB	MRLS	FRLS
$\hat{a}_1^1$ (0.1000)	0.1026±0.0188	0.0781±0.1179
$\hat{a}_2^1$ (0.3000)	0.2937±0.0436	0.2843±0.1546
$\hat{b}_1^1$ (4.0000)	3.9853±0.1560	3.8779±0.6245
$\hat{b}_2^1$ (1.5000)	1.5241±0.2398	1.4630±0.3172
$\hat{a}_1^2$ (-0.2000)	-0.2009±0.0383	-0.1883±0.0392
$\hat{a}_2^2$ (0.5000)	0.5015±0.0504	0.4915±0.0356
$\hat{b}_1^2$ (2.0000)	1.9756±0.2380	2.0020±0.3588
$\hat{b}_2^2$ (5.0000)	4.9722±0.3314	4.8998±0.5557

4.6 Summary

In this chapter, online identification of SLSs has been studied. We have analyzed two types of mode mismatches and proposed a RLS algorithm with a resetting strategy. This algorithm can effectively reduce the effect of mode mismatch and significantly improve the estimation performance.

Chapter 5

The Hough Transform Based Online Identification

As mentioned in Chapter 4, the key of developing a good online identification approach is to guarantee the success of data classification. To achieve this goal, one may resort to a robust identification algorithm that is able to compensate negative effects caused by mode mismatch, see, e.g., the MRLS algorithm. An alternative way is to make the initial parameters sufficiently close to true values, which helps reducing the chance of mode mismatch.

This chapter studies the online identification of SLSs with the second approach. We firstly introduce the well-known Hough transform (HT) technique and develop an online HT-based estimator. Then, we propose a new online identification algorithm, namely, HT-clustering algorithm, by integrating the online HT-based estimator with an online clustering estimator using a feedback mechanism. Compared with the MRLS algorithm in Chapter 4, the HT-clustering algorithm does not need to know the number of modes, as it has similarity to the set membership identification.

The remaining of this chapter is organized as follows: Section 5.1 gives a background of developing the HT-clustering algorithm. Section 5.2 introduces the HT technique and its usage for identification. Section 5.3 provides an online implementation of the HT estimator. Section 5.4 presents the details of the proposed HT-clustering algorithm. Section 5.5 shows some numerical examples and Section 5.6 summarizes the work in this chapter.

5.1 Introduction

In the previous chapter, we have developed the MRLS algorithm for online identification of SLSs. Although it has shown some effectiveness through simulation results, we cannot deny that there exist a few limitations in the algorithm: (i) it requires knowing the number of modes; (ii) data information is not fully used due to the resetting procedure.

As discussed in Chapter 2, by solving a sequence of MAX FS problems, we can estimate parameters without knowing the number of modes. It has inspired us to find an online solution to the MAX FS problem. In [2], the authors mentioned that the HT technique can be applied to solve the MAX FS problem, but the discussion was made in offline cases. In comparison with other offline solutions, e.g., the improved Agmon-Motzkin-Schoenberg relaxation solution [12] and the sparse optimization solution [5, 51], the HT technique is more suitable to be extended to an online version. Since the HT-based estimator has a similarity to the set membership identification, we consider the following SARX model:

$$
\begin{aligned}
&y_k = \phi_k^T \theta_0^\sigma + v_k, \quad \sigma \in \{1, \ldots, s\} \\
&\phi_k \triangleq [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n, \\
&\theta_0^\sigma \triangleq [a_1^\sigma \ \cdots \ a_{n_a}^\sigma \ \ b_1^\sigma \ \cdots \ b_{n_b}^\sigma]^T \in \mathbb{R}^n,
\end{aligned}
$$

where $|v_k|$ is assumed to be bounded by $\bar{v}$; s is assumed to be unknown; other notation is the same as we defined in previous chapters.

The contribution of this chapter is two-fold. Firstly, it provides an online HT-based identification framework to estimate parameters. Because of the similarity to the set membership identification, the HT-based identification is capable of dealing with noisy data and cases with an unknown number of modes. Secondly, we propose the HT-clustering algorithm in this chapter, which can be used for time-varying systems or some nonlinear systems. With the help of a clustering estimator and a "feedback" mechanism, we can guarantee a low memory use and a high model fit in the estimation process.

5.2 HT-based identification

5.2.1 Main idea of the HT

The HT is a classical feature extraction technique that is typically used in image processing for detecting arbitrary curves, such as line, circle and ellipse. In this chapter, we concentrate on the detection of high dimensional hyperplanes, for which point-hyperplane duality is very important.

By Π we denote a n dimensional hyperplane in the *data space*,

$$\Pi = \{(\mathbf{x}, y) : \ y = c_0 + \mathbf{c}^T\mathbf{x}\}, \tag{5.1}$$

where $y \in \mathbb{R}$, $c_0 \in \mathbb{R}$, $\mathbf{c} \in \mathbb{R}^{n-1}$ and $\mathbf{x} \in \mathbb{R}^{n-1}$. For any given point on Π, $[\mathbf{x}_i^T, \ y_i]^T$, it can be transformed into the following hyperplane in the *parametric space*,

$$\Gamma_i = \{(\mathbf{c}, c_0) : \ c_0 = y_i - \mathbf{x}_i^T\mathbf{c}\}. \tag{5.2}$$

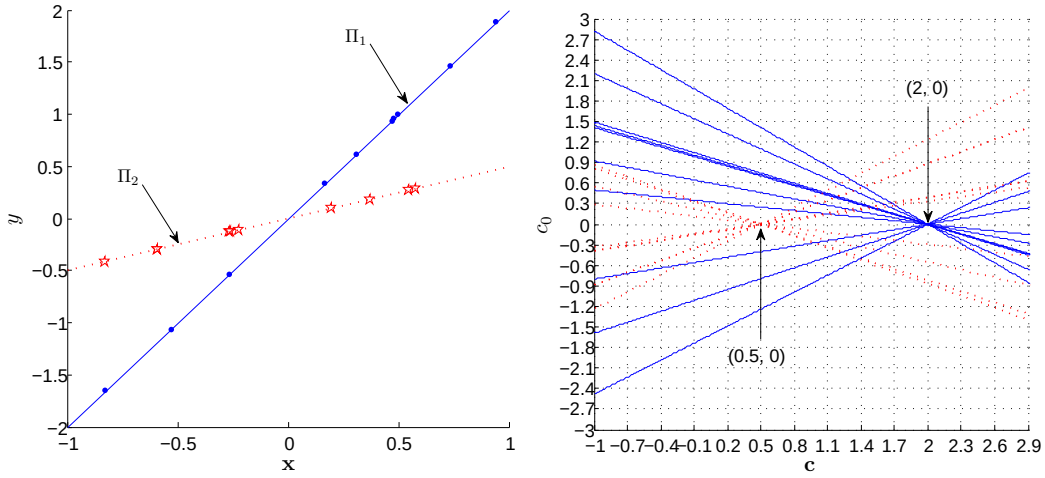
With a collection of n different points $\{[\mathbf{x}_1^T, \ y_1]^T, [\mathbf{x}_2^T, \ y_2]^T, \cdots, [\mathbf{x}_n^T, \ y_n]^T\}$, the parameters c_0 and $\mathbf{c}$ can be obtained by solving the following equations,

$$\begin{cases} c_0 = y_1 - \mathbf{x}_1^T\mathbf{c}, \\ c_0 = y_2 - \mathbf{x}_2^T\mathbf{c}, \\ \quad \vdots \\ c_0 = y_N - \mathbf{x}_n^T\mathbf{c}. \end{cases} \tag{5.3}$$

Geometrically, the solution of equations (5.3) is the crossing point of the hyperplanes, $\Gamma_1, \ \Gamma_2, \ \cdots, \ \Gamma_n$. Since the crossing point corresponds to the parameters, it is termed as *parametric point*.

Fig. 5.1 shows a two dimensional example of HT. Fig. 5.1 (a) displays some data points on two different straight lines, $\Pi_1 : \ y = 2\mathbf{x}$ and $\Pi_2 : \ y = 0.5\mathbf{x}$. Each line in Fig. 5.1 (b) is transformed from a point in Fig. 5.1 (a) and the lines of the same mode intersect at the parametric points. The slopes of Π_1 and Π_2 can be reflected by the parametric points, which are located in the grids with a large number of line-hits.

Here, we note that the hyperplane described in (5.1) is parameterized in the Cartesian coordinate, which is not suitable for the cases when the hyperplane has an infinite slope or intercept. However, for the identification problem, Cartesian parametrization is enough to characterize all SLSs, since the model parameters to be identified are typically finite.



(a) data space (b) parametric space

Figure 5.1: An example of the HT

5.2.2 Standard HT estimator

We now introduce the standard HT estimator. Let us first define a hypercube, $H(\theta)$, which is used to find the parametric points:

$$H(\theta) \triangleq \{(\mathbf{c}', c_0') : \ |\mathbf{c}' - \mathbf{c}| \preceq \frac{\mathbf{d}}{2}, \ |c_0' - c_0| \leq \frac{d}{2}\}, \tag{5.4}$$

where $\theta = [\mathbf{c}^T, c_0]^T$ and $d \in \mathbb{R}^+$ is a constant; $|\cdot|$ returns the element-wise absolute value of "$\cdot$"; $\mathbf{d}$ is an all-d's vector with a compatible dimension; $\preceq$ denotes the element-wise inequality. Then, the parametric points can be estimated with the following standard HT estimator, see [31],

$$\hat{\theta} = \arg\max_{\theta \in \mathbb{R}^n} \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\{H(\theta) \cap \Gamma_i \neq \emptyset\}. \tag{5.5}$$

where the indicator function $\mathbf{1}\{X\}$ returns 1 if predicate X is true and 0 otherwise.

The standard HT estimator provides an intuitive way to capture the parametric points with several advantages: (1) it is relatively robust with outliers; (2) it is applicable to the cases when the number of modes or subsystems is unknown; (3) it is online implementable and does not need to initialize parameters; (4) the estimation convergence and consistency have been established; see, e.g., [31].

With regards to the online implementation of HT, a typical way is to partition the parametric space into small cells and find the cell with the largest

number of hyperplane-hits. Clearly, with this implementation, as the number of dimension increases, the number of cells grows exponentially and computation becomes an overburden.

5.3 Online implementation of HT

In this section, we provides a new online implementation of the HT estimator. There are two tasks that should be done: one is to determine the value of d; the other is to reduce the computational complexity for online implementation of HT.

5.3.1 Relation to set membership identification

Before using the HT estimator in (5.5), we need to prescribe the value of d, that is, the size of the hypercube, which directly affects the performance of estimation. An oversized hypercube will cause many mis-classified data or outliers to be involved, while an undersized hypercube may lead to over-fit.

The value of d can be determined according to the noise level. When $v_k = 0$, d may be set to an arbitrary small positive number, since hyperplanes in the parametric space intersect at the parametric points. When $v_k \neq 0$, it is necessary to find out the connection of d and residuals. To this end, we would like to show the analogy between the HT-based identification and the set membership identification.

Using the set membership identification, see, e.g., [13], we may identify a SARX model by solving a sequence of MAX FS problems, which can be formulated in the following form,

$$\hat{\theta} = \arg\max_{\theta \in \mathbb{R}^n} \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}\{|y_k - \phi_k^T \theta| \leq \bar{v}\}, \tag{5.6}$$

On the other hand, we may also apply the HT estimator to identify the SARX model. To make it clear, we rewrite the data and parametric hyperplanes as follows:

$$\begin{aligned} \Pi' &= \{(\phi_k, y_k) : \ y_k = (\theta_0^\sigma)^T \phi_k\}, \\ \Gamma'_k &= \{\theta_0^\sigma : \ 0 = y_k - \phi_k^T \theta_0^\sigma\}. \end{aligned} \tag{5.7}$$

Here, we consider the following line segment cell instead of the hypercube in (5.4):

$$L(\theta) = \{\theta' : \ |\theta' - \theta| = \mathbf{0}, \ |y_k - \phi_k^T \theta| \leq \frac{d}{2}\}. \tag{5.8}$$

In fact, $L(\theta)$ is identical with $H(\theta)$ when $\mathbf{d} = \mathbf{0}$. Using the line segment cell to find the parametric points, we can compute the parameters with the following equation:

$$\hat{\theta} = \arg\max_{\theta \in \mathbb{R}^n} \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}\{L(\theta) \cap \Gamma_k' \neq \emptyset\}. \tag{5.9}$$

By prescribing $d = 2\bar{v}$, we find that $L(\theta) \cap \Gamma_i' \neq \emptyset$ if and only if $|y_k - \phi_k^T \theta| \leq \bar{v}$. This means that the $L(\theta)$ is just in a right size. In this situation, the HT-based identification is consistent with the set membership identification.

Remark 5.1. Although we have shown that the HT-based identification can be converted into the set membership identification, so far we cannot find any online solution to the MAX FS problem. By taking advantage of the geometric property of HT, we believe that developing an HT-based online implementation is a better option.

5.3.2 Online implementation procedure

In the preceding section, we mentioned that a standard online implementation of HT can hardly meet our needs because of the complexity. This motivates us to develop a new online implementation procedure.

Recall that in the classic implementation of HT, one needs to partition the parametric space into a huge number of small cells and keep all of them in storage. It is impractical for identification, as the parametric space is usually with high dimension. To avoid large usage of memory, we may adopt a clustering idea to classify the I/O data into clusters. The idea is based on the fact that the parametric hyperplanes of the data w.r.t. the same mode will intersect within a small region. First, let's define the cluster we used in the sequel.

Definition 5.1. *Suppose $\mathcal{C}$ is a non-empty subset of D, if the following conditions are satisfied,*

1) existing a θ such that $\max_{(\phi_k,y_k)\in\mathcal{C}} |y_k - \phi_k^T\theta| \leq \bar{v}$;

2) including any data pair (ϕ_k, y_k) that belongs $D \setminus \mathcal{C}$ into $\mathcal{C}$, condition 1) is violated;

then we define $\mathcal{C}$ as a cluster.

We note that the cluster defined above is based on the line segment cell. For the noise free case, as $\bar{v} = 0$, the cell is turned into a point. If no data fits two different modes, then the data in the same cluster are associated with the same mode. For the noisy case, the data in the same cluster are possibly w.r.t. different modes.

Next, we define some notations. We assume that we have a collection of R clusters at time k, namely $\{\mathcal{C}(r)\}_{r=1}^R$, where $\mathcal{C}(r)$ denote the r^{th} cluster. $\hat{\theta}_c^r$ is the ℓ_∞ projection estimate for cluster $\mathcal{C}(r)$, which is defined as $\hat{\theta}_c^r \triangleq \arg\min_\theta \max_{(\phi_k,y_k)\in\mathcal{C}(r)} |y_k - \phi_k^T\theta|$.

Then, at time $k+1$, the online implementation procedure of HT can be given as below:

5-A1 Check whether there is a $\hat{\theta}_c^r$ that makes $|y_{k+1} - \phi_{k+1}^T\hat{\theta}_c^r| \leq \bar{v}$. If the answer is yes, go to step 5-A2; otherwise, go to step 5-A3.

5-A2 Put $(\phi_{k+1},\ y_{k+1})$ into the corresponding cluster(s) and update the associated ℓ_∞ projection estimate(s).

5-A3 Construct new clusters and obtain their ℓ_∞ projection estimates, using $(\phi_{k+1},\ y_{k+1})$ and any $(n-1)$-combination of the existing data.

For the proposed online implementation of HT, we have the following remarks:

Remark 5.2. (**On the computation**) In the implementation, we can save some extra computation by replacing the ℓ_∞ projection estimator by the least squares estimator.

Remark 5.3. (**On the memory use**) The memory used here is not affected by the number of parameters and the size of d. However, we notice that it is dependent on the number of data and the number of clusters.

Remark 5.4. (**On the consistency**) In the case of $v_k = 0$, the online implementation procedure will provide us a consistent estimation. When $v_k \neq 0$, the clustering procedure may introduce some mis-classified data and then the consistency is likely to be lost.

5.4 The HT-clustering algorithm

To further improve the performance of the online HT implementation, we develop a new online identification algorithm in this section, namely the HT-clustering algorithm. It integrates the online HT estimator with the clustering estimator by [6] with a feedback mechanism. Fig. 5.2 describes the framework of the HT-clustering online identification, which consists of three sections: HT, clustering and feedback.

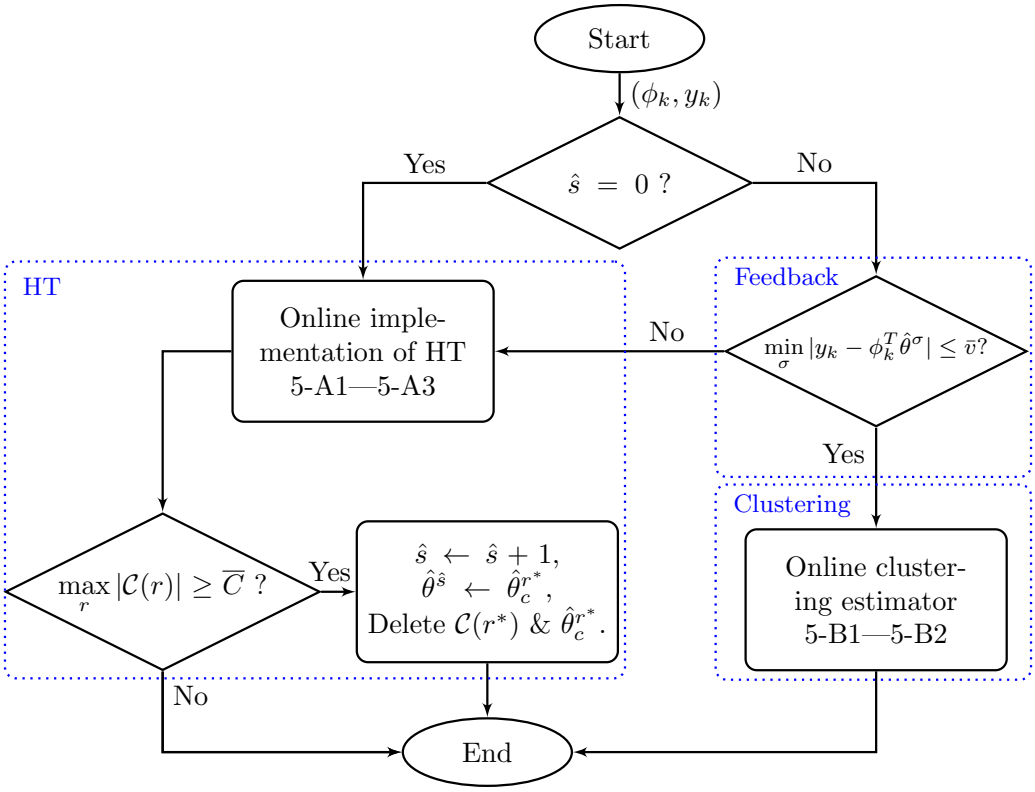


Figure 5.2: A workflow of the HT-clustering algorithm ($\hat{s}$ initially equals to 0)

5.4.1 The "HT" section

The "HT" section plays the role of feeding new parameter vectors, $\{\hat{\theta}^\sigma\}_{\sigma=1}^{\hat{s}}$, to the "clustering" section. It is invoked when the estimated number of modes $\hat{s}$ is zero or there is no $\hat{\theta}^\sigma$ that meets the following criterion on the residual: $|y_k - \phi_k^T \hat{\theta}^\sigma| \leq \bar{v}$.

In the HT section, the online implementation procedure, 5-A1—5-A3, is initially executed. It is then followed by checking the size of clusters. If there exists any $|\mathcal{C}(r)|$ larger than a size upper bound, $\overline{C}$, then we create a new parameter vector,

$$\hat{s} \leftarrow \hat{s} + 1,\ \hat{\theta}^{\hat{s}} \leftarrow \hat{\theta}_c^{r^*}, \tag{5.10}$$

and delete the cluster $\mathcal{C}(r^*)$ with $r^* = \arg\max_r |\mathcal{C}(r)|$. Note that $\overline{C}$ is used for the purpose of restricting the number of data and clusters in storage.

We are now interested in how to select an appropriate $\overline{C}$. It is actually a tradeoff problem: $\overline{C}$ should be large enough to make the $\hat{\theta}^\sigma$ sufficiently close to the true parameter vector of one mode and small enough to meet memory requirement. As the memory requirement is specified by the used hardware and uncontrollable, let's focus on the lower bound of $\overline{C}$.

It is known that we need at least $n_a + n_b$ data to decide a parameter vector for a mode. If the number of mode is s, then it implies that $\overline{C}$ should be greater than $s(n_a + n_b - 1)$ to ensure the existence of one cluster in $\{\mathcal{C}(r)\}_{r=1}^{R}$ that corresponds to a real mode.

5.4.2 The "clustering" section

The "clustering" section contains an online clustering estimator, which stems from the work in [6]. Its main function is to further update the passed parameters from the HT section. The implementation of the online clustering estimator consists of two steps: 5-B1 mode detection and 5-B2 parameter update.

At the step of mode detection, the mode number of data (ϕ_k, y_k) can be detected in several ways, see [6]. In this chapter, we use the following formula for mode detection:

$$\sigma^* = \arg\min_{\sigma=1,\ldots,\hat{s}} |y_k - \phi_k^T \hat{\theta}^\sigma|. \tag{5.11}$$

Notice that the detected mode is not guaranteed to be the same as the true one. A number of mode mismatches may exist in this stage. However, it is

apparently that if $\|\hat{\theta}^{\sigma} - \theta_0^{\sigma}\|$ is sufficiently small, then the mode mismatch will rarely happen.

At the step of parameter update, we update $\hat{\theta}^{\sigma^*}$ with the following recursive least squares procedure:

$$\begin{aligned} e_k &= y_k - \phi_k^T \hat{\theta}^{\sigma^*}, \\ P_k^{\sigma^*} &= w^{-1} P_{k-1}^{\sigma^*} \left[I - \frac{\phi_k \phi_k^T P_{k-1}^{\sigma^*}}{w + \phi_k^T P_{k-1}^{\sigma^*} \phi_k} \right], \\ \hat{\theta}^{\sigma^*} &= \hat{\theta}^{\sigma^*} + P_k^{\sigma^*} \phi_k e_k. \end{aligned} \tag{5.12}$$

where w is the forgetting factor and $P_0^{\sigma^*} = pI$ with $p \gg 1$.

5.4.3 The "feedback" section

The feedback section is a link between the online HT estimator and the online clustering estimator, which helps to achieve a high level of model fit.

Recall that the consistency of estimation in the HT section is not always guaranteed. Thus, it is necessary to activate the HT estimator from time to time. From the results in [6] and our extensive simulations, we find a few number of activations are enough to make the estimated parameters convergent to their true values.

Because the target of the identification problem is to find a group of parameter vectors $\{\hat{\theta}^{\sigma}\}_{\sigma=1}^{\hat{s}}$ such that the residuals of all sampled data are located in the interval, $[-\bar{v}, \bar{v}]$, we select the minimum magnitude of $|y_k - \phi_k^T \hat{\theta}^{\sigma}|$ as the measure to check the necessity to activate the HT section.

5.5 Simulation results

Example 5.1.

Consider the following SARX model with three third-order modes:

$$\begin{aligned} &A^{\sigma}(z) y_k = B^{\sigma}(z) u_k + v_k, \qquad \sigma \in \{1, 2, 3\}, \\ &\text{Mode 1}: \begin{cases} A^1(z) &= 1 - 0.2z^{-1} - 0.05z^{-2} + 0.006z^{-3}, \\ B^1(z) &= z^{-1} - 0.5z^{-2}, \end{cases} \\ &\text{Mode 2}: \begin{cases} A^2(z) &= 1 - 0.05z^{-1} - 0.25z^{-2} + 0.044z^{-3}, \\ B^2(z) &= -0.2z^{-1} + 0.07z^{-2}. \end{cases} \\ &\text{Mode 3}: \begin{cases} A^3(z) &= 1 + 1.8z^{-1} + 1.08z^{-2} + 0.216z^{-3}, \\ B^3(z) &= 4z^{-1} + 3.2z^{-2}. \end{cases} \end{aligned}$$

where input $u_k \sim \mathcal{N}(0,1)$ and noise $v_k \sim \mathcal{N}(0, \upsilon_0^2)$. We generated 2000 I/O data pairs; the first 1500 data are used to estimate parameters and the rest data for cross validation. The switching signal is randomly generated with an equal probability for each mode. We set the initial covariance matrix $P_0 = 1000I$, $w = 0.95$, $\bar{C} = 40$ and $\bar{v} = 3\upsilon_0$. The following measures, namely, mean squared error (MSE) and FIT, are applied for self validation and cross validation tests, respectively:

$$\text{MSE}(k) \triangleq \frac{1}{k}\sum_{i=1}^{k} \min_{\sigma=1,\ldots,\hat{s}} (y_i - \phi_i^T \hat{\theta}^{\sigma})^2,$$

$$\text{FIT} \triangleq (1 - \|\hat{\mathbf{y}} - \mathbf{y}_{1501:2000}\| / \|\mathbf{y}_{1501:2000} - \text{mean}(\mathbf{y}_{1501:2000})\|) \times 100\%,$$

where $\hat{\mathbf{y}}$ and $\mathbf{y}$ indicate the predicted and true outputs, respectively; the subscript of $\mathbf{y}$ refers to the corresponding rows of $\mathbf{y}$; mean$(\cdot)$ is the mean value vector of " $\cdot$ ". Here, the mode number of the data for validation is estimated by $\sigma_k = \arg\min_{\sigma=1,\ldots,\hat{s}} \|\mathbf{y}_k - \Phi_k \hat{\theta}^{\sigma}\|_{\infty}$.

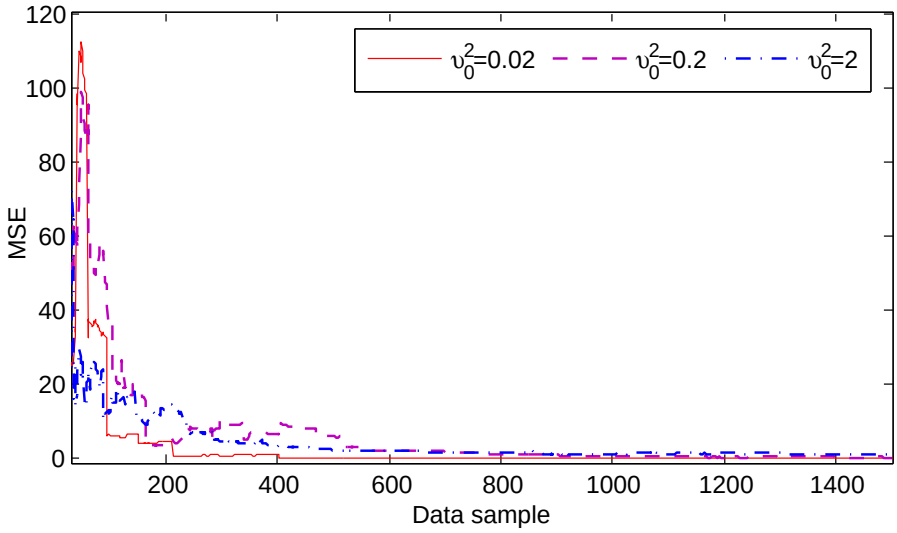


Figure 5.3: The evolution of MSE

Fig. 5.3 displays the evolution of MSE. As time goes on, the value of MSE is approaching zero very quickly. To further verify the accuracy of the estimated model, Table 5.1 reports the output fit under different noise levels. For each noise level, we run the simulation 100 times using different I/O data pairs. The mean value and standard deviation error of FIT are recorded. As can be seen from this table, the proposed algorithm is of high performance.

Table 5.1: The cross validation with 100 Monte Carlo simulations

	$\upsilon_0^2 = 2$ (SNR≈15dB)	$\upsilon_0^2 = 0.2$ (SNR≈25dB)	$\upsilon_0^2 = 0.02$ (SNR≈35dB)
FIT(%)	86.43% ± 3.3%	94.63% ± 1.9%	98.16% ± 0.62 %

Example 5.2.

The second example is concerned with nonlinear system identification problem using the HT-clustering algorithm.

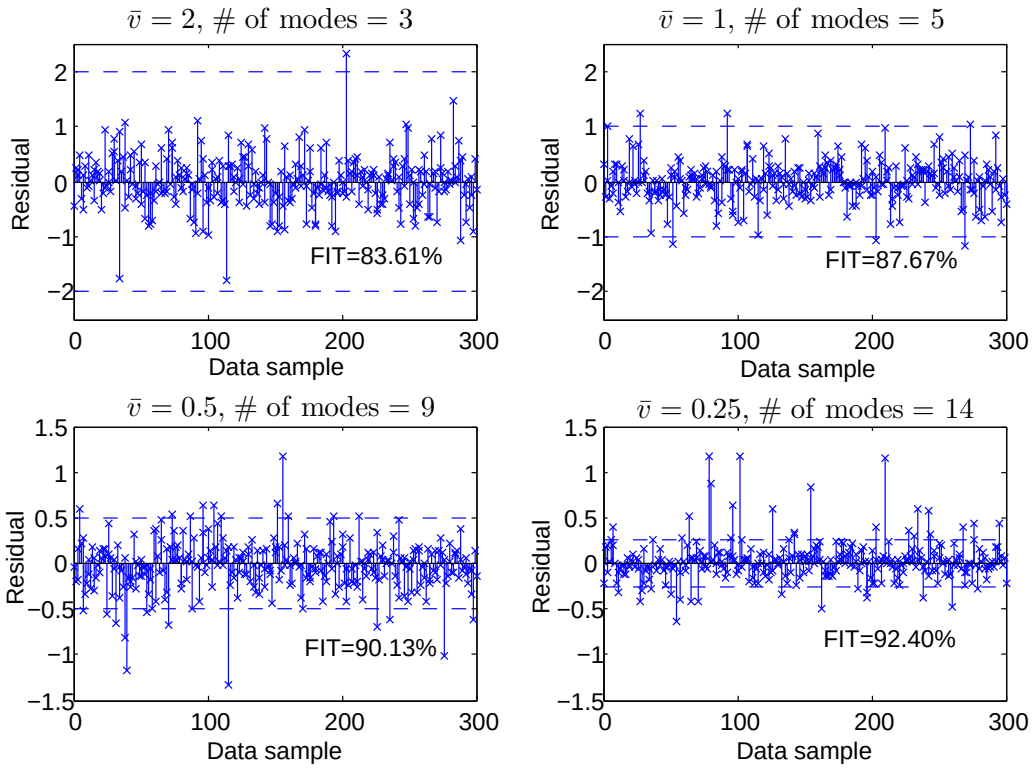


Figure 5.4: The relationship of residual, $\bar{v}$ and the estimated # of modes

We consider the following nonlinear system,

$$y_{k+1} = \frac{y_k}{1 + y_k^2} + u_k^3.$$

which is taken from [37]. Same as in Example 5.1, we let u_k to be a white Gaussian noise signal with variance equal to 1. As we aim at approximating the nonlinear system using the SARX model, the model discrepancy can be treated as the noise part. In this example, additional noise is not added. We

generated 2000 data for estimation and 300 data for cross validation. Other parameters, such as P_0, w and $\bar{C}$, are set as the same as in Example 5.1.

Fig. 5.4 shows the cross validation results w.r.t. different noise bounds. It is seen the nonlinear system can be well represented by the obtained SARX model. As $\bar{v}$ decreases and the number of modes increases, the model fit is getting improved.

5.6 Summary

In this chapter, we have studied the online identification of SLSs with the HT technique. We have shown that the proposed algorithm can identify the SLSs or some nonlinear systems without dependence on initial parameters. Due to similarity with the set membership identification, it can deal with noisy I/O data and the cases with unknown number of modes. From the simulation results, it was shown that the estimated parameters often give a high model fit w.r.t. different noise levels.

Chapter 6

Identification of Periodically Switched Linear Systems*

In previous chapters, we have discussed offline and online identification methods for SLSs without considering underlying data features in mode switching sequences. However, there exist theoretical and practical needs for study of the structure of switching sequences, as it may further improve the performance of identification in terms of efficiency and accuracy.

In this chapter, we aim at parameter estimation of periodically switched linear systems (PSLSs) by exploiting the structure of switching sequences. The addition of estimation of the period p_0 is not only critical to reduce the size of identification problem, but also helpful to eliminate the data misclassification. To fully utilize the feature in mode switching sequences, we develop a new identification method by embedding a data classification step.

This chapter is organized as follows. Section 6.1 presents the main motivation of studying the PSLSs. Section 6.2 describes the concerned system model structure and introduces the parameters estimation problem to be solved. Section 6.3 analyzes I/O data and reveals the underlying relationship between data sequences and the period p_0 to be estimated. Section 6.4 proposes two strategies to speed up the period estimation process. Section 6.5 presents both offline and online implementation methods. Finally, simulation results are given in Section 6.6 and a summary in Section 6.7.

*A version of Chapter 6 has been published in [70].

6.1 Introduction

In SLSs, when the evolution of system modes follows a specific rule, we can categorize them as a special class of SLSs. For example, in Jump Markov Linear Systems (JMLSs), mode switching follows the process of a Markov chain. An identification framework was already proposed in [22] and a nonparametric identification method was introduced in [27]. Periodically switched linear systems (PSLSs) form another special class of SLSs, in which the mode switching sequence is periodic. PSLSs have been employed in many applications; for example, periodic controllers are usually involved in control loops, giving rise to closed-loop PSLSs [28]. On the other hand, some controlled plants, such as boost DC-DC converters [1] and switched reluctance motors [72], consist of essentially a few periodically switching modes. PSLSs identification, to the best of our knowledge, has not received much attention in prior work. Although it can be solved by using some conventional methods as discussed in Chapter 1, they may not provide satisfactory estimation with efficient implementation.

The main reasons can be explained from the perspective of efficiency and accuracy. For efficiency, conventional methods require to handle all I/O data as a whole. In fact, exploiting the periodicity of switching sequences creates the possibility of estimating the period p_0 by using a finite long data sequence (to be discussed in Section 6.4). As a result, this can significantly reduce computational effort. For accuracy, conventional methods (especially clustering methods and bounded error methods) are basically not able to distinguish "undecidable" data, see [13] for further information, which will lead to some avoidable data misclassification and affect the accuracy of estimation. In contrast, data classification with a given period will be performed without such problems. These facts motivate the development of a customized identification method for PSLSs, which is the focus of this chapter.

6.2 PSLS

We consider PSLSs described by the following SARX model:

$$\begin{aligned} A^{\sigma_k}(z)y_k &= B^{\sigma_k}(z)u_k + v_k, \\ A^{\sigma_k}(z) &= 1 + a_1^{\sigma_k}z^{-1} + \cdots + a_{n_a}^{\sigma_k}z^{-n_a}, \\ B^{\sigma_k}(z) &= b_1^{\sigma_k}z^{-1} + b_2^{\sigma_k}z^{-2} + \cdots + b_{n_b}^{\sigma_k}z^{-n_b}, \end{aligned} \tag{6.1}$$

where $u_k,\ y_k$ are respectively the input and output at time k; v_k is a white Gaussian noise, i.e., $v_k \sim \mathcal{N}(0,\ v_0^2)$; $\sigma_k \in \{1, 2, \ldots, s\}$ is the switching signal of this system and the mode switching sequence $\{\sigma_1,\ \sigma_2,\ \ldots\}$ is assumed to be a periodic sequence with the period being p_0, i.e.,

$$\{\sigma_1,\ \sigma_2,\ \ldots,\ \sigma_{p_0},\ \sigma_1,\ \sigma_2,\ \ldots,\ \sigma_{p_0},\ \ldots\}; \tag{6.2}$$

We denote $\hat{\theta}^{\sigma_k}$ ($\hat{\vartheta}^{\sigma_k}$) as the (augmented) parameter vector to be estimated and ϕ_k (φ_k) the (augmented) regression vector, which take the form,

$$\begin{aligned}
&\hat{\theta}^{\sigma_k} = [\hat{a}_1^{\sigma_k} \ \ \cdots \ \ \hat{a}_{n_a}^{\sigma_k} \ \ \hat{b}_1^{\sigma_k} \ \ \cdots \ \ \hat{b}_{n_b}^{\sigma_k}]^T \in \mathbb{R}^n,\\
&\hat{\vartheta}^{\sigma_k} = [1 \ \ -(\hat{\theta}^{\sigma_k})^T]^T \in \mathbb{R}^{n+1},\\
&\phi_k = [-y_{k-1} \ \cdots \ -y_{k-n_a} \ u_{k-1} \ \cdots \ u_{k-n_b}]^T \in \mathbb{R}^n,\\
&\varphi_k = [y_k \ \ \phi_k^T]^T \in \mathbb{R}^{n+1},
\end{aligned} \tag{6.3}$$

where $n = n_a + n_b$. Same as in the previous chapters, $\theta_0^{\sigma_k}$ (or $\vartheta_0^{\sigma_k}$) is referred to as the true parameter vector and $\hat{p}_0 \in \mathbb{Z}^+$ is a period candidate of p_0 with $2 \le \hat{p}_0 \le \overline{p}_0$. The prediction error is represented by

$$\begin{aligned}
e_k &= y_k - \phi_k^T \hat{\theta}^{\sigma_k},\\
&= \varphi_k^T \hat{\vartheta}^{\sigma_k}.
\end{aligned} \tag{6.4}$$

Based on (6.4), prediction error methods can be directly applied for parameter estimation, if the I/O data are associated with the same mode.

When p_0 is known, it can be easily achieved by classifying the data into p_0 groups. For example, we divide the output data sequence $\{y_1,\ y_2,\ \ldots\ \}$ into the following subsequences:

$$\{y_1,\ y_{\hat{p}_0+1},\ \ldots\},\ \{y_2,\ y_{\hat{p}_0+2},\ \ldots\},\ \ldots,\ \{y_{\hat{p}_0},\ y_{2\hat{p}_0},\ \ldots\}. \tag{6.5}$$

where we let $\hat{p}_0 = p_0$. For $r \in \{1,\ 2,\ \ldots, \hat{p}_0\}$, we define a new index sequence $\{t_i : i = 1,\ 2,\ \ldots\}$ as follows:

$$t_1 = r,\ t_2 = \hat{p}_0 + r,\ t_3 = 2\hat{p}_0 + r,\ \ldots \tag{6.6}$$

With this notation, the subsequences in (6.5) can be generally represented by $\{y_{t_1},\ y_{t_2},\ \ldots\}$ when r ranges over the set $\{1,\ 2,\ \ldots, \hat{p}_0\}$. Similarly, we may represent the subsequences of ϕ_k as $\{\phi_{t_1},\ \phi_{t_2},\ \ldots\}$.

Then, for the r^{th} group of data, the parameter vector $\hat{\theta}^r$ can be estimated by the least squares estimator:

$$\hat{\theta}^r = [\sum_{i=1}^{\infty} \phi_{t_i} \phi_{t_i}^T]^{-1} \sum_{i=1}^{\infty} \phi_{t_i} y_{t_i}. \tag{6.7}$$

However, when p_0 is unknown, the parameter estimation problem is much more difficult. Since no prior knowledge is available for the system modes, we have to extract underlying information from the mixed I/O data. The following sections will mainly focus on this issue.

Before we proceed, we want to mention that the superscripts of $\hat{\theta}^{\sigma_k}$ ($\theta_0^{\sigma_k}$, $\hat{\vartheta}^{\sigma_k}$ or $\vartheta_0^{\sigma_k}$) will be omitted in the following context if it is not necessary.

6.3 Analysis of I/O data

In this section, we analyze the I/O data characteristics of the concerned system in (6.1) and establish the potential connection with p_0. To this end, the regression vector φ_k, containing both input and output data, shall be studied.

Considering a PSLS in (6.1) without noise, i.e., $v_k = 0$, we have,

$$\varphi_k^T \vartheta_0 = 0. \tag{6.8}$$

This property only corresponds to the data at time k. If a batch of data are associated with the same mode, we must be able to find a $\hat{\vartheta}$ satisfying (6.8) for all data. Again, we divide the data sequence $\{\varphi_k | k = 1, 2, \ldots\}$ into $\hat{p}_0$ groups as shown in (6.9).

$$\begin{bmatrix} \varphi_1^T \\ \varphi_{\hat{p}_0+1}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_2^T \\ \varphi_{\hat{p}_0+2}^T \\ \vdots \end{bmatrix}, \ \ldots, \ \begin{bmatrix} \varphi_{\hat{p}_0}^T \\ \varphi_{2\hat{p}_0}^T \\ \vdots \end{bmatrix}. \tag{6.9}$$

By replacing the subscripts with the index sequence in (6.6), the above data sequences are generally indicated by $\{\varphi_{t_i} |\ i = 1, 2, \ldots\}$.

We now define a concept against to the data sequences in (6.9) and it helps us to distinguish the data with different modes.

Definition 6.1 (Identity)**.** *For the PSLS in* (6.1) *with* $v_k = 0$*, if there exists one* $\hat{\vartheta}$ *such that*

$$\begin{bmatrix} \varphi_{t_1}^T \\ \varphi_{t_2}^T \\ \vdots \end{bmatrix} \hat{\vartheta} = 0, \tag{6.10}$$

then, we call the data sequence $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ *identical.*

Remark 6.1. If we can not find a $\hat{\vartheta}$ satisfying (6.10), it means that the data in $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ are with at least two different modes and $p_0 \neq \hat{p}_0$.

Regarding the PSLS in (6.1) with $v_k \neq 0$, even if all data in $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ are collected from the same mode, condition (6.10) may not be satisfied, due to the impact of v_k. Thus, we modify Definition 6.1 for noisy systems. As in set membership identification, we assume that the magnitude of measurement noise v_k is bounded by $\bar{v}$. Consequently, the definition of identity for stochastic systems is addressed as follows:

Definition 6.2. *For the PSLSs in* (6.1) *with* $|v_k| \leq \bar{v}$*, if the following linear inequalities are solvable,*

$$-\bar{v} \preceq \begin{bmatrix} \varphi_{t_1}^T \\ \varphi_{t_2}^T \\ \vdots \end{bmatrix} \hat{\vartheta} \preceq \bar{v}, \tag{6.11}$$

then, we call the data sequence $\{\varphi_{t_i}|i = 1, 2, \ldots\}$ *identical.*

Definitions 6.1 and 6.2 are basically the same; the only difference is that Definition 6.1 is used for noise free systems and Definition 6.2 for noisy systems. In this chapter, we will pay more attention to noisy systems and analysis is mostly based on Definition 6.2. Moreover, it is remarked that if inputs $u_k, u_{k-1}, \ldots, u_{k-n_b}$ in φ_k are persistently exciting of order $n_a + n_b$, that is, the I/O data set is informative enough [44], then a solvable (6.10) implies that its solution $\hat{\vartheta}$ is unique.

Geometrically, each inequality in (6.11) forms a narrow space (or called 'hyperstrip', see [13]) between two hyperplanes, which is denoted by

$$S_{t_i} = \{x \in \mathbb{R}^{n+1}| -\bar{v} \leq \varphi_{t_i}^T x \leq \bar{v}\},\ i \in \mathbb{Z}^+. \tag{6.12}$$

Thus, the set of solutions to (6.11) consists of a convex polytope as follows,

$$P = \{x \in \mathbb{R}^{n+1}| -\bar{v} \leq \varphi_{t_i}^T x \leq \bar{v},\ i = 1, 2, ...\} \tag{6.13}$$

Clearly, we have $P = S_{t_1} \cap S_{t_2} \cap \cdots$ as shown in Fig. 6.1, where S_{t_1}, S_{t_2} and S_{t_3} enclose ϑ_0 inside of P.

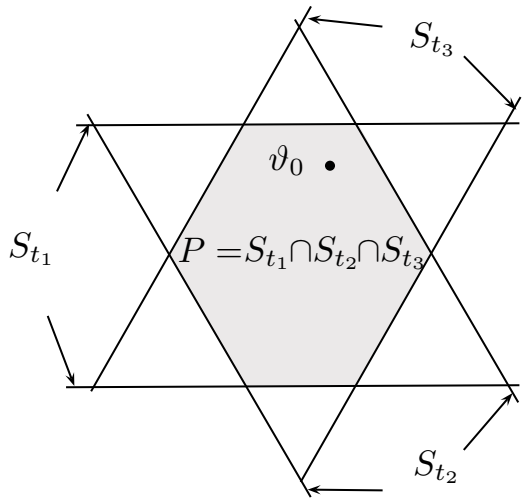


Figure 6.1: Geometric explanation (two dimensional cases)

Remark 6.2. From Fig. 6.1, we can claim that if $\{\varphi_{t_i} | i = 1, 2, \ldots\}$ is not identical, that is, $P = \emptyset$, then there is at least one data point corresponding to a different mode. However, the converse statement is not true. An identical data sequence doesn't mean that all data are collected from the same mode; for instance, there could be two different ϑ_0 in P.

We're now interested in the question: under what conditions will the converse be correct? This will be the key to connect identity with periodicity. To this end, we pay attention to the possibility of a narrow space that passes through more than one ϑ_0. In Fig. 6.2, we still consider a two-dimension case and assume that polytope P^i is constructed by a data sequence $\{\varphi_{t_i} | i = 1, 2, \ldots, q-1, q+1, \ldots\}$, in which the data are associated with mode i; while the data φ_{t_q} comes from mode j and it forms a space S_{t_q}. The distance between these two different modes is represented by $D_{\{i,\ j\}} = \|\vartheta_0^i - \vartheta_0^j\|_2$. For polytope P^i, it has a minimal sphere containing P^i, whose radius is R^i.

Fig. 6.2 shows that S_{t_q} may pass through P^i if S_{t_q} is in the position of $S_{t_{q'}}$. The probability of this is dependent on two factors:

(F1) $R^i / D_{\{i,\ j\}}$;

(F2) the randomness of φ_k.

In fact, R^i further relies on (F2) and $\bar{v}$. Then, we have the following result.

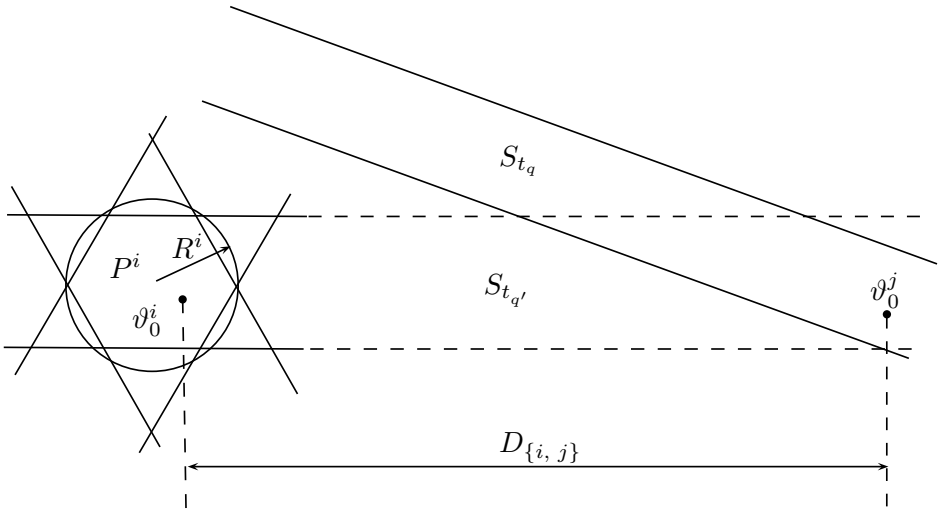


Figure 6.2: The probability of S_{t_q} containing ϑ_0^i and ϑ_0^j

Theorem 6.1. *If inputs $u_k, u_{k-1}, \ldots, u_{k-n_b}$ in φ_k are persistently exciting of order $n_a + n_b$ and $\bar{v}/D_{\{i,\ j\}}$ is sufficiently small for all different $i,\ j \in \{1, 2, \ldots, s\}$, then*

(1) $Pr\{S_{t_q} \cap P_i = \emptyset\} \approx 1$;

(2) the data in $\{\varphi_{t_i} |\ i = 1, 2, \ldots\}$ are almost surely with a single mode if and only if the data sequence is identical.

Proof. Firstly, let's compute the probability (indicated by Pr) of ϑ_0^i belong to S_{t_q}, which is equal to

$$\begin{aligned} &\Pr\{|\varphi_{t_q}^T \vartheta_0^i| \le \bar{v} \cap |\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\} \\ =&\Pr\{|\varphi_{t_q}^T \vartheta_0^i| \le \bar{v} \mid |\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\} \Pr\{|\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\}. \end{aligned} \tag{6.14}$$

We now calculate $\Pr\{|\varphi_{t_q}^T \vartheta_0^i| \le \bar{v} \mid |\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\}$ and start from a simple case when $\bar{v} = 0$. In this case, the thickness of S_{t_q} is also 0, since it is proportional to the value of $\bar{v}$, see (6.12). Then, the hyperstrip S_{t_q} becomes a hyperplane that passes through ϑ_0^j and its position is specified by φ_{t_i}, see Fig. 6.2. Define $\sin\alpha = d/D_{\{i,\ j\}}$, where d is the distance from ϑ_0^i to S_{t_q}. We see that S_{t_q} passes through ϑ_0^i only when $\sin\alpha = 0$. On the other hand, we can easily understand that $\sin\alpha$ is dependent on φ_{t_i} and the value of $\sin\alpha$ is then bounded in a finite interval $[\underline{\alpha},\ \bar{\alpha}]$ with $\bar{\alpha} - \underline{\alpha} > 0$. As $u_k, u_{k-1}, \ldots, u_{k-n_b}$ in φ_k are persistently exciting of order $n_a + n_b$, $\sin\alpha$ will generally not be

equal to zero. Thus, we clearly have $\Pr\{\sin\alpha = 0\} = 0$, which implies $\Pr\{|\varphi_{t_q}^T \vartheta_0^i| \le \bar{v} \mid |\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\} = 0$ when $\bar{v} = 0$.

When $\bar{v} \neq 0$, we can get a similar derivation. In this situation, only if $\sin\alpha \in [\underline{\alpha}',\ \bar{\alpha}']$, ϑ_0^i and ϑ_0^j are both contained in S_{t_q}. Here, $[\underline{\alpha}',\ \bar{\alpha}']$ belongs to $[\underline{\alpha},\ \bar{\alpha}]$ and the length of $[\underline{\alpha}',\ \bar{\alpha}']$ depends on the value of $\bar{v}/D_{\{i,\ j\}}$. When $\bar{v}/D_{\{i,\ j\}}$ is sufficiently small, we get $(\bar{\alpha}' - \underline{\alpha}')/(\bar{\alpha} - \underline{\alpha}) \approx 0$ and this means

$$\Pr\{|\varphi_{t_q}^T \vartheta_0^i| \le \bar{v} \ \mid \ |\varphi_{t_q}^T \vartheta_0^j| \le \bar{v}\} \approx 0. \tag{6.15}$$

From (6.14) and (6.15), we immediately have $\Pr\{\vartheta_0^i \in S_{t_q}\} \approx 0$.

For S_{t_q} and ϑ_0^j, polytope P_i can be considered as a point as long as $\bar{v}/D_{\{i,\ j\}}$ is sufficiently small. Any point $x \in P_i$ has the same property as ϑ_0^i, i.e.,

$$\Pr\{|\varphi_{t_q}^T x| > \bar{v}\} \gg \Pr\{|\varphi_{t_q}^T x| \le \bar{v}\}. \tag{6.16}$$

Therefore, we have $\Pr\{S_{t_q} \cap P_i = \emptyset\} \approx 1$.

Theorem 6.1 (2) follows easily from part 1). □

Remark 6.3. Theorem 6.1 suggests an approach to estimate p_0. By checking the identity of all data sequences in (6.9) for $\hat{p}_0 = 2, 3, \ldots, \overline{p}_0$, then p_0 is equal to the minimum $\hat{p}_0$ such that all its data sequences are identical.

6.4 Implementation strategies

Following the idea mentioned in the preceding section, the calculation would be heavy if the period upper bound is very large or the data sequence is extremely long. Is there an efficient way to implement this? Or, can we avoid checking $\hat{p}_0$ one by one? Another question is: for a long data sequence, is it possible to evaluate some partial data instead of all? These questions will be investigated in this section.

• Reverse order search

In most cases, due to the lack of prior system knowledge, we have to set up a large upper bound for $\hat{p}_0$ to guarantee $p_0 \le \overline{p}_0$, which means we need to handle a large candidate set of p_0, i.e., $\{\hat{p}_0 |\ \hat{p}_0 = 2, 3, \ldots, \overline{p}_0\}$.

For different candidates of p_0, their data sequences are usually very different and there is no evident relationship among them. However, if $\hat{p}_0 \in \mathbb{Z}^+$ is

a period of a data sequence, then $m\hat{p}_0,\ m \in \mathbb{Z}^+$ will also be a period. Let "mod" denote the modulus operation, i.e., a mod b is equal to the remainder of a/b. Thus, the following result is obtained.

Theorem 6.2. *For a period candidate $\hat{p}_0 \in \mathbb{Z}^+$, if one of its data sequences $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ is not identical, then $p_0 \notin \{x|\ \hat{p}_0\ mod\ x = 0\}$.*

Proof. Assuming $p_0 \in \{x|\ \hat{p}_0 \text{ mod } x = 0\}$ and hence $\hat{p}_0 = mp_0,\ m \in \mathbb{Z}^+$, their data sequences can be respectively written as

$$\{\varphi_r,\ \varphi_{r+p_0},\ \varphi_{r+2p_0},\ \ldots\}, \tag{6.17}$$

$$\{\varphi_r,\ \varphi_{r+mp_0},\ \varphi_{r+2mp_0},\ \ldots\}. \tag{6.18}$$

Clearly, sequence in (6.18) is a subsequence of (6.17). If (6.18) is not identical, then (6.17) is not identical as well. This contradicts the initial assumption. Hence Theorem 6.2 holds. □

The following example is used to highlight this point.

Example 6.1.

$$\hat{p}_0 = 2: \begin{bmatrix} \varphi_1^T \\ \varphi_3^T \\ \varphi_5^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_2^T \\ \varphi_4^T \\ \varphi_6^T \\ \vdots \end{bmatrix};\ \hat{p}_0 = 3: \begin{bmatrix} \varphi_1^T \\ \varphi_4^T \\ \varphi_7^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_2^T \\ \varphi_5^T \\ \varphi_8^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_3^T \\ \varphi_6^T \\ \varphi_9^T \\ \vdots \end{bmatrix};$$

$$\hat{p}_0 = 6: \begin{bmatrix} \varphi_1^T \\ \varphi_7^T \\ \varphi_{13}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_2^T \\ \varphi_8^T \\ \varphi_{14}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_3^T \\ \varphi_9^T \\ \varphi_{15}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_4^T \\ \varphi_{10}^T \\ \varphi_{16}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_5^T \\ \varphi_{11}^T \\ \varphi_{17}^T \\ \vdots \end{bmatrix}, \begin{bmatrix} \varphi_6^T \\ \varphi_{12}^T \\ \varphi_{18}^T \\ \vdots \end{bmatrix}.$$

By inspecting $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ in this example, we may find that each data sequence in the case of $\hat{p}_0 = 6$ is a subsequence of one data sequence with $\hat{p}_0 = 2$ and $\hat{p}_0 = 3$. Therefore, $p_0 \neq 6$ immediately implies $p_0 \neq 2$ and $p_0 \neq 3$.

Owing to Theorem 6.2, we shall check period candidate set $\{\hat{p}_0|\ \hat{p}_0 = 2, 3, \ldots, \overline{p}_0\}$ from $\overline{p}_0$ to 2, that is, in the reverse order. If data sequences of $\hat{p}_0$ are not all identical, then we can remove a few candidates from the candidate set as it suggested; if they are, we shrink the search range to the divisors of $\hat{p}_0$. Then p_0 is the smallest divisor whose data sequences are still identical.

• **Finite data selection**

The reverse order search strategy reduces the examination times on $\hat{p}_0$. However, for an individual $\hat{p}_0$, the computation could also be expensive because of a long data sequence. As the identity is possible to be decided by a small portion of data, we now investigate how much data is necessary to determine the identity of a data sequence.

Firstly, we give an important result about the mode switching sequence of $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$.

Theorem 6.3. *For any period candidate $\hat{p}_0 \in \mathbb{Z}^+$, the mode switching sequence of $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ is also periodic with period p_0.*

Proof. As we know, for the whole data sequence $\{\varphi_k|\ k = 1, 2, \ldots\}$, the mode switching sequence in (6.2) is periodic with $\sigma_k = \sigma_{k+mp_0},\ m \in \mathbb{Z}^+$.

For a subsequence $\{\varphi_{t_i}|\ i = 1,\ 2,\ \ldots\ \}$, as the index sequence is defined as $t_1 = r,\ t_2 = r + \hat{p}_0,\ \ldots,\ t_{p_0} = r + (p_0 - 1)\hat{p}_0,\ t_{p_0+1} = r + p_0\hat{p}_0,\ \ldots$, its mode switching sequence should be in the following form:

$$\{\sigma_r,\ \sigma_{r+\hat{p}_0},\ \ldots,\ \sigma_{r+(p_0-1)\hat{p}_0},\ \sigma_{r+p_0\hat{p}_0},\ \ldots\}.$$

Therefore, in $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$, the mode corresponding to $(t_k)^{\text{th}}$ data is the same as that of $(t_{k+mp_0})^{\text{th}}$ data. This completes the proof. □

Remark 6.4. Given Theorem 6.1 and Theorem 6.3, we can state that if $\{\varphi_{t_i}|\ i = 1, 2, \ldots\}$ is not identical, there must exist two data points in every p_0 data set that are associated with two different modes. Furthermore, since the probability of S_{t_i} that contains more than two different ϑ_0 is nearly zero, the identity of an infinite data sequence $\{\varphi_{t_i}|i = 1, 2, \ldots\}$ is a.s. consistent with that of its finite subsequence $\{\varphi_{t_i}|\ i = 1, 2, \ldots,\ \widetilde{p}\}$ for $\widetilde{p} \geq p_0$.

In practice, despite the inputs are constrained within a limited range, the consistency of identity can still be guaranteed by considering a few periods of data. We can choose $\widetilde{p} = 2\overline{p}_0 \sim 5\overline{p}_0$.

• **Selection of $\bar{p}_0$**

We now go back to discuss the selection of $\bar{p}_0$. As mentioned above, $\overline{p}_0$ is difficult to be set precisely, if prior system knowledge is unavailable. However,

we can actually find a possible range of $\overline{p}_0$. One lower bound is equal to the number of mode s, since each mode should be visited at least once during one switching period. The upper bound of $\overline{p}_0$ can be inferred by the potential requirement on the data length for each mode. If the total number of I/O data is N and input signals are persistently exciting of order $n_a + n_b$, a mode with model order n_a and n_b requires at least $n_a + n_b$ I/O data to identify its parameters. Hence, we have $\overline{p}_0 \leq N/\lceil \frac{n_a+n_b}{z_0} \rceil$, where z_0 is the minimum number of data that corresponds to the same mode in one switching period; $\lceil x \rceil$ is the ceiling function that gives the smallest integer larger than x.

6.5 Implementation methods

In this section, we introduce offline and online implementation methods to examine identity and estimate parameters. For the offline case, we regard all data as a whole and identity is determined based on all data, while in the online case the identity is evaluated at each sample time instant.

For offline implementation, there are two kinds of methods: one is the singular value decomposition (SVD) method and the other is an optimization-based method. Both are in terms of the definitions on identity.

• Via SVD

In the noise free case, according to Definition 6.1, if the data $\{\varphi_{t_i} | i = 1, 2, \ldots\}$ are in general positions, there is only one solution satisfies (6.10). In other words,

$$\text{nullity} \left(\begin{bmatrix} \varphi_{t_1}^T \\ \varphi_{t_2}^T \\ \vdots \end{bmatrix} \right) = 1. \tag{6.19}$$

We can easily get nullity by checking the rank of this matrix.

For stochastic systems, the nullity may be always equal to 0 due to the impact of noise. Therefore, we apply SVD to the above matrix as suggested in [68]. The singular values are denoted as: $\varsigma_1 \geq \varsigma_2 \geq \ldots \varsigma_{n-1} \geq \varsigma_n$.

When $\hat{p}_0 = mp_0$, the smallest singular value (ς_n) is attributed only by v_k. Therefore, the ratio of the last two singular values, i.e., $\varepsilon = \varsigma_{n-1}/\varsigma_n$, should be much larger than that when $\hat{p}_0 \neq mp_0$.

• **Via optimization**

For stochastic systems, as suggested by Definition 6.2, an alternative way to judge identity is to check the solvability of (6.11) directly, which can be achieved by solving a linear or quadratic programming problem with the constraints in (6.11). For example, we formulate the following quadratic programming (QP) problem:

$$\begin{aligned} \min_{\hat{\vartheta}} \quad & f(\hat{\vartheta}) = \hat{\vartheta}^T \hat{\vartheta} \\ \text{s.t.} \quad & -\bar{v} \preceq \begin{bmatrix} \varphi_{t_1}^T \\ \varphi_{t_2}^T \\ \vdots \end{bmatrix} \hat{\vartheta} \preceq \bar{v} \end{aligned} \tag{6.20}$$

The dual active-set algorithm [11] is an efficient approach to solve this QP problem.

Remark 6.5. Comparing the SVD method with an optimization-based method, they both require some prescribed tolerance (ε or $\bar{v}$) due to the uncertainty caused by noise. In practice, we recommend to use optimization-based methods, since $\bar{v}$ is related to y_k and it is more straightforward to define.

For the offline implementation, given an estimate of p_0, we can calculate $\hat{\theta}^r$ ($r = 1, 2, \ldots, \hat{p}_0$) by the formula in (6.7). Among the estimated $\hat{\theta}^r$, some of them might be very similar to each other and correspond to the same mode. The measure of similarity, in this situation, is defined below, as suggested in [52].

$$M_{\{r,\ r'\}} = \frac{\|\hat{\theta}^r - \hat{\theta}^{r'}\|_2}{\min\{\hat{\theta}^r,\ \hat{\theta}^{r'}\}} \tag{6.21}$$

For a given threshold γ_1, if $M_{\{r,\ r'\}} > \gamma_1$ holds for all $r,\ r' \in \{1, 2, \ldots, \hat{p}_0\}$, then we have $s = \hat{p}_0$; otherwise, we merge data sequence r and r' with $M_{\{r,\ r'\}} \leq \gamma_1$ and re-estimate the parameters.

In what follows, the online implementation is considered. We present a recursive vertex enumeration (VE) method, similar to [17], to examine identity.

• **Via recursive VE**

As discussed in Section 6.3, the solutions of (6.11) consist of a polytope P, which has two representation forms: H (half-space)-representation and V

(vertex)-representation. The problem that generates a V-representation from a known H-representation is referred to as the vertex enumeration, while a recursive VE method solves the vertex enumeration problem at each sample time instant.

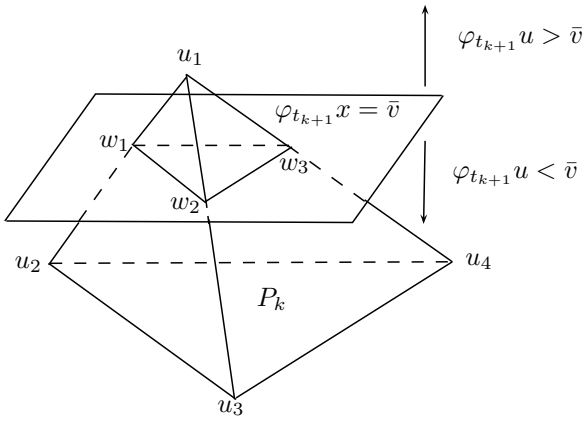


Figure 6.3: An illustration of the recursive VE method

Let P_k denote the polytope at time k, given by an intersection of half spaces $-\bar{v} \leq \varphi_{t_i}^T x \leq \bar{v},\ i = 1, 2, \ldots, k$, and $V(P_k)$ the vertex set of P_k. adj(u) is the adjacency list of vertex u and act(u) stores all indices of active constraints at vertex u. In addition, the operator "$|\cdot|$" returns the cardinality of a set. The basic idea of this method is illustrated in Fig. 6.3.

Given a polytope at time t_k, it is divided by a hyperplane specified by $\varphi_{t_{k+1}} x = \bar{v}$ into two parts. $V^-(P_k)$ is the vertex set that contains vertices such that $\varphi_{t_{k+1}} u \leq \bar{v}$. Similarly, we define $V^+(P_k)$ for vertices such that $\varphi_{t_{k+1}} u > \bar{v}$. If $|V^-(P_k)| = 0$, then the data sequence is not identical; if $|V^+(P_k)| = 0$, then the incoming constraint is redundant. In other cases, we create a new vertex set W ($W = \{w_1,\ w_2,\ w_3\}$ in Fig. 6.3) and then update $V(P_k)$, adj(u) as well as act(u). The algorithm details are provided in Algorithm 3 (on the next page).

Remark 6.6. Algorithm 3 has two steps. The first step is used to generate a large polytope, which contains $\{\vartheta_0^i\}_{i=1}^s$ inside. The second step is the implementation of vertex enumeration that is based on the polytope generated in step 1.

Algorithm 3 Recursive Vertex Enumeration

Step 1. Initialization step

Set up an initial polytope $V(P_0)$ with the following H-representation:

$$L \preceq \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} x \preceq U,$$

where L, U are the lower and upper bounds of parameters.

Step 2. Iteration step

for $k = 1 : \widetilde{p}$ **do**
  Determine $V^-(P_k)$ and $V^+(P_k)$
  if $|V^-(P_k)| = 0$ **then**
    the data sequence is not identical
    return
  else if $|V^+(P_k)| = 0$ **then**
    the new constraint is redundant
  else
    for $u \in V^-(P_k)$ **do**
      for all $v \in \mathrm{adj}(u) \cap V^+(P_k)$ **do**
        calculate new vertices w
        $W \leftarrow W \cup w$
        $\mathrm{adj}(u) \leftarrow \mathrm{adj}(u) \setminus v \cup w$ and $\mathrm{adj}(w) \leftarrow u$
        $\mathrm{act}(w) = \mathrm{act}(u) \cap \mathrm{act}(v) \cup \{k\}$
      end for
    end for
  end if
  for all $u, v \in W$ **do**
    if $|\mathrm{act}(u) \cap \mathrm{act}(v)| = n - 2$ **then**
      $\mathrm{adj}(u) \leftarrow \mathrm{adj}(u) \cup v$
      $\mathrm{adj}(v) \leftarrow \mathrm{adj}(v) \cup u$
    end if
  end for
  $V(P_k) \leftarrow V^-(P_k) \cup W$
end for

In the online case, $\hat{\theta}^r$ can be obtained in the same manner as we did for the offline case. However, if $s < \hat{p}_0$, we are unable to merge the data sequences by comparing the similarity of $\hat{\theta}^r$, since data in an online case are not repeatedly available. In other words, we have to perform the merging task prior to estimation of parameters.

By virtue of the recursive VE method, we have access to the vertices of polytope $V(P_k)$ and it should contain ϑ_0^r inside. Consequently, we can distinguish data sequences by resorting to the similarity of vertices. For mode r, $V(P_k^r)$ is referred to as the corresponding vertex set. Then, the measure of similarity in vertices is define by

$$M'_{\{r,\ r'\}} = \frac{\|A(P_k^r) - A(P_k^{r'})\|_2}{\min\{A(P_k^r), A(P_k^{r'})\}}, \tag{6.22}$$

where

$$A(P_k^r) = \frac{1}{|V(P_k^r)|} \sum_{u \in V(P_k^r)} u \tag{6.23}$$

denotes the average of the vertices in $V(P_k^r)$ and $r,\ r' \in \{1, 2, \ldots, \hat{p}_0\}$. For a given threshold γ_2, if $M'_{\{r,\ r'\}} \leq \gamma_2$, we then combine data sequence r with r'. Parameters are subsequently estimated by the recursive least squares algorithm. We refer readers to [45] for further details about this algorithm.

6.6 Simulation results

To show the effectiveness of the proposed approaches, the following second-order system with three modes is used:

$$\begin{aligned}
&A^{\sigma_k}(z)y_k = B^{\sigma_k}(z)u_k + v_k,\ \sigma_k \in \{1,\ 2,\ 3\} \\
&\text{Mode 1}: \begin{cases} A^1(z) &= 1 + 0.1z^{-1} + 0.3z^{-2}, \\ B^1(z) &= 4z^{-1} + 1.5z^{-2}, \end{cases} \\
&\text{Mode 2}: \begin{cases} A^2(z) &= 1 - 0.2z^{-1} + 0.5z^{-2}, \\ B^2(z) &= 2z^{-1} + 5z^{-2}. \end{cases} \\
&\text{Mode 3}: \begin{cases} A^3(z) &= 1 - 0.7z^{-1} + 0.4z^{-2}, \\ B^3(z) &= 1.5z^{-1} - 3z^{-2}. \end{cases}
\end{aligned}$$

In the simulations, the input u_k is a zero-mean white Gaussian noise signal with unity variance. The disturbance v_k is also a white Gaussian noise with

zero mean and the signal to noise ratio (SNR) being 35 dB. The mode switching sequence is $\{1,2,3,1,3,2;1,2,3,1,3,2;\ldots\}$. This system is simulated for 1500 I/O data pairs. The first 1000 data are used to estimate parameters and the rest of them are for model validation. In addition, we let $\overline{p}=10$, $\widetilde{p}=2\overline{p}$ and $\bar{v}=0.5$.

Before doing the simulation, we would like to give a pre-analysis of this example. Let's start period estimation from $\hat{p}_0=10$. The following displays the data sequences and the corresponding mode switching sequences when $\hat{p}_0=10,\ 9,\ 8,\ 7$, respectively. For brevity, only two data sequences of each are listed here and it is enough for the analysis on this example though.

$$
\begin{array}{cccc}
(\hat{p}_0=10) & (\hat{p}_0=9) & (\hat{p}_0=8) & (\hat{p}_0=7) \\
\begin{bmatrix}\varphi_1^T\\ \varphi_{11}^T\\ \varphi_{21}^T\\ \vdots\end{bmatrix}\ \begin{bmatrix}\varphi_2^T\\ \varphi_{12}^T\\ \varphi_{22}^T\\ \vdots\end{bmatrix}, &
\begin{bmatrix}\varphi_1^T\\ \varphi_{10}^T\\ \varphi_{19}^T\\ \vdots\end{bmatrix}\ \begin{bmatrix}\varphi_2^T\\ \varphi_{11}^T\\ \varphi_{20}^T\\ \vdots\end{bmatrix}, &
\begin{bmatrix}\varphi_1^T\\ \varphi_9^T\\ \varphi_{17}^T\\ \vdots\end{bmatrix}\ \begin{bmatrix}\varphi_2^T\\ \varphi_{10}^T\\ \varphi_{18}^T\\ \vdots\end{bmatrix}, &
\begin{bmatrix}\varphi_1^T\\ \varphi_8^T\\ \varphi_{15}^T\\ \vdots\end{bmatrix}\ \begin{bmatrix}\varphi_2^T\\ \varphi_9^T\\ \varphi_{16}^T\\ \vdots\end{bmatrix}, \\
\begin{bmatrix}1\\3\\3\\ \vdots\end{bmatrix}\ \begin{bmatrix}2\\2\\1\\ \vdots\end{bmatrix}, &
\begin{bmatrix}1\\1\\1\\ \vdots\end{bmatrix}\ \begin{bmatrix}2\\3\\2\\ \vdots\end{bmatrix}, &
\begin{bmatrix}1\\3\\3\\ \vdots\end{bmatrix}\ \begin{bmatrix}2\\1\\2\\ \vdots\end{bmatrix}, &
\begin{bmatrix}1\\2\\3\\ \vdots\end{bmatrix}\ \begin{bmatrix}2\\3\\1\\ \vdots\end{bmatrix}.
\end{array}
$$

Focusing on the last row above, we observe that none of the data sequences, except one with $\hat{p}_0=9$, is identical. Hence, we have $p_0\notin\{7,\ 8,\ 9,\ 10\}$ and it further implies that $p_0\notin\{2,\ 3,\ 4,\ 5\}$ according to Theorem 6.2. Hence, the only possibility is $p_0=6$.

6.6.1 Period estimation

We now verify the above results using the simulation. Fig. 6.4 illustrates the period estimation by the SVD method. Again, two data subsequences are selected and the ratio $\varsigma_{n-1}/\varsigma_n$ is calculated for each period candidate. Only at $\hat{p}_0=6$, both data sequences are with high values in $\varsigma_{n-1}/\varsigma_n$, which in turn means they are identical and it verifies the results we obtained in the pre-analysis.

Fig. 6.5 shows the simulation results by using the dual active set method and the recursive VE method. Since these methods only return a judgement on the identity of data sequences, we need to define an indicator function as

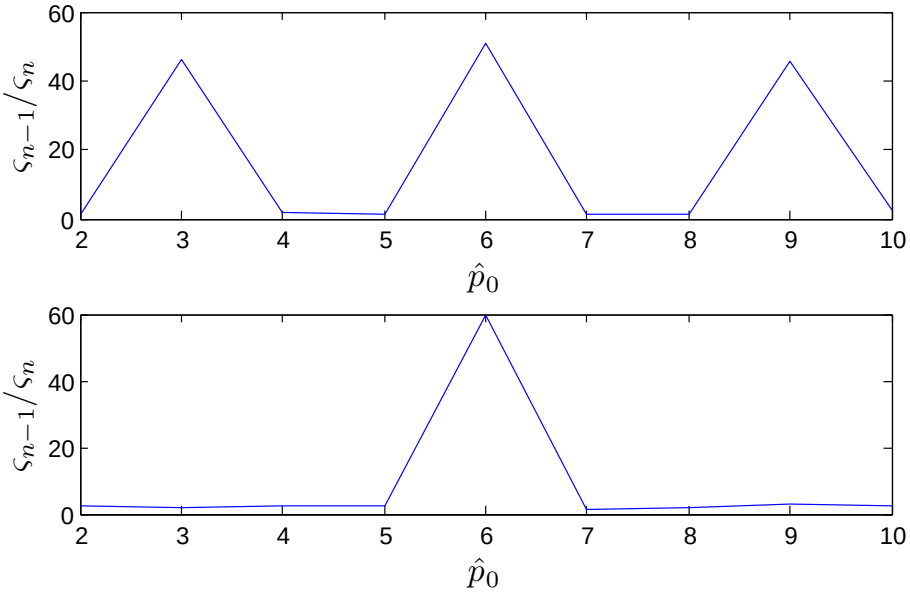


Figure 6.4: Period estimation via SVD

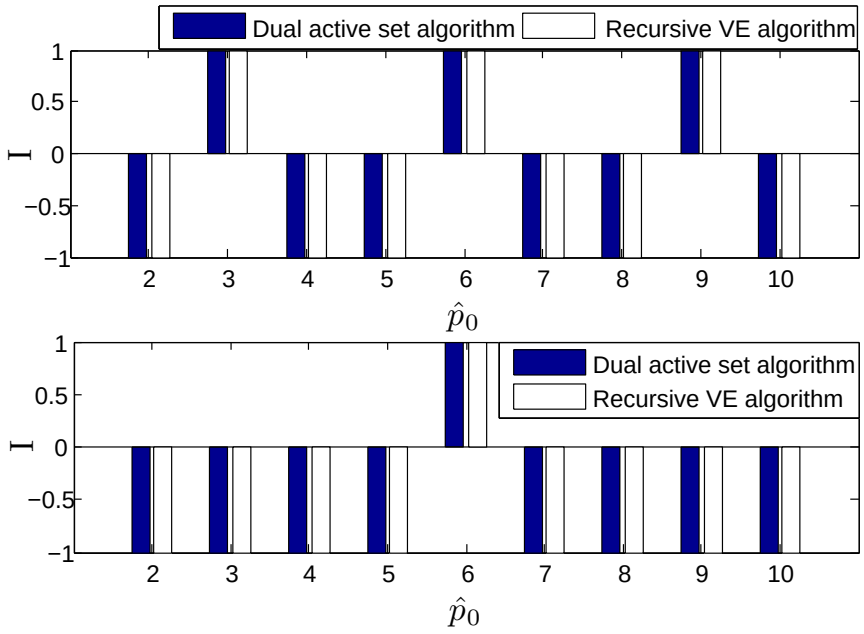


Figure 6.5: Period estimation via dual active-set algorithm & VE algorithom

below to quantify this:

$$I = \begin{cases} 1, & \{\varphi_{t_i} | i = 1, 2, \ldots, \widetilde{p}\} \text{ is identical,} \\ -1, & \text{otherwise.} \end{cases} \tag{6.24}$$

Comparing Fig. 6.5 with Fig. 6.4, they're consistent on the identity of data

sequences. Also, all these methods are able to generate correct estimation on the period, i.e., $p_0 = 6$.

6.6.2 Parameters estimation

Before proceeding to estimation of parameters, the similarity of $\hat{\theta}^r$ for offline case and $A(P_k^r)$ for online case are calculated. As illustrated in Fig. 6.6, each node represents a $\hat{\theta}^r$ or $A(P_k^r)$ $(r = 1, 2, \ldots, 6)$. The weights of the edges, obtained by formula (6.21) or (6.22), give the quantified similarity. The smaller the value, the larger the similarity.

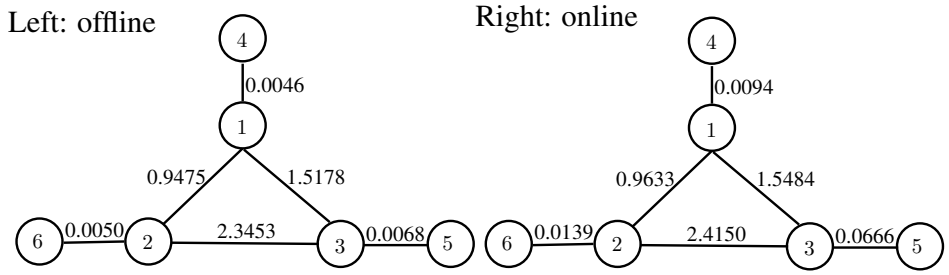


Figure 6.6: Similarity of $\hat{\theta}^{\sigma_r}$ (offline) or $A(P_k^{\sigma_r})$ (online)

For similar nodes, we combine the associated data sequences and then perform offline or online recursive least squares parameter estimation. Table 6.1 reports the the normalized errors between the estimated parameters and the true parameters.

Table 6.1: Parameters accuracy computed by $(\|\hat{\theta}^i - \theta_0^i\|_2 / \|\theta_0^i\|_2) \times 100\%$

	Mode 1	Mode 2	Mode 3
offline estimation	0.05%	0.23%	0.33%
online estimation	0.07%	0.16%	0.32%

Using the estimated model, we finally carry out a validation test to compare the predicted outputs with the true outputs. The output difference is evaluated by

$$\frac{\|\hat{Y} - Y\|_2}{\|Y\|_2} \times 100\%,$$

where $Y \in \mathbb{R}^{500}$ and $\hat{Y} \in \mathbb{R}^{500}$ are the output vectors generated by the true and estimated models, respectively. For offline and online estimation, we obtain that the output difference values are both as low as 1.93%.

6.7 Summary

In this chapter, we focused on parameter estimation of a special class of switched linear systems, in which the evolution of mode makes up a periodic sequence. From the geometric point of view, the relationship between a data sequence and the mode switching period p_0 is investigated and it has been proven that period p_0 can be accurately estimated. Meanwhile, we proposed two efficient implementation strategies to reduce the complexity of computation and introduced effective approaches to estimate the period and parameters for both offline and online cases.

We close by mentioning that the exploration of the underlying properties in the mode switching sequence is beneficial for increase accuracy of I/O data classification, and similarly we can also apply this idea to other special class of SLSs.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this work, identification of SLSs has been fully investigated and analysed. We have developed a few robust identification methods for SLSs with an unknown number of modes.

Firstly, in the offline identification framework, we developed an RH solution in Chapter 2, which is very effective to solve the MAX FS problem. For SLSs with multiple unknown noise levels, in Chapter 3, we proposed an extended RH solution. It can be used to estimate parameters as well as noise bounds. Secondly, in Chapters 4 and 5, we have presented two online identification algorithms, namely, MRLS and HT-clustering, which can be applied in adaptive control or other circumstances that require parameter update in real time. The newly developed online algorithms are based on a clustering approach [6], since it is simple and effective for online data classification. Compared with the early work, the proposed MRLS algorithm has a better performance, because it can significantly reduce the negative effects caused by mode mismatches. By taking advantage of the HT technique, the HT-clustering algorithm is not only applicable for SLSs with an unknown number of modes, but also works well for some nonlinear systems that can be approximated by SLSs. Lastly, in Chapter 6, we put efforts on the identification of periodically switched linear systems, which exist in many applications. By exploring the data feature of mode switching sequences, model parameters can be estimated more efficiently and accurately than the existing methods.

To conclude, the proposed methods in the PhD thesis have made an important progress in the identification of SLSs. They have both theoretical

and practical significance. The thesis work provides researchers and engineers more powerful tools to build mathematical models for SLSs or some nonlinear systems. Moreover, it lays groundwork for future improvements.

7.2 Future work

The research on the identification of SLSs is still at the developing stage and there are many issues that have not fully explored. We would like to provide a few interesting ones that are valuable to be further studied.

• Global optimization

In Chapter 1, it is mentioned that identification of SLSs becomes quite complicated when the number of modes and the switching signal are both unknown. In such a situation, we have to trade off the model quality and complexity in terms of prediction errors and estimated number of modes. In [5, 13, 51] and Chapter 2, the set membership identification approach was applied, where we imposed a noise bound to limit the number of modes. It is actually a suboptimal way to identify a model.

In fact, we may design a cost function to characterize the extent of model accuracy and complexity at the same time. For example, the identification problem can be set up in the following form:

$$\begin{aligned} \min_{\theta^i, w_k^i, s} \quad & \sum_{k=\bar{n}}^{N} \sum_{i=1}^{s} \ell(y_k - \phi_k^T \theta^i) w_k^i + \gamma f(s) \\ \text{s.t.} \quad & \sum_{i=1}^{s} w_k^i = 1 \quad \forall k \\ & w_k^i \in \{0, 1\} \quad \forall k, i \end{aligned}$$

where $f(s)$ is a function of s and γ is a weighting factor; other notation is defined in the same way as in Chapter 1. Though the above problem appears difficult to solve, it does help reducing data mis-classification and hence improving the performance of estimation.

• Region partition in the regressor domain

The PWARX model, mentioned in Chapter 1, is another popular type of models being studied, which can describe SLSs with mode switches dependent

on the position of the regressor. Identification of a PWARX model includes parameter estimation as well as finding a complete partition of the regressor domain. For this, there are two kinds of approaches. One is to find a boundary between any two modes with a linear classifier. It can be efficiently implemented, but the obtained partition is not a complete one—there may exist a black hole in the regressor domain. The other approach is to apply a piecewise linear classifier that discriminates all modes; this is able to make a complete partition, but the computational complexity is much higher. Designing a more advanced approach is highly desirable.

• Identification of JMLSs

In Chapter 6, we have discussed identification of periodically switched linear systems. The idea is to use underlying patterns of switching sequences to estimate parameters. For some types of SLSs, such as, JMLSs, this idea is still effective. In JMLSs, the switching signal is controlled by a transition probability matrix; identification of the transition probabilities should be helpful to improve the estimation accuracy. One feasible approach is perhaps to match the estimated probabilities with the one obtained by estimated switching signals. Related work can be found in a few references, see, e.g., [22, 33]. However, there still exist some issues that are not explored yet. For example, estimating a transition probability matrix without knowing the number of modes, and finding a transition probability matrix in the PWARX model. More research effort is required to have a better understanding.

Bibliography

[1] S. Almer, S. Mariethoz, and M. Morari. Piecewise affine modeling and control of a step-up DC-DC converter. In *Proc. American Control Conf.*, pages 3299–3344, Baltimore, MD, USA, 2010.

[2] E. Amaldi and M. Mattavelli. The MIN PFS problem and piecewise linear model estimation. *Discrete Applied Mathematics*, 118:115–143, 2002.

[3] A. Atkinson and M. Riani. *Robust Diagnostic Regression Analysis*. Springer-Verlag, New York, USA, 2000.

[4] A. Atkinson, M. Riani, and A. Cerioli. *Exploring Multivariate Data with the Forward Search*. Springer-Verlag, New York, USA, 2004.

[5] L. Bako. Identification of switched linear systems via sparse optimization. *Automatica*, 47(4):668–677, 2011.

[6] L. Bako, K. Boukharouba, E. Duviella, and S. Lecoeuche. A recursive identification algorithm for switched linear/affine models. *Nonlinear Analysis: Hybrid Systems*, 5(2):242–253, 2011.

[7] L. Bako, G. Mercre, and S. Lecoeuche. A least squares approach to the subspace identification problem. In *Proc. 47th IEEE Conf. on Decision and Control*, pages 3281–3286, Cancun, Mexico, 2008.

[8] L. Bako, G. Mercre, and S. Lecoeuche. On-line structured subspace identification with application to switched linear systems. *International Journal of Control*, 82(8):1496–1515, 2009.

[9] L. Bako, G. Mercre, R.Vidal, and S. Lecoeuche. Identification of switched linear state space models without dwell time. In *Proc. 15th IFAC Symposium on System Identification*, vol. 15, pages 569–574, Saint-Malo, France, 2009.

[10] L. Bako and R. Vidal. Algebraic identification of MIMO SARX models. In *Proc. 11th international workshop on Hybrid Systems*, pages 43–57, St. Louis, MO, USA, 2008. Springer-Verlag.

[11] R. A. Bartlett and L. T. Biegler. QPSchur: A dual, active-set, Schur complement method for large-scale and structured convex quadratic programming. *Optimization and Engineering*, 7(1):5–32, 2006.

[12] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino. Set membership identification of piecewise affine models. In *Proc. 13th IFAC Symposium on System Identification*, pages 1789–1794, Rotterdam, The Netherlands, 2003.

[13] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino. A bounded-error approach to piecewise affine system identification. *IEEE Trans. on Automatic Control*, 50(10):1567–1580, 2005.

[14] J. Borges, V. Verdult, and M. Verhaegen. Iterative subspace identification of piecewise linear systems. In *Proc. 14th IFAC Symposium on System Identification*, pages 368–373, Newcastle, Australia, 2006.

[15] J. Borges, V. Verdult, M. Verhaegen, and M. A. Botto. A switching detection method based on projected subspace classification. In *Proc. 44th IEEE Conf. on Decision and Control*, pages 344–349, Seville, Spain, 2005.

[16] R. Carloni, R. G. Sanfelice, A. R. Teel, and C. Melchiorri. A hybrid control strategy for robust contact detection and force regulation. In *Proc. American Control Conf.*, pages 1461–1466, New York City, USA, July 2007.

[17] P. C. Chen, P. Hansen, and B. Jaumard. On-line and off-line vertex enumeration by adjacency lists. *Operations Research Letters*, 10(7):403–409, 1991.

[18] Y. Chen and Y. Wu. Modified recursive least-squares algorithm for parameter identification. *International Journal of Systems Science*, 23(2):187–205, 1992.

[19] J. W. Chinneck. Fast heuristics for the maximum feasible subsystem problem. *INFORMS Journal on Computing*, 13(3):210–223, 2001.

[20] J. W. Chinneck. *Feasibility and Infeasibility in optimization: Algorithms and Computational Methods*. Springer Science+Business Media, LLC, New York, USA, 2008.

[21] E. Cinquemani, A. M. Argeitis, and J. Lygeros. Identification of genetic regulatory networks: a stochastic hybrid approach. In *Proc. 17th IFAC World Congress*, pages 301–306, Seoul, Korea, 2008.

[22] E. Cinquemani, R. Porreca, G. Ferrari-Trecate, and J. Lygeros. A general framework for the identification of jump Markov linear systems. In *Proc. 46th IEEE Conf. on Decision and Control*, pages 5737–5742, New Orleans, USA, 2007.

[23] F. Ding and T. Chen. Combined parameter and output estimation of dual-rate systems using an auxiliary model. *Automatica*, 40(10):1739–1748, 2004.

[24] E. Domlan, B. Huang, J. Ragot, and D. Maquin. Robust identification of switched regression models. *IET Control Theory and Applications*, 3(12):1578–1590, 2009.

[25] P. Egbunonu and M. Guay. Identification of switched linear systems using subspace and integer programming techniques. *Nonlinear Analysis: Hybrid Systems*, 1(4):577–592, 2007.

[26] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. *Automatica*, 39(2):205–217, 2003.

[27] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. Nonparametric Bayesian identification of jump systems with sparse dependencies. In *Proc. 15th IFAC Symposium on System Identification*, vol. 15, pages 1591–1596, Saint-Malo, France, 2009.

[28] S. Galeani, O. M. Grasselli, and L. Menini. Strong stabilization with infinite multivariable gain margin through linear periodic control. *International Journal of Control*, 77(5):441–460, 2004.

[29] A. Germani, C. Manes, and P. Palumbo. Simultaneous system identification and channel estimation: a hybrid system approach. In *Proc. 46th IEEE Conference on Decision and Control*, pages 1764–1769, 2007.

[30] W. Glover and J. Lygeros. A stochastic hybrid model for air traffic control simulation. In R. Alur and G. J. Pappas, editors, *Hybrid Systems: Computation and Control*, vol. 2993 of *Lecture Notes in Computer Science*, pages 372–386. Springer-Verlag, New York, 2004.

[31] A. Goldenshluger and A. Zeevi. The Hough transform estimator. *The Annals of Statistics*, 32(5):1908–1932, 2004.

[32] G. C. Goodwin and K. S. Sin. *Adaptive Filtering Prediction and Control*. Prentice-Hall, Englewood Cliffs, N.J., 1984.

[33] X. Jin and B. Huang. Identification of switched Markov autoregressive eXogenous systems with hidden switching state. *Automatica*, 48(2), 2012.

[34] X. Jin and B. Huang. Robust identification of piecewise/switching autoregressive exogenous process. *AICHE Journal*, 56(7):1829–1844, 2012.

[35] A. L. Juloski, W. P. M. H. Heemels, G. Ferrari-Trecate, R. Vidal, S. Paoletti, and J. H. G. Niessen. Comparison of four procedures for the identification of hybrid systems. In *Proc. of the 8th international conference on Hybrid Systems: Computation and Control*, HSCC'05, pages 354–369, Zurich, Switzerland, 2005.

[36] A. L. Juloski, S. Weiland, and W. P. M. H. Heemels. A Bayesian approach to identification of hybrid systems. *IEEE Trans. on Automatic Control*, 50(10):1520–1533, 2005.

[37] C. Y. Lai. *Identification and Control of Nonlinear Systems using Multiple Models*. PhD thesis, National University of Singapore, Singapore, 2011.

[38] H. K. Lam, F. H. F. Leung, and P. K. S. Tam. A switching controller for uncertain nonlinear systems. *IEEE Control Systems*, 22(1):7–14, 2002.

[39] F. Lauer and G. Bloch. A new hybrid system identification algorithm with automatic tuning. In *Proc. of the 17th World Congress, IFAC*, pages 10207–10212, Seoul, Korea, 2008.

[40] J. Lee, S. Bohacek, J. P. Hespanha, and K. Obraczka. Modeling communication networks with hybrid systems. *IEEE/ACM Trans. on Networking*, 15(3):630–643, 2007.

[41] B. Lennartson, M. Tittus, B. Egardt, and S. Pettersson. Hybrid systems in process control. *IEEE Control Systems Magazine*, 16(5):45–56, 1996.

[42] Z. G. Li, C. Y. Wen, and Y. C. Soh. Switched controllers and their applications in bilinear systems. *Automatica*, 37(3):477–481, 2001.

[43] H. H. Liao, A. Widd, N. Ravi, A. F. Jungkunz, J. M. Kang, and J. C. Gerdes. Control of recompression HCCI with a three region switching controller. *Control Engineering Practice*, 21(2):135–145, 2013.

[44] L. Ljung. *System Identification: Theory for the User*. PTR Prentice Hall, Upper Saddle River, NJ, 2 edition, 1999.

[45] L. Ljung and T. Soderstrom. *Theory and Practice of Recursive Identification*. MIT Press, Cambridge, MA, 1983.

[46] Y. Ma and R. Vidal. Identification of deterministic switched ARX systems via identification of algebraic varieties. In *Proc. 8th International Conference on Hybrid Systems: Computation and Control*, vol. 3414, pages 449–465, Zurich, Switzerland, 2005.

[47] M. Milanese and A. Vicino. Optimal estimation theory for dynamic systems with set membership uncertainty: an overview. *Automatica*, 27(6):997–1009, 1991.

[48] H. Nakada, K. Takaba, and T. Katayama. Identification of piecewise affine systems based on statistical clustering technique. *Automatica*, 41(5):905–913, 2005.

[49] N. N. Nandola and S. Bhartiya. A multiple model approach for predictive control of nonlinear hybrid systems. *Journal of Process Control*, 18(2):131–148, 2008.

[50] R. H. Nyström, K. V. Sandström, T. K. Gustafsson, and H. T. Toivonen. Multimodel robust control of nonlinear plants: a case study. *Journal of Process Control*, 9(2):135–150, 1999.

[51] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps. A sparsification approach to set membership identification of switched affine systems. *IEEE Trans. on Automatic Control*, 57(3):634–648, 2012.

[52] S. Paoletti, A. L. Juloski, G. Ferrari-Trecate, and R. Vidal. Identification of hybrid systems: a tutorial. *European Journal of Control*, 13(2-3):242–260, 2007.

[53] K. M. Pekpe, G. Mourot, K. Gasso, and J. Ragot. Identification of switching systems using change detection technique in the subspace framework. In *Proc. 43rd IEEE Conf. on Decision and Control*, vol. 4, pages 3720–3725, Paradise Island, Bahamas, 2004.

[54] M. Prandini and J. Hu. Application of reachability analysis for stochastic hybrid systems to aircraft conflict prediction. *IEEE Trans. on Automatic Control*, 54(4):913–917, 2009.

[55] B. Pregelj and S. Gerkšič. Hybrid explicit model predictive control of a nonlinear process approximated with a piecewise affine model. *Journal of Process Control*, 20(7):832–839, 2010.

[56] M. Riani, A. Atkinson, and A. Cerioli. Finding an unknown number of multivariate outliers. *Journal of the Royal Statistical Society: series B (statistical methodology)*, 71(2):447–466, 2009.

[57] J. Roll, A. Bemporad, and L. Ljung. Identification of piecewise affine systems via mixed-integer programming. *Automatica*, 40(1):37–50, 2004.

[58] P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

[59] P. J. Rousseeuw and K. Van Driessen. Computing LTS regression for large data sets. *Data Mining and Knowledge Discovery*, 12:29–45, 2006.

[60] T. Schlegl, M. Buss, and G. Schmidt. A hybrid systems approach toward modeling and dynamical simulation of dextrous manipulation. *IEEE/ASME Trans. on Mechatronics*, 8(3):352–361, sep. 2003.

[61] V. Solo. The convergence of AML. *IEEE Trans. on Automatic Control,*, 24(6):958–962, 1979.

[62] W. Tan, H. J. Marquez, T. Chen, and J. Liu. Multimodel analysis and controller design for nonlinear processes. *Computers and Chemical Engineering*, 28(12):2667–2675, 2004.

[63] A. Thomasson and L. Eriksson. Model-based throttle control using static compensators and pole placement. *Oil & Gas Science and Technology - Rev. IFP Energies nouvelles*, 66(4):717–727, 2011.

[64] A. Uppal, W. H. Ray, and A. B. Poore. On the dynamic behavior of continuous stirred tank reactors. *Chemical Engineering Science*, 29:967–985, 1974.

[65] M. Vašak, L. Mladenović, and N. Perić. Clustering-based identification of a piecewise affine electronic throttle model. In *31st Annual Conf. of IEEE Industrial Electronics*, pages 177–182, Raleigh, USA, Nov. 2005.

[66] V. Verdult and M. Verhaegen. Subspace identification of piecewise linear systems. In *Proc. 43rd IEEE Conf. on Decision and Control*, vol. 4, pages 3838–3843, Paradise Island, Bahamas, 2004.

[67] R. Vidal. Recursive identification of switched ARX systems. *Automatica*, 44(9):2274–2287, 2008.

[68] R. Vidal, S. Soatto, Y. Ma, and S. Sastry. An algebraic geometric approach to the identification of a class of linear hybrid systems. In *Proc. 42nd IEEE Conf. on Decision and Control*, vol. 1, pages 167–172, Maui, USA, 2003.

[69] J. Wang and T. Chen. Online identification of switched linear output error models. In *2011 IEEE International Symposium on Computer-Aided Control System Design (CACSD)*, pages 1379–1384, Denver, USA, Sept. 2011.

[70] J. Wang and T. Chen. Parameter estimation of periodically switched linear systems. *IET Control Theory Applications*, 6(6):768–775, 12 2012.

[71] J. Wang and T. Chen. Identification of switched linear systems with multiple unknown noise levels. *Systems & Control Letters*, 2013. Submitted for publication.

[72] H. Yang, B. Jiang, and V. Cocquempot. A fault tolerant control framework for periodic switched non-linear systems. *International Journal of Control*, 82(1):117–129, 2009.

[73] W. Zhao and T. Zhou. Weighted least squares based recursive parametric identification for the submodels of a PWARX system. *Automatica*, 47(4):668–677, 2012.