

## **MIT Document Services**

Room 14-0551  
77 Massachusetts Avenue  
Cambridge, MA 02139  
ph: 617/253-5668 | fx: 617/253-1690  
email: docs@mit.edu  
<http://libraries.mit.edu/docs>

### **DISCLAIMER OF QUALITY**

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort to provide you with the best copy available. If you are dissatisfied with this product and find it unusable, please contact Document Services as soon as possible.

Thank you.

DUE TO THE POOR QUALITY OF THE ORIGINAL THERE IS  
SOME SPOTTING OR BACKGROUND SHADING ON THIS THESIS.

**On-line Identification and Control Algorithm for System  
Model with Jump Parameters Using Wavelets**

by

Fadi Nabih Karamah  
B.E., American University of Beirut (1993)

Submitted to the Department of Electrical Engineering and Computer Science  
in partial fulfillment of the requirements for the degree of

Master of Science  
in  
Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1996

©Massachusetts Institute of Technology 1996. All Rights Reserved.

Author .....  
Department of Electrical Engineering and Computer Science  
May 17, 1996

Certified by .....  
Munther A. Dahleh  
Associate Professor of Electrical Engineering  
Thesis Supervisor

Accepted by .....  
Frederic R. Morgenthaler  
Chairman, Departmental Committee on Graduate Students



**On-line Identification and Control Algorithm for System Model with Jump  
Parameters Using Wavelets**

by

Fadi Nabih Karameh

Submitted to the Department of Electrical Engineering and Computer Science  
on May 18, 1996, in partial fulfillment of the requirements for the degree of  
**Master of Science**

**ABSTRACT**

On-line parametric identification of time-varying systems using basis expansion methods is often challenged with the need for isolating and locking into different rates of model variations while keeping the complexity of the estimation relatively low. In recent years, the localization properties of wavelets have offered an efficient and compact description of details as well as global characterizations in signal processing. Under such a motive, this research develops an on-line block recursive algorithm that isolates and estimates step changes in an *ARX* system model using Haar wavelets as basis expansion functions for the time-varying parameters' profile. The recursive algorithm uses statistical tests to move from coarse to finer description of each parameter when localized behavior is prevalent, as well as to improve global estimate when no changes occur. Subsequently, indirect adaptive control of the plant employs a similar multiresolution approach to design a (time-varying) controller to stabilize the changing system via frozen time methods and to achieve tracking. The performance of this algorithm is finally simulated to demonstrate both the estimation and the control schemes.

Thesis Supervisor: Munther A. Dahleh  
Title: Associate Professor of Electrical Engineering



## Acknowledgments

Many Thanks to God, *Alhamdu liLlah*, for everything.

I deeply thank my advisor Prof. Munther Dahleh, for his support and confidence in me. In addition to being very flexible with my choice of the topic, he was always there when I needed his tutelage. He was also there for me, sometimes to express my anguish and others to feel good about things, always supportive and understanding. I have seen his uncanny insight, deep knowledge, humor, and enthusiasm in teaching classes, simplifying complex problems, and organizing lectures, and in all he never ceased to impress me.

This research was made possible by, and to whom I am grateful, fundings from NSF 9157306-ECS and AFOSR F49620-95-0219.

The list of people who lighted the tunnel for me is long. It starts in Lebanon with my professors at AUB, who helped and encouraged me to come to MIT, and continues with the many MIT faculty, LIDS friendly environment, staff, and students.

My friends at LIDS has devoted invaluable hours that inevitably aided in getting me focused, being both the resource and simply the soundingboard. "Lets do something with wavelets" was the first idea, and then comes the learning process. I would like to thank my labmates, Mike Branicky, Stefano Casadei, Seema Jaggi, Sanjay Lall, Steve Patek, Sekhar Taticonda, and especially my office mates: Han Chen (for putting up with most of my whining), Jorge Goncalves, and Sean Warnick who were all always there when I needed to talk to someone; Ibrahim Abou Faycal, Nicola Elia, Hamid Krim, Wesley McDermott, Fernando Paganini, and Saligrama Venkatesh. They have given me help and stimulating discussion ground as I set to understand wavelets, system identification and adaptive control. I also want to thank Joaquin-Lacalle Melero (for all those beer talks), Julio Castrillon, Jeffery Chapman, and Marcos Escobar; the TA group Beth Gerstein, O. Patrick Kreidl, Ahmed Mitwalli, Rama Mukkamala, and Dewey Tucker for fun times.

Of course the Lebanese "gang" at MIT has an invaluable role in providing a much needed social activity. All the events we organized, the folk dance I've learned along with this thesis, and the laughs we shared were very important to put aside work problems occasionally. I am very grateful for their distractions and would like to thank them all.

The invisible hand of, and the spirit of support which my family provided me with, helped me survive difficult times. My wonderful parents, to whom I owe it all, who raised

my ambition, will to search higher grounds, the good man in me: my father will always be my example of morals, wisdom and insight, and my mother who taught me to prey and to love. My young sister who is a lovely image of my own past; my two elder sisters, who bought me candies when I was too shy, now along with their husbands and youngsters all fill my heart with incomparable joy. Although far away, I could hear my family's prayers, and feel their longing.... Thank you and God bless you all.

*To my parents,  
Nabih and Nawal*

# Contents

<b>1</b>	<b>Introduction</b>	<b>9</b>
1.1	Motivation . . . . .	9
1.2	Outline of the Thesis . . . . .	11
<b>2</b>	<b>Background and Problem Definition</b>	<b>13</b>
2.1	Problem Setup . . . . .	13
2.1.1	Plant Specification . . . . .	13
2.1.2	Controller Specification . . . . .	14
2.2	Time Varying System Identification . . . . .	16
2.2.1	Recursive Forgetting Factor Methods . . . . .	16
2.2.2	Basis Expansion Methods . . . . .	18
2.3	Adaptive Control . . . . .	18
2.3.1	Model Reference Controllers . . . . .	19
2.3.2	Self Tuning Controllers . . . . .	20
<b>3</b>	<b>Wavelets in System Identification</b>	<b>21</b>
3.1	What are Wavelets . . . . .	21
3.1.1	Perfect Reconstruction Filter Banks . . . . .	23
3.2	Previous Work by Tsatsanis et al . . . . .	26
3.3	Why Wavelets . . . . .	28
<b>4</b>	<b>The Identification and Control Algorithm</b>	<b>31</b>
4.1	Assumptions . . . . .	32
4.2	Identification on a Single Interval . . . . .	33
4.2.1	Algorithm Setup . . . . .	33
4.2.2	Searching for Relevant Resolutions . . . . .	38



4.2.3	Efficient Multichannel Order Update Algorithm . . . . .	42
4.3	Block Recursive Identification . . . . .	45
4.3.1	Moving into Lower Resolutions . . . . .	47
4.3.2	Hypothesis Test . . . . .	51
4.3.3	The Block Recursive Algorithm . . . . .	53
4.4	Closing the Loop: the Control Algorithm . . . . .	55
<b>5</b>	<b>Simulations and Analysis</b>	<b>59</b>
5.1	Open Loop Simulations . . . . .	59
5.1.1	Comparison with Standard Methods . . . . .	62
5.1.2	Extensions to Other Applications . . . . .	62
5.2	Closed Loop Simulations . . . . .	64
5.3	Convergence and Stability . . . . .	66
<b>6</b>	<b>Conclusions</b>	<b>69</b>
6.1	Summary . . . . .	69
6.2	Directions of Future Research . . . . .	70

# List of Figures

2-1	Block diagram of the closed loop adaptive control design . . . . .	16
2-2	General model of adaptive control mechanisms. . . . .	18
2-3	Adaptive control mechanisms: (a) model reference adaptive system MRAS and, (b) self tuning regulator STR . . . . .	19
3-1	Decomposition of the time frequency plane: (a) Fourier (b) wavelets . . . . .	22
3-2	Example of the Haar wavelet and scaling function . . . . .	23
3-3	Dyadic perfect reconstruction filter bank. Shown are the analysis and syn- thesis stages . . . . .	24
3-4	The nobel identity . . . . .	25
3-5	Cascade structure corresponding to a multiresloution of depth $j = 2$ . . . . .	25
3-6	Equivalent filters for the cascade in figure 3-5 . . . . .	25
4-1	Filter interpretation of the basis expansion . . . . .	44
4-2	Example of moving into lower resolutions (higher indexed subspaces) . . . . .	49
4-3	Decision regions for jump change between two intervals . . . . .	52
4-4	General outline of the identification algorithm . . . . .	54
4-5	Flow chart of the single interval VALIDATE algorithm . . . . .	57
4-6	Flow chart of block recursive identification algorithm . . . . .	58
5-1	Parameter estimates for the ARX model (case 1) . . . . .	60
5-2	Effect of noise increase in short identification intervals on parameter estimates of ARX model (case 2) . . . . .	61
5-3	Improved parameter estimates for the ARX model (case 2) as longer disjoint intervals are used . . . . .	61

5-4	Comparison of the proposed algorithm with recursive ARX algorithm with constant forgetting factor . . . . .	63
5-5	Estimates of the plant parameters operating in closed loop . . . . .	65
5-6	Plant output ( $y$ ) and tracking error ( $e$ ) under frozen time controller design	65
5-7	Tuning parameter $Q$ for the frozen time controller design . . . . .	66

# Chapter 1

## Introduction

### 1.1 Motivation

The problem of time varying parametric system identification has been treated using several approaches. A popular approach employs adaptive algorithms where it is assumed that the output process is not too far from stationary so that the evolution of the system, being rather slow, can be tracked within reasonable time. Other approaches assume more explicit modeling of the system coefficients' time evolution and use kalman filtering techniques to estimate the system trajectory.

Parametric identification of time varying systems can also be performed by expanding the system model coefficients onto a finite set of basis functions and, hence, transforming the problem into that of identifying a set of time invariant linear weights of the basis functions. Although the number of parameters to be identified will increase, this seems to be a little price to pay in keeping the problem time invariant. With such an approach, it is desirable to retain a low order solution and to achieve a parsimonious expansion where both the global and local behavior of the system trajectory are captured in a sparse representation which lends itself to an efficient on-line identification of the model.

Wavelet theory has become an effective tool to develop compact representations of signals at different scales. Applications have been extensive in image coding where wavelets provided a concise or compact representation with good accuracy. On the same track, a proper choice of wavelets basis in parameter identification has been demonstrated in Tsatsanis et al [1] to offer parsimonious representation without sacrificing tracking capability. This is due to the local nature of wavelets (scaled and translated versions of a single function) which will describe local details better than other expansions and avoid end effects

and discontinuities associated with, for example, Fourier basis. The analogy can be carried to image coding since, in expanding onto wavelet basis, more details or higher resolutions will offer a “zoom in” into the places where local transitions occur (much like edge detection in images) whereas most of the information of the global behavior is retained in low resolutions.

This thesis will utilize the Haar basis functions which are one type of the wavelet family as a natural choice to model and control an on-line operating system with jumps or abrupt changes in some of its parameters. Such jumps can be thought of as parametric failures, and hence they describe a transition mode which is reasonably assumed to be less frequent than a normal operating constant-parameter mode. Therefore, the latter (normal operation) mode can be represented at low resolution Haar basis whereas more details or higher resolutions are utilized to represent abrupt changes. This will lead to a desirable sparse basis expansion which is the building block for on-line identification.

By employing multi-channel order recursive algorithms, selective addition of details is recursively performed in an efficient computational manner that will also decouple the identification of different parameters of an ARX model of the plant, thus enhancing isolation and estimation of changes.

Basis expansions will be performed on the smallest possible time block as dictated by data quality, giving an estimate of the localized behavior of the parameters. In the event of noisy measurements, it may be necessary to improve upon the estimate as more data becomes available. Such global description is attained by simply and decisively moving into the so called lower resolutions in a “zoom out” manner upon the system trajectory thus enhancing the parameters estimates at longer time intervals. Hence, the whole process can be described as *event triggered* estimation; that is, one which moves into fitting different basis functions according to whether changes in the trajectory occur as the time axis is swept.

With the belief of possessing a flexible tool for isolating and estimating plant variations, a controller can be appropriately designed based on a certainty equivalence, that is according to the most recent estimate of the plant model, so as to meet specific performance criteria. The frozen-time controller design can be casted as identifying a system with abrupt parameter changes, and hence can be obtained in a similar fashion as done with the original open loop plant.

## 1.2 Outline of the Thesis

In the next chapter, the identification-tracking problem will be presented exactly, together with necessary background on common practices in time varying system identification and adaptive control design . Chapter 3 will briefly present wavelet theory and the associated discrete perfect reconstruction filter banks. It will also present the identification methodology set earlier in [1]. The identification and control algorithm will be developed in chapter 4. There, wavelet selection to fit system trajectory on short intervals is first presented; later, block recursions will be utilized on longer data records to capture global trajectory and improve estimates. The controller is developed next using time varying setting of its tuning parameter and is shown as a replica of the identification methods presented earlier in the chapter. Simulations of the algorithm are presented in chapter 5. Also in that chapter, common tradeoffs between accuracy and fast detection on one hand and excitation, noise, and rate of jump occurrence on the other hand are discussed. Chapter 6 gives conclusions and future directions of this research.



## Chapter 2

# Background and Problem Definition

### 2.1 Problem Setup

Consider a plant  $P$  operating in a feedback control configuration which aims at minimizing the tracking error of the output with respect to a reference command  $r$ . The plant, however, may undergo a series of unexpected transient jumps or abrupt faults that reflect in similar step changes in some of the system model parameters. One may think of these jumps as possibly due to failures in plant components (actuators, short circuit current surges in the stator of a rotating machine, failure of loose bearings etc..) or a sudden misconnection in sensor mechanism at the plant output. We would like to detect these changes, isolate the relevant parameters and estimate the subsequent model while keeping the performance criterion minimized by simultaneously tuning the utilized control mechanism.

#### 2.1.1 Plant Specification

Under reasonable a priori knowledge of the system, the plant, along with its operating peripherals, will be closely approximated as an output prediction error model. Specifically, a time varying  $ARX$  model of prespecified order of both the auto-regressive and the exogenous parts is assumed to well describe the single input-single output process,

$$y(n) = \sum_{k=1}^p \alpha_k(n)y(n-k) + \sum_{k=1}^q \beta_k(n)u(n-k) + w(n) \quad (2.1)$$

where  $u(n)$  and  $y(n)$  are the input and output of the process respectively and  $w(n)$  is a zero mean gaussian white noise that enters additively into the measurements of the output. We will denote this process by  $M_{\theta_o}$ , where  $\theta_o = [\alpha_k(\cdot), \beta_j(\cdot)]$  is the vector of model parameters.



Under various abrupt faults, the plant will be described by a new ARX model  $M_{\theta_1}$ . A bound on the magnitude of such abrupt changes is assumed to be inferred from the physical operating point or loading conditions of the system. Hence, variations with time that are below some threshold and that are attributable to slow deterioration of the system components will not be considered here ( these occur at longer time scales and can be safely corrected or controlled). The above assumptions will be addressed in chapter 4.

### 2.1.2 Controller Specification

A time varying controller is incorporated into the closed loop and will be designed based on frozen time methods. That is, each time a controller is designed, the plant is thought of as linear time invariant plant with its defining operators fixed at the values they had at that particular time. Hence, we will assume a linear time invariant controller which is parametrized in terms of a single affine FIR parameter  $Q$ , as implied by an all-stabilizing controller description of an LTI system presented below.

Let  $K_o$  be a real rational stabilizing controller of an LTI real rational plant  $P$  as dictated by classical stability methods, then both  $P$  and  $K_o$  can be written as the ratio of coprime functions. Coprimeness here is meant with respect to the space of stable systems, that is the stable rational SISO functions have no common zeros in the closed unit disk (no common non-minimum phase zeros).

$$P(\lambda) = \frac{N(\lambda)}{M(\lambda)}, \quad K_o(\lambda) = \frac{Y(\lambda)}{X(\lambda)} \quad (2.2)$$

where  $\lambda$  is the backward shift conventionally written also as  $P(\lambda) = P(z^{-1})$  where  $P(z)$  is the system function in the  $Z$  domain. It is well known that if  $K_o$  is a stabilizing controller, written as above, then the closed loop stability condition is represented as a Bezout identity

$$M(\lambda)X(\lambda) - N(\lambda)Y(\lambda) = I \quad (2.3)$$

All stabilizing controllers of  $P$  are given by

$$K(\lambda) = \frac{Y(\lambda) - M(\lambda)Q(\lambda)}{X(\lambda) - N(\lambda)Q(\lambda)} \quad (2.4)$$

where  $Q(\lambda)$  is any stable function. In subsequent literature, we will omit the index  $\lambda$  with the belief that it is understood within the context. Under the above setting, the error to

reference transfer function is given by

$$e = M(X - NQ)r \quad (2.5)$$

By requiring that the amplitude of  $e$  be minimized, we have

$$\min_{Q_{stable,LTV}} \|e\|_2 = \min_{Q_{stable,LTV}} \|M(X - NQ)r\|_2 \quad (2.6)$$

We can think of the above equation as  $MXr = MNQr + error$  and defining  $y_c \stackrel{def}{=} MXr$  and  $u_c \stackrel{def}{=} MNr$ , we can write

$$y_c = Qu_c + e_c \quad (2.7)$$

By assuming that the input  $u_c$  to output  $y_c$  relation in the “system”  $Q$  is time varying and that  $Q$  can be well represented by a finite order model of TV coefficients  $q_i$ , we can set up the problem as

$$y_c(n) = \sum_{k=0}^{p_c-1} q_k(n)u_c(n-k) + e_c(n) \quad (2.8)$$

where  $p_c$  is the order of the assumed AR model and  $n$  is a sample data point in time. Hence, the controller performance criterion is transformed into a time varying regressor selection problem similar to the one set forth in modeling the time varying parameters of the system  $P$ . Since  $Q$  is a stable function, it is important to make sure that the TV parameters obtained by solving the regression problem correspond to a *bounded control action*. Unconstrained minimization of the regression solution might result in excessive control requirements in a case similar to a time invariant control of non-minimum phase plant which, trying to invert the zero dynamics (as implied by the minimization problem), might end up with unstable pole-zero cancellations. To avoid this, the minimization will also include the constraints of keeping the magnitude of the TV parameters as small as possible. The latter is needed since stability of a LTV matrix ( $Q$ ) requires that the sum of its elements in a row be bounded. Therefore, the minimization problem becomes

$$\min_{Q_{stable,LTV}} \left\| \begin{array}{c} e \\ 0 \end{array} \right\|_2 = \min_{Q_{stable,LTV}} \left\| \begin{array}{c} M(X - NQ)r \\ Q \end{array} \right\|_2 \quad (2.9)$$

Finally, a block diagram of the closed loop identification/control design is shown in figure 2-1.

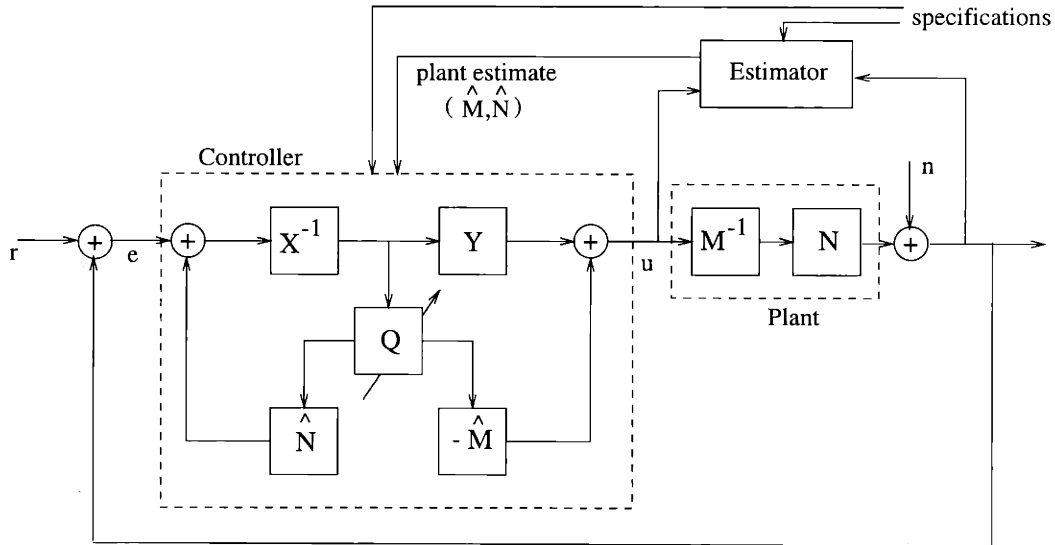


Figure 2-1: Block diagram of the closed loop adaptive control design.

## 2.2 Time Varying System Identification

Non-stationary processes with rational spectra can be represented as time-dependent parametric models. Estimation of these model might involve non-linear as well as linear approaches. For our purposes, we will only discuss ARX models, in which the parameter appear linearly in the estimation problem.

There exists in the literature several adaptation techniques which will track the variations of the system parameters. The performance of these methods, however, depends on the rate of such variations, and also on the level of a priori available knowledge of the system. We will next outline some of the common methods.

### 2.2.1 Recursive Forgetting Factor Methods

Several estimation techniques discount old information about a system in favor of new information in order to track the time variations. These methods, mainly extensions of the recursive time invariant versions are based on adding a forgetting factor into the information matrix. For the model given in equation 2.1, rewritten as

$$y(n) = \Phi(n)\theta + w(n), \quad (2.10)$$

we have:

- a) Exponential weighting with constant forgetting factor [12]: this is based on a weighted quadratic error criterion

$$\theta = \arg \min_{\theta} V = \arg \min_{\theta} \sum_{m=1}^n s(n, m) [y(m) - \Phi(m)\hat{\theta}(m)]^2 \quad (2.11)$$

where the squares of the errors are weighted by

$$s(n, m) = \lambda^{n-m}, \quad 0 < \lambda < 1$$

In a weighted recursive least square setting, for example, the parameter vector is updated recursively as:

$$\begin{aligned} \hat{\theta}(n+1) &= \hat{\theta}(n) + \gamma(n)[y(n+1) - \Phi(n+1)\hat{\theta}(n)] \\ \gamma(n) &= \frac{1}{\lambda + \Phi^T(n+1)\mathbf{P}(n)\Phi(n+1)} \mathbf{P}(n)\Phi(n+1) \\ \mathbf{P}(n+1) &= \frac{1}{\lambda} \left[ \mathbf{P}(n) - \mathbf{P}(n)\Phi(n+1)\Phi^T(n+1)\mathbf{P}(n) \right] \end{aligned} \quad (2.12)$$

where

$$\mathbf{P}^{-1}(n) = \mathbf{R}(n) = \sum_{m=1}^n s(n, m) \Phi(m)^T \Phi(m)$$

The forgetting factor  $\lambda$  is selected small if the parameter changes fast; in this case then only small noise levels are allowed. On the other hand,  $\lambda$  is selected large if the parameters are known to vary slowly; higher noise levels are tolerable then.

- b) Exponential weighting with variable forgetting factor: to adjust  $\lambda$  to the required situation, it can be controlled, for example, by an a posteriori error [12]. In the case when the error is small or there is a lack of excitation, we keep more information, and hence  $\lambda = 1$ . As the error become larger,  $\lambda$  decreases to allow for faster tracking of the variation.
- c) Other methods include covariance matrix resetting (or revitalizing of the algorithm) at particular prespecified instants of time where the adaptation rate goes low (cf. [14]).

## 2.2.2 Basis Expansion Methods

Another approach in estimating time varying parameters  $\theta_i$  is to expand the system parameters onto some basis functions.

$$\theta_i(k) = \sum_{j=1}^m \xi_{j,i} f_j(k) \quad (2.13)$$

This obviously has the implicit assumption that a close approximation of  $\theta_i$  can be reached with a relatively low number of basis functions which requires an a priori minimal knowledge of the type of non-stationarity for a particular system. Reported in earlier literature are expansions on second order polynomials (Rao), other higher orders, and Legendre polynomials (orthogonal basis). Still other methods use Fourier basis (with the assumption of periodic parameter behavior). Gernier [3] utilizes discrete prolate spheroidal basis which give good approximation of band limited functions.

The advantage of basis expansion methods is in the transformation of a linear time varying estimation problem into a linear time invariant one. The model structure is now specified by the original order ( $p$ ) and the number of the basis functions used ( $m$ ). The number of unknowns is increased by a factor of  $m$ , but this is not a restrictive price for obtaining a linear time invariant problem, especially if we can manage to keep  $m$  small, without sacrificing accuracy of the estimation.

## 2.3 Adaptive Control

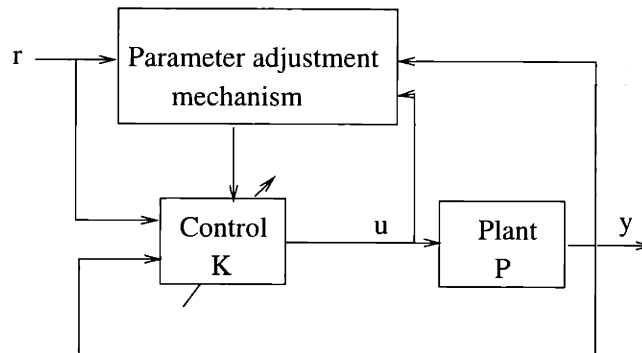


Figure 2-2: General model of adaptive control mechanisms.

An adaptive controller is a *controller with adjustable parameters and a mechanism for adjusting the parameters* [13]. An adaptive control system can be thought of as consisting of

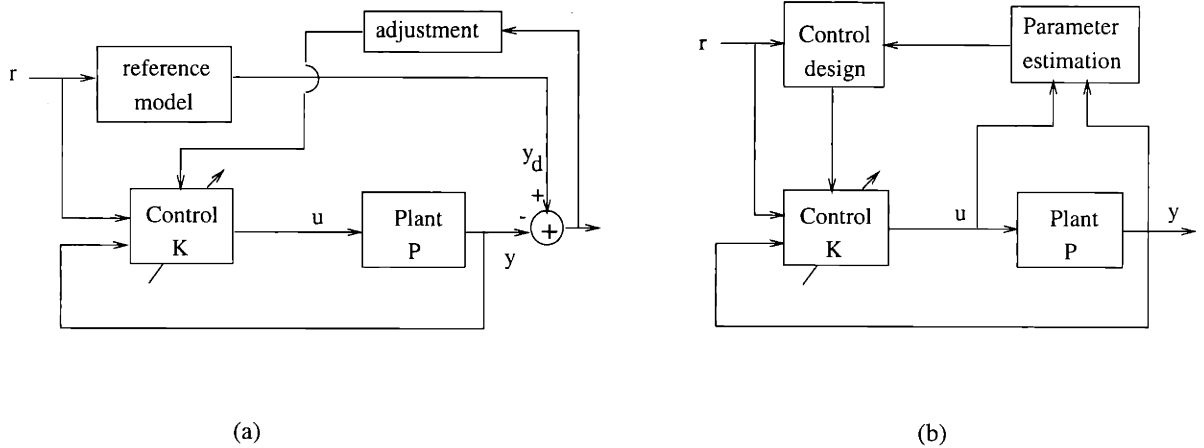


Figure 2-3: Adaptive control mechanisms: (a) model reference adaptive system MRAS and, (b) self tuning regulator STR

two loops. One loop is the normal feedback with the plant to stabilize and achieve performance; the other loop is an adjustment mechanism. A general diagram is shown in figure 2-2, where parameter tuning occurs at a slower rate as compared to the normal feedback. Adaptive control systems employ either *direct methods*, where the adjustment of the controller parameters is done directly to improve the performance index, or *indirect methods* where estimates of the process parameters are obtained first and then used to design the control law. The performance criterion is measured in terms of the error between the plant output  $y$  and a desired output  $y_d$ . If  $y_d$  is a constant, the problem is that of regulation, whereas if  $y_d$  varies with time, it is a problem of tracking. When the characteristics of the plant are unknown, both regulation and tracking are viewed as adaptive control problems [15]. Examples of systems that deal with such performance problems are gain scheduling, model-reference adaptive control and self-tuning regulators [13].

### 2.3.1 Model Reference Controllers

In a model reference adaptive system (MRAS), the output of the controlled process is required to track a desired output generated by an ideal reference model of the system (figure 2-3(a)). Hence, given an LTI plant  $P$  with input-output pair  $\{u, y\}$  and a reference LTI model  $M$  specified by its input-output pair  $\{r, y_d\}$ , this scheme adjusts the feedback law so as to minimize the error between the desired output  $y_d$  and the process output  $y$ . Considerable a priori knowledge about the plant and the achievable performance (choice of reference model  $M$ ) is normally required to design for the feedback law. The mechanism for

designing the control parameters can be obtained in two ways : by using a gradient method or by applying stability theory.

### 2.3.2 Self Tuning Controllers

Self tuning regulators (STR) employ the *certainty equivalence principle* where the parameters of the process are estimated and then used as if they were equal to the true parameters, that is, the controller automatically tunes its parameters so as to meet the desired closed loop properties. A block diagram of STRs is shown in figure 2-3 (b). Different combinations of recursive estimation methods and controller designs can be used, the performance of which generally depends on the quality of the estimation scheme (errors) and the robustness of the control design against the associated uncertainties in the plant model. In stochastic settings, which originated as minimum variance control, the performance criterion is mainly expressed in terms of the steady state variance of the output and possibly the control signals the performance of which is affected by the level of disturbance and noise signals.

$$J = E\{y^2 + \rho u^2\} \quad (2.14)$$

Minimization of the above cost function is known as *Linear Quadratic Control* (LQG). For  $\rho = 0$ , the problem is that of *Minimum Variance Control*.

## Chapter 3

# Wavelets in System Identification

### 3.1 What are Wavelets

Although numerous references can be found on this topic, a nutshell introduction to wavelets will be given next.

There exists many types of wavelet transforms. Among those, we distinguish between continuous wavelet transform, and discrete wavelet transform [7]. The discrete wavelet transforms include the orthonormal bases of wavelets with which is associated the idea of multiresolution analysis. The discrete wavelet transform expands all admissible functions onto a basis library of continuous-time functions called wavelets. The special feature of the wavelet basis is that all the basis functions are constructed from compressions (or dilations) and shifts (or translations) of a single function, called the mother wavelet  $w(t)$ .

$$f(t) = \sum_{j,m} \xi_{j,m} w_{j,m}(t)$$

where

$$w_{j,m} = w(2^{-j}t - m)$$

For instance, decreasing the parameter  $j$  will give higher compressions (called scales) of  $w(t)$  which correspond in frequency to higher shifts in the frequency content of the original mother function. The parameter  $m$  leads to translates of the function at a particular scale into different time origins.

Hence we have a double indexed skeleton in which shifts in time and scales describe the exact location of a basis function in the time-frequency plane (figure 3-1). These families of scales lead to a multiresolution analysis of functions. In expanding a function  $f(t)$ , we can express the approximations according to different scale spaces (rather than frequencies). A



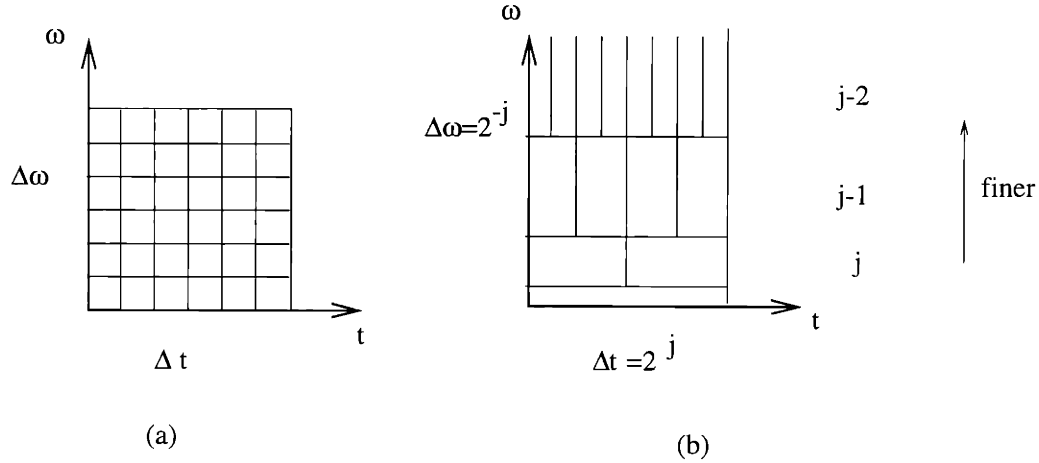


Figure 3-1: Decomposition of the time frequency plane: (a) Fourier (b) wavelets

scale or resolution describes particular frequency contents of  $f(t)$ , as implied by the scale  $j$ , over time, as implied by  $m$ . At a particular resolution level  $j$ , the time steps are  $2^j$ , and the basis are called scaling functions  $\phi(2^{-j}t - m)$  which form the resolution space  $V_j$ .

$$V_j = \text{linear span of } \{\phi_{j,m}\}, m \in \mathbf{Z}$$

$$\phi_{j,m}(t) = 2^{-j/2} \phi(2^{-j}t - m)$$

The new details (not in  $V_j$ ) at level  $j$  are represented by the wavelets  $\psi(2^{-j}t - m)$ . Again, the wavelet basis at one level  $j$  form the detail space  $W_j$ .

$$W_j = \text{linear span of } \{\psi_{j,m}\}, m \in \mathbf{Z}$$

$$\psi_{j,m}(t) = 2^{-j/2} \psi(2^{-j}t - m)$$

Both the coarse representation of the signal and the details at level  $j$  combine to give a multiresolution of the signal at level  $j - 1$  [9]. Thus, in describing functions, we can talk about a multiresolution decomposition into scaling spaces and wavelet spaces:

$$\left. \begin{array}{l} \text{coarse representation of } f(t) \text{ at level } j \\ + \\ \text{details of } f(t) \text{ at level } j \end{array} \right\} = \text{(finer) representation at level } j - 1$$

Hence  $W_j$  is the compliment of  $V_j$  in  $V_{j-1}$  :

$$V_j \oplus W_j = V_{j-1} \quad \text{and} \quad V_j \cap W_j = \{0\} \quad (3.1)$$

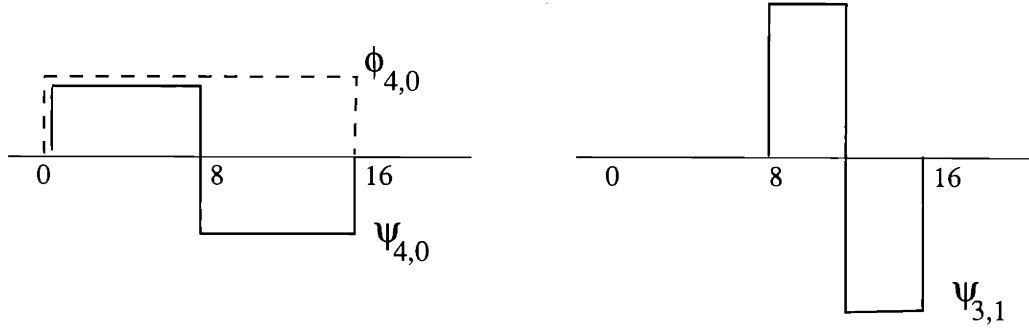


Figure 3-2: Example of the Haar wavelet and scaling function

and

$$V_j \subset V_{j-1} \subset V_{j-2} \dots \quad \text{the "ladder" property} \quad (3.2)$$

Denoting the orthogonal projection operator onto  $V_j$  and  $W_j$  by  $A_j$  and  $D_j$  respectively, then, for  $f(t) \in \mathcal{L}^2$ ,

$$\begin{aligned} D_j f(t) &= \sum_m d_{j,m} \psi_{j,m}(t) \\ \underbrace{A_{j-1} f(t)}_{\text{finer}} &= \underbrace{A_j f(t)}_{\text{coarse}} + \sum_m d_{j,m} \psi_{j,m}(t) \end{aligned} \quad (3.3)$$

and

$$\bigoplus_{j \in \mathbf{Z}} W_j = \mathcal{L}^2(\mathcal{R}) \quad \text{and} \quad V_0 \oplus \sum_{j=-\infty}^0 W_j = \mathcal{L}^2(\mathcal{R}) \quad (3.4)$$

Finally, the simplest type of wavelets, the Haar bases are shown in figure 3-2.

### 3.1.1 Perfect Reconstruction Filter Banks

Corresponding to the continuous time scaling function and wavelet are a discrete low pass filter  $h_0(n)$  and a high pass filter  $h_1(n)$  respectively whose effect will be in computing the expansion coefficients of a function  $f(t)$  onto the aforementioned multiresolution spaces. The pair  $(h_0, h_1)$  is called dyadic perfect reconstruction filter bank (PRFB) as demonstrated by their function. In figure 3-3, a signal vector  $g[n]$  at the input is operated upon in one channel by  $h_0$  ( represented in the  $Z$  domain by  $H_0(z)$ ), which has an averaging effect, and in the other channel, a shifted version of  $g[n]$  is filtered by  $h_1$  which expresses differences

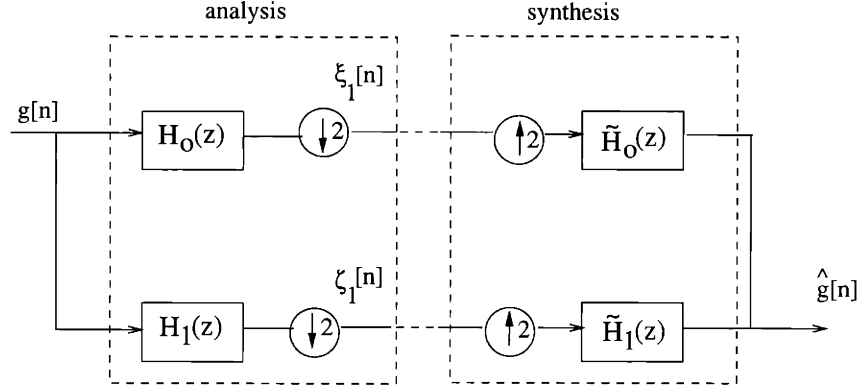


Figure 3-3: Dyadic perfect reconstruction filter bank. Shown are the analysis and synthesis stages

between components of  $g[n]$ ; both outputs are then downsampled by a factor of 2. The net effect is to obtain a smoothed version of  $g[n]$  in channel 1 ( $\xi_1[n]$ ) and the remaining detail at channel 2 ( $\zeta_1[n]$ ). In the Z domain, this is expressed as

$$\begin{aligned} \mathcal{Z}\{\xi_1[n]\} &= H_o(z) \frac{G(z)+G(-z)}{2} \\ \mathcal{Z}\{\zeta_1[n]\} &= H_1(z) \frac{G(z)+G(-z)}{2} \end{aligned} \quad (3.5)$$

$h_1$  is usually designed as the alternating flip of  $h_o$ ; for an N-tap filter  $h_o$ , this is

$$\begin{aligned} H_1(z) &= -z^{-N} H_o(-z^{-1}) && \text{(frequency domain)} \\ h_1(n) &= (-1)^n h_o(N-n) && \text{(time domain)} \end{aligned} \quad (3.6)$$

The synthesis bank ( $\tilde{h}_o, \tilde{h}_1$ ) can then, after upsampling  $\xi_1[n]$  and  $\zeta_1[n]$  by a factor of 2, give a perfect reconstruction of  $g[n]$ . Two conditions on the dyadic analysis/synthesis banks to give a perfect replica of  $g[n]$  are :

$$\begin{aligned} H_o(z)\tilde{H}_o(z) + H_1(z)\tilde{H}_1(z) &= 2z^{-l} && \text{(no distortion)} \\ H_o(z)\tilde{H}_o(-z) + H_1(z)\tilde{H}_1(-z) &= 0 && \text{(no aliasing)} \end{aligned} \quad (3.7)$$

where  $l$  is an odd delay factor. The synthesis bank, according to the above constraints, becomes:

$$\tilde{H}_o(z) = H_1(-z) \quad \text{and} \quad \tilde{H}_1(z) = -H_o(-z) \quad (3.8)$$

The cascade of analysis filters will actually give a simple method to move from one resolution space in a wavelet decomposition of signals to another . A function  $\sum_m \xi_{j,m} \phi_{j,m}(t)$

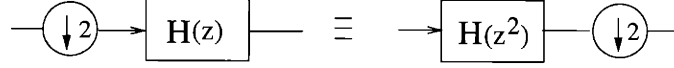


Figure 3-4: The noble identity

in the space  $V_j = V_{j+1} \oplus W_{j+1}$  has coefficients  $\xi_{j+1,k}$  on  $V_{j+1}$  and  $\zeta_{j+1,k}$  on  $W_{j+1}$  as

$$\xi_{j+1,k} = \sum_m h_0(m-2k)\xi_{j,m} \quad \text{and} \quad \zeta_{j+1,k} = \sum_m h_1(m-2k)\xi_{j,m} \quad (3.9)$$

Conversely, moving from low to higher resolutions is equivalent to using the synthesis bank, the expansion coefficients become

$$\xi_{j,m} = \sum_k \tilde{h}_0(2k-m)\xi_{j+1,k} + \tilde{h}_1(2k-m)\zeta_{j+1,k} \quad (3.10)$$

Moreover, the cascaded application of PRFBs has a simple closed form upon moving across scales [1]. Consider for example moving from scale  $j_o$  to  $j_o + 2$  (figure 3-5). Using the noble identity (figure 3-4), the cascade is equivalent to a single filter  $H_i(z)H_i(z^2)$ ,  $i = 0, 1$ , followed by downsampling by a factor of  $2^2 = 4$ . Similar equivalent filter is derived for a reconstruction synthesis cascade (figure 3-6). In general, for expansions of  $j = 0$  up to a

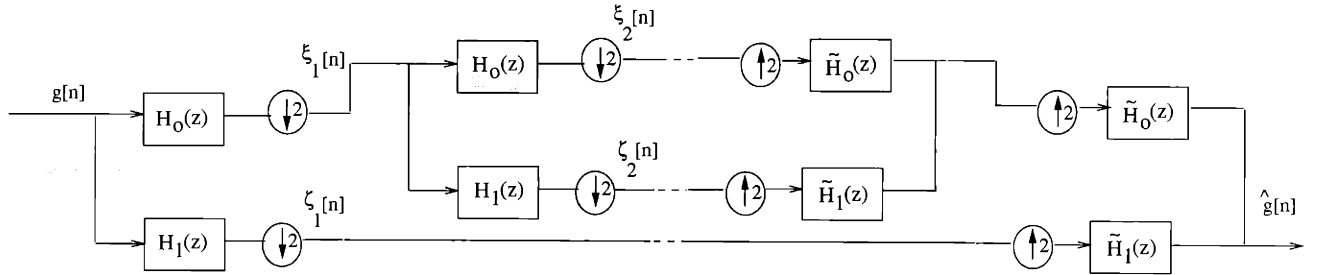


Figure 3-5: Cascade structure corresponding to a multiresolution of depth  $j = 2$

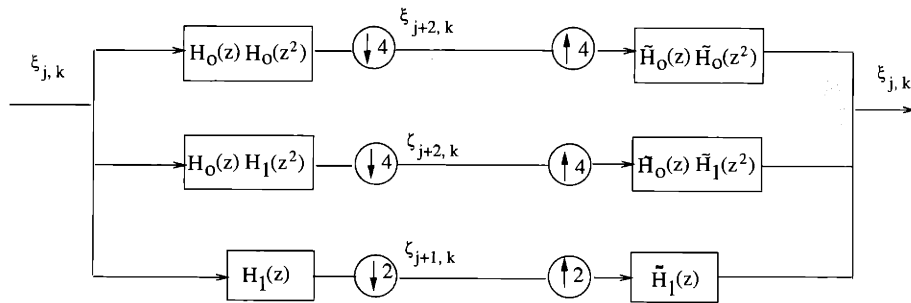


Figure 3-6: Equivalent filters for the cascade in figure 3-5

depth of  $J_{max}$  the low resolution branch has an equivalent filter of

$$H_o^{(J_{max})}(z) = \prod_{i=1}^{J_{max}} H_o(z^{2^{i-1}}) \quad (3.11)$$

and a downsampling by  $2^{J_{max}}$ . At a depth of  $j$ ,  $j = 1 \dots J_{max}$  the detail branch has an equivalent high pass filter of  $H_1^{(j)}(z)$  and a down sampling of  $2^j$  where

$$H_1^{(j)}(z) = H_1(z^{2^{j-1}}) \prod_{i=2}^j H_o(z^{2^{j-i}}) \quad (3.12)$$

### 3.2 Previous Work by Tsatsanis et al

In this section we will present the framework that Tsatsanis [1] set for identifying time varying system parameters using wavelets as expansion basis.

For a discrete TV-AR process

$$y(n) = \sum_{k=1}^p \alpha_k(n)y(n-k) + w(n) \quad (3.13)$$

the coefficients  $\alpha_k(n)$  are modeled as a linear combinations of wavelet basis of resolutions  $J_{max}$  through 1. The resolution level  $j = 0$  is assigned as the level into which  $\alpha_k(n)$  belongs. Therefore, for  $j = 1$ , we have the synthesis equation 3.10

$$\alpha_k(n) = \sum_m \xi_{1,m}^{\alpha_k} \tilde{h}_o(n-2m) + \sum_m \zeta_{1,m}^{\alpha_k} \tilde{h}_1(n-2m) \quad (3.14)$$

Using the equivalent cascade structure presented in the previous section (eqs 3.11, 3.12), and for an analysis depth of  $J_{max}$ ,  $\alpha_k(n)$  is reconstructed as

$$\alpha_k(n) = \sum_m \xi_{J_{max},m}^{\alpha_k} \tilde{h}_o^{(J_{max})}(n-2^{J_{max}}m) + \sum_{j=1}^{J_{max}} \sum_m \zeta_{j,m}^{\alpha_k} \tilde{h}_1^{(j)}(n-2^j m) \quad (3.15)$$

where  $\tilde{h}_o^{(J_{max})}$  and  $\tilde{h}_1^{(j)}$  are the inverse Z transforms of the  $H_o^{(J_{max})}$  and  $H_1^{(j)}$  equivalent filters respectively.

Substituting into 3.13, we obtain

$$y(n) = \sum_{k=1}^p \left\{ \sum_m \tilde{h}_o^{(J_{max})}(n-2^{J_{max}}m) \cdot u(n-k) \cdot \xi_{J_{max},m}^{\alpha_k} + \sum_{j=1}^{J_{max}} \sum_m \tilde{h}_1^{(j)}(n-2^j m) u(n-k) \zeta_{j,m}^{\alpha_k} \right\} + w(n) \quad (3.16)$$

which is linear in the expansion coefficients. The above equation however will in general represent an *underdetermined* system of equations which, by choosing an optimal PRFB,

will contain many insignificant terms as most of the signals energy will be concentrated in low resolution. In other words, select  $(h_0, h_1)$  so that, by compressing the information content into low resolution wavelet basis functions, the reconstruction MSE error due to deleting fine details

$$\lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{-N}^N (\hat{\alpha}_k(n) - \alpha_k(n)) \quad (3.17)$$

is minimized. Of the conditions on how to select the optimal PRFB, it has been proposed that if the spectral content of  $\alpha_k$ , or  $S^{\alpha_k}(\omega)$  satisfies  $S^{\alpha_k}(\omega) \geq S^{\alpha_k}(\pi - \omega)$ ,  $\forall \omega \in [0, \pi/2]$ , then a low-pass /high-pass half band bank is the best dyadic PRFB in the sense of minimizing the reconstruction MSE. This is an assurance that the standard PRFB is actually applicable in most cases as slow behavior of a signal corresponds to low frequencies whereas fast transients are of high frequency nature.

Since a finite set of data is available, end effects are treated using periodic extensions. Matrix form of equation 3.16 can be written as

$$\underline{y} = \mathbf{H}\underline{\Gamma} + \underline{w} \quad (3.18)$$

where

$$\begin{aligned} \underline{y} &= [y(0) \quad y(1) \quad \dots \quad y(N-1)]^T \\ \underline{w} &= [w(0) \quad w(1) \quad \dots \quad w(N-1)]^T \\ \underline{\Gamma} &= [\Gamma^{(\alpha_1)} \quad \Gamma^{(\alpha_2)} \quad \dots \quad \Gamma^{(\alpha_p)}] \end{aligned} \quad (3.19)$$

with  $\Gamma^{(\alpha_k)}$  being the vector of expansion coefficients of parameter  $\alpha_k$ , that is

$$\begin{aligned} \Gamma^{(\alpha_k)} &= [\zeta_{J_{max},0}^{\alpha_k} \dots \zeta_{J_{max},\frac{N}{2J_{max}}-1}^{\alpha_k} \quad \zeta_{J_{max},0}^{\alpha_k} \dots \zeta_{J_{max},\frac{N}{2J_{max}}-1}^{\alpha_k} \dots \\ &\quad \dots \zeta_{J_1,0}^{\alpha_k} \dots \zeta_{1,\frac{N}{2}-1}^{\alpha_k}]^T \end{aligned} \quad (3.20)$$

In the matrix  $\mathbf{H}$ , the  $n$ th row is  $\underline{h}(n) = [\underline{h}_c(n)y(n-1) \quad \dots \quad \underline{h}_c(n)y(n-p)]$  where  $\underline{h}_c(n)$  is the  $n$ th row of the matrix  $\mathbf{H}_c$  which contains all *circularly* shifted versions of the basis sequences of all depths  $J_{max}$  through 1

$$\mathbf{H}_c = [\mathbf{H}_{0c}^{(J_{max})} \quad \mathbf{H}_{1c}^{(J_{max})} \quad \dots \quad \mathbf{H}_{1c}^{(1)}] \quad (3.21)$$

where the columns of  $\mathbf{H}_{0c}^{(J_{max})}$  and  $\mathbf{H}_{1c}^{(j)}$  are circular shifts of  $\tilde{h}_o^{(J_{max})}$  and  $\tilde{h}_1^{(j)}$  with steps of  $2^{J_{max}}$  and  $2^j$  respectively.

Underdeterminacy in the above linear regression problem can be avoided if, based on

the parameters' behavior and the compression achievable by the PRFB, it is possible to ignore details in the parameter representation beyond some resolutions level  $J_{min}$ . That is, expand only on  $J_{max}$  through  $J_{min}$  where  $J_{min} > \log_2 p$ . Insignificant details in the remaining basis library are then actually eliminated using common methods in regression model selection. Tsatsanis et al suggested, starting from a prespecified levels  $J_{min}$  and  $J_{max}$ , undergo an up-to-bottom approach in "peeling off" irrelevant regressors so as to decrease the model order without sacrificing accuracy. In "trimming" a current model  $\mu_o$  with an order of  $d_o$ , to a lower order models  $\mu_i$  of order  $d_i$ , the selection criterion used might be

- an F-test, used first by Astrom, and accepts  $\mu_i$  with confidence levels  $\alpha$ .
- an information criterion, namely Akaike's AIC which weighs a model order relative to the approximation error. This is a more objective test and will be discussed in chapter 4.

The paper also presents several examples to highlight the advantage of multiscale representations in achieving more efficient model descriptions, this will be documented further in the next section.

### 3.3 Why Wavelets

Wavelets have been used extensively in image coding where it provided a concise or compact representation with good accuracy. The ability to capture a global behavior of a signal on low resolutions and to move progressively into details as needed provides a flexible tool for an efficient decomposition. On a time-frequency lattice, the gridding of the time frequency plane under wavelets naturally makes use of the uncertainty principle: the wavelets are time-width adapted to their frequency, high frequency are very narrow and low frequencies are much broader (figure 3-1). That is by using different scales of wavelets and their translates, we are able to span efficiently those relevant grid atoms that correspond to time-localized signal spectrum.

On the same track, the localization properties of wavelets can be useful in parameter identification. For a parametric system model, and under suitable wavelet bases (correspondingly good PRFBs) most of the information about the global behavior of a time varying parameter is retained at low resolutions of the basis expansion. Also, local transitions manifest themselves at higher details into which a length-adapted wavelet basis

can actually 'zoom' into, much like edge detection in images. Hence an obvious gain is avoiding the redundancy in information upon using constant width basis. Since features of a parameter profile are obtained in various detail, the hierarchical property of wavelets helps in controlling the trade-off between accuracy and speed of an approximation, which is especially important in on-line identification.

Moreover, upon processing finite length data blocks, wavelets avoid discontinuities and Gibbs phenomena associated with, for example, Fourier basis. Hence, and as has been demonstrated in Tsatsanis et al [1], we expect multiresolution expansions to offer parsimonious representation of the parameters profile without sacrificing tracking capability.





## Chapter 4

# The Identification and Control Algorithm

This chapter will present in detail an identification algorithm which can be used on-line in a block recursive mode to estimate the parameters of an *ARX* model. Block recursive refers to the notion that the parameter estimates are updated in units of  $N$  sample points. The algorithm will expand time varying parameters onto wavelet basis functions as introduced in chapter 3. Emphasis is given to the jump parameter case although several ideas will be shown as extendable to a general orthogonal wavelet setting. After stating necessary assumptions on the data availability and the behavior and stability of the system, identification on a single interval is discussed where the relevant bases are added starting from a minimal low resolution description and moving to a more detailed one, an approach denoted by a bottom-up regressor order selection. This will help capture local profile of parameter variation. The global profile is approximated by moving into lower resolutions of the expansion library whenever possible, and this is mainly done using hypothesis testing among successive time intervals.

The control problem will then be formulated where we will be identifying the plant parameters and using a certainty equivalent design (frozen time) controller. The controller is parametrized using a single time varying tuning parameter  $Q$  which is identified in much the same manner as the scheme used to estimate the plant  $P$  (chapter 2).

## 4.1 Assumptions

### Available Data

- The data set to be used in identification is generated by a plant  $P$  that can be closely approximated by an ARX model of a prespecified order  $(p,q)$ . Knowledge of the order is necessary as the algorithms are not order-correcting although such extensions to find the correct order can be clearly attained since, just as the current algorithm searches among regressors to find the basis expansion coefficients that correctly describe the process, we can cast the selection so as to also search for a good order approximation of the original time varying model [1].
- The input data set is sufficiently exciting. The availability of a rich data is a common and vital assumption in particular to identification on relatively short intervals by exciting modes that well describe the given system.
- Since the signal to noise ratio or data contamination has a direct effect on the length of an identification block  $N$  considered, additive noise to the data sets is assumed to be of a white nature. This is also a common and practical limitation on the identifiability of a short data set. Later in the sequel, and using hypothesis testing, a link will be made between how well the algorithm performs and the level of noise existing in the system.
- The data arrives sequentially into the estimator. Define a *time unit*  $d_l$  to be a set of  $N$  sample points  $d_l = \{n \in [d_l N + 1, (d_l + 1)N]\}$ . Denote a block of input /output data within a time unit  $d_l$  by  $D_{d_l}$ :  $\mathcal{D}_{d_l} = \{\mathcal{D}(n), n \in [(d_l)N + 1, (d_l + 1)N]\}$ . Thus we will be using data available at time instant  $d_l N$  to identify the system parameters within that time unit  $d_l$ . The estimate, therefore, will only be available after this time unit has elapsed, the speed of which depends on the computational efficiency, which is enhanced in the sequel by utilizing efficient order update methods.

### System Behavior

- Parameter changes are of a fault type; that is, they occur instantaneously as sudden jumps. It follows that the plant  $P$  belongs to or can be closely approximated by the set of ARX models  $\mathcal{M}$  with step parameter changes. The magnitude of a parameter jump

(on a per unit basis) is assumed to exceed some minimum which can be based upon physical knowledge of the system (amount of loading of a machine for instance). This is contrasted with the slow drifting of the system parameters due to deterioration, in which case the change is approximated by a series of small sudden jumps. The latter case will not be addressed here, and is expected to be a simpler problem on which other conventional methods perform well.

- In performing off-line identification, the expansion bases are very well chosen to capture various jump rates in a parameter profile due to the localized nature of the expansion library, and is expected to perform particularly well. However, in carrying on-line identification for control purposes, we assume that there exists an a priori known upper bound on the rate of jump changes per unit time. This is necessary to attain a minimum confidence level in parameter convergence after a jump occurs for proper control design to be used. The tracking time of a change upon expanding onto Haar basis, however, is relatively short compared to other expansion methods, since the Haar functions expansion, being a natural selection to such problems, allows multiple scales, which means that, for a data set and time unit length satisfying the assumptions presented earlier, we expect to attain confidence in the estimate at fast rates.
- The closed loop system response will be bounded in a short interval following a model jump. This is necessary since there will always exist a non zero lag time before a jump is detected and a proper control action is applied. The time elapse of this grey mode of operation is directly affected by natural estimation tradeoffs, mainly noise and data excitation.

## 4.2 Identification on a Single Interval

### 4.2.1 Algorithm Setup

The ARX model used for approximating a plant  $P$  was given by equation 2.1:

$$y(n) = \sum_{k=1}^p \alpha_k(n)y(n-k) + \sum_{k=1}^q \beta_k(n)u(n-k) + w(n) \quad (4.1)$$

We have seen that the time varying parameters  $\alpha_k(n)$  and  $\beta_k(n)$  can be expanded onto wavelet basis functions. If we assume  $\alpha_k(n), \beta_k(n) \in V_{J_{min}-1}$  (chapter 3), then

$$\alpha_k(n) = \sum_m \xi_{J_{max},m}^{\alpha_k} \tilde{h}_o^{(J_{max})}(n - 2^{J_{max}}m) + \sum_{j=J_{min}}^{J_{max}} \sum_m \zeta_{j,m}^{\alpha_k} \tilde{h}_1^{(j)}(n - 2^j m) \quad (4.2)$$

and

$$\beta_k(n) = \sum_m \xi_{J_{max},m}^{\beta_k} \tilde{h}_o^{(J_{max})}(n - 2^{J_{max}}m) + \sum_{j=J_{min}}^{J_{max}} \sum_m \zeta_{j,m}^{\beta_k} \tilde{h}_1^{(j)}(n - 2^j m) \quad (4.3)$$

Rearranging the terms in the model equation

$$y(n) = \sum_{k=1}^q \left\{ \sum_m \tilde{h}_o^{(J_{max})}(n - 2^{J_{max}}m) u(n-k) \xi_{J_{max},m}^{\alpha_k} + \sum_{j=J_{min}}^{J_{max}} \sum_m \tilde{h}_1^{(j)}(n - 2^j m) u(n-k) \zeta_{j,m}^{\alpha_k} \right\} \\ + \sum_{k=1}^p \left\{ \sum_m \tilde{h}_o^{(J_{max})}(n - 2^{J_{max}}m) y(n-k) \xi_{J_{max},m}^{\beta_k} + \sum_{j=J_{min}}^{J_{max}} \sum_m \tilde{h}_1^{(j)}(n - 2^j m) y(n-k) \zeta_{j,m}^{\beta_k} \right\} \quad (4.4)$$

For a data set  $\mathcal{D}_{d_i}$ , we can write the above equation in matrix form:

$$\underline{y} = \mathbf{H}\underline{\Gamma} + \underline{w} \quad (4.5)$$

where

$$\underline{y} = [y(0) \quad y(1) \quad \dots \quad y(N-1)]^T \\ \underline{w} = [w(0) \quad w(1) \quad \dots \quad w(N-1)]^T \\ \underline{\Gamma} = [\Gamma^{(\alpha_1)} \quad \Gamma^{(\alpha_2)} \quad \dots \quad \Gamma^{(\alpha_q)} \\ \Gamma^{(\beta_1)} \quad \Gamma^{(\beta_2)} \quad \dots \quad \Gamma^{(\beta_p)}]^T \quad (4.6)$$

with  $\Gamma^{(\alpha_k)}$  ( $\Gamma^{(\beta_k)}$ ) the expansion coefficients of parameter  $\alpha_k$  ( $\beta_k$ ), that is

$$\Gamma^{(\alpha_k)} = [\zeta_{J_{max},0}^{\alpha_k} \dots \zeta_{J_{max},\frac{N}{2^{J_{max}}}-1}^{\alpha_k} \zeta_{J_{max},0}^{\alpha_k} \dots \zeta_{J_{max},\frac{N}{2^{J_{max}}}-1}^{\alpha_k} \\ \dots \zeta_{J_{min},0}^{\alpha_k} \dots \zeta_{J_{min},\frac{N}{2^{J_{min}}}-1}^{\alpha_k}]^T \quad (4.7)$$

In the matrix  $\mathbf{H}$ , the  $n$ th row is

$$\underline{h}(n) = [\underline{h}_c(n)u(n-1) \quad \dots \quad \underline{h}_c(n)u(n-q) \quad \underline{h}_c(n)y(n-1) \quad \dots \quad \underline{h}_c(n)y(n-p)]$$

where  $\underline{h}_c(n)$  is the  $n$ th row of the matrix  $\mathbf{H}_c$  which contains all *circularly* shifted versions of the basis sequences of all depths  $J_{max}$  through  $J_{min}$

$$\mathbf{H}_c = [\mathbf{H}_{0c}^{(J_{max})} \quad \mathbf{H}_{1c}^{(J_{max})} \quad \dots \quad \mathbf{H}_{1c}^{(J_{min})}] \quad (4.8)$$

where the columns of  $\mathbf{H}_{0c}^{(J_{max})}$  and  $\mathbf{H}_{1c}^{(j)}$  are circular shifts of  $\tilde{h}_o^{(J_{max})}$  and  $\tilde{h}_1^{(j)}$  with steps of  $2^{J_{max}}$  and  $2^j$  respectively.

We can represent the above lengthy model (equations 4.5, 4.6, 4.7) as the following

general linear regression model:

$$y(n) = \sum_{k=1}^{p\theta} \theta_k(n) \Phi(n-k) + w(n) \quad (4.9)$$

where  $\Phi(\cdot)$  is some combination of the input-output data and  $\theta_k(\cdot)$  is the time varying regression vector, which, when expanded onto the basis functions can be casted as

$$y(n) = \sum_{k=1}^{p\theta} \sum_{m=1}^{m_{tot}} a_{m,k} f_m(n-k) \Phi(n-k) + w(n) \quad (4.10)$$

For the problem in equation 4.9, transformed into a regressor selection problem (4.10), only regressors corresponding to relevant resolution levels are retained. This process of zeroing out some of the detail signals so that the representation is parsimonious can be done in two approaches:

- a) The Up-Bottom approach: this is used by Giannakis in his paper [1] where a base model is assumed in which the expansion is performed on all resolution levels  $J_{max}$  through  $J_{min}$ . Alternative models in which regressors corresponding to high resolutions (greater details) are zeroed out are tested against the base model until all regressors are decided upon as relevant. This approach is exhaustive and was noticed upon testing to work well in the case of off-line identification under low noise levels. In such cases where we are given a long data record, the effect of noise can be reduced by utilizing detail levels that are of lower resolution than a noise signal, so as to avoid fitting the noise, and more importantly, by using different noise thresholding methods where the main features or the signal can be extracted usually at a different resolution level than at which noise occurs (peak detection methods and wavelet shrinkage developed by Donoho et al [11]). In an on-line setting, however, we are often limited with a short data record upon which we need to decide a general profile of the signal, hence starting off with high resolutions might result in *fitting the noise* which, should it be alleviated, needs a validation data set. This was confirmed by actual simulations that were performed at earlier stages in this thesis work.
- b) The Bottom-Up approach: Another method to select relevant details that was found significantly more successful than the aforementioned approach is going up the tree from coarse representations and adding details that are decided upon as necessary in

describing the model. This is especially useful when we have some a priori knowledge as to how well low resolutions describe the parameters. For our purposes, a coarse representation, corresponding to a parameter staying constant, describes the process very well over a long interval of time whereas higher resolutions represent the less frequent jumps or variations in parameters which suggests that the bottom-up approach is the natural choice for this case. In adding higher resolutions, the stopping criterion is mainly that of parsimony, or a balance between the order of the model increase and the decrease in the estimation error, as will be discussed in details within the algorithm next (an upper bound on the resolution corresponds to the sampling period at which the coefficient estimate will, in our case, be highly corrupted by noise).

In the following formulation we will be treating the general linear regression presented in equation 4.9 with the understanding that it could be an AR or an ARX model. We will start adding basis functions into each parameter fit and then test whether that addition is actually relevant by using some error-model order compensation criterion (the AIC criterion).

Given an interval length  $N$ , set up a basis function library  $H_c$  which includes  $\tilde{h}_o^{(J_{max})}$  and details  $\tilde{h}_1^{(J_{max})} \dots \tilde{h}_1^{(J_{min})}$ . For notational convenience, the columns of  $H_c$  will be denoted by  $f_i, i = 1 \dots m_f$  as

$$\begin{aligned} \mathbf{H}_c &= [\mathbf{H}_{1c}^{(J_{min})} \quad \dots \quad \mathbf{H}_{1c}^{(J_{max})} \quad \mathbf{H}_{0c}^{(J_{max})}] \\ &= [f_0 \quad f_1 \quad \dots \quad f_m \quad f_{m+1} \quad \dots \quad \dots \quad \dots \quad f_{m_f}] \end{aligned}$$

By observing the structure of the circular shift matrix  $\mathbf{H}_c$ , at resolution level  $J_{max} - j_o + 1$  we have  $\{f_i\}, i = 2^{j_o-1} + 1, \dots, 2^{j_o}$  :

$f_1$		coarse level $J_{max}$
$f_2$		detail level $J_{max}$
$f_3$	$f_4$	detail level $J_{max} - 1$
$f_5$	$f_6$ $f_7$ $f_8$	detail level $J_{max} - 2$
...		

Consider the model presented in equation 4.10:  $y(n) = \sum_{k=1}^{p_\theta} \sum_{m=1}^{m_{tot}} a_{m,k} f_m(n-k) \phi(n-k) + w(n)$ . Denote by

$$\begin{aligned}
x_{ij} &= f_i(n-j)\phi(n-j) \\
\mathbf{x}_{\mathbf{m}_i}(n) &= [f_i(n-1)\phi(n-1) \quad f_i(n-2)\phi(n-2) \quad \dots \quad f_i(n-p_\theta)\phi(n-p_\theta)]^T \\
&= [x_{i1} \quad x_{i2} \quad \dots \quad x_{ip_\theta}]^T
\end{aligned} \tag{4.11}$$

hence

$$\mathbf{x}_{\mathbf{mtot}}(n) = [\mathbf{x}_{\mathbf{m}_1}^T(n) \quad \mathbf{x}_{\mathbf{m}_2}^T(n) \quad \dots \quad \mathbf{x}_{\mathbf{m}_{m_f}}^T(n)]^T \tag{4.12}$$

Also let

$$\mathbf{a}_{\mathbf{m}_i} = [a_{i,1} \quad a_{i,2} \quad \dots \quad a_{i,p_\theta}]^T \tag{4.13}$$

and

$$\mathbf{\Gamma} = [\mathbf{a}_{\mathbf{m}_1}^T \quad \mathbf{a}_{\mathbf{m}_2}^T \quad \dots \quad \mathbf{a}_{\mathbf{m}_{m_f}}^T]^T \tag{4.14}$$

Therefore we can rewrite the system as

$$y(n) = \mathbf{x}_{\mathbf{mtot}}^T(n)\mathbf{\Gamma} + w(n) \tag{4.15}$$

and set up the corresponding linear regression problem for  $N$  data points as

$$\mathbf{y} = \mathbf{H}\mathbf{\Gamma} + \mathbf{w} \tag{4.16}$$

where  $\mathbf{H} = [\mathbf{x}_{\mathbf{mtot}}(1), \mathbf{x}_{\mathbf{mtot}}(2), \dots, \mathbf{x}_{\mathbf{mtot}}(N)]^T$  and  $\mathbf{y}$ ,  $\mathbf{w}$  the corresponding output and noise vectors in  $[1, \dots, N]$ .

Define  $M_{\mu_o}$  to be an index vector of total length equal to  $p_\theta m_f$  and which will have the entry  $p_\theta(j-1) + i$ , call it  $m_{ji}$ , equal to 1 if the function  $f_j$  is included as a basis to the parameter  $\theta_i$ , that is

$$M_{\mu_o} = [m_{11} \quad m_{12} \quad \dots \quad m_{1p_\theta} \quad m_{21} \quad \dots \quad m_{m_f p_\theta}]. \tag{4.17}$$

For example, if for

$$M_{\mu_o} = [1 \quad 0 \quad 0 \quad \dots \quad 1 \quad 1 \quad \dots],$$

entry  $(p_\theta + q)$  is equal to one ( $m_{21} = 1$ ), then  $\theta_q$  has the function  $f_2$  added as a basis function. In terms of the model in equation 4.15, this means that entry  $x_{2q}$  actually exists in the model.

Denote by  $\mu_o$  the current model in which the expansion basis, or the regressors, are



exactly those whose corresponding entries in  $M_{\mu_o}$  are equal to 1, that is

$$\mu_o : \{x_{pq} \mid M_{\mu_o}(p\theta(p-1) + q) = 1\}.$$

## 4.2.2 Searching for Relevant Resolutions

Let us set up the problem where we test how well the current model  $\mu_o$  represents the system as compared to another larger dimension model. Denote by  $\mu_{i,j}$  the proposed model which contains all the functions in the current model  $\mu_o$  and in addition it increments on  $\mu_o$  by containing (a previously not added ) function  $f_i$  as a basis for  $\theta_j$ .

$$\mu_{i,j} : \{x_{pq} \mid M_{\mu_o}(p\theta(p-1) + q) = 1\} \cup \{x_{ij}\}$$

Similarly, define  $\underline{\Gamma}_o$  and  $\underline{\Gamma}_{i,j}$  to be the regressor vectors corresponding to  $\mu_o$  and  $\mu_{i,j}$ , respectively. That is,  $\underline{\Gamma}_{i,j}$  will contain  $\underline{\Gamma}_o$  plus the expansion coefficients  $a_{i,j}$  for which  $x_{ij} \in \mu_{i,j}$ . The dimensions of the vectors  $\underline{\Gamma}_o$  and  $\underline{\Gamma}_{i,j}$  will be named  $dim_o$  and  $dim_{i,j}$  respectively. Using the notation above, we can then write the LR problem (4.5) under a model  $\mu_{i,j}$  as

$$\underline{y} = \mathbf{H}_{\mu_{i,j}} \underline{\Gamma}_{i,j} + \underline{w} \quad (4.18)$$

where  $\mathbf{H}_{\mu_{i,j}}$  contains columns of  $\mathbf{H}$  whose corresponding bases functions are included in model  $\mu_{i,j}$ . Under model  $\mu_{i,j}$ , the average mean square estimation error can be computed as :

$$V_{\mu_{i,j}} = \frac{1}{N} \sum_{k=1}^N (y(n) - \hat{y}(n \mid \hat{\Gamma}_{i,j}))^2 \quad (4.19)$$

where  $\hat{\Gamma}_{i,j}$  is the LS solution of 4.18 and  $\hat{y}(n \mid \hat{\Gamma}_{i,j})$  is the corresponding estimated output of 4.15 under  $\mu_{i,j}$ . This cost function  $V_{\mu_{i,j}}$  will be important in testing the validity of the proposed model  $\mu_{i,j}$  versus the alternative current model  $\mu_o$ . This is equivalent to testing if indeed some part of the signal is in the assumed detail or not. Due to the existence of noise, however, minimizing the cost function  $V_{\mu}$  does not necessarily lead to the correct model, since increasing the dimension ( or adding details) with no constraints will ultimately lead to fitting the noise (which is perfectly representable at the highest resolution level). Therefore, a method which penalizes both the increase in  $V_{\mu}$  and the increase in the dimension of the model  $dim_{\mu}$  (or model complexity) is needed. This is the old problem of parsimony or flexible modeling and has been treated using various methods. Of these, two main methods

to decide between two alternative models have been mentioned in Giannikis [1]: (i) the *F-Test* used earlier by Astrom in deciding between two models  $\mu_1$  and  $\mu_2$  ( $\mu_1 \subset \mu_2$ ) with some level of confidence  $\alpha$ , and (ii) a more objective test, which will be adopted here, and is based on Akaike's information theoretic criterion, known as *AIC*, and is given by

$$AIC(\mu_{i,j}) = -2\frac{1}{N}\log(L_{\mu_{i,j}}) + 2\frac{1}{N}dim_{\mu_{i,j}} \quad (4.20)$$

where  $L_{\mu_{i,j}}$  is the maximized likelihood of the data under model  $\mu_{i,j}$ . Since we are assuming the residuals in (4.18) to be Gaussian i.i.d., the least squares solution coincides with the maximum likelihood solution. In this case,  $AIC(\mu_{i,j})$  reduces to

$$AIC(\mu_{i,j}) = \log(V_{\mu_{i,j}}) + 2\frac{1}{N}dim_{\mu_{i,j}} \quad (4.21)$$

As mentioned earlier, this is an objective criterion: between  $\mu_o$  and  $\mu_{i,j}$ , the one with the lower *AIC* would win. It is, however, noted that *AIC* will perform better when the ratio of the model order  $dim_{\mu}$  to the number of data points  $N$  is small. This calls on for another advantage in using a bottom-to-up approach when we start with a low order model and increment  $\mu_o$  as opposed to the up-to-bottom case where a complete library of basis functions might start with a high dimension cost  $\frac{dim_{\mu_{i,j}}}{N}$  ratio.

A point of concern in applying the *AIC* criterion under short data records, however, is the possible overweighting of the dimension cost. This means that the criterion might be selecting the candidate model which has a lower dimension but a significantly less likelihood of being the correct model. Consider a candidate model  $\mu_s$  where "s" could mean any model order and let its corresponding regressor estimates be  $\hat{\underline{\Gamma}}_{\mu_s}$ . The likelihood of a model is measured in terms a confidence interval around the obtained estimate. This is done as follows:

Denote by  $\underline{\Gamma}_{\mu_s}$  the random variable with mean  $\hat{\underline{\Gamma}}_{\mu_s}$ , and covariance  $\Lambda_{\hat{\underline{\Gamma}}_{\mu_s}}$  computed from the least squares solution as

$$\Lambda_{\hat{\underline{\Gamma}}_{\mu_s}} = \sigma_w^2 (R_N)^{-1} \quad (4.22)$$

where  $\sigma_w^2 = V_{\mu_s}$  (equation 4.19) is the estimated variance of the error (assumed to be Gaussian i.i.d, hence leading to the gaussianity of  $\underline{\Gamma}_{\mu_s}$ ), and the associated regression matrix

$$R_N = \frac{1}{N} [\mathbf{H}_{\mu_s}^T \mathbf{H}_{\mu_s}]$$

Then a confidence level  $\gamma$  in how good the estimate obtained is can be found from

$$Prob(|\underline{\Gamma}_{\mu_s} - \hat{\underline{\Gamma}}_{\mu_s}| < \underline{\alpha}) > \gamma \quad (4.23)$$

where  $\underline{\alpha}$  is an assumed bound on the *percentage* deviation in the components of  $\underline{\Gamma}_{\mu_s}$  (that is in  $\frac{\underline{\Gamma}_{\mu_s}}{\hat{\underline{\Gamma}}_{\mu_s}}$ ). For example, if we consider the coefficient  $a_{i,1}$  to be in  $\mu_s$  (equation 4.13), then (4.23) is actually

$$Prob\left(\left|\frac{a_{i,1} - \hat{a}_{i,1}}{\hat{a}_{i,1}}\right| < 0.3\right) > 0.75 \quad (4.24)$$

gives a 75% confidence that  $a_{i,1}$  (which is gaussian distributed) will be within 30% deviation from its estimated mean  $\hat{a}_{i,1}$ .

The above method of assigning confidence will trade some of the objectivity gained by the *AIC* criterion, but might be necessary if two candidate models have close *AIC* values, or the data record is very short as mentioned earlier, especially when the signal to noise ratio of our measurements decreases.

At this point, we will now have a preliminary description of the identification algorithm:

- a) Start off with  $M_{\mu_o}$  having all zero entries (empty model). Since each parameter  $\theta_i$  should then be expanded onto at least the coarsest resolution, update  $\mu_o$  to be a constant parameter model, that is, set

$$M_{\mu_o}(i) = 1 \text{ if } i = \text{multiple of } (p_\theta).$$

Set up the linear regression problem  $\underline{y} = \mathbf{H}_{\mu_o} \hat{\underline{\Gamma}}_{\mu_o} + \underline{w}$ . Solve for  $\hat{\underline{\Gamma}}_{\mu_o}$  and compute the corresponding  $V_{\mu_o}$  and  $AIC(\mu_o)$ .

- b) Denote by  $W_{stat}$  a matrix having as many rows as there are resolution levels ( $J_{max} - J_{min} + 2$ ) and  $p_\theta$  columns.  $W_{stat}$  is a status matrix which indicates whether a parameter  $\theta$  is sufficiently expanded up to resolution level  $J_o$ . If a detail from resolution level  $J_o$  is added to the parameter  $\theta_i$  then entry  $(J_o, i) = 1$ .

$$W_{stat}(J_o, i) = \begin{cases} 1 & \text{if } M_{\mu_o}(p_\theta(i-1) + m) = 1, \text{ for any } m = 2^{(J_o-1)} + 1, \dots, 2^{J_o} \\ 0 & \text{otherwise} \end{cases} \quad (4.25)$$

For example, for  $p_\theta = 4$ ,

$$W_{stat} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

the 3rd parameter is expanded onto the coarse level only ( $J_{max}$ ), whereas other parameters (1, 2, & 4) have the first level of detail ( $J_{max}$ ) included into their expansion. For any problem, therefore, and corresponding to coarsest resolution, set row 1 to all ones vector.

c) Denote the current detail level by  $J_o$ ; set  $J_o = 2$

d) *while*{row ( $J_o - 1$ ) in  $W_{stat} \neq$  zero vector },

for all functions  $f_i$  on the level  $J_o$  ( $i = 2^{J_o-1} + 1, \dots, 2^{J_o}$ ),

for  $j = 1 \dots p_\theta$ ,

Create the alternative model  $\mu_{i,j}$

Solve for  $\hat{\Gamma}_{\mu_{i,j}}$  using some efficient upgrade technique (to be presented next);

compute the corresponding residual  $V_{\mu_{i,j}}$ .

Formulate the hypothesis

$$H_0 : \mu_{true} = \mu_o$$

$$H_1 : \mu_{true} = \mu_{i,j}$$

and compute the *AIC* index for  $\mu_{i,j}$ .

*if*  $AIC(\mu_{i,j}) < AIC(\mu_o)$

Compute the parameter covariance matrix estimate  $\Lambda_{\hat{\Gamma}_{\mu_{i,j}}}$  (4.22)

$$\text{is } Prob(|\hat{\Gamma}_{\mu_{i,j}} - \hat{\Gamma}_{\mu_{i,j}}| < \underline{\alpha}) > \gamma ?$$

*if yes*,

accept  $H_1$ ;

set  $\mu_o = \mu_{i,j}$

set  $M_{\mu_o}(p_\theta(i-1) + j) = 1$ ; set  $W_{stat}(J_o, j) = 1$ ;

*else*

accept  $H_0$

set  $M_{\mu_o}(p_\theta(i-1) + j) = 0$ ; set  $W_{stat}(J_o, j) = 0$ ;

*end*; { for j }

*end*; {for all functions  $f_i$  }

increment  $J_o = J_o + 1$

*end* {while}

e) Calculate the system parameters  $\underline{\theta}$  using the obtained model  $\mu_o$ .

**Remark** : The algorithm has the flexibility of estimating each parameter  $\theta_i$  separately. In terms of fault isolation, this is quite desirable since only  $\theta_i$ 's that require more details on top of the coarsest resolution are those that are faulty, hence we are able to detect the fault, localize it to a particular parameter, and approximate the amount of fault. More emphasis will be given to this point later in the thesis.

### 4.2.3 Efficient Multichannel Order Update Algorithm

In the identification process presented above, creating increasing order models and computing the resulting parameters using the conventional least squares pseudo-inverse method might become computationally demanding when the order of the system increases. Instead, to update the order of a model, we will follow an algorithm developed by Glentis et al [2] on multichannel filter order update. This algorithm is efficient in that it does not require inversion of the regression matrix, but only multiplications and divisions, as will be presented next.

Consider the system model of equation (4.15). It can be thought of as a multichannel FIR filter with  $m_f$  input channels  $\mathbf{x}_{m_i}, i = 1 \dots m_f$  and each channel has  $p_\theta$  delays  $x_{ij}, j = 1 \dots p_\theta$ . The filter has a single output channel  $\hat{y}(n)$ , as shown in Figure 4-1.

Glentis et al presented a Levinson type algorithm which can recursively update the  $m_f$  input channels order by increasing the number of delays in any channel. This order update algorithm, which can be found in detail in [2], will be discussed briefly next.

Consider the multichannel FIR filter in equation (4.15), repeated here for convenience

$$y(n) = \underline{\Gamma}^T \cdot \mathbf{x}_{\text{mtot}}(n) \quad (4.26)$$

and let the  $m_f$  channels have (not necessarily equal) delays  $d_1, d_2, \dots, d_{m_f}$ . Also, assign a multi-index  $M = [d_1, d_2, \dots, d_{m_i}, \dots, d_{m_f}]$ , then we can perform the update

$$[d_1, d_2, \dots, d_{m_i}, \dots, d_{m_f}] \rightarrow [d_1, d_2, \dots, d_{m_i} + 1, \dots, d_{m_f}] \quad (4.27)$$

for any  $m_i$ , which gives total maneuverability in the index space, and such that we still satisfy an error minimization criterion between the desired signal (actual output of the system) and the output of the filter  $\hat{y}(n)$ ,

$$e = \sum_{n=0}^N [y(n) - \hat{y}(n)]^2$$

The unupdated order system is characterized by a system of equations

$$\mathbf{R}_{\mathbf{mtot}} \cdot \underline{\Gamma} = -\mathbf{z}_{\mathbf{mtot}}$$

where

$$\mathbf{R}_{\mathbf{mtot}} = \sum_n [\mathbf{x}_{\mathbf{mtot}}(n) \cdot \mathbf{x}_{\mathbf{mtot}}(n)^T], \quad \mathbf{z}_{\mathbf{mtot}} = \sum_n [\mathbf{x}_{\mathbf{mtot}}(n) \cdot y(n)]$$

Starting from the LS filter  $\underline{\Gamma}$ , and following the notation presented in [2], denote by  $\underline{\Gamma}_{m_{tot}+1/i}$  an increased order filter by one delay in channel  $i$ , as presented in equation (4.27). The associated increased order vector:

$$\mathbf{x}_{\mathbf{mtot}+1/i}(n) = [\mathbf{x}_{\mathbf{m}_1}^T(n), \dots, \mathbf{x}_{\mathbf{m}_{(i-1)}}^T(n), \mathbf{x}_{\mathbf{m}_{i+1}}^T(n), \mathbf{x}_{\mathbf{m}_{(i+1)}}^T(n), \dots, \mathbf{x}_{\mathbf{m}_{\mathbf{m}_f}}^T(n)]^T \quad (4.28)$$

indicates that  $\mathbf{x}_{\mathbf{mtot}}$  now contains an increased order subvector  $\mathbf{x}_{\mathbf{m}_{i+1}}^T(n)$  of dimension  $(m_i + 1)$  :

$$\mathbf{x}_{\mathbf{m}_{i+1}}^T(n) = [x_{i1} \ x_{i2} \ \dots \ x_{im} \ x_{i(m+1)}]$$

Then we have to solve the corresponding system of equations

$$\mathbf{R}_{\mathbf{mtot}+1/i} \cdot \underline{\Gamma}_{m_{tot}+1/i} = -\mathbf{z}_{\mathbf{mtot}+1/i} \quad (4.29)$$

This is done by exploiting the structure of the correlation matrices, and then updating a forward and a backward predictors, which will relate  $\underline{\Gamma}_{m_{tot}+1/i}$  after being rearranged to the original filter  $\underline{\Gamma}$ . The correlation matrix  $\mathbf{R}_{\mathbf{mtot}}$  can be either of Toeplitz entries as we assume zero initial conditions and sum the regressor data over  $n = 1 \dots N$ , which is called the autocorrelation problem, or it can be of a near-to-Toeplitz form, which is the covariance method [17]; in this case, the summation is carried out for  $n = p_\theta \dots N + p_\theta$  which is a more natural way of handling a stream of data coming in sequentially [10]. the resulting single step update  $\underline{\Gamma} \rightarrow \underline{\Gamma}_{m_{tot}+1/i}$  requires operations of the order of  $\mathcal{O}(m_f D_f)$  MADS (multiplications and additions),  $D_f = \sum_{i=1}^{m_f} d_i$ .

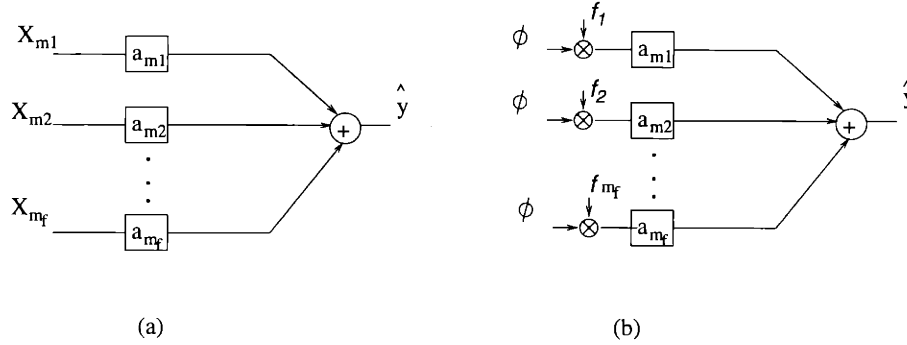


Figure 4-1: Filter interpretation of the basis expansion

Mapping the above algorithm to our problem, we can consider the multi-index  $M$  to be exactly our index vector  $M_{\mu_o}$  (4.17). Hence we can start off with zero delays in the filter  $[0, 0, 0, \dots, 0]$  and apply the single step recursion whenever a new  $f_i$  is to be added. In such a setting, therefore, we have  $m_f p_\theta$  input channels of constant gain filters. Under zero initial conditions on the regressors, we can view the model (4.15) as having  $m_f$  inputs with  $p_\theta$  delays (FIR with a maximum order of  $p_\theta$ ), as is shown in figure 4-1. In terms of the computational effort of the algorithm, both scenarios are obviously equally efficient (as we have a single step recursion). Note that this method, to be called *updatechannel* depends on windowed data. It can also be applied for block recursive identification in consecutive time intervals due to the localized nature of the basis functions which gives a *decoupled* structure in the regression matrix between successive data blocks in effect representing zero initial conditions on these vectors, and hence naturally windowed intervals. This idea will become more apparent in the next section.

Finally, we present the short-interval ARX model identification algorithm, which will be called *VALIDATE*, and which uses the above multichannel order update method (called *updatechannel*).

*Input* data vector  $D_{d_i}$ ,  $J_{max}$ ,  $J_{min}$ ,  $p$ ,  $q$ , and wavelet type confidence interval ( $\gamma$ ), allowable fluctuations ( $\alpha$ ) (4.23)

*Compute*  $\tilde{h}_o^{(j)}$ ,  $\tilde{h}_1^{(j)}$ ,  $j = J_{max}, \dots, J_{min}$ , and  $H_c$

Algorithm *VALIDATE*

*Initialize*  $M_{\mu_o} = [0, 0, \dots, 0]$ ;  
 $W_{stat} = \mathbf{0}_{m_f \times p_\theta}$  (zero matrix), where  $p_\theta = p + q$   
 $\mu_o = \emptyset$  (empty model)

*for*  $i = 1 \dots p_\theta$ ,  
    create model  $\mu_{i,1}$   
     $\hat{\Gamma}_{\mu_o} = \text{updatechannel}(i, f_1)$   
    set  $\mu_o = \mu_{i,1}$   
    compute  $V_{\mu_o}$ ,  $AIC(\mu_o)$   
*end* {*for* }

set  $M_{\mu_o}(1 \dots i) = 1$   
 $W_{stat}(1, 1 \dots p_\theta) = 1$   
 $J_o = 2$

*while*  $W_{stat}(J_o - 1, 1 \dots p_\theta) \neq \mathbf{0}_{1 \times p_\theta}$   
    *for*  $i = 2^{J_o-1} + 1 \dots 2^{J_o}$   
        *for*  $j = 1 \dots p_\theta$ ,  
            Create alternative model  $\mu_{i,j}$   
             $\hat{\Gamma}_{\mu_{i,j}} = \text{updatechannel}(j, f_i)$   
            compute  $V_{\mu_{i,j}}$ ,  $AIC(\mu_{i,j})$   
            Formulate the hypothesis  
                 $H_0 : \mu_{true} = \mu_o$   
                 $H_1 : \mu_{true} = \mu_{i,j}$   
            *if*  $AIC(\mu_{i,j}) < AIC(\mu_o)$   
                Compute  $\Lambda_{\mu_{i,j}}$  (4.22)  
                *is*  $Prob(|\hat{\Gamma}_{\mu_{i,j}} - \hat{\Gamma}_{\mu_o}| < \underline{\alpha}) > \gamma$  ?  
                *if yes*,  
                    accept  $H_1$ ;  
                     $\mu_o = \mu_{i,j}$   
                     $M_{\mu_o}(p_\theta(i-1) + j) = 1$ ;  $W_{stat}(J_o, j) = 1$ ;  
                *else*  
                    accept  $H_0$   
                     $M_{\mu_o}(p_\theta(i-1) + j) = 0$ ;  $W_{stat}(J_o, j) = 0$ ;  
            *end* {*for j*}  
        *end*; {*for i*}  
         $J_o = J_o + 1$   
    *end*; {*while*}  
    compute  $\hat{\theta}$

### 4.3 Block Recursive Identification

We will now use the *VALIDATE* algorithm as a building tool to develop an on-line block recursion for parameter estimation. We will be identifying parameters on the smallest feasible block  $N$  as dictated by data quality, which gives an estimate of the localized behavior of the parameters, and then as more data become available, move onto larger time scales in order to improve the estimates at long time intervals. Due to the localized nature of the expansion basis, however, it may become tedious to decide upon relevant regressors



in the expansion by simply using *VALIDATE* or a similar identification algorithm over the longer intervals, since such approach requires handling growing longer data records. Instead, the identification interval is fixed to  $N$ , and hypothesis testing will be utilized to decide whether and when a larger time scale (or coarser resolution in the wavelet terminology) should be used to improve an estimate of the global parameter behavior; in the wavelet case, this is a straight forward process as the coarse resolution subspaces are included within subspaces of higher resolutions and can be obtained directly by filtering, as presented earlier in chapter 3. The developed algorithm will be “*event triggered*” in the sense that even though identification is done at some resolution  $J_{max}$ , this level will be used in the event that a change in the estimate is detected, otherwise lower resolutions are updated so as to improve an estimate once it does not change.

An outline of the algorithm follows. The detailed version will be presented after necessary tools have been developed. Let the input data vectors  $D_{d_l}$  be incoming sequentially  $d_l = 1, 2, \dots$ . Recall that  $d_l$  defines a time unit of  $N$  samples  $[d_l N + 1, (d_l + 1)N]$ , then as  $D_{d_l}$  becomes available,

a) Apply *VALIDATE* on  $D_{d_l}$ , define the following:

- The parameter vector at time unit  $d_l$  as  $\underline{\theta}^{d_l}$ , the corresponding regression vector expansion coefficients as  $\underline{\Gamma}_{d_l}$ . The corresponding estimates are  $\hat{\underline{\theta}}^{d_l}$  and  $\hat{\underline{\Gamma}}_{d_l}$
- The initial conditions on the data vector as  $D_{init_{d_l}} = \{D_i, i < d_l\}$
- A basis structure index matrix  $W_{global}$  of  $p_\theta$  columns which indicates the highest resolution used in expanding a parameter  $\theta_m$  at consecutive time units. For the jump parameter case, and at time unit  $d_l$ :

$$W_{global}(d_l, m) = \begin{cases} 1 & \text{if } W_{stat}^{(d_l)}(2, m) = 0 \\ 0 & \text{otherwise} \end{cases} \quad (4.30)$$

where  $W_{stat}^{(i)}$  is the status matrix defined earlier in *VALIDATE* applied over interval  $i$  (equation 4.25). Thus an entry of 1 in (4.30) above means that the parameter  $\theta_m$  is a constant or is represented at the coarsest resolution over intervals  $d_l$  whereas a zero means that a fault has occurred in that interval. Variations of this matrix can be used according to particular structures of the expansion basis that we might be interested in (for example to signal for the existence

of resolution level  $j = 4$  which carries encoded information on a transmission channel..)

- A counter vector  $Const$  of length  $p_\theta$  which indicates the number of intervals over which a parameter  $\theta_m$  stays constant

$$Const(m) = \begin{cases} Const(m) + 1 & \text{if } W_{global}(dl, m) = 1 \\ 1 & \text{otherwise} \end{cases}$$

This counter will be used upon moving into larger scales (or longer data intervals) so as to improve the estimate of a parameter of a constant structure.

- Decide whether the parameter vector is different in interval  $d_l$  from what it was in interval  $d_l - 1$ . The Decision rule is based on a hypothesis test which will be presented in section 4.3.2.
- If  $\theta_{d_l}$  and  $\theta_{d_l-1}$  are the same, improve the estimate. This is discussed in the next subsection.

**Note** : the methodology presented has emphasis on jump changes in system parameters (as specified by  $W_{global}$  above). Update into lower resolutions is an inherent characteristic of this problem because we are searching for  $\theta_m$  across nested subspaces  $\{V_j\}$ . In other problems, for example that  $\theta_m \in W_{J_{\theta_m}}$ , only the hypothesis test is useful in signalling for change in the parameter; improvement of the estimate is not done by moving into lower resolutions (as we do not have the nested structure on  $\{W_j\}$ ), but old data can be useful if we know of periodicity of  $\theta_m$  parameter.

### 4.3.1 Moving into Lower Resolutions

In using an identification time unit of length  $N = 2^{J_{max}}$ ,  $J_{max}$  is the coarsest representation, we are assuming that the parameter  $\theta_m$  lies within the subspace  $V_{J_{max}}$ . However, since  $\theta_m$  is unchanged for at least 2 intervals, then it actually lies in some subspace  $V_{J_{\theta_m}}$  of a coarser resolution  $J_{\theta_m} > J_{max}$ . Hence, moving into coarser representations of  $\theta_m$  helps capture the behavior globally on a longer data interval. This is essentially reconstructing the signal (which actually lies in lower dimensional subspaces ) from higher resolutions ( which will be more corrupted by noise; in fact, white noise is very prominent at the highest resolution  $j = 1$  and will remain of white characteristic over coarser scales ). That is, if

$$\theta_m \in V_{J_{\theta_m}} \longrightarrow \theta_m \in V_j, \quad \forall j > J_{\theta_m}.$$

Therefore, moving into coarser scales  $J_{\theta_m} \geq j > J_o$  will improve the estimate of  $\theta_m$  (which lies in  $J_{\theta_m}$ ) since the effect of noise will decrease as longer data records are processed. In the Haar basis case, this reconstruction is simply averaging of the previous intervals. The mechanism of moving into coarser scales is outlined below.

- a) Since  $W_{stat}(d_l, m) \neq 0$ , and the decision rule signalled “unchanged” for parameter  $\theta_m$ , use the estimates in the last  $Const(m)$  intervals, (there are  $Const(m) - 1$  previous intervals along with the current interval), to improve the estimate.
- b) It has been presented before (chapter 3) that if the expansion coefficients  $\xi_j \in V_j$ , then reconstructing  $\xi_{j+1} \in V_{j+1}$  is done by first downsampling and then convolving with the low pass filter of the analysis filter bank ( $h_o$ ). This is presented in matrix form

$$\begin{bmatrix} \xi_{j+1}(0) \\ \xi_{j+1}(1) \\ \vdots \\ \vdots \end{bmatrix} = \begin{bmatrix} \dots & h_o(1) & h_o(0) & 0 & \dots & \\ \dots & h_o(3) & h_o(2) & h_o(1) & h_o(0) & 0 \\ & \dots & \ddots & & & \\ & & & & \ddots & \end{bmatrix} \begin{bmatrix} \xi_j(0) \\ \xi_j(1) \\ \xi_j(2) \\ \vdots \end{bmatrix} \quad (4.31)$$

or in the frequency domain,

$$\Xi_{j+1}(z) = H_o(z) \cdot \left( \frac{\Xi_j(z) + \Xi_j(-z)}{2} \right) \quad (4.32)$$

also, the time domain convolution equation

$$\xi_{j+1}(n) = \sum_k h_o(k) \xi_j(n - 2k) \quad (4.33)$$

Notice in the formulas above that we are adding weighted double shifted entries  $\xi_j(2n), \xi_j(2n - 1), \dots$  to compute  $\xi_{j+1}$ . In the Haar case,  $h_o = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ , we have

$$\xi_{j+1}(n) = \frac{1}{\sqrt{2}} (\xi_j(2n) + \xi_j(2n - 1)). \quad (4.34)$$

In general, for  $k$  increments up the resolution tree,

$$\xi_{j+k}(n) = \frac{1}{(\sqrt{2})^k} (\xi_j(2n) + \xi_j(2n - 1) + \dots + \xi_j(2n - 2^k + 1)) \quad (4.35)$$

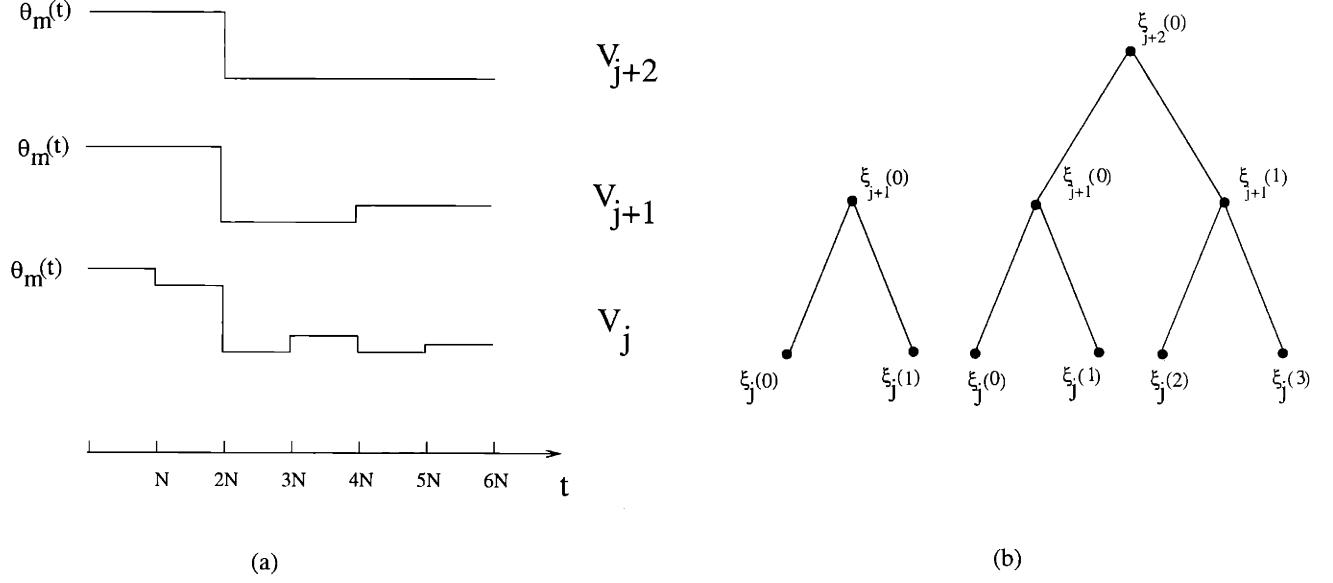


Figure 4-2: Example of moving into lower resolutions (higher indexed subspaces)

To facilitate the general case, let's first consider a typical example. Assume that as data comes in, the current interval  $d_l$  of block length  $N = 2^{J_{max}}$  has  $Const(j) = 2$  and that no change is detected (figure 4-2). Then we need to move from  $V_{J_{max}}$  into  $V_{J_{max}+1}$ . Augmenting the regressor vectors coming from intervals  $d_l$  and  $d_{l-1}$  as  $[\hat{\Gamma}_{d_{l-1}}^T \quad \hat{\Gamma}_{d_l}^T]^T = \xi_{J_{max}+1}$ , then  $\xi_{J_{max}+1}$  is computed according to (4.35), and later the improved parameter  $\theta_m$  can be computed from the corresponding synthesis matrix (at level  $J_{max} + 1$ ). Alternatively, we can compute  $\hat{\theta}_m$  from  $\hat{\theta}_m^{d_l}$  and  $\hat{\theta}_m^{d_{l-1}}$  directly. If we consider the trivial case when  $J_{max} = 1$  (only 2 sample points per interval),  $\hat{\theta}_m$  is actually the running average of  $\hat{\theta}_m^{d_l}$  and  $\hat{\theta}_m^{d_{l-1}}$ :

$$\begin{aligned}
 \begin{bmatrix} \xi_j(0) \\ \xi_j(1) \end{bmatrix} &= \begin{bmatrix} \hat{\Gamma}_{d_{l-1}} \\ \hat{\Gamma}_{d_l} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \hat{\theta}_m^{d_{l-1}}(0) \\ \hat{\theta}_m^{d_{l-1}}(1) \\ \hat{\theta}_m^{d_l}(2) \\ \hat{\theta}_m^{d_l}(3) \end{bmatrix} \\
 &= \begin{bmatrix} \frac{1}{\sqrt{2}}(\hat{\theta}_m^{d_{l-1}}(0) + \hat{\theta}_m^{d_{l-1}}(1)) \\ \frac{1}{\sqrt{2}}(\hat{\theta}_m^{d_l}(2) + \hat{\theta}_m^{d_l}(3)) \end{bmatrix}
 \end{aligned} \tag{4.36}$$

Hence, moving into  $J_{max} + 1 = 2$ , and using (4.34),

$$\xi_{J_{max}+1} = \frac{1}{\sqrt{2}} \left( \frac{1}{\sqrt{2}} (\hat{\theta}_m^{d_l-1}(0) + \hat{\theta}_m^{d_l-1}(1)) + \frac{1}{\sqrt{2}} (\hat{\theta}_m^{d_l}(2) + \hat{\theta}_m^{d_l}(3)) \right) \quad (4.37)$$

Using the reconstruction filter  $\tilde{h}_o^{(2)}$ ,

$$\hat{\theta}_m(i) = \xi_2 \cdot \tilde{h}_o^{(2)} = \frac{1}{4} \left( \hat{\theta}_m^{d_l-1}(0) + \hat{\theta}_m^{d_l-1}(1) + \hat{\theta}_m^{d_l}(2) + \hat{\theta}_m^{d_l}(3) \right) \quad i = 0, \dots, 3 \dots (4.38)$$

If  $\hat{\theta}_m^{d_l-1}(0) = \hat{\theta}_m^{d_l-1}(1)$  and  $\hat{\theta}_m^{d_l}(2) = \hat{\theta}_m^{d_l}(3)$ , which is the case of a constant parameter, then  $\hat{\theta}_m = \frac{1}{2} \left( \hat{\theta}_m^{d_l} + \hat{\theta}_m^{d_l-1} \right)$ .

In general, at a resolution  $J$ , the equivalent filter is simply an averaging filter over  $2^J$  sample points. If  $\hat{\theta}_m^i$  is constant  $i = d_l - k + 1 \dots d_l$ , moving up the tree from  $J_{max}$  to  $J_{max} + k$ ,

$$\hat{\theta}_m = \frac{1}{2^k} \sum_{i=d_l-k+1}^{d_l} \hat{\theta}_m^i \quad (4.39)$$

c) Moving up from  $V_j \rightarrow V_{j+1}$  requires twice as much data; in general, moving from  $V_j \rightarrow V_{j+k}$  requires an increase factor by  $2^k$ . Hence as we obtain more data,

(i) If

$$\log_2(\text{Const}(m)) = k \quad k \text{ integer,}$$

increase the scale by 1 (move from  $V_{k-1}$  into coarser subspace  $V_k$ ). That is  $\hat{\theta}_m$  will be the average over the last  $\text{Const}(m) = 2^k$  intervals.

(ii) If, however,

$$k - 1 \leq \log_2(\text{Const}(j)) < k \quad k \text{ integer,}$$

then the estimate is in  $V_{k-1}$  and is the running average of the last  $2^{k-1}$  intervals.

**Note :**  $V_j \rightarrow V_{j+1}$  resolution update will decrease the effect of noise as is expected and seen in the averaging of the noise effect. The estimate  $\hat{\theta}_m$  is solved in a least square setting where the noise is white and zero mean. Hence, over an interval  $d_l$ ,

$$\hat{\theta}_m^{d_l} = (H^T H)^{-1} H^T (\underline{y} + \underline{e}) = \hat{\theta}_m + (H^T H)^{-1} H^T \underline{e}$$

The resolution update is averaging  $\hat{\theta}_m^{d_l}(i)$  over larger intervals. In the limit, the second term on the right above will go to zero as it is proportional to  $E(e) = 0$ .

### 4.3.2 Hypothesis Test

We will now develop a test that would help assess the possibility whether a given parameter has changed its constituent basis function set, which is expressed in terms of a change in the resolution representation of that parameter.

Let  $\hat{\Gamma}_{\mu_{d_l}}$  and  $\hat{\Gamma}_{\mu_{d_l+1}}$  be two estimates of the regressor vector  $\Gamma_{\mu}$  over two consecutive intervals  $d_l$  and  $d_l + 1$ . Assume that the regressor vector have some component  $a_{i,j}$  which did not change resolution between the two estimates, that is, the corresponding parameter still has the same basis functions set. In the case of jump parameters, this means that both  $a_{i,j}^{(d_l)}$  and  $a_{i,j}^{(d_l+1)}$  are constants and we need to determine whether a jump has actually occurred in the inter-block time. To do this, we set up the following hypothesis:

Since the additive noise is assumed to be of white gaussian nature, the parameters' estimate will, in the linear estimation setup, inherit this gaussian distribution. Hence, denote by  $a_{i,j}^{(d_l)}$  a gaussian random variable  $\mathcal{N}(\hat{a}_{i,j}^{(d_l)}, \sigma_{i,j}^{2(d_l)})$  with mean  $\hat{a}_{i,j}^{(d_l)}$  and variance  $\sigma_{i,j}^{2(d_l)}$ . Similarly for  $a_{i,j}^{(d_l+1)} = \mathcal{N}(\hat{a}_{i,j}^{(d_l+1)}, \sigma_{i,j}^{2(d_l+1)})$ .

Define  $\tilde{a}_{i,j}^{(d_l)} = a_{i,j}^{(d_l)} - a_{i,j}^{(d_l+1)}$  with mean  $\hat{\tilde{a}}_{i,j}^{(d_l)}$  and variance  $\tilde{\sigma}_{i,j}^{2(d_l)}$ , then the hypothesis test will actually be performed on whether this random variable, which represents the difference in the estimates, is different from zero.

Recall that since we are assuming the model does not exhibit slow deteriorations with time but rather jump changes, we do have an a priori knowledge about the minimum expected jump change in the parameters of our model, thus if a jump occurs at all, which we will denote by  $\delta$  given in per-unit (normalized with respect to the mean of the parameter), it will have a *minimum* value of  $\delta_{min} > 0$ . Since this  $\delta$  is also imposed on the basis expansion coefficients, the two hypotheses, corresponding to the minimum expected change, are:

Hypothesis  $H_0$ :  $\tilde{a}_{i,j}^{(d_l)} = 0 + e_1$ , that is no jump has occurred and the offset is due to noise and possibly lack of rich excitation in the two intervals,

Hypothesis  $H_1$ :  $\tilde{a}_{i,j}^{(d_l)} = \delta + e_2$ , that is a jump  $\delta \geq \delta_{min}$  has occurred in the inter-block time.

The corresponding decision rule is :

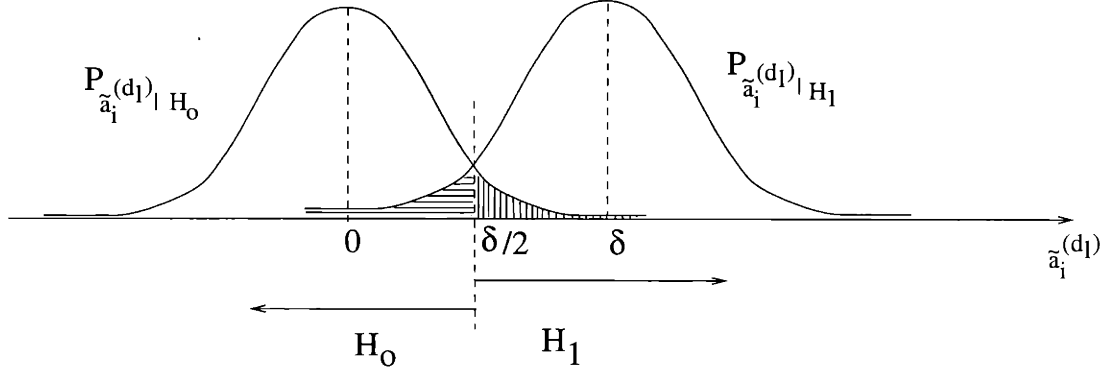


Figure 4-3: Decision regions for jump change between two intervals

if  $\tilde{a}_{i,j}^{(d_l)} \leq \delta/2$  accept  $H_0$ .

if  $\tilde{a}_{i,j}^{(d_l)} > \delta/2$  accept  $H_1$ .

This is directly justified (figure 4-3) by recognizing that

$$P_{\tilde{a}_{i,j}^{(d_l)} | H_0}(\tilde{a}_{i,j}^{(d_l)} | H_0) = \mathcal{N}(0, \sigma_{i,j}^{2(d_l)} + \sigma_{i,j}^{2(d_l+1)}) = \mathcal{N}(0, \tilde{\sigma}_{i,j}^{2(d_l)})$$

$$P_{\tilde{a}_{i,j}^{(d_l)} | H_1}(\tilde{a}_{i,j}^{(d_l)} | H_1) = \mathcal{N}(\delta_{min}, \sigma_{i,j}^{2(d_l)} + \sigma_{i,j}^{2(d_l+1)}) = \mathcal{N}(\delta_{min}, \tilde{\sigma}_{i,j}^{2(d_l)}).$$

The probability of decision error  $P_{error}^{a_{i,j}}$  is the sum of the missed detection  $P_m$  and the false alarm  $P_f$  (no jump) probabilities

$$\begin{aligned} P_m &= P(H_0 | H_1) & P_f &= P(H_1 | H_0) \\ &= P_{\tilde{a}_{i,j}^{(d_l)} | H_1}(\tilde{a}_{i,j}^{(d_l)} < \frac{\delta}{2}) & &= P_{\tilde{a}_{i,j}^{(d_l)} | H_0}(\tilde{a}_{i,j}^{(d_l)} \geq \frac{\delta}{2}) \end{aligned} \quad (4.40)$$

and

$$\begin{aligned} P_{error}^{a_{i,j}} &= \frac{1}{2}P_m + \frac{1}{2}P_f \\ &= \frac{1}{2}\Phi\left(\frac{\frac{\delta}{2} - \delta}{\tilde{\sigma}_{i,j}^{(d_l)}}\right) + \frac{1}{2}\Phi\left(\frac{\frac{\delta}{2} - 0}{\tilde{\sigma}_{i,j}^{(d_l)}}\right) = \Phi\left(\frac{\delta}{2\tilde{\sigma}_{i,j}^{(d_l)}}\right) \end{aligned} \quad (4.41)$$

where  $\Phi(\cdot)$  is the cumulative probability function of a gaussian process. Note that the probability of error is maximum for  $\delta = \delta_{min}$ , and will be denoted by

$$P_{error,max}^{a_{i,j}} = \Phi\left(\frac{\delta_{min}}{2\tilde{\sigma}_{i,j}^{(d_l)}}\right) \quad (4.42)$$

The above test was done for a single component  $a_{i,j}$ . If we now consider the whole vector of regressors  $\underline{\Gamma}_{\mu_{d_l}}$ , then the maximum probability of error over all parameters is

$$P_{error}^{max} = \max_{i,j} \{P_{error,max}^{a_{i,j}}\}, \quad i = 1 \dots m_f, j = 1 \dots p_\theta \quad (4.43)$$

Thus, we can place a confidence level on our estimate. By requiring  $\mathcal{P}_{error}^{max} \leq P_o$  for some prespecified  $P_o$ , it follows that

$$\Phi\left(\frac{\delta_{min}}{2\tilde{\sigma}_{i,j}^{(d_l)}}\right) \leq P_o \quad \forall i, j \quad (4.44)$$

which gives explicit bounds on  $\delta_{min}$  and  $\tilde{\sigma}_{i,j}^{(d_l)}$ . A large deviation in the parameters will decrease erroneous decisions, so does a decrease in the variance of the estimated parameters which is equivalent to the intuitive argument one might give.

### 4.3.3 The Block Recursive Algorithm

It is now possible to present the previous ideas in unison.

set  $d_l = 1$ ,  $D_{init} = \emptyset$ ,  $W_{global} = \emptyset$ ,  $Const = \mathbf{0}_{1 \times p\theta}$ ,

apply *VALIDATE* on  $D_1$ , compute  $\hat{\theta}^1$

while more data is available,

$d_l = d_l + 1$ ; get  $D_{d_l}$

set  $D_{init} = D_i$ ,  $i < d_l$

apply *VALIDATE* to obtain  $\hat{\theta}^{d_l}$

for  $m = 1 \dots p\theta$

if  $W_{stat}(2, m) = 0$

set  $W_{global}(d_l, m) = 1$ ; (indicate that  $\hat{\theta}_m^{d_l}$  is constant)

else  $W_{global}(d_l, m) = 0$ ;

end {for  $m$ }

if any of  $W_{global}(d_l, m) = 1$   $m = 1 \dots p\theta$ ,

while  $W_{global}(d_l - 1, m) = W_{global}(d_l, m) = 1$  for some  $m$  ( $m \leq p\theta$ )

use Hypothesis test 4.3.2 with  $\mathcal{P}_{error}^{max}$ ,  $\delta_{min}$  to check if  $\theta_m^{(d_l)} = \theta_m^{(d_l-1)}$ :

$H_o$  :  $\theta_m$  changed

$H_1$  :  $\theta_m^{(d_l)} = \theta_m^{(d_l-1)}$

choose regressor coefficient  $a_{i,m}$  with desired resolution  $i$  ( $i = 1$  for fault detection)

if  $\tilde{a}_{i,m}^{(d_l)} \leq \delta_{min}/2$  accept  $H_o$

set  $Const(m) = Const(m) + 1$

if  $\tilde{a}_{i,m}^{(d_l)} > \delta_{min}/2$  accept  $H_1$



```

    set  $Const(m) = 1$ 
     $m = m + 1$ 
end ; { while }
for all m such that  $Const(m) > 1$ ,
    if  $\log_2 Const(m) = k$ ,  $k$  integer,
        update resolution  $V_{k-1} \rightarrow V_k$  by using  $\hat{\theta}_m^{(s)}$ ,  $s = d_l - Const(m) + 1 \dots d_l$ 
        and get improved  $\hat{\theta}_m \in V_k$ 
    else ( $k - 1 \leq \log_2 Const(m) < k$ )
        use  $\hat{\theta}_m^{(s)}$ ,  $s = d_l - Const(m) + 1 \dots d_l$  to improve  $\hat{\theta}_m \in V_{k-1}$ 
    end { if }
end { for all }
end { if any }
end; { while more data }

```

Finally we will present an outline (figure 4-4) and flow charts for the identification algorithm (figures 4-5 and 4-6).

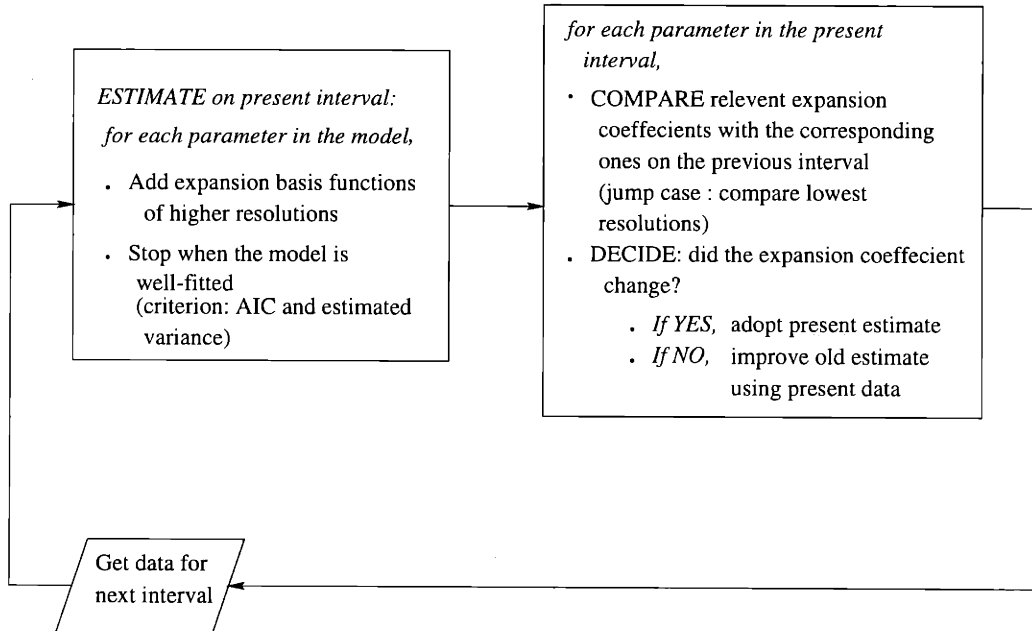


Figure 4-4: General outline of the identification algorithm

## 4.4 Closing the Loop: the Control Algorithm

The controller design will be based on the specifications given earlier in chapter 2. Recall that the error minimization criterion is casted as a regressor selection problem of the expansion basis of the affine parameter  $Q$  defining the set of all-stabilizing controllers of the plant  $P$ .

At time epochs  $d_l N$  (or what we refer to as time units), the controller is designed based on certainty equivalence: for the best parameter estimate  $\hat{P}$  obtained by the identification algorithm, design  $Q$  so as to minimize the tracking error under the model  $\hat{P}$ . In real time, the following operations occur

- a) For the most recent estimate of the plant parameter vector  $\hat{\theta}^{(d_l)}$ , let  $\hat{M}^{(d_l)}$ ,  $\hat{N}^{(d_l)}$  be the estimates of the coprime factors (with no common unstable roots) of the transfer function  $\hat{P}$  at time epoch  $d_l N$ . (Coprime-ness in the plant factors is assumed in this case as the order of the plant is known exactly and no unstable pole-zero cancellations in the real plant occur; moreover, if we assume simple unstable modes of the plant that are properly estimated within the identification time allotted, such cancellations are not allowed to happen in the *closely* estimated plant). Solve for a stabilizing controller  $K_o = \frac{Y}{X}$  of  $\hat{P}$  using the Bezout identity (equation 2.3).

$$\hat{M}^{(d_l)} X - \hat{N}^{(d_l)} Y = I \quad (4.45)$$

If the plant is minimum phase, an obvious solution would be inversion of the plant dynamics. For a stable system, let  $\hat{N}^{(d_l)} = \hat{P}$ ,  $Y = 0$ , and  $X = I$ .

- b) Let  $\hat{r}^{(d_l+1)}$  be the “estimated” command signal in the next time unit  $n = [d_l+1, (d_l+1)N]$ .  $\hat{r}^{(d_l+1)}$  can be reasonably assumed as known (or extrapolated from most recent data). Set up the equivalent regression problem (equation 2.8) which designs the transfer function  $Q$  repeated here for convenience:

$$\hat{y}_c^{(d_l+1)}(n) = \sum_{k=0}^{p_c-1} q_k(n) \hat{u}_c^{(d_l+1)}(n-k) + e_c^{(d_l+1)}(n) \quad n = d_l N + 1 \dots (d_l+1)N \quad (4.46)$$

where  $\hat{y}_c^{(d_l+1)} = \hat{M}^{(d_l)} X \hat{r}^{(d_l+1)}$  and  $\hat{u}_c^{(d_l+1)} = \hat{M}^{(d_l)} \hat{N}^{(d_l)} \hat{r}^{(d_l+1)}$ .

- c) Solve 4.46 for  $q_k(n)$  using Haar wavelet basis expansions in an identical fashion to the method developed in previous section 4.2 (*VALIDATE*). The resulting  $Q$  will in general

be time varying (in the case  $J_{max} > J_{min}$ ). A time invariant version is simply obtained by restricting the projection to be only on the scaling basis ( a constant ).

- d) Apply the designed controller for the next time interval  $[d_l + 1, (d_l + 1)N]$ . The subsequently obtained data  $\mathcal{D}_{d_l+1}$  is used in identification.
- e) Repeat the design procedure at time  $(d_l + 1)N$

Note that for any time unit  $d_l$ , the controller designed is possibly time varying. Although the plant is treated to be time invariant for the considered time unit such that the optimal controller over infinite horizon is actually time invariant, we will be investigating the performance of a time-varying control over the given short time intervals. Moreover, the design scheme allows for other control applications, such as for the case of a continuously jumping-parameter system which changes the rate of jump occurrence between known frequencies.

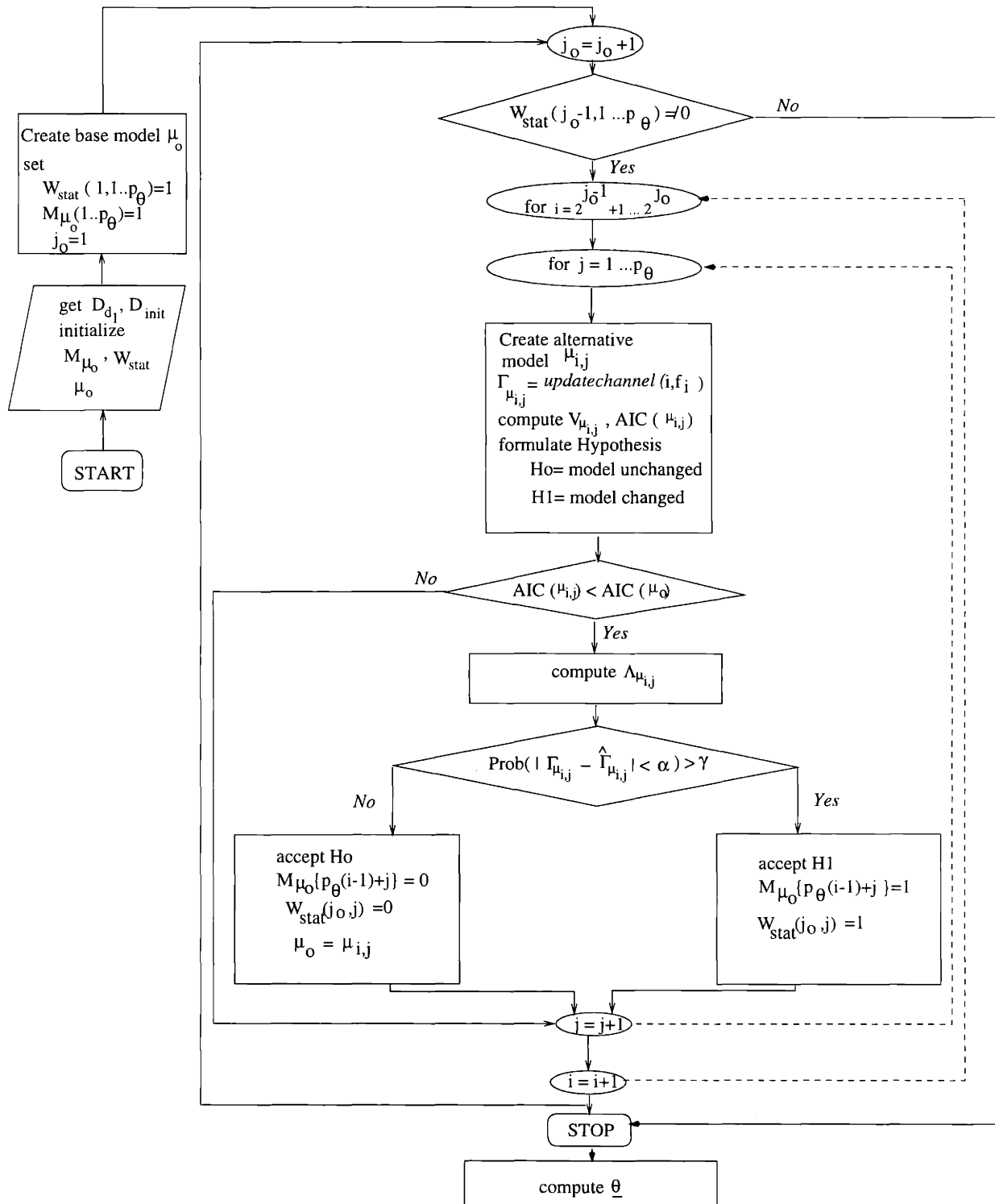


Figure 4-5: Flow chart of the single interval VALIDATE algorithm

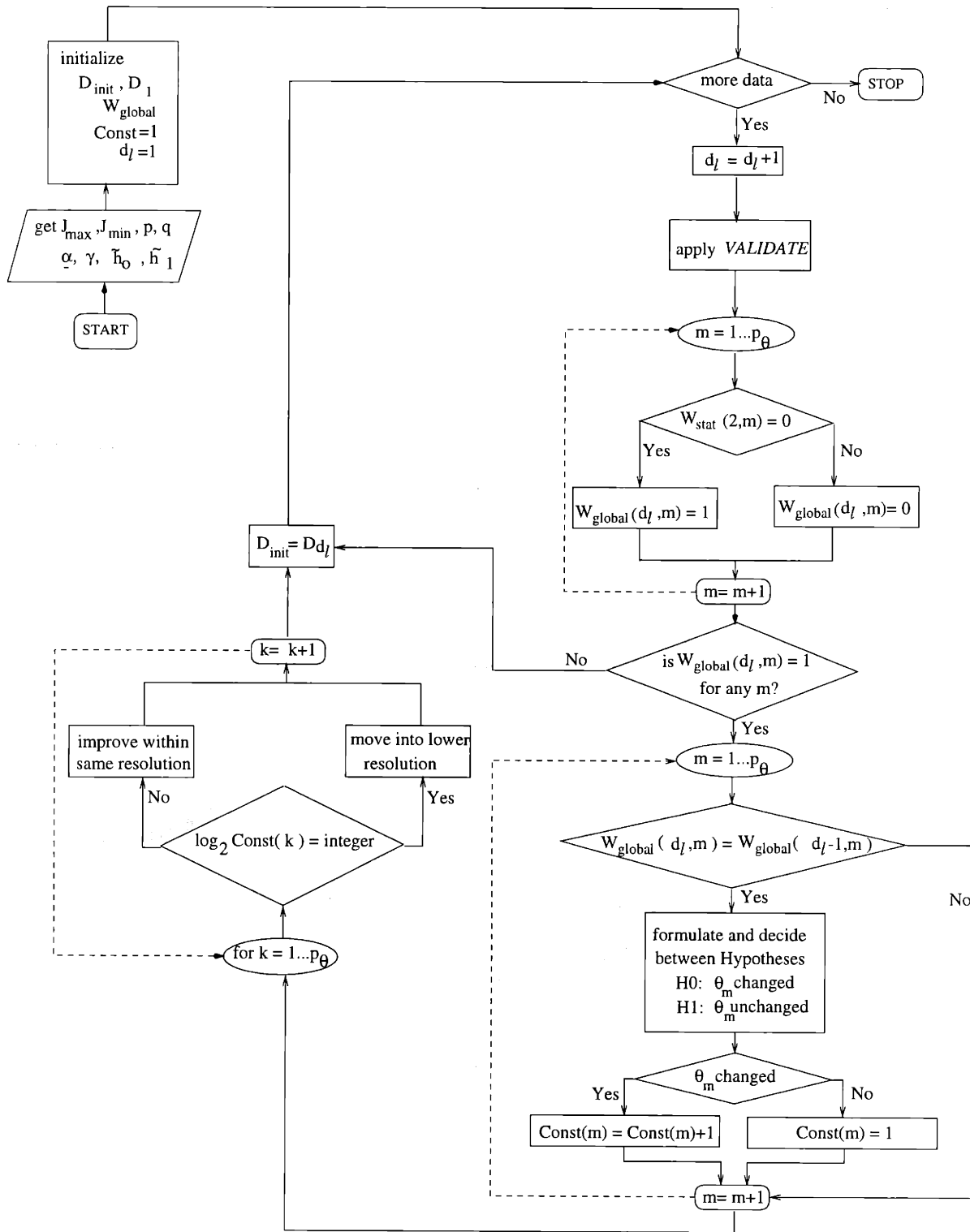


Figure 4-6: Flow chart of block recursive identification algorithm

## Chapter 5

# Simulations and Analysis

This chapter will present some example systems onto which the algorithms of chapter 4 are applied. The fault isolation and estimation properties of the identification algorithm will be studied under various levels of noise contamination. Later as the presented control scheme is applied to achieve tracking, tradeoffs that can arise in on-line detection and control mechanisms will be demonstrated. It is assumed that an a priori minimal knowledge of the signal to noise ratio in estimation data as well as the stability margin of a given plant is available. This is a necessary assumption to allow for obtaining good plant estimates within a reasonable time in which the plant transient behavior is tolerable.

### 5.1 Open Loop Simulations

This section aims at presenting the identification part of the algorithm. Assume a second order *ARX* model of a plant undergoes a series of sudden jumps as follows.

$$\begin{aligned}y(n) &= u(n-1) - 0.5u(n-2) - 1.25y(n-1) - 0.333y(n-2) & n \in [1, 272] \\ &= 2u(n-1) - 0.5u(n-2) - 0.8y(n-1) - 0.083y(n-2) & n \in [273, 832] \\ &= 1.25u(n-1) - 0.5u(n-2) - 1.25y(n-1) - 0.375y(n-2) & n \in [833, 1024]\end{aligned}\tag{5.1}$$

The input is composed of a step and a probing signal which insures that the system modes are sufficiently excited. Sensor noise occurs at the output and is assumed to be white in nature with a signal to noise ratio  $SNR = 18$ . Open loop estimation of the parameters, following procedures presented in chapter 4, is shown in figure 5-1 (denoted by case 1). The method used disjoint intervals of length  $2^5 = 32$  sample points. The basis expansions corresponded to Haar functions scaled from  $J_{min} = 2$  through  $J_{max} = 5$ .

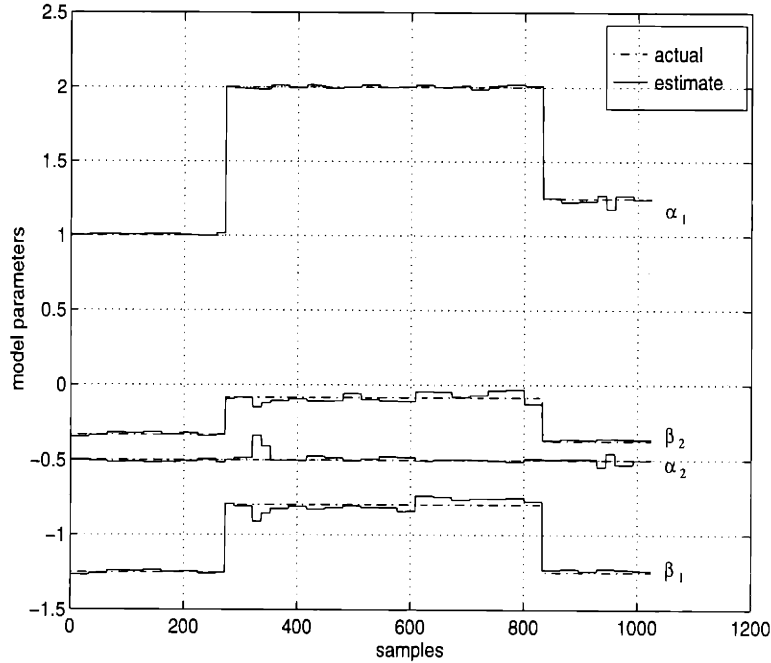


Figure 5-1: Parameter estimates for the ARX model (case 1)

A priori knowledge about the minimum jump levels were incorporated (denoted by  $\delta_{min}$  in equation 4.44 of the previous chapter), as well as a 85% confidence level that the real parameter is within 20% deviation from the estimate over one interval. The effect of various design parameters was then studied, mainly,

- Sensitivity to noise: as expected a decrease in the signal to noise ratio has a direct impact on short interval identification. If the noise level at the output of model 5.1 is increased by a factor of 2.5 ( $SNR = 14$ ), estimation on disjoint interval length of  $N = 32$  is not satisfactory, as shown in figure 5-2 (denoted by case 2). By increasing  $N$  to  $2^6 = 64$  sample points, however, we are gaining necessary confidence in the estimates (figure 5-3). This falls under the traditional conflict of noise and estimation accuracy and is application dependent; in modeling more sensitive processes, adopting several identification algorithms running in parallel at different time epochs, or using a sliding window in the estimation procedure may save a “wait” time, however under higher computational expense.
- Sensitivity to “tuning” parameters : that is, the confidence levels utilized in the two hypothesis decision rules: first, upon selecting relevant bases on one interval ( $\alpha, \gamma$  in equation 4.23),

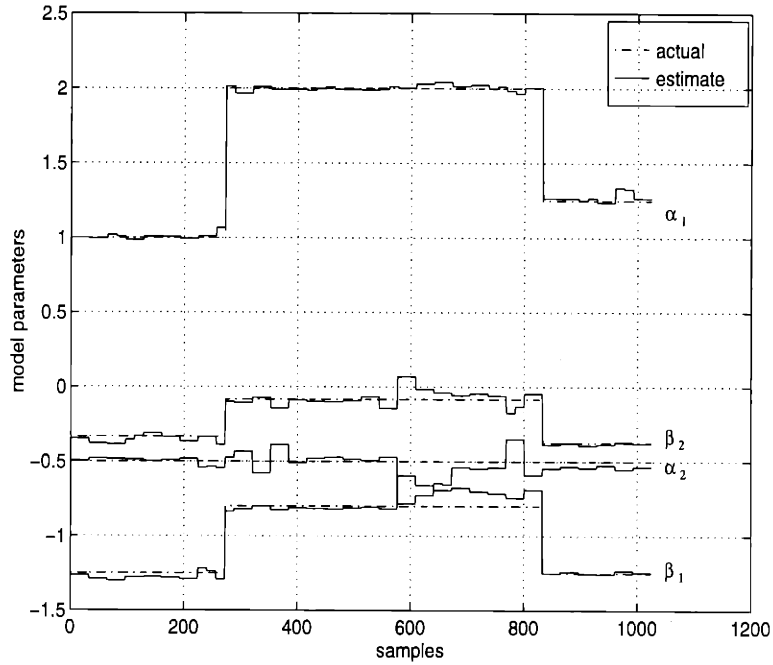


Figure 5-2: Effect of noise increase in short identification intervals on parameter estimates of ARX model (case 2)

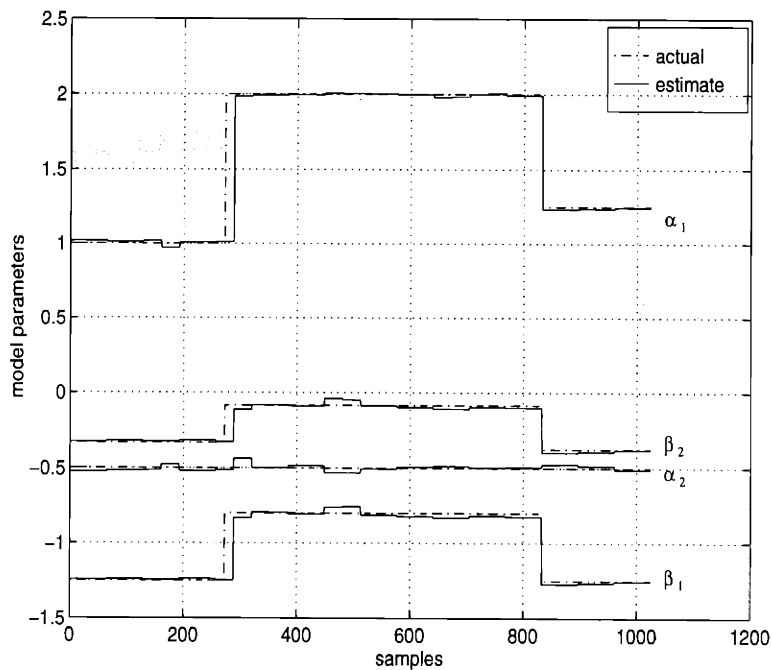


Figure 5-3: Improved parameter estimates for the ARX model (case 2) as longer disjoint intervals are used

and second, upon deciding whether two adjacent estimates are identical ( $\delta_{min}$ ). It has been experimentally noted that the criterion (4.23) which gives more confidence



in the variance of the estimate is useful in cases where using  $AIC$  only may tend to under estimate the correct order [5], in particular when the ratio  $\frac{d\mu}{N}$  is large in equation 4.21. Both decision rules allow assignment of  $\alpha$ ,  $\gamma$  and  $\delta$  for each the parameter models separately, although the reported results above were obtained for the same setting of these values.

**Note** : the plots given above are *delayed in practice* by one time unit ( $N$  sample points). The exact matching onto the real parameters serves for proper comparison. Actual behavior is presented, for instance in the plot shown next (figure 5-4).

### 5.1.1 Comparison with Standard Methods

In comparison with known on-line identification mechanisms mainly forgetting factor or information discounting methods (chapter 2), results are shown in figure 5-4. Being a natural basis for jump representation, Haar wavelets are seen to adapt more accurately to fast and more frequent jumps. Better isolation and estimation of a faulty parameter is also achieved due to the parameter separation structure used. At higher levels of noise, longer disjoint estimation intervals may be inappropriate if fault occurrence time is of high priority as it becomes increasingly delayed. The general profile, however, is more accurate due to decisive averaging out of the noise in no-change intervals as opposed to uniform discounting of old information in the conventional method.

### 5.1.2 Extensions to Other Applications

Although the focus of the algorithms has been on identifying jumps, use of the presented methodology in *recursive* identification of time varying phenomena can be achieved if proper *orthogonal* wavelet basis are chosen. For instance, one can look in a single interval for the relevant basis functions using the *VALIDATE* algorithm and subsequently look for common characteristics among adjacent disjoint intervals. This will possibly allow moving into lower resolutions thus reducing the localized effect of noise in estimation (the projection might be sub-optimal, cf. sec 6.2), all under a banner of isolated estimation of the parameters of the system model used and the associated flexibility in adding a priori knowledge into the estimation problem design.

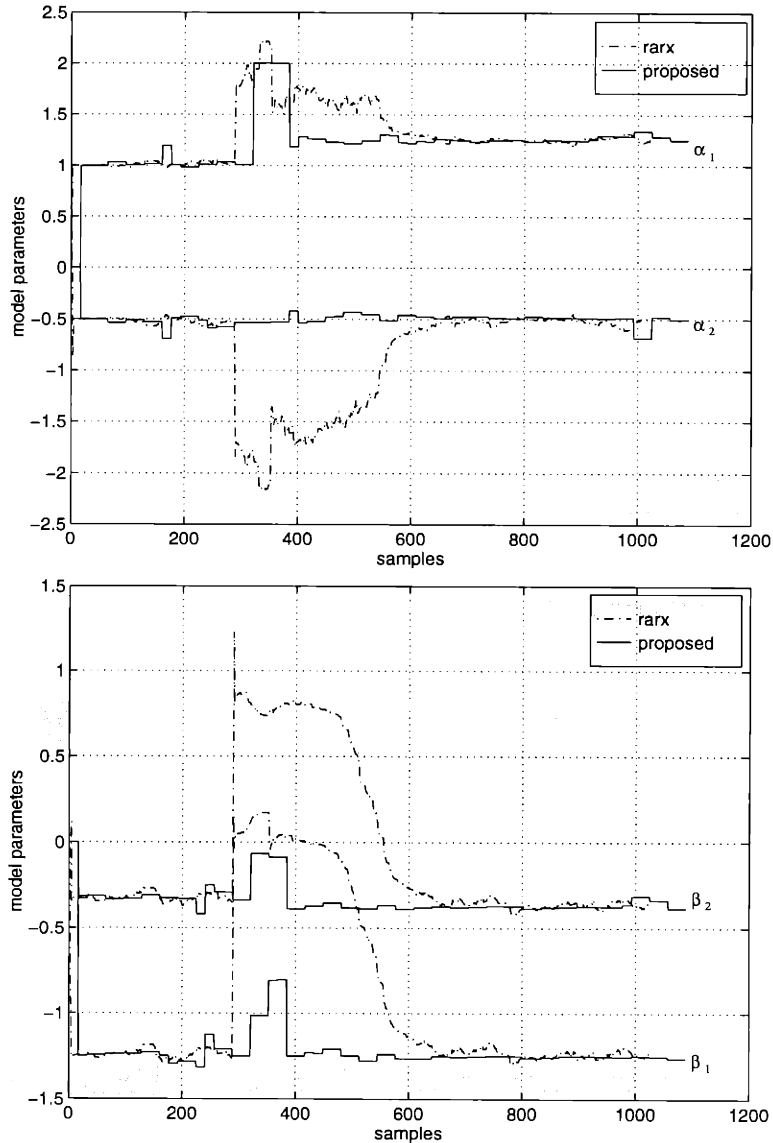


Figure 5-4: Comparison of the proposed algorithm with recursive ARX algorithm with constant forgetting factor

**A word on parsimony** The bottom-up-approach of regressor selection penalizes choosing irrelevant regressors and trades-off the complexity of an obtained model with the associated approximation error. With all the properties characterizing a wavelet expansion discussed in chapter 3, we can conclude that a proper choice of filter banks will result in the concentration of most of the signal energy in the low resolutions of the expansion [1]. Hence, we have a compression effect of the amount of information needed to describe a parameter profile. In the case of jumping parameters detailed earlier, efficiency in the parametrization

is apparent when, for instance, a model parameter undergoes a series of jumps for a short period of time and then settles later to a constant value. Numerous other examples can be drawn onto other orthogonal wavelet basis function, all emphasizing the “zoom in” property of such localized, scaled, and translated bases.

## 5.2 Closed Loop Simulations

Consider the tracking problem of a step input command applied to an unknown but pre-specified order plant  $P$  and controlled by  $K$  whose specifications were given in chapter 2 and which is designed according to the algorithm set forth in chapter 4. The plant model, a second order  $ARX$ , incurs a series of jumps, some of which are actually open-loop unstable, as follows:

$$\begin{aligned}
 y(n) &= u(n-1) + u(n-2) - 1.25y(n-1) - 0.667y(n-2) & n \in [1, 128] \\
 &= 2u(n-1) - 0.5u(n-2) - 0.8y(n-1) - 0.083y(n-2) & n \in [129, 272] \\
 &= 1.5u(n-1) - 0.5u(n-2) - 1.25y(n-1) - 0.375y(n-2) & n \in [273, 384] \\
 &= u(n-1) + u(n-2) - 1.25y(n-1) - 0.667y(n-2) & n \in [385, 512]
 \end{aligned} \tag{5.2}$$

The all stabilizing controller is designed using the bezout identity at each time unit (equation 4.45). The tuning parameter  $Q$  is described by a second order  $AR$  model, allowed to be time varying.

$$u = Qe : \quad u(n) = q_0(n)e(n) + q_1(n)e(n-1) + q_2(n)e(n-2) \tag{5.3}$$

where  $u$  and  $y$  are computed according to equation 4.46.

A closed loop diagram of the controlled process is shown in figure 2-1. A simulated system response under frozen time design in disjoint 32 sample point intervals is shown in figure 5-6 along with the  $Q$  parameter and the plant estimates in figures 5-5 and 5-7.

The plant estimator behaves essentially in the same manner as presented in the previous section on open-loop system and hence imposes a limitation on the tracking speed once a jump in the plant model occurs. The given model satisfies the assumption that performance does not deteriorate unboundedly in the time needed to properly identify and estimate the jump, which is in turn obviously related to the level of noise and excitation persistency of the available data (cf. sec 5.1). Frequent variations in the plant will help avoid losing the

necessary richness in the input signal, a potential problem that often arises in closed loop identification under the given tracking constraint when excitation decreases after some time due to minimization of error signal (input to the plant).

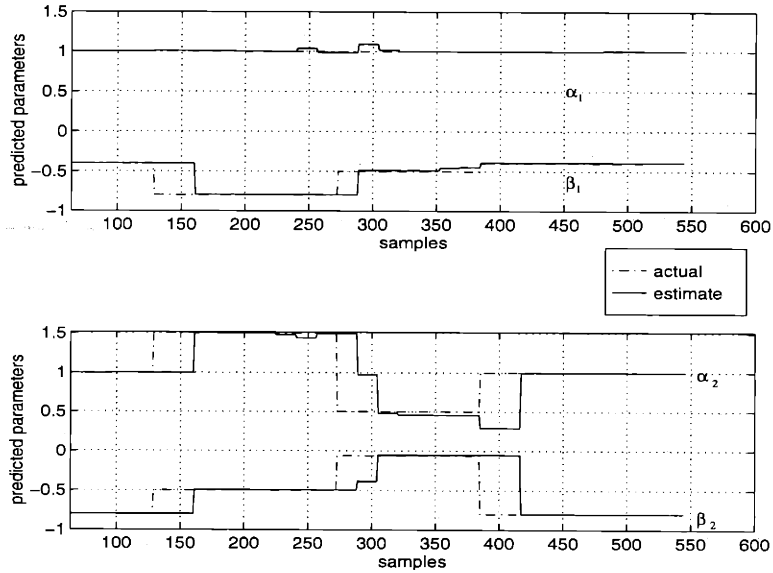


Figure 5-5: Estimates of the plant parameters operating in closed loop

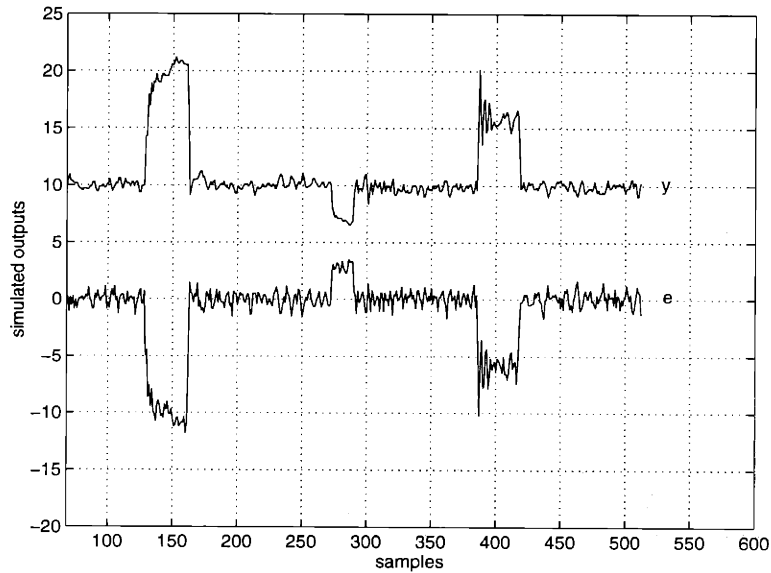


Figure 5-6: Plant output ( $y$ ) and tracking error ( $e$ ) under frozen time controller design

It is worthwhile to note here that although the parameter  $Q$  is allowed to be time varying, it actually had a constant value over one interval (figure 5-7). In another simulation on a 4th order plant model ( $p = q = 4$ ), a time invariant model of  $Q$  over one time unit yielded

the same error residual as when a time varying model is used over the same interval. In principle, a TV solution for  $Q$  was sought to investigate whether initial transients that occur after a jump can be swiftly damped over a given time unit using a linear time varying controller as opposed to the conventional LTI design. However, the examples sought did not yield sufficiently affirmative results.

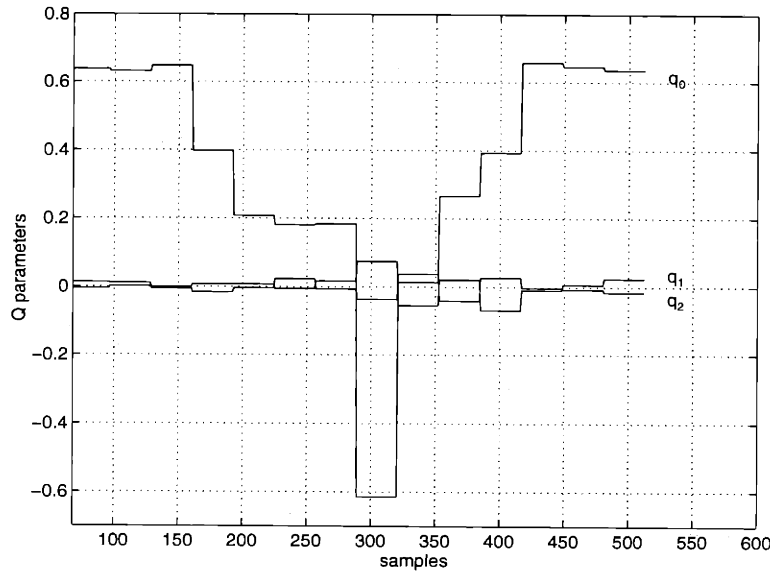


Figure 5-7: Tuning parameter  $Q$  for the frozen time controller design

### 5.3 Convergence and Stability

In performing on-line estimation of the parameter profile, priority was given in the developed algorithms for *convergence to the task* of first isolating the unknown change and second estimating the state of the parameters after changes occur. It was demonstrated that under low levels of noise the estimates track jumps accurately and fast. As presented in chapter 4, performance is measured in terms of both estimate validity over one time unit and estimate improvement over adjacent time units.

Consider the case where a jump occurs at time  $t_1$  and no changes occur thereafter. Two possibilities of the location of  $t_1$  with respect to the identification blocks exist, namely, within one time unit or between two adjacent time units. If  $t_1$  falls within some time unit of identification then, with previous assumptions on data quality, the expansion basis can lock onto the correct jump profile. Even under higher levels of noise, the bottom-up approach can be tuned to penalize higher order models, thus avoiding unnecessary false

jump estimates. On the other hand, care must be taken as to not increase the chance of a jump miss. In the event of a miss, independent identification over following time units will lead to geometric decrease of the probability of sustaining that miss.

If  $t_1$  occurs between two adjacent intervals a hypothesis test is used to decide whether a jump has occurred or the bias in the estimate between the two consecutive intervals is merely due to noise, with a probability of not detecting such a jump of  $P_m$  (probability of miss, equation 4.40). In steady state operation where no changes occur, and as more data becomes available, the estimated parameters are continuously projected onto lower resolutions thus becoming more accurate due to decrease of noise effect. This estimate improvement which is conducted whenever no changes occur in a particular parameter, is carried for any time unit with a confidence of  $1 - P_f$ ,  $P_f$  being the probability of falsely alarming that a jump does exist (section 4.3.2). Hence, the hypothesis test serves in connecting the identification on two disjoint time units and sustains a maximum probability of erroneous decision given in equation 4.42 and 4.43:

$$\mathcal{P}_{error}^{max} = \max_{i,j} \Phi\left(\frac{\delta_{min}}{2\tilde{\sigma}_{i,j}^{(dl)}}\right) \quad i = 1 \dots m_f, j = 1 \dots p_\theta \quad (5.4)$$

which is related to a priori knowledge of jump levels ( $\delta_{min}$ ) and data quality ( $\sigma_{i,j}^{(dl)}$ ).

It is believed that although the history profile of the system trajectory may not be identified exactly, fast tracking of jumps is our highest priority for such on-line operation and control of the system. Global approximation can be performed in an off-line manner similar to that presented in [1], or simply by using the *VALIDATE* algorithm onto whole past data, with the computational advantage of a bottom-up approach and recursive channel updates.

We will next discuss the stability of the overall closed loop adaptive control system. As seen earlier, an essential assumption is that the closed loop system response will remain bounded in the inevitable delay time following a jump and required to properly identify the change and apply the corresponding control action (section 4.1).

In finding the frozen time controller at time epoch  $d_l N$ , the plant estimate is treated in a *certainty equivalence* manner, that is, assuming it were the true plant we are trying to control. Consequently, obtaining a good estimate of the model is required for both stability and desirable performance. When compared with conventional methods, the fast

tracking, or lock onto, of the estimate after a change is detected (figure 5-4) will help improve performance in a shorter period of time. On the other hand, the delay incurred in obtaining an estimate due to the block-recursion used will leave the closed loop system with the inappropriate control for some non-zero period of time. The length of this “grey operation” period ranges as follows. Assuming that the estimate is computed at time  $t_e$  using data in  $[t_e - N + 1, t_e]$ , the delay is a minimum of 2 sample points (jump occurs at  $t_j = t_e - 2$  and is detected at the finest resolution) and a worst case of  $t_j = N$  samples assuming close estimates are obtained (otherwise the worst case is dictated by the missed detection as presented earlier in this section). Consequently, such a delay imposes the constraint that, in addition to boundedness of signals during the identification period, no changes occur in the following interval of at least  $N$  sample points (else an incorrect controller will be designed). In other words, the frequency at which changes occur is less than the estimation/control time requirements.

Under the above precautions, the systems simulated exhibited acceptable performance. Stability proofs has not been conducted, yet possible approaches to prove stability mainly by using Lyapunov theory, and others to decrease the delay in detection mainly by using sliding windows for identification, are presented in the next chapter.

## Chapter 6

# Conclusions

### 6.1 Summary

This research has ventured into applying a multiscale basis expansion approach to estimating a simple *ARX* process model with jumping parameters and subsequently using similar multiresolution approach to design a frozen time controller to stabilize the closed loop system and to achieve tracking. The main motive has been the localization properties of wavelets that can offer an efficient and effective description of details as well as long term behavior through various scales and translates of a compactly supported function.

By expanding the parameters of the model using wavelet basis, the identification problem is transformed into computing a set of time invariant linear weights of these basis functions. In general, expansion onto many scales form an underdetermined system of equations where the significant regressors should be selected ( much like avoiding overfitting in a polynomial basis case). [1] describes an up-to-bottom approach in searching for the regressors starting with high resolutions and eliminating irrelevant basis functions (or the overfits) in an off-line manner. This, however, is highly computational, may require validation data sets, and assumes no knowledge about the system. We noticed that a better approach, especially in on-line application where computational speed becomes crucial, is a bottom-up approach where more details are added according to whether or not local behavior is dominant at a particular instant and how well low resolutions of the basis describe the global system behavior. That is, we need only to “zoom in” using higher details at instants where insufficient parameter approximation exists due to local behaviors. This is precisely the approach used in our setting of systems with abrupt changes and the Haar wavelets that become a natural way to describe it . Normal operation corresponds to constant parameters, or



low resolutions, and jumps correspond to details, or higher resolutions. The bottom-up approach is applied onto blocks of incoming data and utilized efficient multichannel order update algorithm [2] to recursively create models of increasing order. In deciding among candidate models, Akaike's objective order selection criterion AIC, as well as confidence levels on the variance of an obtained estimate in the presence of gaussian white noise, are utilized. Therefore, we were able to describe the system behavior in a recursive sparse way which seems efficient to adopt as a building block of an *on-line* procedure.

The on-line identification procedure is block-recursive in nature; that is, it applies the bottom-up approach on consecutive short disjoint blocks of data. Hypothesis testing is then performed to compare the expansion coefficients of corresponding basis functions on adjacent intervals. It is decided whether a change has occurred, in which case the new estimate is adopted, or the coefficient remained constant, in which case the estimate of the coarse resolution is improved by expanding onto even coarser resolutions (longer scales).

The developed identification algorithm thus moves along the time axis isolating and "zooming in" to the parameters of the model that exhibit changes while simultaneously improving the estimate of other parameters that remain constant. Due to the short-interval identification and the localization properties (especially the decoupling character) of the Haar basis, low levels of noise were tolerable, with the obvious price of requiring longer intervals in the presence of higher noise levels.

Frozen time control is then designed as new estimates of the plant model are obtained. The design criterion is casted in an analogous manner to the identification problem: selecting a (possibly time varying) control tuning parameter to achieve tracking. Although a time-varying (step-changing) control was designed, it was not clear that such a controller will outperform a time invariant solution to tracking-error minimization over short intervals of time.

## 6.2 Directions of Future Research

Future extensions of this thesis is mainly two folds: first, generalization of the estimation scheme to include a wider class of time varying systems, and second, stabilization and performance of the on-line control scheme.

The work done utilized Haar basis functions which form an orthonormal set of non-overlapping wavelet basis. Many of the ideas presented in the identification algorithm are

extendible to a general orthogonal wavelet setting. The bottom-up approach in adding finer details of the parameter profile is directly applicable to many compact basis setting. Extending the block recursion can become attractive as an on-line tool to “zoom in and out” over the global profile locking into changes in the event these occur yet improving the estimate when no changes in the constituent basis happen, all under the banner of isolated parameter estimation. In its present form, this recursion is applicable to periodic systems (or those processes that actually satisfy the assumptions on how data behave outside a given short time interval - in our case periodic extensions were employed). For other general models, this recursion can be mapped in a sub-optimal manner; that is, two adjacent blocks can be used to improve the estimate over the whole time support of both intervals; however, the resulting estimate is not necessarily equivalent to the “global” estimator over the two-interval data record. This is mainly due to end-effects in estimation over finite data sets which is approached here by periodic extensions ([9], chapter 8). Hence, future generalization of the algorithm should consider conditions on how well the short-interval recursion performs when compared to an off-line estimate such as that given in [1].

Stability of the estimation/control scheme should be analyzed. The adaptive control problem at hand is complex in nature as it deals with a time-varying plant with stochastic noise. Hence, a reasonable way of approaching an analysis is to start-off with conservative conditions on the system behavior as well as bounding of the noise effect. Even in such a deterministic setting, common assumptions on slow-time varying system behavior such as those presented in [25] and [26] are not satisfied in the case where jump changes occur. One might think of a Lyapunov theory approach, such as the one outlined next.

Following a jump in some parameter, we saw that a non-zero delay will exist before a correct estimate is obtained and a stabilizing controller is applied. For this “grey” operation interval, a bound on the error should be found. A potential approach, although may prove conservative, is through transforming the *ARX* model description into a state space model, and trying to relate the increase in the state magnitude to the distance between old system model (to which controller was actually correct) and post-jump system model (to which the old controller is incompatible). Next, as the model estimate and controller are corrected, a Lyapunov function which decreases along the system trajectory should be sought and proven to bring the system state back to within the same bounds that prevailed before the jump happened. The time delay needed for this error to decrease will be a practical

limitation on how often jumps in the system can occur. Such limits can be subsequently compared with corresponding tradeoffs other adaptive control methods possess. Finally, questions on how a time-varying controller (utilized in the presented scheme) performs over relatively short time horizons in comparison with the time-invariant counterpart also need to be addressed for a more thorough comparison with frozen time controller design.

# Bibliography

- [1] M.K. Tsatsanis, G.B. Giannakis, "Time Varying Identification and Model Validation Using Wavelets", *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, pp. 3512-3523, December 1993.
- [2] G.O. Glentis, N. Kalouptsidis, "Efficient Multicannel FIR Filtering Using a Single Step Versatile Order Recursive Algorithm", *Signal Processing* 37 (1994), pp. 437-462.
- [3] Y. Gernier, "Time-Dependent ARMA Modeling of Nonstationary Signals", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-31, No. 4, pp. 899-911 August 1983.
- [4] A.S. Willsky, "A Survey of Design Methods for Failure Detection in Dynamic Systems", *Automatica*, Vol. 12 (1976), pp. 601-611.
- [5] K.J. Astrom, "Maximum Likelihood and Prediction Error Methods", *Automatica*, Vol. 16 (1980), pp. 551-574.
- [6] K.J. Astrom, P. Eykhoff, "System Identification- A Survey", *Automatica*, Vol. 7 (1971), pp. 123-162.
- [7] I. Daubechies, "Ten Lectures on Wavelets", CBMS-61 Siam, 1992.
- [8] I. Daubechies, "The Wavelet Transform, Time-frequency Localization and Signal Analysis", *IEEE Transactions on Information Theory*, Vol. 36 (1990), pp. 961-1005.
- [9] G. Strang T. Nguyen, "Wavelets and Filter Banks", Wellesley-Cambridge Press 1996.
- [10] L. Ljung, "System Identification, Theory for the User", Prentice Hall 1987.
- [11] D. Donoho, "Denoising by Soft Thresholding", *IEEE Transactions on Information Theory*, Vol. 41 (1995), pp. 613-627.

- [12] R. Isermann, K.-H. Lachmann, D. Matko, "Adaptive Control Systems", Prentice Hall 1992.
- [13] K.J. Astrom, B. Wittenmark, "Adaptive Control", Addison-Wesley, 2nd edition, 1995.
- [14] G.C. Goodwin, K.S. Sin, "Adaptive Filtering Prediction and Control", Prentice-Hall, 1984.
- [15] K. Narendra, A. Annaswamy, "Stable Adaptive Systems", Prentice-Hall, 1989.
- [16] S. Sastry, M. Bodson, "Adaptive Control", Prentice Hall, 1989.
- [17] M. Morf et al, "Efficient Solutions of Covariance Equations for Linear Prediction", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-25, No 5, October 1977, pp. 429-433.
- [18] G.O. Glentis, N. Kalouptsidis, "Fast Adaptive Algorithms for Multichannel Filtering and System Identification", IEEE Transactions on Signal Processing, Vol. 40, No 10, October 1992, pp. 2433-2458.
- [19] S. Mallat, "A Theory of Multiresolution Signal Decomposition: the Wavelet Representation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No 7, July 1989, pp. 674-693.
- [20] J. Pesquet, H. Krim, Carantan, "Time Invariant Orthonormal Wavelet Representations", LIDS report, LIDS-P-2261.
- [21] E. Kamen, P. Khargonekar, K. Poolla, "A Transfer Function Approach to Linear Time-Varying Discrete Time Systems", SIAM Journal on Control and Optimization, Vol. 23, No 4, July 1985, pp. 550-565.
- [22] B. Baskshi, G. Stephanopoulos, " Wave-Net: A Multiresolution, Hierarchical Neural Network with Localized Learning", AIChE Journal, Vol 39, No 1, January 1993, pp. 57-81.
- [23] J. Williams, K. Amaratunga, "Introduction to Wavelets in Engineering", International Journal for Numerical Methods in Engineering, Vol 37, 1994, pp. 2365-2388.
- [24] R. Johansson, "System Modeling and Identification", Prentice Hall, 1993.

[25] M. Dahleh, M. A. Dahleh, "On Slowly Time-Varying Systems", MIT-Technical Report LIDS-P-1852.

[26] M. Dahleh, M. A. Dahleh, "A Class of Adaptive Controllers with Application to Robust Adaptive Control", MIT-Technical Report LIDS-P-1759.