THE UNIVERSITY OF CALGARY

Identification of Linear, Time-varying Systems

by

Sreemoyi Sanyal

A

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL AND COMPUTER

ENGINEERING

CALGARY, ALBERTA

December, 2005

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a entitled "Identification of Linear, Time-Varying Systems" submitted by Sreemoyi Sanyal in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

Supervisor, Dr. David T. Westwick
Department of Electrical and Computer Engineering

Dr. Anders Nygren
Department of Electrical and Computer Engineering

Dr. Brent J. Maundy
Department of Electrical and Computer Engineering

Dr. Kristina D. Rinker
Department of Chemical and Petroleum Engineering

22-12-05

Date

Abstract

System identification creates mathematical models of systems using measurements of their inputs and outputs. System identification techniques have been used in a variety of applications including in biomedical, control, and aeronautical engineering. Linear time-invariant models have been widely used in all these applications since they provide a simple description that nonetheless predicts the system's behaviour within acceptable errors. In some cases, linear, time-varying models can be used to provide more accurate descriptions of various non-linear or time-varying phenomena, while still retaining a relatively simple description. This thesis reviews two existing approaches for identifying time-varying systems: Temporal basis expansions and Ensemble techniques. This thesis presents two new algorithms for identifying linear time-varying systems including a novel term-selection technique. The algorithms have been demonstrated by applying them to ankle compliance dynamics and to experimental human elbow stiffness data

Acknowledgements

I would like to express my sincere gratitude and acknowledgement to Dr. David Westwick for his continuous encouragement, help, support and constructive guidance throughout the research and writing of this thesis. I would also like to thank him for accomodating all my various requests over the last two years.

I acknowledge the Natural Science and Engineering Council and the Department of Electrical and Computer Engineering for their financial support.

I would like to acknowledge Dr. Eric Perreault for providing the experimental data.

I would like to thank all my wonderful, amazing friends all over the world and my family in India for all their encouragement, support and all the MSN chats that helped me through the ups and downs of my research.

Finally, I thank my parents for always believing in me and pushing me to constantly challenge myself. I would never have accomplished so much without them.

In loving memory of Mamoni

And To

Dudunbhai, Dada

Ma, Babai

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Much of engineering deals with the analysis and manipulation of mathematical models. Many different design procedures, from different facets of engineering, including the design and analysis of control systems, electronic circuits and bridges consist of manipulating and analyzing mathematical models of the system under consideration. These mathematical models are either generated by *a priori* modelling or by system identification techniques.

System identification involves creating mathematical models of systems using measurements of their inputs and outputs. System identification, unlike mathematical modelling, is a top-down approach where minimal knowledge of the system characteristics is required.

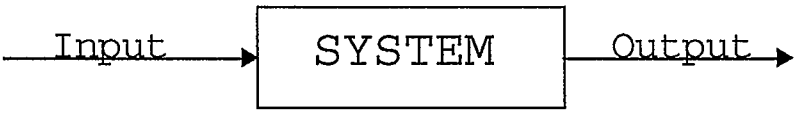

Figure 1.1: System Identification Block Diagram

As shown in Fig. 1.1, the system is treated as a black-box for system identification purposes. The system transforms the inputs into the outputs, but few assumptions are made regarding how that is done. System identification is the process of filling in the black box with a mathematical model of the system.

Specialized system identification techniques have a wide-variety of applications which are reflected in this thesis. Applications of these methodologies include modelling control systems, various physiological systems, aerodynamical systems and telecommunications systems.

As varied as the applications of system identification are, the models themselves vary tremendously as well. The models come in various degrees of accuracy and complexity. They range from static, linear and time-invariant to dynamic, non-linear and time-varying. The art of modelling a system properly comes from choosing a model with the simplest description which still gives a useful level of accuracy. Linear time-varying models form a very important sub-class of models that require the use of specialized system identification techniques. They are often a reasonable compromise between simplicity and accuracy - either for weakly nonlinear systems, or for truly time-varying systems. Thus, many biomedical, aerodynamical, and telecommunication systems can be modelled simply and accurately as linear, time-varying systems.

State-of-the art techniques in this field include using temporal basis expansions, [12], [41], [53] and ensemble techniques [32], [34]. In the temporal basis expansion technique, each time-varying parameter is expanded onto a set of time-varying basis functions. Thus, each time-varying parameter is represented as a weighted sum of basis functions. The weights, which are time-invariant, together with the temporal basis functions, fully describe the time-varying parameter. As a result, each time-varying parameter is replaced by a vector of time-invariant weights. One of the disadvantages of this method is the huge number of parameters that need to be estimated, which increases the sensitivity of the parameter estimates to noise in the

measured data. A number of term-selection techniques like the Fast Orthogonal Search (FOS) [24], and the Optimal Parameter Search (OPS) [33], [53] have been used in the past to deal with this issue. The FOS and OPS methods, however, perform local searches, and cannot be guaranteed to find the globally optimum model structure. In the ensemble technique, the model is not identified through time but across an ensemble of trials. An ensemble of input-output realizations is used to identify the time-varying dynamics at each point in time. The practical application of the ensemble method is very limited since it is extremely difficult to consistantly repeat experiments enough times to obtain the necessary sets of input-output data.

In this thesis a novel method of identifying linear time-varying systems is discussed. The method combines temporal basis expansions and the ensemble technique. The time-varying parameters are expanded onto a set of basis functions and multiple ensembles are used in the final estimation of the parameters and the final term-selection is done using Least Absolute Shrinkage and Selection Operator (LASSO).

The body of this thesis is organized into three main chapters. The second chapter of the thesis includes an extensive literature survey. It describes in detail all the background material necessary for understanding this research area and discusses the two state-of-the art techniques in the field: temporal basis expansions and ensemble techniques.

The third chapter of the thesis deals with a combination of the two techniques (basis expansions and the ensemble technique) [41]. The first section of this chapter deals with the theory behind this new technique. Then, a simulation involving human ankle dynamics is described in detail. The last section of this chapter presents the

results obtained from the simulation and a discussion on the validity of this technique.

The next chapter of the thesis deals with Temporal Basis Expansions using an optimal term selection strategy. A novel term-selection technique, LASSO, is introduced in this chapter [43]. The first part of this chapter introduces LASSO and gives a theoretical description of the technique. The next part discusses the algorithm (temporal basis expansions and LASSO in conjunction) in detail. Next, the algorithm is demonstrated in human elbow stiffness dynamics. The application is presented in detail including discussions of accuracy and significance of the results obtained from the analysis.

Finally, the thesis ends with a conclusion which includes a brief discussion of future work and a list of personal contributions made to the project.

Chapter 2

Background

This chapter contains an overview of all the background material relevant to this thesis work. It starts with an overview of system identification. Focusing on linear systems, the concepts behind the identification of linear time-invariant and time-varying systems are reviewed. Then, the current state-of-the art techniques in the field of linear time-varying system identification: Basis Expansions and Ensemble Techniques, are discussed in detail.

2.1 Notation Conventions

Unless specified otherwise, discrete time systems and signals will be used throughout. Thus, t, which represents time, will be an integer. The forward shift operator will be denoted by q, so that

$$qy(t) = y(t+1)$$

Raising q to the power k results in a forward time shift of k samples. In particular, setting $k = -1$ results in a backward time shift

$$q^{-1}y(t) = y(t-1)$$

In this thesis the symbol $\hat{}$ has been used to denote an estimate for example, $\hat{u}$ denotes an estimate of u. Unless specified otherwise, N denotes the total number of data points. The input and output are denoted by $x(t), y(t)$ respectively and Z^N is

a collection of N inputs and outputs

$$\mathbf{Z}^N = \begin{bmatrix} u(0) & y(0) \\ u(1) & y(1) \\ \vdots & \vdots \\ u(N) & y(N) \end{bmatrix}$$

Matrices are denoted by upper case bold letters and $\mathbf{M}(i,j)$ means an element of matrix $\mathbf{M}$ in the i^{th} row and j^{th} column. Vectors are denoted by lower case bold letters

$$\mathbf{y} = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}$$

2.2 System Identification

2.2.1 Definition

System identification refers to a set of techniques that create mathematical models of systems using measurements of their inputs and outputs. Thus, unlike mathematical modelling, system identification is a top-down approach, that is, the system can be treated as a black box and minimal *a priori* knowledge of system characteristics is needed. The four basic entities required for system identification are [31]:

1. A data set (input and output data)

2. A set of candidate models

3. A rule or process for model selection

4. A process for validating the selected model.

The input and output data are collected from measurements of the physical system. The next step is to choose an appropriate set of candidate models to describe the system. For example, one must decide whether to use models that are linear or nonlinear, time invariant or time-varying etc. Once that choice has been made, the model is further specialized. For example, if the system is to be linear and time-invariant, one could represent it using an impulse response or a frequency response or even a state-space model. Then a specific rule/process is identified to select the best possible model: for example, the model that describes the system most accurately, out of the set of candidate models. The last step is the validation step, where the selected model is tested by using it to predict the response to novel data.

2.2.2 Applications

System identification has applications in various different fields including control system design, [3], [17], biomedical engineering (systems physiology), [2], [34], aerodynamics [7], and wireless communications, [6], [51]. In the area of control systems it is very important to have a mathematical model of the system that is to be controlled. In this area the applications emphasize short-range prediction of the system state, typically only one time-step into the future. In addition, recent model outputs may be used if the prediction is limited to one-step in the future. Thus, recursive model structures are commonly used in control applications. The identified model may be used online, or it may be used to design a controller. In either case, a simpler model is preferred. Since the emphasis is on one-step-ahead prediction, linear models are often adequate. Hence, controls applications often use low-order parametric

system models. A major exception is in Model based Predictive Control (MPC), where the system model is used to predict many steps into the future [4], [9]. In those cases, more accurate models are often required, and so more complex model structures are sometimes used.

The physiological systems encountered in biomedical engineering are generally non-linear and time-varying. A great deal of research is being done in this field to model various sensory [8], [14], neuromuscular, [20], [46] and cardiopulmonary [50], [52] systems in order to gain a better understanding of their function and of the human body in general. Biomedical applications emphasize system analysis and understanding more so than short-range prediction. Models of physiological systems are often used in simulations, so, again, complex system descriptions are often needed (as in MPC) since recent system outputs are not available for use.

2.2.3 Model Selection

Once a suitable model class (set of candidate models) has been selected, the next step according to the list in section 2.2.1 is model selection. Model selection for a system becomes a search for the best model within the set using a particular rule. All of the models in a given model class can be represented by a vector, the parameter vector $\boldsymbol{\theta}$, containing all of the variables that define the model. The search for the best model is thus equivalent to the search for the best parameter vector. Evaluating the candidate models is done by a test which evaluates the models' ability to describe observed data and predict unknown data. Although, a good model should predict novel data well, in the searching process a model that predicts observed data accurately is adequate. A good model is a model that produces small prediction errors when applied to the

observed data. The prediction error is given by:

$$\epsilon(t,\boldsymbol{\theta}) = y(t) - \hat{y}(t|\boldsymbol{\theta}) \tag{2.1}$$

where $\hat{y}(t|\boldsymbol{\theta})$ is the predicted output which is a function of time t, and $\boldsymbol{\theta}$. Let the prediction error be filtered by a linear filter $L(q)$:

$$\epsilon_F(t,\boldsymbol{\theta}) = L(q)\epsilon(t,\boldsymbol{\theta}),\ 1 \leq t \leq N \tag{2.2}$$

The function $V_N(\boldsymbol{\theta}, Z^N)$ is, a well-defined scalar-valued cost function of the model parameter vector, $\boldsymbol{\theta}$ and is defined by:

$$V_N(\boldsymbol{\theta}, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \ell(\epsilon_F(t,\boldsymbol{\theta})) \tag{2.3}$$

where $\ell(\epsilon_F(t,\boldsymbol{\theta}))$ is a non-negative, scalar valued function of its argument, and, $\hat{\boldsymbol{\theta}}_N$ is an estimate of the parameters based on N data points and is defined by the minimization of (2.3)

$$\hat{\boldsymbol{\theta}}_N = \arg\min_{\boldsymbol{\theta}} V_N(\boldsymbol{\theta}, Z^N) \tag{2.4}$$

Here arg min denotes "the minimizing argument of the function". If the minimum is not unique, we let arg min denote the set of minimizing arguments. The filter $L(q)$ in (2.2) acts as a *frequency weighting*. For our purposes $L(q) = 1$ has been used, so all frequencies will be equally weighted in the cost function. The standard choice for ℓ is the quadratic norm given by:

$$\ell(\epsilon) = \frac{1}{2}\epsilon^T\epsilon \tag{2.5}$$

where ϵ is the vector which contains the prediction errors $\epsilon(t)$. The scalar form of (2.5) is given by:

$$\ell(\epsilon(t)) = \frac{1}{2}\epsilon^2(t) \tag{2.6}$$

From Eq. (2.6) V_N is given by:

$$V_N = \frac{1}{2}\epsilon^T\epsilon \tag{2.7}$$

This norm leads to the Least Squares method of model selection which is the most common method for solving Linear System Identification problems. The Least Squares method is described in detail below.

2.2.4 Linear System Identification

Linear system identification is one of the most highly developed "sub-fields" of system identification [31]. It refers to the identification of systems whose outputs are linear functions of their inputs. Figure 2.1 illustrates the linear scaling property.

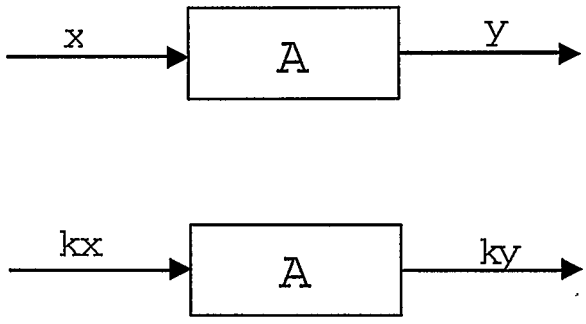

Figure 2.1: Linear Scaling Property

Mathematically that means if $y = Ax$, where x is the input and y is the output and A denotes the linear system, then if the input is scaled by a factor of k, the output will be scaled by a factor of k as well: $ky = Akx$.

As well, linear systems follow the principle of superposition, i.e. if two inputs were added together and passed through a linear system, the final output would be

the same if the the two inputs were passed through individually and their respective outputs were added together: $y_1 = Ax_1$ and $y_2 = Ax_2$, then, $y_1 + y_2 = A(x_1 + x_2)$. Fig. 2.2 illustrates the principle of superposition.

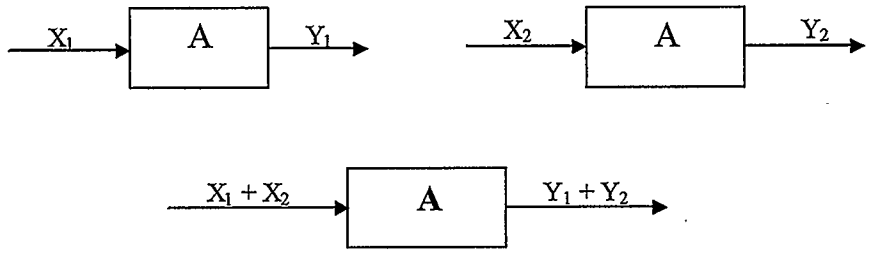

Figure 2.2: The principle of superposition

Linear systems include parametric models such as AutoRegressive with Exogenous input (ARX), AutoRegressive Moving Average with Exogenous input (ARMAX), Output Error (OE), Box-Jenkins (BJ) and state-space models, as well non-parametric models such as Impulse Response Functions (IRFs) and Transfer Functions. A generic linear system is shown in eq. (2.8)

$$y(t) = G(q)u(t) + H(q)e(t) \tag{2.8}$$

where $G(q)$ and $H(q)$ are potentially time-invariant (TI), infinite polynomials in the shift operator q. $G(q)$ filters the input, $u(t)$, while $H(q)$ filters the noise, $e(t)$. Equation (2.8) is the most general form of linear TI systems and can also be written

as two infinite sums.

$$y(t) = \sum_{k=1}^{\infty} g(k)q^{-k}u(t) + \sum_{k=0}^{\infty} h(k)q^{-k}e(t) \tag{2.9}$$

Several special cases of this model will be considered in the next two sections. For linear time-varying (TV) systems, the TI invariant polynomials $G(q)$ and $H(q)$ become time-dependant giving us the following equation:

$$y(t) = G(q,t)u(t) + H(q,t)e(t) \tag{2.10}$$

where, $G(q,t)$ and $H(q,t)$ are time-varying (TV) polynomials in the shift operator q. Under linear system identification, TI linear systems form the most important class of dynamical systems considered in practice and in literature. Even though they represent idealizations of the processes encountered in real life, the approximations involved are often justified, and design considerations based on linear TI theory lead to good results in many cases [31]. The two major system classes under linear system identification include parametric and nonparametric models.

Parametric Models

Parametric models are obtained by replacing $G(q)$ and $H(q)$ with recursive, infinite impulse response (IIR) digital filters, or state space models. Examples of parametric models include ARX and OE models, explained in detail in this section.

The ARX model structure, shown in Fig. 2.3, is linear in its variables. System dynamics can be easily understood using transfer function analysis or IRFs derived from ARX models [33]. The equation for Fig. 2.3 is given by:

$$A(q)y(t) = B(q)u(t) + e(t) \tag{2.11}$$

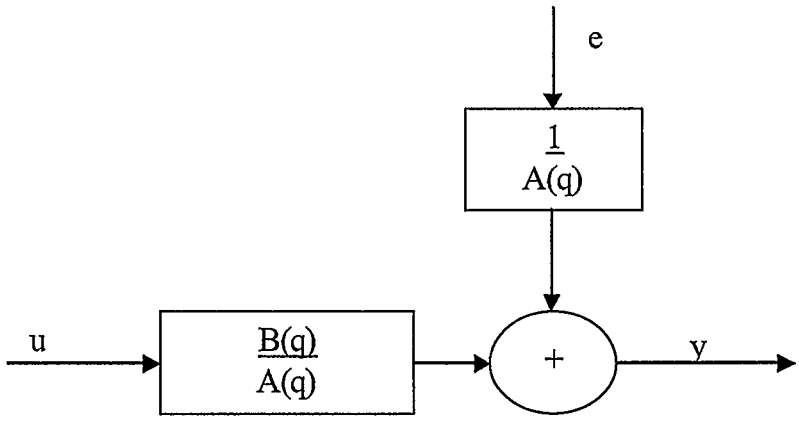

Figure 2.3: The ARX model structure

where $A(q)$ is the polynomial in the shift operator filtering the output $y(t)$, $B(q)$ is the shift polynomial filtering the input $u(t)$ and $e(t)$ is the unknown error in the system. Eq. (2.11) is equivalent to the following difference equation:

$$y(t) = \sum_{j=0}^{Q} b(j)x(t-j) - \sum_{i=1}^{P} a(i)y(t-i) + e(t) \tag{2.12}$$

where $a(i)$ and $b(j)$ are TI ARX parameters that need to be estimated. In this equation t represents time, P and Q represent the degrees of the denominator and numerator polynomials respectively. Since the parameters are time-invariant and they appear linearly in the output, this problem can be solved by linear regression:

$$\mathbf{y} = \mathrm{M}\theta + \mathbf{e} \tag{2.13}$$

where $\mathbf{y}$ and $\mathbf{e}$ are vectors containing the output and noise, respectively, and M is

the regression matrix shown below:

$$\mathrm{M} = \begin{bmatrix} x(0) & 0 & 0 & \dots & -y(1) & 0 & 0 \\ x(1) & x(0) & 0 & \dots & -y(2) & -y(1) & 0 \\ x(2) & x(1) & x(0) & \dots & -y(3) & -y(2) & -y(1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{2.14}$$

The parameter vector, $\boldsymbol{\theta}$, contains the ARX coefficients that need to be estimated.

On the other hand the Output Error (OE) model is a linear model which is non-linear in its variables. See Figure 2.4 below for a block diagram of this model.

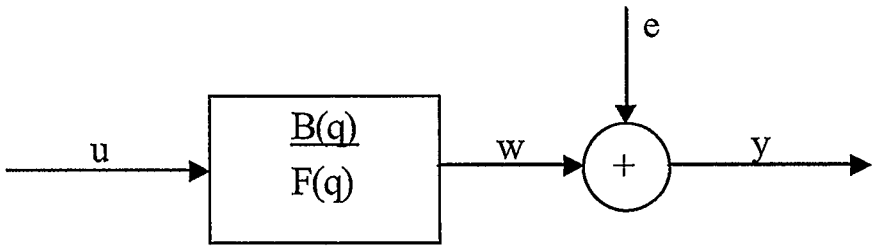

Figure 2.4: Output-Error Block Diagram

Now, writing the relationship between the input and the undisturbed output w as a linear difference equation we get:

$$\begin{aligned} w(t) \; + \; & f_1 w(t-1) + \ldots + f_n w(t-n_f) \\ = \; & b_1 u(t-1) + \ldots + b_{nb} u(t-n_b) \end{aligned} \tag{2.15}$$

the output of the OE model is then given by

$$y(t) = w(t) + e(t) \tag{2.16}$$

Define

$$F(q) = 1 + f_1 q^{-1} + ... + f_n q^{-n_f} \tag{2.17}$$

we can write the Output Error (OE) model structure as:

$$y(t) = \frac{B(q)}{F(q)} u(t) + e(t) \tag{2.18}$$

which is defined by the parameter vector:

$$\boldsymbol{\theta} = [b_1 b_2 ... b_{nb} f_1 f_2 ... f_{nf}]^T \tag{2.19}$$

In the OE model the predicted output is given by:

$$\begin{aligned} \hat{y}(t, \boldsymbol{\theta}) &= w(t) = \frac{B(q)}{F(q)} u(t) \\ &= [u(t)u(t-1)...u(t-n_b, -w(t-1, \boldsymbol{\theta}), ..., -w(t-n_f, \boldsymbol{\theta})]\boldsymbol{\theta} \end{aligned} \tag{2.20}$$

Thus, the regression vector depends on the value of the parameter vector. Nevertheless, the output may be computed using the matrix multiplication:

$$\boldsymbol{y} = \boldsymbol{M}(\boldsymbol{\theta})\boldsymbol{\theta} + \boldsymbol{e} \tag{2.21}$$

where the regressors in the matrix $\boldsymbol{M}(\boldsymbol{\theta})$ depend on the parameter vector $\boldsymbol{\theta}$. Since, the regression matrix depends on the parameters, the OE model is nonlinear in the parameters. Various methods including Bootstrap, Pseudolinear Regression, Instrumental Variables and Gradient-Descent Optimization can be used to identify an OE model [31]. The "bootstrap" method of solving for OE models includes constructing an intial estimate using Least Squares. This initial estimate is set up as if the system were ARX and thus the estimate is biased due to the error in the model structure. Then using the estimates of the parameters, the intermediate signal, $w(t)$,

is generated and used to generate an updated regressor matrix. Using the updated regressor, the parameters are re-estimated, the estimate of $w(t)$ is updated, and the process is repeated a number of times. It is clear that Linear Regression is a much simpler and more efficient way of solving for the unknown parameters, when and if it is applicable. Thus, linear in the variable models are often preferred over other more complex models if the two produce comparable results.

Nonparametric Models

A linear time-invariant system can be described by its transfer function or corresponding impulse response. Nonparametric methods deal with determining these functions directly from the data, without assuming an underlying parametric model structure (such as the ARX or OE models discussed in the previous section). A nonparametric model is represented by its values at a large number of points. Thus, the term "nonparametric model" is a bit of a misnomer since identifying a nonparametric model will require estimating a huge number of "parameters".

The output y(t) obtained from convolution is given by:

$$y(t) = \sum_{\tau=0}^{T} g(\tau)u(t-\tau) + e(t) \tag{2.22}$$

where $g(\tau)$ is the TI impulse response function and $u(t-\tau)$ is the input delayed by τ samples i.e. $q^{-\tau}u(t)$. The above equation is an Finite Impulse Response (FIR) filter, which relating back to equation (2.8) is obtained by truncating the deterministic filter G(q) and setting the noise filter $H(q) = 1$. An example of a TI impulse response function is shown in Figure 2.5.

As with the parametric model, eq. (2.22) can be solved using the linear regression in (2.13), where y is the vector containing the output and M is the regressor matrix

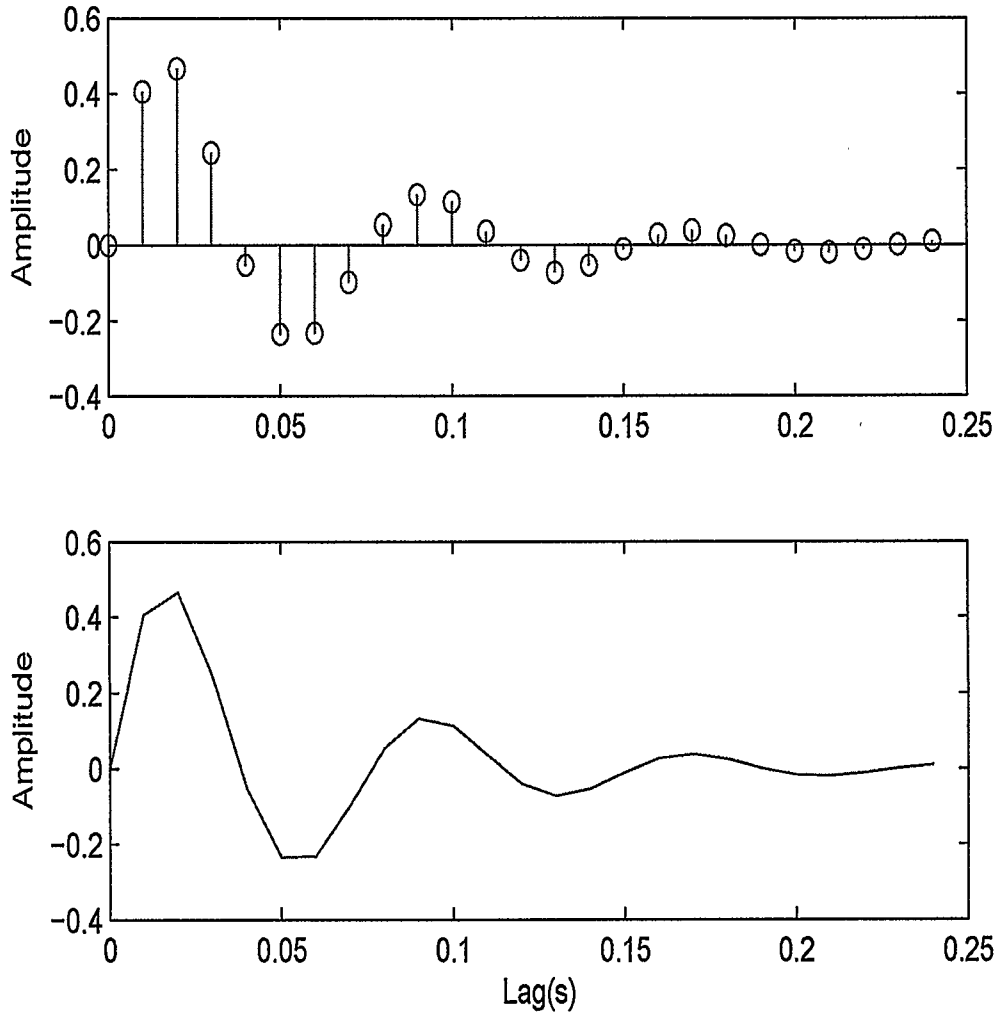

Figure 2.5: Representations of Discrete-Time, Time-Invariant Impulse Response Function. The upper panel shows a discrete time TI IRF sampled at 0.01s. The lower panel shows the linear interpolation of the discrete time IRF, which is the representation used in the body of this thesis.

containing columns of delayed input vectors shown below. The values of the IRFs are contained in $\boldsymbol{\theta}$.

$$\mathrm{M} = \begin{bmatrix} u(0) & 0 & 0 & \dots \\ u(1) & u(0) & 0 & \dots \\ u(2) & u(1) & u(0) & \dots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{2.23}$$

Linear Acausal Systems

Linear acausal systems have also been considered in this thesis [15]. Acausal systems can be described by a two-sided IRF and is shown in Figure 2.6.

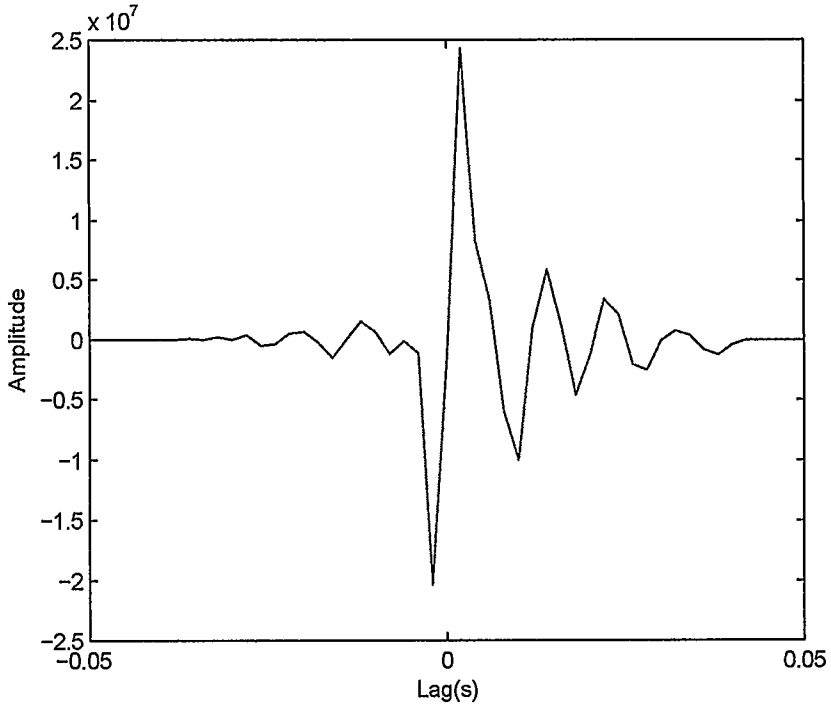

Figure 2.6: 2-sided Impulse Response Function representing acausal systems

Eq. (2.24) describes a two-sided IRF.

$$y(t) = \sum_{j=-Q_{ac}}^{Q_c} b(j,t)x(t-j) + e(t) \tag{2.24}$$

where Q_c is the number of causal terms and Q_{ac} is the number of acausal terms. Note that if $b(j,t) \neq 0$ for $-Q_{ac} < j < 0$, the system will respond to an input before it arrives and hence the system will be acausal [16], [18], [19], [42]. Linear regression can be used to identify the IRF of an acausal system. However, in this case the regressor matrix M will contain both lagged and forward shifted copies of the input.

Least Squares

For FIR and ARX models, Equation (2.1) gives us the prediction error. Now, $\hat{y}(t|\boldsymbol{\theta})$ in equation (2.1) can be calculated using:

$$\hat{y}(t|\boldsymbol{\theta}) = \boldsymbol{M\theta} \tag{2.25}$$

Substituting (2.25) into equation (2.1) we get

$$\boldsymbol{\epsilon} = \boldsymbol{y} - \boldsymbol{M\theta} \tag{2.26}$$

Substituting (2.26) in (2.3) results in

$$\begin{aligned} V_N(\boldsymbol{\theta}, Z^N) &= \frac{1}{2N} \|\boldsymbol{y} - \boldsymbol{M\theta}\|_2^2 \\ &= \frac{1}{2N}(\boldsymbol{y} - \boldsymbol{M\theta})^T(\boldsymbol{y} - \boldsymbol{M\theta}) \end{aligned} \tag{2.27}$$

For a given dataset, Z^N, the function $V_N(\boldsymbol{\theta}, Z^N)$ is a well-defined scalar-valued function of the model parameter vector $\boldsymbol{\theta}$ [31]. The parameter vector that minimizes (2.27) satisfies the least squares criterion, also called the Mean Square Error (MSE).

Differentiating (2.27) w.r.t. $\boldsymbol{\theta}$ and setting the result to 0 solves the minimization given in (2.4) and in closed form we get:

$$\hat{\boldsymbol{\theta}} = [\mathrm{M}^T\mathrm{M}]^{-1}\mathrm{M}^T\mathrm{y} \tag{2.28}$$

Many commercial software packages contain routines for solving Eq. (2.28). For example, it is implemented in $MATLAB^{TM}$ using the backslash operator.

$$\boldsymbol{\theta} = M \backslash \boldsymbol{y} \tag{2.29}$$

Linear regression forms the backbone for solving linear in the variables model structures, such as the FIR and ARX models discussed previously, and is used quite extensively in the balance of this thesis.

2.2.5 Statistical Properties of Least Squares Parameter Estimates

This section establishes the statistical properties of least squares parameter estimates. This includes brief discussions of model accuracy, cross validation and noise sensitivity.

Model Accuracy

Model validation is an integral part of system identification. Once a model has been selected for a particular system, its important to verify the accuracy of that particular model. The percent Variance Accounted For (VAF) is one such measure of model accuracy.

$$VAF = 100(1 - (\mathrm{var}(\boldsymbol{y} - \hat{\boldsymbol{y}})/\mathrm{var}(\boldsymbol{y})) \tag{2.30}$$

where $\boldsymbol{y}$ is the system output and $\hat{\boldsymbol{y}}$ is output of the estimated model.

In-sample vs Cross-Validation

In-sample validation is sometimes used in system identification, due to the availability of only a single data-set. In in-sample validation, the model is validated using the identification data. This however gives biased results because the noise present in the data may have already been modelled.

On the other hand, validating on a novel data-set, cross-validation, gives a more accurate representation of the model's predictive power. Thus, ideally, two data-sets are required for CV, one for identification and one for validation. The common practice is to use the first 70-80% of the data for identification and reserve the last 20-30% for validation. However, this requires having a TI system and having the input that is stationary.

However, having a short data length, TV data, or both limits the use of simple cross-validation, i.e. identifying on 80% of the data and validating on the remaining 20%. Since the usual CV approach is not practical in these scenarios, alternate estimators of a model's predictive power have been proposed. These can be divided into two groups, the first one being methods that subdivide the data "point-by-point" (i.e. 5 or 10 fold CV). The second group consists of methods that make statistical assumptions about the data, and try to infer the predictive power from in-sample results (i.e. Akaike's Information Criteria [1], Approximate Cross-Validation (APCV) [42]).

Fivefold-CrossValidation

Given a data-set of length N, N_v and N_c are integers that list the number of points in the validation and identification sets [23]. So, one set contains N_v points, while

the other contains N_c points, where N_v and N_c are integers and $N_v + N_c = N$. There are $\binom{N}{N_v}$ different subsets of size N_v. For each model M_α, the CV estimate of the squared prediction error is obtained by averaging over all or some sub-sets of size N_v. The model M_α is fitted using the construction data N_c and the prediction error is assessed using the validation data N_v. The squared prediction error is minimized over all the different subsets of the data. However, this becomes quite impractical even for relatively small data-sets, and some kind of simplified short-cut is required. In Tibshirani's paper [43] cross-validation is estimated using fivefold cross-validation. This method is a specific case of the Balanced Incomplete CV (BICV) described in Shao's paper [42] with $N_v = 5$. In Fivefold CV, the data-set is split into two, with every 5th row being put into the validation set. Five such trials are done, the first beginning with row 1, the second with row 2 and so on. The final squared prediction error is obtained by averaging the 5 intermediate prediction errors obtained from each of the 5 different models.

2.2.6 Noise Sensitivity

One of the steps in model validation is to assess how variable the parameters are to noise, or in other words, their sensitivity to noise. It is extremely important to evaluate the reliability of the parameter estimates. For a linear regression model, the covariance matrix of the estimated parameter vector is given by the following equation [49]:

$$\mathrm{C}_{\hat{\theta}} = \hat{\sigma}_v^2(\mathrm{M}^T\mathrm{M})^{-1} \tag{2.31}$$

where, $\mathbf{C}_{\hat{\theta}}$ is the covariance matrix of the estimated parameters in $\boldsymbol{\theta}$. $\hat{\sigma}_v^2$ is an unbiased estimate of the noise variance and $\mathbf{M}$ is the regressor matrix.

From Eq. (2.31) it is clear that the covariance of the parameter estimates depends on the noise level and $(\mathbf{M}^T\mathbf{M})^{-1}$. Obtaining the Singular Value Decomposition (SVD) of $(\mathbf{M}^T\mathbf{M})$ we get:

$$\mathbf{M}^T\mathbf{M} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{2.32}$$

where $\mathbf{U}$ is an NxN matrix with orthogonal columns, $\mathbf{S}$ is an NxN diagonal matrix containing the singular values, and $\mathbf{V}$ is an NxN orthogonal matrix. So, the inverse of $(\mathbf{M}^T\mathbf{M})$ could be written as:

$$\begin{aligned}(\mathbf{M}^T\mathbf{M})^{-1} &= (\mathbf{U}\mathbf{S}\mathbf{V}^T)^{-1} \\ &= (\mathbf{V}^T)^{-1}\mathbf{S}^{-1}\mathbf{U}^{-1} \\ &= \mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T \end{aligned} \tag{2.33}$$

From Eq. (2.33) it is clear that $(\mathbf{M}^T\mathbf{M})^{-1}$ depends on $\mathbf{S}^{-1}$. For square, positive semi-definite matrices, the singular values are the same as the eigenvalues, and:

$$S^{-1} = \begin{bmatrix} \frac{1}{\lambda_1} & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{\lambda_2} & 0 & \dots & 0 \\ 0 & 0 & \frac{1}{\lambda_3} & \dots & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{\lambda_n} \end{bmatrix} \tag{2.34}$$

where $\lambda_1....\lambda_n$ are the n eigenvalues of $\mathrm{M}^T\mathrm{M}$. Combining Eq (2.33) and (2.34) it is clear that eigenvalues have an inverse relationship with matrix inverses. Thus, larger eigenvalues lead to a smaller inverses and smaller eigenvalues lead to bigger inverses.

Relating the above with noise sensitivity, the larger ($\mathrm{M}^T\mathrm{M}$) is, the better resistant the model is to noise. Thus, larger eigenvalues are preferred. Also, the size of the largest term in C_θ is determined by the smallest eigenvalue, and adding rows to M reduces the smallest eigenvalue, thereby increasing the size of the largest term in C_θ. This is due to the Interlacing Property which states that [11],

"If $\mathbf{A}_r$ denotes the leading $r-by-r$ principal submatrix of an $n-by-n$ symmetric matrix $\mathbf{A}$, then for $r = 1 : n-1$ the following interlacing property holds":

$$\lambda_{r+1}(\mathbf{A}_{r+1}) \leq \lambda_r(\mathbf{A}_r) \leq \lambda_r(\mathbf{A}_{r+1}) \leq \cdots \leq \lambda_2(\mathbf{A}_{r+1}) \leq \lambda_1(\mathbf{A}_r) \leq \lambda_1(\mathbf{A}_{r+1}) \quad (2.35)$$

where $\mathbf{A}$ represents $\mathrm{M}^T\mathrm{M}$. From Eq.(2.35) it is evident that larger matrices have larger eigenvalues which ultimately lead to smaller inverse matrices thus giving a smaller covariance of the parameter estimates. Adding more rows (ensembles) to M leads to larger eigenvalues and a smaller covariance matrix. Also, adding parameters adds more eigenvalues, with the smallest one getting smaller and the biggest one getting bigger, thereby increasing the critical measure of conditioning (i.e. the ratio between the smallest and the largest eigenvalues). Thus, provided the model structure is capable of representing the system, estimating the smallest possible number of parameters from the longest available data record is likely to lead to the most accurate estimates possible.

2.2.7 Linear Time-Varying System Identification

Most systems in the real-world do not follow the idealized linear time-invariant model. Most real systems are usually non-linear and/or time-varying. There is a great deal of research being conducted in the area of non-linear system identification: constructing models which explicitly include representations of the system's non-linearities [29], [35]. However, a discussion of these techniques is beyond the scope of this thesis.

A common engineering approach for dealing with nonlinear systems is to linearize the non-linearity about a particular operating point. For example, the force exerted by a non-linear spring is a non-linear function of its displacement. The simplest nonlinear spring has force equal to:

$$F(x) = -kx + bx^3 \tag{2.36}$$

where, k and b are spring constants, x is the displacement and F is the force. The simplest nonlinear spring needs to be cubic in order to avoid having an unstable region, which would be caused by a quadratic relationship between the force and displacement. The Taylor expansion about an operating point x_0 for a short distance δ is given by:

$$\begin{aligned} F(x_0+\delta) &= -kx_0 + bx_0^3 + \delta(F'(x_0)) + \frac{\delta^2}{2!}(F''(x_0)) + ... \\ &= -kx_0 + bx_0^3 + \delta(-k + 3bx_0^2) + ... \end{aligned} \tag{2.37}$$

Thus, extremely short displacements about a particular operating point can be modelled as a linear function. The equation also takes into account the changing equi-

librium point for a spring.

The full non-linear system description is often very complicated, and identifying it would require testing the system at a large number of operating points (in order to map the entire non-linearity). However, if the behaviour is needed along a given trajectory, it is often simpler to treat the system as a linear, time-varying system, where the operating point moves along the given trajectory. For example, the relationship between the applied torque and the resulting motion of the human ankle from full plantarflexion to full dorsiflexion is non-linear as a whole [32]. However, it can be linearized by considering only small perturbations throughout the movement.

The other type of linear TV systems are those which genuinely have TV properties and thus have to be modeled using TV structures. The time-variations can be classified according to the magnitude of the timescale in relation to the dominant time-constants of the system's dynamics. Fatigue in neuromuscular systems and fading in wireless channel can be classified as time-variations with a time-scale that is of a similar (perhaps an order or two of magnitude longer) to the relevant dynamics, while aging of components in a control system is an example of time-variation with a relatively long time-scale. Specialized models and techniques are used for faster variations as opposed to the adaptive (but essentially TI) methods that are used for slower variations.

2.2.8 Linear Parameter Varying Systems

In this section we will consider systems that can be represented using linear regression models with TV coefficients. For example the output of a TV-Auto Regressive

Exogenous input (ARX) model is given by:

$$y(t) = \sum_{j=0}^{Q} b(j,t)x(t-j) - \sum_{i=1}^{P} a(i,t)y(t-i) + e(t) \tag{2.38}$$

where $a(i,t)$ and $b(j,t)$ are TV filter coefficients. The index t represents time and the indices Q and P are the maximum degree of the numerator and denominator polynomials, respectively, throughout the time variation. Sometimes the parameters $b(Q,t)$ and $a(P,t)$ may have values of 0, so that the instantaneous degree is less than P(or Q), however the maximum degree remains TI. As in LTI models, the term $e(t)$ is the innovation present in the system. Two state-of-the art techniques for dealing with rapidly varying TV coefficients, temporal basis expansions and ensemble technique, are described in detail in the sections below.

Temporal Basis Expansions for Parametric Models

In temporal basis expansion techniques, each of the TV coefficients is expanded onto a set of TV basis functions, $\pi(t)$, selected by the user [12]. Expanding the TV numerator and denominator coefficients in (2.38) onto a set of basis functions:

$$\begin{aligned} a(i,t) &= \sum_{k=0}^{V} \alpha(i,k)\pi_k(t), \\ b(j,t) &= \sum_{k=0}^{V} \beta(j,k)\pi_k(t) \end{aligned} \tag{2.39}$$

where $\alpha(i,k)$ and $\beta(j,k)$ are the expansion coefficients and $\pi_k(t)$ are the basis functions. Substituting (2.39) into (2.38),

$$\begin{aligned} y(t) \;=\; & \sum_{j=0}^{Q}\sum_{k=0}^{V} \beta(j,k)x_k(t-j) \\ & - \sum_{i=1}^{P}\sum_{k=0}^{V} \alpha(i,k)y_k(t-i) + e(t) \end{aligned} \tag{2.40}$$

where,

$$\begin{aligned} y_k(t-i) &= \pi_k(t)y(t-i), \\ x_k(t-j) &= \pi_k(t)x(t-j) \end{aligned} \tag{2.41}$$

Comparing (2.40) and (2.12) as a means of setting up the Least Squares solution, the regressor matrix M contains all the candidate vectors $\mathbf{y}_0(t-1)...\mathbf{y}_V(t-P), \mathbf{x}_0(t)...\mathbf{x}_V(t-Q)$ arranged in the following way:

$$\begin{aligned} M = [\ \mathbf{y}_0(t-1)\ &,....\mathbf{y}_V(t-1), \mathbf{x}_0(t), ...\mathbf{x}_V(t) \\ \mathbf{y}_0(t-2)\ &,...\mathbf{y}_V(t-2), ..., \mathbf{x}_0(t-1), ...\mathbf{x}_V(t-1), ...] \end{aligned} \tag{2.42}$$

Once the π_k have been chosen, x_k and y_k are computed using eq. (2.41). The regressor matrix M can be constructed using (2.42). Using M and the output vector $\mathbf{y}$, a linear regression can be solved to determine the unknown parameters α and β. Once, the α and β values have been determined, the values are substituted back into Eq. (2.39) to determine the original time varying parameters $a(i,t)$ and $b(j,t)$.

Non-parametric temporal basis expansions

The TV counterpart of (2.22) is given by:

$$y(t) = \sum_{\tau=0}^{T} g(t,\tau)x(t-\tau) \tag{2.43}$$

where $g(t,\tau)$ is the time-dependent IRF and $x(t)$ is the input. Fig. 2.7 shows an example of a TV IRF. The IRFs last 0.2 seconds and are shown evolving over a period of 2 seconds.

The IRFs in the figure represent the simulated intrinsic compliance of a human ankle. The ankle was moved from full plantarflexion to full dorsiflexion over a period

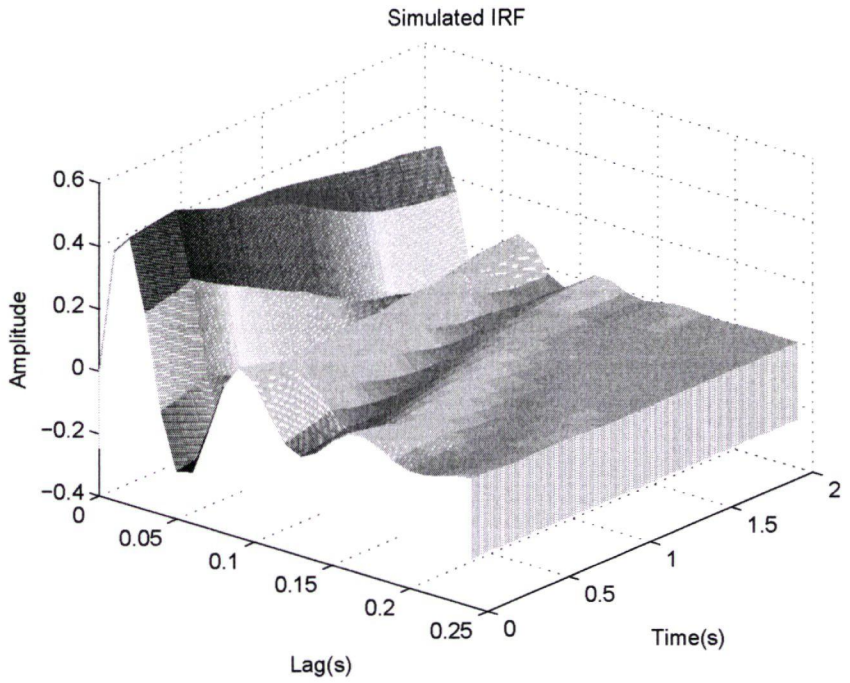

Figure 2.7: Time-Varying Impulse Response Functions

of 2s. The 2s movement was divided into 200 equally spaced positions, each corresponding to a given sampling time [32]. The data were sampled at a rate of 200 Hz. The compliance IRFs have been used to illustrate TV IRFs here and the simulation is described in greater detail in section 3.2.

Again, expanding the TV IRF onto a set of basis functions we get:

$$g(t,\tau) = \sum_{k=0}^{K} \alpha(k,\tau)\pi_k(t) \tag{2.44}$$

where $\alpha(k,\tau)$ are TI parameters and $\pi_k(t)$ are the basis functions. Substituting (2.44) into (2.43), we get

$$y(t) = \sum_{\tau=0}^{T}\sum_{k=0}^{K} \alpha(k,\tau)\pi_k(t)x(t-\tau) \tag{2.45}$$

Now, we can define new variables such that:

$$x_k(t-\tau) = \pi_k(t)x(t-\tau) \tag{2.46}$$

and by combining (2.45) and (2.46), we get

$$y(t) = \sum_{\tau=0}^{T}\sum_{k=0}^{K} \alpha(k,\tau)x_k(t-\tau) \tag{2.47}$$

From (2.40) and (2.47) it is evident that the time variation is contained in the basis functions. As a result, the identification problem has been transformed into the estimation of a set of (TI) expansion coefficients, which can be accomplished by solving a linear regression (2.13). The regressor matrix in this case would contain delayed versions of the input vector multiplied by the basis functions.

$$M_t = [\mathbf{x}_0(t)...\mathbf{x}_K(t), \mathbf{x}_0(t-1)..., \mathbf{x}_0(t-2)...] \tag{2.48}$$

where

$$\begin{aligned} x_k(t-\tau) &= \pi_k(t)x(t-\tau) \\ &= [\pi_k(1)x(1-\tau), \pi_k(2)x(2-\tau), ..., \pi_k(N)x(N-\tau)]^T \end{aligned} \quad (2.49)$$

Appropriate Choice of Basis Functions

One of the disadvantages of the basis expansion method is that the basis functions have to be chosen prior to the identification and have to model the time variation appropriately. Different basis functions show their own application dependant tractability and accuracy [53]. Thus, the successful application of this approach depends on the choice of an appropriate basis function. The basis functions need not be orthogonal or orthonormal but must be linearly independent of each other. Smooth variations can be modeled using orthogonal polynomials such as Tchebychev or Legendre polynomials, periodic variations can be expanded onto sinusoids and sharp variations can be modeled with Walsh functions or Haar wavelets. The first 4 Tchebychev polynomials and Walsh functions are shown in Figures 2.8 and 2.9, respectively

Term-selection techniques, discussed in the next section, can be used to evaluate the suitability of a proposed expansion basis.

Term-Selection

A major disadvantage with temporal basis expansions is the large increase in the number of parameters that must be estimated, and hence the resulting increase in the variance of those estimates. Specifically, if V is the number of elements in the temporal expansion basis, and $P+Q+1$ is the number of filter coefficients, a total of $d = V(P+Q+1)$ parameters must be estimated. This can be mitigated by using some

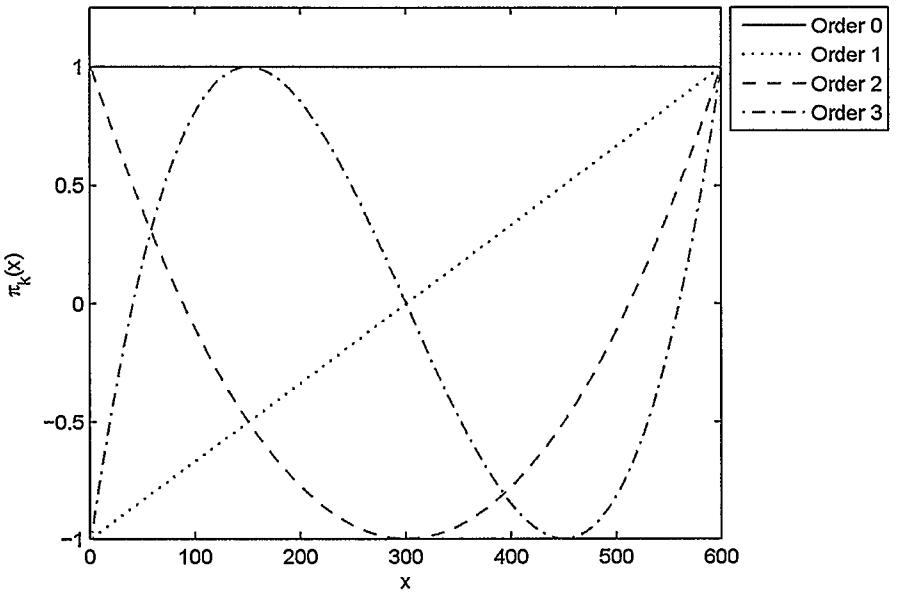

Figure 2.8: Tchebychev Polynomials of degree 0,1,2,3 ove the domain [0 600]

form of term selection to eliminate unnecessary regressors, which do not significantly contribute to the predictor output, and therefore reduce the dimensionality of the final estimation problem as much as possible. The term selection problem comes up in NARMAX identification as well [29], since NARMAX models are (pseudo) linear regression models and term selection techniques are applicable to all linear regression models. A great deal of the term-selection research has been done in this area, since similar overparameterization problems arise [23], [25], [44]. Several algorithms have been developed to address the term-selection problem. These include techniques based on orthogonal forward regression searches [24], [35], which are discussed in detail below. Other recently developed orthogonal searches include the Optimal Parameter Search (OPS) [53]. These algorithms all perform local searches, and,

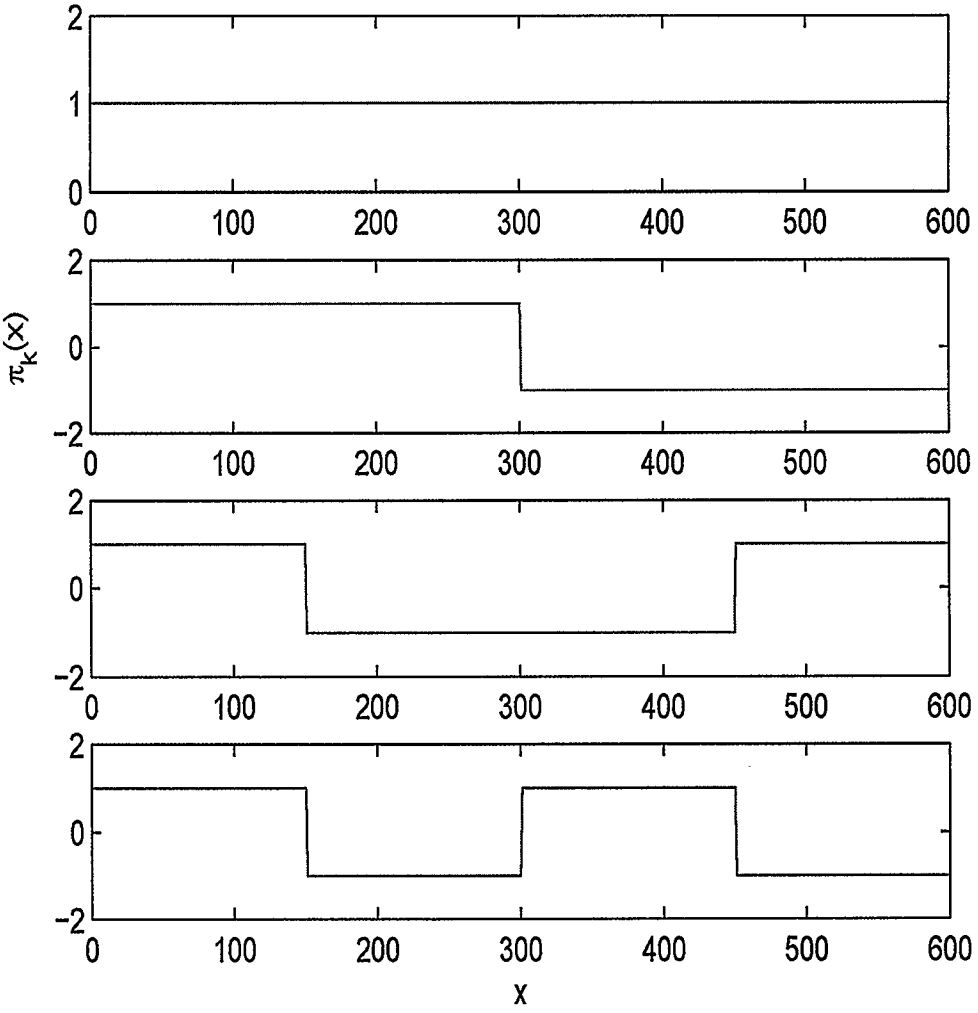

Figure 2.9: First four Walsh Functions

therefore cannot be guaranteed to obtain the sparsest possible representation of the model. Other more global approaches to the term-selection problem include genetic algorithms [35] and significance testing based on boostrap resampling [29]. These methods though theoretically elegant require excessive computational time.

Orthogonal Regression Searches

Orthogonal regression searches rely on orthogonalizing the regressor matrix and using the orthogonalized relationship to compute how much each term will reduce the total mean-squared error [27]. The regressor matrix $\mathbf{M}$ is decomposed into $\mathbf{H} = \mathbf{M}^T\mathbf{M}$ and $\boldsymbol{\beta} = \mathbf{M}^T\mathbf{y}$. The ordered columns and rows of $\mathbf{H}$ are placed in new orthogonalized matrices $\mathbf{C}$ and $\mathbf{R}$ respectively.

An error reduction ratio, Err is defined as the square $\boldsymbol{\beta}$ divided by the diagonal of $\mathbf{H}$

$$Err = \frac{\boldsymbol{\beta}^2}{diag(\mathbf{H})} \tag{2.50}$$

A row and column of $\mathbf{H}$ is selected which gives the best reduction in error as the first column and row to include in new orthogonal matrices $\mathbf{C}$ and $\mathbf{R}$. Elements of $\boldsymbol{\beta}$ are swapped as well. The rest of the rows and columns in $\mathbf{H}$ are orthogonalized w.r.t. to the first row and column. This process is repeated until no more columns contribute a greater reduction in the MSE than a pre-set tolerance, ρ.

Ensemble Techniques

Ensemble techniques produce very accurate models of TV systems without any *a priori* knowledge of their time variations, but as their name suggests, they require an ensemble of responses each of which contains the same time-varying behavior [32].

They identify the dynamics at each point in time from a series of m input-output realizations; $x(t,k)$ and $y(t,k)$, k = 1...m, where t is the time since the beginning of the k^{th} realization of the time variation. Thus, for the TV IRF model in (2.43),

$$y(t,k) = \sum_{j=0}^{\infty} h(j,t)x(t-j,k) + e(t,k) \tag{2.51}$$

At each instant, t, in the time-variation, there are m separate equations, corresponding to $k = 1...m$. For TV FIR and ARX models respectively, for each instant of time, t, (2.43) or (2.38) are collected for all responses and rewritten in matrix notation as

$$\mathbf{Y_t} = \mathbf{X_t}\mathbf{H_t} \tag{2.52}$$

where $\mathbf{Y_t}$ is a vector of length m with kth element $y(t,k)$,

$$\begin{bmatrix} y(t,1) \\ y(t,2) \\ y(t,3) \\ \vdots \\ y(t,m) \end{bmatrix} \tag{2.53}$$

$\mathbf{H_t}$ is a vector of length Q with j^{th} element $h(t,j-1)$,

$$\begin{bmatrix} h(t,0) \\ h(t,1) \\ h(t,3) \\ \vdots \\ h(t,Q) \end{bmatrix} \tag{2.54}$$

and $\mathbf{X_t}$ is a m x q matrix with k,j^{th} element $x(k,t-j+1)$

$$\boldsymbol{X}_t = \begin{bmatrix} x_t(1,1) & \dots & x_t(1,q) \\ x_t(2,1) & \dots & x_t(2,q) \\ \vdots & \vdots & \vdots \\ x_t(m,1) & \dots & x_t(m,q) \end{bmatrix} \tag{2.55}$$

In practice, Eq.(2.52) is solved for each time point, t. Conceptually, the equations can be grouped into a single relationship and collected into a single matrix that looks like:

$$\begin{bmatrix} \mathbf{Y}_{t_1} \\ \mathbf{Y}_{t_2} \\ \vdots \\ \mathbf{Y}_T \end{bmatrix} = \begin{bmatrix} \mathbf{X}_{t_1} & 0 & \dots & 0 \\ 0 & \mathbf{X}_{t_2} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \mathbf{X}_T \end{bmatrix} \begin{bmatrix} \mathbf{H}_{t_1} \\ \mathbf{H}_{t_2} \\ \vdots \\ \mathbf{H}_T \end{bmatrix} \tag{2.56}$$

Equation (2.52) is solved by linear regression for $t = 1...T$. For (2.52) and hence (2.56) to have a solution, $\boldsymbol{X}_t$ from (2.55) must have more rows than columns. In this method the input-output data are chosen from the same point in the system's time-variation across an ensemble of responses, rather than over the time course of a single response [34]. Thus, at least one response is required for each point in the IRF.

A big advantage of this method is that additive noise can be averaged out by using multiple ensembles. Also, no prior assumptions are required about the TV basis functions. However, the practical use of this method is limited because in many applications it is very difficult to consistently repeat experiments enough times to obtain the necessary sets of input-output data. Ensemble techniques have been

used in biomedical applications such as tracking dynamic ankle stiffness during a rapid, voluntary, isometric contraction [34]. The ensemble technique has also been extended to identify parallel cascade models of ankle stiffness [2].

Chapter 3

The Combined Ensemble Basis Expansion Algorithm

This chapter includes a description of the development and testing of a new algorithm for time-varying system identification which combines the two techniques, Basis Expansion and Ensemble algorithms, described in Chapter 2. The combined Ensemble/Basis Expansion (EBE) algorithm is derived, and its performance is demonstrated using data from a simulation of the time-varying compliance dynamics of the human ankle, as the ankle moves from full dorsiflexion to full plantarflexion [1].

3.1 Theory

Two recently developed techniques for linear, time-varying system identification were reviewed in Chapter 2, each of which had a potentially serious drawback. With basis expansions a large number of parameters must be estimated. With the ensemble technique more parameters have to be estimated than with basis expansions (a set of parameters is estimated for every time-point) and to do this a large number of data-sets must be obtained. The biggest drawback with the ensemble technique is that its extremely difficult to consistantly repeat experiments enough times to obtain the necessary sets of input-output data. Thus, the two methods, Basis Expansions

[1] An abbreviated version of this chapter was presented at IEEE Canadian Conference on Electrical and Computer Engineering, 2005

and the Ensemble Technique, were combined for this new method of system identification. In this new method, there is an ensemble of input/ouptut records where the system undergoes the same time variation in each realization (just as in the ensemble technique). However, instead of estimating a full set of parameters at each point in the time variation, the TV parameters are projected onto a set of basis functions, thereby reducing the number of parameters to be estimated. The EBE algorithm reduces the number of parameters that must be estimated, as compared to the ensemble method [34], and therefore reduces the number of datasets necessary to achieve a given noise performance. The EBE algorithm uses the signal averaging properties of the ensemble algorithm to reduce its noise sensitivity, as compared to the temporal basis expansion technique [53] using an indentical expansion basis. The EBE algorithm reduces the severity of the disadvantages of each of the individual techniques.

Since an ensemble of datasets will be required, let u(k,t) and y(k,t) be the input and output at time t in the k^{th} realization. The output y(k,t) can be written as:

$$y(k,t) = \sum_{j=0}^{Q}\sum_{l=0}^{V}\beta(j,l)u_l(k,t-j) - \sum_{i=0}^{Q}\sum_{l=0}^{V}\alpha(i,l)y_l(k,t-i) \tag{3.1}$$

where,

$$\begin{aligned} u_l(k,t-j) &= \pi_l(n)u(k,t-j) \\ y_l(k,t-i) &= \pi_l(n)y(k,t-i) \end{aligned} \tag{3.2}$$

where $\beta(j,l)$ and $\alpha(i,l)$ are the TI parameter with j,i representing the numerator and denominator orders respectively and l representing the order of the basis functions with a maximum of V parameters. From Eq.(3.1) it is clear that the basis elements

l do not depend on the ensemble number k. Thus, we have:

$$\mathbf{y}_1 = \begin{bmatrix} y_0(1,n-1) \\ y_0(1,n-2) \\ \vdots \\ y_1(1,n-1) \\ y_1(1,n-2) \\ \vdots \\ y_V(1,n-1) \\ y_V(1,n-2) \\ \vdots \end{bmatrix} \tag{3.3}$$

Similarly, M can be written as:

$$\mathbf{M}_1 = \begin{bmatrix} u_0(1,n) & 0 & 0 & \cdots & 0 & 0 & \cdots \\ u_0(1,n-1) & u_0(1,n) & 0 & \cdots & y_0(1,n-1) & 0 & \cdots \\ u_0(1,n-2) & u_0(1,n-1) & u_0(1,n) & \cdots & y_0(1,n-2) & y_0(1,n-1) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{3.4}$$

Since the true expansion coefficients are the same in each realization, the basis vector is repeated the same number of times as the number of ensembles. The y and M matrices are also stacked with multiple ensembles. Thus we are solving the linear regression in equation (1.4), but using the following matrices:

$$\begin{aligned} \mathbf{y} &= [\mathbf{y}_1^T \mathbf{y}_2^T \ldots\ldots \mathbf{y}_k^T]^T, \\ \mathbf{M} &= [\mathbf{M}_1^T \mathbf{M}_2^T \ldots\ldots \mathbf{M}_k^T]^T \end{aligned} \tag{3.5}$$

where $\mathbf{y}_k$ and $\mathbf{M}_k$ are the input and output matrices from the k^{th} ensemble, as defined in Eqs. (3.4) and (3.5). $\boldsymbol{\theta}$, however, contains the same number of elements as in (1.4). Combining the two methods helps eliminate some disadvantages of each of the two methods. Using the ensemble method reduces the effective noise level and hence the noise sensitivity of the parameters by signal averaging. On the other hand, the use of basis expansions reduces the number of parameters needed to represent the time-varying system, and thereby reduces the number of ensembles required for identification. In the ensemble technique the number of input/output records must at least equal the number of model parameters at one time instant. However, the number of ensembles required for the combined technique is drastically reduced since the time-variation is contained in the basis functions.

The last step of the algorithm is using a term-selection technique, which is identical to the ordinary basis expansion method (except in this case we have reduced noise sensitivity due to the use of the ensembles). Term-selection techniques were discussed in section 2.2.8.

3.2 Simulation

The EBE technique was tested on simulated TV data that represents the intrinsic compliance of the ankle joint as its position varies from full plantarflexion to full dorsiflexion over a period of 2s [32]. Intrinsic joint compliance is defined as the dynamic relationship between torque arising from intrinsic properties of joint and muscle and the joint angular position [32]. For small displacements about a stationary operating point, ankle dynamics can be modeled by a linear, second-order low pass filter with

transfer function (TF):

$$\frac{\theta(s)}{T(s)} = \frac{1}{Is^2 + Bs + K} \tag{3.6}$$

where $\theta(s)$ is the Laplace transform of the joint's angular position, $T(s)$ is the torque produced about the joint, and I,B and K are the position-dependant inertia, viscosity, and elasticity parameters respectively.

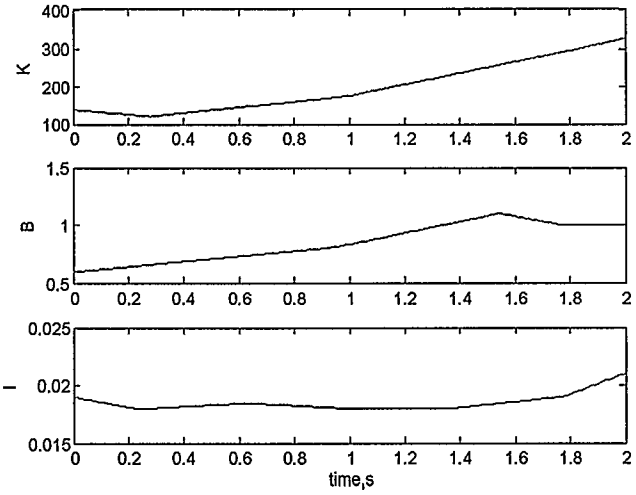

Figure 3.1: Time-Varying parameters I,B,K - Parameters are actually functions of joint angle which is shown varying from full plantarflexion to full dorsiflexion over a period of 2 seconds

The model parameters I, B, K have been shown to depend on the average ankle angle and background contraction level [16], [18], [19], [45]. However, the LTI model in Eq.(3.6) remains valid for minor displacements about a fixed operating point, provided the corresponding parameter values are used. For the simulation, the values of the I, B, K parameters and their dependance on the ankle position, θ were determined from previously published values [32]. The ankle was simulated moving from full plantarflexion to full dorsiflexion over a 2s period. Thus, the I, B, K parameters

were effectively time-dependant as shown in Figure 3.1.

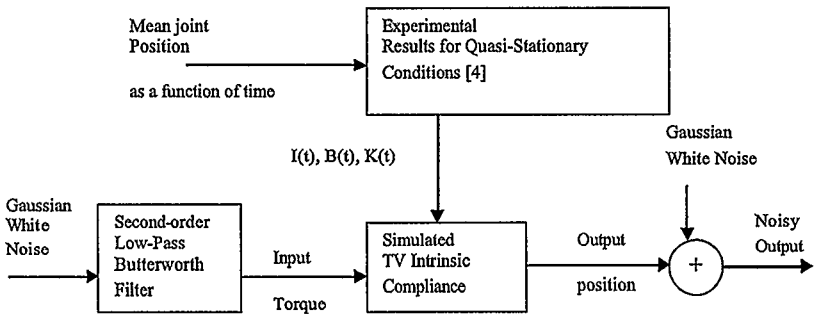

Figure 3.2: Block diagram for ankle dynamics simulation

The simulation was performed by dividing the ankle movement into 200 equally spaced positions, each corresponding to a given sampling time during the 2s movement. The input perturbation for the simulation was generated by filtering Gaussian white noise with a second-order low-pass Butterworth filter with cut-off frequency of 25 Hz. Since Gaussian white noise has a flat frequency spectrum, the maximally flat pass-band of a Butterworth filter makes it an ideal choice for this application. This was done to simulate the capabilities of the experimental apparatus described in [32]. At each sampling time, a time-invariant transfer function was calculated. The process is illustrated in Figure 3.2. For simulation purposes, the TI continuous time TFs corresponding to each point in the 2s simulation, were discretized using a Zero Order Hold and the numerators and denominators were extracted from the discretized system. The numerator and denominator were then used to form an ARX model, which was then simulated using eq. (2.38). The ARX model was used to generate the output. The true TV IRFs are obtained by calculating the impulse

response of the TFs from the ARX models and are shown in Fig. 3.3. The input and output were then used to form the regressor matrix **M**. Three different simulations were performed: one with noise-free data, the second with 7 dB of additive noise and the third with 12 dB of additive noise. For the noise-free data, identification was performed using just the basis expansion technique, where the expansion basis consisted of the first four Tchebychev polynomials shown in Fig 2.8. These were chosen because of their smooth time-variations spread evenly over the entire time-scale. For the noisy data, identification was performed using just the basis expansion and then a combination of 6 ensembles and basis expansions using the same set of Tchebychev basis functions. Once the time-varying parameters had been expanded onto the basis functions the final coefficient estimates were obtained using Least Squares and the FOS [24] was used for term-selection for all three cases.

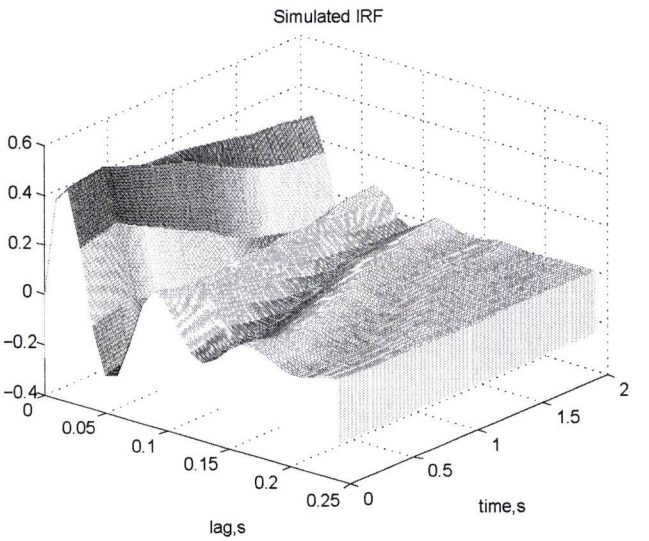

Figure 3.3: Simulated TV IRFs

Figure 3.3 shows TV IRFs. The x-axis is the lag, while the y-axis represents

time. Each slice parallel to the lag axis corresponds to an IRF at a particular point in time.

The results were validated using the Variance Accounted For (VAF) which was described in eq. (2.30). For this case the VAF was computed for the impulse responses, at each point in time as follows:

$$VAF_{irf}(t) = 100\left(1 - \frac{\sum_{\tau=0}^{T}(h(t,\tau) - \hat{h}(t,\tau))^2}{\sum_{\tau=0}^{T} h^2(t,\tau)}\right) \tag{3.7}$$

where $h(t,\tau)$ is the true simulated IRF and $\hat{h}(t,\tau)$ is the estimated IRF and the VAF is calculated across all lag τ, for each instant in time.

3.3 Results: Basis Expansion Algorithm

The model obtained from the first simulation, using the noise-free data and a single-trial, was accurate. It matched the simulated IRFs closely with an average VAF of 99.9%. This result validates the use of Tchebychev polynomials as the temporal basis expansion funcion. Thus, the model is capable of representing system dynamics, so that any errors in the following simulations are due to effects of noise and not due to undermodelling.

The TV IRFs estimated from a single-trial on data with 7 dB of additive noise are shown in Fig. 3.4. Significant errors are evident at the beginning and end of the data records. The magnitude of these errors masks the behaviour at all other times in the ensembles. Figure 3.5 shows the same estimated time-varying IRFs from 1s to 1.5s.

As we can see the middle section matches up very well with the actual IRFs. The accuracy of the identified IRFs shown in Figure 3.4, further corroborates this. The

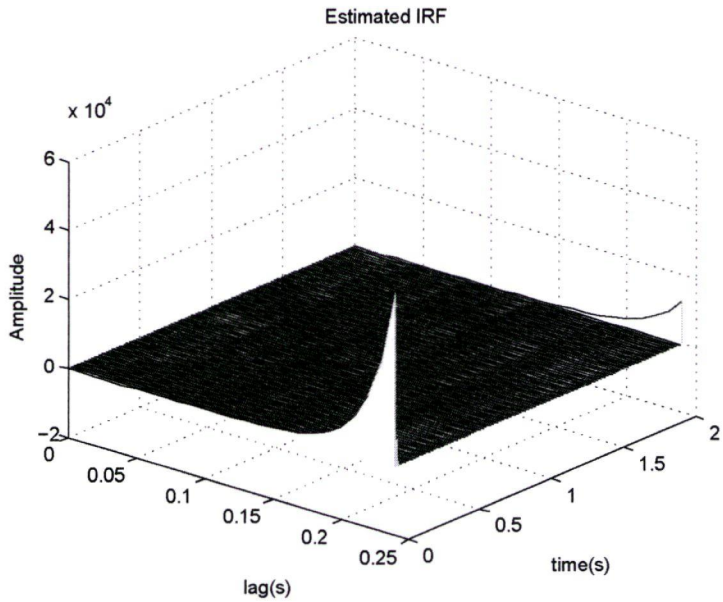

Figure 3.4: Estimated IRFs for SNR = 7dB, Single Trial

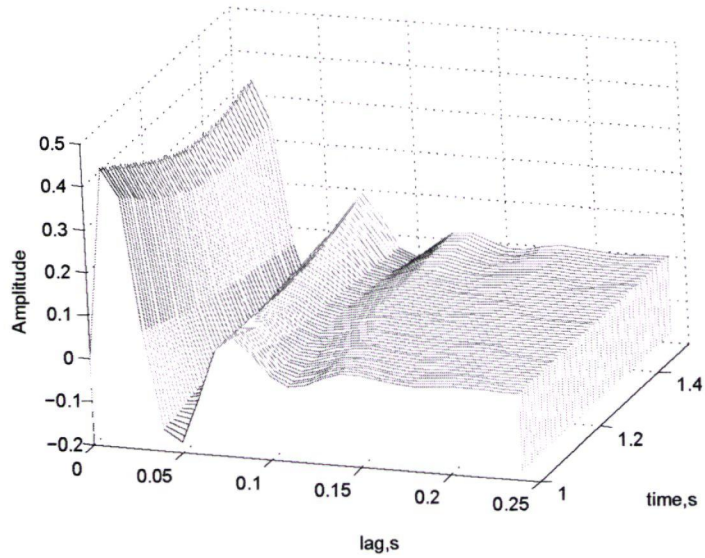

Figure 3.5: Estimated IRFs for SNR = 7dB, Single Trial, t = 1-1.5s

average VAF between the true IRFs and the single-trial IRFs shown in Fig. 3.6 is $-3.62 \times 10^{9}\%$, ranging from 99% to $-5.66 \times 10^{11}\%$.

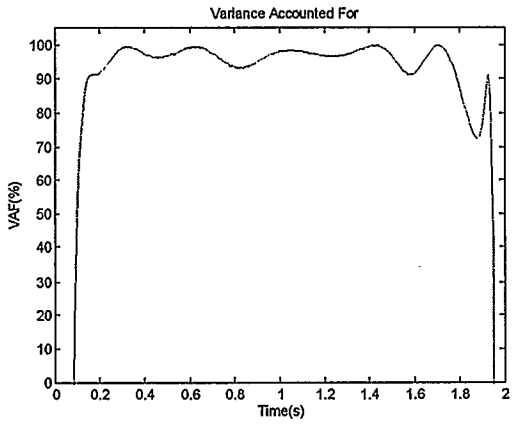

Figure 3.6: Variance Accounted For, 7dB single trial model

The second trial was performed at 12dB SNR and somewhat better results were obtained. Fig 3.7 shows the IRFs obtained using the basis expansion algorithm with 12 dB of SNR.

Again, zooming in on the figure and displaying the IRFs from 1s to 1.5s, it is evident that it is only the extremities that are estimated inaccurately.

The average VAF for this particular trial is 72.6%, ranging from 99% to -1986%. Model accuracy is plotted as a function of time and is shown in Fig. 3.9 and again it is clear that the extremities are estimated extremely inaccurately whereas the rest of the IRF estimates are quite accurate. Comparing Fig. 3.6 and Fig. 3.9 we can see that the accuracy of the 12dB model is far less variable than that of the model identified at 7dB SNR.

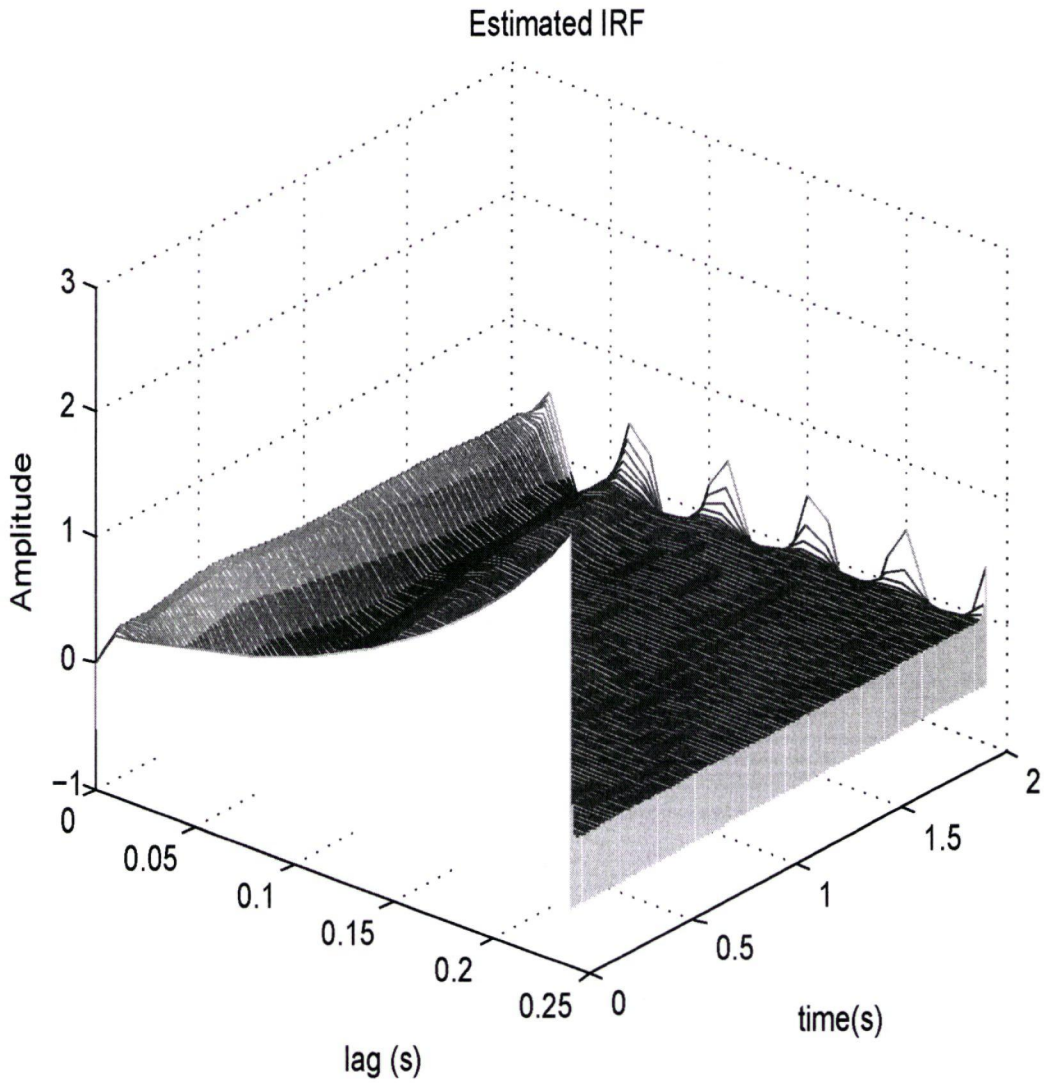

Figure 3.7: Estimated IRFs for SNR = 12dB, Single Trial

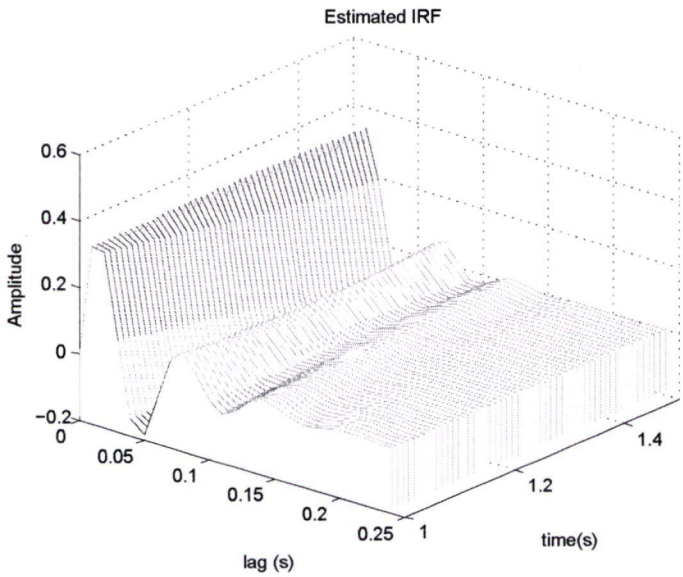

Figure 3.8: Estimated IRFs for SNR = 12dB, Single Trial, t = 1-1.5s

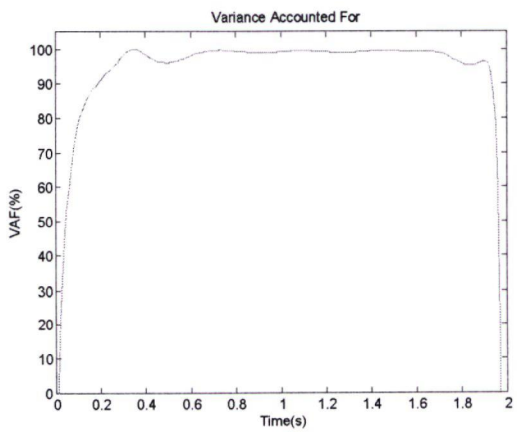

Figure 3.9: Variance Accounted For, 12dB single trial model

3.4 Results: Ensemble Basis Expansion Algorithm

The ensemble basis expansion technique was applied to noisy data obtained under the same conditions: 7 dB SNR. The EBE used 6 times as much data (i.e. 6 trials), as compared to the traditional BE algorithm.

At 7 dB SNR, the estimated IRFs using the EBE algorithm is shown in Fig. 3.10. It is clear from this figure that the combined technique provides much better estimates of the IRFs, particularly at the beginning and ends of the records. The average VAF is 94.3%, ranging from 82.2% to 99.9% which supports that this is a much better model. Fig.3.11 shows the VAF vs time obtained from this method. Again comparing this figure to the two previous VAFs vs time Fig. 3.6 and 3.9, that the combined technique provides a much more accurate model estimate.

3.5 Statistics of the Estimates

Monte-Carlo simulations (100 trials) were performed to estimate the means and variances of the IRFs estimated by both methods at an SNR of 7dB. Figure 3.12 shows the standard deviations of the IRFs obtained from using a single trial at 7dB of SNR from time, t = 0.2s to 1.94s. The standard deviation is plotted in a 3-dimensional graph against Lag and Time. Again, we see that at the extremities the standard deviation starts to vary quite a bit.

Fig. 3.14 shows the same 3 dimensional standard-deviation graph from time, t = 0.2s to 1.94s, except this was obtained from using the combined technique at the same SNR of 7dB. The differences between the two graphs show that the ensemble technique is indeed more robust than just using temporal basis expansions.

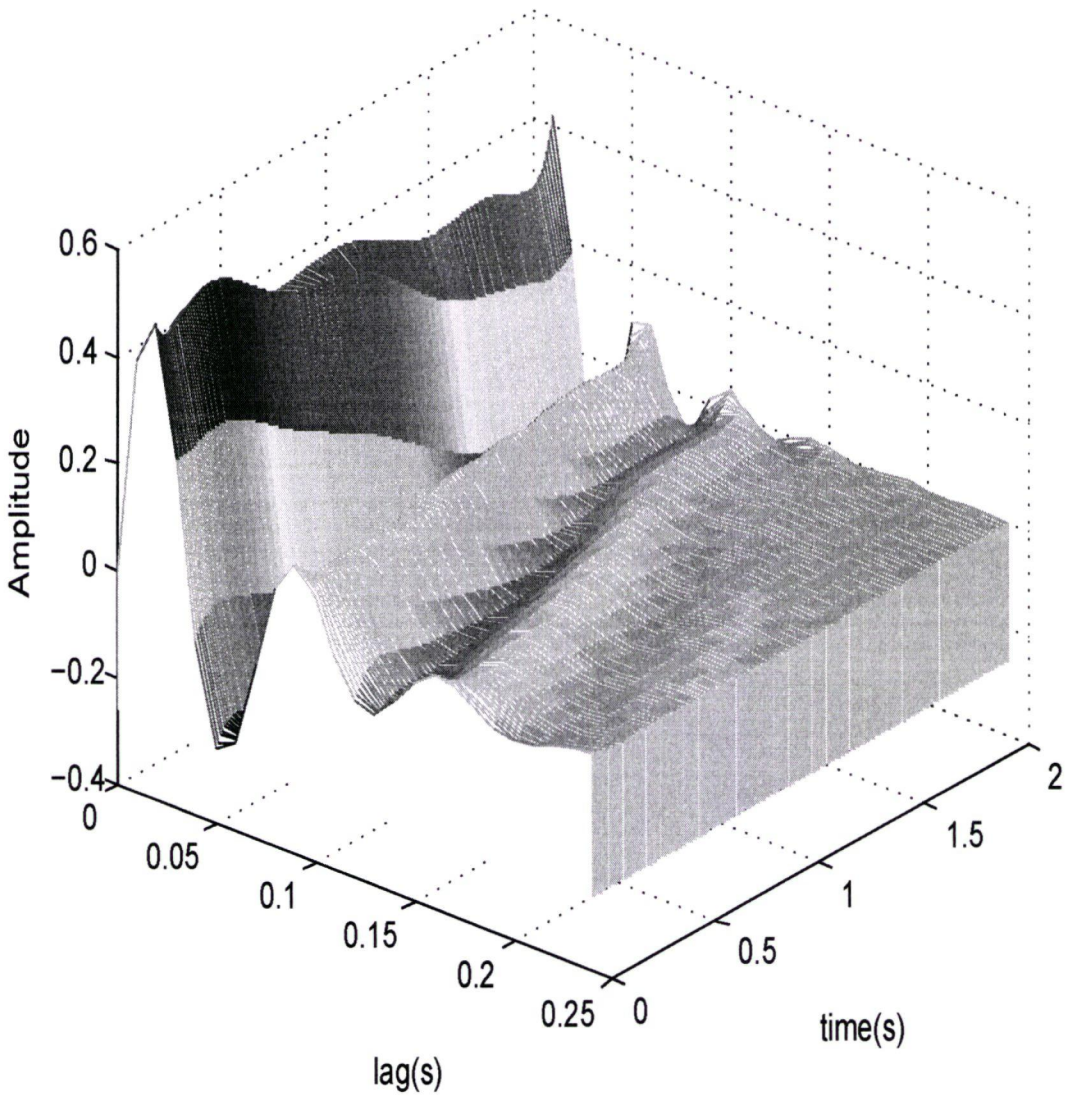

Figure 3.10: Estimated IRFs,6 ensembles (7dB noise)

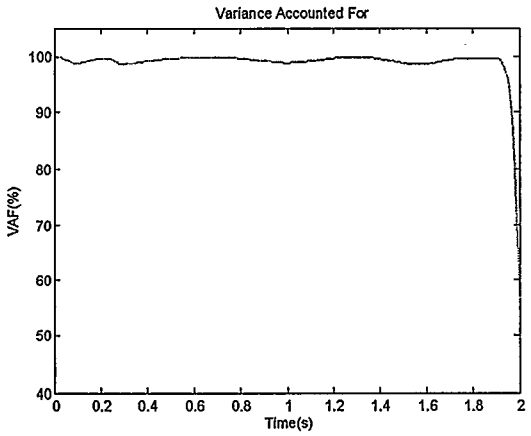

Figure 3.11: Variance Accounted For (VAF) between IRFs plotted as a function of time, SNR = 7 db EBE algorithm, 6 ensembles

The standard deviations on this graphs have much lower values and the standard deviations do not start varying much towards the extremities. This clearly indicates that there is much less deviation in parameter estimates when using the combined techniques as opposed to just using temporal basis expansions.

Fig 3.14 shows a comparison of the standard deviations obtained from the two methods at the same point of time (t = 0.1s).

In this figure the average IRF is plotted in solid for all lags and time t = 0.1s. The dotted lines show the mean plus and minus one standard deviation (SD) of the IRF using the combined method, while the dashed lines show the same bounds but using just the basis expansions. Again, it is obvious that the combined method provides much less variability in the estimates, proving that it is a better method than just the temporal basis expansion technique for this particular application. The results also strongly suggest that this method would provide superior results if used for identification of other linear, time-varying systems.

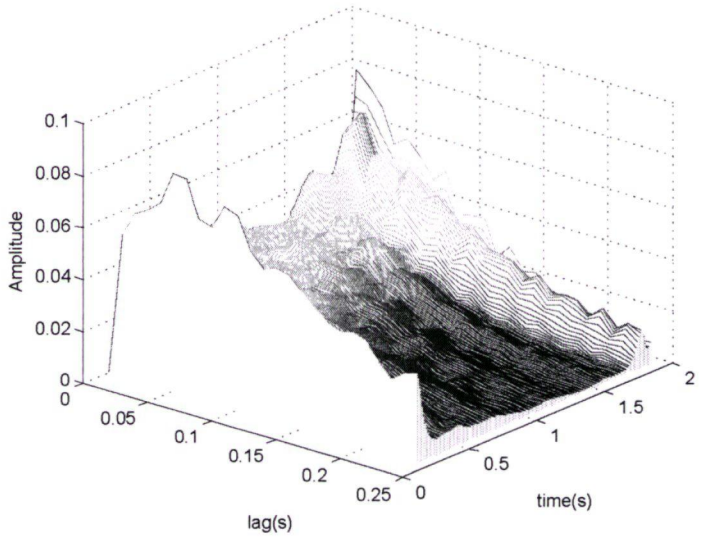

Figure 3.12: 3D Standard Deviation for Single Trial t = 0.2 - 1.94s, SNR 7dB, estimated from 100 Monte-Carlo simulations

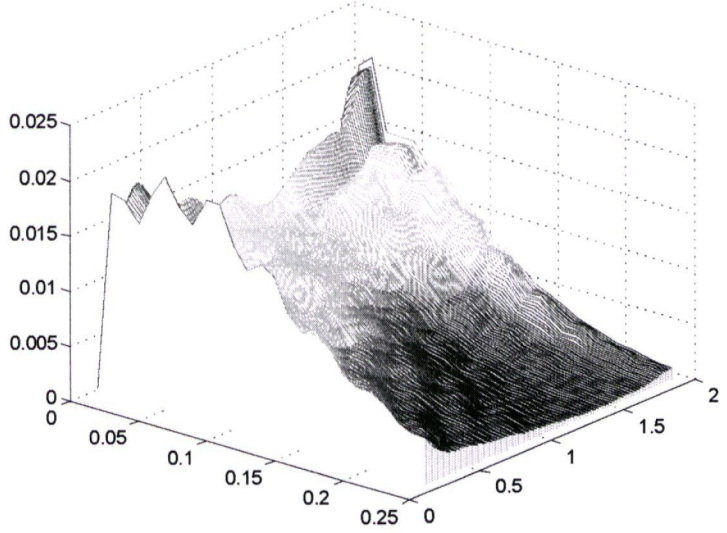

Figure 3.13: 3D Standard Deviation for Combined Technique t = 0.2 - 1.94s, SNR 7dB, estimated from 100 Monte-Carlo simulations

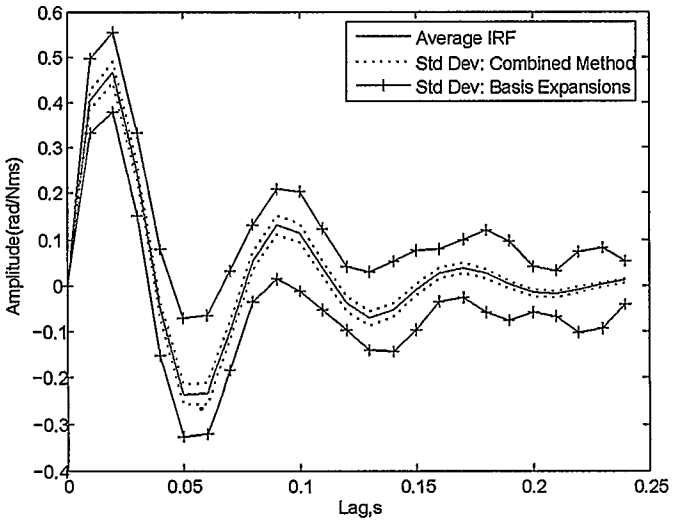

Figure 3.14: Average IRF and SD for the combined and basis expansions methods (t = 0.1s)

Chapter 4

Temporal Basis Expansions using an Optimal Term Selection Strategy

One of the biggest problems with the temporal basis expansion algorithm introduced in Chapter 2 is the large number of potential model terms created by the expansion. While the ensemble approach described in Chapter 2 and the EBE technique described in Chapter 3 can reduce the effective noise level, they do nothing to address the overparameterization. Robust identification of the model requires pruning the set of terms to produce a minimal set of significant terms, that can be estimated with far greater reliability than was possible before the excess terms had been removed. Since the term-selection process reduces the number of columns in the regression matrix, it should also result in a decrease in the variability of the parameter estimates, as a result of the mechanisms described in Section 2.2.6. This chapter provides a detailed description of a recently proposed term-selection technique, called the Least Absolute Shrinkage and Selection Operator (LASSO) [43], that was originally developed for linear-regression models. The LASSO term-selection technique constructs a model structure with a nearly minimal number of non-zero terms, deleting as many terms as possible without eliminating any significant terms, and hence having parameters with low variance and low bias. This term-selection technique was adapted for use in identifying linear, time-varying models [1].

[1] An abbreviated version of this chapter was presented at the IEEE Engineering in Medicine and Biology Conference, 2005, [40]

The balance of this chapter is organized as follows. In Section 4.1 the theory behind the LASSO term selection technique is reviewed, and the technique is adapted for use in with TV LTI models. The LASSO based term-selection is proposed as an alternative to currently available techniques including Fast Orthogonal Search (FOS) [24], the Optimal Parameter Search (OPS) [33], [53], and the Bootstrap technique [29]. These existing term selection techniques are described in Section 2.3.1. Section 4.2 outlines the actual algorithm in detail. The algorithm is demonstrated in detecting changes in the dynamic stiffness of the human elbow immediately following the onset of a broadband perturbation, which is presented in Section 4.4, 4.5 and 4.6.

4.1 Theory

LASSO is a term-selection, and in this context a model-structure selection, technique. Given a linear regression, the LASSO attempts to construct the sparsest possible solution, i.e. that with the fewest non-zero elements [43]. Kukreja et al. [30] have used LASSO for term-selection in the identification of parametric non-linear time-invariant models, a similar application in which a relatively small number of significant model terms must be extracted from an initial model structure that includes a large number of mostly insignificant terms. LASSO has also been used for other applications including PET imaging and analysis of cerebral blood flow [13], [26].

LASSO was developed for selecting terms in a linear regression in standard form such as $\mathbf{y} = \mathbf{M}\boldsymbol{\theta}$. The columns of $\mathbf{M}$ contain delayed versions of the input and output multiplied by basis functions because of the specific application of LASSO in this

case i.e. identifying linear, TV systems, and must be normalized before the LASSO can be used. Once that has been done, the sum of squared errors is minimized

$$\min_{\theta} \frac{1}{2} \|\mathbf{y} - \mathbf{M}\boldsymbol{\theta}\|_2^2 \text{ subject to } \sum_i |\theta_i| < T \tag{4.1}$$

which is equivalent to minimizing

$$\min_{\theta} \frac{1}{2} \|\mathbf{y} - \mathbf{M}\boldsymbol{\theta}\|_2^2 + \lambda \|\boldsymbol{\theta}\|_1 \tag{4.2}$$

where the relaxation parameter λ is essentially a Lagrange multiplier associated with the constraint in (4.1). This is also the cost function that is minimized by a similar method called Basis Pursuit Denoising (BPDN) [5]. While the term-selection problem is a non-convex combinatorial problem, the minimization in (4.2) can be transformed into a quadratic programming (QP) problem with linear constraints. To accomplish this define $\boldsymbol{\theta}_+$ to be a vector containing the positive elements of $\boldsymbol{\theta}$, with zeros used as place-holders for the negative elements. Thus,

$$\theta_{+,i} = \max(\theta_i, 0) \tag{4.3}$$

and let $\boldsymbol{\theta}_-$ contain elements

$$\theta_{-,i} = \max(-\theta_i, 0) \tag{4.4}$$

Thus, all the coefficients in the vector

$$\boldsymbol{\psi} = \begin{bmatrix} \boldsymbol{\theta}_+ \\ \boldsymbol{\theta}_- \end{bmatrix} \tag{4.5}$$

are non-negative. By constructing the regression matrix,

$$\mathbf{A} = \begin{bmatrix} \mathbf{M} & -\mathbf{M} \end{bmatrix} \tag{4.6}$$

the minimization in (4.2) becomes,

$$\min_{\boldsymbol{\psi}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\boldsymbol{\psi}\|_2^2 + \lambda \sum_i \psi_i \quad \text{s.t.} \quad \psi_i \geq 0 \tag{4.7}$$

which is a standard quadratic program. The relaxation parameter, λ, determines the relative weight given to the two terms in (4.2), or equivalently, (4.7), and hence controls the number of non-zero terms retained in the final model. Increasing λ increases the cost associated with each term retained in the model, and tends to reduce the number of terms in the final model. Conversely, decreasing the value of the relaxation parameter tends to result in an increase in the number of model terms. The structure selection problem, therefore, has been reduced to searching for the optimal value for the relaxation parameter, λ. Optimal is used in the sense that the corresponding model produces the minimum mean squared error (MMSE) output predictions, when it is applied to novel data: data not used in the structure detection and parameter estimation procedure. Cross-validation is performed using methods described in Section 2.2.5, under Model Accuracy.

4.2 Algorithm

In practice searching for the MMSE value of the relaxation parameter is difficult, as the resulting cost-function is non-convex. As a result, there exists a potential for multiple local minima, as is evident in Figures 4.1 and 4.6, which show typical λ

vs MSE curves. An exhaustive search will be required to find the global minimum. However, such an exhaustive search is possible, since we can constrain λ to lie in

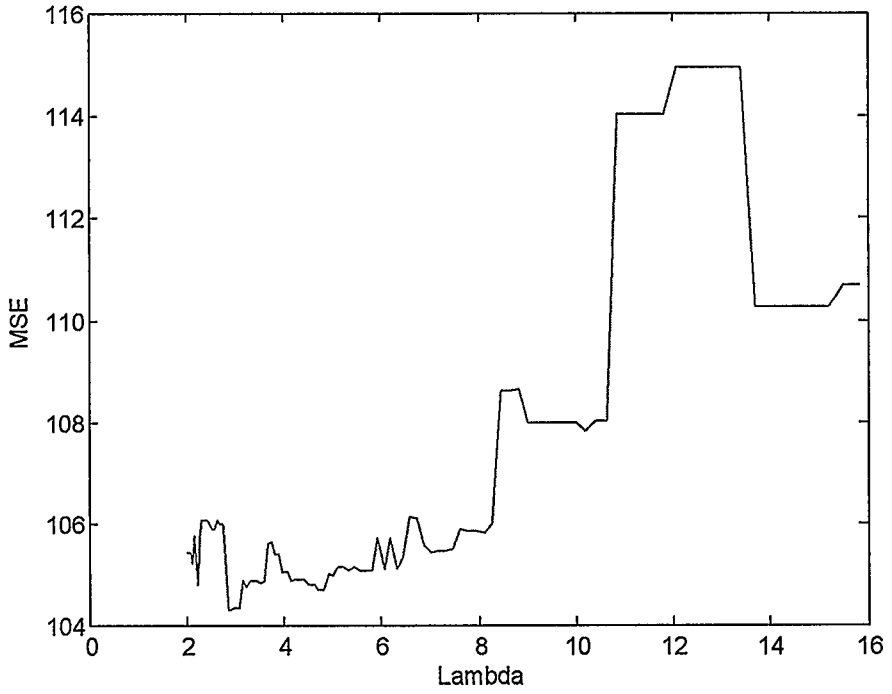

Figure 4.1: A typical Mean Squared Error vs relaxation parameter λ graph

a feasible range. The lower limit on λ is the point at which LASSO retains all of the model terms, clearly any further reduction to λ will not cause any additional terms to be added to the model. Similarly, there is an upper limit, at which point the QP has no feasible points, and hence no solution. We use a logarithmic sweep to identify these two extremes, and then use a finer grained search to construct the MSE between them, and finally choose the value of λ that minimizes the MSE. For every value of λ, the optimization in (4.7) is solved using the primal-dual Log Barrier method, resulting in a reduced model structure. In practice, the interior

point iteration does not reach the boundary, where one or more parameters are set exactly to zero, but terminates very close to it. Thus, the parameter vector is further simplified by setting those particular parameter values to 0 that reduce the cost-function. This step moves the parameter vector from an Interior Point (IP) to the boundary of the convex search area. Once the optimal model structure has been determined, a final linear regression is used to determine the parameter values that minimize the MSE, without any of the bias that the regularization introduces.

4.3 Application: Elbow Stiffness Dynamics

In this section we present an application of the basis expansion algorithm, described in Chapter 2, in combination with the LASSO based term selection technique discussed in the previous section. The final algorithm was tested on data from an experiment to investigate human elbow stiffness dynamics.

Joint dynamics describe the relationship between the torques applied about a joint and the position, velocity and acceleration of the corresponding limbs [48]. This relationship can be approximated by the linear, second-order low-pass filter given in eq. (3.6), which is repeated here for convenience.

$$\frac{\theta(s)}{T(s)} = \frac{1}{Is^2 + Bs + K} \tag{4.8}$$

The parameters I, B and K in eq. (4.8) represent inertia, damping and spring constant and depend on background activation, position and perturbation amplitude. These linearized I, B, K models are only valid under specific operating conditions, and furthermore, they do not include any provision for modelling the actions of reflexes. System identification techniques have been used to estimate models of joint

dynamics by fitting models between input perturbations and the system's response. Results of such studies have been presented in [16], [19], [22], [45], . These studies indicate that human joint dynamics can be modelled quite effectively using the linear, second-order function shown in Eq.(4.8).

Applications of these identified models have also been quite wide-spread. In Mirbagheri et al., system identification techniques were used to measure ankle dynamic stiffness at different positions through the range of motion, for passive and active conditions, in spastic, spinal cord injured (SCI) and healthy subjects. The study was undertaken to address issues regarding spastic hypertonia. Spasticity is a motor disorder associated with lesions at different levels of the nervous system due to traumatic injury, multiple sclerosis, cerebral palsy, or stroke. Hypertonia is an abnormal increase in muscle tone (which is characterized as the sensation of resistance felt as one manipulates a joint through a range of motion, with the subject attempting to relax) and is regarded as the defining feature of spasticity [37]. Two important issues the study addressed were describing the mechanical properties of the joint in terms of its dynamic stiffness and characterizing the relative contributions of intrinsic and reflex stiffness using a parallel-cascade identification technique. The results from this study demonstrated that the overall joint stiffness was substantially greater in SCI subjects, both intrinsic and reflex mechanical responses were significantly increased in SCI patients, but the major mechanical abnormality arose from increased reflex stiffness, and the relative contributions of reflex and intrinsic changes were strongly dependant on position and condition [37].

In another different application time-domain and frequency-domain multiple-input, multiple-output (MIMO) linear system identification techniques were used

to estimate the dynamic endpoint stiffness of a multijoint limb [39]. Estimates of the endpoint stiffness provides insight into how the nervous system normally controls motor behaviour. Understanding the stabilizing actions of the stiffness properties are useful in efforts to restore movements in disabled individuals by providing an indication of the effectiveness of rehabilitation interventions such as reconstructive surgery, external orthoses, and/or functional neuromuscular stimulation [39].

The above studies are just two examples of the various applications where system identification techniques and joint dynamics studies can be used. These studies provide useful quantitative results which can be used in clinical environments.

However, one of the uncertainties regarding this method of identification is whether the perturbation actually changes the system dynamics. The linearized models depend on the perturbation amplitude, so turning on the perturbation is much like moving from one operating point (zero amplitude) to another operating point [19]. The goal of this experiment was to use TV system identification to detect transients at the onset of perturbation. The actual experiment involved perturbing elbow posture to estimate elbow dynamics. The identification algorithm described in this chapter was used to estimate elbow dynamics using experimental data.

4.3.1 Experimental Apparatus

These experiments were performed at the Rehabilitation Institute of Chicago by Dr. Eric Perreault in March 2005. A linear motor (Copley ThrustTube, TB3806; Copley Controls, Canton MA) was used to perturb elbow posture. The motor was oriented orthogonal to the forearm and the upper limb was strapped into an adjustable trough to restrict motion to the elbow joint only. A custom-fitted fiberglass cast extending

from the fingers to the middle of the forearm was used to maintain the wrist joint in a neutral position and to attach the forearm to the actuator via an aluminum plate epoxied under the cast and a precision bearing centered at the wrist. This configuration allowed rotation only in the horizontal plane. Elbow torques were measured by a load-cell (67M25; JR3; Woodland,CA) mounted between the cast and linkage. This sensor has a force range of $\pm 200N$, and a moment range of $\pm 12Nm$, with an accuracy of better than $0.05N$. Actuator motion was measured using a linear encoder (RGH24; Renishaw, Gloucestershire, UK) with a measurement resolution of $1\mu m$. The actuator was controlled by a dedicated computer running Matlab xPC (The Mathworks, Natick MA). This system was also used for data collection and to provide the subject with visual feedback of the voluntarily generated joint torques. Motor control and data acquisition occurred at $5KHz$. The analog force signals were sampled by a 16-channel, 16-bit data acquisition system (PCI-DAS1602/16; Measurement and Computing, Middleboro, MA). Prior to sampling the force signals were anti-alias filtered using custom-built, differential input, 4th order Bessel filters with a cut-off frequency of 500Hz. The experimental set-up is shown in Fig. 4.2.

In the experiments, the subject was instructed to maintain the background contraction level presented on an oscilloscope. Once the target force had been reached, a broadband perturbation was applied to the input of the position servo for 5 seconds. The goal of the experiment was to detect any changes that might have resulted in the dynamic stiffness of the elbow as the result of the perturbation. A total of 28 such trials were performed.

Fig. 4.3 shows the data recorded during Trial 20, and is typical of the experimental recordings. For the stiffness models, displacement was taken as the input and

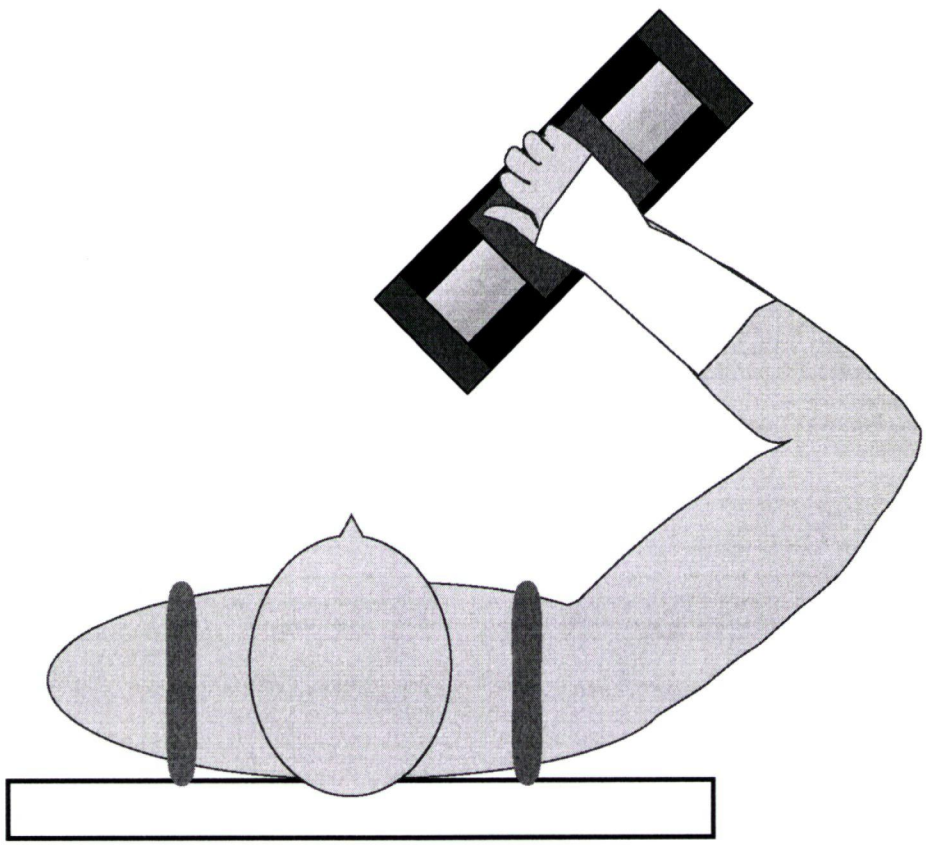

Figure 4.2: Experimental Setup for the Elbow Dynamic Experiment - Figure courtesy Eric Perreault

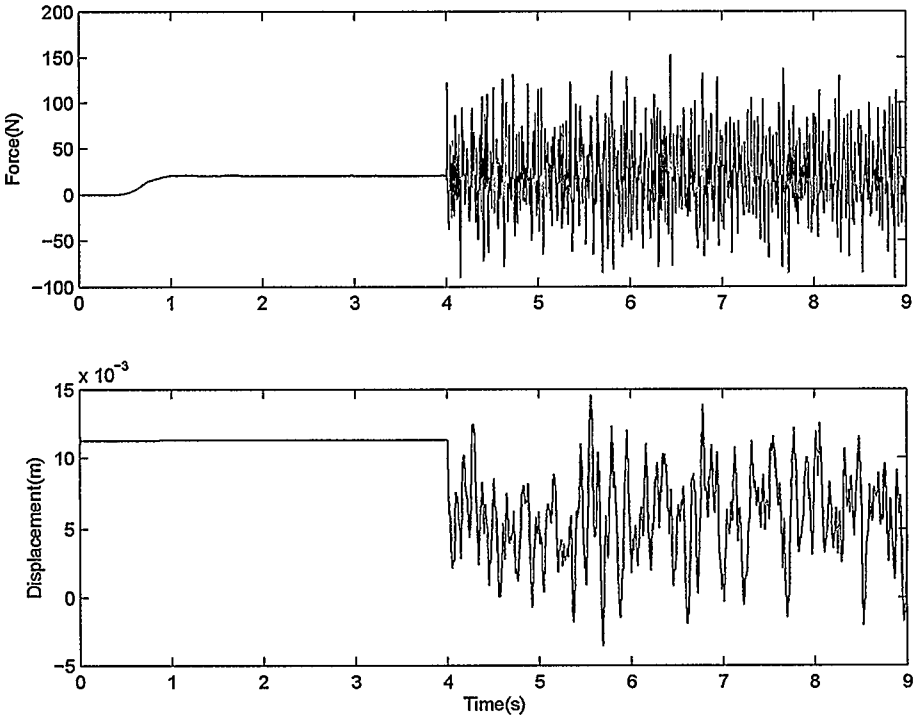

Figure 4.3: Typical Experimental Data, Force and Displacement over 9s

force as the output. Since the broadband perturbation was applied for 5 seconds, only the last 5 seconds of the data were used for the identification purposes. The DC value from the first 4 seconds of each trial was removed from the actual data used.

4.4 Results: Elbow Stiffness Dynamics

Since displacement was used as the input and torque as the output, stiffness models were identified using the two-sided IRFs described in Section 2.2.4. Acausal models represented by two-sided IRFs, as shown in Eq. (2.24) were fitted to the data [15]. The equation has been repeated here for convenience.

$$y(t) = \sum_{j=-Q_{ac}}^{Q_c} b(j,t)x(t-j) + e(t) \tag{4.9}$$

The identified model included a total of 31 time-varying IRF weights with 5 anti-causal coefficients, so that $Q_{ac} = 5$ and $Q_c = 25$ in equation (2.24). The time-variations at each point in the IRF were represented using the first 4 Tchebychev polynomials as shown in Figure 2.8. The stiffness IRFs that were estimated using the basis expansion algorithm, but without any term-selection, are shown in Fig. 4.4. Using this particular basis functions a subtle time-dependance is evident, especially in the amplitudes of the highest two peaks.

Next, the LASSO procedure was used to remove any insignificant terms from the model. First, the optimal value of the relaxation parameter was determined using an exhaustive search. Figure 4.6 shows the MSE estimated using 5-fold cross-validation as a function of the relaxation parameter, λ. It reaches a minimum at $\lambda = 0.28$, where the LASSO eliminated 80 of the 186 terms (43%) in the initial model. The resulting time-varying IRFs are shown in Fig. 4.5. As we can see the time-variations apparent

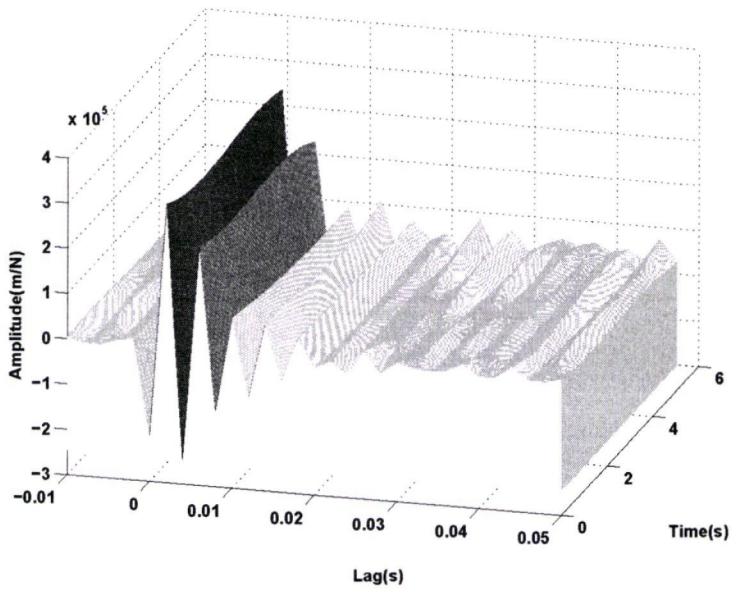

Figure 4.4: Time-Varying impulse responses for elbow stiffness dynamics using Tchebychev basis functions and estimated using linear regression

in this figure are very similar to the one in Fig 2.8. Again, subtle time-dependance is evident in the amplitudes of the highest two peaks.

Finally, the model was tested on data from a separate trial. The results are shown in Fig. 4.7. This particular model could not predict output from novel data very well with the VAF being -6255.2%.

Using 4 Tchebychev polynomials as basis functions, we obtained results that had subtle time variations. From the results obtained by testing the model with novel data it seems that the model is not well characterized using these low degree polynomials.

If any time-variation is present in the data, it is likely to be a transition between an initial, pre-perturbation state, and a final, steady-state. Note, however, that the low-degree polynomials used in the previous analysis are not well-suited to such

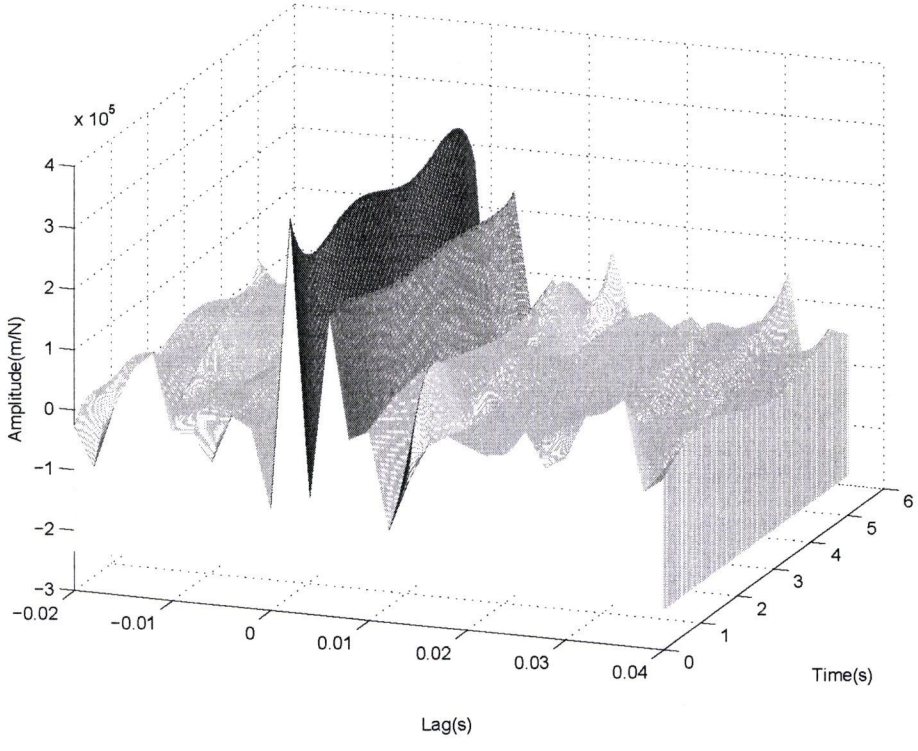

Figure 4.5: Impulse responses for elbow stiffness dynamics estimated using Tchebychev polynomials as basis functions and LASSO for term-selection

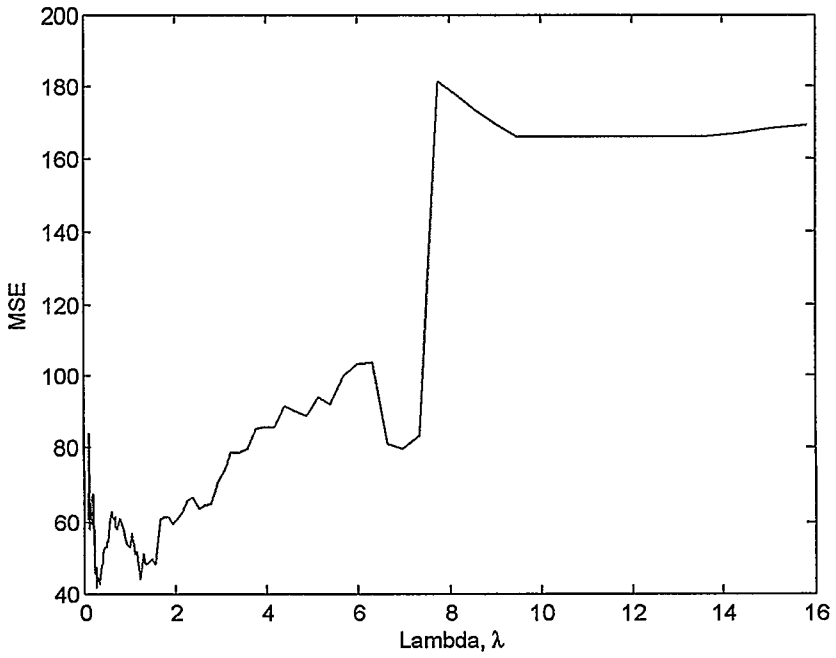

Figure 4.6: Mean squared-error estimated using 5-fold cross-validation, as a function of the relaxation parameter,λ

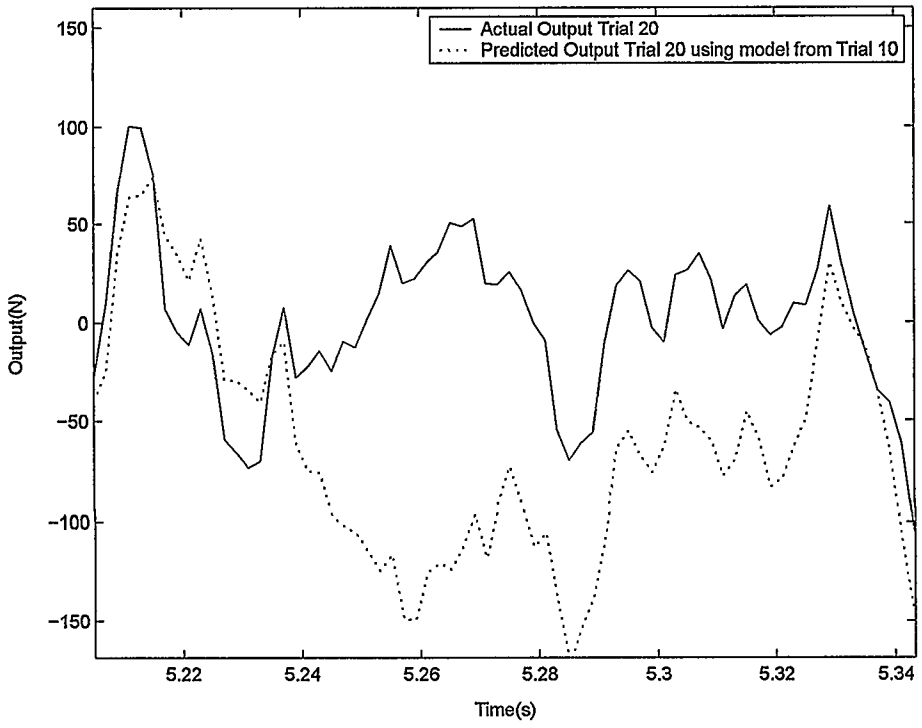

Figure 4.7: Output (solid) and model prediction (dashed) for data from Trial 20, using model identified from trial 10, LASSO model

transient time-variations as they tend to spread the TV across the whole record. Thus, for this particular application, a basis that can represent transient phenomena is required (in addition to a constant term that will represent the steady-state). Laguerre filters, which are described in detail in the next section, may well be suitable basis functions since they exhibit transients that are likely similar to the anticipated time-varying behaviour.

Laguerre Filters

The basis of discrete Laguerre filters comprises the IRFs defined by

$$h_k(\tau) = \alpha^{(\tau-k)/2}(1-\alpha)^{1/2}\sum_{i=0}^{k}(-1)^i \binom{\tau}{i}\binom{k}{i}\alpha^{k-i}(1-\alpha)^i \;\; \tau \geq 0 \qquad (4.10)$$

From equation (4.10) it is clear that all the IRFs are dependent on a single parameter, α, called the 'decay parameter'. Fig. 4.8 illustrates the IRFs of the first four Laguerre filters for $\alpha = 0.25$, a value often used in system identification because it provides a reasonable compromise between number of parameters and the length of the resulting IRFs [47].

Laguerre filters have three very desirable properties [36], [49]:

1. The IRFs are orthogonal and lead to well-conditioned estimation problem.

2. The IRFs decay exponentially to zero as τ goes to infinity.

3. The outputs of the Laguerre filters can be computed efficiently [38], by recursively applying the same filter. Thus, the zero-order filter output is obtained using the relation

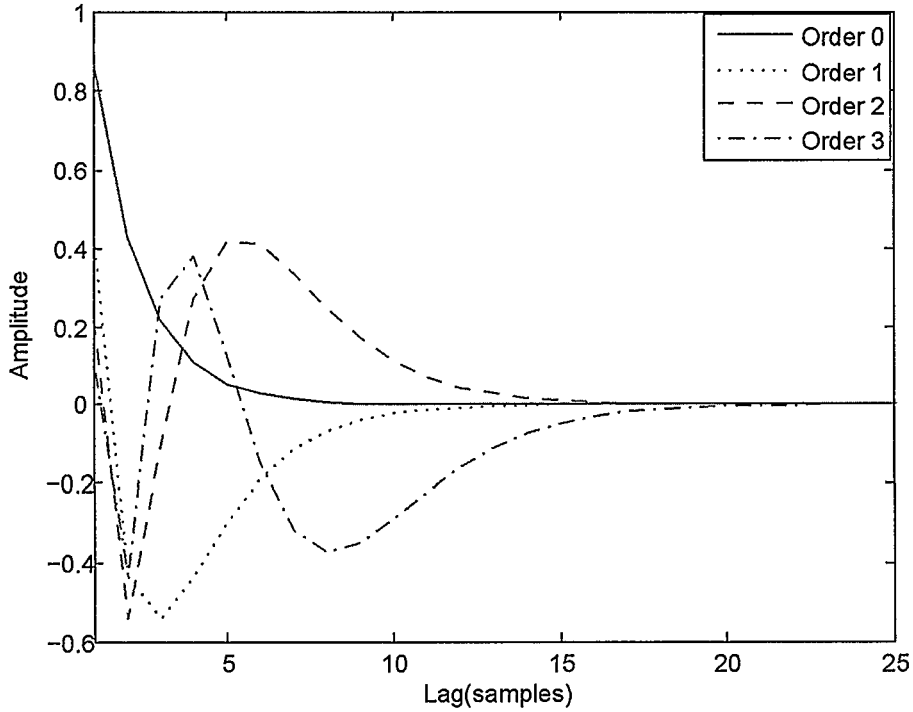

Figure 4.8: The first four impulse responses of Laguerre filters for $\alpha = 0.25$

$$x_0(t) = \sqrt{\alpha}x_k(t-1) + \sqrt{1-\alpha}u(t),\ x_0(0) = 0 \tag{4.11}$$

where u(t) is the input. The z-domain transfer function of this filter can be written as:

$$H_0(z) = \frac{\sqrt{1-\alpha}}{1-\sqrt{\alpha}z^{-1}} \tag{4.12}$$

The output of filter k can be obtained by filtering the $(k-1)^{th}$ filter with

$$x_k(t) = \sqrt{\alpha}x_k(t-1) + \sqrt{\alpha}x_{k-1}(t) - x_{k-1}(t-1),\ x_k(0) = 0 \tag{4.13}$$

The z-domain transfer function of equation (4.13) can be written as:

$$H_{ap}(z) = \frac{\sqrt{\alpha}-z^{-1}}{1-\sqrt{\alpha}z^{-1}} \tag{4.14}$$

so,

$$X_k(z) = H_{ap}(z)X_{k-1}(z) = H_o(z)(H_{ap}(z))^k U(z) \tag{4.15}$$

where, q is the forward shift operator defined in section 2.1.

Laguerre Results

While Laguerre filters (with the inclusion of a constant term) seem to be an ideal choice for basis functions, the performance of the basis expansion depends on the value of the decay parameter α. Thus, application of the LASSO algorithm required finding the optimum values for both the decay parameter, α, and the relaxation parameter λ. This was done by setting λ to various values from 0.1 to 15. For each

different value of λ, 100 iterations of the algorithm, each with a different α were processed. The optimum α was chosen based on the α value that minimized the Mean Squared Error (MSE, also called the 5-Fold CV Cost-Function). The following table shows the various λ values and the corresponding α and MSE values.

λ	Optimum α	MSE
0.10	0.89	295.10
1.50	0.87	33.12
2.80	0.84	63.41
5.00	0.86	78.23
7.00	0.85	79.53
10.00	0.84	59.55
12.00	0.70	65.32
15.00	0.84	85.03

Table 4.1: Various λs and the corresponding optimal α

From Table 4.1, $\lambda = 1.5$ and $\alpha = 0.87$ minimizes the MSE. Figure 4.9 shows a graph of λ vs MSE with λ ranging from 0.1 - 15 and $\alpha = 0.87$.

The next step was to perform a further search, fixing α to 0.87 and searching over λ from 0.1 to 6. This search yielded an optimum $\lambda_{opt.} = 0.81$. A finer search over α was performed with λ varying from 0.50 to 2.0 in steps of 0.25.The results are tabulated in Table 4.2.

So, again $\lambda = 0.81$ and $\alpha = 0.87$ minimizes the MSE. A further refined search was done by setting $\alpha = 0.87$ and searching over λ from 0.4 to 2.0. This search yielded $\lambda_{opt} = 0.81$. Thus, the final values used for α and λ were 0.87 and 0.81 respectively. Figure 4.10 shows the MSE as a function of λ for $\alpha = 0.87$ and λ varying from 0.4 to 2.

From Fig.4.10, it is clear that the Laguerre filters are a better fit with the stiffness

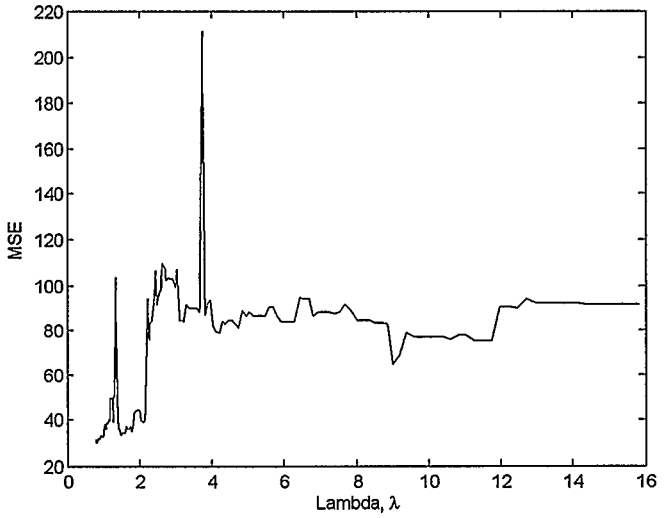

Figure 4.9: Lambda vs MSE graph for elbow stiffness dynamics using Laguerre Basis, $\alpha = 0.87$

λ	Optimum α	MSE
0.50	0.88	31.43
0.75	0.86	32.11
0.81	0.87	30.52
1.00	0.86	33.68
1.25	0.86	34.26
1.50	0.88	33.12
1.75	0.87	34.62
2.00	0.88	37.34

Table 4.2: λs and the corresponding optimal α, further refined search

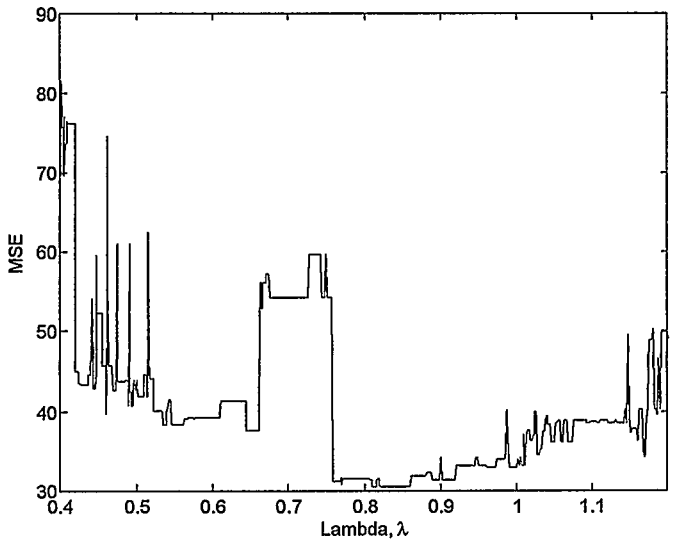

Figure 4.10: Lambda vs MSE graph for elbow stiffness dynamics using Laguerre Basis, $\alpha = 0.84646$

models than the Tchebychev filters. The MSE values from Fig.4.10 are much lower than the MSE values obtained from using the Tchebychev basis functions.

Once the optimum α and λ were chosen, the identification algorithm was used to identify dynamic elbow stiffness using Laguerre Functions as the basis functions. Several different trials were used for the model estimation. The results shown below were obtained from the application of the new algorithm to Trial 10 of the experimental data.

The first 6 Laguerre filters with the addition of a constant were used as basis functions. As before, acausal IRFs were fitted to the data. The model included a total of 31 time-varying IRF weights with 10 anticausal coefficients, such that, Q_{ac} = 10 and $Q_c = 20$ in equation (2.24). The time-varying IRFs are shown in Fig. 4.11. From the figure it is apparant that the model has transient time-variations. However,

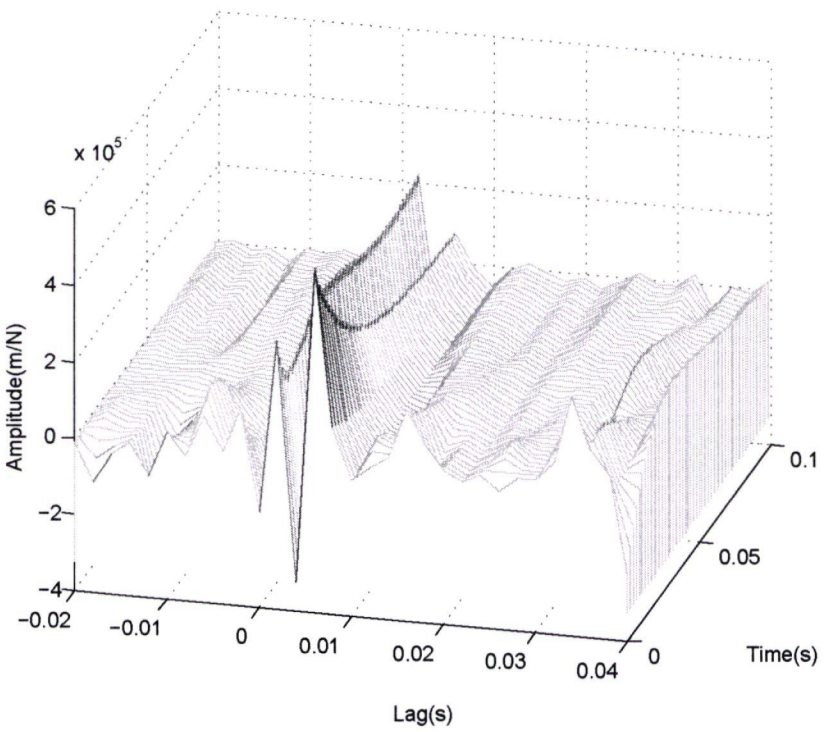

Figure 4.11: Time-Varying IRFs obtained for elbow stiffness dynamics using Laguerre Basis

the time-variations settle down very quickly (within 0.1s as seen in the figure). One of the reasons for this is the choice of Laguerre filters as basis functions because of their inherently transient nature. The LASSO term-selection procedure eliminated 120 out of 186 terms (64.5%) to obtain this relatively sparse model structure. An in-sample data prediction was done and as with the model-structure obtained with the Tchebychev polynomials, the model was tested on data from a separate trial. The results are shown in Fig. 4.13.

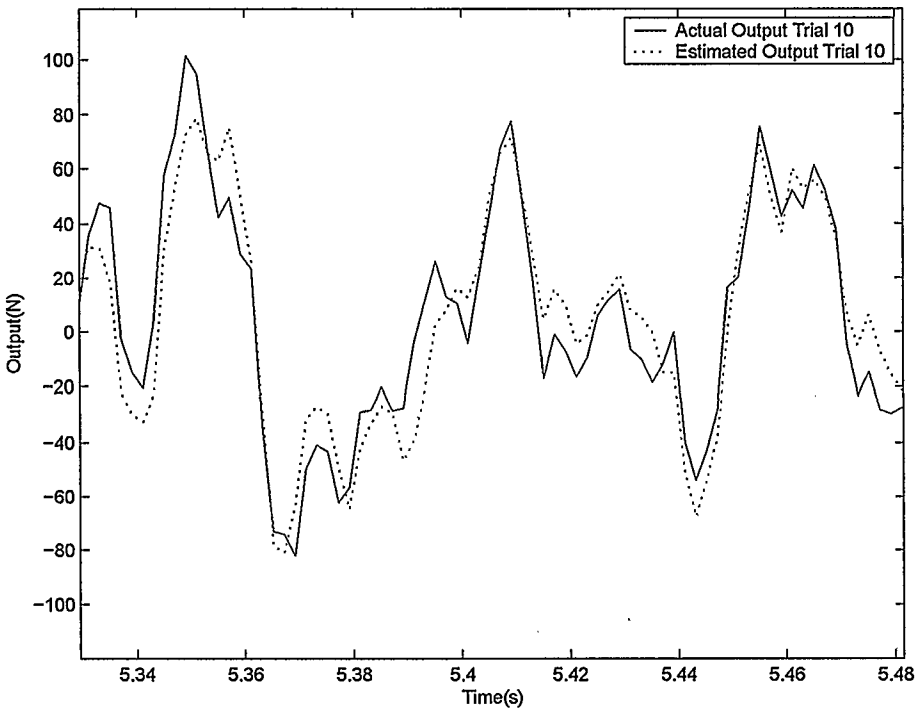

Figure 4.12: In-sample prediction using Trial 10

From Fig. 4.12, 4.13 it is clear that the actual and predicted output match quite closely. To quantify the results, the average VAF was calculated. The average VAF for the in-sample prediction was 82.7% while the average VAF obtained for the novel

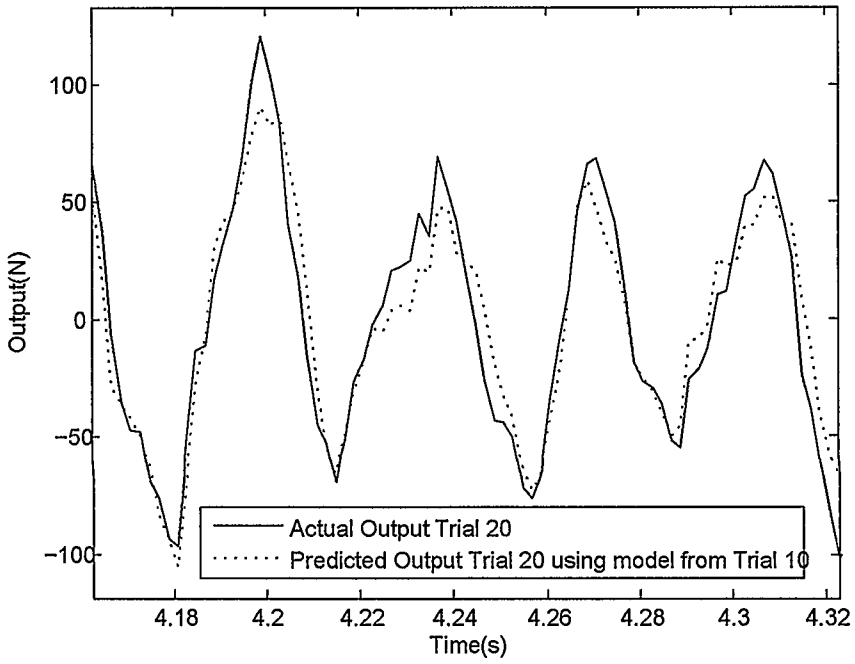

Figure 4.13: Output (solid) and model prediction (dashed) for data from Trial 20 using model identified from Trial 10

data prediction was 80.32% which shows that the model obtained is quite robust. The slight variation between the two VAFs could be due to either noise-fitting or slight differences between the two trials.

4.5 Discussion: Elbow Stiffness Dynamics

In this section the novel algorithm for identifying linear, time-varying systems was demonstrated on real data. Elbow stiffness dynamics were identified using this algorithm which included temporal basis expansions and term selection using LASSO. Two different basis functions were used to identify the time-variations in the IRFs. Initially, the first four Tchebychev polynomials (including a constant term) were used as basis functions. The IRFs showed subtle time-variations spread evenly across the entire data record. However, the model performed poorly when tested on novel data, with the VAF between the actual and predicted output being -6255.2%. Since short, transient time-variations were expected, Laguerre filters were chosen as the next set of basis functions. The first six Laguerre filters (plus a constant) were used as basis functions. Time-varying acausal IRFs were then obtained and short transient changes, lasting 0.1s, were observed at the onset of the perturbation. This model was tested on in-sample and novel data and the VAFs obtained were 82.6% and 80.34% respectively, which were far superior than the results obtained using Tchebychev polynomials. This demonstrated the suitablility of Laguerre filters as basis functions for this particular application. Also, since the results obtained from the Laguerre filters show that the time-variations appear to be over after 0.1s, the low-order Tchebychev polynomials were bound to fail, since they spread the vari-

ations across the whole 5 second record. It is clear that higher-order Tchebychev polynomials need to be used, to model such fast time-variations. The experiment was performed to study the dynamic brought upon by the onset of perturbation. To this end, short, transient changes were observed at the start of the perturbation in the acausal, TV IRFs. Immediately following the perturbation, significant changes were observed in the stiffness IRFs. The robust performance of the TV model on novel data suggests strongly that these transient time-variations are real. These transient changes could be due to a sudden change in the perturbation amplitude, which have been demonstrated in [19]. A similar study performed to track the change in ankle joint stiffness during a large imposed movement involved superimposing a small, stochastic perturbation onto the "ramp" perturbation [21], [22]. The stiffness dynamics identified in this study last about 0.3s after the onset of the ramp. This result is consistant with our findings which show the transient changes settling down after 0.1s after the onset of perturbation. The results obtained from this experiment do not provide conclusive answers to whether the perturbations have an effect on the system dynamics being measured. These results should be taken as a starting point and applied to further detailed experiments in this area.

Chapter 5

Conclusion

This thesis describes two novel techniques for time-varying linear system identification. An algorithm that combines ensemble and basis expansion (BE) techniques is presented in Chapter 3, while the use of a novel structure detection technique is discussed in Chapter 4. The new algorithm described in Chapter 3 is called the Ensemble Basis Expansion (EBE) algorithm and also contains the details of the ankle-compliance simulation used to demonstrate the validity of this algorithm.

The ankle-compliance simulation generated the data. The BE and the EBE algorithms were used to identify models using the generated data at various noise levels and the results were compared. The BE algorithm performed very well in the noise-free case. However, when data with 7dB of additive noise were used, significant errors were observed at the beginning and at the end of the data records. A second simulation at 12dB SNR showed somewhat better results using the BE algorithm. The EBE algorithm used 6 times as much data i.e. 6 ensembles and performed much better than the BE algorithm at both 7 and 12 dB SNR. In addition, Monte-Carlo simulations (100) were performed to estimate the means and variances of the IRFs at 7dB SNR. Again, the standard deviations obtained from the BE algorithm show much greater variability than the ones obtained from the EBE algorithm, proving the robustness of this new technique.

Chapter 4 introduced the new term selection technique LASSO. The new algorithm combining BE and LASSO was described in detail. The algorithm was

applied to the identification of elbow stiffness dynamics to detect transient changes at the onset of broadband perturbation. The basis expansion in conjunction with the term-selection using LASSO was used to identify linear, time-varying, acausal IRFs to model the elbow stiffness dynamics. Initially, the first four Tchebychev polynomials (including a constant) were used as basis functions. The time-variations obtained were spread smoothly across the whole data record. However, the model proved to be not very robust when tested on novel data. The model obtained from Trial 10 was used to predict the output from Trial 20. The VAF between the predicted and actual output was -6255.5%. Since short, transient time-variations were expected, Laguerre filters were the next logical choice as basis functions. Indeed, with the use of Laguerre filters as basis functions, short, transient variations were observed within 0.1s of the onset of perturbation. The final model was tested to predict the in-sample output and was further tested on novel data. The in-sample VAF was 82.7%. The output from Trial 20 was predicted using the model from Trial 10. The VAF between the predicted and actual output was 80.34%. The difference between the two could be due to noise-fitting or differences between the two trials. The VAF percentage suggests strongly that the transient time-variations seen in the IRFs are real. These changes could be due to a number of reasons including due to a sudden change in operating point at the onset of perturbation. The results obtained from this application prove that the term selection has been well integrated into the algorithm. Also, the algorithm is quite robust and it has been proven to model linear, time-varying systems successfully.

5.1 Contributions

The completion of a successful research project includes a combination of ideas and work from many different people and sources. This section highlights my personal contributions to this research. Temporal basis expansions [53] and Ensemble techniques [32] are the current state-of-the-art techniques for identifying linear, time-varying systems. My contribution included integrating the two algorithms together into the EBE algorithm, and successfully demonstrating the superior performance of the EBE algorithm over just BE, using ankle joint simulations. Next, the issue of eliminating irrelevant parameters introduced by BE was dealt with. Again, LASSO was an existing mathematical technique for obtaining the sparsest possible representations of objects. My contribution in this case was integrating LASSO with the temporal basis expansions. I performed extensive research into the most effective way of determining the optimum relaxation parameter, λ. Minimizing the 5-fold CV cost-function and searching over a range of values was finally determined to be the most efficient way of determining the optimum λ. Once the algorithm was perfected the next step was to apply it to experimental data. Again, my personal contribution included applying the algorithm to model elbow stiffness dynamics. The experimental data was obtained from Northwestern University in Chicago and my contribution was applying the new algorithm on the data and identifying time-varying, acausal IRFs. This included experimenting with two different basis functions (Tchebychev Polynomials and Laguerre Filters) and identifying IRFs and testing the robustness of the algorithm by testing the model with in-sample and novel data.

5.2 Future Work

Developing and proving the robustness of the EBE algorithm and the LASSO technique for term-selection in identifing linear, time-varying systems was the first step. It is recommended that future work in this area be conducted in further evaluating the algorithm and extending it for modelling more complex systems. This algorithm needs to be evaluated against other term selection techniques like FOS [24], OPS [53] and Bootstrap [29]. The evaluation procedure needs to be quite extensive and the algorithms have to be compared at various different noise levels and various degrees of overparameterization i.e. adding irrelevant parameters to the model to test the robustness of the algorithm. Initial studies in this direction have been done in the NARMAX context [30]. Monte-carlo simulations have been done in the area of non-linear system identification comparing the rate of exact, over and under modelling with changing data-length and various levels of SNR for LASSO and Bootstrap [28]. The Bootstrap technique in particular has been compared quite extensively to T-Tests and Stepwise Regression [29]. Similar studies need to be performed for linear, time-varying systems as well.

Furthermore, research could be done in extending the algorithm from identifying linear, TV systems to non-linear, TV systems with iterative minimizations including using it identify Hammerstein systems [46]. Also, if this algorithm is extended to identify parallel cascade models [2], these models which include reflexes can be applied to the elbow stiffness models to get more accurate representations.

Lastly, further detailed experiments could be performed using the results presented in this thesis as a starting point. Experiments designed to subject the elbow

to a "ramp on" and "ramp off" perturbation with sharp transitions, could look at dynamics at the start and at the end of the perturbation. Such changes could be modelled by using Walsh functions and Laguerre filters in conjunction as basis functions. Such experiments would provide more conclusive answers as to the changes brought upon by the system dynamics by the onset of perturbation.

Bibliography

[1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, AC-19:716–723, 1974.

[2] M. Baker, L. Zhao, R. Wagner, and R.E. Kearney. Time-varying parallel-cascade system identification of ankle stiffness. *Engineering in Medicine and Biology Conference Proceedings*, 2:4688–4691, 2004.

[3] I.S. Baruch, L.R. Beltran, M.R. Garrido, and E. Gortcheva. A fuzzy-neural multi-model for mechanical systems identification and control. *Proceedings of the 2003 IEEE International Conference on Systems, Man and Cybernetics*, 3:2113–2119, 2003.

[4] H.H.J. Bloemen, T.J.J.V.D. Boom, and Verbruggen H.B. Model-based predictive control for hammerstein-wiener system. *International Journal of Control*, 74(5):482–495, 2001.

[5] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20:33–61, 1998.

[6] C. Chong-Yung, P. Chun-Hsieng, and T. Po-Shun. Blind identification of instantaneous mimo systems via a system to hos based source extraction filter relation. *IEEE 5th Workshop on Signal Processing Adavances in Wireless Communications*, pages 541–545, 2004.

[7] E.G. Jr. Collins, Y. Zhao, F. Alvi, M.I. Alidu, and P.J. Strykowski. Feedback

control for counterflow thrust vectoring. *Proceedings of 2004 American Control Conference*, 4:3654–3659, 2004.

[8] A.S. French and M.J. Korenberg. Analysis of a non-linear cascade model for sensory encoding by modification of ion channels. *Proceedings of the 12th Annual International Conference of the IEEE, Engineering in Medicine and Biology Society*, pages 25–26, 1990.

[9] K. Fruzetti, A. Palazoglu, and K.A. McDonald. Nonlinear maximum predictive control using hammerstein models. *Journal of Process Control*, 7(1):31–41, 1997.

[10] P.E. Gill, W. Murray, D.B. Ponceleon, and M.A. Saunders. Solving reduced kkt systems in barrier methods for linear and quadratic programming. *Technical Report SOL*, 91(7), 1991.

[11] G.H. Golub and C.F. Van Loan. *Matrix Computation.* The John Hopkins University Press, North Charles Street, Baltimore, Maryland, 2 edition, 1993.

[12] Y. Grenier. Time-dependent arma modelling of nonstationary signals. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-31(4):899–911, 1983.

[13] R.A. Gunn, S.R. Gunn, F.E. Turkheimer, J.A.D. Aston, and V.J. Cunningham. Positron emission tomography compartmental models: A basis pursuit strategy for kinetic modelling. *Journal of Cerebral Blood Flow and Metabolism*, 22(12):1425–1439, 2002.

[14] J.A. Hoffer, M. Haugland, and T. Li. Obtaining skin contact force information from implanted nerve cuff recording electrodes. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 3:928–929, 1989.

[15] I. Hunter and R. Kearney. Two-sided linear filter identification. *Med. Biol. Eng. Comput.*, 21:203–209, 1983.

[16] I.W. Hunter and R.E. Kearney. Dynamics of human ankle stifness: variation with mean ankle torque. *Journal of Biomechanics*, 15:747–752, 1982.

[17] S.D Hyunchul, J.K. Hyoun, and S. Sastry. Control system design for rotorcraft-based unmanned aerial vehicles using time-domain system identification. *Proceedings of the 2000 IEEE International Conference on Control Applications*, pages 808–813, 2000.

[18] R.E. Kearney and I.W. Hunter. System identification of human joint dynamics. *CRC Critical Reviews in Biomedical Engineering*, 18:55–87, 1990.

[19] R.E. Kearney and Hunter.I.W. Dynamics of human ankle stiffness: variation with displacement amplitude. *Journal of Biomechanics*, 15:753–756, 1982.

[20] R.E. Kearney, R.B. Stein, and L. Parameswaran. Identification of intrinsic and reflex contributions to human ankle stiffness dynamics. *IEEE Transactions on Biomedical Engineering*, 44(6):493–504, 1997.

[21] R.F. Kirsch and R.E. Kearney. Time-varying identification of human ankle joint stiffness dynamics during imposed movement. *Engineering in Medicine*

and Biology Conference Proceedings, 13:2030–2031, 1991.

[22] R.F. Kirsch and R.E. Kearney. Identification of time-varying stiffness dynamics of the human ankle joint during an imposed movement. *Experimental Brain Research*, 114:71–85, 1997.

[23] M.J. Korenberg. Identifying nonlinear difference equation and functional expansion representations: The fast orthogonal algorithm. *Annals of Biomedical Engineering*, 16:123–142, 1988.

[24] M.J. Korenberg. A robust orthogonal algorithm for system identification and time series analysis. *Biological Cybernetics*, 60:267–276, 1989.

[25] M.J. Korenberg and I.W. Hunter. The identification of nonlinear biological systems: Wiener kernel approaches. *Annals of Biomedical Engineering*, 18:629–654, 1990.

[26] S. Kukreja and R. Gunn. Bootstrapped DEPICT for error estimation in PET functional imaging. *NeuroImage*, 21:1096–1104, 2004.

[27] S.L. Kukreja. Parametric methods for nonlinear system identification. *Ph.D. Thesis, McGill University, Department of Biomedical Engineering*, 2001.

[28] S.L. Kukreja. A suboptimal bootstrap method for structure detection of nonlinear output-error models with application to human ankle dynamics. *International Journal of Control*, 78(12):937–948, 2005.

[29] S.L. Kukreja, H.L. Galiana, and R.E. Kearney. A bootstrap method for structure detection of narmax models. *International Journal of Control*, 77(2):132–

143, 2004.

[30] S.L. Kukreja, J. Löfberg, and M.J. Brenner. A least absolute shrinkage and selection operator (lasso) for nonlinear system identification. Technical report, Swiss Federal University of Technology Zurich, Institute of Automatic Control, Sep 2005.

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

[32] M. Lortie and R.E. Kearney. Identification of physiological systems: estimation of linear time-varying dynamics with non-white inputs and noisy outputs. *Medical and Biological and Engineering and Computing*, 39:381–390, 2001.

[33] S. Lu, K.H. Ju, and K.H. Chon. A new algorithm for linear and nonlinear arma model parameter estimation using affine geometry. *IEEE Transactions on Biomedical Engineering*, 48(10):1116–1124, 2001.

[34] J.B. MacNeil, R.E. Kearney, and I.W. Hunter. Identification of time-varying biological systems from ensemble data. *IEEE Transactions on Biomedical Engineering*, 39(12):1213–1225, 1992.

[35] K. Mao and S. Billings. Algorithms for minimum model structure detection in nonlinear dynamic system identification. *International Journal of Control*, 68(2):311–330, 1997.

[36] V.Z. Marmarelis. Identification of nonlinear biological systems using laguerre expansions of kernels. *Annals of Biomedical Engineering*, 21(6):573–589, 1993.

[37] M.M. Mirbagheri, H. Barbeau, M. Ladouceur, and R.E. Kearney. Intrinsic and reflex stiffness in normal and spastic, spinal cord injured subjects. *Experimental Brain Research*, 141:446–459, 2001.

[38] H. Ogura. Estimation of wiener kernels of a nonlinear system and fast algorithm using digital laguerre filters. *15th NIBB Conference*, pages 14–62, 1986.

[39] E.J. Perreault, R.F. Kirsch, and A.M. Acosta. Multiple-input, multiple-output system identification for characterization of limb stiffness dynamics. *Biological Cybernetics*, 80:327–337, 1999.

[40] S. Sanyal, S.L. Kukreja, E.J. Perreault, and D.T. Westwick. Identification of linear time-varying systems using basis pursuit. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2005.

[41] S. Sanyal and D.T. Westwick. Identification of time-varying joint dynamics. *Canadian Conference on Electrical and Computer Engineering Proceedings*, 2005.

[42] J. Shao. Linear model selection by cross-validation. *Journal of the American Statistical Association*, 88(422):486–494, 1993.

[43] R. Tibshirani. Regression shrinkage and selection via the lasso. *J.R.Statist. Soc. B.*, 58(1):267–288, 1996.

[44] H.L. Wei, S.A. Billings, and J. Liu. Term and variable selection for non-linear system identification. *International Journal of Control*, 77(1):86–110, 2004.

[45] P.L. Weiss, R.E. Kearney, and I.W. Hunter. Position dependence of ankle joint dynamics - I. passive mechanics. *Journal of Biomechanics*, 19:727–735, 1986.

[46] D.T. Westwick and R.E. Kearney. Separable least squares identification of nonlinear hammerstein models: Application to stretch reflex dynamics. *Annals of Biomedical Engineering*, 29:707–718, 2001.

[47] D.T. Westwick and R.E. Kearney. *Identification of Nonlinear Physiological Systems.* John Wiley and Sons, Inc., River Street,NJ, 1 edition, 2003.

[48] D.T. Westwick and E.J. Perreault. Identification of apparently acausal stiffness models. *Proceedings of the 2005 IEEE EMBC*, (1426), 2005.

[49] D.T. Westwick, B. Suki, and K.R. Lutchen. Sensitivity analysis of kernel estimates: Implications in nonlinear physiological system identification. *Annals of Biomedical Engineering*, 26:488–501, 1998.

[50] X. Xiao, Y. Li, and R. Mukkamala. A model-order selection criterion with application to cardio-respiratory renal systems. *IEEE Transactions on Biomedical Engineering*, 52(3):445–453, 2005.

[51] H. Yingbo, K. Abed-Meraim, and M. Wax. Blind system identification using minimum noise subspace. *IEEE Transactions on Signal Processing*, 45(3):770–773, 1997.

[52] M. Yoshizawa, H. Takeda, M. Miura, T. Yambe, Y. Katahira, and S. Nitta. Real-time cardiac output estimation of the circulatory system under left ventricular assistance. *IEEE Transactions on Biomedical Engineering*, 40(3):266–275,

1983.

[53] R. Zou, H. Wang, and K.H. Chon. A robust time-varying identification algorithm using basis functions. *Annals of Biomedical Engineering*, 31:840–853, 2003.

Appendix A

Primal-Dual Log Barrier Method

This section describes the Primal-Dual Log Barrier Method that was used to solve the quadratic program generated by BPDN. In [5], the QP in (4.7) was transformed to a perturbed linear program (LP), which was solved using the primal dual log barrier method. While both LP and QP are convex, and can be solved using standard commercial tools, solving an LP is generally much simpler and faster than solving an equivalent QP.

The large-scale perturbed linear programming problem mentioned in Chapter 4 was solved using the primal-dual log barrier method. A linear program in standard form is given by:

$$\min \mathbf{c}^T\mathbf{x} \text{ subject to } A\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq 0. \tag{A.1}$$

Equation (A.1) is called the *primal* linear program. The primal linear program is equivalent to the dual linear program given in the following equation:

$$\max \mathbf{b}^T\mathbf{y} \text{ subject to } A^T\mathbf{y} + \mathbf{z} = \mathbf{c}, \ \mathbf{z} \geq 0. \tag{A.2}$$

where $\mathbf{x}$ is called the primal variable, and $\mathbf{y}$ and $\mathbf{z}$ are called the dual variables. The term primal infeasibility refers to the quantity $\|\mathbf{b} - A\mathbf{x}\|_2$ and the term dual infeasibility refers to $\left\|\mathbf{c} - \mathbf{z} - A^T\mathbf{y}\right\|_2$. The term duality gap refers to the difference between the primal objective and the dual objective: $\mathbf{c}^T\mathbf{x} - \mathbf{b}^T\mathbf{y}$.

One of the fundamental theorems of LP states that (x, y, z) solves the LP in equations (A.1) and (A.2) if and only if the primal infeasibility, the dual infeasibility, and the duality gap are all zero [21]. The primal-dual log-barrier algorithm is based on solving the following perturbed LP [10]:

$$\min \mathbf{c}^T\mathbf{x} + \frac{1}{2}\|\gamma\mathbf{x}\|^2 + \frac{1}{2}\|\mathbf{p}\|^2 \text{ subject to} A\mathbf{x} + \delta\mathbf{p} = \mathbf{b},\ \mathbf{x} \geq 0, \tag{A.3}$$

where γ is a small regularization parameter.

For our case, $c = [\lambda, \lambda, \lambda...\lambda]^T$, $b = y$, $x = \theta$, and $A = M$.

The main steps of the algorithm is given below [5]:

1. Set parameters: the feasibility tolerance FeaTol, the duality gap tolerance PDGapTol, the regularization parameters γ.

2. Initialize $\mathbf{x} > 0$, $\mathbf{y}$,$\mathbf{z} > 0$, $\mu > 0$. Set $k = 0$.

3. Loop

   - Set

$$\begin{aligned} \mathbf{t} &= \mathbf{c} + \gamma^2\mathbf{x} - \mathbf{z} - A^T\mathbf{y}, \\ \mathbf{r} &= \mathbf{b} - A\mathbf{x} - \delta^2\mathbf{y}, \\ \mathbf{v} &= \mu\mathbf{e} - Z\mathbf{x}, \\ D &= (X^{-1}Z + \gamma^2 I)^{-1} \end{aligned} \tag{A.4}$$

   where $\mathbf{X}$ and $\mathbf{Z}$ are diagonal matrices composed of $\mathbf{x}$ and $\mathbf{z}$; $\mathbf{e}$ is a vector of ones.

- Solve

$$(ADA^T + \delta^2 I)\Delta \mathbf{y} = \mathbf{r} - AD(X^{-1}\mathbf{v} - \mathbf{t}) \tag{A.5}$$

for $\Delta \mathbf{y}$ and set

$$\Delta x = DA^T \Delta \mathbf{y} + D(X^{-1}\mathbf{v} - \mathbf{t}), \Delta \mathbf{z} = X^{-1}\mathbf{v} - X^{-1}Z\Delta \mathbf{x}. \tag{A.6}$$

- Calculate the primal and dual step sizes, ρ_p, ρ_d and update the variables:

$$\begin{aligned}
\rho_p &= 0.99 \times max\rho : \mathbf{x} + \rho\Delta \mathbf{x} \geq 0, \\
\rho_d &= 0.99 \times max\rho : \mathbf{z} + \rho\delta \mathbf{z} \geq 0; \\
\mathbf{x} &= \mathbf{x} + \rho_p \Delta \mathbf{x}, \\
\mathbf{y} &= \mathbf{y} + \rho_d \Delta \mathbf{y}, \\
\mathbf{z} &= \mathbf{z} + \rho_d \Delta \mathbf{z}, \\
\mu &= (1 - min(\rho_p, \rho_d, 0.99)))\mu.
\end{aligned} \tag{A.7}$$

- Increase k by 1.

4. Terminate if the following three conditions are satisfied:

   - Primal infeasibility < FeaTol
   - Dual infeasibility < FeaTol
   - Duality Gap < PDGapTol

Appendix B

Basis Pursuit Denoising using Primal-Dual Log Barrier Method, $MATLAB^{TM}$ Code

function theta = bpdn4(X,yy,lambda);

use basis pursuit denoising to fit a linear regression model. uses primal-dual log-barrier method described in S.S. Chen et al, 1999. linear system is solved using the direct left divide to find dy

```
[N,m] = size(X);
A = [X -X];
b = yy;
c = lambda*ones(2*m,1);
FeaTol = 0.01;
PDGapTol = 0.01;
CGTol = 0.001;
gamma = 1e-4;
mu = 0.01;
k = 0;
theta = X\yy;
x = [theta.*(theta>0); -theta.*(theta<0)] + 1e-10;
z = c;
y = zeros(N,1);
```

```
e = ones(2*m,1);
CGtime = 0;
searching = 1;
while searching
  t = c + gamma^2*x - z - A'*y;
  r = b - A*x - y;
  v = mu*e - z.*x;
  d = 1./(gamma^2 + z./x);
  D = diag(d);
  dvxt = d.*(v./x-t);
  time1 = cputime;
  dy = (A*D*A' + eye(N))\(r - A*dvxt);
  dx = d.*(A'*dy) + dvxt;
  dz = v./x - (z./x).*dx;
  rp = rho(x,dx);
  rd = rho(z,dz);
  x = x + rp*dx;
  y = y + rd*dy;
  z = z + rd*dz;
  mu = (1 - min([rd,rp,0.99]))*mu;
  k = k + 1;
  cond1 = norm(r)/(1+norm(x));
  cond2 = norm(t)/(1+norm(y));
  cond3 = (z'*x)/(1 + norm(x)*norm(z));
```

```
  if (cond1 < FeaTol) && (cond2 < FeaTol) && (cond3 < PDGapTol)
    searching = 0;
  end
end

theta = x(1:m) - x(m+1:2*m);



function rr = rho(x,dx);
test = -x./dx;
t2 = test(find(test>0));
if length(t2)<1
  t2 = 1;
end
rr = 0.99*min(t2);
```