

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]

PARAMETRIC METHODS FOR NONLINEAR SYSTEM IDENTIFICATION

Sunil L. Kukreja

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

Department of Biomedical Engineering



McGill University, Montréal

March 2001

© Sunil L. Kukreja, 2001



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-70068-2

Canada

Dedicated to my Family

My Mother

....for leading me into intellectual pursuits and never wavering in her support,

My Father

....for teaching me persistence,

My Brother

....for supporting me through some very crucial years.

I have often pondered over the roles of knowledge or experience, on the one hand, and imagination or intuition, on the other, in the process of discovery. I believe that there is a certain fundamental conflict between the two, and knowledge, by advocating caution, tends to inhibit the flight of imagination. Therefore, a certain naiveté, unburdened by conventional wisdom, can sometimes be a positive asset.

- Harish Chandra (Representations of Semisimple Lie Groups)
11 October 1923 - 16 October 1983 Kanpur, Uttar Pradesh, India
Quoted in R Langlands, Harish-Chandra, *Biographical Memoirs of
Fellows of the Royal Society* **31** (1985) 197 - 225.

Whatever you do will be insignificant, but it is very important that you do it.

- Mahatma Gandhi

Abstract

In this thesis, we have developed practical methods for the identification of linear, nonlinear and hybrid (multimode) systems which are applicable under relatively general conditions, i.e., when assumptions and conditions of the estimation technique are not violated. Since these algorithms were not designed specifically with any system(s) in mind, they should be applicable to experiments on a variety of systems in many different disciplines.

Results demonstrate that the (polynomial) NARMAX (Nonlinear Autoregressive, Moving Average eXogenous) model class is useful for modeling the input-output behavior of a block-structured representation of two biological models. Extensive simulations demonstrated that our bootstrap model order selection (BMOS) and bootstrap structure detection (BSD) algorithms have a high probability of success for selecting the order and structure of NARMAX models and are robust in the presence of measurement noise. In addition, we illustrate that the NARMAX model structure is well suited for modeling dynamics of nonlinear hybrid systems and develop a modified extended least squares (MELS) algorithm to estimate coefficients of these systems. Application of this algorithm to a model of the vestibulo-ocular reflex (VOR) showed that it is a robust method for estimating the coefficients of multimode systems.

Résumé

Dans cette thèse, nous avons développé des méthodes pratiques pour l'identification des systèmes linéaires, non linéaires et hybrides (à plusieurs modes de fonctionnement). Ces méthodes sont applicables dans des conditions relativement générales, c'est-à-dire lorsque les hypothèses et les conditions de la technique d'évaluation sont vérifiées. Puisque ces algorithmes n'ont pas été conçus pour un système spécifique, ils devraient être applicables aux expériences sur une variété de systèmes, dans un grand nombre de disciplines.

Les résultats démontrent que la classe de modèles NARMAX (Nonlinear Autoregressive, Moving Average eXogenous) est utile pour modéliser le comportement en entrée-sortie d'une représentation par blocs pour deux modèles biologiques. Des simulations poussées ont démontré que nos algorithmes de sélection d'ordre de modèle (bootstrap model selection, BMOS) et de détection de structure (bootstrap structure detection, BSD) ont une forte probabilité de succès et sont robustes en présence de bruit de mesure. En outre, nous illustrons que la structure NARMAX est appropriée pour modéliser la dynamique des systèmes hybrides non linéaires et nous développons un algorithme de moindres carrés étendus modifiés (modified extended least squares, MELS) pour estimer les coefficients de ces systèmes. L'application de cet algorithme à un modèle du réflexe vestibulo-oculaire a prouvé qu'il s'agit d'une méthode robuste pour estimer les coefficients des systèmes à plusieurs modes de fonctionnement.

Acknowledgments

In the course of completing requirements for a doctorate, it is sometimes unfortunate that a number of (the best) years slip by, and it is, therefore, difficult to recall all the people who have been instrumental in your intellectual growth. Nevertheless, I will attempt to recall who these people are. If I have, unintentionally, forgotten anyone I offer my sincere apologies.

I am deeply indebted to Dr. Socrates Rapagna, for many useful discussions on statistical matters, proof reading of many manuscripts for conferences, this thesis, and for encouragement. Dr. George P. H. Styan has been an inspiration to me in my continued pursuit for a deeper understanding of matrix analysis and statistics. I wish to thank him for his many useful discussions on regression and for his friendship. Dr. Peter E. Caines is a role model for me, a dedicated researcher who is truly concerned about a student's intellectual growth and welfare. I would like to thank Peter for his many insightful comments, genuine interest in my work and cheerful, friendly nature. I would like thank Dr. Benoit Boulet for providing me with useful suggestions regarding some contents of this thesis and for his stimulating discussions on system identification and control. I would like to thank Dr. Nitish V. Thakor, a good mentor and friend who has spent with many hours of his time discussing career goals and providing me with valuable insight and advice. I would like to thank Dr. David T. Westwick for providing me with some key guidance at the start of this Ph.D. work. It would be remiss of me at this point not to acknowledge the financial support of my thesis co-supervisors Dr. Henrietta L. Galiana and Dr. Robert E. Kearney. In addition, I would like to thank Dr. Henrietta L. Galiana for her personal guidance in matters regarding the completion of this thesis.

I would like to acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada, the Medical Research Council of Canada and the Max Stern Fellowship of McGill University.

I have had many fruitful discussions with my colleagues here in the department. In particular, I would like to mention Tatiana Nikitina, Gillian Walter and Dr. Ross Wagner. In addition, I would like to give a very special thanks to David Grennan who has been a good friend and colleague and for his many interesting late night discussions at Thompson House.

There have been a few other people along this long and winding road that have made a significant impact on my life. I would like to take this opportunity to thank them for their encouragement, support and guidance; my cousins Dr. Sushil Shirodkar & Kanchan Tambe and my friend Dr. Sašo Jezernik.

Last and most important, I owe a debt of gratitude to my one-person cheerleader, Margherita, without whom this thesis would have never been completed, for putting up with all my thesis and paper revisions, for never losing patience with my constant oscillations between industry and academia and for being my beacon of light.

Contents

1	Introduction	1
1.1	Thesis Overview	3
2	Literature Review	4
2.1	Introduction	4
2.2	Modeling Techniques	5
2.2.1	Morphological Modeling	5
2.2.2	Black-Box Modeling	5
2.3	The Identification Problem	7
2.4	Classes of System Identification	8
2.4.1	Nonparametric System Identification	9
2.4.2	Parametric System Identification	11
2.5	The NARMAX Model	15
2.5.1	NARMAX Representations of Nonlinear Systems	15
2.6	Current Methodology	19
2.6.1	Model Order Selection	20
2.6.2	Structure Detection	32
2.6.3	Parameter Estimation	36
2.6.4	Model Validation	46
3	NARMAX Representation of Ankle Dynamics	51
3.1	Introduction	51
3.2	Parallel Pathway Model of Ankle Dynamics	52

3.2.1	Discrete-Domain Approximation to a Derivative vs. Bilinear Transform	53
3.2.2	Theoretical Analysis	53
3.3	Simulations	56
3.3.1	Output Accuracy of NARMAX Ankle Model	56
3.3.2	Parameter Estimation of NARMAX Ankle Model	59
3.3.3	Reduction of Dimensionality	63
3.3.4	Estimation of Continuous-Time Parameters of Ankle Model	68
3.3.5	Input Noise Sensitivity	76
3.3.6	Input and Output Noise Sensitivity	79
3.3.7	Closed-Loop Simulation	82
3.4	Experimental Data	86
3.4.1	Apparatus	86
3.4.2	Perturbations	86
3.4.3	Procedures	86
3.4.4	Results	87
3.5	Discussion	89
3.5.1	Discrete-Time Parameter Estimation	89
3.5.2	Continuous-Time Parameter Estimation	91
3.5.3	Input Noise Sensitivity	92
3.5.4	Closed-Loop Simulation	93
3.5.5	Experimental Data	94
3.5.6	Discrete to Continuous-Time Parameter Mapping	95
3.5.7	Continuous to Discrete-Time Transformations	95
3.5.8	Simulation Techniques	96
3.5.9	Total Least-Squares	97
3.5.10	Relevance of NARMAX Parameters to Physiology	98
3.5.11	Summary of Findings	98
3.6	Conclusions	99

4	Bootstrap Structure Detection	101
4.1	Overview	101
4.2	Introduction	102
4.3	Structure Detection	102
4.4	Mathematical Preliminaries	103
4.5	Application to Structure Detection	108
4.5.1	BSD Algorithm	110
4.6	Implementation of BSD Algorithm	110
4.7	Simulations	111
4.7.1	Bandlimited Input and White Noise	112
4.7.2	Assessment of Parameter Statistics	114
4.7.3	Convergence Analysis of Bootstrap	117
4.7.4	Bandlimited Input and Bandlimited Noise	120
4.8	Simulated Biological Example	123
4.8.1	Simulation Protocol	124
4.8.2	Results	125
4.9	Discussion	126
4.9.1	Gaussian White Noise	126
4.9.2	Convergence Analysis	127
4.9.3	Bandlimited Noise	128
4.9.4	Biological Example	128
4.9.5	Computational Expense	128
4.9.6	Global Search Versus BSD	129
4.9.7	Applicability of BSD to More Complicated Structures	130
4.9.8	Combined Fast Orthogonal Search and BSD	131
4.9.9	Summary of Findings	131
4.10	Conclusions	132
5	Bootstrap Model Order Selection	134
5.1	Overview	134

5.2	Introduction	135
5.2.1	Model Order	135
5.2.2	Existing Methods	135
5.2.3	Proposed Approach	137
5.3	Model Order Selection	137
5.3.1	The Linear Model	137
5.3.2	Average Loss	139
5.3.3	Effects of Cross-Validation without Bootstrap	140
5.3.4	Error in Prediction	142
5.3.5	Model Order Selection Using Bootstrap	144
5.4	Simulations	146
5.4.1	Simple NARMAX Model	146
5.5	Simulated Biological Example	147
5.5.1	VOR Model	147
5.5.2	Simulation Protocol and Results	149
5.6	Discussion	152
5.6.1	Simulation Studies	152
5.6.2	Optimal m	152
5.6.3	Computational Requirements	152
5.6.4	Applicability of BMOS to More Complicated Structures	153
5.6.5	Future Work	154
5.6.6	Summary of Findings	154
5.7	Conclusions	155
6	Parameter Estimation of Hybrid Systems	156
6.1	Overview	156
6.2	Introduction	157
6.3	Multimode Model Formulation	159
6.4	Modified Extended Least-Squares	161
6.4.1	MELS Algorithm	165

6.5	VOR Model	165
6.6	Simulation	167
6.7	Experimental Data	171
6.7.1	Procedures	172
6.7.2	Perturbation	172
6.7.3	Apparatus	172
6.7.4	Data Processing	173
6.7.5	Data Analysis	174
6.7.6	Results	175
6.8	Discussion	177
6.8.1	MELS Algorithm	177
6.8.2	Simulation Study	177
6.8.3	Experimental Data	178
6.8.4	Future Work	179
6.8.5	Summary of Findings	179
6.9	Conclusions	180
7	Conclusions	181
7.1	Introduction	181
7.2	Statement of Original Contributions	181
7.3	Suggestions for Further Research	183
7.3.1	Identification of Ankle Dynamics	183
7.3.2	Structure Detection	185
7.3.3	Model Order Selection	186
7.3.4	Combined Structure Detection – Model Order Selection	186
7.3.5	Hybrid Systems	187
7.3.6	Application to Real Data	187
7.4	Discussion	188
A	A Matlab Toolbox for Nonlinear System Identification	190
A.1	Introduction	190

A.2	NARMAX Model Simulation	190
A.3	NARMAX Model Order Selection	192
A.4	NARMAX Structure Detection	193
A.5	NARMAX Parameter Estimation	194
A.6	M-files	195
Bibliography		219

List of Figures

2.1	Generalized identification problem. Redrawn from [158].	7
2.2	Least-squares system description.	36
2.3	Noisy system configuration.	38
3.1	Model structure assumed for identification of intrinsic and reflex contributions to overall ankle torque. Redrawn from [79].	52
3.2	Simulink model of parallel pathway ankle model in continuous-time. .	56
3.3	Second-order least-squares approximation to the half-wave rectifier used in simulations.	58
3.4	Left: Input to simulated parallel pathway model in continuous-time and NARMAX description model. Right: Output of simulated parallel pathway model in continuous-time and NARMAX description model.	59
3.5	Typical input-output sequence for parallel pathway ankle model. Upper: Position input. Lower: Measured torque output (sum of true system output and Gaussian, white, zero-mean noise sequence with 20 dB SNR).	61
3.6	Full NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=4,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. Theoretically expected parameter relationships: $b_8 = b_9 = -b_{10} = -b_{11}$; $b_{13} = 3b_{12}$; $b_{14} = 3b_{12}$; $b_{15} = b_{12}$; $b_{16} = -2b_{12}$; $b_{17} = -4b_{12}$; $b_{18} = -2b_{12}$. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	62

3.7	Full NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=12,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. Theoretically expected parameter relationships: $b_8 = b_9 = -b_{10} = -b_{11}$; $b_{13} = 3b_{12}$; $b_{14} = 3b_{12}$; $b_{15} = b_{12}$; $b_{16} = -2b_{12}$; $b_{17} = -4b_{12}$; $b_{18} = -2b_{12}$. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	64
3.8	Compressed NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=4,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	66
3.9	Cross-validation: Measured and predicted output for identified NARMAX ankle model with $N_v = 2,000$ and Gaussian, white, zero-mean output additive noise (20 dB SNR).	67
3.10	Continuous-time parameters of parallel pathway ankle model. Bandlimited, zero-mean, random input and NARMAX identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	70
3.11	Continuous-time parameters of parallel pathway ankle model. Bandlimited, zero-mean, random input and nonparametric identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	71
3.12	Typical position input and torque output recorded from simulation.	73

3.13	Continuous-time parameters of parallel pathway ankle model. PRBS input and NARMAX identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	74
3.14	Continuous-time parameters of parallel pathway ankle model. PRBS input and nonparametric identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	75
3.15	Typical position input and torque output recorded from simulation used to assess input noise sensitivity.	77
3.16	Compressed NARMAX model: PRBS input, Gaussian, white, zero-mean input noise and $N=7,000$. Ordinate: STD about mean. Abscissa: Input SNR= 70, 65, 60, 55 and 50 dB. Dashed line: True parameter value. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	78
3.17	Compressed NARMAX model: PRBS input, Gaussian, white, zero-mean input and output noise and $N=7,000$. Input SNR=60 dB. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	80
3.18	Continuous-time parameters of parallel pathway ankle model. PRBS input. Gaussian, white, zero-mean input and output noise, $N=7,000$ and NARMAX identification. Input SNR=60 dB. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10 and 5 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	81
3.19	Closed-loop system of ankle dynamics. $U(s)$: System input. $Y(s)$: System output. $F(s)$: Output of feedback loop. $V(s)$: Error signal. .	83

3.20	Simulink model of closed-loop ankle model in continuous-time.	84
3.21	Typical position, velocity and torque record from closed-loop simulation of ankle dynamics.	85
3.22	Typical recorded position input and torque output.	88
3.23	Cross-validation: Measured and predicted output for identified NARMAX ankle model for experimental data set with $N_v = 1,000$	88
3.24	Continuous-time parameters of parallel pathway ankle model. NARMAX identified values for 8 trials of human ankle dynamics. Dashed lines: Maximum-minimum physiological range.	90
3.25	Cross-validation. %NMSE vs. experimental trial.	91
4.1	Procedure for forming bootstrap parameter replications.	109
4.2	Idealized bootstrap-t interval as $B \rightarrow \infty$	109
4.3	Predicted structure of a simple NARMAX model using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection. . .	112
4.4	Error rate for highly over-parameterized system using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection. . .	114
4.5	Parameter distribution of spurious and true terms for a simple NARMAX model (Model 4.15) when the full model is postulated. Abscissa: Parameter mean. Ordinate: Probability of i th parameter. Vertical Line: Estimated mean for each approach.	115
4.6	Parameter distribution of true terms for a simple NARMAX model (Model 4.15) when the exact model is postulated. Abscissa: Parameter mean. Ordinate: Probability of i th parameter. Vertical Line: Parameter mean for each approach.	117

4.7	Simple NARMAX model (Model 4.15): Rate of model selection as a function of data length N and bootstrap replications $B = 100$. Gaussian, white, zero-mean noise added to output. Abscissa: $\pi = p^2/N = 1, 0.6, 0.4, 0.1$ (i.e., increasing N). Ordinate: Percent selection.	118
4.8	Complex NARMAX model (Model 4.16): Rate of model selection as a function of data length N and bootstrap replications $B = 100$. Gaussian, white, zero-mean noise added to output. Abscissa: $\pi = p^2/N = 1, 0.6, 0.4, 0.1$ (i.e., increasing N). Ordinate: Percent selection.	119
4.9	Simple NARMAX model (Model 4.15): Rate of model selection as a function of $\pi = p^2/N$ and bootstrap replications. Gaussian, white, zero-mean noise added to output. Abscissa: Bootstrap replications $B = 40, 80, 120, 160, 200$. Ordinate: Percent selection.	121
4.10	Error rate of a simple NARMAX model (Model 4.15) using the t-test, BSD and stepwise regression in the presence of bandlimited output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.	122
4.11	Error rate for highly over-parameterized model (Model 4.16) using the t-test, BSD and stepwise regression with 0 dB SNR of a bandlimited noise sequence. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.	123
4.12	A Hammerstein structure model of VOR.	123
4.13	Bandlimited Input. Error rate for theoretical model of VOR slow-phase using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output additive noise sequence with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.	125
5.1	Cross-validation selection rate versus data length for a simple NARMAX model. Abscissa: Data length, $N_e = N_v$. Ordinate: % correct selection.	142

5.2	Cross-validation selection rate versus SNR for a simple NARMAX model. Abscissa: SNR. Ordinate: % correct selection. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)	143
5.3	Global model structure of slow-phase component for the VOR system.	149
6.1	General Hammerstein model structure for a $\mathcal{M}$ mode hybrid system with output additive noise where $U(s)$ is the input, $Y(s)$ the true ($Y(s)$ is the selected $Y_m(s)$ driven by $X_m(s)$, $m = 1, 2, \dots, \mathcal{M}$) system output, $E(s)$ a Gaussian, white, zero-mean, noise sequence and $Z(s)$ the measured output.	157
6.2	Hammerstein model structure for a dual mode hybrid system.	159
6.3	Example of VOR input-output data segmentation for a dual mode system.	163
6.4	Typical plot of VOR output.	166
6.5	Simulation input-output data used for identification.	169
6.6	Experimental VOR data. Top: Head velocity input. Middle: Eye position output. Bottom: Eye velocity output.	173
6.7	Predicted eye velocity of experimental VOR data. (a): Measured eye velocity superimposed on top of predicted output due to classical approach (i.e., linear fit). (b): Measured eye velocity superimposed on top of predicted output due to cubic approach (i.e., cubic fit). (c): Measured eye velocity superimposed on top of predicted output due to our hybrid identification (i.e., MELS technique).	176
7.1	Low-pass IIR system preceded by a static nonlinearity (squarer).	188

List of Tables

3.1	Theoretical relationship of NARMAX parameters to continuous-time system coefficients for parallel pathway model of ankle dynamics. . .	55
3.2	Continuous-time coefficient values. I: inertia, B: viscosity, K: elasticity, ω : natural frequency, ζ : damping parameter, g : reflex stiffness gain and Δ : reflex delay.	57
3.3	Coefficient values of static nonlinearity. c_0 : DC term, c_1 : linear term, c_2 : squared term and T: sampling interval.	57
3.4	Theoretical relationship of compressed NARMAX model parameter set to continuous-time system coefficients.	65
3.5	Discrete to continuous-time relationships for parameters I, B, K, ω , ζ and g of the parallel pathway ankle model.	69
3.6	Continuous-time coefficients of ankle model simulated in closed-loop but identified assuming a parallel pathway model. Coefficients: Continuous-time parameters values used in simulations. Trail 1: Estimated continuous-time parameters using the nonparametric and NARMAX approach, respectively. Trail 2: Estimated continuous-time parameters using the nonparametric and NARMAX approach, respectively.	85
3.7	Continuous-time parameter ranges for parallel pathway model of ankle dynamics.	89

4.1	VOR slow-phase parameter values. Coefficient values of static nonlinearity: c_0 - DC term, c_1 - linear term, c_2 - squared term c_3 - cubic term. Dynamic system parameters: τ - time constant, g - dynamic gain and T - sampling interval.	124
5.1	MSE result of a typical trial for a simple NARMAX model with bootstrapping. First column: nonlinearity order, l . Second column: input lag, n_u . First row: output lag, n_y . Optimal order (minimum MSE) in bold. Result divided by 1000.	147
5.2	Monte-Carlo simulation results for simple NARMAX model. Left: Empirical probability (percentage) without bootstrapping. Right: Empirical probability (percentage) with bootstrapping, i.e., BMOS algorithm. Optimal order (maximum probability) in bold.	148
5.3	MSE result of a typical trial for the slow-phase component of the VOR system with bootstrapping. First column: nonlinearity order, l . Second column: input lag, n_u . First row: output lag, n_y . Optimal order (minimum MSE) in bold. Result divided by 1000.	150
5.4	Monte-Carlo simulation results for the slow-phase component of the VOR system. Left: Empirical probability (percentage) without bootstrapping. Right: Empirical probability (percentage) with bootstrapping, i.e., BMOS algorithm. Optimal order (maximum probability) in bold.	151
6.1	Data segmentation.	162
6.2	Discrete-time relationship of NARMAX model parameters to underlying continuous-time parameters.	168
6.3	Continuous-time coefficient values. τ_1 : slow-phase time-constant, τ_2 : fast-phase time-constant, K_1 : slow-phase gain (velocity gain = $K_1/\tau_1 = -0.628$) and K_2 : fast-phase gain (velocity gain = $K_2/\tau_2 = 0.222$). . .	168
6.4	Coefficient values of static nonlinearity. a : DC term, b : linear term, c : squared term, d cubic term and T : sampling interval.	169

6.5	Theoretical and estimated discrete-time coefficients of VOR sub-system 1 (slow-phase). θ Column: Theoretical parameter values. $\hat{\theta}_{ELS}$ Column: Estimated parameters using ELS. $\hat{\theta}_{MELS}$ Column: Estimated parameters using MELS. Values correspond to continuous-time slow-phase parameter value in Tables 6.3 & 6.4.	170
6.6	Theoretical and estimated discrete-time coefficients of VOR sub-system 2 (fast-phase). θ Column: Theoretical parameter values. $\hat{\theta}_{ELS}$ Column: Estimated parameters using ELS. $\hat{\theta}_{MELS}$ Column: Estimated parameters using MELS. Values correspond to continuous-time slow-phase parameter value in Tables 6.3 & 6.4.	171
6.7	Discrete to continuous-time relationships for parameters $p_{1,2}$ and $K_{1,2}b$ of the VOR model.	171
6.8	Continuous-time parameter estimates of $p_{1,2}$ and $K_{1,2}b$ of the VOR model.	172
6.9	Identified continuous-time parameters from experimental VOR data. a : DC term, b : linear term, c : squared term, d cubic term, τ_1 : slow-phase time constant and %NMSE percent normalized mean-squared-error.	175

List of Symbols

Symbol	Definition	Page
n	Index of sampled data point(s).	10
T	Sampling rate.	10
$y(n)$	True system output.	11
$u(n)$	Exogenous input.	11
$e(n)$	Innovation process.	11
q^{-1}	Backward shift operator.	12
$y(n) = F^l[\cdot] + e(n)$	The Kolmogorov-Gabor polynomial, i.e., NARMAX model.	15
$\delta(\cdot)$	Kronecker impulse function.	19
n_u	Maximum input lag.	20
n_y	Maximum output lag.	20
n_e	Maximum error lag.	20
l	Maximum nonlinearity order.	20
$O = [n_u \ n_y \ n_e \ l]$	System order for NARMAX models.	20
p	Number of parameters or candidate terms.	20

Symbol	Definition	Page
σ^2	Unknown noise variance.	25
$Y = X\theta + e$	General linear model.	36
$\hat{\theta}_{OLS}$	Ordinary least-squares estimate of θ .	36
$E[\cdot]$	Expected value.	37
$D[\cdot]$	Dispersion matrix.	37
$\hat{\sigma}^2$	Unbiased estimate of noise variance, σ^2 .	37
Z	Vector of measured outputs.	39
Ψ_{zu}	Input-output regressor matrix.	39
$\hat{\epsilon}$	Prediction errors.	40
$\Psi = [\Psi_{zu} \Psi_{zu\hat{\epsilon}} \Psi_{\hat{\epsilon}}]$	Extended regressor matrix.	41
$\hat{\theta}_{ELS}$	Extended least-squares estimate of θ .	41
$\hat{\theta}_{MLE}$	Maximum likelihood estimate of θ .	45
%NMSE	Percent normalized mean-squared-error.	59
P	Projection matrix.	104
$Z^*, \hat{\epsilon}^*, \hat{\theta}^*$	Bootstrap analog of $\hat{Z}, \hat{\epsilon}, \hat{\theta}$.	106
$d_\alpha(\cdot, \cdot)$	Mallows metric.	105
$\pi = p^2/N \rightarrow 0$	Bootstrap growth condition.	107
B	Number of bootstrap replications.	109
Z_e	Estimation set.	138
N_e	Length of estimation data.	138

Symbol	Definition	Page
Z_v	Validation set.	138
N_v	Length of validation data.	138
$L_{N_e}(O)$	Mean squared error (average loss).	139
$\ \cdot\ $	Euclidean norm on $\mathbb{R}^2$.	139
$\mathcal{D}$	Model set on $\mathbb{R}^{\infty \times \infty \times \infty \times \infty}$.	139
$\hat{\Gamma}_{N_v}(O)$	Average conditional expected loss in prediction.	139
O_0	Model order with smallest size on $\mathbb{R}^{\infty \times \infty \times \infty \times \infty}$.	140
$\hat{\Gamma}_{N_v,m}(\hat{O})$	Bootstrap prediction error estimator.	143
$\mathcal{M}$	Modes of operation.	157
U_{mi}, Z_{mi}	i th input-output segment of sub-system m .	162
$\mathcal{U}, \mathcal{Z}$	Concatenated input-output segments of m th sub-system.	162
$\Psi_{\mathcal{Z}\mathcal{U}}$	Concatenated matrix of sub-regressor matrices.	163
$\Phi = [\Psi \ \Psi_\delta]$	Modified extended regressor matrix.	164
$\hat{\theta}_{MELS}$	Modified Extended least-squares estimate of θ .	164

Chapter 1

Introduction

System identification is the process of developing or improving a mathematical representation of a physical system based on observed data. The study and analysis of systems is related to the development of new identification algorithms, or tools, for scientists. When these tools are developed with specific applications in mind, they may be difficult or impossible to apply to problems in other disciplines, and will, therefore, render the tools relevant to only an explicit problem or set of problems. Consequently, one of the fundamental goals of this thesis was to develop tools that are generalizable to a large set of problems that bridge many branches of research.

Over the past several decades, significant achievements have been made in several areas of nonparametric nonlinear system identification [167]. Robust techniques have been developed within different disciplines. However, many of these methods restrict the types of inputs that can be applied and the system structures that can be studied. In addition, nonparametric methods may represent nonlinear systems at the expense of introducing an excessive number of coefficients which are not readily linked to the underlying system.

Parametric representations of nonlinear systems typically contain a small number of coefficients which can be varied to alter the behavior of the equation and may be linked to the underlying system. In this thesis, we only consider a family of parametric models described by linear-in-the-parameters, linear and nonlinear, difference equations. This family of parametric models is popularly known as the NARMAX

(Nonlinear Autoregressive, Moving Average eXogenous) model structure. Only the discrete-time case is considered since most systems for identification purposes are represented in discrete-time.

Parametric identification, while theoretically attractive, has been difficult to apply since basic questions in this field have been left open. Specifically, these open questions have been: (1) how to estimate the order of the input-output and error lag and nonlinearity, *model order selection*, (2) how to select which parameters to include in the model, *structure detection* and (3) how to model and estimate parameters of “hybrid” or “multimode” systems, systems that can switch between various modes of operation. Hence, applications have only been possible in cases where a significant amount of *a priori* information has been available. In this thesis, we attempt to further the discipline of parametric nonlinear system identification by developing tools which address these questions and, therefore, allow our tools to be applied under liberal conditions.

We have designed our tools with little or no restrictions so that they may be applied to a wide range of problems and in various fields of study. They have been designed to allow the study of biological systems, where the systems being examined can be highly nonlinear and may switch between various modes of operation. Moreover, since the nature of experimental apparatus typically constrains the type of stimulus that can be applied to a system under study, these tools have been designed to ensure that the system is identifiable with the minimum requirement that the input be “persistently exciting”, i.e., that the system dynamics be persistently excited over the measurement time. Finally, measurements of system input-outputs are often corrupted by noise. Therefore, these algorithms have been designed with this restraint taken into consideration.

Although we have developed these tools with biological applications in mind, the overall goal of this work was not only applicability to biomedical engineering but to provide an expanded and improved set of tools for the identification of *both* linear and nonlinear systems that fall under the linear regression “umbrella”. We believe they will render parametric nonlinear identification a feasible tool for modeling un-

known (black-box) systems in many areas outside the biomedical realm. Some potential applications for these methods outside the biomedical field are, for example, efficient controller design, aircraft/spacecraft/robotic control and design, communications, analysis of economic trends, analysis of geophysical or ecological phenomena, etc.

1.1 Thesis Overview

Chapter 2 presents a general introduction to the topic of system identification. This is followed by a detailed review of recent system identification literature, with special focus on methods that may be applicable to linear regression models. Linear regression techniques are given special consideration since they are applicable to discrete-time linear and nonlinear systems that are linear-in-the-parameters, i.e., NARMAX models.

In chapter 3 we commence our study of the NARMAX model class by demonstrating the usefulness of NARMAX models for biological modeling. A parallel pathway model of ankle dynamics is theoretically analyzed and its NARMAX representation is derived.

Structure detection is a crucial procedure for estimating a parsimonious system description. Chapter 4 presents an algorithm for determining model structure, based on least-squares and bootstrap theory.

Chapter 5 continues the application of bootstrap to nonlinear identification and presents an algorithm for estimation of model order for linear and nonlinear systems.

Chapter 6 presents a new algorithm for the identification of linear and nonlinear hybrid (multimode) systems. Systems of this type may be quite common in physiology since, in many cases, the biology is known to abruptly switch control strategies based on the input stimulus.

Finally, in Chapter 7 we summarize the contributions made in this thesis, and offer suggestions for further developments and improvements. We finish the chapter by discussing further potential applications for these techniques.

Chapter 2

Literature Review

2.1 Introduction

In this review, our primary objective will be to describe techniques that are suitable for building models described by linear-in-the-parameters, nonlinear, difference equations. Our description of this model structure will reveal several deficiencies in existing methodology for the identification of these types of systems. The complexities associated with such model descriptions limit the widespread applicability of existing methods unless considerable *a priori* knowledge of the underlying system is available.

The full identification problem (as posed by Verhaegen and Dewilde, 1992) is examined in detail. Two broad classes of identification are introduced: (i) nonparametric and (ii) parametric. The shortcomings and strengths of each approach is reviewed. The model structure(s) used for nonparametric identification is/are reviewed in the context of the benefits and difficulties associated with nonparametric identification. The rationale for using parametric identification is stated and various types of model structures that fall within the parametric family are reviewed. Then the NARMAX (Nonlinear Autoregressive, Moving Average eXogenous) model structure is introduced and discussed. The steps involved in parametric (NARMAX) identification are outlined and a review of the current literature is presented, focusing on the strengths and weaknesses of each approach.

2.2 Modeling Techniques

A review of the techniques used to model complex systems exposes two broad classes: *a priori* or morphological modeling (also known as physical models, “first principles” models or white-box models) and *a posteriori* or black-box modeling (commonly known as system identification) [101]. The complexities associated with a system may limit the type of experiments that can be performed and will determine the choice of modeling techniques that can be considered. However, a minimum level of complexity is required by the nature of any system. Experimental limitations on the type of data that can be obtained also impose an upper limit on the complexity of the model that can be justified [101, 167]. With these objectives and limitations in mind we will present a detailed discussion of the advantages and disadvantages of morphological and black-box modeling techniques.

2.2.1 Morphological Modeling

The objective of morphological, or *a priori*, modeling procedures is to derive a system description from basic physical laws only. In this approach, the system is decomposed into subsystems whose properties are well established from previous experience (i.e., knowledge of both the system structure and the function of the subsystems) that have their roots in earlier empirical work [101]. The subsystems are then joined mathematically to produce a model of the whole system. Models of this type are often known as “morphological” since the individual elements and interconnections are often related directly to the structure of the system being modeled. These models have the advantage that they possess a clear physical interpretation. However, systems that are modeled using morphological techniques often embody many parameters that must be determined experimentally.

2.2.2 Black-Box Modeling

If nothing at all is known or assumed about the physical structure of a system, the modeling procedure is known as *a posteriori* or black-box modeling [101]. In black-box

modeling, the system inputs and outputs are recorded and then analyzed to obtain a model description. This approach is often referred to as black-box modeling since the resulting mathematical description (model) of the system is simply a “black-box” that mimics the behavior of the system. This type of modeling provides a relationship between the system inputs and outputs. However, it generally provides little structural or functional information about the system or its components.

In practice, neither of these two extreme approaches are particularly fruitful in isolation. In order to model complex processes, such as those found in biology, the two approaches have to be combined. Therefore, in practice, morphological modeling is combined with system identification methods. The theoretical framework for doing this systematically is known as “grey-box” modeling.

Black-box models provide “simpler” models. Analysis of input-output data can provide useful process insights that can be used in subsequent development or refinement of physical models. In particular, all morphological models are based on assumptions (e.g., these effects are important and those are negligible) which may be incorrect [116, 119]. Black-box models may help uncover such surprises. However, depending on the character of the system, one of the two modeling activities, i.e., morphological modeling or parameter estimation (black-box modeling), may dominate the overall modeling effort.

Generally, there are two uses for models obtained using system identification approaches. In the design of control systems, models are required to predict the plant's response to its input in order to design an effective controller. In this case, it is usually desirable to have the simplest possible model that describes the dynamics of the plant to be controlled. However, if the objective of the identification is to gain insight into the function of the underlying system, it is necessary to extract the maximum amount of information from the input-output data. In general, models identified for insight are often more complex than those used for control.

2.3 The Identification Problem

Consider the general identification problem posed by Verhaegen and Dewilde [158], shown in Figure 2.1. The “system” is defined as everything within the dashed box

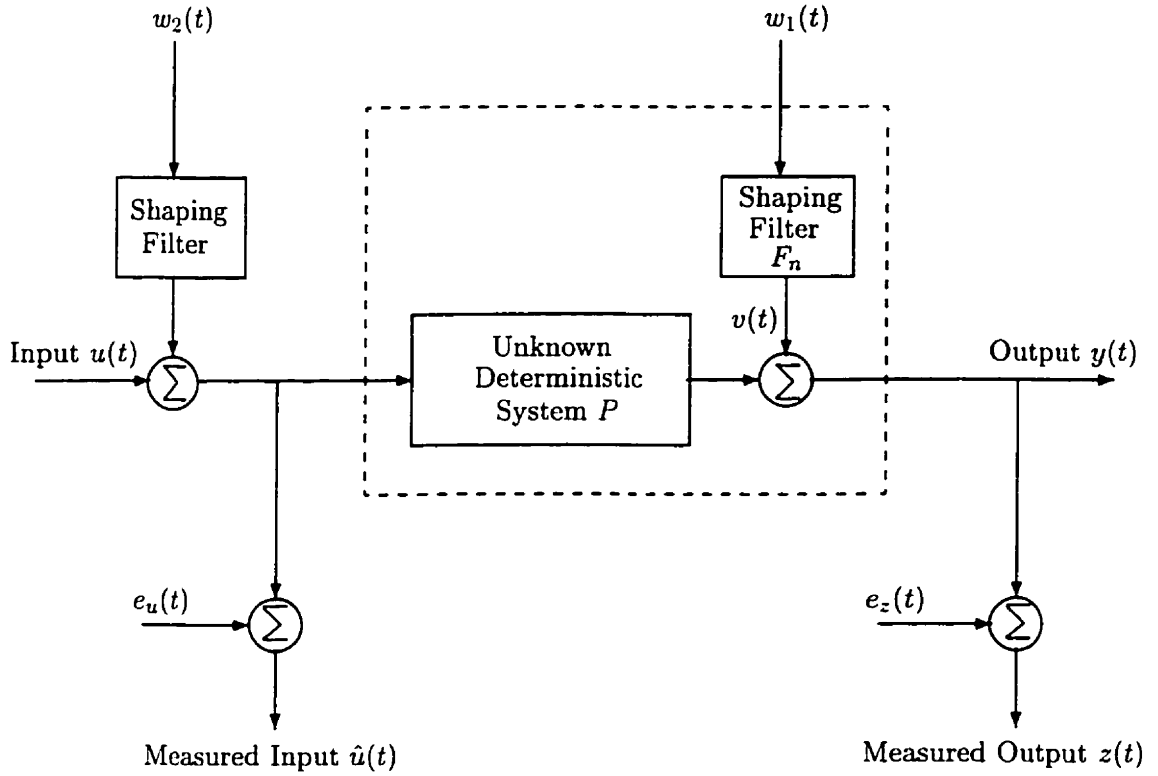


Figure 2.1: Generalized identification problem. Redrawn from [158].

and consists of two parts: (i) stochastic and (ii) deterministic. The stochastic part is driven by a white noise process, $w_1(t)$, which is not available to the experimenter. The deterministic part, the system to be characterized, is driven by the sum of a controlled input, $u(t)$, and a filtered, inaccessible white noise process, $w_2(t)$. It is assumed that the experimenter has control over $u(t)$ and is able to access a noise-corrupted version of the input signal, $\hat{u}(t)$. The noise-free output, $y(t)$, is the “true” system output which is the sum of the stochastic and deterministic parts of the system. However, the experimenter only has access to a noise-corrupted version of the output signal, $z(t)$.

This leads to several identification problems.

1. *Identification of the stochastic or noise model, F_n .* Here, the focus is placed on the relationship between $w_1(t)$ and $z(t)$, given observations of only the system output, $z(t)$. The input signal, $u(t)$, is assumed to be zero or constant. This type of identification is commonly known as time series analysis and has applications for the study of economics systems, analysis of geophysical or astronomical phenomena, analysis of biological data (e.g., heart rate, EEG), etc., where the inputs are not available to the experimenter, or where it is unclear which signals are inputs and which are outputs.
2. *Identification of the deterministic model, P .* This problem consists of finding a relationship between $u(t)$ and $y(t)$, assuming that the process noise, $w_1(t)$, is zero. Both the input and output may still be corrupted by observation noise, $e_u(t)$ and $e_z(t)$ respectively. However, it is commonly assumed that $\hat{u}(t)$ is recorded with negligible error, i.e., $e_u(t) = 0$. The identification of deterministic systems is generally pursued when the objective is to gain insight into the functioning of a system. This is the problem that will be pursued in this thesis.
3. *Identification of the stochastic and deterministic (complete) models.* When both the input and output signals are available for identification, the goal may be to estimate both P and F_n , the deterministic and noise models. This problem formulation is used when accurate predictions are desired, such as in the design of model-based control systems for aircraft, spacecraft or robotics.

This thesis concerns itself with identifying the deterministic model, P , and we assume that the input is recorded with negligible error ($e_u(t) = 0$). For notational simplicity, we will henceforth use 'e' to denote output additive noise ($e_z(t)$) unless explicitly stated otherwise.

2.4 Classes of System Identification

There are two broad classes of techniques that can be pursued to accomplish the task of system identification: (i) nonparametric and (ii) parametric methods.

2.4.1 Nonparametric System Identification

Causal, time-invariant linear and nonlinear systems form some of the most important classes of dynamical systems used in practice. Although they represent idealizations of the processes encountered in practice, the approximations involved are often justified and lead to good results in many cases [101].

A system is considered to be causal if the output at any time depends on the input up to that time only [101, 138]. A system is said to be time-invariant if its response to a certain input signal does not depend on absolute time. Moreover, a system is linear if its output response to a linear combination of inputs is the same linear combination of the output responses of the individual inputs.

The response to a linear, time-invariant, causal system is well known to be described by its impulse response as [13, 101, 138]:

$$y(t) = \int_{\tau=0}^{\infty} h(\tau)u(t - \tau)d\tau. \quad (2.1)$$

If $\{h(\tau)\}_{\tau=0}^{\infty}$ and $u(\nu)$ are known for $\nu \leq t$, the corresponding output, $y(\nu)$, $\nu \leq t$, can be computed for any input. Therefore, the impulse response is a complete characterization of a causal, linear, time-invariant system.

In any practical experimental situation, the data available to the experimenter will be finite; therefore, Equation 2.1 cannot be applied directly. The finite impulse response function (FIR) has been widely used for modeling linear time-invariant systems. The continuous-time output of this model is represented as the convolution integral:

$$y(t) = \int_0^M h(\tau)u(t - \tau)d\tau \quad (2.2)$$

where $h(\tau)$ is the impulse response, M is the memory length, and the lower bound of the integration is zero to represent a causal system. The FIR can be represented in discrete-time as

$$y(n) = T \sum_{\tau=0}^{M-1} h(\tau)u(n - \tau) \quad (2.3)$$

where $n = 1, 2, \dots, N$, the sampled data point index; τ , the lag, are integers; and the sampling rate, T , is often assumed to be 1 for notational simplicity. The input, u , is often assumed to be white and the lower limit of the summation is zero to represent a causal system. This system description is known as nonparametric because it is a numeric representation of the system's impulse response or kernel. However, since many systems are inherently nonlinear the rich behavior of their dynamics cannot be fully described using linear techniques.

Classically, the model structure used in nonlinear system identification has been the functional series expansions of Volterra or related techniques [103]. The discrete Volterra series expansion is commonly shown as [166]:

$$y(n) = \sum_{i=0}^l \left\{ \sum_{\tau_1=0}^{M-1} \dots \sum_{\tau_i=0}^{M-1} k_i(\tau_1, \dots, \tau_i) u(n - \tau_1) \dots u(n - \tau_i) \right\}. \quad (2.4)$$

In this series l represents the model order and M the memory length. These types of descriptions represent a wide class of nonlinear, fading memory systems [29, 167]. One such representation of this class of systems is the Wiener-Bose model which consists of a bank of linear filters whose outputs are combined and transformed by a multiple input polynomial. As such, they have been used successfully for physiological modeling for a number of years. Many of these techniques are robust in the presence of noise and require few *a priori* assumptions. Recently some of these methods also allow the use of non-white inputs [130, 167].

Although nonparametric methods can be used to represent many classes of nonlinear systems, they do so at the expense of introducing an excessive number of unknown coefficients which must be estimated. Most expansions map the past inputs into the present output and so require a very large number of coefficients to characterize the process. For example, even a simple quadratic nonlinearity in cascade with a first-order linear dynamic system could easily require 400-500 coefficients to specify the first and second-order Volterra kernels [22]. Therefore, the resulting system description is not represented concisely and may be redundant. Alternative system descriptions, such as block-structured models [17, 83, 167], alleviate some of these

difficulties provided the system under study belongs to the relevant class of models. Moreover, the parameters are not readily linked to the underlying system, except in special cases where much *a priori* knowledge of the system has been assumed (e.g., parallel cascade structure of ankle dynamics [75, 77, 78, 79, 167]).

2.4.2 Parametric System Identification

In recent years parametric identification methods have been developed for use in the design of better control systems. Parametric models have some advantages in applications. First, they are easier to understand and interpret. Second, they can simplify forecasts (e.g., obtaining forecast intervals). Third, model comparison in a parametric context (i.e., parameter estimates, model order and model structure) has been well studied; so the difficulty of model comparison encountered using nonparametric tools can be avoided [36].

A parametric model consists of a set of differential or difference equations describing the system dynamics. Such equations usually contain a “small” number of parameters which can be varied to alter the behavior of the equation. Here we will only discuss the discrete-time case since most systems for identification purposes are represented in discrete-time.

2.4.2.1 Parametric Representations

For linear systems, the relationship between input-output and noise can be written as a linear difference equation [61, 68, 101]:

$$\begin{aligned} y(n) = & - a_1 y(n-1) - \cdots - a_{n_y} y(n-n_y) \\ & + b_1 u(n-1) + b_2 u(n-2) + \cdots + b_{n_u} u(n-n_u) \\ & + e(n) + c_1 e(n-1) + \cdots + c_{n_e} e(n-n_e). \end{aligned} \quad (2.5)$$

This is known as the AutoRegressive, Moving Average eXogenous (ARMAX) model. In this model structure the current output $y(n)$ depends on an exogenous input, $u(n)$, an innovation process, $e(n)$ ($e_z(t)$ in Figure 2.1), and past values of the output. This

structure can be represented more compactly as

$$A(q)y(n) = B(q)u(n) + C(q)e(n) \quad \text{or} \quad (2.6)$$

$$y(n) = G(q)u(n) + H(q)e(n) \quad \text{where}$$

$$G(q) = \frac{B(q)}{A(q)}, \quad H(q) = \frac{C(q)}{A(q)}$$

where $A(q) = 1 + a_1q^{-1} + \dots + a_{n_y}q^{-n_y}$, q^{-1} is the backward shift operator and the a 's are the parameters of the output. The ARMAX model structure is a widely recognized tool in control and econometrics for both system description and control design [101]. This system representation has several special cases [61, 68, 101].

2.4.2.1.1 The Autoregressive (AR) Model

$$A(q)y(n) = e(n) \quad (2.7)$$

In this model representation the output depends on the unknown current disturbance as well as the n_y previous values of the output.

2.4.2.1.2 The Moving Average (MA) Model

$$y(n) = C(q)e(n) \quad (2.8)$$

Here, the output depends on the previous n_e values of the disturbance $e(n)$.

2.4.2.1.3 The Autoregressive Moving Average (ARMA) Model A combination of the previous two yields the Autoregressive Moving Average (ARMA) model:

$$A(q)y(n) = C(q)e(n). \quad (2.9)$$

2.4.2.1.4 The Autoregressive Exogenous Input (ARX) Model If an accessible input, $u(n)$, is added to the AR model the result is an Autoregressive exogenous

input (ARX) model:

$$A(q)y(n) = B(q)u(n) + e(n). \quad (2.10)$$

In this structure, the output depends on the current disturbance as well as n_u previous values of the input and n_y previous values of the output.

2.4.2.1.5 The Finite Impulse Response (FIR) Model A special case of the ARX model structure is when there is no disturbance input. This is known as the finite impulse response model:

$$y(n) = B(q)u(n). \quad (2.11)$$

For this model type the output depends only on the previous values of the exogenous input. This model structure forms the basis of many so-called nonparametric identification schemes.

Once a model structure has been determined, the unknown parameters can be estimated by using techniques that optimize the vector of parameters to some cost function.

2.4.2.1.6 State Space Model The state space model is another class of parametric model. The generalized state space equations for a causal linear time-invariant system are [156]:

$$\begin{aligned} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) + Du(n). \end{aligned} \quad (2.12)$$

In this equation $x(n)$ represents the finite dimensional state vector with the states going forward in time, $y(n)$ is the output vector and $u(n)$ is the input vector. The matrices A , B , C and D are the system quadruple, where A contains the system modes. Traditional methods for identifying systems of this type were developed by Kalman *et al.* [68].

More recently, these types of systems have been identified using subspace methods [156, 157, 158, 159]. The input-output data is arranged into Hankel matrices. It is possible to relate these Hankel matrices to the state equation in terms of the extended observability matrix and Markov parameters. Using RQ factorization and singular value decomposition (SVD), the system quadruple can be solved for, and, as a by-product model order selection is performed. However, selecting the model order using this method requires interpretation of singular value plots which often requires expertise beyond that of the average user from a non-mathematical discipline. It is often quite difficult even for the experienced user, when the signal-to-noise ratio (SNR) is low.

2.4.2.2 Summary

Many of these system descriptions have the disadvantage that they require *a priori* assumptions regarding the system order. However, parametric methods have the advantage of giving a concise description to the underlying system, since they estimate the unknown parameters of the analytic expression describing the system, and may yield results that can be related directly to the system structure. Since these models are linear-in-the-parameters they can be estimated using parameter estimation algorithms which are not dependent upon specialized input signals.

While most systems encountered in practice are nonlinear, for control purposes linear techniques are adequate because the systems are often approximately linear over the range under consideration or can be approximated linearly by fixing them about an operating point. Therefore, controller design can be accomplished using less complicated methods. However, to gain insight into the underlying structure of the system and to obtain an efficient global system description, parametric nonlinear identification is necessary. Some important classes of models which fall into this category include the Hammerstein model, Wiener model, polynomial state models [64], and classes of nonlinear difference equations [22, 98, 99]. Since the Hammerstein model can only represent a small class of systems, and expansions based on system states imply that all the states can be measured, only nonlinear difference equation

models are considered here.

2.5 The NARMAX Model

The Kolmogorov-Gabor polynomials have been well-known in control engineering for many years [51, 155]. However, these equations have recently been popularized by Billings and co-workers [98, 99] for use in identification, modeling, and control. This general parametric structure is commonly known as NARMAX (Nonlinear Autoregressive, Moving Average eXogenous). This so-called NARMAX structure can be used for the identification of both the stochastic and deterministic components of a system. The input-output relationship of many nonlinear dynamic systems can be written in the NARMAX form as the nonlinear difference equation

$$y(n) = F^l[y(n-1), \dots, y(n-n_y), u(n), \dots, u(n-n_u), \\ e(n-1), \dots, e(n-n_e)] + e(n) \quad (2.13)$$

where F is a nonlinear mapping, u is the “controlled” (i.e., exogenous) input, y is the output, and e is a zero-mean additive noise sequence (i.e., innovation; $e_z(t)$ in Figure 2.1). Note that in Equation 2.13 to model the stochastic component of a system e is replaced with w_1 (i.e., uncontrolled input). This nonlinear difference equation model or NARMAX model, may include a variety of nonlinear terms, such as terms raised to an integer power (e.g., $u^2(n-3)$), products of past inputs (e.g., $u(n)u(n-1)$), past outputs (e.g., $y(n-1)y(n-2)$), or cross-terms (e.g., $u^2(n-1)y(n-2)$). This system description encompasses most forms of nonlinear difference equations that are linear-in-the-parameters [92]. Since the NARMAX model is linear in its parameters, linear regression can be used for parameter estimation [10, 22, 61, 133].

2.5.1 NARMAX Representations of Nonlinear Systems

An important question in linear and nonlinear modeling and identification is how to describe the input-output relationship of a dynamic system. The input-output

relationship should be straightforward, should provide an adequate approximation to a large class of systems, and have minimal computational cost. It is well known that, for linear discrete-time systems, linear difference equation models exist that involve a fixed and finite number of calculations at each stage, if the Hankel matrix of the system has finite rank. This often provides system descriptions that are more concise than the impulse response function [37, 61, 101]. A similar situation exists for discrete-time nonlinear systems [98, 99].

The NARMAX model structure (Equation 2.13) is a general and natural representation for many discrete-time, time-invariant, nonlinear systems and provides a unified representation for a wide class of nonlinear systems as special cases [37]. This has obvious advantages over functional series representations such as the Wiener or Volterra series which suffer from excessive parameterization [37, 131]. Leontaritis & Billings [98, 99] have proved that a nonlinear, discrete-time, time-invariant system can always be represented by model 2.13 in a region around an equilibrium point subject to two sufficient conditions [37, 98, 99]:

1. The response function f of the system is finitely realizable and
2. A linearized model exists if the system is operated close to the equilibrium point.

Condition (1) simply excludes distributed parameter systems since the power any input-output term is raised to is not always some constant integer independent of n [99]. Condition (2) implies that if the system is perturbed with a small amplitude input in the linear region around the equilibrium point, a linearized model of the system exists. For notational simplicity, the discussions presented here are in context of the single-input single-output case. However, most of the discussions are valid for multiple input-output systems [20, 98].

2.5.1.1 Nonlinear Polynomial Models

In many situations it is reasonable to believe that higher-order polynomial functions will, in general, yield better approximations to the system under study than a linear

model (i.e., a polynomial model of degree 1 in $\{y(n-1), \dots, y(n-n_y), u(n), \dots, u(n-n_u)\}$). Practical identification of several biomedical systems has shown that many can be adequately modeled by polynomial NARMAX models [90, 91, 92, 93, 112, 137].

A general polynomial input-output model takes the form

$$\begin{aligned}
 y(n) = & a_0 + \sum_{i=1}^{n_y} a_i y(n-i) + \sum_{i=0}^{n_u} b_i u(n-i) \\
 & + \sum_{i=1}^{n_y} \sum_{j=1}^{n_y} a_{ij} y(n-i) y(n-j) + \sum_{i=1}^{n_y} \sum_{j=0}^{n_u} c_{ij} y(n-i) u(n-j) \\
 & + \sum_{i=0}^{n_u} \sum_{j=0}^{n_u} b_{ij} u(n-i) u(n-j) + \dots + \text{higher-order terms up to degree } l
 \end{aligned} \tag{2.14}$$

where the a 's, b 's and c 's are unknown system coefficients. The polynomial form in Equation 2.14 can be concisely expressed as Equation 2.13. Therefore, difference equation models that are linear-in-the-parameters are naturally represented by the general NARMAX formulation (model 2.13) and are a special case of it.

2.5.1.2 Bilinear Models

Bilinear system theory has been widely studied in the context of continuous-time systems, e.g., distillation columns, nuclear and thermal control processes [37, 64]. Bilinear systems are quite common continuous-time systems since any continuous, causal functional can be approximated arbitrarily well by a bilinear system within any bounded time interval [37, 52].

A general bilinear input-output model has the form

$$\begin{aligned}
 y(n) = & a_0 + \sum_{i=1}^{n_y} a_i y(n-i) + \sum_{i=1}^{n_u} b_i u(n-i) \\
 & + \sum_{i=1}^{n_y} \sum_{j=1}^{n_u} c_{ij} y(n-i) u(n-j)
 \end{aligned} \tag{2.15}$$

which is a special case of the NARMAX model 2.13.

2.5.1.3 Rational and Output-affine Models

The response function, f , of a system is said to be a polynomial response function if for each n , f_n is a polynomial of finite degree in all variables, although this degree may tend to ∞ as $n \rightarrow \infty$. A polynomial response function f is said to be bounded if for all n the maximum power any individual variable is raised to in f_n is less than a certain bound [37]. The realization of polynomial response functions has been investigated in detail by Sontag [146].

It is known that a polynomial response function is finitely realizable if, and only if, it satisfies the rational difference equation [146]

$$\begin{aligned} a[y(n-1), \dots, y(n-r), u(n-1), \dots, u(n-r)]y(n) = \\ b[y(n-1), \dots, y(n-r), u(n-1), \dots, u(n-r)] \end{aligned} \quad (2.16)$$

or

$$y(n) = \frac{b[y(n-1), \dots, y(n-r), u(n-1), \dots, u(n-r)]}{a[y(n-1), \dots, y(n-r), u(n-1), \dots, u(n-r)]} \quad (2.17)$$

where r is the order of the system, $a[\cdot]$ and $b[\cdot]$ are polynomials of finite degree.

In addition, Sontag [146] showed that f is a finitely realizable and bounded polynomial response function if, and only if, it satisfies an affine difference equation

$$\begin{aligned} a_0[u(n-1), \dots, u(n-r)]y(n) = \\ \sum_{i=1}^r a_i[u(n-1), \dots, u(n-r)]y(n-i) + a_{r+1}[u(n-1), \dots, u(n-r)] \end{aligned} \quad (2.18)$$

or

$$\begin{aligned} y(n) = \sum_{i=1}^r \frac{a_i[u(n-1), \dots, u(n-r)]}{a_0[u(n-1), \dots, u(n-r)]} y(n-i) \\ + \frac{a_{r+1}[u(n-1), \dots, u(n-r)]}{a_0[u(n-1), \dots, u(n-r)]} \end{aligned} \quad (2.19)$$

where $a_i[\cdot]$, $i = 0, 1, \dots, r+1$ are polynomials of finite degree.

The rational model (Equation 2.17) and the output-affine model (Equation 2.19)

are globally valid [37]. The response function of the system is, however, restricted to a polynomial response [37]. By choosing the particular forms in Equations 2.17 and 2.19 for the nonlinear mapping, $F(\cdot)$, it is easily seen that the rational and output-affine models are a special case of the general NARMAX model 2.13. However, identification and modeling issues concerning the rational and output-affine models is not addressed in this thesis.

2.5.1.4 Nonzero-initial-state Models

Although, the derivation of the NARMAX model (Equation 2.13) is based on zero-initial-state response [98, 99], the results can be extended to the nonzero-initial-state case by including lagged impulse values to account for nonzero-initial-states or discontinuities as [37]

$$\begin{aligned} y_m(n) = & F^l[y_m(n-1), \dots, y_m(n-n_y), u_m(n), \dots, u_m(n-n_u), \\ & \delta_m(n), \dots, \delta_m(n-n_\delta), e_m(n-1), \dots, e_m(n-n_e)] + e_m(n) \\ & \text{for } m = 1, 2, \dots, \mathcal{M} \end{aligned} \quad (2.20)$$

where $\mathcal{M}$ represents the modes of operation, u_m , y_m , e_m are as defined previously, and δ_m are Kronecker impulse functions, i.e., initial conditions. This makes the NARMAX model structure well suited for modeling nonlinear hybrid or multimode systems [91].

2.6 Current Methodology

Identifying a NARMAX model has four stages: (1) *model order selection*, estimating the maximum order of the input-output and error lags and nonlinearity order, (2) *structure detection*, selecting which parameters to include in the model, (3) *parameter estimation*, determining values for these parameters and (4) *model validation*, detecting terms in the residuals which if ignored will cause bias in the parameter estimates. These four topics encompass a wide range of literature; therefore, only methods relevant to polynomial NARMAX model identification are discussed here.

2.6.1 Model Order Selection

Many parametric methods require *a priori* assumptions about the system order. An ideal parametric method for system identification would estimate both the system order and the parameters.

The current literature offers no commonly accepted way to define NARMAX order. Therefore, we define the system order for NARMAX models as an ordered tuple

$$O \triangleq [n_u, n_y, n_e, l] \quad (2.21)$$

where n_u is the maximum lag on the input, n_y the maximum lag on the output, n_e the maximum lag on the error and l is the maximum nonlinearity order.

The maximum number of NARMAX parameters, p , is related to the model order, O , as

$$\begin{aligned} p &= \sum_{i=1}^l p_i; \quad \text{where } l \text{ is nonlinearity order} \\ p_i &= \frac{p_{i-1}(n_y + n_u + n_e + i - 1)}{i}, \quad p_0 = 1. \end{aligned} \quad (2.22)$$

We define the maximum number of terms, p , as the number of “candidate” terms to be initially considered for identification, i.e., the number of terms in the “full” model. Note that the number of candidate terms can be very large for NARMAX models, possibly resulting in an over-parameterized full model description.

2.6.1.1 Correlation Method

Correlation functions have been widely used for estimating model order of linear FIR systems [101, 140]. The cross-correlation function is defined by

$$R_{uy}(\tau) = E[u(n - \tau)y(n)]. \quad (2.23)$$

The cross-correlation function is rarely used since its value depends on the mean and variance of $u(n)$ and $y(n)$.

In practice the cross-covariance function is used and is defined by

$$C_{uy}(\tau) = E[(u(n - \tau) - \mu_u)(y(n) - \mu_y)] \quad (2.24)$$

where μ_u and μ_y are the mean of $u(n)$ and $y(n)$, respectively. Note, if $\mu_u = 0$ or $\mu_y = 0$ then the cross-covariance and the cross-correlation functions will be the same. Often the cross-covariance function is referred to (incorrectly) as the cross-correlation function.

The cross-correlation coefficient function is defined by

$$\phi_{uy}(\tau) = \frac{C_{uy}(\tau)}{\sqrt{C_{uu}(0)C_{yy}(0)}}; \quad -1 \leq \phi_{uy}(\tau) \leq 1, \quad \forall \tau \quad (2.25)$$

where the auto-covariance function at lag zero (e.g., $C_{uu}(0)$) is equal to the variance of the signal, i.e., $C_{uu}(0) = \sigma_u^2$, $C_{yy}(0) = \sigma_y^2$. The cross-correlation coefficient function may be thought of as the “normalized” cross-correlation function because its value is unaffected by either the mean or standard deviation. Hence, signals of different amplitudes can be compared easily.

The expressions shown in Equations 2.23 - 2.25 are generally known as “first-order” correlations. Traditional methods for model order selection, based on first-order correlations, generally fail for nonlinear systems due to a common problem [12, 153]. For Gaussian input data, the cross-correlation of any squared input-output terms will be zero [12, 13]. Therefore, the lag associated with even-order nonlinear terms cannot be determined using first-order correlations; hence “higher-order statistics”, e.g., second-order correlations, must be used [12, 154].

The second-order cross-correlation function is defined by

$$R_{uuy}(\tau_1, \tau_2) = E[u(n - \tau_1)u(n - \tau_2)y(n)]. \quad (2.26)$$

The second-order cross-covariance function is defined by

$$C_{uuy}(\tau_1, \tau_2) = E[(u(n - \tau_1) - \mu_u)(u(n - \tau_2) - \mu_u)(y(n) - \mu_y)]. \quad (2.27)$$

The second-order cross-correlation coefficient function is defined by

$$\phi_{uuy}(\tau_1, \tau_2) = \frac{C_{uuy}(\tau_1, \tau_2)}{\sqrt{C_{uu}(0)C_{uu}(0)C_{yy}(0)}}; \quad -1 \leq \phi_{uuy}(\tau_1, \tau_2) \leq 1, \quad \forall \tau_1, \tau_2. \quad (2.28)$$

In general, estimating model order for infinite impulse response (IIR) systems, such as the NARMAX model, even using higher-order correlations, also fails because the system “theoretically” could have infinite memory. For example, consider a system described by order $O = [1, 1, 1, 2]$:

$$y(n) = \theta_1 y(n-1) + \theta_2 u(n) + \theta_3 u^2(n-1) \quad (2.29)$$

$$y(n-1) = \theta_1 y(n-2) + \theta_2 u(n-1) + \theta_3 u^2(n-2) \quad (2.30)$$

$$y(n-2) = \theta_1 y(n-3) + \theta_2 u(n-2) + \theta_3 u^2(n-3) \quad (2.31)$$

$\vdots$

where we have omitted the lagged noise terms for notational simplicity. Substituting Equation 2.30 into 2.29 yields

$$\begin{aligned} y(n) &= \theta_1 [\theta_1 y(n-2) + \theta_2 u(n-1) + \theta_3 u^2(n-2)] \\ &+ \theta_2 u(n) + \theta_3 u^2(n-1) \end{aligned} \quad (2.32)$$

and substituting Equation 2.31 into 2.32 yields

$$\begin{aligned} y(n) &= \theta_1^2 [\theta_1 y(n-3) + \theta_2 u(n-2) + \theta_3 u^2(n-3)] \\ &+ \theta_1 \theta_2 u(n-1) + \theta_1 \theta_3 u^2(n-2) + \theta_2 u(n) + \theta_3 u^2(n-1). \end{aligned} \quad (2.33)$$

Note that it is possible to re-express the current output in terms of subsequent lagged input-output values by expanding Equation 2.31 further, then substituting the resulting expression for $y(n-3)$ into Equation 2.33. Theoretically, this expansion can go on to infinity or practically to the data length, N .

By substituting the right side of Equation 2.33 into Equation 2.28, it is readily seen that the input lag order (n_u) is over-estimated. This is because the recursive expression for $y(n)$ seen in Equation 2.33 contains lag orders that exceed the maximum

present in the true system description, given by Equation 2.29. In addition, the system will have non-zero input lags for multiples of the system lags, making it impossible to estimate maximum lag order using correlations. Furthermore, it is impossible to estimate the output and error lag (n_y and n_e). Therefore, correlation based techniques are limited to system structures that are described by FIR models. Although it is possible to equivalently describe many systems as either FIR or IIR, an IIR description is generally more efficient.

Recently, Tungnait [153, 154] proposed a method for order selection based on model validation and “higher-order statistics”. This method is closely related to correlation techniques and thus suffers from similar problems mentioned above. To date, applications have only been shown for linear AR or ARMA models.

2.6.1.2 Error in Cross-Validation

There are many methods for estimating model order in linear system identification, a common method is cross-validation [45, 101]

$$\hat{\Gamma}_N(O) = E \frac{\|Z - \Psi_O \hat{\theta}_O\|^2}{N} \quad (2.34)$$

where Z is a $N \times 1$ vector of measured outputs, Ψ_O is a $N \times p$ matrix of regressors and $\hat{\theta}_O$ is the $p \times 1$ vector of estimated parameters for a given order O . In the limit, the cross-validation error in fit will be minimized when the model is of the correct order, i.e., when the regressor matrix contains all the appropriate terms. With finite data lengths, however, statistical errors will often lead to inconsistent or inaccurate estimates of model order for particular realizations [56].

2.6.1.3 Reduction in Residual Sums of Squares

Another commonly used technique in time-series analysis (Equation 2.9) is to examine the reduction in the residual sums of squares (RSS) to determine model order. This technique tests two models, to assess which one can be justified on the basis of the reduction in RSS.

Specifically, this relies on calculating an F-ratio of the RSS between the current and past model, which are of different orders [34]. This is a test of the hypothesis that some of the parameters in a model are restricted to zero. If the linear regression model has p parameters and the experimenter wants to test whether r of these are zero based on N observations, the criterion is

$$F = \frac{\frac{A_1 - A_0}{r}}{\frac{A_0}{N - p}} \sim F(r, N - p) \quad (2.35)$$

where A_0 is the, smaller, sum of squares of the unrestricted model, A_1 is the, larger, sum of squares of the restricted model, and $F(r, N - p)$ denotes the F -distribution with r and $N - p$ degrees of freedom [115].

In the conditional or statistical aspect, the $\text{ARMA}(n_y, n_e)$ is exactly a linear regression model; hence the above criterion can be used to test the hypothesis that r out of its $(n_y + n_e) = p$ parameters are zero. Then, A_0 becomes the residual sum of squares of the $\text{ARMA}(n_y, n_e)$ model and A_1 that of the same model with r parameters dropped out. The justification of the criterion for the unconditional or dynamic aspect of the ARMA model, together with its interpretation as a convergence criterion, may be found in [34, 81, 113, 115, 120].

Model order selection based on the incremental change in RSS is known to give inaccurate estimates of model order [34, 63, 113]. The problem is that when spurious parameters are introduced into the model they may model the noise, giving biased estimates of RSS. Although the F-test is designed to account for over-fitting due to noise, in many practical applications this technique gives unreliable estimates of model order [30, 65].

2.6.1.4 Final Prediction Error

For simplicity, this discussion is restricted to AR models (Equation 2.7), where n_y is the model order. Consider a sequence $\{y(n)\}$, $n = 1, 2, \dots, N$, which is to be

predicted using some linear predictor. Its final prediction error is defined as:

$$E \left\{ y(n) - \sum_{j=1}^{n_y} \hat{a}_j y(n-j) \right\}^2 \quad (2.36)$$

which asymptotically approaches $(N + n_y)\sigma^2$ as $N \rightarrow \infty$ for a given n_y and unknown noise variance σ^2 [39]. With this definition, an estimate of σ^2 and the coefficients can be obtained by modeling a “training” sequence. Let the training sequence be $\{x(n)\}$, $n = 1, 2, \dots, N$, which is fitted with a linear predictor of order n_y . The expectation of the residual variance is then asymptotically given as $\hat{\sigma}_{n_y}^2 = (N + n_y)\sigma^2$. Using this as an estimate of σ^2 the FPE of $y(n)$ can be computed as

$$\text{FPE}(n_y) = \frac{N + n_y}{N - n_y} \hat{\sigma}_{n_y}^2. \quad (2.37)$$

This is the definition of final prediction error (FPE) for an AR model of order n_y where N is the number of samples to which the model was fitted and $\hat{\sigma}_{n_y}^2$ is the estimated residual noise variance for a model of order n_y .

The final prediction error estimate of model order gives a minimal FPE value over a finite range of $n_y = 1, 2, \dots, n_{y_{\max}}$. The FPE technique assumes that the optimum model order is achieved when the estimated residual noise variance is minimized. This value will always decrease as the model order is increased and, as a result, it is not a reliable estimator for model order.

2.6.1.5 Akaike's Information Criterion

Akaike developed a more general criterion based on information theoretic concepts and called it AIC (Akaike's information criterion) [5].

The Kullback-Liebler distance is defined as

$$I(g; f(\cdot|\theta)) = S(g; g) - S(g; f(\cdot|\theta)) \quad (2.38)$$

where $S(\cdot)$ is the residual noise variance, $S(g; f(\cdot|\theta)) = \int g(x) \log f(x|\theta) dx$ is the expected log-likelihood for a p.d.f. of x , $g(x)$ and the conditional p.d.f. of x given

a vector parameter θ . It was shown in [95] that under certain conditions $I(g; f(\cdot|\theta))$ can be approximated by

$$\left(\frac{1}{2}\right) \|g(x) - f(x|\theta)\|_{J^2} \quad (2.39)$$

where J is the Fisher information matrix [45, 101] and $\|\Delta q\|_{J^2} = \Delta q' J \Delta q$, for $\Delta q = g(x) - f(\cdot|\theta)$. This approximation can be shown to have a distribution function which is asymptotically (non-central) chi-squared for $N \rightarrow \infty$ and n_y degrees of freedom. Its expectation can be shown to be

$$N\|\hat{\theta} - \theta_0\|_{J^2} + n_y, \quad (2.40)$$

for a maximum likelihood (ML) estimate of the parameter vector, $\hat{\theta}$, and the unknown true parameter vector, θ_0 . The optimal model minimizes this expectation. If the estimate $N\|\hat{\theta} - \theta_0\|_{J^2}$ is computed by

$$2 \left(\sum_{i=1}^N \log \frac{f(x_i|\theta_0)}{f(x_i|\hat{\theta})} \right) \quad (2.41)$$

it needs correction by $2n_y$ to give the expected log-likelihood for a (general) model of order n_y . This leads to the definition of AIC,

$$\text{AIC}(n_y) = (-2) \log[\text{maximized likelihood}] + 2n_y. \quad (2.42)$$

For AR models (also MA and ARMA), the log-likelihood function for N observations is given by

$$L = -\frac{N}{2} \log \sigma^2 - \frac{1}{2\sigma^2} Q(\mathbf{a}) + \text{const}, \quad (2.43)$$

where $Q(\mathbf{a}) = \sum_{n=1}^N [y(n) + \sum_{j=1}^{n_y} a_j x(n-j)]^2$, and $\mathbf{a}$ denotes the set of coefficients for the model. For the ML solution, $\hat{\sigma}^2$ is obtained as

$$\hat{\sigma}^2 = \frac{1}{N} Q(\mathbf{a}) \quad (2.44)$$

and for the maximum of L

$$\hat{L} = -\frac{N}{2} \log \hat{\sigma}^2 - \frac{N}{2}. \quad (2.45)$$

The last term in Equation 2.45 is constant for a given sample size and can be ignored, so that AIC becomes

$$\text{AIC}(n_y) = N \log \hat{\sigma}^2 + 2n_y. \quad (2.46)$$

The best model is determined as the one for which $\text{AIC}(n_y)$ attains its minimum value.

2.6.1.6 Minimum Description Length

One problem with AIC is that it is inconsistent, i.e., its variance does not tend to zero for larger sample sizes. This was shown to be due to the penalty term which does not decrease fast enough with N to balance the first term [72]. A model estimator proposed by Rissanen [128], called the minimum description length (MDL), was designed to overcome this problem.

The number of parameters necessary to reproduce an observed sequence $\{y_1, \dots, y_N\}$ of a time series depends on the model and parameters assumed to have generated the data [128]. The MDL technique finds the model which minimizes the description length and thereby computes an estimate of model order [128].

Binary prefix codes are used to encode data strings. These data strings can be made up of symbols, parameters, numbers, etc. It is known that the average length of a code word is bounded by Shannon's theorem [128]. Therefore, it is possible to write [128]

$$\sum_x p(x) L(x) \geq - \sum_x p(x) \log p(x) \quad (2.47)$$

where $L(x)$ is the length of the code word (i.e., length of parameter vector θ) and

$p(x)$ is the probability of x . It is also possible to write

$$L(y|x, \theta) = -\log p(y|x, \theta) \quad (2.48)$$

where $L(y|x, \theta)$ is known as the log-likelihood function (to be maximized). Let $\hat{\theta}$ denote the value of the parameter which maximizes the likelihood and thus minimizes the parameter vector length (i.e., code word length) $L(y|x, \theta)$. Since $\hat{\theta}$ can only be encoded up to a certain precision, the code word length, $L(y|x, \theta)$, becomes longer than the desired minimum $L(y|x, \hat{\theta})$, given noise considerations. Let the precision be $\delta = 2^{-q}$ where q is the number of bits used for encoding the parameter. It is possible to save on the code word length if q is small. However, the result is a loss in precision. The optimal precision depends on the size of the observed data via $-\log \delta = 0.5 \log N$, and hence the total code word length for k parameters is given by the MDL,

$$\text{MDL}(k) = -\log[\text{maximized likelihood}] + \frac{1}{2}k \log N \quad (2.49)$$

which, for an $\text{AR}(n_y)$ model gives

$$\text{MDL}(n_y) = \log[\text{maximized likelihood}] + \frac{n_y}{N} \log N. \quad (2.50)$$

2.6.1.7 Relationships Between FPE, AIC and MDL

FPE was developed specifically for AR model order determination. AIC is a general measure and can be applied to other models. For large sample sizes N , FPE and AIC can be shown to be asymptotically equivalent methods [120]:

$$\begin{aligned} \log[\text{FPE}] &= \log \left[\frac{1 + n_y/N}{1 - n_y/N} \hat{\sigma}^2 \right] \\ &= \log \hat{\sigma}^2 + \frac{2n_y}{N} \quad \text{for large } N, \end{aligned} \quad (2.51)$$

$$\text{therefore } \text{AIC}(n_y) = N \log[\text{FPE}(n_y)].$$

FPE and AIC are called asymptotic measures, as their derivation involves taking the number of samples N to infinity. MDL does not make this assumption. The penalty term is the highest for MDL and lowest for FPE, i.e., the penalty term in the MDL definition is larger than that of AIC by a factor of approximately $\log N$. This causes a much steeper minimum. In practice, this normally results in a lower and a less variable estimate for the optimal model order.

All these techniques are theoretically a function of the residual noise variance only. The residuals, however, are a function of the method used to obtain the model parameters. Consequently, residual variance estimates can be shown to be a function of the input variance and parameter variance [31]. Therefore, in practice, the training sample size and methods used for model order selection and parameter estimation are crucial.

In context of order selection for nonlinear systems, we believe that these methods fail for the following reasons:

1. Number of possible terms for a given order can be very large (see Equation 2.22). Due to over-parameterization residual estimate will be under-dispersed, i.e., biased.
2. All three approaches rely on accurate estimates of $\hat{\sigma}^2$, i.e., accurate estimates of residuals.
3. All three approaches rely on optimal parameter estimates which depend on the data size N . For finite data lengths these methods may give inconsistent estimates.
4. Inadequacy of the penalty term in each method is known to give inconsistent estimates of order for linear systems.

2.6.1.8 False Nearest Neighbors Method

Recently, Kennel *et al.* [2, 82] proposed the false nearest neighbors (FNN) algorithm for determining the smallest dimension regression vector needed to recreate the dy-

namics of autoregressive (autonomous) chaotic systems [126]. The FNN method has also been applied to non-autoregressive (non-autonomous) systems for model order estimation [27, 82, 127]. The focus of this method is to determine the functional relationship between a regression vector and an output vector. Given a set of observed regressors $\psi(n)$ for $n = 1, 2, \dots, N$ and observed outputs, $y(n)$, related to the regression vector, this method attempts to find a functional relationship

$$y(n) = G[\psi(n)] = G[y(n-1), \dots, y(n-n_y), u(n), \dots, u(n-n_u)] \quad (2.52)$$

that minimizes some error function ξ . This error function is often of the form

$$\xi = \|y(n) - G[\psi(n)]\|_2 \quad (2.53)$$

where G is some unknown function of the underlying system. If the system is linear the function G can be determined exactly using the z -transformation. When the function is nonlinear computing the function G is impossible except in trivial cases [53].

The FNN method starts by determining the closest point to a given point in the regression space. In other words, for a given regressor $\psi_{n_y, n_u}(i)$ find another regressor $\psi_{n_y, n_u}(j)$ in the data set which minimizes distance d :

$$d = \|\psi_{n_y, n_u}(i) - \psi_{n_y, n_u}(j)\|_2. \quad (2.54)$$

The indices i and j are not necessarily close to one another. If i and j are always close to one another the sampling time may be too small and there may be problems in accurately estimating the dimension of the regression vector [54]. To determine whether neighbors are “true” or “false”, a test is defined to assess whether these neighbors have future outputs that are “far apart”. A ratio test

$$\frac{|y(i) - y(j)|}{\|\psi_{n_y, n_u}(i) - \psi_{n_y, n_u}(j)\|_2} \leq R \quad (2.55)$$

is used to determine whether the distance between future outputs is significantly larger

than the distance between time-delay regression vectors that are close in the regressor space. If the distance between future outputs is “large” (i.e., $\geq R$) when divided by the distance between two points that are “nearest neighbors” in the regressor space, then the neighbors are considered to be false. The percentage of points in the data set that have false nearest neighbors are calculated for all times i . This is continued for increasing n_y and n_u until the percentage of false nearest neighbors drops to zero or some acceptably small number.

Two search methods may be used with the FNN method: (i) global and (ii) local search. In the global search the FNN indices become surfaces in two dimensions. It is possible to find a “global” solution (or solutions) for the model orders by computing the desired index over all values of input and output lag in a certain range and determine which point (or points) satisfy the order determination conditions [27, 125]. For a “local” solution, initial guesses for the minimum input and output lags are used and the optimum model order is computed competitively; at each iteration either the input or output lag is increased by one, depending on which reduces the FNN index by the greatest amount [27]. The competitive search method provides a “local” solution to the optimum model order which may not agree with the “global” solution.

The FNN technique requires the selection of a tolerance level, R , to determine true and false neighbors. Selection of this tolerance requires *a priori* knowledge about the true errors and system output, which are seldom available [54]. In the presence of noise, implementing either search technique may not provide a unique solution since noise can be considered as another dynamical system with a very high order [1]. In addition, this technique does not provide the “full” system order (Equation 2.21) for NARMAX models since the nonlinear mapping G cannot be computed [54, 125, 126, 127]. Hence, only n_y and n_u can be estimated.

2.6.1.9 Summary

Currently, no optimal method for model order estimation of nonlinear systems exists. In the sequel (Chapter 5) we will present a method for model order selection of NARMAX models based on minimization of the error in cross-validation.

2.6.2 Structure Detection

Many NARMAX models are described by only a few terms. However, if the order of the system is high the number of candidate terms will be very large. Equation 2.22 gives the maximum number of possible terms in a NARMAX model. For example, a system described by tenth-order lag on the input-output and third-order nonlinearity (i.e., $n_u = n_y = 10$, $l = 3$) has $p = 1771$ candidate terms.

The structure detection problem is that of selecting the subset of candidate terms that best predicts the output while maintaining an efficient system description.

Existing methods for determining model structure include hypothesis testing of differences between parameters (means) via the t-test, stepwise regression and Korenberg's orthogonal structure detection routine. The t-test (in conjunction with regression analysis) detects structure by determining those parameters whose values are significantly different from zero. Conversely, stepwise regression computes how much each parameter contributes to the overall reduction in mean squared error resulting from adding or removing a parameter. Korenberg's orthogonal method is similar to stepwise regression since it determines how much each parameter contributes to the overall reduction in mean squared error by using orthogonal relationships. The t-test and stepwise regression are widely used in regression analysis [45, 49, 133, 142]. However, all of these have difficulty in nonlinear system identification, but for the different reasons discussed below.

2.6.2.1 t-test

The t-test in combination with regression analysis is sometimes referred to as a form of hypothesis testing by computing the differences between means [45].

In regression the significance of the coefficients, θ , are checked using statistical tests. If the model that was postulated is more general than needed, tests of hypothesis are necessary to give a minimal model description. Suppose the following model was fit

$$E(Z) = \hat{\theta}_0 + \hat{\theta}_1\psi_1 + \hat{\theta}_1\psi_2 + \dots + \hat{\theta}_{p-1}\psi_{p-1}. \quad (2.56)$$

The $\hat{\theta}$'s are then tested against the hypothesis, H_0 (null hypotheses [33, 56]) or $\hat{\theta}_i = 0$, $i = 1, 2, \dots, p$. This allows the experimenter to assess which parameters are significant and which are not, consequently which ones to retain.

The t-distribution is defined as the ratio of a normal random variable divided by the square root of a Chi-squared random variable

$$t = \frac{\text{normal}}{\sqrt{\chi^2}}. \quad (2.57)$$

In standard regression analysis it is assumed that the errors are normally distributed, therefore, Z must be normally distributed which also implies that $\theta - \hat{\theta}$ is normally distributed (see e.g., [45, 56, 133]). In addition, it is assumed that estimates of the variance have a χ^2 distribution [45, 133]. A χ^2 distribution is defined as the sum of squared normal random variables, i.e., the sum of squared errors [45, 56, 133]. These statistics fit the definition of the t-test and are easily calculated as a by-product to the regression procedure, e.g., ELS or MLE. These estimates are typically used to compute the t-distribution of the estimated parameters (with $n - p$ degrees of freedom).

The t-test is:

$$-t_1 < \frac{\theta - \hat{\theta}}{\sqrt{\hat{\sigma}^2}} < t_1. \quad (2.58)$$

The range $-t_1$ to t_1 is determined from standard t-tables for some α level of significance. If the computed t-statistic is in the range $-t_1$ to t_1 the parameter is accepted otherwise it is rejected, i.e., removed from the regression.

This procedure assumes that an accurate estimate of parameter variances, i.e., residuals, is available [133]. Our results indicate that this assumption is violated for over-parameterized models and, therefore, may lead to inaccurate estimates of system structure [90, 93].

2.6.2.2 Stepwise Regression

The stepwise regression algorithm that is widely used is due to the original work of Efroymson [49]. Stepwise regression relies on the incremental change in the residual sums of squares (RSS) resulting from adding or removing a parameter. Specifically, two F-levels, F_{out} and F_{in} , are formed to determine whether a parameter should be removed from the model (F_{out}) or included in the model (F_{in}) [49, 133, 142, 141]. These F-levels are based on an F-distribution with 1, $N - p$ degrees of freedom.

This algorithm is summarized as [106]:

1. Enter into the regression model any variables (parameters) that are to be “forced” in.
2. Find the variable from those not in the model but, available for inclusion, with the largest F_{in} value. If it is at least as great as a pre-specified value of F_{in} , then add the variable to the model. Stop if no variables can be added.
3. Find the variable among those in the model, other than those forced in, that has the smallest F_{out} value. If it is less than a pre-specified value of F_{out} , then remove the variable from the model. Repeat this step until no further variables can be removed. Go to step 2.

Efroymson states that F_{out} must not be greater than F_{in} for “good” model parameterizations [49, 106, 142, 141]. In the procedure above, if RSS_p is the residual sums of squares for a model with p parameters, then the F_{in} statistic is given as [106]:

$$F_{in} = \frac{RSS_p - RSS_{p+1}}{RSS_{p+1}/(N - p - 1)} \quad (2.59)$$

where N is the data length. Similarly the F_{out} statistic is given as [106]:

$$F_{out} = \frac{RSS_{p-1} - RSS_p}{RSS_p/(N - p)}. \quad (2.60)$$

This method is sensitive to the order in which the regressors are introduced and often gives models with incorrect structure [41, 71, 102, 141].

2.6.2.3 Orthogonal Structure Detection

Korenberg [3, 4, 84, 85, 87] developed an orthogonal structure detection routine specifically for nonlinear systems. This method relies on orthogonalizing the regressor matrix and using the orthogonal relationships to compute how much each term would reduce the total mean-squared error. The regressor matrix Ψ is decomposed into W and A where W is an $N \times p$ matrix of orthogonal columns and A is a $N \times N$ unit upper triangular matrix. This yields an auxiliary orthogonal system description

$$\begin{aligned} Z &= Wg + \xi \quad \text{where } g = A\theta \quad \text{and} \\ \xi &= Z - \Psi\theta = (\Psi A^{-1})(A\theta) = Z - Wg. \end{aligned} \quad (2.61)$$

An error reduction ratio Err

$$Err_i = g_i^2 \frac{\langle w_i, w_i \rangle}{\langle z, z \rangle}; \quad i = 1, 2, \dots, p \quad (2.62)$$

is defined: the square of the auxiliary system parameters, g , times the inner product of the orthogonal columns, W , over the inner product of Z .

A column of W is selected which gives the best reduction in error as the first column to include in a new orthogonal matrix $\tilde{W}$ and upper triangular matrix $\tilde{A}$. This is continued until no more columns contribute to the reduction in error as compared with some preselected tolerance, ρ .

This algorithm suffers from having to select a desired tolerance level to determine which terms to accept or reject. Ideally, ρ should be set as

$$\rho = \frac{E[e^2(n)]}{E[y^2(n)]}; \quad (2.63)$$

the ratio of the expected value of the squared noise process over the expected value of the true system output [38]. This requires *a priori* knowledge about the true errors and system output, which are seldom available. Therefore, the tolerance level is set by trial and error [38].

2.6.2.4 Summary

Our results suggest that these structure detection techniques are difficult to apply to highly over-parameterized models possibly because the underlying assumptions are often violated [90, 93]. Consequently, in Chapter 4 we develop an alternative approach for structure computation of NARMAX models.

2.6.3 Parameter Estimation

Many parameter estimation techniques are based on least-squares theory. Therefore, some of the basic assumptions made in least-squares estimation are stated.

2.6.3.1 Least-Squares

Consider the system shown in Figure 2.2. This system can be described parametrically

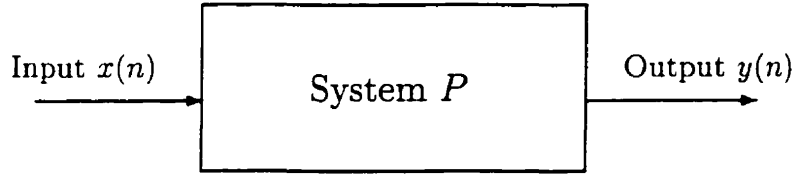


Figure 2.2: Least-squares system description.

as:

$$Y = X\theta + e \quad (2.64)$$

where Y is a $N \times 1$ vector of outputs, X is a $N \times p$ matrix of inputs (regressors), θ is a $p \times 1$ vector of unknown parameters and e is the $N \times 1$ vector of errors. The vector e represents the error in measuring the output $X\theta$ so that $X\theta$ is the true response and Y is the observed response. This is known as the least-squares problem.

A method for obtaining an estimate of the system parameters, θ , is the method of least-squares [133]. This method consists of minimizing $\sum_{n=1}^N e(n)^2$ with respect to θ . It can be shown that the least-squares solution to this minimization is

$$\hat{\theta}_{OLS} = (X^T X)^{-1} X^T Y. \quad (2.65)$$

This is known as the ordinary least-squares (OLS) estimate of θ .

Some basic properties of least-square are mentioned briefly.

Assumptions:

1. $E[e] = 0$ assumes zero-mean noise. As a result $E[Y] = X\theta$.
2. X is a deterministic signal, $E[X] = X$.
3. Y is stochastic.
4. $\text{cov}[e_i, e_j] = \delta_{ij}\sigma^2$ assumes white noise.
5. e_i 's are uncorrelated.

If the “errors” are unbiased, that is, $E[e] = 0$ then

$$\begin{aligned} E[\hat{\theta}] &= (X^T X)^{-1} X^T E[Y] \\ &= (X^T X)^{-1} X^T X \theta \\ &= \theta. \end{aligned} \tag{2.66}$$

Hence, $\hat{\theta}$ is an unbiased estimate of θ .

If it is assumed that the e_i are uncorrelated and have the same variance (homoskedastic), that is, $\text{cov}[e_i, e_j] = \delta_{ij}\sigma^2$, then $D[e] = \sigma^2 I_n$ (where “ D ” denotes the “dispersion” or “variance-covariance” matrix)

$$D[Y] = D[Y - X\theta] = D[e]. \tag{2.67}$$

It then follows that

$$\hat{\sigma}^2 = \frac{(Y - X\hat{\theta})^T (Y - X\hat{\theta})}{N - p} \tag{2.68}$$

is an unbiased estimate of σ^2 . Therefore,

$$\begin{aligned}
D[\hat{\theta}] &= D[(X^T X)^{-1} X^T Y] \\
&= (X^T X)^{-1} X^T D[Y] X (X^T X)^{-1} \\
&= \sigma^2 (X^T X)^{-1} X^T X (X^T X)^{-1} \\
&= \sigma^2 (X^T X)^{-1}
\end{aligned} \tag{2.69}$$

is the variance of the estimated parameters $\hat{\theta}$.

To summarize, the OLS estimate assumes that X is deterministic and the equation errors are zero-mean and uncorrelated. This yields an unbiased minimum variance estimate of the unknown parameters, which is commonly known as the best linear unbiased estimate, or BLUE, of $X\hat{\theta}$.

2.6.3.2 Parameter Estimation for NARMAX Models

Although nonlinear structures based on expansions of lagged inputs and outputs may provide a very concise system representation, any measurement noise will enter the model as product terms with the system input and output. Consider the system shown in Figure 2.3. The system P is assumed to be a function of both current and

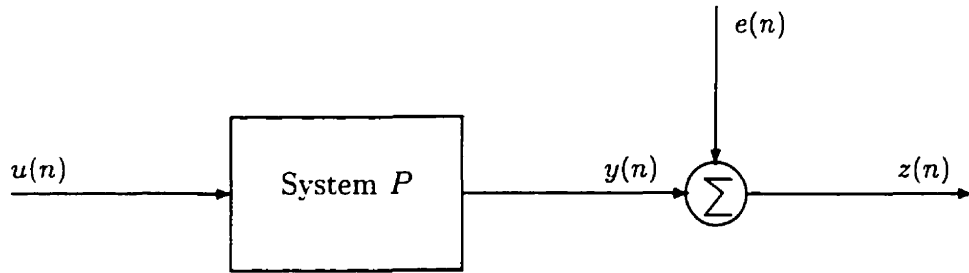


Figure 2.3: Noisy system configuration.

past inputs and past outputs. The noise term $e(n)$ is assumed to be a stationary, zero-mean random process with auto-correlation function $R_{ee}(\tau) = 0, \forall \tau > 0$. Since the noise term is stochastic it is difficult to obtain a “good” estimate of the noise process. As a result estimates of the error terms are correlated because (1) estimates of the noise process are usually poor and (2) the system is a function of past outputs. However, it is assumed that the errors are not correlated with either the true system output ($y(n)$) or the system input.

Let system P be represented by the following NARMAX model

$$\begin{aligned} y(n) = & \theta_1 u(n - d_1) + \theta_2 u(n - d_2) y(n - d_3) + \theta_3 y^2(n - d_4) \\ & + \theta_4 y(n - d_5) \end{aligned} \quad (2.70)$$

where d_i represents the delay and θ_i the coefficient. The measured output is related to the measurement noise or errors and true system output as

$$z(n) = y(n) + e(n) \implies y(n) = z(n) - e(n). \quad (2.71)$$

Substituting Equation 2.71 into Equation 2.70 gives

$$\begin{aligned} z(n) - e(n) = & \theta_1 u(n - d_1) + \theta_2 u(n - d_2) [z(n - d_3) - e(n - d_3)] \\ & + \theta_3 [z(n - d_4) - e(n - d_4)]^2 + \theta_4 [z(n - d_5) - e(n - d_5)]. \end{aligned} \quad (2.72)$$

Note that although the model is linear-in-the-parameters and the noise is output additive, the noise can enter the system as multiplicative terms with the input and output. Consequently, most parameter estimation algorithms for linear systems cannot be applied directly because the assumption that the noise terms in the model are independent of the input is violated and X is no longer deterministic [68].

The least-squares formulation for this system (Equation 2.72) is

$$\begin{aligned} Z &= \Psi_{zu} \theta + \varepsilon \quad \text{where} \\ \varepsilon &= \theta_2 u(n - d_2) e(n - d_3) - 2\theta_3 z(n - d_4) e(n - d_4) + \theta_3 e^2(n - d_4) \\ &+ \theta_4 e(n - d_5) + e(n) \end{aligned} \quad (2.73)$$

where Z is a $N \times 1$ vector of measured outputs, Ψ_{zu} is a nonsingular $N \times p$ matrix of regressors, based on input-output only, and ε is a $N \times 1$ vector of modeling errors and noise. The regressor matrix, Ψ_{zu} , is related to Equation 2.13 since the columns of Ψ_{zu} represent an expansion of NARMAX model terms, for a given model order. This model formulation assumed that $n_e = 0$, i.e., that Ψ_{zu} is deterministic. However, since Ψ_{zu} is a function of the system inputs ($u(n)$) and *measured* outputs ($z(n)$), Ψ_{zu}

is not deterministic. The ordinary least-squares estimate, $\hat{\theta}_{OLS}$, based on this is

$$\hat{\theta}_{OLS} = (\Psi_{zu}^T \Psi_{zu})^{-1} \Psi_{zu}^T Z. \quad (2.74)$$

which will give a biased estimate of the parameters. This is shown by taking the expectation of $\hat{\theta}_{OLS}$:

$$\begin{aligned} E[\hat{\theta}_{OLS}] &= E[(\Psi_{zu}^T \Psi_{zu})^{-1} \Psi_{zu}^T Z] \\ &= E[(\Psi_{zu}^T \Psi_{zu})^{-1} \Psi_{zu}^T [\Psi_{zu} \theta + \varepsilon]] \\ &= \theta + E[(\Psi_{zu}^T \Psi_{zu})^{-1} \Psi_{zu}^T \varepsilon] \end{aligned} \quad (2.75)$$

where $E[(\Psi_{zu}^T \Psi_{zu})^{-1} \Psi_{zu}^T \varepsilon] \neq 0$ or equivalently $E[\Psi_{zu}^T \varepsilon] \neq 0$. This induces a biased parameter estimate when OLS is applied directly. To obtain an unbiased estimate of θ , other parameter estimation techniques based on least-squares are needed.

2.6.3.3 Extended Least-Squares

ELS is a technique that addresses the bias problem by modeling the lagged errors to obtain an unbiased parameter estimate. Extended least-squares (ELS) for linear systems has been widely studied and is also referred to as Panuska's method, the extended matrix method, or approximate maximum likelihood [61].

Let the least-squares problem be defined as in Equation 2.73. In general, since the noise sequence is a stochastic process, it is not possible to solve for the noise source e , and it will not be equal to the prediction errors [22]. The prediction errors are defined as

$$\hat{\varepsilon} = Z - \hat{Z} \quad (2.76)$$

where $\hat{Z}$ is the predicted output

$$\hat{Z} = \Psi_{zu} \hat{\theta}_{OLS}. \quad (2.77)$$

In ELS, the NARMAX formulation of Equation 2.13 is redefined into a prediction

error model with $\hat{e}$ replacing e ; making it a deterministic least-squares problem.

The ELS formulation is an extension of ordinary least-squares and is defined as

$$\hat{\theta}_{ELS} = (\Psi^T \Psi)^{-1} \Psi^T Z: \quad \text{where} \quad \Psi = [\Psi_{zu} \Psi_{zu\hat{e}} \Psi_{\hat{e}}]. \quad (2.78)$$

Ψ is a partitioned regressor matrix where Ψ_{zu} is a function of z and u only, $\Psi_{zu\hat{e}}$ represents all the cross products involving $\hat{e}$, and $\Psi_{\hat{e}}$ is a polynomial function of the prediction errors only.

The ELS approach is straightforward and is summarized as:

1. Calculate the ordinary least-squares estimate, $\hat{\theta}_{OLS}$.
2. Calculate an estimate of the prediction errors, $\hat{e}$.
3. Form the extended regressor matrix, Ψ , with the estimated prediction errors and calculate the ELS estimate, $\hat{\theta}_{ELS}$.
4. Go to step 2 until convergence, i.e., until prediction errors are white.

It is well documented that this algorithm does converge when applied to linear systems [22]. Simulation results for nonlinear systems confirm that the method is well suited for nonlinear polynomial identification also [22, 61].

The major disadvantage encountered when this method is applied to nonlinear systems is that noise or prediction errors must be included in the estimation vector. This results in introducing many additional candidate terms to the model. The maximum number of entries in the parameter vector is given by Equation 2.22 and can be large even for moderately complex models. If the nonlinearity within the system is high-order (i.e., large l) the dimension of the parameter vector increases rapidly.

In an attempt to limit the dimension of the coefficient vector, other least-squares algorithms are considered.

2.6.3.4 Instrumental Variable Method

The principle of instrumental variables (IV) has been applied to linear system identification in several ways [22, 61]. Consider again the least-squares problem defined in Equation 2.73. The IV method is based on selecting an instrument matrix V which satisfies the conditions

$$\begin{aligned} \lim_{N \rightarrow \infty} \frac{1}{N} V^T \Psi_{zu} &= R; \quad \text{where } R \text{ is nonsingular} \\ \lim_{N \rightarrow \infty} \frac{1}{N} V^T (Z - \Psi_{zu} \theta_0) &= 0 \end{aligned} \quad (2.79)$$

where θ_0 denotes the true parameter vector, Z is the output, Ψ_{zu} denotes the regressor matrix and “lim” refers to limit in probability. The conditions of Equation 2.79 require (1) the instrumental matrix to be linearly independent and invertible, i.e., $V^T \Psi_{zu}$ have full rank, and (2) the errors have mean zero and be uncorrelated with V . This ensures that the estimate

$$\hat{\theta} = (V^T \Psi_{zu})^{-1} V^T Z \quad (2.80)$$

is unbiased since the instrument matrix is not correlated with the errors [68].

The most popular way to satisfy Equation 2.79 is to define V^T to have the same structure as Ψ_{zu}^T but with the measured outputs replaced by predicted outputs [61, 68]. The columns of V^T associated with input are unchanged since it is assumed that the input is measured with negligible error. This algorithm is often referred to as the auxiliary model algorithm. Unfortunately, instrumental variables can only be applied to nonlinear systems if certain properties of the system noise are satisfied.

Consider the NARMAX model with error sequence redefined as

$$\xi = \Psi_{zu\hat{\epsilon}} \theta_{zu\hat{\epsilon}} + \Psi_{\hat{\epsilon}} \theta_{\hat{\epsilon}} + \hat{\epsilon} \quad (2.81)$$

to yield the description

$$Z = \Psi_{zu} + \xi. \quad (2.82)$$

This model formulation leads to biased parameter estimates whenever the system under test is nonlinear because [22]

$$\lim_{N \rightarrow \infty} \frac{1}{N} V^T (Z - \Psi_{zu} \theta_{zu}) = \lim_{N \rightarrow \infty} \frac{1}{N} V^T (\Psi_{zu\hat{\epsilon}} \theta_{zu\hat{\epsilon}} + \Psi_{\hat{\epsilon}} \theta_{\hat{\epsilon}} + \hat{\epsilon}) \neq 0. \quad (2.83)$$

A typical term in Equation 2.83 takes the form [22]

$$\lim_{N \rightarrow \infty} \frac{1}{N} V^T [z^i u^j \hat{\epsilon}^k] \quad \text{for some } i, j, k \quad (2.84)$$

and will not in general be zero even when $\hat{\epsilon}(n)$ is a zero-mean white noise sequence (see example in [22] on pp. 608-9). Therefore, in general, the instrumental variable algorithm will yield biased estimates for nonlinear systems. However, the IV method gives unbiased results for one special case of the NARMAX model. When the noise terms are represented within the NARMAX model as a linear expansion (i.e., the model is represented by linear output terms), it always satisfies the conditions of Equation 2.79. Only for this special case, is it possible to use IV for NARMAX model identification.

Often, however, in many applications, there is insufficient *a priori* information about the system structure which does not allow for an intelligent choice about when the IV algorithm can be used. This ambiguity arises when the system structure is completely known and, therefore, it may be represented by a nonlinear expansion of output terms. The IV parameter estimation technique is limited to cases in which the process is nonlinear and the prediction errors are linear, i.e., output terms are linear. The class of models that give linear output and error terms are blocked structured N-L models (static nonlinearity followed by a causal, linear, time-invariant, dynamic system), i.e., Hammerstein models. Consequently, the class of models that can be identified using this method is limited.

2.6.3.5 Suboptimal Least-Squares

The number of parameters in the NARMAX model increases significantly if the noise model is included in the estimation vector. Therefore, it would be advantageous

if unbiased parameter estimates could be obtained without specifically estimating a noise model. This can be achieved for the NARMAX model, whenever the noise enters as an additive signal at the output, by using a suboptimal least-squares (SOLS) algorithm [22].

The suboptimal algorithm is a variant of the ordinary least-squares method that achieves an unbiased estimate of the parameters by re-expressing Equation 2.13 as:

$$z(n) = F^t[y(n-1), \dots, y(n-n_y), u(n), \dots, u(n-n_u)] + e(n) \quad (2.85)$$

eliminating all cross-product terms involving noise. Parameter estimation based on this new expansion would, therefore, require significantly less computational effort compared with ELS. The noise-free output, $y(n)$, cannot be measured but may be estimated recursively as

$$\hat{Y} = \Psi_{\hat{y}u} \hat{\theta}_{yu}. \quad (2.86)$$

The noise-free output $y(n)$ in Equation 2.85 is effectively replaced by the estimate $\hat{y}(n)$. This algorithm was derived by Billings and Voon [22] specifically for nonlinear systems.

The SOLS algorithm significantly reduces the size of the parameter vector. However, the convergence properties of this algorithm crucially depend on the ability to compute a “good” estimate of “noise-free” output, $\hat{Y}$. Computation of a good estimate of $\hat{Y}$ depends on the underlying system and noise characteristics. Even when it is possible to compute a good estimate of noise-free output, the additive sequence, $e(n)$, may be highly correlated with itself. Therefore, using $e(n)$ to estimate parameter variance will result in a highly biased estimate. This makes structure detection, which is necessary to obtain a parsimonious description of the system, inaccurate or impossible.

2.6.3.6 Maximum Likelihood Estimate

While least-squares based methods are the most popular techniques for parameter estimation, they are not the only methods. Maximum likelihood estimators (MLE) are also widely used, especially when least-squares methods converge slowly or not at all [24, 61].

The MLE has been shown to be equivalent to the weighted least-squares estimator for Gaussian innovations [61]. The MLE of θ is:

$$\hat{\theta}_{MLE} = (\Psi^T \hat{\Sigma}^{-1} \Psi)^{-1} \Psi^T \hat{\Sigma}^{-1} Z \quad (2.87)$$

where

$$\hat{\Sigma} = \frac{1}{n} (Z - \Psi \hat{\theta}_{MLE})(Z - \Psi \hat{\theta}_{MLE})^T.$$

The procedure for the maximum likelihood algorithm is similar to the ELS method and is summarized below [61].

1. Pick any value of $\hat{\Sigma}$ (say I)
2. Solve for $\hat{\theta}_{OLS}$ using $\hat{\Sigma}$
3. Solve for $\hat{\Sigma}$ using $\hat{\theta}_{OLS}$ as an initial estimate of $\hat{\theta}_{MLE}$
4. Solve for $\hat{\theta}_{MLE}$ using $\hat{\Sigma}$
5. Solve for $\hat{\Sigma}$ using $\hat{\theta}_{MLE}$
6. Stop if converged, otherwise go to 4

The MLE of the parameters are consistent, asymptotically normally distributed and asymptotically efficient [61]. Results show that the maximum likelihood algorithm derived for Gaussian innovations can be applied to general distributions without any of the essential properties being lost. Asymptotic normality results for the prediction-error method implies that statistical tests can be applied to determine significant parameters in the estimated model [24]. It should be noted that using MLE also requires modeling the noise; therefore, it is computationally more expensive than

the ELS method. Moreover, computing $\hat{\Sigma}^{-1}$ may be unstable since $\hat{\Sigma}$ may be ill-conditioned [42]. In this situation, $\hat{\theta}_{MLE}$ will be biased or impossible to compute. The choice of method depends on the application, the type of system being investigated and if there are problems with convergence.

2.6.3.7 Summary

The IV algorithm can be applied only to Hammerstein structure systems, severely restricting the class of nonlinear systems that can be identified with this approach. SOLS has problems with convergence and gives an error vector that is highly correlated. Since both the IV and SOLS methods have limitations, they cannot be used in most cases. However, ELS and MLE can be used for parameter estimation. In most cases ELS is preferred because it is computationally less expensive and does not involve inverting a possibly ill-conditioned matrix.

2.6.4 Model Validation

Model validity tests should be a fundamental part of any system identification procedure. This is an important step in identification since it is often the final check on the goodness of fit of any identified model.

Model validation is really concerned with “model falsification”. That is, the user tries to establish convincing evidence that a certain model could not have produced the observed data. A model that “so far” has not been falsified can be considered — for the moment — to be “validated” [100]. The essence of model validation, while trying to falsify the model, is to find evidence that the bias error is significantly larger than the random error [160]. The bias error is defined as the systematic contribution of the model error that stems from incorrect model structure, while random error is the contribution that has roots in the various disturbances that affect the data [160].

Many model validity tests have been designed to indicate the inadequacy of the fitted model. However, most assume that the system under investigation is linear (see, e.g., [62, 89, 96, 100, 111, 118, 145, 160]). Few authors have addressed the problem

of model validation for nonlinear systems.

West [165] considered model validation for nonlinear systems by studying nonlinear signal distortion correlation. This study was limited to characteristics of static nonlinearities [21]. West split the output from the nonlinear element into two portions: one proportional to the input signal and the other a distortion noise. He then showed that there is no correlation between the input and distortion signals whenever the input belongs to a separable class of random process. Douce [44] investigated this further and proved that the same property occurs for a specific class of nonlinear dynamic systems. Douce developed a system identification technique based on cross-correlating the residuals with a test signal obtained by passing the system input through a pre-specified nonlinearity [21]. These techniques are used to assess if model residuals contain any unmodeled dynamics, a test of model “goodness”.

Model validity involves detecting terms in residuals, which, if ignored will cause bias in the parameter estimates. Traditional linear techniques for model validation, based on covariance tests, can easily be shown to be inadequate for nonlinear systems, i.e., residuals are not Gaussian, white, zero-mean. This was illustrated by Billings and Voon [21] with the following example.

Assume in the identification of a system the following terms were inadvertently omitted and hence appear in the residuals, $\xi(n)$, as

$$\xi(n) = \theta u(n-1)e(n-1) + e(n) \quad (2.88)$$

where $e(n)$ is Gaussian, white noise and $e(n)$ and $u(n)$ are independent zero-mean. Computing the normalized auto-correlation function of the residuals and the normalized cross-correlation function between the system input and residuals gives [21]

$$\begin{aligned} \phi_{\xi\xi}(\tau) &= \delta(\tau) \\ \phi_{u\xi}(\tau) &= 0 \quad \forall \tau. \end{aligned} \quad (2.89)$$

Using standard linear identification criterion, these residuals are considered to contain no further information and appear white. However, Equation 2.88 clearly shows that

unmodeled dynamics exist in the residuals and will undoubtedly introduce bias into the parameter estimates [19, 21, 28]. This clearly demonstrates that linear covariance techniques do not, in general, detect predictable nonlinear effects [12, 13, 17, 21].

An alternate approach would be to use multidimensional correlation functions such as $\phi_{\xi\xi\xi}(\tau_1, \tau_2)$, $\phi_{uu\xi}(\tau_1, \tau_2)$ and $\phi_{u\xi\xi}(\tau_1, \tau_2)$ to check for nonlinear terms in the residuals [12, 25]. However, this approach involves two-dimensional correlations and greatly increases computation. The approach could be extended to higher dimensional cases but is clearly unrealistic in practice.

Alternatively an r dimensional correlation function can be projected into a single index higher-order correlation function with r points [25].

Billings and Voon (1983) [21] developed a model validation technique for nonlinear systems based on this principle. They argued that residuals will be unpredictable from all past inputs and outputs if, and only if [21],

$$\begin{aligned}\overline{\phi_{\xi\xi}}(\tau) &= \delta(\tau) \\ \overline{\phi_{u\xi}}(\tau) &= 0 \quad \forall \tau \\ \overline{\phi_{\xi\xi u}}(\tau) &= 0 \quad \forall \tau\end{aligned}\tag{2.90}$$

where the overbar '-' is used to indicate a zero-mean process.

These criteria are based on correlation coefficient functions of sampled input-output systems computed according to the formula [21]

$$\hat{\phi}_{xy}(\tau) = \frac{\frac{1}{N} \sum_{n=1}^{N-\tau} (x(n) - \bar{x})(y(n+\tau) - \bar{y})}{\sqrt{\phi_{xx}(0)\phi_{yy}(0)}} \quad -1 \leq \hat{\phi}_{xy}(\tau) \leq 1.\tag{2.91}$$

In practice, confidence intervals are used to determine whether the correlation between variables is significant. If N is large the standard deviation of the correlation estimate is $1/\sqrt{N}$ and the 95% confidence limits are, therefore, approximately $\pm 1.96/\sqrt{N}$ [21].

The tests in Equation 2.90 can only be applied if a noise model is fitted as part of the estimation procedure so that $\xi(n)$ is an unpredictable sequence. When IV or SOLS routines are used only an estimate of the process model is obtained and, therefore, alternative model validation tests are needed. Billings and Voon (1986)

[23] derived validation tests for IV and SOLS methods. The details of these tests are not provided here.

Korenberg and Hunter [86, 88] showed several counter-examples to the work of Billings and Voon [21, 23], illustrating that the Billings and Voon results are incorrect.

Suppose the true test system has the following output [86, 88]

$$y(n) = u(n) + u^3(n-1) - 3u(n-1) \quad (2.92)$$

where u is zero-mean, white, Gaussian input with unity variance. The terms $u(n-j)$ for $j = 1, 2, \dots, N$ and $u^3(n-1) - 3u(n-1)$ will then be mutually orthogonal for all j [86, 88]. If the system is fit by a linear model; estimating coefficient θ_1 by minimizing the mean-square error

$$E \left[\{y(n) - \hat{\theta}_1 u(n)\}^2 \right] \quad (2.93)$$

it is seen that $\hat{\theta}_1 = 1$ and the residuals will be $\xi(n) = u^3(n-1) - 3u(n-1)$ [86, 88]. This residual sequence is an unpredictable sequence since $u(n)$ is white, Gaussian [86, 88]. In addition, the residual has zero-mean and its normalized auto-correlation is a δ -function [86, 88]. Korenberg and Hunter [88] showed that the three cross-correlation tests in Equation 2.90 are satisfied for this simple example [86]. However, the linear model is clearly not valid and the residual is completely predictable from the input.

Since the correlation based methods discussed above are necessary but *not* sufficient conditions to determine model “goodness”, a complementary measure is typically computed to validate identified models. This measure is commonly known as cross-validation. Cross-validation evaluates the predictive capability of a model using fresh data. The ideal situation is when the predicted outputs are capable of explaining a major part of the actual (measured) output. The ratio

$$R_z^2 = 1 - \frac{\sum_{n=1}^N \hat{\epsilon}_N^2(n)}{\sum_{n=1}^N z_N^2(n)} \quad (2.94)$$

measures the proportion of the total variation of z that is explained by the regression.

This measure is also known as the squared multiple correlation coefficient and is often expressed in percent.

2.6.4.1 Summary

Although Korenberg and Hunter [86, 88] have shown that the techniques for model validation of nonlinear systems provided by Billings and Voon [21, 23] are only necessary but not sufficient conditions for nonlinear systems, they should still be implemented to check for any obvious modeling errors. In addition, a model validation test such as the one given in Equation 2.94 should also be part of any validation procedure. However, a high R_z^2 should be viewed with skepticism until extensive tests of the system are performed.

Chapter 3

NARMAX Representation of Ankle Dynamics

3.1 Introduction

Traditional approaches to nonlinear system identification of human ankle dynamics have relied on quasi-linear methods, e.g., IRF method [78]. These methods provide a convenient, robust means of characterizing the dynamics of nonlinear systems without requiring *a priori* assumptions regarding the system structure. However, nonparametric techniques may require many parameters to describe even simple systems and can be difficult to relate to the structure and parameters of the underlying physiological system.

Although the NARMAX structure is capable of modeling a wide class of nonlinear systems, to date it has been used mostly for control where the main objective is to achieve a parsimonious system description. In biological modeling the objective is more often to gain insight into the function of the underlying system. Therefore, in this chapter, we (1) theoretically analyze a parallel pathway model of ankle dynamics to derive its NARMAX representation, (2) assess the applicability of this nonlinear model for the identification of biological systems and (3) determine the suitability of NARMAX identification methods applied to ankle dynamics.

3.2 Parallel Pathway Model of Ankle Dynamics

Our laboratory, the Neuromuscular Control Laboratory, has developed a parallel pathway model (Figure 3.1) to describe ankle dynamics [79]. The upper, linear pathway

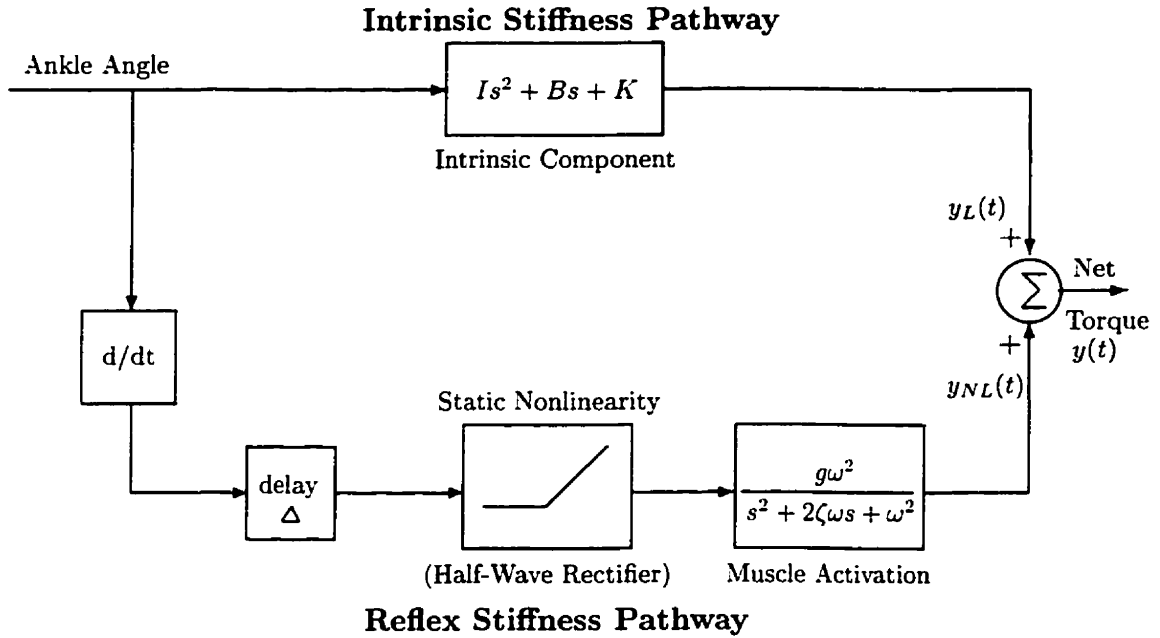


Figure 3.1: Model structure assumed for identification of intrinsic and reflex contributions to overall ankle torque. Redrawn from [79].

models intrinsic stiffness as a second-order system with parameters corresponding to inertia (I), viscosity (B) and elasticity (K). The lower, nonlinear pathway models reflex stiffness as a cascade of a derivative, a time delay, a static nonlinearity (i.e., half-wave rectifier), and a low-pass system. The latter is simplified to second-order, though in many cases it has been shown to be better represented by a third-order filter [108, 109]. A second-order model is justified for the reflex path since we assume that the “system” is a normal subject under passive conditions [108, 109]. The parameters associated with the low-pass system are damping parameter (ζ), natural frequency (ω) and gain (g).

3.2.1 Discrete-Domain Approximation to a Derivative vs. Bilinear Transform

The discrete-domain approximation to a derivative (Newton's backwards formula) maps points from the left-half s -plane into a circle of radius $1/2$, centered at $z = 1/2$ in the z -plane [66, 121]. Since this mapping confines the discrete-time poles to low frequencies, its use is restricted to systems with low resonant frequencies [121]. This is a good approximation to a derivative given the bandlimit of interest is confined to low frequencies. For our work the bandlimit of interest is 0.15 of the sampling rate (see § 3.3 & 3.3.2); therefore, it is appropriate for approximating the intrinsic component of ankle dynamics. Conversely, the bilinear transform maps the left-half s -plane into the entire unit circle and, hence, does not have the same restrictions as above. The bilinear transform gives a better fit to the transient portion of a step response than does the discrete-domain approximation [66, 121]. For this reason it is used to transform muscle activation dynamics, modeled as an IIR system, in Figure 3.1.

However, in the FIR case (intrinsic stiffness pathway Figure 3.1) the bilinear transform cannot be used because, in general, it cannot transform an all-zero system into a stable discrete equivalent. (The bilinear transform is valid up to half the sampling rate, i.e., the Nyquist frequency.) Using the bilinear transform, a derivative operator in continuous-time transforms into a pole-zero system in discrete-time with a pole at $z = -1$. This discrete pole maps back to an unstable pole on the $j\omega$ -axis in the s -plane. For this reason, the derivative operator is transformed to discrete-time using Newton's backwards formula [8].

3.2.2 Theoretical Analysis

The two pathways can be decoupled and analyzed separately since they are summed to yield the net torque.

The discrete-domain approximation to a derivative (Newton's backwards formula

[8])

$$s = \frac{d u(t)}{dt} \approx \frac{u(n) - u(n-1)}{T} \quad \text{where } T \equiv \text{sampling rate} \quad (3.1)$$

was used to approximate the intrinsic pathway dynamics. In the nonlinear path the first derivative in the cascade was approximated using the same derivative approximation as in the linear path. The continuous-time delay was converted to discrete-time as $\tau = \frac{\Delta}{T}$, where Δ is the continuous-time delay and T the sampling rate. The static nonlinearity was approximated as $c_0 + c_1x + c_2x^2$. The activation dynamics were converted to discrete-time via the bilinear transform

$$s = \frac{2}{T} \left(\frac{z-1}{z+1} \right) \quad (3.2)$$

where T is the sampling rate.

After collecting terms and combining, the overall nonlinear model was represented as a nonlinear difference equation with 19 terms as

$$\begin{aligned} y(n) = & b_0 + b_1y(n-1) + b_2y(n-2) + b_3u(n) + b_4u(n-1) \\ & + b_5u(n-2) + b_6u(n-3) + b_7u(n-4) + b_8u(n-\tau) \\ & + b_9u(n-\tau-1) + b_{10}u(n-\tau-2) + b_{11}u(n-\tau-3) \\ & + b_{12}u^2(n-\tau) + b_{13}u^2(n-\tau-1) + b_{14}u^2(n-\tau-2) \\ & + b_{15}u^2(n-\tau-3) + b_{16}u(n-\tau)u(n-\tau-1) \\ & + b_{17}u(n-\tau-1)u(n-\tau-2) + b_{18}u(n-\tau-2)u(n-\tau-3). \end{aligned} \quad (3.3)$$

This is a NARMAX model since (1) it includes input-output terms that are combinations of linear, nonlinear and cross-products and (2) is linear-in-the-parameters.

Table 3.1 shows the relationship of discrete-time NARMAX parameters in Equation 3.3 to the underlying continuous-time coefficients. Note that in this case many of the coefficients are related to each other. This will be addressed later (see § 3.3.3).

NARMAX Coefficient	Linear Relationships	Relationship to Continuous-time Coefficient
b_0		$\frac{4c_0g\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_1		$-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_2		$-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_3		$\frac{I}{T^2} + \frac{B}{T} + K$
b_4		$(\frac{-2I}{T^2} - \frac{B}{T}) - ((-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2} + \frac{B}{T} + K))$
b_5		$(\frac{I}{T^2}) - ((-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{-2I}{T^2} - \frac{B}{T}))$ $-((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2} + \frac{B}{T} + K))$
b_6		$-(-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2}) - ((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{-2I}{T^2} - \frac{B}{T}))$
b_7		$-((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2}))$
b_8	$b_8 = b_9 = -b_{10} = -b_{11}$	$\frac{g\omega^2T^2c_1}{(4+\omega^2T^2+4\zeta\omega T)T}$
b_9		$\frac{g\omega^2T^2c_1}{(4+\omega^2T^2+4\zeta\omega T)T}$
b_{10}		$\frac{-g\omega^2T^2c_1}{(4+\omega^2T^2+4\zeta\omega T)T}$
b_{11}		$\frac{-g\omega^2T^2c_1}{(4+\omega^2T^2+4\zeta\omega T)T}$
b_{12}		$\frac{g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{13}	$b_{13} = 3b_{12}$	$\frac{3g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{14}	$b_{14} = 3b_{12}$	$\frac{3g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{15}	$b_{15} = b_{12}$	$\frac{g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{16}	$b_{16} = -2b_{12}$	$\frac{-2g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{17}	$b_{17} = -4b_{12}$	$\frac{-4g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$
b_{18}	$b_{18} = -2b_{12}$	$\frac{-2g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$

Table 3.1: Theoretical relationship of NARMAX parameters to continuous-time system coefficients for parallel pathway model of ankle dynamics.

3.3 Simulations

Accuracy of this system representation was validated by simulating the parallel pathway model in continuous-time using Simulink (see Figure 3.2). The parameters used

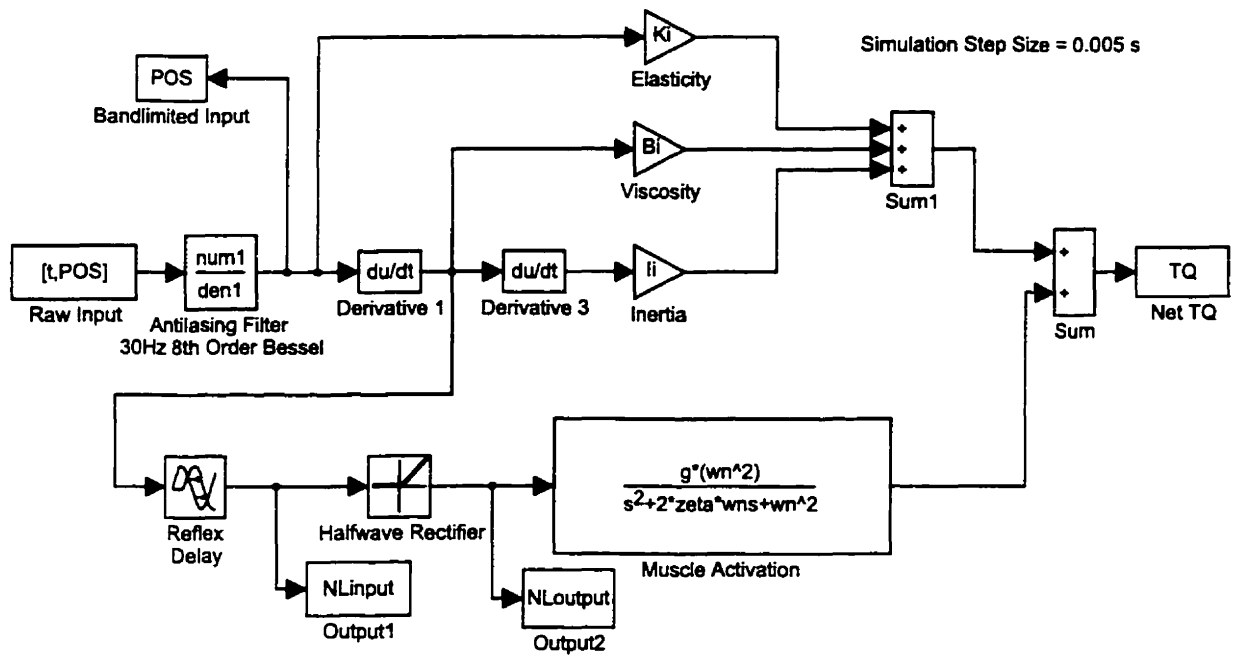


Figure 3.2: Simulink model of parallel pathway ankle model in continuous-time.

in the simulation were typical values found in experiments (Tables 3.2 & 3.3) [79]. The system was perturbed using a 30 Hz bandlimited, uniformly distributed, white, zero-mean, random input.

3.3.1 Output Accuracy of NARMAX Ankle Model

To determine the validity of this NARMAX description model (Equation 3.3) we simulated its response for a parameter set corresponding to those used for the continuous-time model. The input sequence was a bandlimited, uniformly distributed, white,

CT Coefficient	Value
I	0.015 Nm/s ² /rad
B	0.800 Nm/rad/s
K	150 Nm/rad
ω	40.0
ζ	1.00
g	10.00 Nm/rad/s
Δ	0.045 s

Table 3.2: Continuous-time coefficient values. I: inertia, B: viscosity, K: elasticity, ω : natural frequency, ζ : damping parameter, g : reflex stiffness gain and Δ : reflex delay.

NL Coeff.	Value
c_0	2.46
c_1	0.500
c_2	0.016
T	0.005 s

Table 3.3: Coefficient values of static nonlinearity. c_0 : DC term, c_1 : linear term, c_2 : squared term and T: sampling interval.

zero-mean, random input, low-pass filtered with an eighth-order 30 Hz Bessel filter. The bandlimited input had an operating range between ± 0.40 rad (see left panel of Figure 3.4).

To compute a “theoretical” parameter set for this NARMAX model the half-wave rectifier was approximated, using a least-squares fit, as a second-order static polynomial. This second-order fit accounted for over 98% of the output variance of the static nonlinearity (“NLoutput” in Figure 3.2). The operating range of velocity input (“NLinput”) was between ± 30 rad/s and the position input (“POS”) was between ± 0.40 rad. A plot of this second-order approximation to the true half-wave rectifier is shown in Figure 3.3. Assuming a second-order nonlinearity, the frequency content

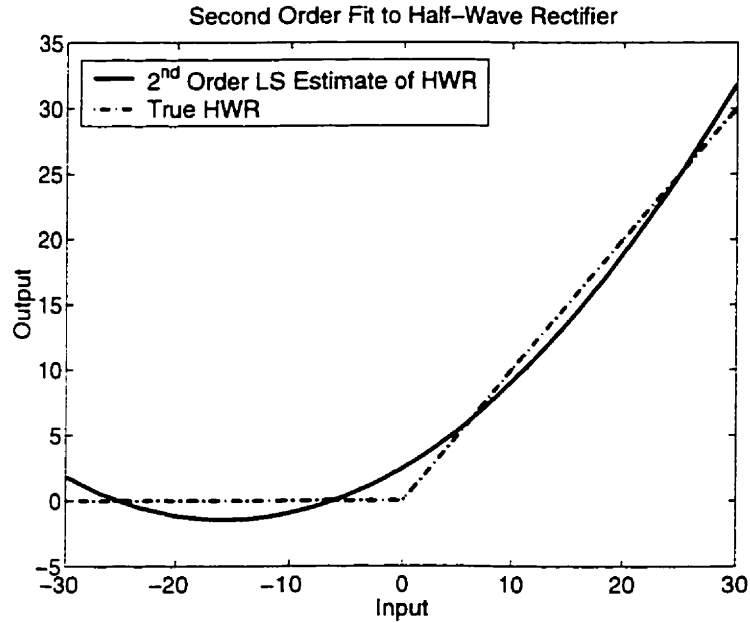


Figure 3.3: Second-order least-squares approximation to the half-wave rectifier used in simulations.

of the signal at the output of the half-wave rectifier (“NLoutput”) will be at least 60 Hz (plus higher-order harmonics). To avoid internal aliasing, we selected a sampling rate of $T = 0.005$ s (200 Hz); 3.3 times greater than the internal 60 Hz signal.

The estimated output ($\hat{y}$) of the NARMAX description model was compared with the output of the continuous-time simulation (y) by computing the variance accounted

for by the NARMAX model as the percent normalized mean-squared-error (%NMSE):

$$\%NMSE = \left(1 - \frac{\frac{1}{N} \sum_{n=1}^N (y_n - \hat{y}_n)^2}{\frac{1}{N} \sum_{n=1}^N (y_n)^2} \right) \times 100, \quad (3.4)$$

where N is the record length.

Figure 3.4 shows the simulation input (left panel) and predicted output of the NARMAX description model superimposed on top of the simulated output of the parallel pathway model (right panel). With over 99 % NMSE the NARMAX output

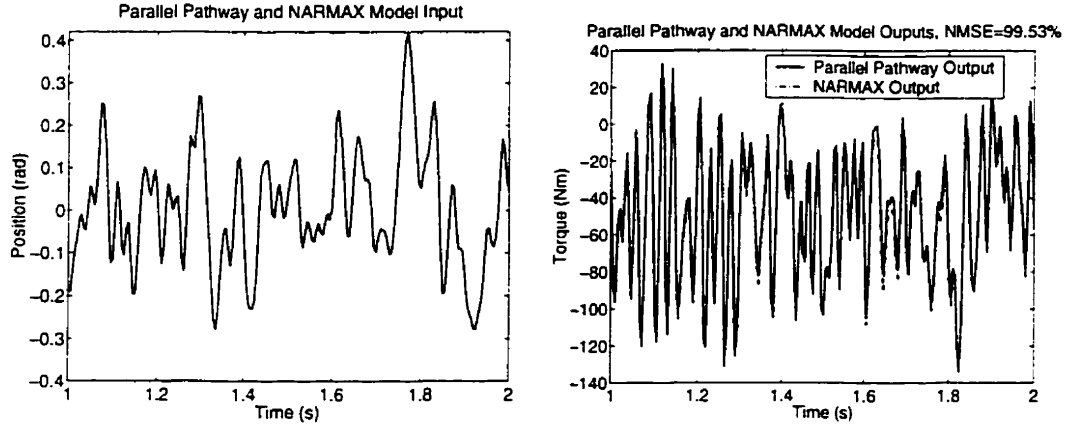


Figure 3.4: Left: Input to simulated parallel pathway model in continuous-time and NARMAX description model. Right: Output of simulated parallel pathway model in continuous-time and NARMAX description model.

matched that of the continuous-time simulation with negligible error [92].

3.3.2 Parameter Estimation of NARMAX Ankle Model

We then assessed the utility of methods developed for identifying NARMAX models using sampled data from this continuous-time simulation. An ELS algorithm [22] was used to identify model parameters.

The NARMAX description of the parallel pathway ankle model (Equation 3.3) is described by past outputs which result in lagged values of disturbance terms in the presence of output additive noise. If these lagged errors are not modeled they induce a bias in the parameter estimates (see §2.6.3.2). The ELS algorithm was

implemented because it is designed to model lagged error terms thereby providing unbiased parameter estimates.

For this study, the system order and structure were assumed to be known with the full coefficient set in Equation 3.3 and Table 3.1. The regressor matrix used by this algorithm was formed to only contain those columns (parameters) that corresponded to our theoretical analysis (Equation 3.3). The estimation set consisted of $N = 4,000$ data points sampled at $T = 0.005$ s. The estimated parameters were cross-validated to compute the %NMSE of the net predicted torque. The validation set consisted of $N_v = 2,000$ data points.

3.3.2.1 Monte-Carlo Analysis of NARMAX Parameters: Noise-Free Input – Output Additive Noise

A Monte-Carlo study of NARMAX parameters describing ankle dynamics was performed to assess their estimation accuracy and variability. Ten Monte-Carlo simulations were used in which each input-output realization was unique and had a unique noise sequence added to the output. Ten Monte-Carlo trails were chosen due to computational considerations (i.e., time) and because it is convenient for subsequent statistical computations. Each input sequence was bandlimited (uniformly distributed, white, zero-mean, random input, low-pass filtered with an eighth-order 30 Hz Bessel filter) while a unique Gaussian, white, zero-mean noise sequence was added to the output. The output additive noise amplitude was increased in increments of 5 dB, from 30 dB to 0 dB SNR and the input sequence was assumed to be measured with negligible error and, therefore, was noise-free.

Figure 3.5 shows a typical input-output sequence used for this analysis (a noise sequence of 20 dB SNR was added to the torque).

Figure 3.6 shows the results of this study. Each figure shows the standard deviation (STD) about the mean, for the NARMAX parameters given in Table 3.1. These values are plotted against SNR, and the theoretical parameter values are given as a dashed line in each plot. Note that some parameters do not appear to be scaled versions of each other since the NARMAX model is over-parameterized. This suggests that

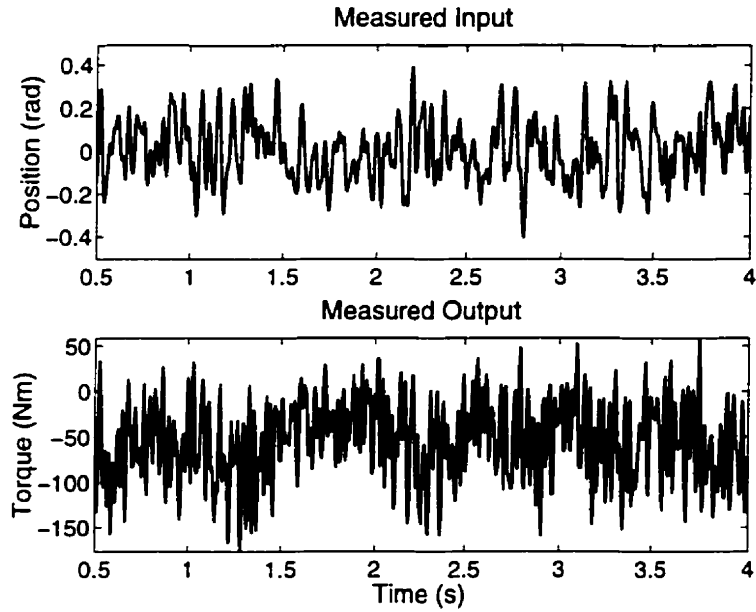


Figure 3.5: Typical input-output sequence for parallel pathway ankle model. Upper: Position input. Lower: Measured torque output (sum of true system output and Gaussian, white, zero-mean noise sequence with 20 dB SNR).

the regressor matrix may be almost singular. These figures show that the identified parameter values did not correspond closely to those derived theoretically. As the SNR was decreased the identified parameter's bias and random error increased.

The results illustrated in Figure 3.6 at first glance could cause some concern for the applicability of NARMAX identification methods to ankle dynamics or other biological systems in which the SNR may be poor. However, this may not necessarily be the case. Below, we investigate whether NARMAX identification can lead to accurate results, in the presence of significant noise, by looking at the effects of data length and the effects of model over-parameterization.

3.3.2.2 Monte-Carlo Analysis of NARMAX Model: Increased Record Length

For parameter estimation, we implement a least-squares algorithm (i.e., ELS). Least-squares algorithms are well known to have good asymptotic properties, i.e., when N is large [26, 35, 45, 61, 68, 113, 133, 147]. Therefore, we investigated whether the full (over-parameterized) model (Equation 3.3) could provide unbiased parameter

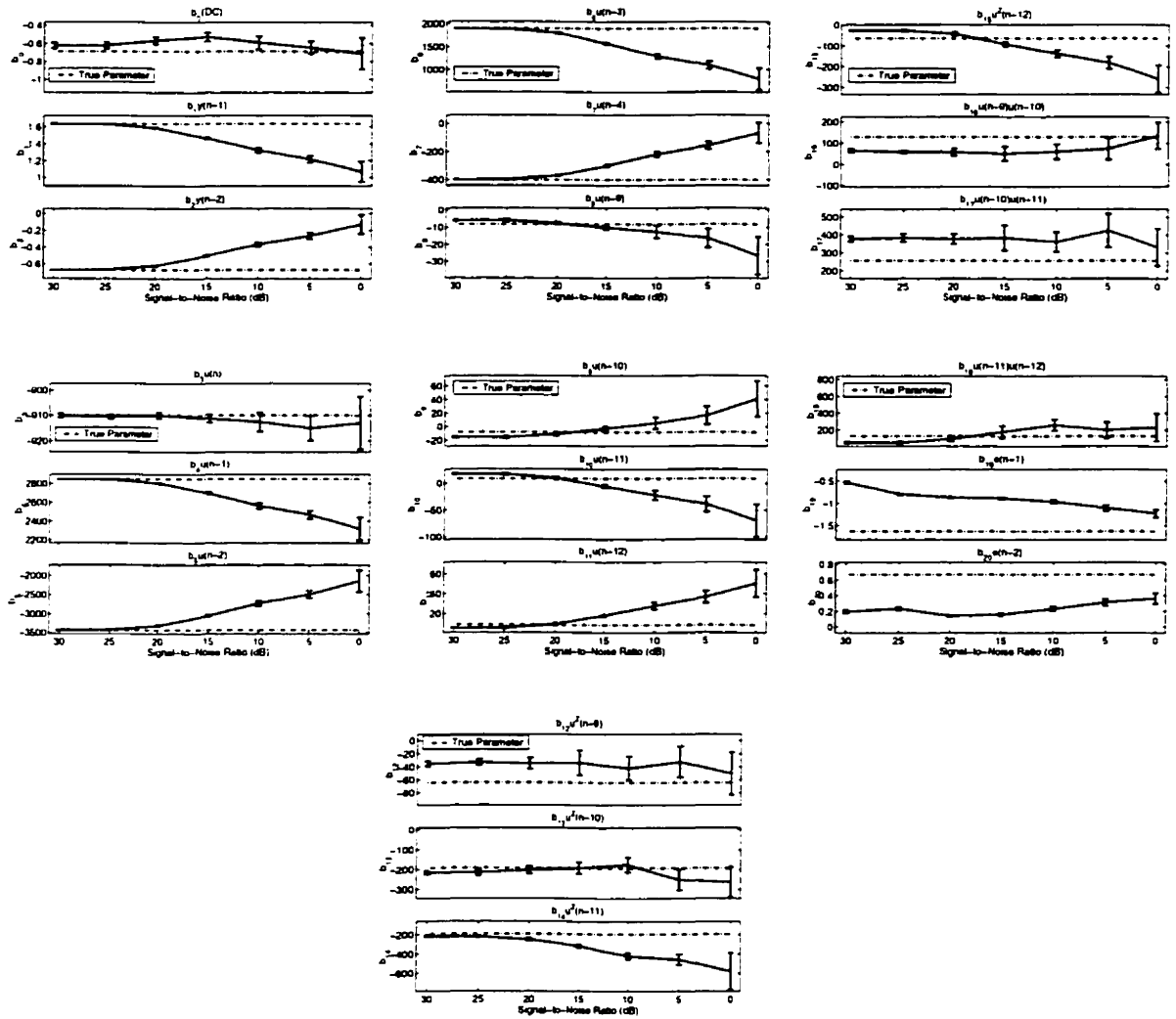


Figure 3.6: Full NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=4,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. Theoretically expected parameter relationships: $b_8 = b_9 = -b_{10} = -b_{11}$; $b_{13} = 3b_{12}$; $b_{14} = 3b_{12}$; $b_{15} = b_{12}$; $b_{16} = -2b_{12}$; $b_{17} = -4b_{12}$; $b_{18} = -2b_{12}$. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

estimates when the data length was increased.

In this study, the same input-output realizations were used as in the previous study but the data length was increased three-fold to $N = 12,000$ points. Figure 3.7 shows the results of this study. The results show that when N was increased by a factor of three the identified parameters were still biased in their mean but did have less variability. The reason is the correlation between some current NARMAX coefficients in this description of ankle dynamics.

3.3.3 Reduction of Dimensionality

The previous analysis shows NARMAX parameters deviated significantly from their true mean for almost all levels of SNR. Therefore, we investigated the effect of reducing the number of terms required to describe this NARMAX model. This is not a general reduction of terms to describe the data but rather a minimization of the number of regressors or d.o.f.'s (degree of freedom) used to form the regressor matrix. This reduction should provide a regressor matrix that is more stable in terms of invertibility since the coefficients will no longer be interrelated [60].

The coefficients of the full NARMAX model (Equation 3.3 & Table 3.1) are partially redundant and, therefore, its input-output description can be redefined. Re-combining all terms in Equation 3.3 according to coefficients of the static nonlinearity (c_2 , c_1 and c_0) yields an overall nonlinear model represented by 10 terms as

$$\begin{aligned}
y(n) &= b_0 + b_1 y(n-1) + b_2 y(n-2) + b_3 u(n) + b_4 u(n-1) \\
&+ b_5 u(n-2) + b_6 u(n-3) + b_7 u(n-4) + m_1 [u(n-\tau) \\
&+ u(n-\tau-1) - u(n-\tau-2) - u(n-\tau-3)] \\
&+ m_2 [u^2(n-\tau) + 3u^2(n-\tau-1) + 3u^2(n-\tau-2) \\
&+ u^2(n-\tau-3) - 2u(n-\tau)u(n-\tau-1) \\
&- 4u(n-\tau-1)u(n-\tau-2) - 2u(n-\tau-2)u(n-\tau-3)] \\
&= b_0 + b_1 y(n-1) + b_2 y(n-2) + b_3 u(n) + b_4 u(n-1) \\
&+ b_5 u(n-2) + b_6 u(n-3) + b_7 u(n-4) + m_1 v(n) + m_2 \chi(n)
\end{aligned} \tag{3.5}$$

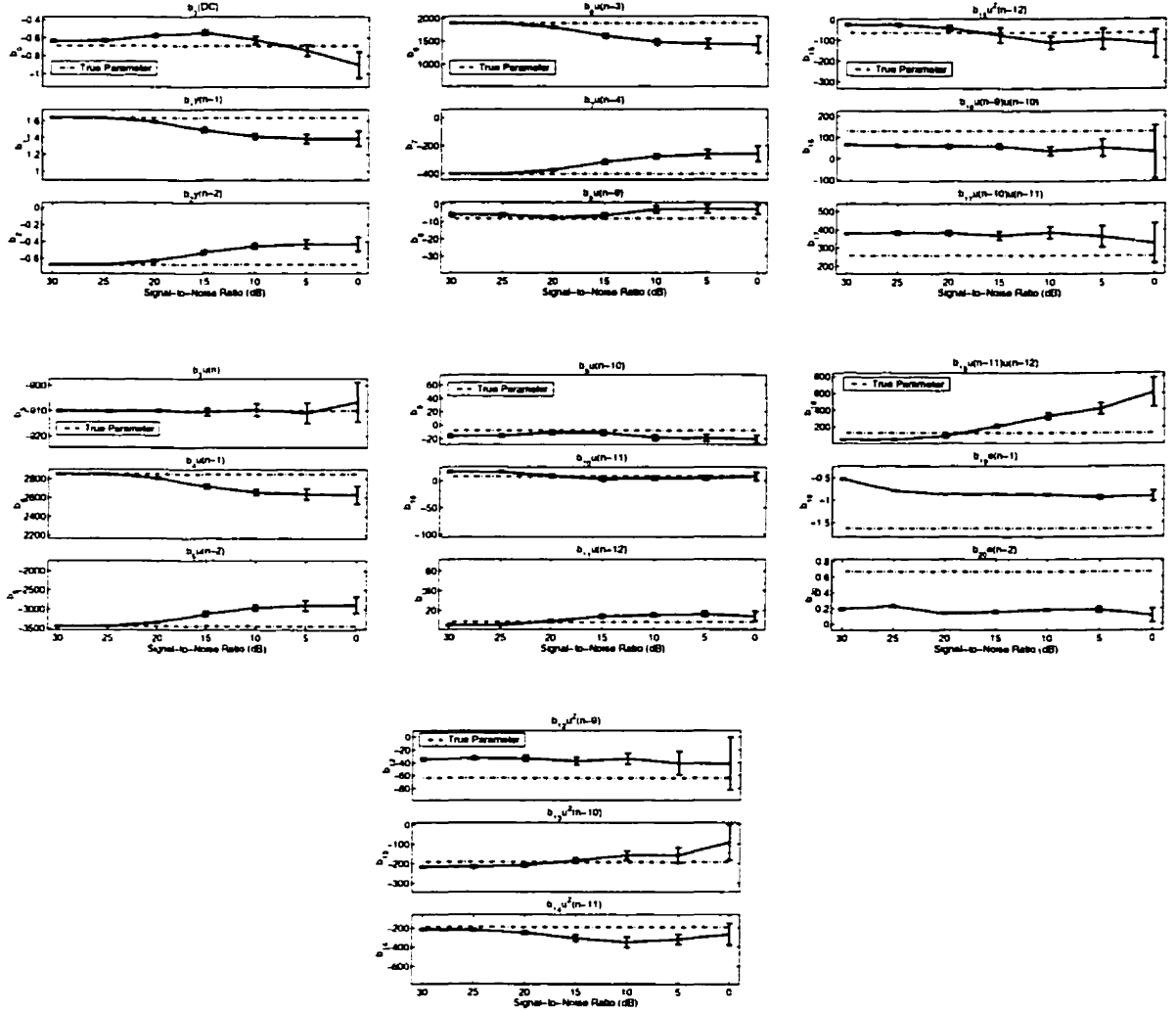


Figure 3.7: Full NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=12,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. Theoretically expected parameter relationships: $b_8 = b_9 = -b_{10} = -b_{11}$; $b_{13} = 3b_{12}$; $b_{14} = 3b_{12}$; $b_{15} = b_{12}$; $b_{16} = -2b_{12}$; $b_{17} = -4b_{12}$; $b_{18} = -2b_{12}$. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

where

$$v(n) = u(n - \tau) + u(n - \tau - 1) - u(n - \tau - 2) - u(n - \tau - 3) \quad (3.6)$$

and

$$\begin{aligned} \chi(n) &= u^2(n - \tau) + 3u^2(n - \tau - 1) + 3u^2(n - \tau - 2) + u^2(n - \tau - 3) \quad (3.7) \\ &- 2u(n - \tau)u(n - \tau - 1) - 4u(n - \tau - 1)u(n - \tau - 2) \\ &- 2u(n - \tau - 2)u(n - \tau - 3). \end{aligned}$$

Table 3.4 gives the relationships of these discrete-time NARMAX parameters (Equation 3.5) to the underlying continuous-time coefficients. This reduced set of coefficients now has the same number of degrees of freedom as its initial continuous-time description (the “extra” coefficient T denotes the sampling rate).

NARMAX Coefficient	Relationship to Continuous-time Coefficient
b_0	$\frac{4c_0g\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_1	$-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_2	$-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T}$
b_3	$\frac{I}{T^2} + \frac{B}{T} + K$
b_4	$(\frac{-2I}{T^2} - \frac{B}{T}) - ((-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2} + \frac{B}{T} + K))$
b_5	$(\frac{I}{T^2}) - ((-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{-2I}{T^2} - \frac{B}{T})) - ((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2} + \frac{B}{T} + K))$
b_6	$-((-\frac{-8+2\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2}) - ((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{-2I}{T^2} - \frac{B}{T})))$
b_7	$-((-\frac{-4\zeta\omega T+4+\omega^2T^2}{4+\omega^2T^2+4\zeta\omega T})(\frac{I}{T^2}))$
m_1	$\frac{g\omega^2T^2c_1}{(4+\omega^2T^2+4\zeta\omega T)T}$
m_2	$\frac{g\omega^2T^2c_2}{(4+\omega^2T^2+4\zeta\omega T)T^2}$

Table 3.4: Theoretical relationship of compressed NARMAX model parameter set to continuous-time system coefficients.

3.3.3.1 Monte-Carlo Analysis of Compressed NARMAX Model Parameters

A Monte-Carlo study of these reduced NARMAX parameters (Equation 3.5 & Table 3.4) was performed to assess their accuracy and variability. The simulation protocol and input-output data sets used for this analysis was the same as described in §3.3.2.

Figure 3.8 shows the results of this study. The NARMAX parameters in this figure correspond to ones given in Table 3.4. Parameters m_1 and m_2 in Figure 3.8 correspond

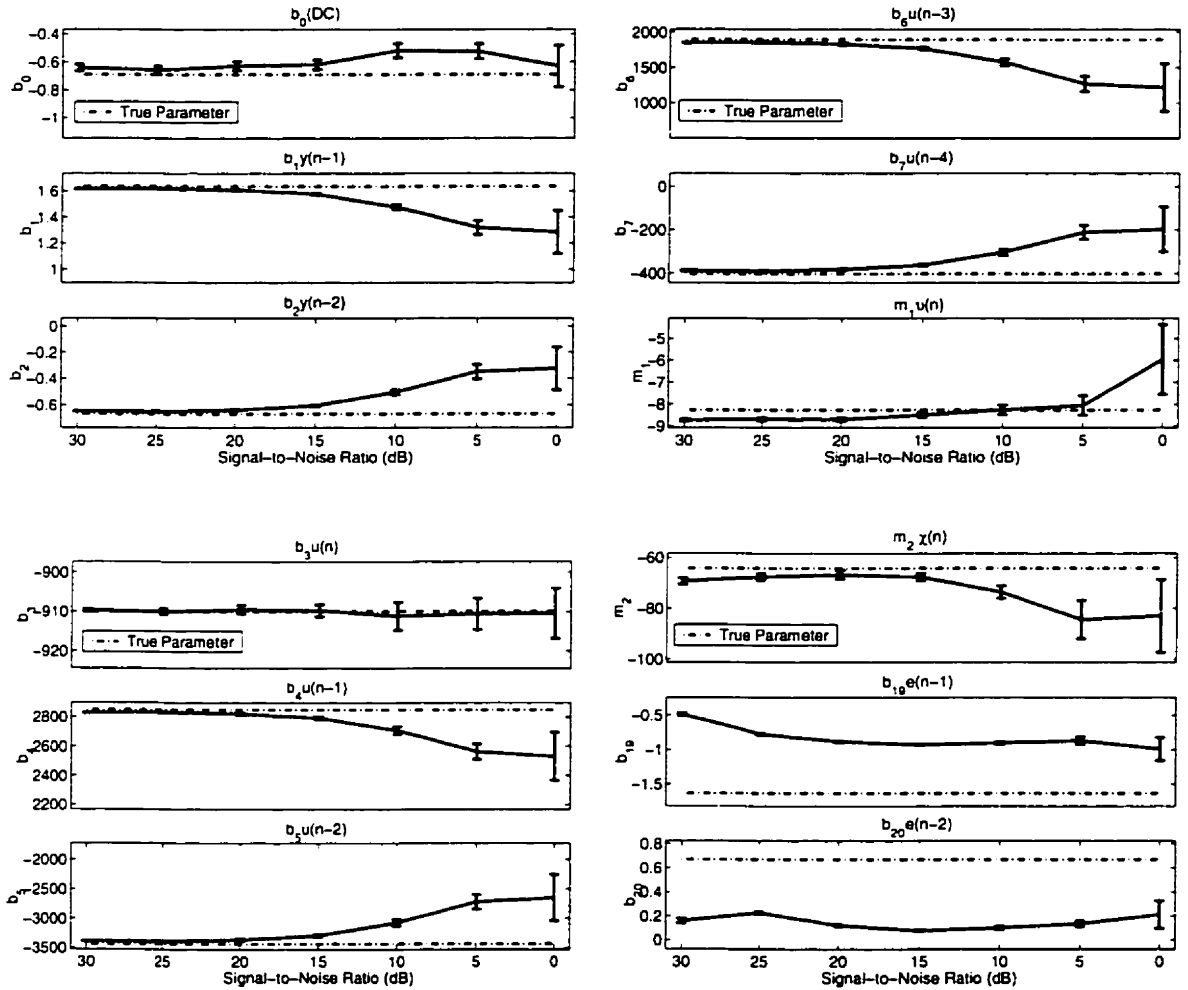


Figure 3.8: Compressed NARMAX model: Bandlimited, zero-mean, random input, Gaussian, white, zero-mean noise and $N=4,000$. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

to parameters $b_8 - b_{11}$ and $b_{12} - b_{18}$ in Figures 3.6 and 3.7, respectively. This figure shows that, when the number of terms describing this NARMAX model was reduced to the appropriate complexity, the identified parameter values corresponded closely to those derived theoretically for SNRs ≥ 20 dB. Note that we expect the mean value of parameters b_{19} and b_{20} to be biased since they correspond to lagged error terms. Lagged error terms are difficult to identify accurately even with high SNR since they model the output additive noise which is a stochastic process and cannot be measured. This stochastic process is modeled (approximated) by a deterministic signal of prediction errors which is only an (poor) estimate of the noise (see §2.6.3.3).

Figure 3.9 shows a cross-validated (predicted) output superimposed on top of the measured output for a typical parameter set. The predicted output matched the

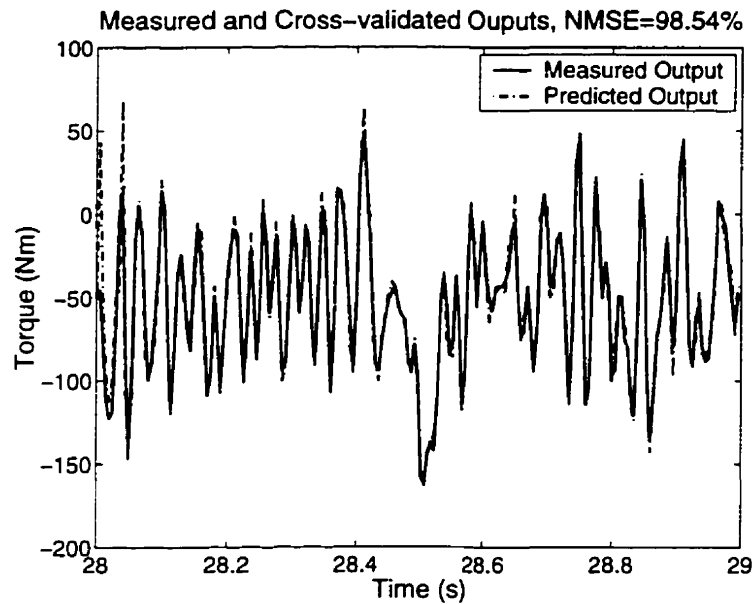


Figure 3.9: Cross-validation: Measured and predicted output for identified NARMAX ankle model with $N_v = 2,000$ and Gaussian, white, zero-mean output additive noise (20 dB SNR).

measured output with over 98% NMSE.

3.3.4 Estimation of Continuous-Time Parameters of Ankle Model

The previous section demonstrates that existing parameter estimation methods yield good results for NARMAX models at low noise levels (i.e., high SNR values). In biological modeling, discrete-time NARMAX parameters may not be relevant or may be difficult to interpret since the biology is often evaluated in terms of physical variables. Therefore, we evaluated (1) whether it is possible to compute the underlying continuous-time parameters when the model structure is predetermined and (2) whether these continuous-time parameters are closer to the true mean and have less variability than ones estimated using traditional nonparametric techniques [79].

Currently, in our laboratory a nonparametric identification technique is used to estimate parameters describing ankle dynamics [70, 75, 77, 78, 79, 80]. In this study, estimates of continuous-time parameters computed using our NARMAX approach were compared to results obtained using the nonparametric method used in our laboratory. This nonparametric method implements a Levenberg-Marquardt nonlinear least-squares algorithm to compute continuous-time parameters [79]. This approach requires an initial “guess” of the unknown continuous-time parameters to compute them. The true parameter values were used as an initial seed to emphasize the problem this technique has in computing a global minimum even in a best-case-scenario where the true values are known. In contrast, our nonlinear parametric approach does not require any initial values to compute continuous-time parameters. Employing our approach, continuous-time parameters are computed directly from discrete-time NARMAX estimates using the theoretical relationships given in Table 3.5.

3.3.4.1 Noise-Free Bandlimited Input – Output Additive White Noise

Ten Monte-Carlo simulations were used to assess the reliability of our approach to estimate these continuous-time parameters. The system was simulated as described in §3.3.2, except that a data record of $N = 7,000$ points was used for identification. The discrete-time parameters were identified using the reduced NARMAX model given in

CT Coefficient	DT Relationship
$I =$	$b_7 \times \frac{T^2}{-b_2}$
$B =$	$\frac{b_6 + b_1 \frac{I}{T^2}}{-b_2} = -(\frac{2I}{T^2} + \frac{B}{T}) + \frac{2I}{T^2} = -\frac{B}{T} \times -T$
$K =$	$b_3 - (\frac{I}{T^2} + \frac{B}{T})$
$\omega =$	$\frac{-4 + 4b_2 + 4b_1}{-1 + b_2 - b_1} = \sqrt{(\omega^2 T^2) \times \frac{1}{T^2}}$
$\zeta =$	$\frac{-2 - 2b_2}{-1 + b_2 - b_1} = \frac{\zeta \omega T}{\omega T}$
$g =$	$m_1 \times (4 + \omega^2 T^2 + 4\zeta \omega T) \times T = \frac{g c_1 \omega^2 T^2}{c_1 \omega^2 T^2}$

Table 3.5: Discrete to continuous-time relationships for parameters I , B , K , ω , ζ and g of the parallel pathway ankle model.

Equation 3.5.

The results of this study are presented below. Figure 3.10 presents plots of standard deviation about the mean for estimated continuous-time parameters of the linear and nonlinear path (I , B , K , g , ω and ζ) using NARMAX identification techniques. Figure 3.11 shows the results of identifying continuous-time ankle parameters using nonparametric identification methods [79]. For the linear path, continuous-time parameters estimated using our NARMAX approach were closer to the true mean and had less variability for high SNR ($\gtrsim 10$ dB SNR) than those obtained using nonparametric techniques. Estimates using a nonparametric approach were consistently biased away from the true mean. Variance of the inertial parameter (I), computed using parametric methods, was approximately 5 times smaller for SNR between 30 to 10 dB and approximately equal for SNR between 5 to 0 dB compared to nonparametric estimates. For viscous and elastic parameters (B , K), the variance was approximately 2 – 4 times smaller for 30 dB SNR, equal for SNR between 25 to 15 dB and 10 – 15 times larger for SNR between 10 to 0 dB than nonparametric estimates.

Results show that, for the nonlinear path, continuous-time parameters computed using a NARMAX approach were closer to the true mean for SNR $\gtrsim 20$ dB but were biased at lower SNRs. However, the nonparametric estimates were biased at all levels of SNR. All parameters computed using NARMAX identification had variability equal to those obtained using nonparametric techniques for SNR $\gtrsim 10$ dB. Between 5 and 0

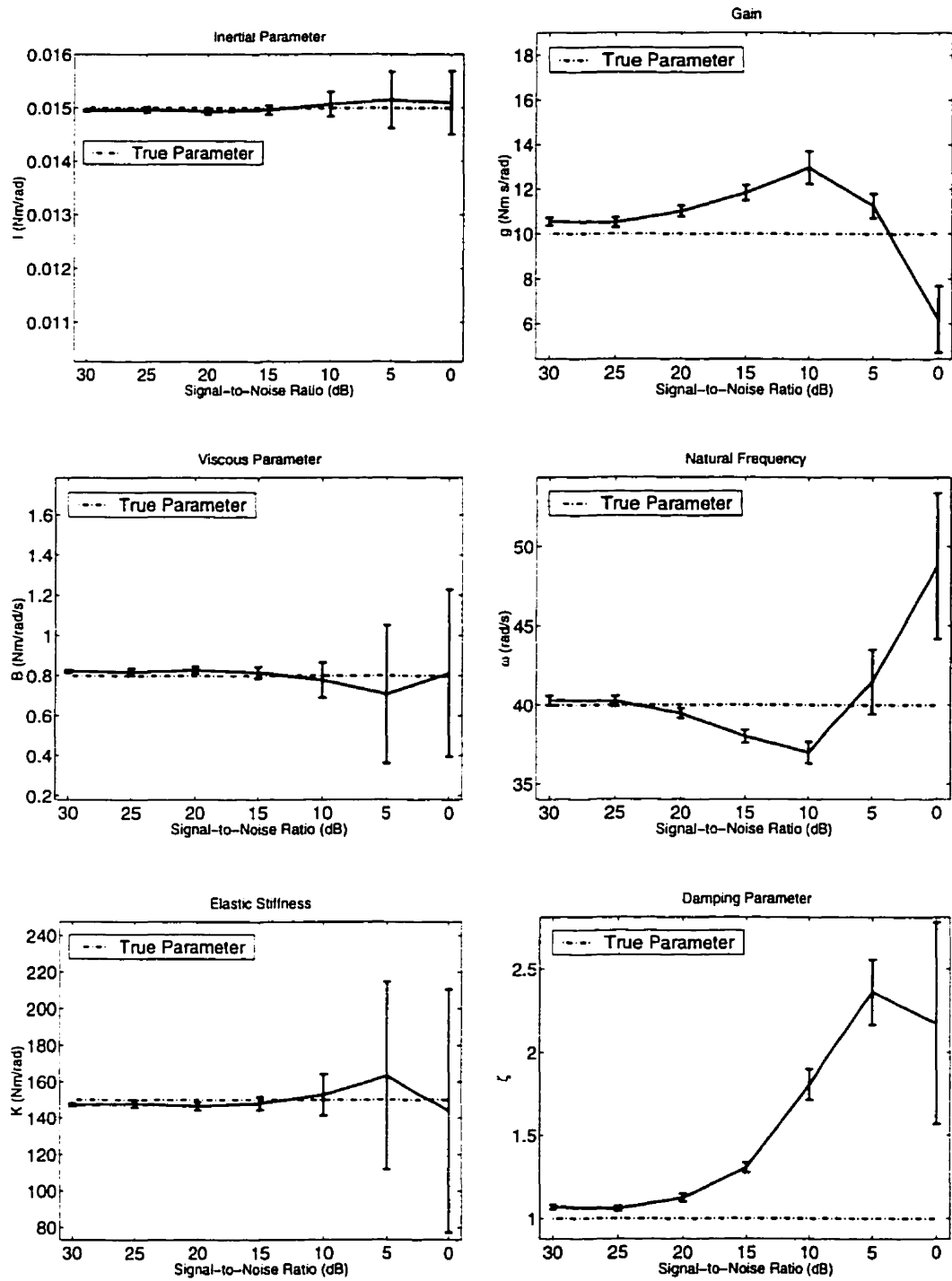


Figure 3.10: Continuous-time parameters of parallel pathway ankle model. Bandlimited, zero-mean, random input and NARMAX identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

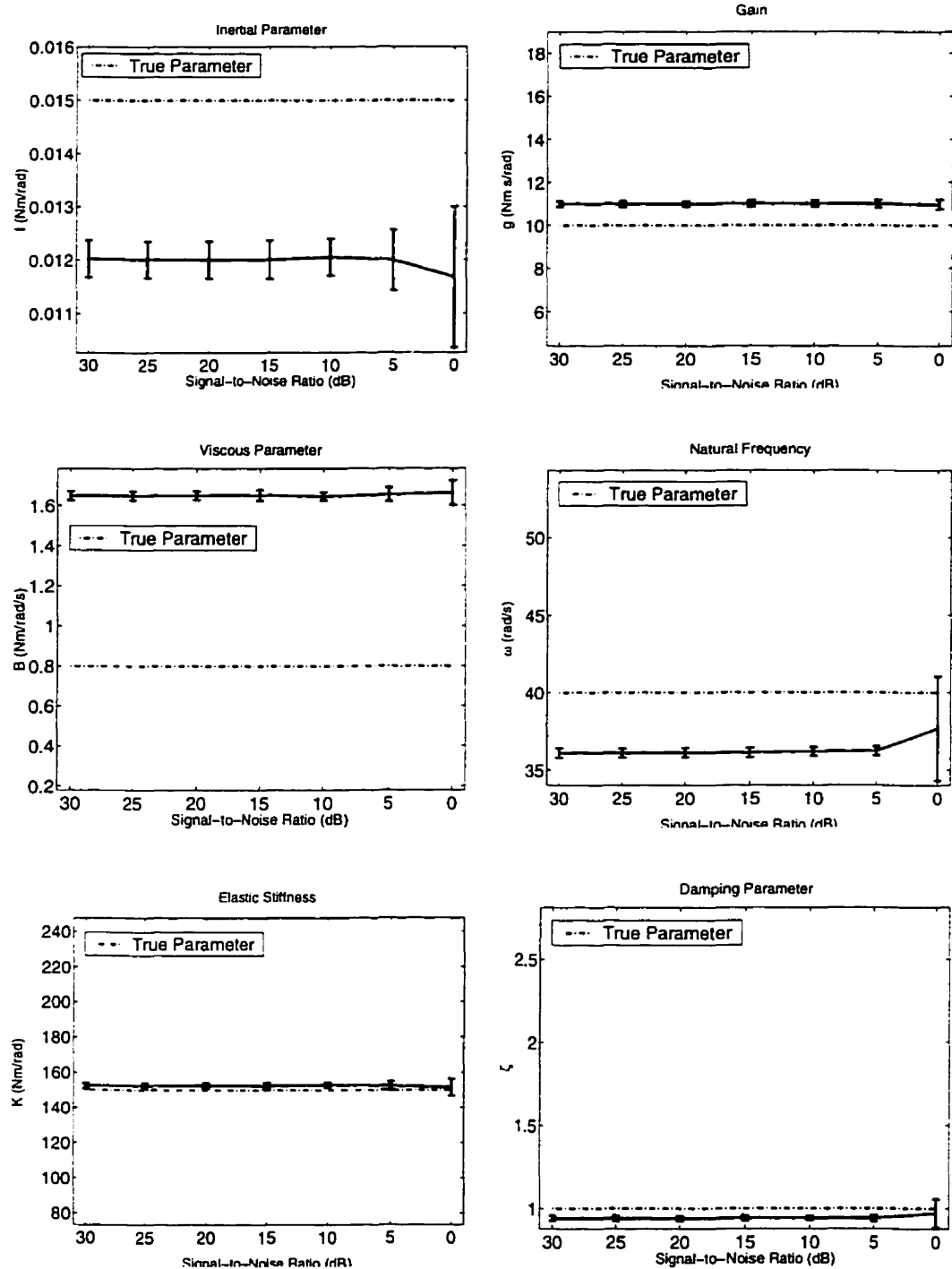


Figure 3.11: Continuous-time parameters of parallel pathway ankle model. Bandlimited, zero-mean, random input and nonparametric identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

dB SNR parameters computed using our identification technique had approximately 10 – 20 times more variability.

3.3.4.2 Noise-Free Colored Input – Bandlimited Output Additive Noise

Assumptions made by least-squares theory do not allow for non-white disturbances. However, in many practical situations assumptions regarding whiteness may be violated or incorrect due to effects of anti-aliasing filters, quantization, etc. [161, 101]. In order to evaluate the effects of bandlimited output additive noise on our estimation technique, we performed ten Monte-Carlo simulations in which the system was simulated using a position input (pseudo-random binary sequence (PRBS)) from experiments conducted in our laboratory. We used a PRBS input for this study because it is the typical input used in our laboratory. The PRBS input was bandlimited with an eighth-order 30 Hz low-pass Bessel filter and the input-output sequence was sampled at $T = 0.005$ s. The output additive disturbance was a zero-mean bandlimited sequence (Gaussian, white, zero-mean sequence, low-pass filtered with an eighth-order 60 Hz Butterworth filter). Each input-output trial and noise sequence was unique. The noise levels were the same as in the bandlimited input, white output additive noise case (see §3.3.4.1), however, $N = 5,000$ points were used for identification. Note that this PRBS input is very non-Gaussian and non-white [78].

Figure 3.12 shows a typical input-output trial used for this analysis. The data represents a PRBS sequence of 0.03 rad (peak-to-peak) and 150 ms switching rate. The characteristics of this trial are consistent with those used for analysis in this section. The torque sequence shown in this figure is the sum of a noise-free torque and a noise sequence with 20 dB SNR.

Plots in Figure 3.13 show results of estimating continuous-time ankle parameters using NARMAX identification techniques.

Figure 3.14 provides plots of ankle parameters computed using nonparametric identification methods. Continuous-time parameters computed for the linear path, using a NARMAX approach, were closer to the true mean for all levels of SNR and had less variability for high SNR ($\gtrsim 10$ dB SNR) than those obtained using

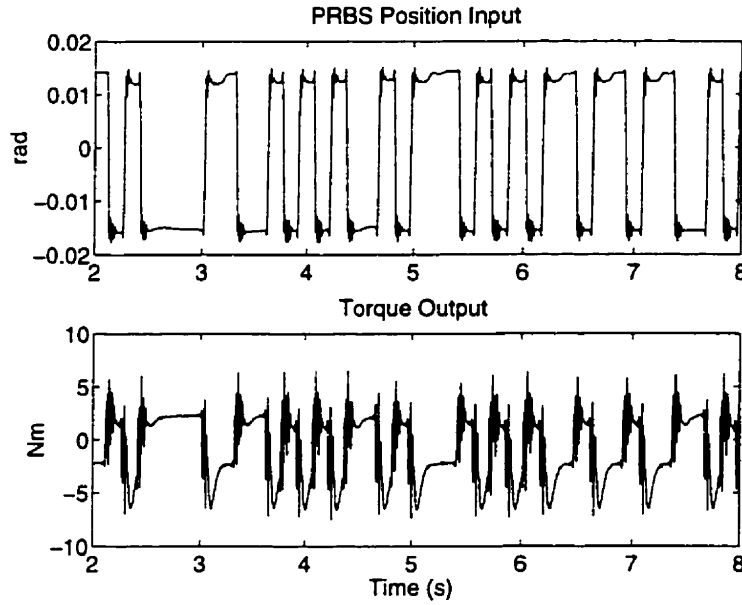


Figure 3.12: Typical position input and torque output recorded from simulation.

nonparametric techniques. Estimates provided by the nonparametric approach were consistently biased away from the true mean. Variance of the inertial parameter (I), computed using this parametric technique, was approximately 200 times smaller for SNR between 30 to 20 dB and 10 – 2 times less for SNR between 15 to 0 dB than its nonparametric counterpart. Using a NARMAX approach, the viscous and elastic parameters (B , K) variance was approximately 10 – 3 times less for SNR between 30 to 15 dB, approximately 10 times larger for SNR between 10 to 5 dB and 50 times larger at 0 dB SNR than nonparametric estimates.

The results show that, using a NARMAX approach, the nonlinear path parameters were all significantly biased except at high SNR ($\gtrsim 20$ dB SNR). However, the bias was not as severe as for nonparametric estimates where all parameters were biased consistently. Nonlinear path parameters computed using our NARMAX method had approximately 7 times less variability for SNR between 30 to 15 dB and equal variance for SNR between 10 to 0 dB as obtained using nonparametric methods.

Figures 3.10 & 3.13 shows the performance of our NARMAX technique in the presence of output additive white and bandlimited noise respectively. With bandlimited noise, coefficients of the linear path were less biased but had greater variability

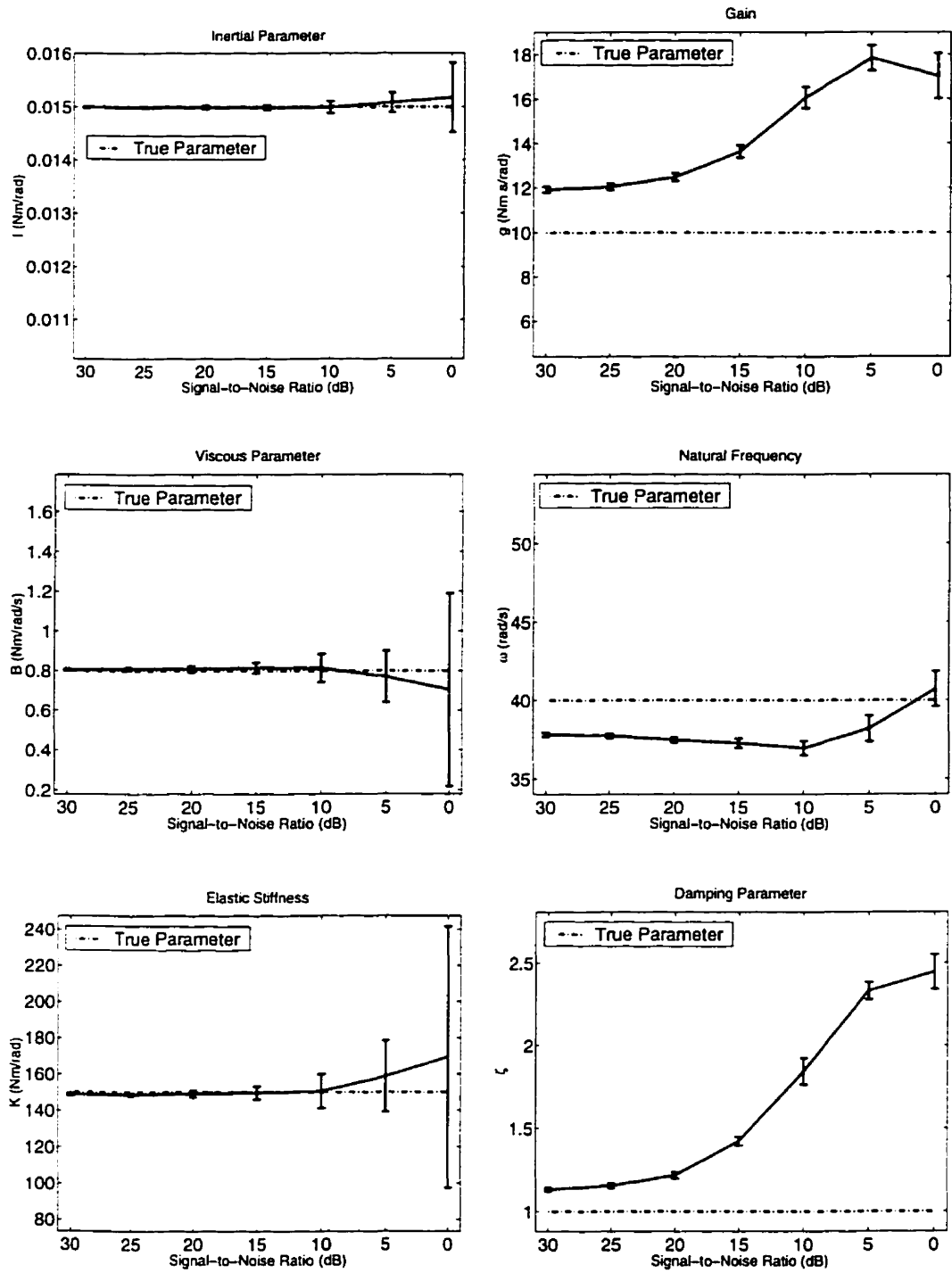


Figure 3.13: Continuous-time parameters of parallel pathway ankle model. PRBS input and NARMAX identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

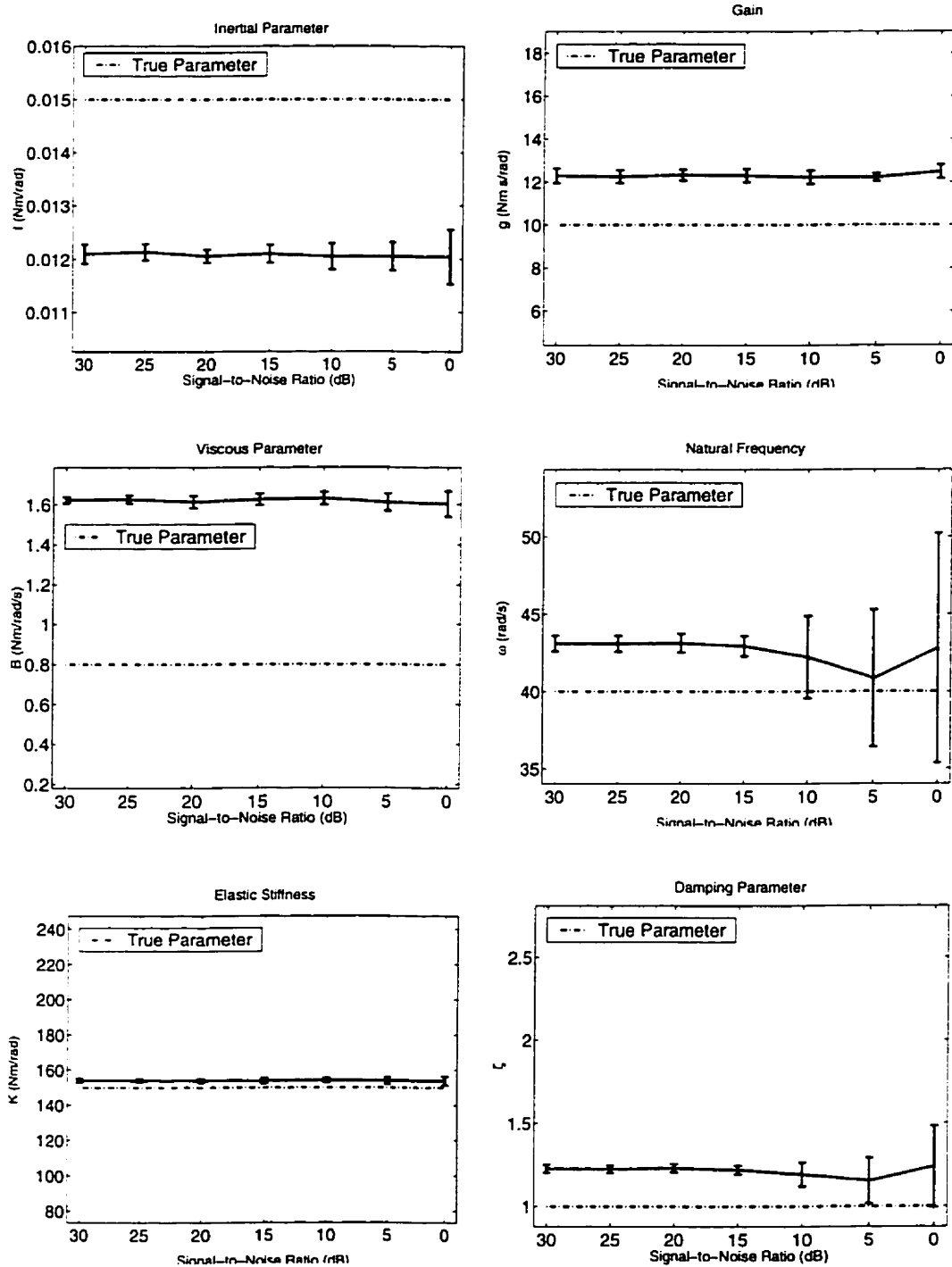


Figure 3.14: Continuous-time parameters of parallel pathway ankle model. PRBS input and nonparametric identification. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

than those with white noise. However, the nonlinear path coefficients were less biased and had greater variability in the presence of white noise than those with bandlimited noise.

Figures 3.11 & 3.14 shows the performance of the nonparametric technique in the presence of output additive white and bandlimited noise respectively. For the linear path coefficients there was no significant bias between the white and bandlimited noise study but there was more variability in the presence of white noise. Coefficients of the nonlinear path had a greater bias and higher variance in the presence of bandlimited noise than those with white noise.

3.3.5 Input Noise Sensitivity

In the preceding sections we only examined the effects of output noise on our parameter estimation technique. It is well known that standard least-squares was not developed to tolerate input noise [45, 105, 113, 133]. When studying ankle dynamics, under experimental conditions, it is known that the input may not be measured with negligible error. In our laboratory, experimental (PRBS) inputs are typically in the range of 0.005 – 0.2 rad. The convention used in the laboratory is 0.1 rad = 1 Volt. The input-output is recorded with a 16-bit A/D (IOTech ADC488) and has a 20 Volt dynamic range. This yields an input SNR (assuming no other input noise source) approximately in the range of 82 dB – 50 dB (see [121] p. 756). Although this corresponds to small amplitude noise, we examined the effects of input noise with our standard least-squares parameter estimation algorithm.

For this study, simulation and estimation protocols remained the same as described in §3.3.2 except that a PRBS input was used to excite the system dynamics. We used a PRBS input for this study because it is the type of input used under experimental conditions in our laboratory.

Figure 3.12 shows a typical input-output trial used for this analysis. The data represents a PRBS sequence of 0.0375 rad (peak-to-peak) and 125 ms switching rate. The characteristics of this trial are consistent with those used for analysis in this section. A unique Gaussian, white, zero-mean noise sequence was added to the input,

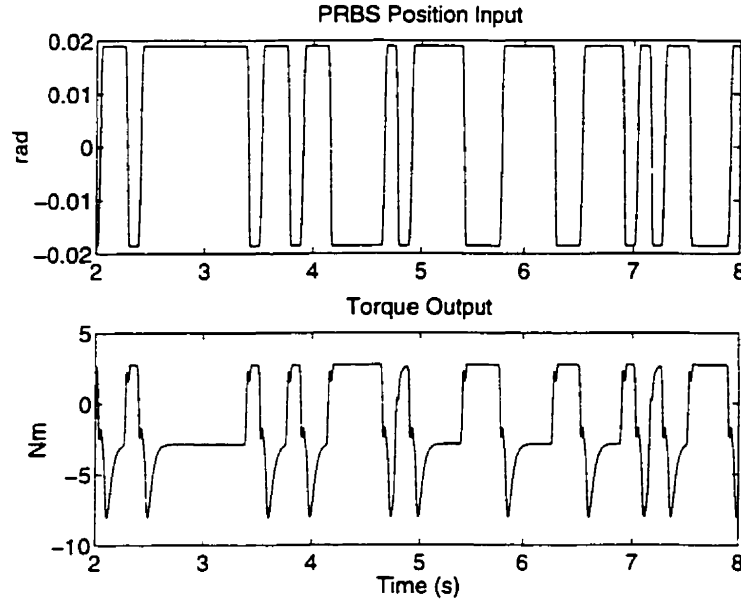


Figure 3.15: Typical position input and torque output recorded from simulation used to assess input noise sensitivity.

noise amplitude was increased in increments of 5 dB, from 70 dB to 50 dB SNR, the output was noise-free, and $N = 7,000$ data points were used for estimation.

3.3.5.1 Monte-Carlo Analysis of NARMAX Parameters: Additive Input Noise – Noise-Free Output

The results of this study are summarized in Figure 3.16 and correspond to the reduced NARMAX model (Equation 3.3 & Table 3.1). These plots show that even when an insignificant amount of input noise was added (with noise-free output) NARMAX parameters deviated significantly from their true values. Note that parameters $b_0, b_4 - b_7$ and $m_1 - m_2$, associated with input only, flipped signs for SNR levels ≤ 55 dB SNR which correspond to continuous-time parameters with incorrect sign, i.e., their STD encompass zero and, therefore, cannot be distinguished from zero. This bias may be the combination of two factors: (1) violation of least-squares theory and (2) model representation, i.e., high-pass nature of the linear path.

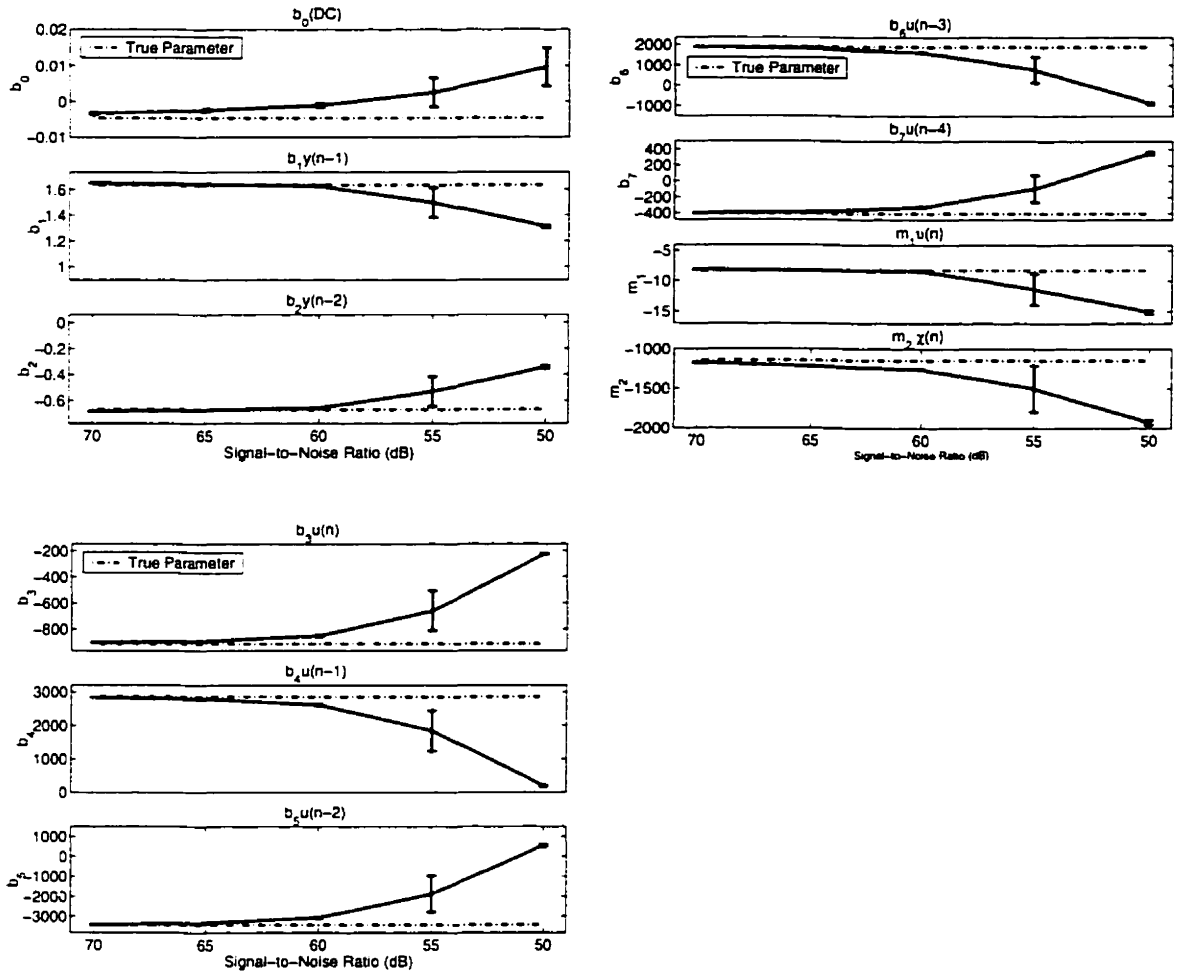


Figure 3.16: Compressed NARMAX model: PRBS input, Gaussian, white, zero-mean input noise and $N=7,000$. Ordinate: STD about mean. Abscissa: Input SNR= 70, 65, 60, 55 and 50 dB. Dashed line: True parameter value. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

3.3.6 Input and Output Noise Sensitivity

Although the previous study showed that even with an insignificant amount of noise on the input yields highly parameters, we investigated sensitivity of this identification approach to both input and output noise. This study was conducted to assess the likely performance of our algorithm under experimental conditions.

Simulation and estimation protocols remained the same as described in §3.3.2 & §3.3.5. The input and output noise sequences were unique Gaussian, white, zero-mean processes which were uncorrelated with the input and each other. The input noise sequence was fixed to have a SNR of 60 dB while the output additive noise amplitude was increased in increments of 5 dB, from 30 dB to 0 dB.

3.3.6.1 Monte-Carlo Analysis of NARMAX Parameters: Additive Input and Output Noise

Results of this study are summarized in Figure 3.17. These parameters correspond to NARMAX parameters given in Table 3.4. The plots in this figure show that, for input SNR of 60 dB, some NARMAX parameters deviated significantly from their true values for output SNR ≤ 10 dB (see parameters b_1, b_2, b_4, b_7 and b_{20}). These parameters, associated with input and output, had incorrect sign, which corresponds to continuous-time parameters with incorrect sign and large variance.

3.3.6.2 Monte-Carlo Analysis of Continuous-time Parameters: Additive Input and Output Noise

Next, we show results of estimating continuous-time parameters from NARMAX estimates given in §3.3.6.1. Figure 3.18 includes parameter estimates of both the linear and nonlinear paths. These parameters were computed using our NARMAX identification approach. In these plots, we did not show the mean and STD for 0 dB SNR to give better resolution at higher SNRs. The STD of continuous-time parameters included zero for 0 dB SNR. The results show that, even with an insignificant amount of noise added to the input, variance of continuous-time parameters increased by more

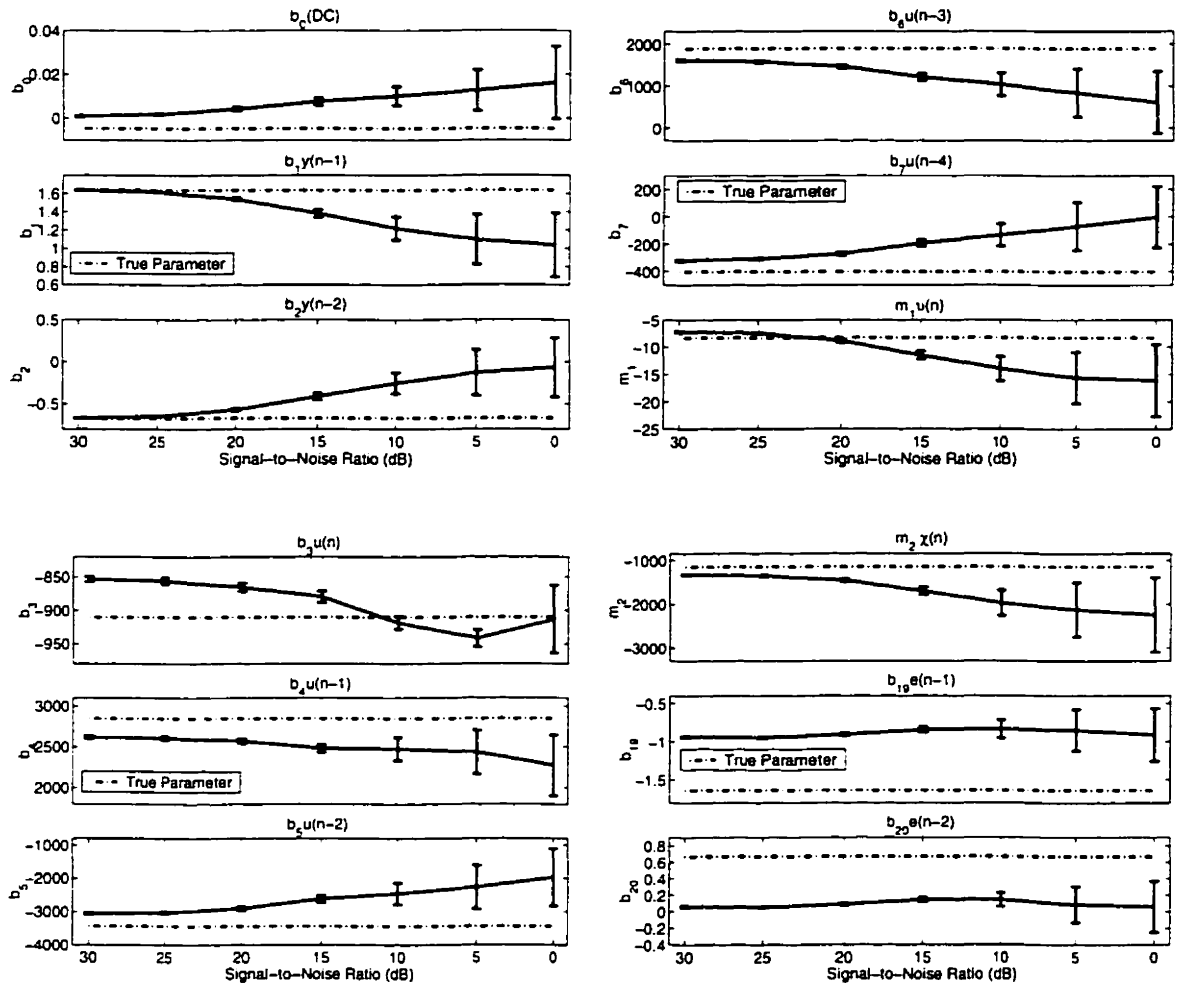


Figure 3.17: Compressed NARMAX model: PRBS input, Gaussian, white, zero-mean input and output noise and $N=7,000$. Input SNR=60 dB. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10, 5 and 0 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

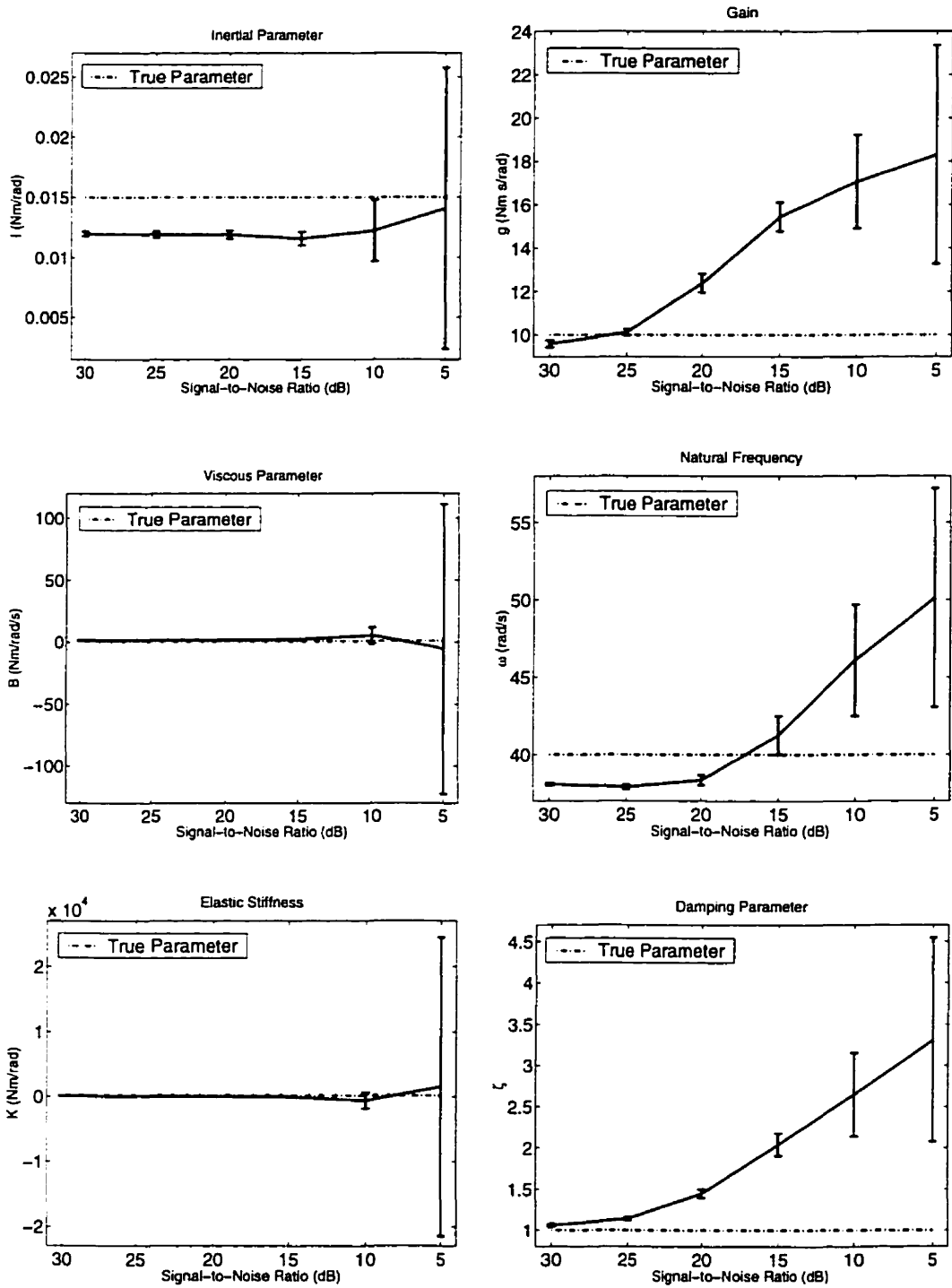


Figure 3.18: Continuous-time parameters of parallel pathway ankle model. PRBS input, Gaussian, white, zero-mean input and output noise, $N=7,000$ and NARMAX identification. Input SNR=60 dB. Ordinate: STD about mean. Abscissa: Output SNR= 30, 25, 20, 15, 10 and 5 dB. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

than 10 times for all SNRs compared to the noise-free input, output additive noise case (see Figure 3.10). Here, STDs of all linear path parameters, except I , included zero for output SNR ≤ 20 dB. The nonlinear path parameters were significantly biased for SNR ≤ 25 dB. Therefore, parameter estimation using our NARMAX approach may give biased results in situations where the input is not recorded with very high precision or a noise-free record of the input is not available.

3.3.7 Closed-Loop Simulation

We investigated the effects of removing continuous-time derivatives from the simulation (see Figure 3.2) as a possible source of bias for the nonparametric identification technique. The derivative operator in Simulink is implemented using Newton's backwards formula [104]. This may give our NARMAX approach an unfair advantage since the derivative operator in the NARMAX formulation of ankle dynamics is also approximated using Newton's backwards formula (see §3.2.2). We removed the derivative operators by simulating the ankle model in closed-loop, i.e., from torque to position.

It is hypothesized that ankle dynamics are physically generated in closed-loop [78]. However, for analysis and identification purposes, system parameters are estimated using an inverse parallel pathway model (Figure 3.1). The nonparametric technique has been designed assuming that the input-output data is generated by such a closed-loop structure. A closed-loop formulation of ankle dynamics is shown in Figure 3.19 [78].

Specifically, we removed the effect of the derivative in the feedback path by simulating the linear path as a bank of integrators in closed-loop (see Figure 3.20). The output of the first integrator is velocity, therefore, it was used as input to the nonlinear path; thereby removing the effects of the derivative.

We simulated this model using two unique torque inputs; uniformly distributed, white, zero-mean, random input, low-pass filtered with an eighth-order 30 Hz Bessel filter. The parameters used in the simulation were the same values given in Table 3.2. The system was identified from position to torque assuming a parallel pathway model,

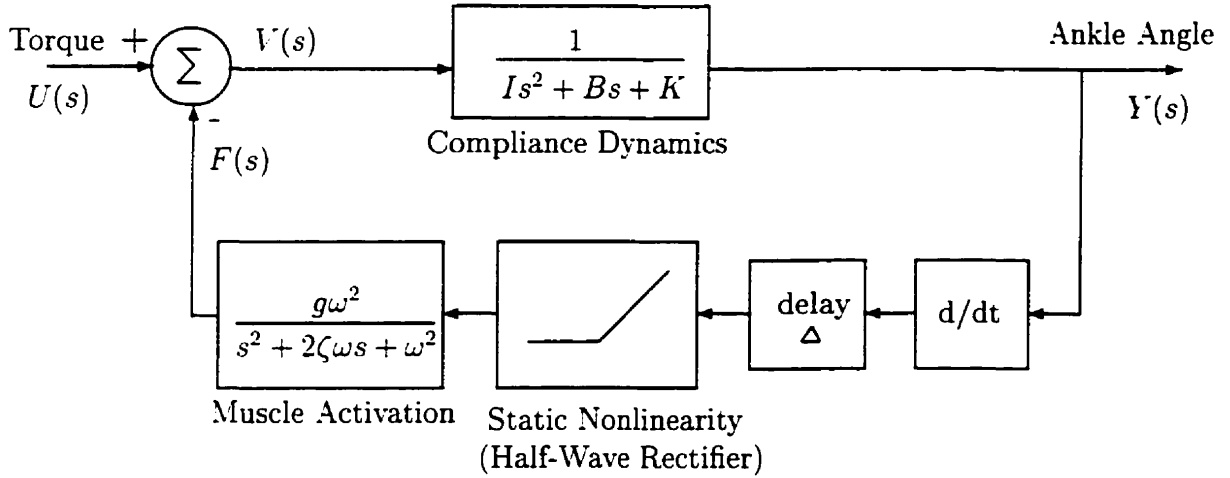


Figure 3.19: Closed-loop system of ankle dynamics. $U(s)$: System input. $Y(s)$: System output. $F(s)$: Output of feedback loop. $V(s)$: Error signal.

as described in §3.3.2. The continuous-time coefficients were estimated as described in §3.3.4. The system was identified under noise-free conditions.

Figure 3.21 shows a typical input-output trial used for this study. The top panel shows a position record (labeled “POS” in Figure 3.20), the middle panel shows a velocity signal (labeled “VEL” in Figure 3.20) and the bottom panel shows a torque record (the sum of “u” and “TQr” in Figure 3.20).

The results of this identification are presented in Table 3.6. Results show that the nonparametric method was able to estimate the continuous-time parameters while estimates computed by our NARMAX approach were highly biased. Note that the NARMAX estimate of B , viscosity, has incorrect sign. Therefore, applying NARMAX identification to experimental data will likely yield biased results. Although the nonparametric technique gave better results than NARMAX, the estimates are still biased even under noise free conditions and consistent with our results in §3.3.4. However, unlike the NARMAX estimates these nonparametric estimates are within reasonable physiological ranges.

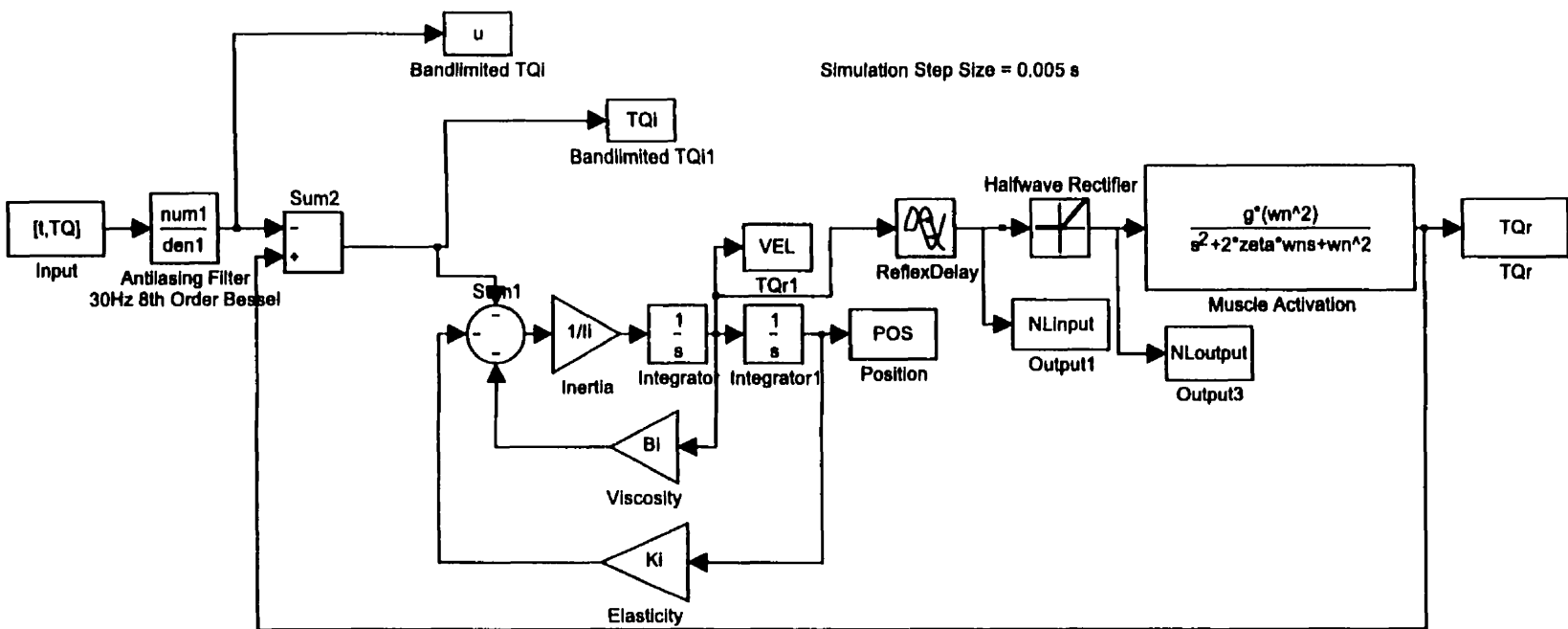


Figure 3.20: Simulink model of closed-loop ankle model in continuous-time.

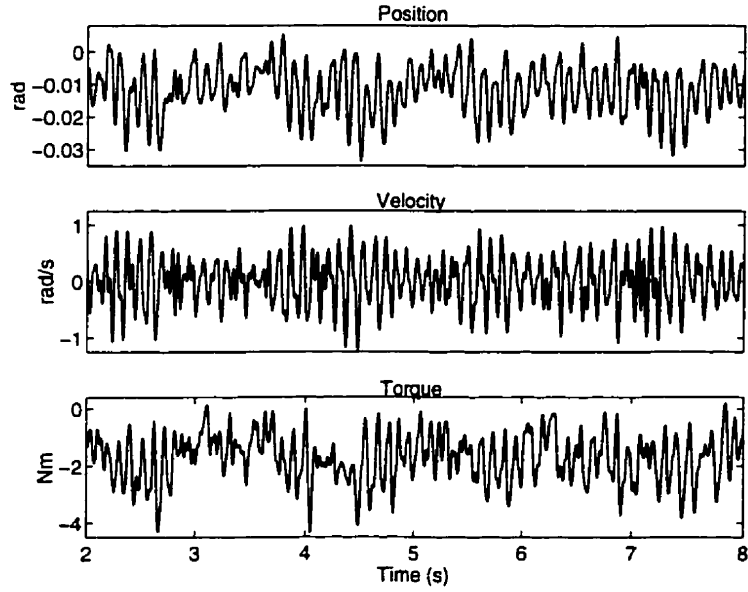


Figure 3.21: Typical position, velocity and torque record from closed-loop simulation of ankle dynamics.

Coefficients	Trial 1		Trial 2	
	Nonparametric	NARMAX	Nonparametric	NARMAX
$I = 1.50 \times 10^{-2}$	1.28×10^{-2}	2.53×10^{-2}	1.21×10^{-2}	2.59×10^{-2}
$B = 8.00 \times 10^{-1}$	8.51×10^{-1}	-4.29×10^0	8.60×10^{-1}	-4.30×10^0
$K = 1.50 \times 10^2$	1.23×10^2	5.75×10^2	1.17×10^2	5.75×10^2
$\omega = 4.00 \times 10^1$	3.84×10^1	6.44×10^1	3.87×10^1	6.44×10^1
$\zeta = 1.00 \times 10^0$	1.04×10^0	1.14×10^0	1.06×10^0	1.14×10^0
$g = 1.00 \times 10^1$	9.56×10^0	4.64×10^0	9.39×10^0	4.03×10^0

Table 3.6: Continuous-time coefficients of ankle model simulated in closed-loop but identified assuming a parallel pathway model. Coefficients: Continuous-time parameters values used in simulations. Trail 1: Estimated continuous-time parameters using the nonparametric and NARMAX approach, respectively. Trail 2: Estimated continuous-time parameters using the nonparametric and NARMAX approach, respectively.

3.4 Experimental Data

Lastly, we assessed our identification technique on experimental human ankle data collected in our laboratory. The data analyzed for this study is from a single subject with no history of neuromuscular disease.

3.4.1 Apparatus

The subject lay supine with his/her foot attached to the pedal of an actuator by a custom fitted fiberglass boot [107, 110]. The fiberglass boot was firm enough to restrict heel movement during experiments without excessively restricting range of ankle rotation. Sandbags and a kneestrap held the knee fully extended.

An electro-hydraulic actuator operated as a position-servo driving the ankle position to follow a command input with a bandwidth of 0 – 80 Hz. Ankle position was measured with a precision potentiometer (Beckman 6273-R5K) while torque was recorded using a torque transducer (Lebow 2110-5K) mounted in series with the subject's ankle [107].

An angle of 90° between the shank and foot was considered as the neutral position and defined as zero. Displacements in the dorsiflexing direction were considered as positive and those in the plantarflexing direction as negative. Torque was assigned a polarity consistent with the direction of movement [107].

3.4.2 Perturbations

PRBS inputs [79] were used to excite the dynamics of this system. Data records in which input sequences had a peak-to-peak amplitude of 0.01 – 0.05 rad and a switching rate of 45 – 260 ms were used for this study.

3.4.3 Procedures

The subject was instructed to maintain a constant level of activation and not to resist the perturbations. Torque generated by the subject was measured, low-pass filtered,

and fed back to an oscilloscope mounted above the subject's head. The subject was asked to match the torque feedback to a command signal displayed on the oscilloscope. Each PRBS sequence was started once the subject matched the desired torque level and recording was initiated after the subject re-established a stable contraction at the desired level [107]. The measured data was anti-alias filtered with an eighth-order 200 Hz Bessel filter (Frequency Devices, 64PF) and sampled at 1000 Hz by a 16-bit A/D converter (IOTech ADC488). Each input-output set was recorded for 30 seconds.

After recording, the experimental data was decimated by a factor of 10, resulting in a final sampling rate of 100 Hz. The system (ankle dynamics) was identified using our NARMAX approach, as outlined in §3.3.4, except that $N = 2,000$ points was used for estimation and $N_v = 1,000$ points was used for validation. Our least-squares algorithm fitted parameters for the model with a fixed delay and repeated the estimation with delays ranging from 50 to 100 ms. The parameter set and delay which yielded the highest cross-validation %NMSE was deemed the best-fit model.

3.4.4 Results

The results of identifying 8 trials of human ankle experiments are presented. Figure 3.22 shows a typical input-output trial used for this analysis. The data represents a PRBS sequence of 0.05 rad and 260 ms switching rate while the subject maintained a mean contraction of -5 Nm. The characteristics of this trial are consistent with those reported in previous work done in our laboratory [107]. Figure 3.23 displays the cross-validation (predicted) output, for this trial. The predicted output matched the measured output with over 98% NMSE.

Table 3.7 shows the “expected” (i.e., “ball-park” estimates) continuous-time parameter ranges for this model of ankle dynamics. These parameter ranges were determined in previous work done in our laboratory using nonparametric techniques [70, 75, 76, 77, 78, 79, 107, 162, 163, 164].

Figure 3.24 presents the results of NARMAX identification of continuous-time parameters for ankle dynamics. Parameter estimates computed using our NARMAX approach were compared with known physiological ranges (Table 3.7) to determine

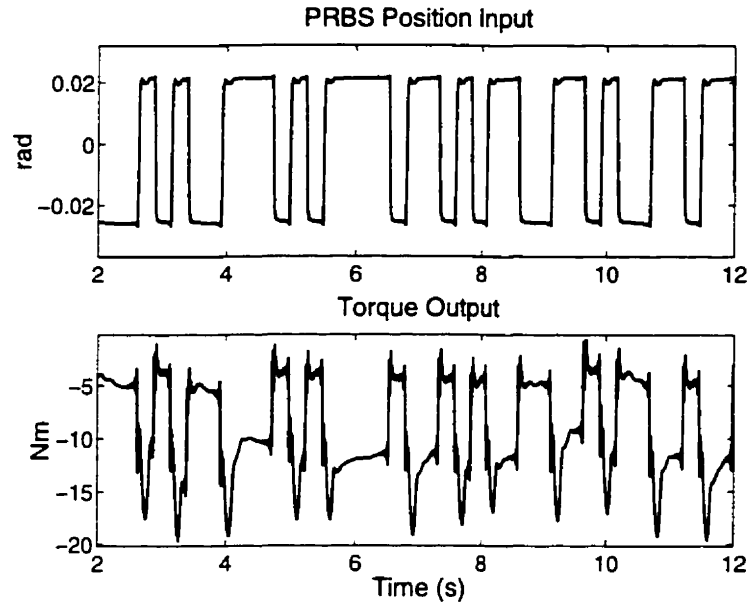


Figure 3.22: Typical recorded position input and torque output.

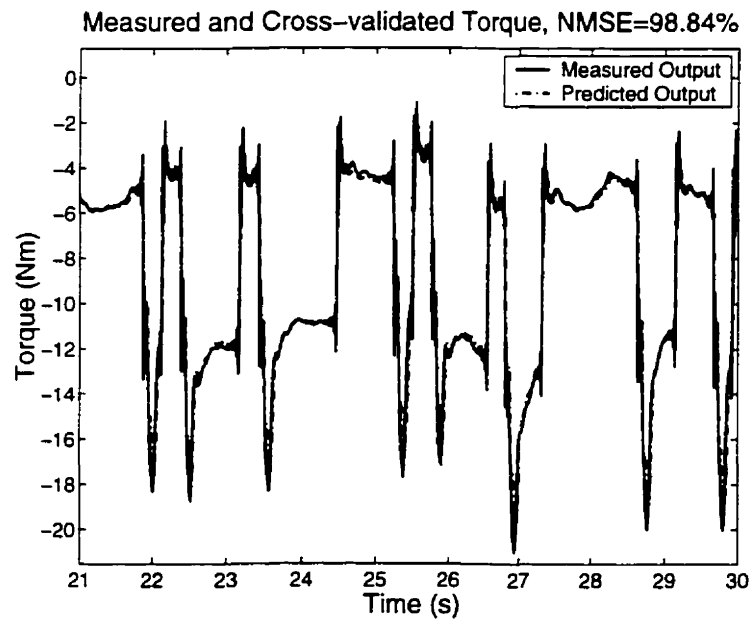


Figure 3.23: Cross-validation: Measured and predicted output for identified NARMAX ankle model for experimental data set with $N_v = 1,000$.

CT Parameter	Minimum	Maximum
I (Nm/rad)	0.0075	0.015
B (Nm/rad/s)	0.35	2.5
K (Nm/rad)	50	1000
ω (rad/s)	20	60
ζ	0.5	1.5
g (Nm s/rad)	1	22.5

Table 3.7: Continuous-time parameter ranges for parallel pathway model of ankle dynamics.

how well NARMAX estimates agree with these values. The dashed lines in each figure represent the maximum and minimum range for each parameter. The results in Figure 3.24 show that estimates obtained using our NARMAX approach failed to identify all but two continuous-time parameters (K , elastic stiffness and g , gain) within given physiological ranges.

Figure 3.25 shows the cross-validation %NMSE for each trial. The results show that the predicted output, for these parameter estimates, account for a large portion of the variance. The range of %NMSE is from a minimum of 94.43% to a maximum of 99.64%. From the 8 trials examined for this study, 62.50% of predicted outputs accounted for more than 99% NMSE of the measured output. Although cross-validation shows a high %NMSE in fit, the parameter values for I and B have incorrect sign. This implies that these subjects have negative inertia and viscosity, which is physically impossible. The incorrect sign may be a result of several factors, (1) too much input noise, (2) a deficiency in the model order and/or (3) incorrect structure.

3.5 Discussion

3.5.1 Discrete-Time Parameter Estimation

Simulation studies of discrete-time ankle parameters (§3.3.2 & §3.3.3) demonstrate that when a model has terms that are closely related, i.e., not described efficiently or the regressors are “almost” linearly dependent, it is difficult to estimate system

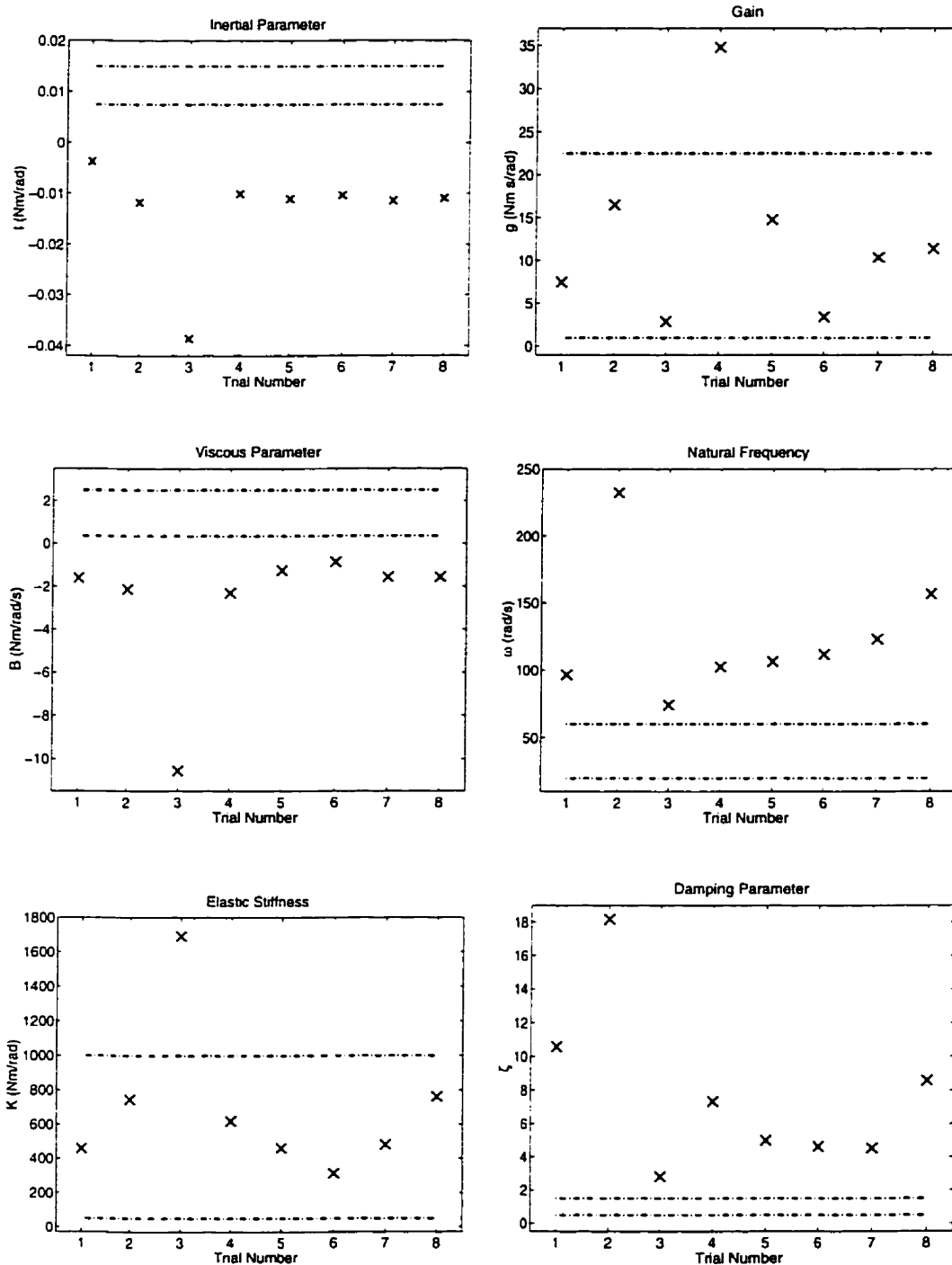


Figure 3.24: Continuous-time parameters of parallel pathway ankle model. NARMAX identified values for 8 trials of human ankle dynamics. Dashed lines: Maximum-minimum physiological range.

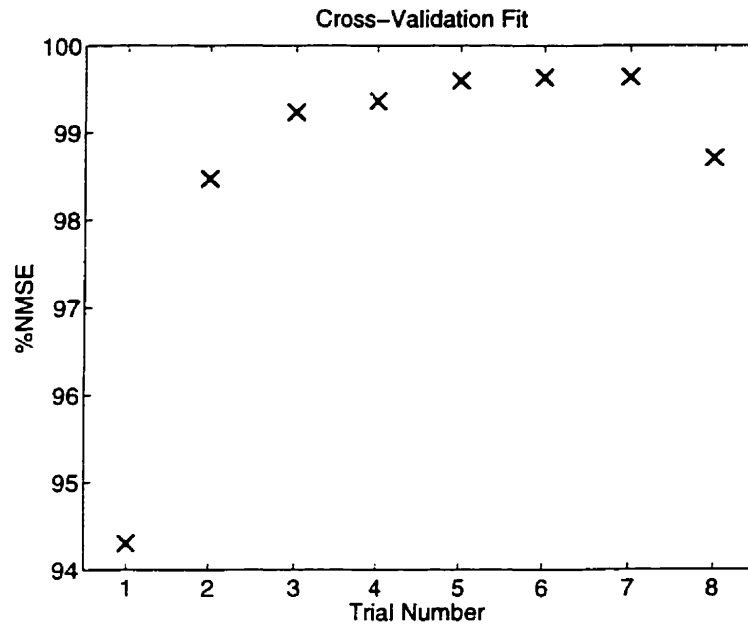


Figure 3.25: Cross-validation. %NMSE vs. experimental trial.

coefficients (see Equation 3.3 & 3.1). This suggests that discrete-time modeling of continuous-time systems could provide more stable parameter estimation when the number of regressors used to describe the system are reduced as much as possible.

In §3.3.3 we showed that, for a compressed NARMAX model representation, the mean of Monte-Carlo estimates for NARMAX parameters matched the theoretical values well for high SNR. However, estimates of some nonlinear parameters, e.g., $m_1 v(n)$, did not correspond well to theoretically computed values. This bias may be a result of using a low order approximation to the half-wave rectifier. A stiff nonlinearity of this form is of high order and, therefore, a second-order fit is only an approximation. Possibly for this reason, the theoretical values for the nonlinear parameters are not accurate. The mean value of the NARMAX identified parameters may therefore give a better estimate of the true system coefficients.

3.5.2 Continuous-Time Parameter Estimation

Estimation of continuous-time parameters of ankle dynamics demonstrates that, with noise-free input and low levels of output additive white and bandlimited noise distur-

bance, NARMAX identification methods provide a better estimate than traditional nonparametric techniques (compare Figure 3.10 with Figure 3.13 and Figure 3.11 with Figure 3.14). The clear advantage lies in the fact that, parametric (NARMAX) techniques do not require an initial guess of system parameters, which are seldom well known *a priori*. Using a nonparametric approach requires an initial guess and, therefore, estimated parameter(s) may be highly biased due to an initial seed far from the true value and/or being attracted to a local minimum [32, 78].

As the SNR was decreased, estimates computed using our parametric method had greater variability and moved further away from the true mean than estimates computed using the nonparametric technique; however, it may be possible to compensate for this by increasing the record length. This may not be the case using a nonparametric approach since the problem of local minima will still persist.

The validity of these “better” estimates, obtained using the NARMAX versus nonparametric approach, depends on the accuracy of our model simulation to represent continuous-time dynamics well, i.e., experimental data. For the parallel pathway model, our NARMAX technique may have an advantage over the nonparametric approach. This is because the derivative operator in the NARMAX formulation of ankle dynamics is implemented using Newton’s backwards formula, which is the same as in Simulink (see §3.2.2). As a result, our NARMAX model formulation matches the Simulink model “almost” perfectly. The nonparametric technique does not make the same model assumptions. Consequently, this may be a source of discrepancy between the estimates computed by these two techniques.

3.5.3 Input Noise Sensitivity

Any noise on the input violates assumptions and conditions for least-squares to yield unbiased parameter estimates. Nevertheless, we studied this effect to provide an understanding of the types of results (or biased results) that can be expected with experimental data. For this model structure, a study of input noise sensitivity (§3.3.5) showed that NARMAX identification was not robust to input noise. A study of the effects of input and output additive noise (§3.3.6) showed that with little noise added

to the input (60 dB SNR), the STD of linear path parameters included zero for output $\text{SNR} \leq 20$ dB. This result suggests that the least-squares algorithm may be very sensitive to input noise.

To determine whether input noise sensitivity is a result of the model structure or the least-squares algorithm we analyzed the behavior of this algorithm with a linear and nonlinear system which had low-pass characteristics. A low-pass system was used since it filters the effects of any amplified input noise due to derivatives. A preliminary study showed that least-squares was well behaved in the presence of large amplitude input and output noise, e.g., 10 dB input SNR and 5 dB output SNR. The parameter estimates were biased; however, none of the parameters had incorrect sign as in §3.3.5 & § 3.3.6.1. Clearly, more investigation is needed. Nevertheless, this showed that our algorithm may not as sensitive to input noise as simulation results indicate. This also suggests that for this ankle model sensitivity to input noise is not only due to a violation of least-squares theory but may also be a result of the model structure. This model is described by derivatives (i.e., unrealizable system) in both the linear and nonlinear paths which amplify input noise. A discrete-time representation of these derivatives is present in the regressor matrix when least-squares is implemented and, therefore, it may also contribute to parameter bias.

3.5.4 Closed-Loop Simulation

A study of simulating ankle dynamics in closed-loop, removing the effects of derivatives, but identifying the model as parallel pathway (§3.3.7) showed that our NARMAX approach gave severely biased estimates. Although estimates computed using the nonparametric technique only had a “slight” bias, giving reasonable estimates (in the same “ball-park”), the bias is consistent with our findings in §3.3.4.

The results of this NARMAX identification indicate that our approach is sensitive to performing inverse identification, i.e., from position to torque when the data is actually generated from torque to position. This suggests that the closed-loop ankle model may not be invertible, in a parametric form, and has a different structure than the one posed for identification, i.e., parallel pathway.

When ankle dynamics are simulated via a parallel pathway model in Simulink, a first-order derivative approximation is implemented which may not be good enough to accurately represent the “true” (experimental) analog data. This may give biased parameter estimates using the nonparametric technique. As noted earlier, this is not a problem for our parametric approach since we derived the NARMAX ankle model assuming the same first-order derivative approximation as in Simulink. Further investigation is needed to determine if our parallel pathway simulation represents the continuous-time process well.

It may be possible to generate more accurate simulations of experimental data if a higher-order derivative approximation is implemented, but one that is still linear-in-the-parameters so it falls within the NARMAX class. This approach may give better estimates via both techniques. An alternate approach may be to theoretically derive a closed-loop NARMAX model of ankle dynamics and identify the system in its natural state, in nonlinear feedback.

3.5.5 Experimental Data

Analysis of experimental data showed that our NARMAX approach gave results that are not consistent with known physiological ranges. There are several factors that may be attributed to this.

Sensitivity to input noise of the least-squares algorithm, used to identify parameters of this model structure, may exhibit significant bias, i.e., parameters with incorrect sign, as demonstrated by our simulation studies in §3.3.5 – 3.3.6 (see Figures 3.16 – 3.18). A similar bias (incorrect sign) was observed when ankle dynamics were simulated in feedback but identified as a parallel pathway model (see Table 3.6).

Physiological ranges used to compare NARMAX estimates of continuous-time parameters are based on results obtained from nonparametric techniques. Using this nonparametric method as a “gold” standard may not be a good approach since there is no certainty that these parameter ranges are correct. Nevertheless, it is unlikely that these parameter ranges are “way” off since they are consistent with a variety of physiological measures (see e.g., [78, 74, 80, 107, 162, 163, 164]).

High %NMSE cross-validation fits obtained for parameter estimates using our NARMAX method (see Figure 3.25) are misleading at first glance. Parameter values computed for I and B have incorrect sign which implies that these subjects have negative inertia and viscosity. Clearly, this is not physically reasonable. Therefore, using %NMSE alone as an indicator of model goodness may lead to incorrect interpretations.

Lastly, the results may reflect bias in NARMAX estimates of continuous-time parameters due to using an incorrect model structure. A structural deficiency may not be obvious using nonparametric techniques since the resulting system description is not represented concisely and may be redundant (see §2.4.1).

It is impossible to make more explicit conclusions regarding interpretation of these results without doing a full study of the data, which would require model order selection and structure detection. In the present study, we only have two indicators of model adequacy: (1) %NMSE and (2) whether estimated parameters fall within physiological ranges. Both can be misleading and may be poor indicators of model adequacy. They are, however, the only ones available.

3.5.6 Discrete to Continuous-Time Parameter Mapping

Another possibility for the source of error in the continuous-time parameters computed using our NARMAX approach may be related to the nonlinear relationships between discrete-time NARMAX parameters and continuous-time parameters of the physical system (see Table 3.5). A small deviation from the true parameter value in discrete-time, due to noise or numerical error, may appear as a significant error in the estimated continuous-time coefficients. As a result it may be advantageous to study these model parameters only in discrete-time.

3.5.7 Continuous to Discrete-Time Transformations

In §3.2.2 we used the bilinear transform and Newton's backwards formula to convert the continuous-time linear dynamics to discrete-time. We used these transforms

since both require only a simple substitution to convert a continuous-time system to discrete-time. Two other techniques that give better approximation for LTI systems are linear extrapolation and linear interpolation methods [66, 94, 139]. However, these techniques are seldom used due to added complexity for little gain.

Linear extrapolation gives an improper transfer function (i.e., more zeros than poles) and linear interpolation gives a transfer function that is strictly proper (i.e., equal zeros and poles) [66, 94, 139]. These methods produce a discrete-time transfer function that gives a better output response than the bilinear transform or Newton's backwards formula. However, the pitfall of these methods is that it is difficult to derive the coefficients of the discrete-time linear system. This is the main reason that almost all engineering text books and literature only discussed the bilinear transform and Newton's backwards formula.

Using linear interpolation or linear extrapolation to derive a NARMAX representation of ankle dynamics it may be possible to derive a better approximation to the derivative than the one used in §3.2.2, i.e., Newton's backwards formula. This may give a better approximation to the derivative and provide simulation data that matches the continuous-time process better. However, one drawback is that it may give a discrete-time approximation to the derivative that is higher than first order thereby increasing the complexity of the NARMAX representation [66, 94, 139]. Another drawback is that the continuous to discrete mapping of the continuous-time parameters will be more complicated since it involves exponential functions. This may result in more sensitivity to continuous-time parameter estimation since small deviations from the "theoretical" or "true" in discrete-time will result in large errors in continuous-time.

3.5.8 Simulation Techniques

Smith [144] states, as that better discrete-time models can be achieved if the product of a zero-order hold and system dynamics are modeled as a discrete-time representation of the model instead of only representing the continuous-time system in discrete-time. In addition, as pointed out by Smith using higher-order integrators such as

Runge-Kutta to simulate continuous-time systems introduces additional poles into the dynamic process [144]. He suggests that integrators such as the zero, first, or second-order be used to achieve more accurate simulations since they do not increase the dynamic order of the system being simulated [144]. However, these integrators require tuning two parameters to give accurate results.

We may be able to achieve better simulations of our nonlinear ankle model if the integration methods suggested by Smith [144] are implemented and if the system is modeled as the product of a zero-order hold and system dynamics in discrete-time.

3.5.9 Total Least-Squares

In the standard linear model ($Y = X\theta + e$) it is often assumed that the exact structure of the regressor matrix is known, e is a vector of random errors which are uncorrelated and have zero means and the same variance, i.e., $E(e) = 0$, $D(e) = \sigma^2 I$. As demonstrated by analysis of experimental ankle data, these assumptions are frequently unrealistic since sampling or modeling errors often affect the regressor matrix X . Therefore, it is necessary to consider methods that also allow for random input errors.

One technique that allows for this is total least-squares [26, 69]. Application of total least-squares may give insight to determine if parameter estimates for this biological data resemble the currently believed ranges and if the ranges are accurate.

Few applications of total least-squares have been published in the literature [43]. Van Huffel and Vandewalle claim that in typical applications (linear systems), gains of 10–15% in accuracy can be obtained by using total least-squares instead of standard least-squares methods [69]. To date there are no known applications of this technique to nonlinear systems. Therefore, a careful development of existence theory for general NARMAX models, which allow for input additive noise, needs to be studied first.

3.5.10 Relevance of NARMAX Parameters to Physiology

In general, it is unclear if NARMAX parameters will have a better physiological relevance for model interpretation since, currently, there are no generally accepted methods to obtain a nonlinear differential equation (continuous-time model) from a nonlinear difference equation (NARMAX model). However, as we have demonstrated here, if the exact system structure is known it may be possible to compute the continuous-time coefficients from the identified discrete-time parameters, thus allowing greater insight into the underlying physiological process. In addition, statistical studies of NARMAX coefficients could lead to direct clinical relevance for diagnosis.

When studying biological systems, as suggested by our results from experimental data, it may not be practical to assume that the exact model order and structure are well known *a priori*. In physiological systems analysis one of the main objectives is not only to estimate system parameters but to gain insight into the structure of the underlying system. Therefore, to address this issue of structure computation, in Chapter 4, we will present a practical method for determining structure for NARMAX models.

3.5.11 Summary of Findings

Simulation studies and analysis of experimental data showed the following.

1. Discrete-time modeling of continuous-time systems could enable more stable parameter estimation when the number of regressors used to describe the system are minimized as much as possible.
2. Estimation of continuous-time parameters of ankle dynamics demonstrated that, with noise-free input and low levels of output additive white and bandlimited noise disturbance, NARMAX identification methods may provide a better estimate than traditional nonparametric techniques (assuming that our simulation represents the analog data well).

3. A study of the effects of input and output additive noise showed, with little noise added to the input, the STD of linear path parameters included zero.
4. A study simulating ankle dynamics in closed-loop, removing the effects of derivatives, but identifying the model as parallel pathway showed that our NARMAX approach gave severely biased estimates.
5. Analysis of experimental data showed that our NARMAX approach gave results that were inconsistent with known physiological ranges.

Clearly, much work remains to be done to resolve the possible source(s) of error for NARMAX identification of this ankle model. The accuracy of our results for simulated data depends on validating the accuracy of our model simulation to represent the analog data well. If our simulation does not accurately represent the experimental data, it implies that the model used for identification may not be appropriate, thereby partly explaining our biased results for experimental data.

3.6 Conclusions

Theoretical results demonstrate that the nonlinear difference equation description for the parallel pathway model is a NARMAX model. Simulation results show that the NARMAX model matches the continuous-time response well [92].

Our analysis of continuous-time parameter estimates and their variance for this ankle model, using a NARMAX approach, demonstrates that this parametric method provides a better estimate of system parameters than nonparametric techniques when the underlying assumptions for standard least-squares are not violated.

We have demonstrated the importance of considering input noise sensitivity when implementing standard least-squares methods for analyzing experimental data. Therefore, unless the input is recorded with high precision or a noise-free record is available, it may be advantageous to consider alternative estimation methods such as total least-squares.

The overall significance of these results has been to demonstrate the importance and relevance of the NARMAX structure for physiological modeling and analysis. We have demonstrated this by modeling the dynamic behavior of a parallel pathway model of ankle dynamics as a NARMAX model. In addition, we have illustrated that appropriate methods exist to identify the dynamics of such systems.

Chapter 4

Bootstrap Structure Detection

4.1 Overview

Many systems may be described by NARMAX models using only a few terms. However, depending on the order of the system the number of candidate terms can become very large. Selection of a subset of these candidate terms is necessary for an efficient system description. This is an unresolved issue in system identification for over-parameterized models. Therefore, in this chapter, we develop a bootstrap structure detection (BSD) algorithm as a means of determining the structure of highly over-parameterized models.

The performance of this BSD technique was evaluated by using it to estimate the structure of (1) a simple NARMAX model, (2) a highly over-parameterized NARMAX model and (3) a NARMAX model describing slow-phase dynamics of the vestibulo-ocular reflex. The results demonstrate that the BSD algorithm is a robust method for detecting the structure of linear regression models. This method provides accurate estimates of parameter statistics without relying on assumptions made by traditional procedures and yields a parsimonious description of the system.

4.2 Introduction

Recently, bootstrap techniques have received considerable attention due to the availability of affordable and powerful computers [46]. The bootstrap is a numerical procedure for estimating parameter statistics that requires few assumptions. The conditions needed to apply bootstrap to regression analysis are quite mild; namely, that the errors be independent, identically distributed, and have zero-mean. This contrasts with regression analysis that requires an accurate estimate of the noise process which is difficult to obtain unless the model structure is correct. Consequently, we hypothesize that bootstrap might be a useful tool for structure detection of nonlinear models.

4.3 Structure Detection

NARMAX representations of many nonlinear systems require only a few terms. However, as the order of the system increases, the number of candidate terms becomes very large (Equation 2.22). The structure detection problem is that of selecting the subset of candidate terms that best describes the output. Several methods for NARMAX structure detection have been proposed including hypothesis testing of differences between means via the t-test, stepwise regression, and Korenberg's orthogonal structure detection routine. However, these all encounter problems with nonlinear systems.

The t-test and stepwise regression are widely used in regression analysis [45, 49, 133, 142]. The t-test relies on accurate estimates of parameter variances to determine significance while stepwise regression relies on the incremental change in residual sum of squares (RSS) resulting from adding or removing a parameter. Both methods need accurate estimates of system noise (computed from an estimate of model residuals) to determine structure. However, accurate or unbiased estimates of residuals are difficult to obtain even when the structure is correct. This is because the noise process is assumed to be generated by a stochastic process. In addition, since the number of candidate terms, p , becomes very large for even moderately complex nonlinear

models, the estimated noise may be highly biased, making structure detection difficult or impossible. We expect both the t-test and stepwise regression to have difficulty with highly over-parameterized models.

Korenberg [3, 4, 84, 85, 87] developed an orthogonal structure detection routine specifically for nonlinear systems. This method relies on orthogonalizing the regressor matrix and using the orthogonal relationships to compute the reduction in the total mean-squared error due to each column. However, it requires selecting a tolerance level to determine which terms to reject or accept. The selection of this tolerance level requires *a priori* knowledge about the true errors and system output, which is seldom available. Therefore, this tolerance level is set by trial and error and may not yield a parsimonious system description unless some *a priori* information is available about the system [38].

4.4 Mathematical Preliminaries

Bickel and Freedman [15, 16, 55] analyzed the linear regression model where the number of data points N and parameters p were both large. For the full p -dimensional distribution of the least-squares estimates, as $p^2/N \rightarrow 0$ the bootstrap distribution will converge to the true unknown distribution [16]. Since, initially, p cannot change, the accuracy of the bootstrap estimate is determined by the data length, N , available for estimation.

Consider the linear model

$$Z = \Psi\theta + e \tag{4.1}$$

with assumptions stated in §2.6.3.1 in force. Z is a $N \times 1$ vector of measured outputs. Ψ is a $N \times p$ ($p \ll N$) matrix of regressors with full rank (i.e., nonsingular) and can be non-deterministic. θ is a $p \times 1$ vector of parameters. e is a $N \times 1$ vector of an i.i.d. (independent and identically distributed) noise sequence with zero-mean, homoskedastic (have the same variance), common distribution F and variance $\sigma^2 > 0$. Both F and σ^2 are unknown.

The number of parameters, p , is given via Equation 2.22 as

$$\begin{aligned} p &= \sum_{i=1}^l p_i \\ p_i &= \frac{p_{i-1}(n_y + n_u + n_e + i - 1)}{i}, \quad p_0 = 1 \end{aligned} \quad (4.2)$$

where n_u is the maximum lag on the input, n_y the maximum lag on the output, n_e the maximum lag on the error and l is the maximum nonlinearity order. We define the maximum number of terms, p , as the number of “candidate” terms to be initially considered for identification.

The least-squares estimate of θ is given by Equation 2.78 as

$$\hat{\theta} = (\Psi^T \Psi)^{-1} \Psi^T Z, \quad (4.3)$$

the fitted values are

$$\hat{Z} = \Psi \hat{\theta} = PZ, \quad \text{where } P = \Psi(\Psi^T \Psi)^{-1} \Psi^T \quad (4.4)$$

and residuals or prediction errors are given by

$$\hat{e} = Z - \hat{Z} = \Gamma e, \quad \text{where } \Gamma = I_{N \times N} - P. \quad (4.5)$$

In Equations 4.4 and 4.5 P and Γ are projection matrices. Ψ is defined as a partitioned regressor matrix

$$\Psi = [\Psi_{zu} \Psi_{zu\hat{e}} \Psi_{\hat{e}}] \quad (4.6)$$

where Ψ_{zu} is a function of z and u only, $\Psi_{zu\hat{e}}$ represents all the cross products involving $\hat{e}$, and $\Psi_{\hat{e}}$ is a polynomial function of the residuals only.

Let $Q^2 = \Psi^T \Psi$ be the cross-product matrix and let $\sigma^2 Q^{-2}$ be the variance-covariance matrix (the so-called Fisher information matrix) [16].

Remark 1 $\Psi^T \Psi$ is positive definite, therefore, it has a unique positive definite square root, Q . Positive definite is taken in the strict sense. $\sigma^2 Q^{-2}$ is interpreted as $\sigma^2 Q^{-2} =$

$$Q^{-2}\sigma^2$$

Let $\Upsilon_{Np}(F)$ be the exact distribution of $Q(\hat{\theta} - \theta)$, with N data points, p parameters and law F governing the disturbance terms.

The “Mallows metrics” are defined in Bickel and Freedman [15]. In summary, let $\|\cdot\|$ be the Euclidean norm on $\mathbb{R}^p$ and let $\alpha \geq 1$. Then

$$d_\alpha(\mu, \nu) = \inf E\{\|U - V\|^\alpha\}^{1/\alpha}, \quad (4.7)$$

where U has law μ , V has law ν and the “inf” is over the joint distribution. Convergence in d_α is equivalent to weak convergence plus convergence of moments of order α or less [15].

The bootstrap estimates the distribution $\Upsilon_{Np}(F)$ by $\Upsilon_{Np}(G)$, where G is an estimate of F . Bounds will be given on $d_2[\Upsilon_{Np}(F), \Upsilon_{Np}(G)]$ in terms of $d_2(F, G)$, for any F and G [16].

Theorem 1 Bickel and Freedman [16]: *Let F and G be two possible laws for the noise process e_n in the model (Equation 4.1); it is assumed that both have mean 0 and finite variance. Then*

$$d_2[\Upsilon_{Np}(F), \Upsilon_{Np}(G)]^2 \leq p \cdot d_2(F, G)^2$$

Proof 1 Proof in Bickel and Freedman 1982 [16].

Remark 2 *The bound in Theorem 1 has an extra factor p on the right due to the fact that it compares p -dimensional distributions.*

G is the empirical distribution $\hat{F}_N$ of the centered residuals. Define $\hat{\mu}_N = \frac{1}{N} \sum_{n=1}^N \hat{\epsilon}_n$ which may be nonzero since constants (D.C. terms) need not be in the column space of Ψ . Let $\hat{F}_N$ be the empirical distribution of the centered residuals, assigning mass $1/N$ to each $\hat{\epsilon}_n - \hat{\mu}_N$. Let F_N be the empirical distribution of the noise process, e_n , $n = 1, 2, \dots, N$.

Review of the bootstrap operation: Given $z_1, \dots, z_N$, let $\hat{\epsilon}_1^*, \dots, \hat{\epsilon}_N^*$ be conditionally independent, with common distribution $\hat{F}_N$. Let

$$\hat{Z}^* = \Psi \hat{\theta} + \hat{\epsilon}^*. \quad (4.8)$$

Informally, $\hat{\epsilon}^*$ is obtained by resampling the residuals, $\hat{\epsilon}$, and $\hat{Z}^*$ is generated from the data, using the regression model with $\hat{\theta}$ as the vector of parameters and $\hat{F}_N$ as the distribution of the residuals.

Next, consider giving the “starred” data $(\Psi^*, \hat{Z}^*)$ to another experimenter and asking him or her to estimate $\hat{\theta}$. The least-squares estimate is simply

$$\hat{\theta}^* = (\Psi^{T*} \Psi^*)^{-1} \Psi^{*T} \hat{Z}^*. \quad (4.9)$$

The bootstrap principle is that the distribution of $(\hat{\theta}^* - \hat{\theta})$, which can be computed directly from the data, i.e., by Monte-Carlo, approximates the distribution of $(\hat{\theta} - \theta)$. Similarly, for the full p -dimensional distributions.

Let $\hat{\epsilon}^*$ be the bootstrap residuals

$$\begin{aligned} \hat{\epsilon}^* &= Z^* - \Psi^* \hat{\theta} = \Gamma \hat{\epsilon} \quad \text{where } \Gamma = I_{N \times N} - \Psi^* (\Psi^{T*} \Psi^*)^{-1} \Psi^{T*} \\ &= I_{N \times N} - P. \end{aligned} \quad (4.10)$$

Let

$$\begin{aligned} \hat{\sigma}^2 &= \frac{1}{N-p} \sum_{n=1}^N \hat{\epsilon}_n^2 \quad \text{and} \\ \hat{\sigma}^{2*} &= \frac{1}{N-p} \sum_{n=1}^N \hat{\epsilon}_n^{*2}. \end{aligned} \quad (4.11)$$

Theorem 2 Bickel and Freedman [16]: *Assume model 4.1 and conditions in §2.6.3.1. Suppose p is fixed and let $N \rightarrow \infty$. Then the d_2 -distance between the distribution of $Q(\hat{\theta} - \theta)$ and the conditional distribution of $Q(\hat{\theta}^* - \hat{\theta})$ given $z_1, \dots, z_N$ tends to zero in probability.*

Proof 2 Proof in Bickel and Freedman 1982 [16].

If $p \rightarrow \infty$ but $p^2/N \rightarrow 0$ and $E[d_2(F_N, F)^2] = o(1/p)$, the distance between the distribution of $Q(\hat{\theta} - \theta)$ and the corresponding bootstrap distribution tends to zero [16]. This has an interesting consequence for the Scheffé method of simultaneous inference. Consider bootstrapping $S = [(\hat{\theta} - \theta)^T Q^2(\hat{\theta} - \theta)]^{1/2}$ or $S/\hat{\sigma}$. Let $S^* = [(\hat{\theta}^* - \hat{\theta})^T Q^2(\hat{\theta}^* - \hat{\theta})]^{1/2}$.

Theorem 3 Bickel and Freedman [16]: *Assume model 4.1 and conditions in §2.6.3.1. If $N \rightarrow \infty$, $p^2/N \rightarrow 0$ and $E[d_2(F_N, F)^2] = o(1/p)$, the d_2 -distance between the distribution of $S - p^{1/2}\hat{\sigma}$ and the conditional distribution of $S^* - p^{1/2}\hat{\sigma}^*$ given $z_1, \dots, z_N$ tends to zero in probability; similarly for the distributions of $S - p^{1/2}\hat{\sigma}$ and $S^* - p^{1/2}\hat{\sigma}^*$.*

Proof 3 Proof in Bickel and Freedman 1982 [16].

Remark 3 *In structure computation the objective is often (directly or indirectly) simultaneous inference testing. For structure detection, to obtain consistent results the relevant criterion for bootstrap convergence is*

$$\pi = p^2/N \rightarrow 0. \quad (4.12)$$

Computing a good estimate of parameter statistics is the central issue for all structure detection algorithms and, hence, a poor estimate of these statistics may lead to models with incorrect structure. The analysis by Bickel and Freedman provides a guideline for data requirements needed for bootstrap estimates of an “unknown” noise distribution to converge to the true unknown distribution. This analysis was done in the context of large p (number of regressors) which is the situation experimenters encounter when computing structure of nonlinear systems; given the system order is known. This work is relevant for nonlinear system identification because, as Bickel and Freedman showed, the bootstrap provides a good estimate of the unknown distribution of the noise or error process. This implies that bootstrap may lead to good parameter statistics, if sufficient data is available. When the condition in Equation 4.12 is applied to the t-test it may give a good estimate of parameter statistics for a particular input-output realization but, in general, will not be consistent, $\neq 1$.

Although Equation 4.12 provides a guideline for computing a “better” estimate of parameter statistics, it only implies that they will on average equal the statistic they are supposed to estimate; it does not imply that any of these values must necessarily be close [56]. Therefore, the bootstrap is justified to obtain a consistent estimate of these “unknown” parameter statistics, thereby providing a consistent estimate of model structure.

4.5 Application to Structure Detection

Application of bootstrap to structure detection involves two steps: (1) computing a series of parameter replications, in which “bootstrap data” is generated to compute new “bootstrap parameter estimates”, and (2) forming percentile intervals for hypothesis testing, where the significance of the parameters is determined. Bootstrap data is formed by first estimating the residuals of the identified model; these residuals are then resampled with replacement, centered (mean is removed), and then added to the predicted output to generate bootstrap replications of the output [46, 48]. A number “B” of bootstrap data sets are generated to estimate B bootstrap parameter replications. Figure 4.1 outlines the procedure.

Significance of the parameters is determined by forming percentile intervals (Figure 4.2). The estimates from B parameter replications are ranked in increasing order and the $B \cdot \alpha$ th and $B \cdot (1 - \alpha)$ th values in the ordered list of the B replications are used as an upper and lower bound for the parameter deviation with an α th and $(1 - \alpha)$ th level of significance, respectively [48]. Significance of each parameter is determined by checking if 0 lies in its interval: if so, the parameter is rejected. This leads to the following algorithm to detect structure of linear regression models.

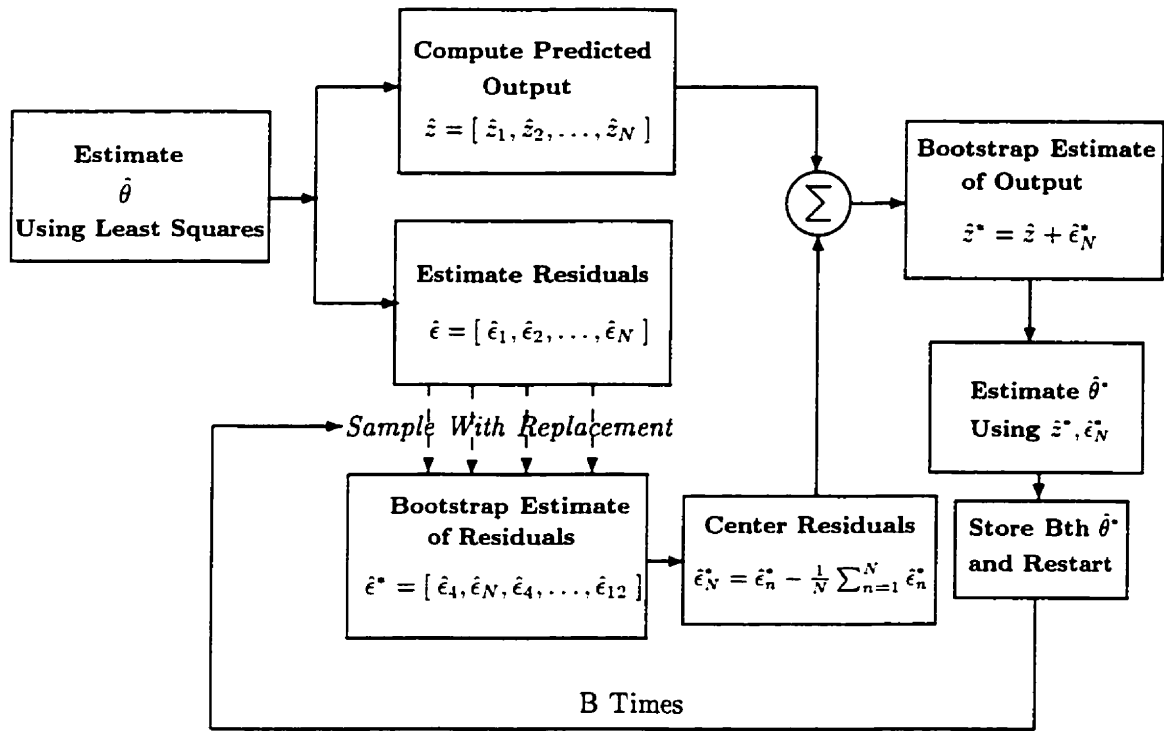


Figure 4.1: Procedure for forming bootstrap parameter replications.

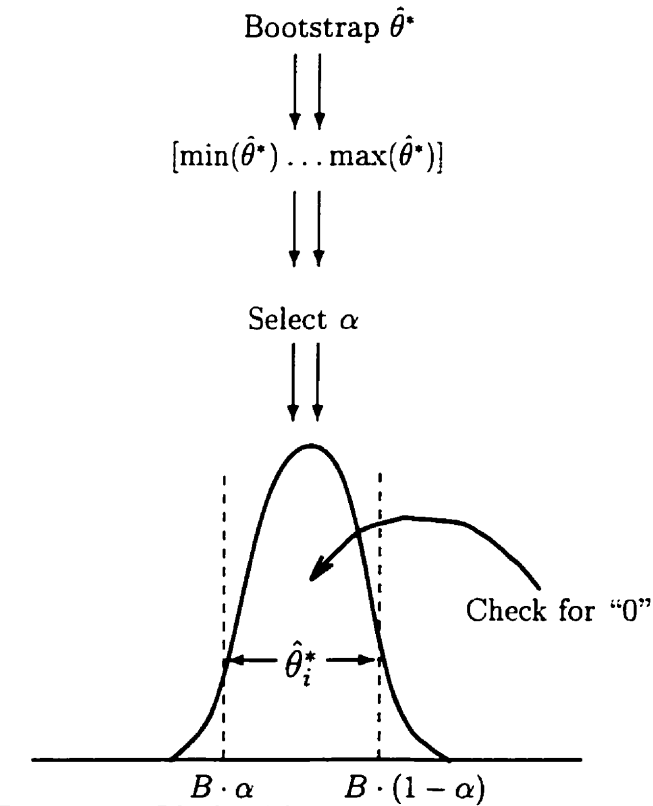


Figure 4.2: Idealized bootstrap-t interval as $B \rightarrow \infty$.

4.5.1 BSD Algorithm

1. Compute an initial estimate of the unknown parameter vector and estimate the residuals.
2. Generate B bootstrap data sets and compute the bootstrap parameter replications.
3. Form percentile intervals for each parameter by ranking estimates from the B parameter replications in increasing order.
4. Estimate the upper and lower bounds of each parameter's confidence interval for a desired level of significance.
5. Determine if zero lies in the interval of each parameter in the vector.
6. If zero lies in the interval for any parameter(s) remove it/them from the regression.
7. Compute a new estimate of the parameter vector and residuals.
8. Go to 2 until convergence.

4.6 Implementation of BSD Algorithm

During each bootstrap replication new bootstrap data is formed which requires the regressor matrix, Ψ , to be reformed and re-orthogonalized. To decrease computations, the bootstrap method was implemented to update only that part of the regressor matrix that depends on the new bootstrap data. Similarly, for orthogonalization, only these parts of the matrices need be updated. This is accomplished using well-known updating schemes for adaptive matrix orthogonalization [60].

This involved restructuring the regressor matrix so that instead of ordering the extended matrix as $\Psi = [\Psi_{zu} \Psi_{zuc} \Psi_{\epsilon}]$ it is reordered as

$$\Psi = [\Psi_u | \Psi_z \Psi_{zuc} \Psi_{\epsilon}] = [\Psi_1 | \Psi_2] \quad (4.13)$$

where Ψ_u is a function of u , Ψ_z a function of z only, $\Psi_{zu\hat{\epsilon}}$ represents all the cross products involving $\hat{\epsilon}$, and $\Psi_{\hat{\epsilon}}$ is a polynomial function of the prediction errors only. Thus all the terms that are updated during the bootstrap procedure are lumped together in the second partition of the matrix making reformation and re-orthogonalization of the regressor matrix efficient. Re-orthogonalization is accomplished using the modified Gram-Schmidt (MGS) algorithm. These changes optimize the computational aspects of the BSD algorithm but do not affect the order in which the steps of the algorithm, presented in §4.5.1, are implemented.

When implementing the MGS algorithm, if $Q_1^T Z$ is explicitly formed, it may introduce error in the estimated parameters. However, it has been shown that if MGS is applied to the augmented matrix (see e.g., [60])

$$\Psi_+ = \begin{bmatrix} \Psi & Z \end{bmatrix} = \begin{bmatrix} Q_1 & q_{N+1} \end{bmatrix} \begin{bmatrix} R_1 & \gamma \\ 0 & \rho \end{bmatrix} \quad (4.14)$$

then $\gamma = Q_1^T Z$. Computing $Q_1^T Z$ in this fashion and solving $R_1 \theta_{ELS} = \gamma$ produces a least-squares solution $\hat{\theta}_{ELS}$ that is “just as good” as the Householder QR method.

4.7 Simulations

The efficacy of the BSD algorithm was assessed using Monte-Carlo simulations of two nonlinear systems. For both systems, we assumed a sampling rate of 8,000 Hz ($T = 0.000125$ s) and bandlimited inputs were used (uniformly distributed, white, zero-mean, random sequence, low-pass filtered with an eighth-order 600 Hz Butterworth filter). Fifty Monte-Carlo simulations were generated in which each input-output realization was unique, and had a unique Gaussian white, zero-mean, noise sequence added to the output, with 0 dB SNR. For identification, a data length of $N = 3,000$ points was used. An initial estimate of the system parameters was computed, and $B = 300$ bootstrap replications were generated to assess the distribution of each parameter. Each parameter was then tested for significance at the 95% confidence level. The BSD routine’s performance was compared with two other structure detection methods: the

t-test and stepwise regression routine.

4.7.1 Bandlimited Input and White Noise

4.7.1.1 Simple NARMAX Model

We first studied the simple system:

$$y(n) = 0.4[u(n-1) + u^2(n-1)] + 0.8y(n-1) \quad (4.15)$$

which is of order $O = [1, 1, 1, 2]$ and has only 15 candidate terms (Equation 4.2). However, only 4 “true” parameters are needed to describe this system: two lagged inputs, one lagged output and one lagged error term. We studied this system since it has a small number of candidate terms.

Figure 4.3 shows the results for this model. The left panel shows the frequency of

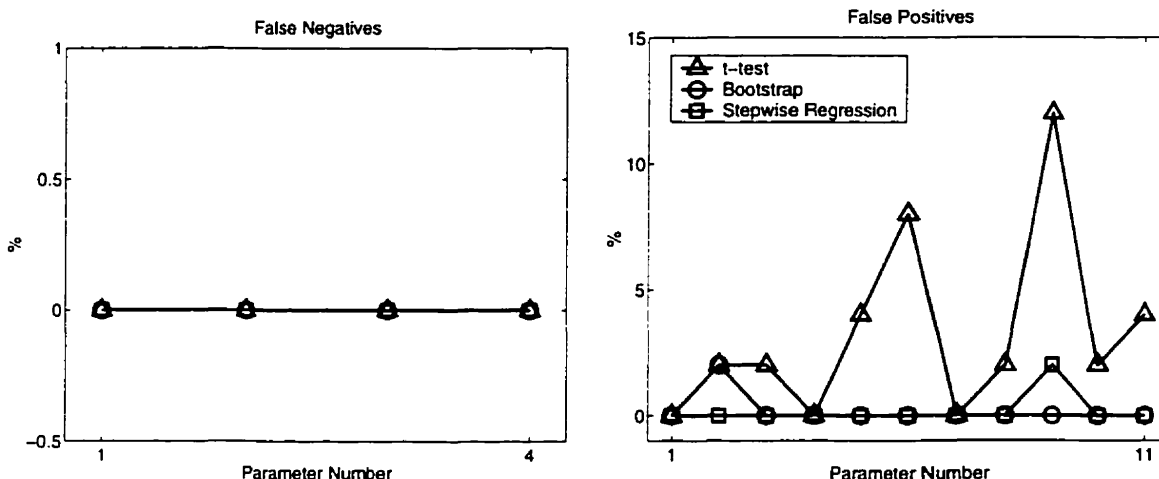


Figure 4.3: Predicted structure of a simple NARMAX model using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.

false negatives, the rate at which parameters actually in the model were rejected. The right panel shows the frequency of false positives, that is, the rate at which parameters not part of the model were selected. All three methods, the t-test, stepwise regression

and BSD methods, selected the true parameters with high accuracy. The false positive rate was similar for stepwise regression and BSD, but was higher for the t-test. Thus, for a simple model with few free parameters these three methods yielded comparable results for selecting “true parameters” while the t-test was less accurate for rejecting “false parameters”.

4.7.1.2 Highly Over-Parameterized System

Next, we examined the performance of the BSD technique for the following system:

$$y(n) = 0.4[u(n-1) + u^2(n-1) + u^3(n-1)] + 0.8y(n-1). \quad (4.16)$$

This system is described by 3 lagged inputs, 1 lagged output, 1 lagged error and third-order nonlinearity ($O = [1, 1, 1, 3]$). A system of this order has 35 candidate terms, but the “true” model has only 5 terms. With 30 spurious parameters this NARMAX model is highly over-parameterized. The identification paradigm was the same as described in §4.7 except $N = 5,000$ data points were used for parameter estimation and $B = 50$ bootstrap replications were generated to assess the distribution of each parameter. In this case, only 50 bootstrap replications were used to reduce computation time required for our BSD algorithm.

Figure 4.4 shows that the BSD method consistently selected the correct structure of this third-order nonlinear system while the t-test and stepwise regression both failed. The t-test had a false negative rate of 30% for the term associated with cubed input, lagged and had false positive frequency of over 20%. Stepwise regression had the same rate of rejecting a true parameter (cubed input term), 30%, and had a higher rate, 30%, of accepting spurious terms. The BSD method selected the true parameters consistently but did have a 10% false positive rate for two terms associated with input cross-terms. For this highly over-parameterized, third-order nonlinear model the BSD method clearly outperformed the t-test and stepwise regression.

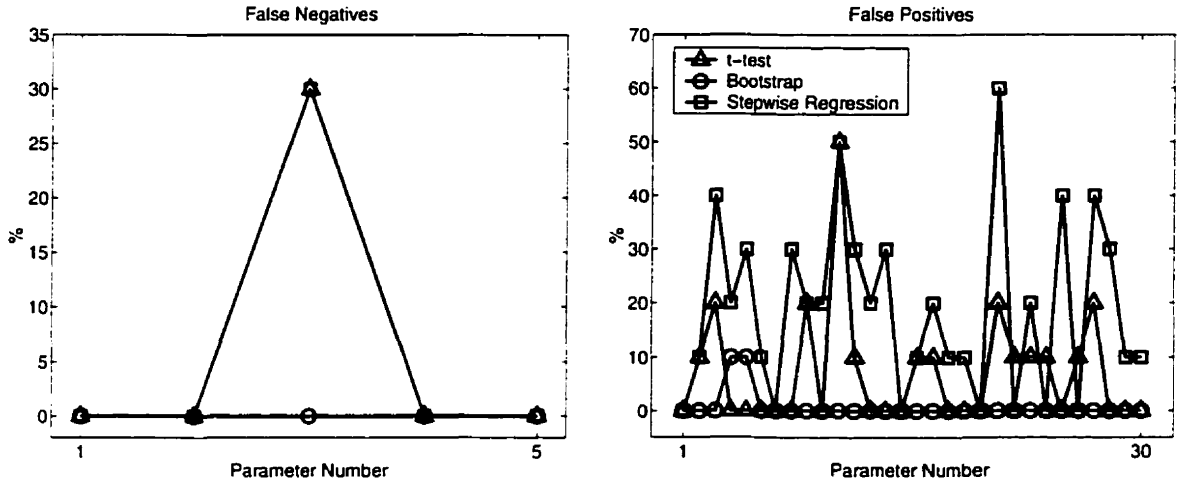


Figure 4.4: Error rate for highly over-parameterized system using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.

4.7.2 Assessment of Parameter Statistics

To assess the accuracy of our BSD technique for estimating parameter statistics the distribution of each parameter was computed for the simple NARMAX model (Model 4.15). The “theoretical” distributions were estimated using Monte-Carlo simulations consisting of 10,000 runs. For the BSD method, estimates were calculated from $B = 10,000$ bootstrap realizations; those for the regression methods (t-test and stepwise regression) were computed using a single realization via standard least-squares methods [133]. Parameter distributions were first calculated for the full model and second for the model including only true parameters, to estimate the “optimal” values for these terms. The simulation paradigm was the same as described in §4.7.

Model 4.15 is fully described as

$$\begin{aligned}
 y(n) = & \theta_0 + \theta_1 u(n) + \theta_2 u(n-1) + \theta_3 u^2(n) + \theta_4 u(n)u(n-1) \\
 & + \theta_5 u^2(n-1) + \theta_6 y(n-1) + \theta_7 u(n)y(n-1) + \theta_8 u(n-1)y(n-1) \\
 & + \theta_9 y^2(n-1) + \theta_{10} u(n)e(n-1) + \theta_{11} u(n-1)e(n-1) \\
 & + \theta_{12} y(n-1)e(n-1) + \theta_{13} e(n-1) + \theta_{14} e^2(n-1)
 \end{aligned} \tag{4.17}$$

where the “true” parameters and regressors are shown in **bold**. Figure 4.5 shows the Monte-Carlo, bootstrap and regression distribution estimates for each parameter given in Equation 4.17. In each panel, the solid line (“—”) represents the Monte-

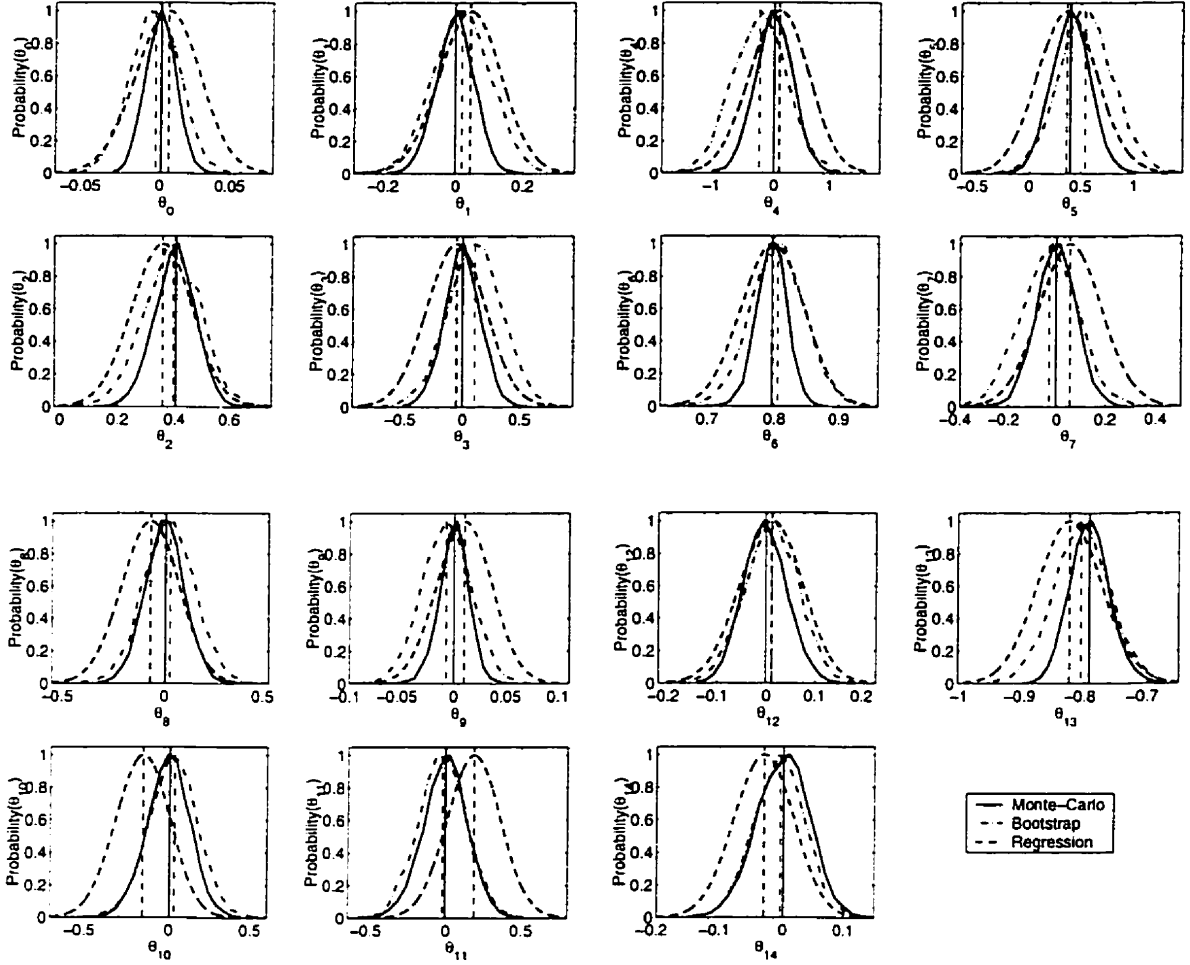


Figure 4.5: Parameter distribution of spurious and true terms for a simple NARMAX model (Model 4.15) when the full model is postulated. Abscissa: Parameter mean. Ordinate: Probability of i th parameter. Vertical Line: Estimated mean for each approach.

Carlo distribution, the dash-dot line (“-.”) represents bootstrap distribution and the dash-dash line (“- -”) represents the regression distribution. The vertical line in each panel is the estimated mean for each parameter distribution. Distribution curves for the Monte-Carlo and bootstrap techniques were plotted using the “hist” function in Matlab, specifying a bin size of 20. For the regression approach the distributions were

plotted by computing parameter statistics [45, 105, 133], substituting these values into the standard formula for a normal distribution [56] and plotting the distribution over a range of parameter values, using a step size of 0.001.

The distributions shown in the plots are initial estimates for the full model, before spurious parameters were removed from the regression to obtain an “improved” estimate. The distributions calculated via the BSD method were closer to the Monte-Carlo distribution, while those associated with the regression method were far from the Monte-Carlo distribution. This result was not surprising since prior to the removal of any parameters the model was over-parameterized; giving an inaccurate estimate of the residuals and, therefore, yielding biased estimates of the distribution. Note that the BSD estimate of each parameter’s standard deviation (distribution spread) and mean was closer to the Monte-Carlo distribution than those obtained from the regression method.

Distribution estimates were recomputed after removing spurious parameters. Parameter distributions calculated for the exact or true model are plotted in Figure 4.6. The results show that the distributions computed using the BSD approach were closer to the Monte-Carlo distribution than those based on regression analysis. It is surprising to see that the regression estimates were significantly different from the Monte-Carlo even when the exact structure was used. This deviation may be because the regression estimates were based on a single realization. For a different realization it may be possible to compute “better” distribution estimates based on regression techniques. Note that this result gives some insight as to why regression methods (e.g., t-test) perform poorly when applied to structure detection; since they may provide poor estimates of parameter statistics. Hence, for this simple NARMAX model the BSD method yielded better parameter statistics than the regression method.

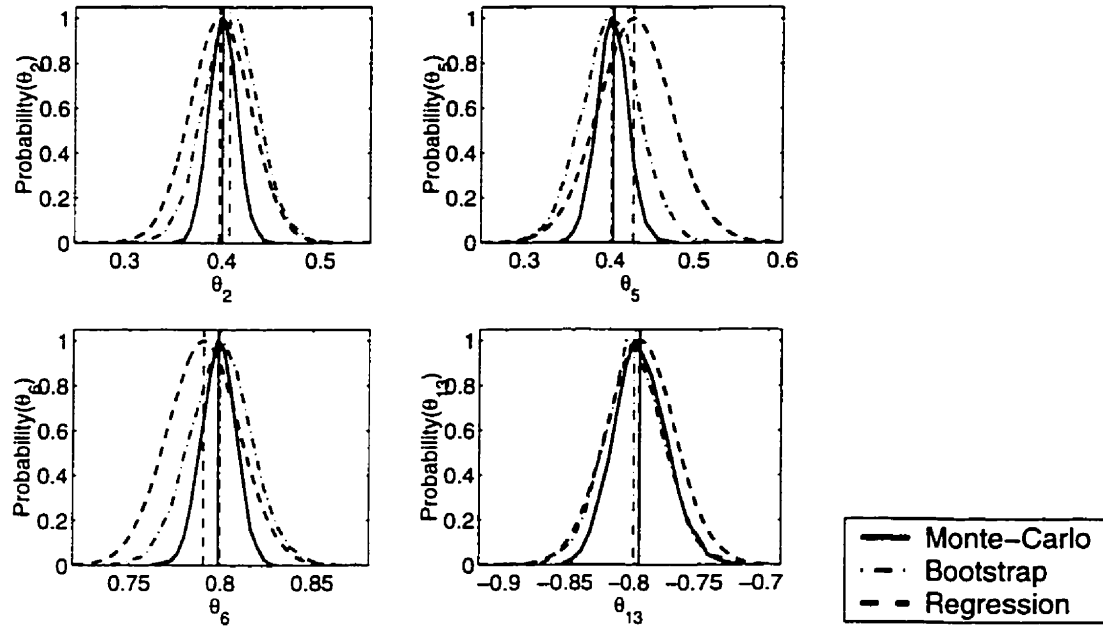


Figure 4.6: Parameter distribution of true terms for a simple NARMAX model (Model 4.15) when the exact model is postulated. Abscissa: Parameter mean. Ordinate: Probability of i th parameter. Vertical Line: Parameter mean for each approach.

4.7.3 Convergence Analysis of Bootstrap

4.7.3.1 Analysis of Decreasing π for Fixed B

To assess the relevance of the theoretical results presented in Bickel and Freedman [16] we empirically determined “how close to zero” $\pi = p^2/N$ must be to achieve consistent results. In this study, the structure of the systems presented in §4.7.1.1-4.7.1.2 were computed using our BSD algorithm. The identification protocol was the same as described in §4.7 except 20 Monte-Carlo simulations were used, $B = 100$ bootstrap replications were generated to assess the distribution of each parameter and $\pi = 1, 0.6, 0.4, 0.1$ were used to determine the data length to be used for parameter estimation.

4.7.3.1.1 Simple NARMAX Model Figure 4.7 shows the empirical probability (in percentage) of selecting the correct model structure for the simple NARMAX model (Model 4.15), when B was fixed and π varied, i.e., N varied. This figure illus-

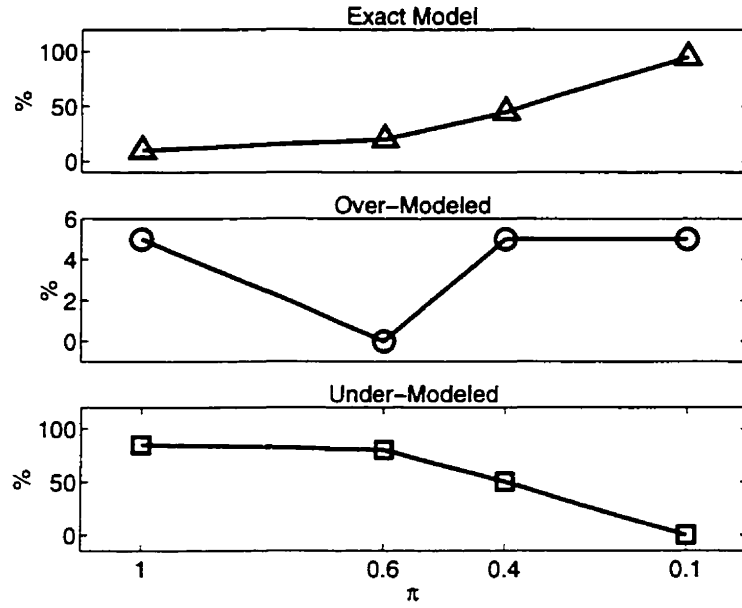


Figure 4.7: Simple NARMAX model (Model 4.15): Rate of model selection as a function of data length N and bootstrap replications $B = 100$. Gaussian, white, zero-mean noise added to output. Abscissa: $\pi = p^2/N = 1, 0.6, 0.4, 0.1$ (i.e., increasing N). Ordinate: Percent selection.

rates: (1) rate of selecting the “exact model”, frequency at which our BSD algorithm computed a model which contained only true system terms, (2) rate of selecting an “over-modeled” system, frequency at which a model with all its true system terms plus spurious parameters was selected and (3) rate of selecting an “under-modeled” system, frequency at which a model without all its true system terms was selected. An under-modeled system may contain spurious terms as well.

The results in Figure 4.7 show that when $\pi > 0.1$ the likelihood of computing the exact model structure was low, 10-40%. However, when $\pi = p^2/N \rightarrow 0.1$ our BSD algorithm computed the correct model structure with high accuracy, $> 95\%$, and had a 0% selection frequency for under-modeling. For the BSD procedure, the rate of selecting the exact model and under-modeled system dominated the structure computation procedure while an over-modeled system was computed at a maximum rate of 5%.

4.7.3.1.2 Highly Over-Parameterized NARMAX Model Figure 4.8 shows the empirical probability of selecting the correct model structure for the over-parameterized NARMAX model (Model 4.16), when B was fixed and π varied. The results in Fig-

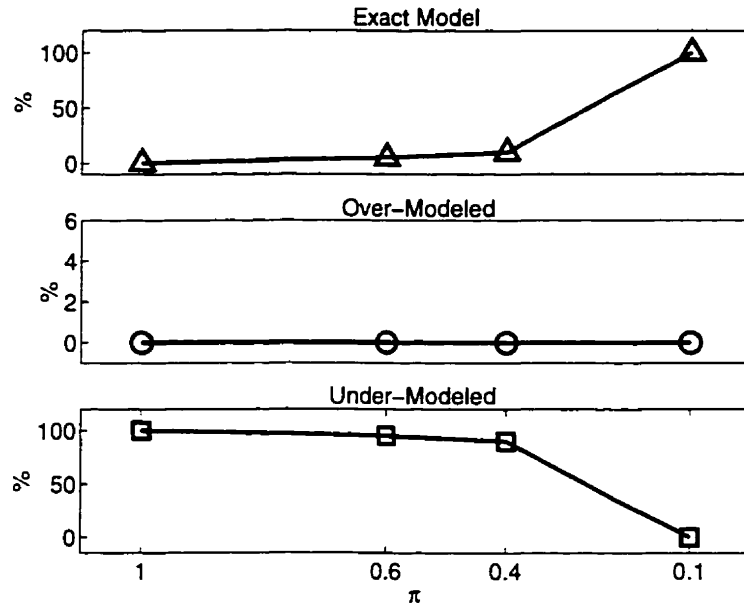


Figure 4.8: Complex NARMAX model (Model 4.16): Rate of model selection as a function of data length N and bootstrap replications $B = 100$. Gaussian, white, zero-mean noise added to output. Abscissa: $\pi = p^2/N = 1, 0.6, 0.4, 0.1$ (i.e., increasing N). Ordinate: Percent selection.

Figure 4.8 show that the bootstrap failed to estimate the true underlying structure for this third-order nonlinear model when $\pi > 0.1$, with an “exact model” selection rate of 0-10%. However, our bootstrap algorithm gave consistent results when $\pi = p^2/N \rightarrow 0.1 \approx 0$. For this highly over-parameterized model, the results, again, show that the rate of selecting the exact model and under-modeled system dominated the structure computation procedure while an over-modeled system was computed at a rate of 0%.

4.7.3.2 Analysis of Increasing B for Fixed π

Next, we appraised how many bootstrap replications, B , are necessary to give accurate estimates of structure and whether increasing B when $p^2/N \rightarrow 0$ could compensate for small N . Here, $\pi = 0.5, 0.3, 0.2, 0.098$ and $B = 40, 80, 120, 160, 200$ for each π .

For computational reasons only Model 4.15 was studied.

4.7.3.3 Simple NARMAX Model

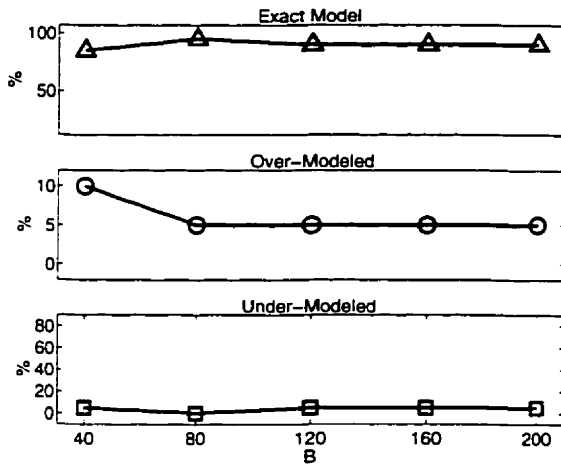
Figure 4.9 shows the empirical probability of selecting the correct model structure for the simple NARMAX system when π was kept constant and B varied. These results illustrate that, in general, increasing the number of bootstrap replications had little effect on the overall probability of selecting the true system structure, i.e., when $\pi = p^2/N \rightarrow 0$. However, when $\pi = p^2/N \rightarrow 0.5$ and $0.2 \approx 0$ increasing B did improve the probability of selecting the optimal structure. Hence, if the condition $\pi = p^2/N \rightarrow \approx 0$ is satisfied, increasing B may be successful in improving the probability of true selection.

4.7.4 Bandlimited Input and Bandlimited Noise

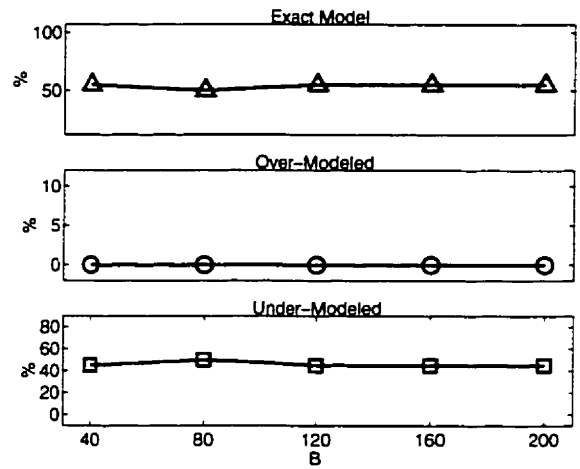
Lastly, we evaluated the performance of our BSD algorithm in the presence of bandlimited noise (Gaussian, white, zero-mean sequence, low-pass filtered with an eighth-order 500 Hz Butterworth filter). We bandlimited the noise process to assess the behavior of our algorithm with low frequency noise. The identification paradigm was the same as described in §4.7 except 20 Monte-Carlo simulations were used, $B = 100$ bootstrap replications were generated to assess the distribution of each parameter, $\pi = 0.1$ was used to determine the data length for parameter estimation and the disturbance had a SNR of 0 dB.

4.7.4.1 Simple NARMAX Model

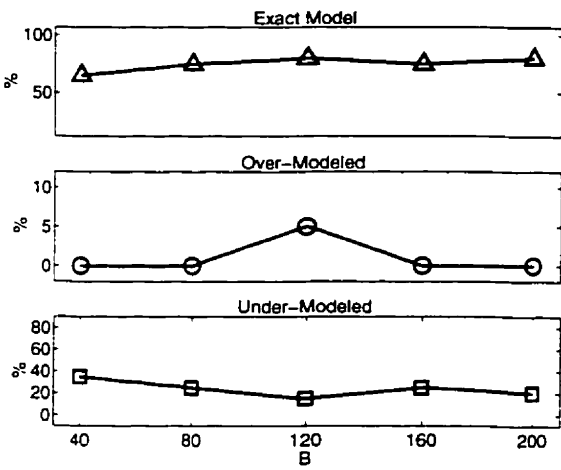
Figure 4.10 shows the results for this simple NARMAX model (Model 4.15). The t-test had a false negative rate of 100% for the parameter associated with the squared input term while stepwise regression had a 100% false negative rate for “all” true parameters except the squared input term. In contrast, our BSD procedure selected true parameters with high accuracy. Stepwise regression had a false positive rate of 0%. The BSD technique had a false positive rate of 5% for an input cross-term and



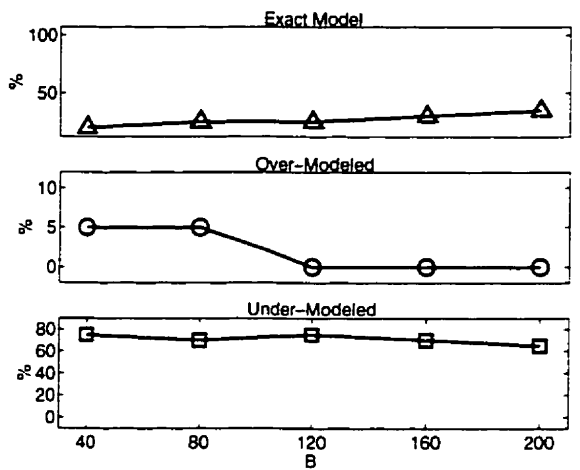
(a) $\pi = 0.098$



(b) $\pi = 0.3$



(c) $\pi = 0.2$



(d) $\pi = 0.5$

Figure 4.9: Simple NARMAX model (Model 4.15): Rate of model selection as a function of $\pi = p^2/N$ and bootstrap replications. Gaussian, white, zero-mean noise added to output. Abscissa: Bootstrap replications $B = 40, 80, 120, 160, 200$. Ordinate: Percent selection.

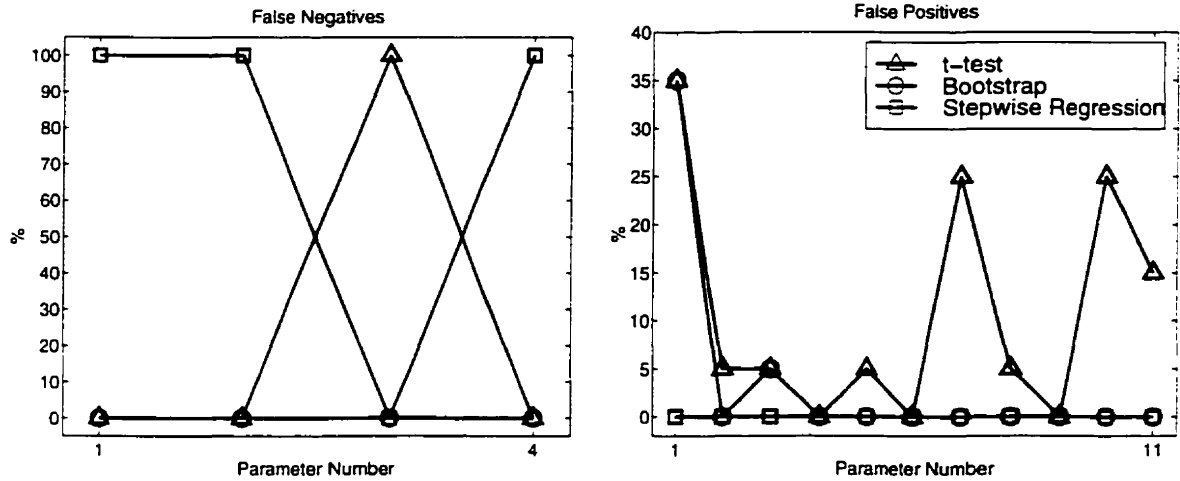


Figure 4.10: Error rate of a simple NARMAX model (Model 4.15) using the t-test, BSD and stepwise regression in the presence of bandlimited output disturbance with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.

35% for the D.C. term, however, the t-test selected more spurious terms and at a greater rate. In the presence of bandlimited noise, the t-test and stepwise regression both failed to select the correct structure for this simple NARMAX model while our BSD method selected the correct structure with high accuracy.

4.7.4.2 Highly Over-Parameterized Model

Figure 4.11 shows the results of our study for the third-order nonlinear system (Model 4.16). Both the t-test and stepwise regression had a high rate of rejecting true terms (10-75%) while the BSD method had a maximum rate of 10% for rejecting true terms. The t-test had an average false negative rate of 25%. Stepwise regression had a rate of 0% for accepting spurious terms while the BSD method had an average 10% false positive rate for several terms. For this highly over-parameterized third-order nonlinear model, in the presence of bandlimited output additive noise, all three methods, the t-test, stepwise regression and BSD method failed to select the correct structure.

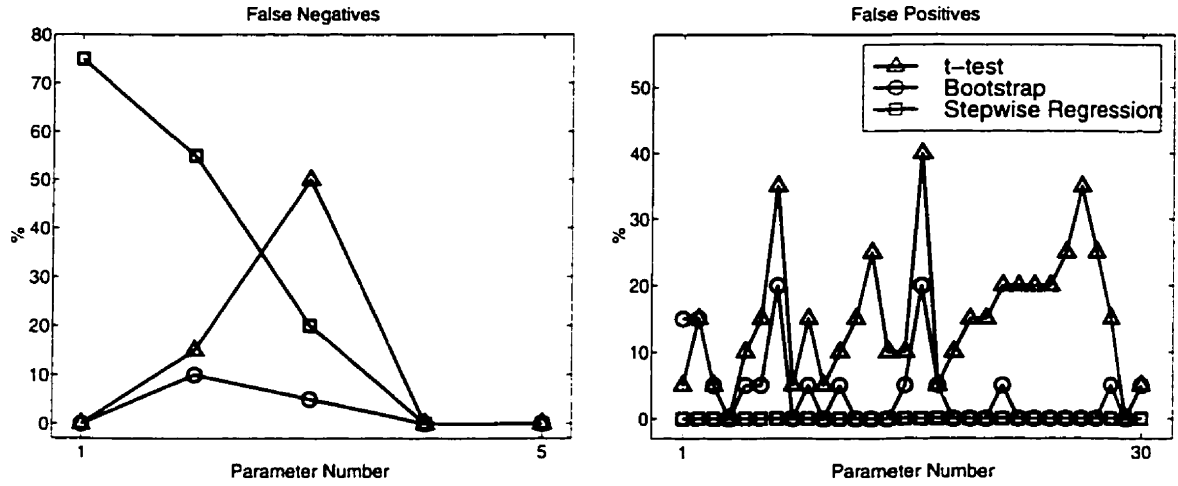


Figure 4.11: Error rate for highly over-parameterized model (Model 4.16) using the t-test, BSD and stepwise regression with 0 dB SNR of a bandlimited noise sequence. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.

4.8 Simulated Biological Example

We assessed our structure computation technique on a simulated model of the vestibulo-ocular reflex (VOR). This system was studied for two reasons: (1) as an example of application of the BSD algorithm to a biological system and (2) the low system order restricts data requirements for structure computation.

Figure 4.12 shows a Hammerstein structure model of the VOR. This model is believed to represent VOR dynamics for normal human subjects [134, 143]. The first

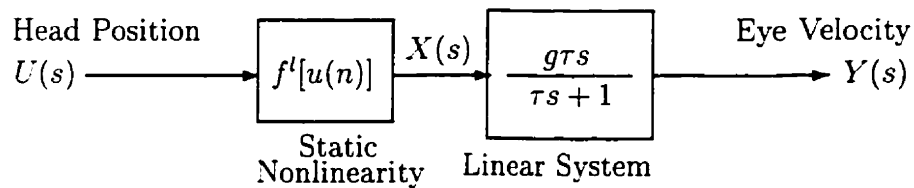


Figure 4.12: A Hammerstein structure model of VOR.

block represents the sensor, horizontal semi-circular canal as a static nonlinearity. This nonlinearity has been shown to be well represented as a third-order static nonlinearity [143]. The second block represents a combination of the central nervous system and eye plant. These dynamics have been shown to be well represented as a

first-order dynamic system with parameters, τ corresponding to time-constant, and g to gain [129, 143].

The first-order dynamics of the central nervous system and eye plant were converted to the discrete domain via the bilinear transform to give the following NARMAX representation of this model:

$$\begin{aligned} y(n) = & \beta_0 y(n-1) + \beta_1 [u(n) - u(n-1)] + \beta_2 [u^2(n) - u^2(n-1)] \\ & + \beta_3 [u^3(n) - u^3(n-1)]. \end{aligned} \quad (4.18)$$

The coefficients β_i , $i = 0, 1, 2, 3$ account for parameters of the continuous-time linear system, nonlinearity and sampling rate.

This model is a theoretical representation of “slow-phase” dynamics of VOR. A realistic representation of VOR dynamics includes two modes of operation: (1) slow-phase and (2) fast-phase dynamics. Issues concerning identification of multiple modes of this system are discussed in Chapter 6.

4.8.1 Simulation Protocol

Input-output data for this model of slow-phase VOR was simulated in continuous-time using Simulink and sampled at a rate of 8 Hz ($T = 0.125$ s). Parameter values used in the simulation were typical values found in experiments (see Table 4.1).

Parameter	Value
c_0	6.45
c_1	3.94×10^{-1}
c_2	1.51×10^{-4}
c_3	-2.84×10^{-7}
τ	10 s
g	-0.7
T	0.125 s

Table 4.1: VOR slow-phase parameter values. Coefficient values of static nonlinearity: c_0 - DC term, c_1 - linear term, c_2 - squared term c_3 - cubic term. Dynamic system parameters: τ - time constant, g - dynamic gain and T - sampling interval.

A Monte-Carlo study of the NARMAX structure describing slow-phase VOR dynamics was performed to assess the applicability of the BSD algorithm for biological systems. Ten Monte-Carlo simulations were used in which each input-output realization was unique and had a unique Gaussian, white, zero-mean, noise sequence added to the output, with 0 dB SNR. The system was excited using bandlimited inputs (uniformly distributed, white, zero-mean, random process, low-pass filtered with an eighth-order 0.5 Hz Butterworth filter). For this study, the system order was assumed to be known. The estimation set consisted of $N = 12,250$ data points ($\pi = 0.1$). The system structure (VOR slow-phase dynamics) was computed using the t-test, stepwise regression and our BSD algorithm, as outlined in §4.7.

4.8.2 Results

The result of structure computation for this model of slow-phase VOR is presented in Figure 4.13. The t-test and BSD technique selected the true parameters with high

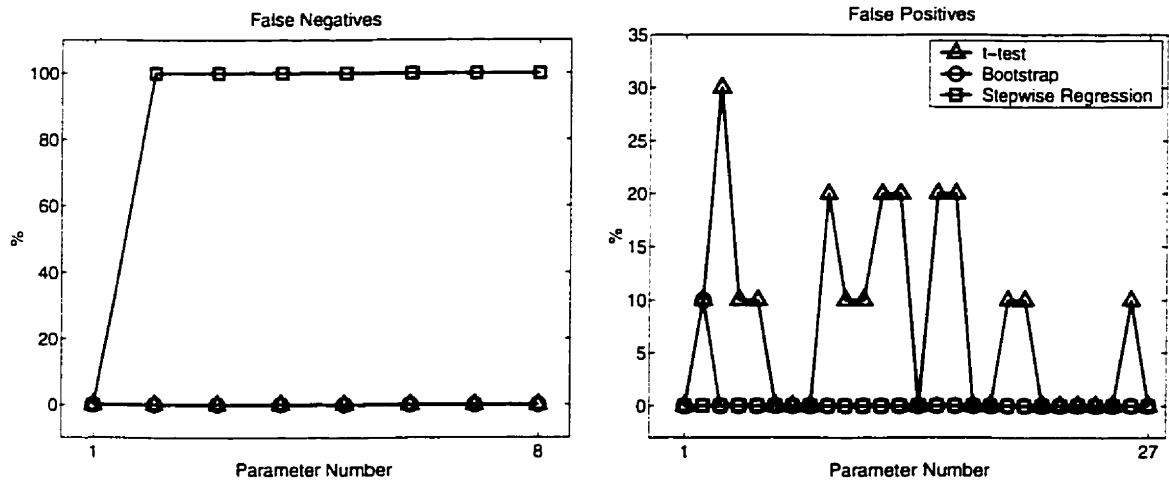


Figure 4.13: Bandlimited Input. Error rate for theoretical model of VOR slow-phase using the t-test, BSD and stepwise regression in the presence of Gaussian, white, zero-mean output additive noise sequence with 0 dB SNR. Abscissa: True/spurious parameter number. Ordinate: Percent true/spurious parameter selection.

accuracy while stepwise regression failed; stepwise regression had a false negative rate of 100% for all but one term. However, stepwise regression had a rate of 0% for

accepting spurious terms while the t-test had a 10-30% false positive rate for several terms. The BSD method had a false positive rate of 10% for only one term. Hence, for this slow-phase model of VOR our BSD algorithm outperformed the t-test and stepwise regression.

4.9 Discussion

4.9.1 Gaussian White Noise

For the Gaussian, white noise case, the t-test, stepwise regression and BSD methods yielded similar results when the number of free parameters was small. However, the BSD technique performed better when there were many candidate terms in the full model. The t-test and stepwise regression techniques both fail in this case but for different reasons. For the t-test, extraneous parameters may model the noise. This often results in a biased estimate of the variance which may give models with incorrect structure. Stepwise regression failed probably because it is sensitive to the order in which terms (regressors) are entered into or removed from the regression [133]. Miller [106] recently stated that Efron's [49] stepwise regression is well documented to stop at a local minimum hence not converge to a global minima.

For the highly over-parameterized model, the BSD method gave better estimates of parameter statistics than the regression technique. The regression method, using the t-test to determine structure, did not give a correct model description likely because initial estimates of the standard deviation were biased and, therefore, its final estimate remained biased. Since the BSD method gave better initial estimates of the standard deviation it was able to obtain a parsimonious model description and accurate parameter statistics. However, the bootstrap method accepted two spurious terms into the model (at a rate of 10%) since the number of bootstrap replications was low, i.e., 50 replications. If the number of replication are increased, estimates of the parameter statistics should improve, thereby, yielding even better results.

4.9.2 Convergence Analysis

Convergence analysis empirically demonstrates that a necessary condition for the BSD to yield a model with correct structure is $\pi = p^2/N \rightarrow 0 \approx 0.1$. The results indicate that when $\pi \not\approx 0.1$, increasing B may have little or no effect in increasing the probability of computing a model with correct structure. However, we demonstrated that when π is in the neighborhood of $\pi = 0.5 - 0.2$ it may be possible to “slightly” improve the correct selection rate by increasing B . This study also suggests that when the estimated structure gives a poor fit to a validation set, i.e., indicating incorrect structure, the system is likely to be under-modeled. In this event the user has no choice but to increase the data length and start anew. If there is some evidence to demonstrate that the computed structure is over-modeled it may be possible to increase N and continue to compute a new structure from the current model.

In addition this study shows, for sufficiently small π ($\pi \approx 0.1$), our BSD algorithm, at worst, provides a model that is slightly over-parameterized. We consider an over-parameterized model “better” than an under-modeled model since it is not possible (with our approach) to re-enter a parameter into the regression (model) once it has been removed. An over-parameterized model which still contains its “true” parameters is clearly more useful than one which has dropped a true parameter.

Our results are given for poor SNR conditions (i.e., 0 dB SNR) therefore they should be widely applicable under most experimental conditions. However, it is important to emphasize that a suitable π and B may vary for different model structures and signal-to-noise ratios. In addition, our results reflect a minimum suitable π value; it is not a conservative value and, therefore, it may be advantageous to decrease π by a factor of ~ 10 for a more conservative value (if there is sufficient recorded data and/or the computing power is available). These results may not apply to all systems as a “golden rule” and should only be used as a “rule-of-thumb”.

4.9.3 Bandlimited Noise

Analysis of the t-test, stepwise regression and BSD technique in the presence of bandlimited noise showed that all three methods failed to compute an accurate structure for the highly over-parameterized model. The t-test and stepwise regression failed for similar reasons. Both rely on white assumptions for the noise process which are violated in this case, giving models with incorrect structure. In general, the BSD does not rely on white assumptions. Therefore, it may be possible to use robust estimation techniques in combination with our BSD method to yield better results under non-white assumptions [35, 122, 147]. Even without using robust estimation techniques, our BSD method gave results superior to those of the t-test and stepwise regression; it had a false negative and false positive rate that was significantly lower.

4.9.4 Biological Example

With respect to the slow-phase VOR model our BSD method had a higher probability of converging to the true structure than the t-test or stepwise regression. However, clinical tests of VOR are often performed with non-ideal harmonic inputs. Certainly performance will degrade for less-than optimal inputs. But in this case, the nature of the response provides another dimension which can be used to still allow identification (see Chapter 6).

4.9.5 Computational Expense

The computational expense of structure detection using the BSD method without implementing our updating scheme is significantly greater than for the “modified” version with updating. For the model with second-order nonlinearity, we observed a four-fold reduction in computation using our updating scheme. Computational savings are realized because rebuilding the entire regressor matrix is not necessary and hence re-orthogonalization of all of Ψ (Q and R) is wasteful. The Householder algorithm is computationally cheaper to solve least-squares problems. This is only true if the formation of Q is not required. However, when updating is used Q is needed.

Therefore, implementing the Householder algorithm to solve this least-squares problem is not suitable and the MGS algorithm is preferable.

4.9.6 Global Search Versus BSD

Since our BSD method is computationally intensive, it may seem that a global search, where ever possible subset of the full model is fit, is more efficient. However, this is not true.

Consider Model 4.15 which contains 15 candidate terms (see Model 4.17). Since only output additive noise was considered in this chapter there are only 10 possible terms (those not involving e) for which we need consider all possible combinations [86]. The noise model terms that will be added to a chosen combination are determined by the output y terms present in the combination [86]. For a model with $p = 10$ possible terms there are

$$\begin{aligned} \binom{p}{r} &= \frac{p!}{r!(p-r)!} \quad \text{for } r = 0, 1, 2, \dots, p \\ &= 1024 \end{aligned} \tag{4.19}$$

possible combinations to try [86].

In §4.7.3 we demonstrated that for $\pi = p^2/N \rightarrow \approx 0.1$ the number of bootstrap replications needed for convergence of our BSD algorithm is $B = 40$. In addition, the number of iterations needed for BSD to yield a parsimonious model description is typically one (if $\pi = p^2/N \rightarrow \approx 0.1$).

For our BSD algorithm, say, we allow for 10–20 iterations for convergence of the “full” noise model. For the global search method, let us not consider the number of iterations needed for convergence of the noise model. In addition, assume that both techniques, global search and BSD, implement the same estimation technique. Even with this optimistic and biased setting, the computational burden of our BSD algorithm requires 2.56–1.28 times less computations than would a global search approach. Under these conditions, our BSD algorithm is computationally more efficient than doing a global search. However, if our BSD algorithm requires, say, two iter-

ations to converge and we allow for 20 iterations for the noise model our method requires 1.56 times more computations than the global search. Hence, depending on the number of iterations required for BSD to converge it may not be as efficient as a brute force global search.

4.9.7 Applicability of BSD to More Complicated Structures

In this chapter, we demonstrated the performance of our BSD algorithm on only three nonlinear systems. These systems were selected for study for the following reasons: (1) all three models are described by only a few terms but the ratio of spurious terms to true terms was large and (2) most systems studied in practice contain at most a second or third-order nonlinearity (or may be reformulated as such). For a comparable number of terms in the full model ($\lesssim 50$), the systems studied in this chapter are general enough to provide insight into the behavior of our algorithm when applied to more complex systems.

Since the BSD algorithm performed well when the model being tested contained both a second and third-order nonlinearity, it suggests that our algorithm provides good estimates of parameter statistics leading to accurate estimates of model structure. Moreover, as noted above, our algorithm “at worst” tends to provide a model that is slightly over-parameterized, if the data record is sufficiently long.

However, BSD may not perform well even when the nonlinearity order is low but the lag order is large [86]. For example, a second order NARMAX model with a maximum input-output lag of 40 and assuming noise free measurements (i.e., $n_u = n_y = 40, n_e = 0, l = 2$) will have in excess of 3,400 candidate terms [86]. Even with noise free data, parameter estimates for the full model could be entirely inaccurate and numerically difficult. Consequently, the BSD algorithm may fail when the system to be identified requires large input or output lags [86]. These difficulties are partly due to over-parameterization of the full model. Over-parameterization results in the residuals being “under-dispersed” and, hence, they will no longer contain useful information about the underlying system.

In many practical identification problems there is often some *a priori* informa-

tion available about the system such as the presence of large input or output lags. If this knowledge is available to the user it may be possible to eliminate many of the candidate terms before starting the structure detection procedure, thereby reducing the problem to one that is of practical dimension and can be solved using our BSD algorithm. For these reasons we expect the results presented in this chapter to be a good representation for comparable structures and, therefore, we expect good behavior of BSD when applied to various linear or nonlinear systems, that are linear-in-the-parameters.

4.9.8 Combined Fast Orthogonal Search and BSD

One difficulty with Korenberg's fast orthogonal search (FOS) [3, 4, 84, 85, 87] is the selection of a threshold value ρ (see §2.6.2.3 and Equation 2.63) for determining significance of a candidate term in the full model. Nevertheless, it may be possible to utilize Korenberg's FOS in combination with our BSD algorithm to provide accurate reduction of NARMAX models.

If an aggressive value for ρ (i.e., small ρ) is selected, it may result in a parsimonious model but one that does not contain all its true terms. However, if a conservative value for ρ (i.e., large ρ) is selected, it will likely give a reduced model that is still over-parameterized but retains its true terms. Therefore, selecting by a conservative value for ρ it may be possible to use a combined FOS-BSD approach to compute structure. With this approach FOS can be used to initially reduce the full model then apply our BSD algorithm to "fine tune" the model to give a parsimonious model with good predictive capability.

4.9.9 Summary of Findings

Simulation studies and convergence analysis showed the following.

1. In the presence of Gaussian, white noise, when the number of free parameters was small, the t-test, stepwise regression and BSD methods yielded similar

results. However, our BSD technique performed better when there were many candidate terms in the full model.

2. In the presence of bandlimited noise, results showed that all three methods failed to compute an accurate structure for a highly over-parameterized NARMAX model.
3. Empirical results demonstrated that a necessary condition for the BSD to yield correct model structure is $\pi = p^2/N \rightarrow 0 \approx 0.1$.
4. Simulations demonstrated that when π is in the neighborhood of $\pi = 0.5 - 0.2$ it may be possible to improve the correct selection rate by increasing the number of bootstrap replications, B .
5. Results showed that for $\pi \approx 0.1$, our BSD algorithm, at worst, provides a model that is slightly over-parameterized.
6. Application of BSD to a theoretical model of slow-phase VOR was successful.
7. Our updating scheme for the BSD algorithm reduces computational requirements significantly.

Using bootstrap, it is possible to compute better estimates of parameter statistics because it requires few assumptions about the error distribution, resulting in more accurate estimates of the model structure. Therefore, the BSD algorithm appears advantageous as a tool for structure detection.

4.10 Conclusions

The results demonstrate that the BSD algorithm is a robust method for detecting the structure of linear regression models and is resistant to noise. This method provides accurate estimates of parameter statistics without relying on assumptions made by traditional procedures and yields a parsimonious description of the system. Convergence results provide an empirical measure for data requirements necessary to

achieve a true model structure. Hence, the BSD method can be used to detect the structure of complex nonlinear dynamic models.

The overall significance of these results has been to demonstrate our BSD algorithm provides consistent and accurate results, requires no *a priori* information regarding the true system output or noise to select a rejection ratio, and works when other methods fail. The importance and relevance of this technique was demonstrated for physiological systems analysis by applying this technique to a theoretical model of slow-phase VOR.

Chapter 5

Bootstrap Model Order Selection

5.1 Overview

Identification of NARMAX models requires determining both the model order and parameter values. Good parameter estimation methods exist if the model order is known, however, model order selection remains a problem.

In this chapter, we develop a bootstrap model order selection (BMOS) algorithm. The bootstrap method is a numerical procedure for estimating parameter statistics that requires few assumptions: the errors must be independent and identically distributed (i.i.d.) with zero mean. The expected error in cross-validation is an appropriate cost function to estimate the “optimal” model order since it does not depend on the estimation set. However, statistical errors may lead to inconsistent or inaccurate estimates of model order for particular realizations. As a means to resolve these inconsistencies, in this study, we examine the hypothesis that the bootstrap method yields an accurate estimate of the “true” prediction error in cross-validation.

Performance of our BMOS algorithm was evaluated by estimating the order of a NARMAX model with a few spurious terms and highest lag order on an even-order nonlinear term. In addition, we show applicability of this technique to biological systems by estimating the order of a theoretical model of slow-phase VOR. Results demonstrate that the BMOS algorithm is a robust method for selecting the order of NARMAX models with a high probability of success.

5.2 Introduction

5.2.1 Model Order

The system order for NARMAX models is defined in Equation 2.21 as

$$O = [n_u, n_y, n_e, l] \quad (5.1)$$

where n_u is the maximum lag on the input, n_y the maximum lag on the output, n_e the maximum lag on the error and l is the maximum nonlinearity order. If the system is assumed to have output additive noise, Equation 2.21 can be reduced to a 3-tuple as

$$n_y = n_e \Rightarrow O = [n_u, n_y, l]. \quad (5.2)$$

Through out this chapter, we assume that the system only contains output additive noise corrupting its output.

Parameter estimation for NARMAX models [18, 22, 24, 37, 61] is straight-forward once the model order is known. The central problem in NARMAX identification is that of selecting the correct model order. Formally the problem is: given the measured output $z(n)$, and the input $u(n)$, where $n = 1, \dots, N$; estimate the parameters $O = [n_u, n_y, l]$ from $n_u = 0, \dots, n_{u_{\max}}$; $n_y = 1, \dots, n_{y_{\max}}$ and $l = 1, \dots, l_{\max}$.

5.2.2 Existing Methods

Several methods for model order selection have been proposed including AIC (Akaike's Information Criterion) [5], Minimum Description Length (MDL) [128] and the False Nearest Neighbors (FNN) algorithm [82]. However, all of these can fail in nonlinear system identification, for various reasons (see §2.6.1).

Both AIC and MDL are widely used in systems analysis to estimate model order. A well known problem with AIC is that it is inconsistent since its variance does not tend to zero for larger sample sizes. This inconsistency is a result of AIC not penalizing the addition of extra parameters heavily enough, i.e., the penalty term does

not decrease fast enough with N to balance the first term in Akaike's criterion [72] (see Equation 2.46). The MDL model order estimator proposed by Rissanen [128] was designed to overcome this problem. AIC is an asymptotic measure since it involves taking the number of samples N to infinity; MDL does not make this assumption. The difference between MDL and AIC is the penalty term (compare Equations 2.46 & 2.50). The penalty term in the MDL definition is larger than that of AIC by a factor of approximately $\log N$, which causes a much steeper minimum. In practice, the order estimated by MDL is normally lower and provides a more consistent estimate for the optimal model order.

Theoretically these model order estimators are a function of residual noise variance. The noise estimate, however, is related to the method that was used to obtain model parameters. The estimated noise variance is a function of output noise variance and parameter variance [31]. Therefore, in practice, the methods used for model order selection and parameter estimation as well as the training sample size are crucial.

As noted in Chapter 2, we conclude that these methods fail for order estimation of nonlinear systems for the following reasons.

1. The number of possible terms for a given order can be very large (see Equation 2.22). Due to over-parameterization residual estimate may be under-dispersed, i.e. biased.
2. Both approaches rely on accurate estimates of $\hat{\sigma}^2$, i.e., accurate estimates of residuals, which may not be the case for incorrect model orders.
3. Both approaches rely on optimal parameter estimates which depend on the data size N . For finite data lengths these methods may give inconsistent estimates.
4. Inadequacy of the penalty term in each method is known to give inconsistent estimates of order for linear systems.

Recently, Kennel *et al.* [82] developed the false nearest neighbors (FNN) algorithm specifically to determine model order for nonlinear systems. This method uses a ratio test to determine whether neighbors, in the regressor space, are "true" or "false",

i.e., whether the neighbors have future outputs that are “far apart”. If the ratio of the distance between two future outputs points, that are “nearest neighbors”, is “large” then the neighbors are considered to be false. The FNN technique is limited to estimating the dynamic order of NARX models, is sensitive to noise and requires selecting a threshold. The selection of this threshold level requires *a priori* knowledge about the true errors and system output, which are seldom available.

5.2.3 Proposed Approach

A model having the correct order will minimize the expected value of the prediction errors [135]. However, statistical errors may lead to inconsistent or inaccurate estimates of model order for particular realizations [56]. One approach to obtain a consistent estimate of model order would be to acquire extensive data sets to minimize expected error. An alternative would be to improve the estimate of expected error in prediction with limited data because in many practical applications it is not possible to collect extensive data sets.

The bootstrap was shown to be a good estimator of parameter statistics, simple to use, and to require few assumptions (see Chapter 4). Consequently, we hypothesize that the bootstrap might also be useful for obtaining a consistent estimate of model order.

5.3 Model Order Selection

5.3.1 The Linear Model

Consider the linear regression model based on model order O

$$Z = \Psi_O \theta_O + e \quad (5.3)$$

where Z is a $N \times 1$ vector of measured outputs, Ψ_O is a $N \times p$ ($p \ll N$) matrix of regressors with full rank (i.e., nonsingular), θ_O is a $p \times 1$ vector of parameters and e is a $N \times 1$ vector of an i.i.d. noise sequence with zero-mean and homoskedastic. Note

that the number of parameters, p , is related to O via Equation 2.22 as

$$\begin{aligned} p &= \sum_{i=1}^l p_i; \quad \text{where } l \text{ is nonlinearity order} \\ p_i &= \frac{p_{i-1}(n_y + n_u + n_e + i - 1)}{i}, \quad p_0 = 1. \end{aligned} \quad (5.4)$$

Let the model be fitted using the least-squares estimator in Equation 2.78 as

$$\hat{\theta}_O = (\Psi_O^T \Psi_O)^{-1} \Psi_O^T Z. \quad (5.5)$$

The predicted outputs are defined as

$$\hat{Z} = \Psi_O \hat{\theta}_O \quad (5.6)$$

and the prediction errors (error in fit) are defined in Equation 2.76 as

$$\hat{\epsilon} = Z - \hat{Z}. \quad (5.7)$$

Ψ is defined as a partitioned regressor matrix for a given order O

$$\Psi_O = [\Psi_{zu} \Psi_{zu\hat{\epsilon}} \Psi_{\hat{\epsilon}}] \quad (5.8)$$

where Ψ_{zu} is a function of z and u only, $\Psi_{zu\hat{\epsilon}}$ represents all the cross products involving $\hat{\epsilon}$, and $\Psi_{\hat{\epsilon}}$ is a polynomial function of the prediction errors only.

Let the measured data be represented as

$$Z = [z_1, \dots, z_N] = [z_{1,1}, \dots, z_{N_e, N_e} | z_{N_e+1,1}, \dots, z_{N, N_v}] = [Z_e | Z_v] \quad (5.9)$$

where Z_e is the estimation set of length N_e and Z_v validation or future set of length N_v .

5.3.2 Average Loss

The efficiency of a model with order O can be measured by the mean squared error or average loss [135],

$$L_{N_e}(O) = \frac{1}{N_e} \sum_{n=1}^{N_e} (z_{e,n} - \psi_{e,n,O}^T \hat{\theta}_O)^2 = \frac{\|Z_e - \hat{Z}_{e,O}\|^2}{N_e} \quad (5.10)$$

where $\|\cdot\|$ is the Euclidean norm. After measuring the data, the objective is to select a model $O \in \mathcal{D}$ so that $L_{N_e}(O)$ will be as small as possible. $\mathcal{D}$ is a collection of some subset of $n_u = \{0, \dots, n_{u\infty}\}$; $n_y = \{1, \dots, n_{y\infty}\}$ and $l = \{1, \dots, l_\infty\}$, i.e., subset of O_∞ . The largest possible $\mathcal{D}$ is the one that minimizes $L_{N_e}(O)$.

Let Z_v be a $N_v \times 1$ vector of future responses at $\Psi_{v,O}$ and assume that Z_v is independent of Z_e . The average conditional expected loss in prediction is [135]

$$\hat{\Gamma}_{N_v}(O) = E \left[\frac{1}{N_v} \sum_{n=1}^{N_v} (z_{v,n} - \psi_{v,n,O}^T \hat{\theta}_O)^2 \mid Z_e, \Psi_e \right] = \sigma^2 + L_{N_v}(O) \quad (5.11)$$

where σ^2 is the noise variance. Therefore, selecting a model with the smallest $L_{N_v}(O)$ over all $O \in \mathcal{D}$ is equivalent to selecting a model with the best prediction ability over all $O \in \mathcal{D}$.

Let

$$P_O = \Psi_O(\Psi_O^T \Psi_O)^{-1} \Psi_O^T \quad (5.12)$$

and

$$\Delta_{N_v}(O) = \frac{\|Z_v - P_O Z_v\|^2}{N_v}. \quad (5.13)$$

Completing the square in Equation 5.11, taking the expected value and substituting the definitions in Equations 5.7, 5.12 & 5.13 gives

$$L_{N_v}(O) = \Delta_{N_v}(O) - \frac{2(Z_v - P_O Z_v)^T \hat{\epsilon}}{N_v} + \frac{\|P_O \hat{\epsilon}\|^2}{N_v}. \quad (5.14)$$

When model O is correct $E[Z | \Psi_O] = \Psi_O \theta_O = P_O Z$, $\Delta_{N_v}(O) = 0$ and

$$L_{N_v}(O) = \frac{\|P_O \hat{\epsilon}\|^2}{N_v}. \quad (5.15)$$

Let O_0 correspond to the model order with the smallest size, i.e., θ_{O_0} contains all components related to the measured output, Z . Therefore [135],

$$\lim_{N_v \rightarrow \infty} \inf \Delta_{N_v}(O) > 0 \quad \text{for any incorrect model order } O, \quad (5.16)$$

(where “inf” is the infimum functional). Model order O_0 is optimal in the sense that it minimizes $L_{N_v}(O)$ over $O \in \mathcal{D}$ for sufficiently large N_v , i.e.,

$$\lim_{N_v \rightarrow \infty} P\{L_{N_v}(O_0) = \min_{O \in \mathcal{D}} L_{N_v}(O)\} = 1. \quad (5.17)$$

The optimal O_0 must be estimated since $L_{N_v}(O)$ involves the unknown parameters θ . Let $\hat{O}$ be the estimate of O_0 based on some model order selection scheme. Therefore, a model order selection procedure is said to be consistent if

$$\lim_{N_v \rightarrow \infty} P\{\hat{O} = O_0\} = 1. \quad (5.18)$$

5.3.3 Effects of Cross-Validation without Bootstrap

One advantage of cross-validation is that, in the limit, the correct model order minimizes the error in prediction. However, in practical situations finite data lengths may lead to statistical errors giving inconsistent results. To evaluate the effect of finite data lengths and various levels of noise in cross-validation we simulated the simple NARMAX model:

$$y(n) = 0.4[u(n-1) + u^2(n-1)] + 0.8y(n-1). \quad (5.19)$$

This system has order $O = [n_u = 1, n_y = 1, l = 2]$.

For this study, a 200 Hz bandlimited input was used (uniformly distributed, white, zero-mean, random sequence, low-pass filtered with an eighth-order 200 Hz But-

terworth filter). With a 200 Hz bandlimited input the nonlinear input term (i.e., $u^2(n-1)$) generates an internal signal that is at least 400 Hz (plus higher-order harmonics). To avoid aliasing, we assumed a sampling rate of $T = 0.0005$ s (2000 Hz); 5 times greater than the internal 400 Hz signal. The search space considered for the optimal model order was from a minimum order of $O_{\min} = [n_u = 0, n_y = 1, l = 1]$ to a maximum of $O_{\max} = [n_u = 2, n_y = 2, l = 3]$.

This model was selected because it is difficult to determine the maximum lag-order for a nonlinear system which contains its maximum lag on an even-order nonlinearity, i.e., $u^2(n-1)$, and because we studied this system in Chapter 4 (see Equation 4.15). Bussgang's theorem states, for Gaussian input data, the cross-correlation of any squared input-output terms will be zero [12, 13]. Therefore, the lag associated with even-order nonlinear terms cannot be determined using first-order correlations; hence "higher-order statistics" must be used [12, 154]. In general, estimating model order for infinite impulse response (IIR) systems, such as the NARMAX model, even using higher-order correlations, also fails because the system "theoretically" could have infinite memory (see Chapter 2 for an example).

5.3.3.1 Effects of Data Length

We assessed the effects of data length using Monte-Carlo simulations. Twenty realizations were generated in which each input-output trial was unique, and had a unique, Gaussian, white, zero-mean, noise sequence added to the output, with 20 dB SNR. The data length was increased from $N_e = N_y = 1,000$ to 6,000 data points. The model that gave the minimum error in prediction in the entire search space (global minimum) was selected as the "optimal" or "true" order.

The results of this study are shown in Figure 5.1. The plot shows our findings in percentage; the rate of correct model selection versus data length. The results show that as the data length was increased the rate of correct selection improved. These results are as expected from theory.

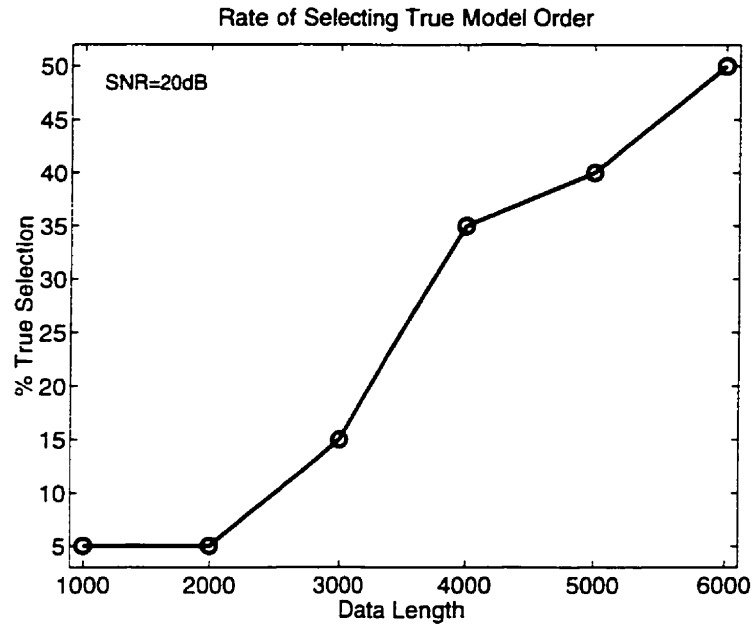


Figure 5.1: Cross-validation selection rate versus data length for a simple NARMAX model. Abscissa: Data length, $N_e = N_v$. Ordinate: % correct selection.

5.3.3.2 Effects of Noise

Next we assessed the effect of noise intensity on estimating the expected error in prediction. Here, the simulation and model order selection protocol was the same as described in the previous study except the data length was kept constant at $N_e = N_v = 5,000$ points while the SNR was decreased from 15 to 0 dB.

The results of this study are presented in Figure 5.2 as a percentage of correct selection versus SNR. The results of this study illustrate that as the SNR was decreased the rate of true selection also decreased. Again, these results are as expected from theory. Consequently, when cross-validation alone is used to estimate model order the probability of true selection may be low, i.e., $P\{\hat{O} = O_0\} \neq 1$.

5.3.4 Error in Prediction

Efron [47, 48] derived a bootstrap estimator for the mean of the prediction error $\Gamma_{N_v}(O)$. Shao [135, 136] showed that this estimator is biased and, therefore, gives inconsistent estimates. Shao [135] proposed a simple, bias-corrected and consistent

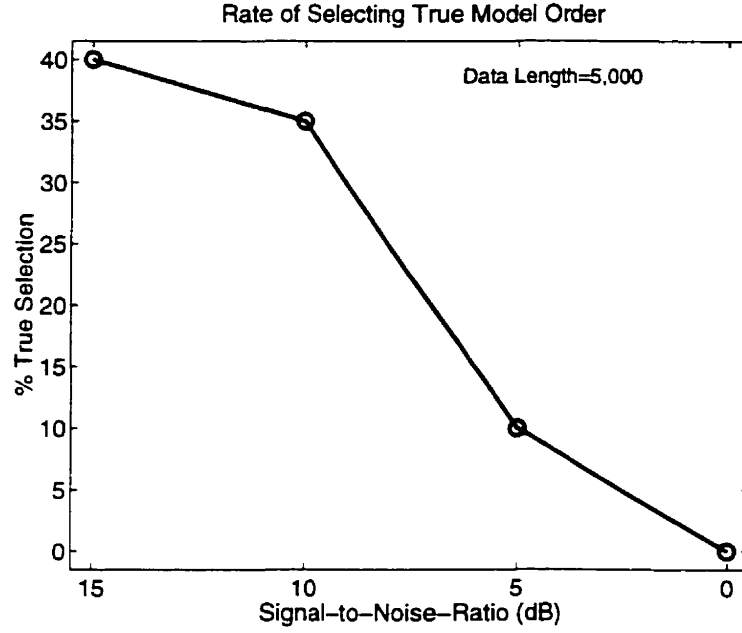


Figure 5.2: Cross-validation selection rate versus SNR for a simple NARMAX model. Abscissa: SNR. Ordinate: % correct selection. (Note that the abscissa is shown in decreasing SNR which corresponds to increasing noise intensity.)

bootstrap estimator for the prediction error based on Efron's [47] original work:

$$\hat{\Gamma}_{N_v, m}(\hat{O}) = E_* \frac{\|Z_v - \Psi_{v, \hat{O}} \hat{\theta}_{\hat{O}, m}^*\|^2}{N_v} \quad (5.20)$$

where E_* denotes expectation operation with respect to bootstrap sampling and $\hat{\theta}_{\hat{O}, m}^*$ is the $p \times 1$ bootstrap analog of estimated parameters, $\hat{\theta}_{\hat{O}}$, based on m i.i.d. pairs $(\psi_{e, n}^*, z_{e, n}^*)$ generated from the empirical distribution putting mass N_e^{-1} on $(\psi_{e, n}, z_{e, n})$, $n = 1, \dots, N_e$; i.e.,

$$\hat{\theta}_{\hat{O}, m}^* = \left(\sum_{n=1}^m \psi_{e, \hat{O}, n}^* \psi_{e, \hat{O}, n}^{*T} \right)^{-1} \sum_{n=1}^m \psi_{e, \hat{O}, n}^* z_{e, n}^*. \quad (5.21)$$

$\hat{\Gamma}_{N_v, m}(\hat{O})$ will be minimized when the optimal model order is achieved, giving a unique minimum [135].

Shao [135] states that, to achieve consistency,

$$\lim_{N_v \rightarrow \infty} P\{L_{m_{N_v}}(O_0) = \min_{\hat{O} \in \mathcal{D}} L_{m_{N_v}}(\hat{O})\} = 1, \quad (5.22)$$

using this bootstrap estimator values of N_v and m need to be selected such that

$$\lim_{N_v \rightarrow \infty} m = \infty \quad \text{and} \quad \lim_{N_v \rightarrow \infty} m/N_v = 0. \quad (5.23)$$

This criterion increases variability among bootstrap observations and achieves consistency [135]. A proof of this result can be found in [135]. Our results presented in Chapter 4 provide a guideline for determining N_e , i.e., $\pi = p^2/N_e \rightarrow 0 \approx 0.1$. Therefore, depending on the number of data points available for validation m must be selected to satisfy $\lim_{N_v \rightarrow \infty} m/N_v = 0$. Based on our results in Chapter 4 we selected $m/N_v = 0.2$. Specifically, a value of 0.2 was selected to ensure that (1) conditions in Equation 5.23 are satisfied and (2) to keep data requirements “reasonable” (both π and m/N_v effect data requirements).

5.3.5 Model Order Selection Using Bootstrap

Application of the bootstrap method to model order selection involves two steps: (1) computing a series of mean-squared error (MSE) replications of prediction, in which “bootstrap data” is used to compute new “bootstrap MSE estimates”, and (2) computing the asymptotic expectation of MSE estimates to determine a global minimum. Bootstrap data is formed by first assuming a model order and then estimating parameter values for the full model. Residuals for this model are calculated, resampled with replacement, and then added to the predicted output to generate bootstrap replications of the output [46, 48]. A number “B” of bootstrap data sets are generated to estimate B bootstrap: (1) parameter replications based on m i.i.d. pairs of the estimation set and (2) MSE replications based on N_v pairs of the validation set. The mean value of B bootstrap MSE replications is the bootstrap estimate of the asymptotic effect of MSE.

The global minimum is determined by storing the bootstrap estimate of the asymptotic expected MSE for each order in an $n_u \times n_y \times l$ multi-dimensional array. The model order is estimated as the index value of the global minimum of this array.

5.3.5.1 BMOS Algorithm

To estimate model order, the BMOS procedure requires the selection of a maximum order, $O = [n_u = n_{u_{\max}}, n_y = n_{y_{\max}}, l = l_{\max}]$, and appropriate values of N_e , N_v and m ; the length of estimation, validation and bootstrap data sets, respectively. This leads to the following algorithm to detect model order in NARMAX models.

1. Select $O = [n_u = n_{u_{\max}}, n_y = n_{y_{\max}}, l = l_{\max}]$, N_e , N_v , and m .
2. Compute an initial estimate of the unknown parameter vector and estimate the residuals for the full model based on N_e pairs of the estimation set.
3. Generate B bootstrap data sets and compute the bootstrap parameter replications (for the full model) based on m pairs of the estimation set.
4. Compute B bootstrap estimates of MSE_b^* , $b = 1, \dots, B$ based on N_v pairs of the validation and the B parameter replications from step 3.
5. Compute the mean value of MSE_b^* replications as $E[\text{MSE}_b^*] = \frac{1}{B} \sum_{b=1}^B \text{MSE}_b^*$.
6. Store $E[\text{MSE}_b^*]$ in a multi-dimensional array, $\hat{\Gamma}(\hat{O})$, at the $(n_{u_i} \text{th}, n_{y_j} \text{th}, l_k \text{th})$ position.
7. If $i < n_{u_{\max}}$ $i = i + 1$, else reset $i = 0$, where i is the current input lag.
8. If $j < n_{y_{\max}}$ $j = j + 1$, else reset $j = 1$, where j is the current output lag.
9. If $k < l_{\max}$ $k = k + 1$, else reset $k = 1$, where k is the current nonlinearity order.
10. If $i = n_{u_{\max}}$, $j = n_{y_{\max}}$ and $k = l_{\max}$ stop, else go to step 2.
11. $[\text{MSE}_{\min}^*, \hat{O}] = \min [\hat{\Gamma}(\hat{O})]$ where $\hat{O}$ is the index of the minimum value, $\text{MSE}_{\min}^*$.

5.4 Simulations

The effectiveness of the BMOS algorithm was assessed using Monte-Carlo simulations of the nonlinear model studied in §5.3.3 and Chapter 4:

$$y(n) = 0.4[u(n-1) + u^2(n-1)] + 0.8y(n-1).$$

Again, this system has order $O = [n_u = 1, n_y = 1, l = 2]$.

Fifty Monte-Carlo simulations were generated in which each input-output realization was unique, and had a unique Gaussian, white, zero-mean noise sequence added to each output realization, with 5 dB SNR. We assumed a sampling rate of $T = 0.0005$ s and each input had the same characteristics as discussed in §5.3.3. For identification, an estimation data length of $N_e = 2,300$ points was used. After an initial estimate of the system parameters was computed, $B = 200$ bootstrap replications were generated, with $m = 400$ points from the estimation data set and $N_v = 2,000$ points from the validation data set (see Equation 5.23), to assess the distribution of the expected value of the cross-validation cost function. To estimate the optimal model order, we searched from a minimum order of $O_{\min} = [n_u = 0, n_y = 1, l = 1]$ to a maximum of $O_{\max} = [n_u = 2, n_y = 2, l = 3]$. The model that gave the minimum error in prediction in the entire search space (global minimum) was selected as the “optimal” or “true” order. We also compared the results of this bootstrap estimator with a cross-validation estimator without bootstrap. For the cross-validation estimator without bootstrap the search space and order selection criteria was the same as for the bootstrap estimator.

5.4.1 Simple NARMAX Model

We studied the simple system shown in §5.3.3 since it has a small number of candidate terms and the maximum lag is associated with an even-order term.

The results of a typical trial for this model is shown in Table 5.1. The result shows that the minimum of $\hat{\Gamma}(\hat{O})$ corresponds to $\hat{O} = [n_u = 1, n_y = 1, l = 2]$. Note particularly that the estimated mean-square error increases for model orders above

l	n_u	n_y		
		1	2	3
1	0	0.0749	0.139	0.0704
	1	0.451	0.329	0.241
	2	0.236	0.229	0.184
	3	0.213	0.171	0.179
2	0	0.6806	1.1642	2.2766
	1	0.0249	0.0344	0.0431
	2	0.0278	0.0384	0.0431
	3	0.0308	0.0425	0.0450
3	0	5.824	8.682	13.146
	1	0.135	0.189	0.254
	2	0.122	0.191	0.256
	3	0.182	0.201	0.262

Table 5.1: MSE result of a typical trial for a simple NARMAX model with bootstrapping. First column: nonlinearity order, l . Second column: input lag, n_u . First row: output lag, n_y . Optimal order (minimum MSE) in bold. Result divided by 1000.

$n_u = 1$, $n_y = 1$, $l = 2$.

To assess the accuracy of this technique for estimating model order, the empirical probability of selecting a particular model order was computed for this example with and without bootstrap. The results for 50 independent runs (in percentage), are given in Table 5.2. The results show that when cross-validation was used without bootstrap the empirical probability of selecting the true model order was low, with a true selection rate of only 52%. However, when bootstrap was used in conjunction with cross-validation (our BMOS algorithm) the rate of selecting the model order was high, with a success rate of 96%.

5.5 Simulated Biological Example

5.5.1 VOR Model

Next, we examined the performance of the BMOS technique with a NARMAX model of VOR slow-phase. Ocular responses consist of interlaced segments classified as

		Without Bootstrap			With Bootstrap		
		n_y			n_y		
l	n_u	1	2	3	1	2	3
1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0
	2	0	0	0	0	0	0
	3	0	0	0	0	0	0
2	0	0	0	0	0	0	0
	1	52	4	2	96	2	0
	2	16	8	0	2	0	0
	3	2	0	2	0	0	0
3	0	0	0	0	0	0	0
	1	0	0	0	0	0	0
	2	6	0	0	0	0	0
	3	8	0	0	0	0	0

Table 5.2: Monte-Carlo simulation results for simple NARMAX model. Left: Empirical probability (percentage) without bootstrapping. Right: Empirical probability (percentage) with bootstrapping, i.e., BMOS algorithm. Optimal order (maximum probability) in bold.

“slow” or “fast”, according to their average speed characteristics, so a time record has a sawtooth-like pattern called ocular nystagmus [57]. This sawtooth-like pattern is a consequence of the VOR switching between two different modes of operation. As stated in Chapter 4, issues concerning identification of multiple modes of this system are discussed in Chapter 6. Here, we focus on a theoretical model of slow-phase VOR shown in Figure 5.3.

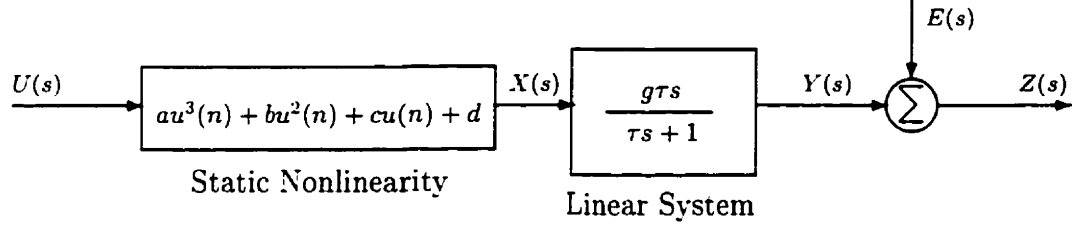


Figure 5.3: Global model structure of slow-phase component for the VOR system.

The slow-phase component is described as (see §4.8) [91]

$$\begin{aligned}
 y(n) = & \beta_0 y(n-1) + \beta_1 [u(n) - u(n-1)] + \beta_2 [u^2(n) - u^2(n-1)] \\
 & + \beta_3 [u^3(n) - u^3(n-1)]
 \end{aligned} \quad (5.24)$$

This physiological system was studied since it exhibits rich dynamic behavior and has a low system order ($O = [n_u = 1, n_y = 1, l = 3]$), hence few candidate terms. Note that this is the same VOR model studied in §4.8.

5.5.2 Simulation Protocol and Results

This VOR model was simulated in continuous-time using Simulink and sampled at a rate of 200 Hz ($T = 0.005$ s). The parameters used in the simulation were the same as those given in Table 4.1 (except the sampling rate, T).

The performance of the BMOS algorithm, as applied to the VOR model, was assessed using Monte-Carlo simulations with bandlimited inputs (uniformly distributed, white, zero-mean, random sequence, low-pass filtered with an eighth-order 10 Hz Bessel filter). Fifty Monte-Carlo simulations were generated in which each input-output realization was unique, and had a unique Gaussian, white, zero-mean, noise

sequence added to the output, with 5 dB SNR. For identification, an estimation data length of $N_e = 12,300$ points was used. An initial estimate of the system parameters was computed, and $B = 200$ bootstrap replications were generated, with $m = 2,400$ points from the estimation data set and $N_v = 12,000$ points from the validation data set (see Equation 5.23), to assess the distribution of $\hat{\Gamma}_{N_v, m}(\hat{O})$. The optimal order was computed from a minimum order of $O_{\min} = [n_u = 0, n_y = 1, l = 1]$ to a maximum of $O_{\max} = [n_u = 2, n_y = 2, l = 4]$.

The result of a representative experiment using bootstrap techniques, is presented in Table 5.3. The result shows that optimal model order is at the global minimum

l	n_u	n_y	
		1	2
1	0	1.76	1.68
	1	1.25	1.25
	2	1.23	1.23
2	0	1.56	1.54
	1	1.00	0.999
	2	1.01	1.02
3	0	1.45	1.46
	1	0.801	0.839
	2	0.834	0.838
4	0	1.47	1.43
	1	0.834	0.840
	2	0.839	0.843

Table 5.3: MSE result of a typical trial for the slow-phase component of the VOR system with bootstrapping. First column: nonlinearity order, l . Second column: input lag, n_u . First row: output lag, n_y . Optimal order (minimum MSE) in bold. Result divided by 1000.

(shown in **bold**): $\hat{O} = [n_u = 1, n_y = 1, l = 3]$.

To assess the accuracy of this technique for estimating model order the empirical probability of selecting a particular model order (in percentage), with and without bootstrapping, was computed (Table 5.4). The result shows that the BMOS technique selected the correct model order with a 90% success rate. However, without bootstrapping the rate of success was 42%.

		Without Bootstrap		With Bootstrap	
		n_y		n_y	
l	n_u	1	2	1	2
1	0	0	0	0	0
	1	0	0	0	0
	2	0	0	0	0
2	0	0	0	0	0
	1	0	0	0	0
	2	0	0	0	0
3	0	0	0	0	0
	1	42	4	90	2
	2	24	12	2	0
4	0	2	0	0	0
	1	14	0	2	0
	2	0	2	0	4

Table 5.4: Monte-Carlo simulation results for the slow-phase component of the VOR system. Left: Empirical probability (percentage) without bootstrapping. Right: Empirical probability (percentage) with bootstrapping, i.e., BMOS algorithm. Optimal order (maximum probability) in bold.

5.6 Discussion

5.6.1 Simulation Studies

For the simple nonlinear model, a cross-validation approach without bootstrap failed to give consistent results for true order selection since it requires asymptotic properties to be invoked, and it is, therefore, limited by finite record lengths. However, our BMOS technique consistently selected the correct model order. With respect to the slow-phase VOR model, the BMOS method again had a higher probability of convergence to the true model order than cross-validation without bootstrap. Clearly, our combined cross-validation/bootstrap algorithm (BMOS) was superior to order selection via cross-validation alone.

5.6.2 Optimal m

In this study, we did not perform a convergence analysis to determine an “optimal” choice of bootstrap resampling size, m . An optimal m may depend on model parameters, noise level/properties and model complexity. Therefore, it is difficult or impossible to determine an optimal m . Instead we heuristically found that as long as the criterion $\pi = p^2/N \rightarrow 0 \approx 0.1$ (see Chapter 4) is not violated a choice of m in the neighborhood of $m \approx 0.2$ may be sufficient to achieve consistency for the bootstrap estimator $E[\Gamma_{N_r}(\hat{O})]$. In this chapter, we again emphasize that a suitable π , m and B may vary for different model structures, perturbations and signal-to-noise ratios. It may be prudent to decrease π and m by a factor of ~ 10 for more conservative values, if sufficient data records and/or computing power is available. These recommendations may not apply to all systems as a “golden rule” and, therefore, should only be used as a “rule-of-thumb”.

5.6.3 Computational Requirements

Our BMOS algorithm requires long data records and considerable computational effort. The computational expense is a result of the data requirement and because B

bootstrap estimates of MSE (MSE_b^* , $b = 1, \dots, B$) are required for each point in the search space to compute an asymptotic bootstrap MSE ($E[\text{MSE}_b^*]$). Consequently, our BMOS algorithm may not be a practical method for order selection for many applications where the data length is limited and a powerful computer is not available or where the user cannot (will not) wait for this algorithm to provide an estimate. However, in the future we expect that computational expense will not be “much” of a limiting factor due to the availability of cheaper and more powerful computers.

5.6.4 Applicability of BMOS to More Complicated Structures

In this chapter, we demonstrated the performance of our BMOS algorithm on only two nonlinear systems. These models were selected for study since they are difficult to identify, for the following reasons: (1) the simple nonlinear model has a maximum lag associated with an even-order nonlinear term and (2) both models are described by only a few terms but, for many system orders, the ratio of spurious terms to true terms may be large.

Since the BMOS algorithm does not rely on correlation techniques it provides an unambiguous estimate of model order and it does not suffer from the effects described by Bussgang’s Theorem (see §2.6.1.1 & §5.3.3). Results from order selection of the second-order nonlinear system ($l = 2$) demonstrated that our BMOS algorithm estimated the lag associated with an even-order nonlinear term with high accuracy. In addition, the BMOS algorithm performed well when the model being tested was highly over-parameterized. This suggests that the BMOS algorithm is not sensitive to having “very accurate” parameter statistics to provide good estimates of model order, as required by AIC and MDL (see §2.6.1.5-2.6.1.7 & §5.2.2). For these reasons we expect the results presented in this chapter to be a good representation for more complicated structures and, therefore, we expect good behavior of BMOS when applied to different linear or nonlinear systems, that are linear-in-the-parameters.

5.6.5 Future Work

Since we only studied the properties of this algorithm in the presence of Gaussian, white, zero-mean output additive noise, further work is necessary to assess how this method performs in the presence of bandlimited output additive noise and when applied to experimental data from biomedical engineering applications. In addition, we did not compare the performance of our BMOS algorithm against any popular techniques such as AIC or MDL. Future work should include a study of this algorithm's ability to select correct model order, i.e., consistency, compared with classic approaches.

5.6.6 Summary of Findings

Simulation studies showed the following.

1. For the simple nonlinear model, a cross-validation approach without bootstrap failed to give consistent results for true order selection.
2. For the simple nonlinear model, our BMOS technique consistently selected the correct model order.
3. The BMOS algorithm requires long data records and considerable computational effort.
4. Our BMOS algorithm may not be a practical method for order selection for many applications where the data length is limited and a powerful computer is not available.

Using our BMOS algorithm, we have demonstrated that it is possible to compute the order of a NARMAX model. The bootstrap computes an asymptotic estimate of the error in cross-validation, resulting in accurate estimates of model order.

5.7 Conclusions

The results demonstrate that the BMOS algorithm provided a robust method for selecting the order of a (1) simple NARMAX model and (2) an example VOR model based on slow-phase dynamics. Repeated trials illustrated that the BMOS algorithm had a high probability of success. This method provides accurate estimates of model statistics without relying on assumptions made by traditional procedures and yields an unambiguous estimate of system order. Hence, the BMOS algorithm may be used to estimate the order of complex nonlinear dynamic models.

The overall significance of these results has been to demonstrate that our BMOS algorithm provides consistent and accurate results. Moreover, the importance and relevance of this technique was demonstrated for biological systems analysis through application of BMOS for order selection of a theoretical model of VOR slow-phase dynamics.

Chapter 6

Parameter Estimation of Hybrid Systems

6.1 Overview

A “hybrid” or “multimode” system is one that can switch between various modes of operation. When a switch occurs from one mode to another, an impulse or discontinuity may result followed by a smooth evolution under the new regime. Characterizing the switching behavior of these systems is not well understood. A consequence of the hybrid nature of these systems is that data available for parameter estimation of any sub-system may be inadequate. As such, identification or parameter estimation of multimode systems remains an unresolved issue. In this chapter, we (1) show how the NARMAX model structure can be used to characterize the impulsive-smooth behavior of these systems and (2) propose a modified extended least squares (MELS) algorithm to estimate the coefficients of such systems.

Although the derivation of the NARMAX model is based on zero-initial-state response, with some extensions (see Chapter 2), the results can be carried over to the nonzero-initial-state case. This makes the NARMAX model structure suitable for modeling nonlinear multimode systems.

The responses of hybrid systems have a discontinuity at each switch time; these will bias parameter estimates if they are not modeled. Therefore, we developed a mod-

ified extended least squares (MELS) algorithm for parameter estimation of multimode systems to address this bias problem. Existing parameter estimation algorithms cannot use data from all measured data segments because smooth continuous behavior is assumed. However, our algorithm allows all the recorded data to be used, and, as such, enjoys the same asymptotic properties as standard least-squares estimators.

We applied this algorithm to a model of the vestibulo-ocular reflex (VOR) and demonstrated that (1) the NARMAX model structure is suited to modeling the dynamics of this nonlinear hybrid system, and (2) the MELS algorithm is a robust method for estimating the coefficients of multimode systems.

6.2 Introduction

A multimode or hybrid system (Figure 6.1) is one that may switch, either by external or internal causes, between a finite number of different modes of operation. Consequently its response may have discontinuities at each mode switch [59].

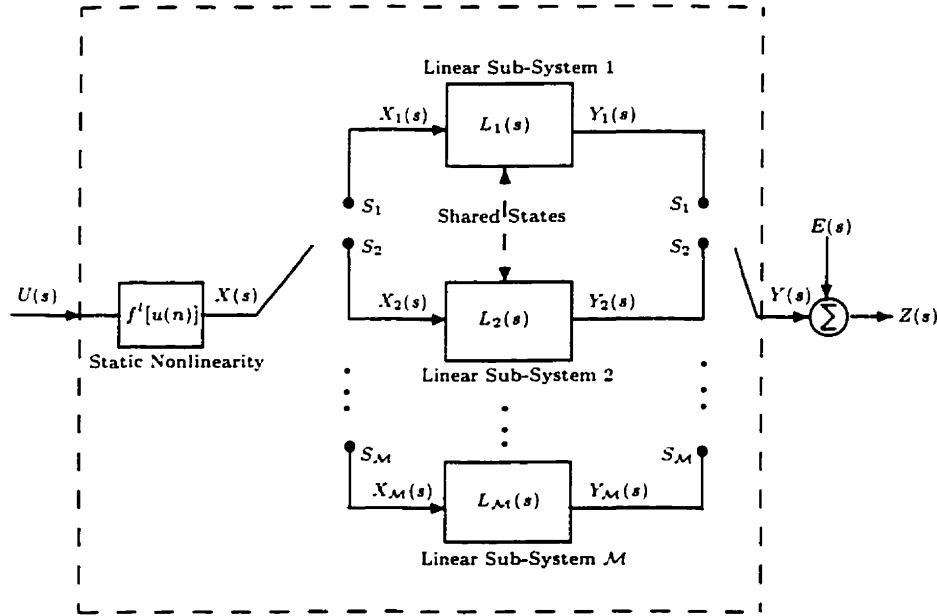


Figure 6.1: General Hammerstein model structure for a $\mathcal{M}$ mode hybrid system with output additive noise where $U(s)$ is the input, $Y(s)$ the true ($Y(s)$ is the selected $Y_m(s)$ driven by $X_m(s)$, $m = 1, 2, \dots, \mathcal{M}$) system output, $E(s)$ a Gaussian, white, zero-mean, noise sequence and $Z(s)$ the measured output.

We consider multimode systems in which the modes of operation are characterized as finite-dimensional, nonlinear, time-invariant, difference equations since they may include nonzero initial conditions. The assumptions we made for this system description are (1) the switch times are known for each sub-system, (2) the output additive noise sequence, e is Gaussian, white, zero-mean and (3) the system is non-zeno, i.e., cannot switch an infinite number of times in a finite time span [6, 7, 14, 67, 152].

The vestibulo-ocular reflex (VOR) is well known to exhibit nonlinear “hybrid” behavior [40, 57, 132]. Presently, descriptions of the VOR rely on linear *a priori* modeling methods [57, 124]. These methods provide convenient means of characterizing slow and fast phase dynamics. However, many models do not account for the rich dynamic behavior due to nonlinearities, therefore limiting their usefulness in diseased cases because of mode interactions through initial conditions [124].

Parameter estimation involves determining values for unknown system coefficients. Many parameter estimation techniques for nonlinear systems depend critically on the choice of model structure, the source of noise within the system and the input excitation [22]. Most parameter estimation algorithms for linear systems cannot be applied directly to NARMAX systems because they assume that the noise terms in the model are independent.

In many situations parameter estimation or identification may be difficult or impossible if the recorded data is not sufficiently long. This is a common problem in many multimode systems since it may not be possible to obtain “long” record lengths due to the switching behavior of the system. Hence, no single data segment may be long enough for parameter estimation, given noise considerations.

The extended least-squares (ELS) algorithm [22, 61] yields unbiased estimates for NARMAX models (see Chapter 2). However, ELS cannot estimate the impulsive behavior of multimode systems or use more than a single measured data segment. Consequently, we will develop a modified ELS (MELS) algorithm to determine parameter values of nonlinear multimode systems, which can take advantage of multiple short data segments. The development in this chapter is specific to the NARMAX polynomial class.

6.3 Multimode Model Formulation

Consider a dual mode system structure shown in Figure 6.2. This is a simple case of

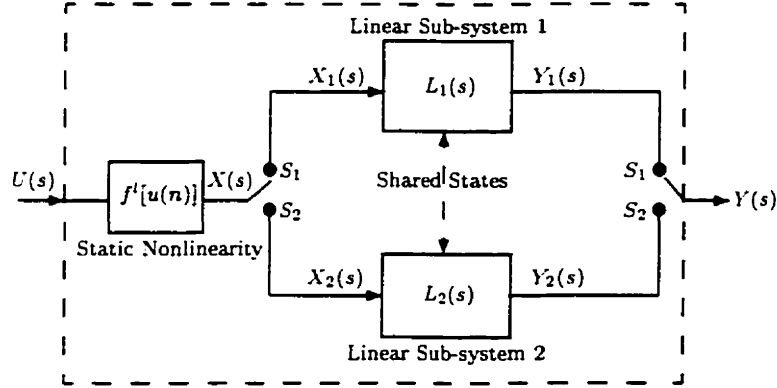


Figure 6.2: Hammerstein model structure for a dual mode hybrid system.

the general form described in Figure 6.1. Let $f^l(\cdot)$, $Y_1(s)$ and $Y_2(s)$ be defined as:

$$\begin{aligned} f^l(\cdot) &= a + bu(n) + cu^2(n) + du^3(n), \\ Y_1(s) &= \frac{K_1 s}{s + p_1} X_1(s) + \frac{Y_{1i}(o)}{s + p_1}; \quad i = 1, \dots, q, \\ Y_2(s) &= \frac{K_2}{s + p_2} X_2(s) + \frac{Y_{2\ell}(o)}{s + p_2}; \quad \ell = 1, \dots, r \end{aligned} \quad (6.1)$$

where $Y_{1,2}(o)$ represents the initial condition in continuous-time and q, r are the number of switches, i.e., data segments, of sub-system one and two, respectively. The two pathways are decoupled, analyzed separately, and then recombined to yield the overall input-output relationship, provided initial conditions are modeled.

The linear system dynamics can be converted to the discrete domain via the bilinear transform to give

$$\begin{aligned} y_1(n) &= \gamma_1 y_1(n-1) + \gamma_2 [x_1(n) - x_1(n-1)] \\ &\quad + \kappa_1 \delta_{11}(n-j) + \dots + \kappa_i \delta_{1i}(n-j_i); \quad i = 1, \dots, q \\ y_2(n) &= \alpha_1 + \alpha_2 y_2(n-1) + \alpha_3 [x_2(n) + x_2(n-1)] \\ &\quad + \lambda_1 \delta_{21}(n-k) + \dots + \lambda_\ell \delta_{2\ell}(n-k_\ell); \quad \ell = 1, \dots, r \end{aligned} \quad (6.2)$$

where the coefficients γ_v , $v = 1, 2$ and α_w , $w = 1, 2, 3$, account for the parameters of

the continuous-time linear system and sampling rate. The input-output data (x, y) are organized according to mode of operation and denoted by a subscript 1 for sub-system one and 2 for sub-system two. (Note, in general, the number of sub-systems can be any finite number, $m = 1, 2, \dots, \mathcal{M}$.) The Kronecker impulse function, δ , is used to represent the onset of an initial condition in discrete-time. κ_i and λ_ℓ are discrete-time initial conditions (coefficients), used as impulse weights to scale the Kronecker impulse, accounting for the discontinuity at each switch time. The indices j, k represent the lag value of the δ_{1i} th and $\delta_{2\ell}$ th impulse.

Substituting $f^t(\cdot)$ for $x(\cdot)$ into Equation 6.2, collecting terms and combining the overall nonlinear model is

$$\begin{aligned}
 y(n) &= \begin{cases} y_1(n) & \text{Switch Position } S_1 \\ y_2(n) & \text{Switch Position } S_2 \end{cases} \quad (6.3) \\
 y_1(n) &= \gamma_1 y_1(n-1) + \beta_1 [u_1(n) - u_1(n-1)] \\
 &\quad + \beta_2 [u_1^2(n) - u_1^2(n-1)] + \beta_3 [u_1^3(n) - u_1^3(n-1)] \\
 &\quad + \kappa_1 \delta_{11}(n-j) + \dots + \kappa_i \delta_{1i}(n-j_i); \quad i = 1, \dots, q \\
 y_2(n) &= \alpha_1 + \alpha_2 y_2(n-1) + \vartheta_1 [u_2(n) + u_2(n-1)] \\
 &\quad + \vartheta_2 [u_2^2(n) + u_2^2(n-1)] + \vartheta_3 [u_2^3(n) + u_2^3(n-1)] \\
 &\quad + \lambda_1 \delta_{11}(n-k) + \dots + \lambda_\ell \delta_{1\ell}(n-k_\ell); \quad \ell = 1, \dots, r
 \end{aligned}$$

which is a NARMAX model since it (1) includes both linear and nonlinear input-output terms and (2) is linear-in-the-parameters. Notice that although the response function of a system varies for different initial states, the input-output model for the system will always be the same regardless of initial states provided the system is maintained within a region around an equilibrium point [37]. This simple model can easily be extended to the general $\mathcal{M}$ mode NARMAX model given in Equation 2.20.

6.4 Modified Extended Least-Squares

Since the input-output model for a system is the same regardless of the initial state, the ELS algorithm can be used to estimate both the input-output and noise models. However, implementing ELS as presented in §2.6.3.3 will result in a biased estimate of the coefficients. Since multimodal systems may produce an impulse at each switch time, these impulses cause a bias of the estimated parameters if they are not modeled. This bias is shown using arguments similar to those in Equation 2.75. For a given mode of operation, the extended least squares formulation is defined as

$$\begin{aligned} Z_m &= \Psi_m \theta_m + \varepsilon_m, \quad m = 1, 2, \dots, \mathcal{M}, \quad \text{where} \\ \Psi_m &= [\Psi_{z_m u_m} \Psi_{z_m u_m \hat{\varepsilon}_m} \Psi_{\hat{\varepsilon}_m}] \quad \text{and} \\ \varepsilon_m(n) &= \delta_m(n - j_1) + \delta_m(n - j_2) + \dots + \delta_m(n - j_i) + e_m(n), \quad i = 1, \dots, I. \end{aligned} \quad (6.4)$$

For a given mode of operation m , Ψ_m is defined to be a partitioned regressor matrix where $\Psi_{z_m u_m}$ is a function of z_m and u_m only, $\Psi_{z_m u_m \hat{\varepsilon}_m}$ represents all the cross products involving $\hat{\varepsilon}_m$, and $\Psi_{\hat{\varepsilon}_m}$ is a polynomial function of the prediction errors only. Note that z_m is the measured or noise corrupted output for a given mode, m .

Taking the expectation of $\hat{\theta}_{m \text{ ELS}}$ gives

$$\begin{aligned} E[\hat{\theta}_{m \text{ ELS}}] &= (\Psi_m^T \Psi_m)^{-1} \Psi_m^T E[Z_m] \\ &= (\Psi_m^T \Psi_m)^{-1} \Psi_m^T [\Psi_m \theta_{m \text{ ELS}} + \varepsilon_m] \\ &= \theta_{m \text{ ELS}} + (\Psi_m^T \Psi_m)^{-1} \Psi_m^T \varepsilon_m \end{aligned} \quad (6.5)$$

$$\text{therefore, } E[(\Psi_m^T \Psi_m)^{-1} \Psi_m^T \varepsilon_m] \neq 0.$$

This bias is a result of the model error, ε_m , containing unmodeled dynamics due to initial conditions, i.e., scaled impulses. To compute an unbiased estimate of θ_m , we develop an alternative estimation technique based on ELS.

Consider the system shown in Figure 6.1. The data segment(s) for each subsystem are defined as shown in Table 6.1 (See Figure 6.3 for an example of data segmentation.): where N_k is the length of the k th segment, $\sum_{q=1}^k N_q$, is the total

	U	Z	
	$u_{1,1,1,1,0}$	$z_{1,1,1,1,0}$	
U_{11}	$\vdots$	$\vdots$	Z_{11}
	$u_{N_1,N_1,1,1,0}$	$z_{N_1,N_1,1,1,0}$	
	$u_{N_1+1,1,2,1,1}$	$z_{N_1+1,1,2,1,1}$	
U_{21}	$\vdots$	$\vdots$	Z_{21}
	$u_{N_2+N_1,N_2,2,1,1}$	$z_{N_2+N_1,N_2,2,1,1}$	
	$u_{N_2+N_1+1,1,1,2,2}$	$z_{N_2+N_1+1,1,1,2,2}$	
U_{12}	$\vdots$	$\vdots$	Z_{12}
	$u_{N_3+N_2+N_1,N_3,1,2,2}$	$z_{N_3+N_2+N_1,N_3,1,2,2}$	
	$u_{N_3+N_2+N_1+1,1,3,1,3}$	$z_{N_3+N_2+N_1+1,1,3,1,3}$	
U_{31}	$\vdots$	$\vdots$	Z_{31}
	$u_{N_4+N_3+N_2+N_1,N_4,3,1,3}$	$z_{N_4+N_3+N_2+N_1,N_4,3,1,3}$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$u_{(\sum_{q=1}^{k-1} N_q)+1,1,m,i,j}$	$z_{(\sum_{q=1}^{k-1} N_q)+1,1,m,i,j}$	
U_{mi}	$\vdots$	$\vdots$	Z_{mi}
	$u_{(\sum_{q=1}^k N_q),N_k,m,i,j}$	$z_{(\sum_{q=1}^k N_q),N_k,m,i,j}$	

Table 6.1: Data segmentation.

data length, $k = 1, 2, \dots, (j + 1)$ is the segment number, $j = 0, 1, 2, \dots, h$ is the switch number, $m = 1, 2, \dots, \mathcal{M}$ is the sub-system number and $i = 1, 2, \dots, I$ is the segment number of the m th sub-system. $z_{\cdot,1,m,i,\cdot} = \delta_{\cdot,1,m,i,\cdot} + u_{\cdot,1,m,i,\cdot} + \dots + u_{\cdot,l,m,i,\cdot}$, is the first output of segment i of sub-system m . U_{mi}, Z_{mi} are the input-output data of the corresponding sub-system and segment. We define the concatenation of all input-output segments of the m th sub-system to be

$$\begin{aligned}
 \mathcal{U} &= [U_{m1}, U_{m2}, \dots, U_{mi}]; \quad m = 1, 2, \dots, \mathcal{M} \\
 \mathcal{Z} &= [Z_{m1}, Z_{m2}, \dots, Z_{mi}]; \quad i = 1, 2, \dots, I.
 \end{aligned} \tag{6.6}$$

Let $\mathcal{U}$ and $\mathcal{Z}$ be $N_m \times 1$ vectors of measured input and output, respectively.

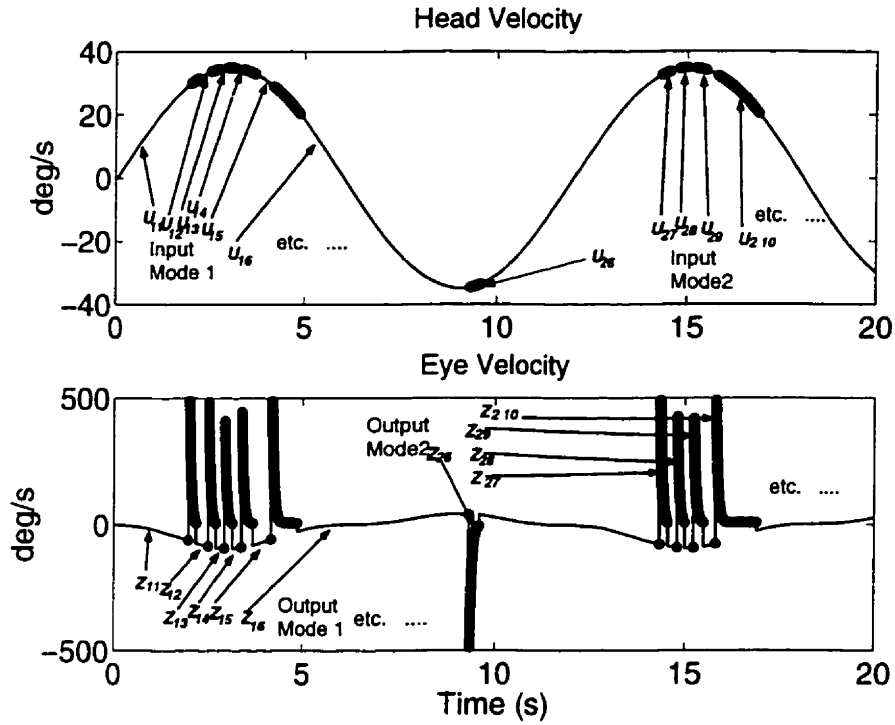


Figure 6.3: Example of VOR input-output data segmentation for a dual mode system.

A least-squares formulation for this system is

$$Z = \Psi_{ZU} \theta + \hat{\epsilon} \quad (6.7)$$

where Z is a $N_m \times 1$ vector of measured outputs, Ψ_{ZU} is a non-singular $N_m \times p$ matrix of regressors, based on input-output only, θ is a $p \times 1$ vector of unknown parameters, and $\hat{\epsilon}$ is a $N_m \times 1$ vector of prediction errors.

To estimate an initial parameter set of the m th sub-system the regressor matrix, Ψ_{ZU} , is formed similar to Ψ_{zu} (see §2.6.3.2), except each data segment is considered as a new input-output segment. Therefore, the regressor matrix is a concatenated matrix of sub-regressor matrices formed from individual data segments as

$$\Psi_{ZU} = \begin{bmatrix} \Psi_{Z_{m1}U_{m1}} \\ \Psi_{Z_{m2}U_{m2}} \\ \vdots \\ \Psi_{Z_{mI}U_{mI}} \end{bmatrix}. \quad (6.8)$$

A modified extended regressor matrix, Φ , used to estimate the noise model and impulses is defined as

$$\Phi = [\Psi \ \Psi_\delta] \quad (6.9)$$

where Ψ is defined as in Equation 6.4. The extension, Ψ_δ , represents the effects of initial conditions when a switch occurs. The number of columns in Ψ_δ is equal to the number of data segments, i.e., number of switches. Note that since multiple switches can occur the columns of Ψ_δ contain impulses lagged in time. In addition, the number of impulses or initial conditions in each segment is equal to the dynamic order of the linear system. For the simple case presented in §6.3 there is one initial condition per segment since the dynamics are of order one. To model impulses due to output and noise the effect of the forcing function needs to be removed, i.e., current inputs. The first input point is subtracted (up to the order of nonlinearity, l) from the first output, of each segment. The columns of Ψ_δ are formed as

$$\begin{aligned} \Psi_\delta = & [(\mathbf{z}_{\cdot,1,m,i,\cdot} - \hat{\theta}_1 \mathbf{u}_{\cdot,1,m,i,\cdot} - \dots - \hat{\theta}_l \mathbf{u}_{\cdot,1,m,i,\cdot}^l \\ & - \{\mathbf{z}_{\cdot,1,m,i,\cdot} - \hat{\theta}_1 \mathbf{u}_{\cdot,1,m,i,\cdot} - \dots - \hat{\theta}_l \mathbf{u}_{\cdot,1,m,i,\cdot}^l\} \hat{\epsilon}_{\cdot,1,m,i,\cdot}) \delta(\cdot)]. \end{aligned}$$

If the impulses due to output and noise are modeled separately as

$$\begin{aligned} \Psi_\delta = & [\begin{pmatrix} \mathbf{z}_{\cdot,1,m,i,\cdot} - \hat{\theta}_1 \mathbf{u}_{\cdot,1,m,i,\cdot} - \dots - \hat{\theta}_l \mathbf{u}_{\cdot,1,m,i,\cdot}^l \end{pmatrix} \delta(\cdot) \\ & \begin{pmatrix} \mathbf{z}_{\cdot,1,m,i,\cdot} - \hat{\theta}_1 \mathbf{u}_{\cdot,1,m,i,\cdot} - \dots - \hat{\theta}_l \mathbf{u}_{\cdot,1,m,i,\cdot}^l \end{pmatrix} \hat{\epsilon}_{\cdot,1,m,i,\cdot} \delta(\cdot)] \end{aligned}$$

Φ will be singular or ill-conditioned.

The extended parameter set

$$\hat{\theta}_{MELS} = (\Phi^T \Phi)^{-1} \Phi^T \mathcal{Z} \quad (6.10)$$

can be shown to be an unbiased estimate of θ_{MELS} since the residuals are zero-mean, in the limit, when all impulses and errors are estimated. This modified extended least-squares algorithm has the same asymptotic properties as OLS and ELS since

it models all dynamics due to initial conditions or discontinuities as well as system dynamics. This leads to the following algorithm to estimate parameters of nonlinear hybrid systems.

6.4.1 MELS Algorithm

1. Segment the input-output data record according to mode of operation.
2. Form Ψ_{zu} for the m th sub-system, for $m = 1, 2, \dots, \mathcal{M}$, compute an initial estimate of the unknown parameter vector and estimate the residuals.
3. Form Φ for the m th sub-system, compute an estimate of the extended parameter vector and compute the residuals.
4. Go to 3 until convergence.
5. Estimate parameters of the next sub-system, $m = m + 1$. Go to 2 until $m > \mathcal{M}$.

6.5 VOR Model

Ocular responses during head perturbations consist of intermingled segments classified as “slow” or “fast”, according to their average speed characteristics. This describes the vestibulo-ocular reflex (VOR) and a time record of the response has a sawtooth-like pattern called ocular nystagmus (Figure 6.4). This sawtooth-like pattern is a consequence of the VOR switching between two different modes of operation: the slow-phase which stabilizes the eye in space ($\dot{E} \approx -\dot{H}$) and the fast-phase which re-orientes the eye in the direction of head rotation ($E \propto \dot{H}$). Note that the VOR is a dual mode hybrid system (see Figure 6.2), a simple case of the general form in Figure 6.1.

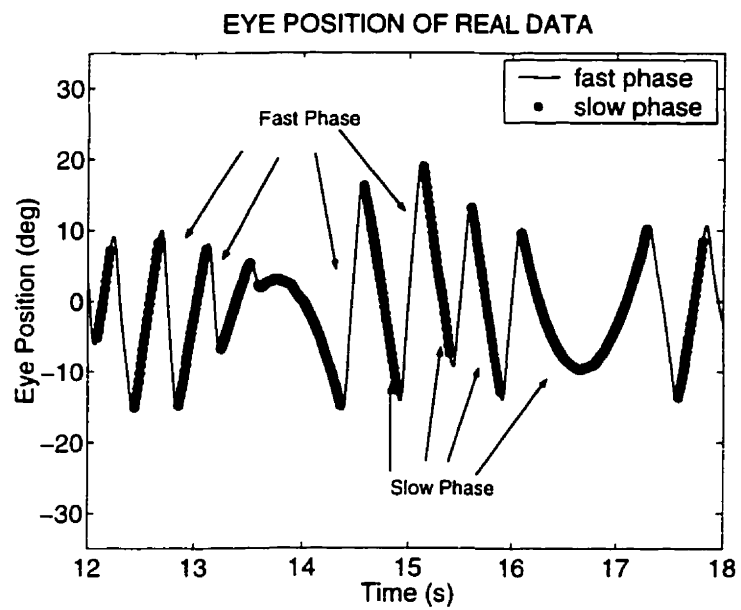


Figure 6.4: Typical plot of VOR output.

In Figure 6.2 let $f^l(\cdot)$, $Y_1(s)$ and $Y_2(s)$ be redefined as:

$$\begin{aligned} f^l(\cdot) &= a + bu(n) + cu^2(n) + du^3(n), \\ Y_1(s) &= \frac{K_1/\tau_1}{\tau_1 s + 1} X_1(s) + \frac{Y_{1i}(o)}{\tau_1 s + 1} = \frac{K_1}{s + p_1} X_1(s) + \frac{Y_{1i}(o)/\tau_1}{s + p_1}; \quad i = 1, \dots, q, \\ Y_2(s) &= \frac{K_2/\tau_2}{\tau_2 s + 1} X_2(s) + \frac{Y_{2\ell}(o)}{\tau_2 s + 1} = \frac{K_2}{s + p_2} X_2(s) + \frac{Y_{2\ell}(o)/\tau_2}{s + p_2}; \quad \ell = 1, \dots, r \end{aligned} \quad (6.11)$$

where $Y_1(s)$ and $Y_2(s)$ are first order approximations for the modes (phases) of the VOR.

Clinically, vestibular patient evaluation relies on the characteristics of only VOR slow-phases. However, our method can provide both the slow and fast phase dynamics in discrete-time. A NARMAX description of VOR slow-phases, $y_1(n)$, and fast-phases, $y_2(n)$ of the model in Equation 6.11 is:

$$\begin{aligned} y(n) &= \begin{cases} y_1(n) & \text{Switch Position } S_1 \\ y_2(n) & \text{Switch Position } S_2 \end{cases} \\ y_1(n) &= \beta_1 + \beta_2 y_1(n-1) + \beta_3 [u(n) + u(n-1)] \\ &\quad + \beta_4 [u^2(n) + u^2(n-1)] + \beta_5 [u^3(n) + u^3(n-1)] \\ &\quad + \kappa_1 \delta_{11}(n-j) + \dots + \kappa_i \delta_{1i}(n-j); \quad i = 1, \dots, q \\ y_2(n) &= \vartheta_1 + \vartheta_2 y_2(n-1) + \vartheta_3 [u(n) + u(n-1)] \\ &\quad + \vartheta_4 [u^2(n) + u^2(n-1)] + \vartheta_5 [u^3(n) + u^3(n-1)] \\ &\quad + \lambda_1 \delta_{11}(n-k) + \dots + \lambda_\ell \delta_{1\ell}(n-k); \quad \ell = 1, \dots, r. \end{aligned} \quad (6.12)$$

Table 6.2 shows the relationship of the discrete-time parameters to the underlying continuous-time parameters. This physiological system was studied since it exhibits rich multimode behavior.

6.6 Simulation

The accuracy of our MELS parameter estimation algorithm was validated by simulating the VOR model (Figure 6.2) in continuous-time using Simulink. The parameters

DT Coefficient	Relationship to CT
β_1, ϑ_1	$\frac{(2K_{1,2}aT)}{2+p_{1,2}T}$
β_2, ϑ_2	$\frac{-(-2+p_{1,2}T)}{2+p_{1,2}T}$
β_3, ϑ_3	$\frac{(2K_{1,2}bT)}{2+p_{1,2}T}$
β_4, ϑ_4	$\frac{(2K_{1,2}cT)}{2+p_{1,2}T}$
β_5, ϑ_5	$\frac{(2K_{1,2}dT)}{2+p_{1,2}T}$
κ_1, λ_1	$\frac{(Y_{1,2}(0)p_{1,2})}{2+p_{1,2}T}$

Table 6.2: Discrete-time relationship of NARMAX model parameters to underlying continuous-time parameters.

used in the simulation were typical values found in experiments and are shown in Tables 6.3 & 6.4 [57]. The system was perturbed using a sinusoid input (1/6 Hz

CT Coefficient	Value
τ_1	15 s
τ_2	50 ms
K_1	-9.42
K_2	4.44

Table 6.3: Continuous-time coefficient values. τ_1 : slow-phase time-constant, τ_2 : fast-phase time-constant, K_1 : slow-phase gain (velocity gain = $K_1/\tau_1 = -0.628$) and K_2 : fast-phase gain (velocity gain = $K_2/\tau_2 = 0.222$).

frequency and 180 deg/s amplitude) while a Gaussian, zero-mean, noise sequence with 5 dB SNR was added to the output (Figure 6.5). The system input-output was sampled at 600 Hz. Thirty one slow and thirty fast-phase segments were used for identification. Extended least-squares and our modified extended least-squares algorithm were used to estimate system parameters from simulated data. ELS ignores the effect of switching and associated initial conditions. Therefore, it is equivalent to the traditional analysis of slow-phases, in the clinic, as belonging to a continuous-smooth envelope, where the gaps due to fast-phases are interpolated to produce a continuous slow-phase response. The MELS treats each slow-phase segment as a transient response including both the forced input and the switching effects. Tables

NL Coefficient	Value
a	3.00×10^{-1}
b	1.20
c	-3.00×10^{-4}
d	-1.50×10^{-6}
T	1.67×10^{-3} s

Table 6.4: Coefficient values of static nonlinearity. a : DC term, b : linear term, c : squared term, d cubic term and T: sampling interval.

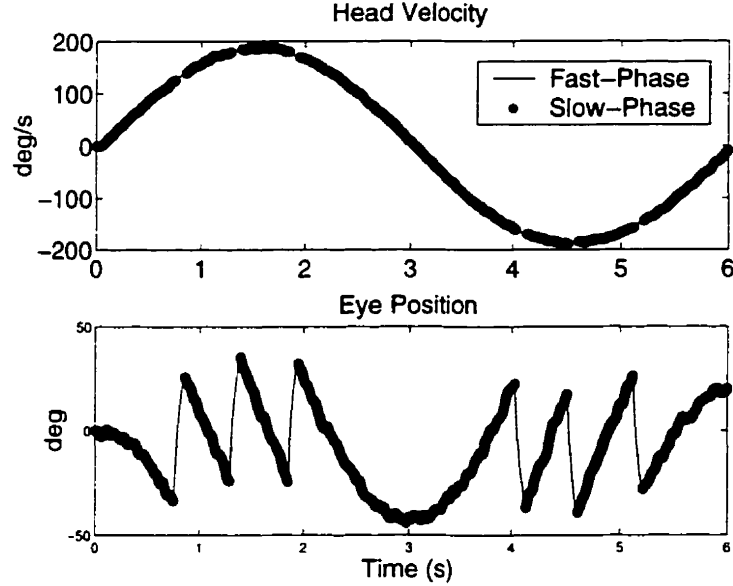


Figure 6.5: Simulation input-output data used for identification.

6.5 and 6.6 compare the results of the ELS and MELS algorithms, after estimating the coefficients of both slow and fast modes in this simulated VOR. In both tables the first column contains the theoretically computed parameter values, the second column contains the estimated parameter values using the ELS algorithm and the third column contains the estimates given by our MELS algorithm. As expected, the ELS estimates are highly biased in both slow and fast-phases. However, our MELS algorithm yields accurate estimates of system parameters in both modes, even in the presence of output additive noise (5dB SNR = noise 75% of signal amplitude).

Next, we estimated the continuous-time parameters using the identified discrete-time parameters in Tables 6.5 & 6.6. The continuous-time parameters were estimated

Term	DT Coefficients of Sub-system 1		
	θ	$\hat{\theta}_{ELS}$	$\hat{\theta}_{MELS}$
$\beta_1 (DC)$	-3.14×10^{-4}	-1.31×10^{-3}	-3.62×10^{-4}
$\beta_2 y(n-1)$	9.99×10^{-1}	9.97×10^{-1}	9.99×10^{-1}
$\beta_3 [u(n) + u(n-1)]$	-6.28×10^{-4}	-1.27×10^{-3}	-6.26×10^{-4}
$\beta_4 [u(n)^2 + u(n-1)^2]$	1.57×10^{-7}	2.91×10^{-8}	1.54×10^{-7}
$\beta_5 [u(n)^3 + u(n-1)^3]$	7.85×10^{-10}	-8.04×10^{-9}	5.87×10^{-10}
$\beta_6 e(n-1)$	-9.99×10^{-1}	-9.45×10^{-1}	-9.57×10^{-1}
$\delta(n)[\kappa_1]$	1.00	–	9.96×10^{-1}
$\kappa_2 \delta(n-k_2)[\kappa_2]$	1.00	–	9.93×10^{-1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\delta(n-k_{31})[\kappa_{31}]$	1.00	–	1.00×10^{-1}

Table 6.5: Theoretical and estimated discrete-time coefficients of VOR sub-system 1 (slow-phase). θ Column: Theoretical parameter values. $\hat{\theta}_{ELS}$ Column: Estimated parameters using ELS. $\hat{\theta}_{MELS}$ Column: Estimated parameters using MELS. Values correspond to continuous-time slow-phase parameter value in Tables 6.3 & 6.4.

using the theoretical relationships in Table 6.7. Since it is impossible to measure the signal at the output of the static nonlinearity, we consider the static nonlinearity to have unity gain and translate the overall gain onto the linear system. For this reason, the estimated gain ($K_{1,2}b$) is a product of the linear system and static nonlinearity gain. Note that it is possible to compute the continuous-time parameters for this nonlinear hybrid system only because we assume the system structure is fully known.

Table 6.8 shows the continuous-time parameter estimates for the slow and fast-phase sub-systems. In this table the first column contains the true parameter values, the second column contains the continuous-time estimates based on the ELS algorithm and the third column contains the estimates based on our MELS algorithm. Although some discrete-time parameters (e.g., β_2, ϑ_2) computed via the ELS algorithm “appear” close to their theoretical values (see Table 6.5 & 6.6), when they were used to estimate the continuous-time parameters the bias due to ELS became noticeable large. However, the continuous-time parameters computed based on our MELS algorithm are close to their true values. Hence, for this model of VOR our MELS algorithm performance was superior to the ELS algorithm and gave good estimates

Term	DT Coefficients of Sub-system 2		
	θ	$\hat{\theta}_{ELS}$	$\hat{\theta}_{MELS}$
$\vartheta_1 (DC)$	2.18×10^{-3}	1.59×10^{-2}	1.81×10^{-2}
$\vartheta_2 y(n-1)$	9.67×10^{-1}	9.21×10^{-1}	9.66×10^{-1}
$\vartheta_3 [u(n) + u(n-1)]$	4.37×10^{-3}	-2.90×10^{-1}	4.26×10^{-3}
$\vartheta_4 [u(n)^2 + u(n-1)^2]$	-1.09×10^{-6}	-7.08×10^{-3}	-1.36×10^{-6}
$\vartheta_5 [u(n)^3 + u(n-1)^3]$	-5.46×10^{-9}	1.36×10^{-3}	-1.70×10^{-10}
$\vartheta_6 e(n-1)$	-9.67×10^{-1}	-9.28×10^{-1}	-9.44×10^{-1}
$\delta(n)[\lambda_1]$	1.00	–	8.18×10^{-1}
$\delta(n-k_2)[\lambda_2]$	1.00	–	9.94×10^{-1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\delta(n-k_{30})[\lambda_{30}]$	1.00	–	9.94×10^{-1}

Table 6.6: Theoretical and estimated discrete-time coefficients of VOR sub-system 2 (fast-phase). θ Column: Theoretical parameter values. $\hat{\theta}_{ELS}$ Column: Estimated parameters using ELS. $\hat{\theta}_{MELS}$ Column: Estimated parameters using MELS. Values correspond to continuous-time slow-phase parameter value in Tables 6.3 & 6.4.

CT Coefficient	DT Relationship
$p_{1,2} =$	$\frac{-2T+2T(\beta_2, \vartheta_2)}{-(\beta_2, \vartheta_2)T^2 - T^2}$
$K_{1,2}b =$	$\frac{-4T(\beta_3, \vartheta_3)}{-(\beta_2, \vartheta_2)T^2 - T^2}$

Table 6.7: Discrete to continuous-time relationships for parameters $p_{1,2}$ and $K_{1,2}b$ of the VOR model.

of the underlying continuous-time parameters.

6.7 Experimental Data

Lastly, we assessed our MELS algorithm on experimental human VOR data collected in our laboratory, the Oculomotor Control Laboratory. The data analyzed for this study is from a single subject with history of vestibulo-ocular disease. The patient is known to have no function in one inner ear. This is associated with large nonlinearities in the VOR and a defective time constant (small). We expect our MELS to be particularly relevant under these conditions.

Term	CT Coefficients of Sub-system 1 (Slow-Phase)		
	True	Estimate Based on ELS	Estimate Based on MELS
p_1	0.0667	1.800	0.0600
K_1b	-0.7540	-1.526	-0.7504

Term	CT Coefficients of Sub-system 2 (Fast-Phase)		
	True	Estimate Based on ELS	Estimate Based on MELS
p_2	20.00	49.34	20.08
K_2b	5.331	-362.3	5.200

Table 6.8: Continuous-time parameter estimates of $p_{1,2}$ and $K_{1,2}b$ of the VOR model.

6.7.1 Procedures

Silver-silver chloride electrodes were used to record conjugate eye position in the horizontal plane, in the dark. The subject remained in dim red light for 20 minutes to adapt to the dark condition and minimize electrode drift during recordings. The subject was then seated on a servo-controlled rotating chair, restrained by seat belts and a head holder. The head and body were fixed en-bloc to the chair during rotations, while the subject was instructed to perform mental arithmetic during rotations in the dark.

6.7.2 Perturbation

The experimental protocol used a sinusoidal rotation at 1/6 Hz, with a peak head velocity ~ 200 deg/s. The test lasted 52s, of which the last 32s were recorded to measure VOR properties with sensory steady state. Full electro-oculogram (EOG) calibrations were performed before and after the rotation, to correct for any drift.

6.7.3 Apparatus

The chair was controlled by a Pentium computer, using software developed in house with Modula-2 (Jensen Partners International, Mountain View, California). Eye position and head (chair) position channels underwent analogue low-pass filtering (8-pole Bessel) to 40 Hz to avoid aliasing when sampled. Data were recorded on separate

channels of a 16-bit National Instruments A/D board, and stored at 500 Hz for later analysis. Signal processing was carried out off-line on a Pentium using software developed locally with Matlab (Mathworks, Natwick, MA).

6.7.4 Data Processing

The sampled signals were digitally low-pass filtered down to 15 Hz and then decimated to 250 Hz sampling rate, to improve the signal-to-noise ratio and to save storage space. Figure 6.6 shows a typical input-output trial used for this analysis. The

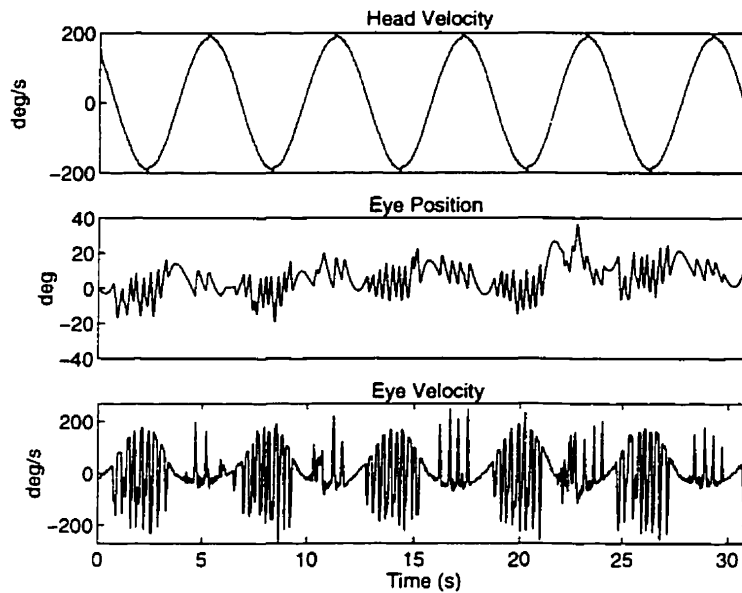


Figure 6.6: Experimental VOR data. Top: Head velocity input. Middle: Eye position output. Bottom: Eye velocity output.

data represents a sinusoidal head velocity of 200 deg/s. The characteristics of this trial are consistent with those reported in previous work done in our laboratory [58]. This 15 Hz bandwidth was sufficient to examine the slow-phase characteristics, in spite of mild distortions on the fast-phase trajectories. The position traces were digitally differentiated to obtain eye and head velocity trajectories, and scanned by our classification algorithm to demark slow phase segments automatically [123].

6.7.5 Data Analysis

The extracted slow-phase data was fitted with a linear and cubic description of VOR characteristics, as described in detail elsewhere [73]. This method required first removing any phase shift (dynamics) between the stimulus and response curves. Then VOR characteristics were modeled by the expressions in Equation 6.13, where “ y ” represents the slow-phase eye velocity, and “ u ” represents the phase-shifted head velocity in a particular sinusoidal protocol. Parameter estimates were obtained by regression with experimental data.

$$\text{linear fit: } y(n) = a + bu(n) \quad (6.13)$$

$$\text{cubic fit: } y(n) = a + bu(n) + cu^2(n) + du^3(n)$$

$$\text{model selection criteria: } BIC = \log(MSE) + \log(N) \frac{p}{2N}$$

$$\text{quality of fit: } \%NMSE = \left(1 - \frac{\frac{1}{N} \sum_{n=1}^N (y_n - \hat{y}_n)^2}{\frac{1}{N} \sum_{n=1}^N (y_n)^2} \right) \times 100$$

In Equation 6.13 the linear gain term (b) defines the VOR sensitivity for low-velocity rotations, the DC offset (or bias) is the zero-order coefficient (a), p is the number of model parameters, N is the number of data points in the pooled slow-phases and MSE is the mean square error of fit, $\frac{\sum_{n=1}^N [y_n - \hat{y}_n]^2}{N}$. Note that these monomial fits are not orthogonal and may be subject to error. In patient cases, a linear fit is often deficient in describing the data. Therefore, it is often necessary to fit the data with a cubic model [9, 58, 73, 114]. Justification for this selection is done on the basis of the associated normalized mean-squared-error and the Bayesian information criterion (BIC), to avoid over-modeling [58].

Our hybrid identification approach consisted of assuming a dual mode Hammerstein model structure: a third order static nonlinearity followed by a dynamic first order high-pass system (see Equation 6.3), was sufficient to describe the data. Next, our MELS algorithm was implemented to estimate model parameters. The quality of fit was assessed by computing the %NMSE as given in Equation 6.13.

6.7.6 Results

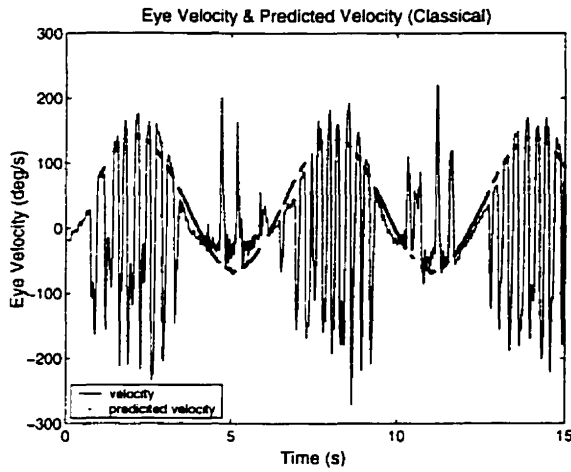
The results presented in this section are a comparison of data fit of the classical methods, described above, to the quality of data fit using our MELS algorithm.

Figure 6.7 show the results of this comparison. Plot (a) shows the linear fit, plot (b) shows the cubic fit, and plot (c) shows the fit obtained using our MELS algorithm to VOR data. The %NMSE obtained using the classic linear method is 91.35%, with the classic cubic fit it is 96.47% and using our method it is 96.62%. Table 6.9 shows the slow-phase time constant, τ_1 , and the nonlinearity coefficients (a, b, c, d) obtained using both the classical and the MELS approaches. Although the improvement in

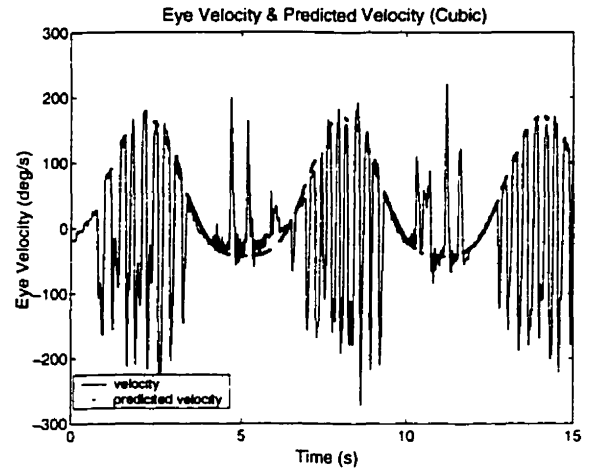
	a	b	c	d	τ_1 s	%NMSE
Classical	37.73	-0.542	–	–	4.99	91.93
Cubic	22.05	-0.594	1.20×10^{-3}	6.66×10^{-7}	5.48	96.47
MELS	20.59	-0.342	6.57×10^{-4}	2.97×10^{-7}	0.631	96.88

Table 6.9: Identified continuous-time parameters from experimental VOR data. a : DC term, b : linear term, c : squared term, d cubic term, τ_1 : slow-phase time constant and %NMSE percent normalized mean-squared-error.

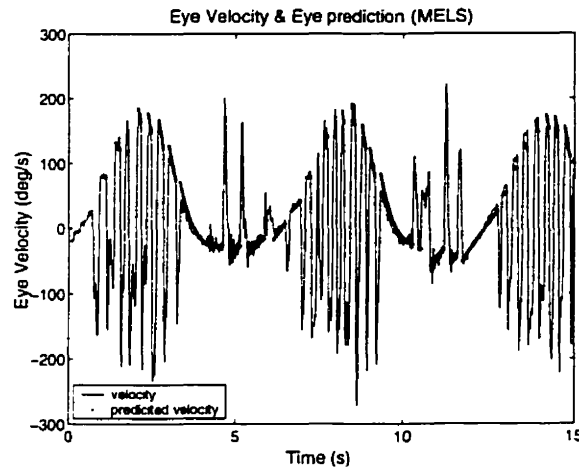
fit between the extended classical nonlinear method (Figure 6.7b) and our MELS algorithm (Figure 6.7c) is apparently small (see Table 6.9), it is misleading at first glance. In the extended classical approach (Figure 6.7b) the average fit is reasonable but the dynamics are clearly poorly described during negative eye speeds. Our MELS method describes the dynamics of individual slow-phases during leftward (negative) eye velocities but at the expense of poorer fits in the opposite direction (positive). This may indicate that the parameter estimates computed by MELS provide a better description of system dynamics at least during part of the cycle, implying input-dependent nonlinearities.



(a)



(b)



(c)

Figure 6.7: Predicted eye velocity of experimental VOR data. (a): Measured eye velocity superimposed on top of predicted output due to classical approach (i.e., linear fit). (b): Measured eye velocity superimposed on top of predicted output due to cubic approach (i.e., cubic fit). (c): Measured eye velocity superimposed on top of predicted output due to our hybrid identification (i.e., MELS technique).

6.8 Discussion

6.8.1 MELS Algorithm

The success of parameter estimation methods for nonlinear systems depends upon the choice of model structure and the development of estimation algorithms to yield unbiased estimates of system parameters. The limitation of our proposed algorithm for estimating parameters depends largely on the number of data segments and length of segments available for estimation. If the number of measurements per sub-system, say N , approaches infinity the statistics of the parameters will approach their asymptotic values. The limiting factors will depend not only the system dynamics but also on the feasibility of measuring the system for long periods of time.

6.8.2 Simulation Study

With respect to simulated data, an examination of Table 6.6 (VOR fast-phase) shows that the estimated parameters ($\hat{\theta}_{MELS}$) are not as “accurate” as those in Table 6.5 (VOR slow-phase) because the segment lengths were shorter. Hence, fewer data points were available to estimate the coefficients of sub-system 2 (VOR fast-phase). We computed the standard deviation (STD) for the estimated parameters of sub-systems 1 and 2 (VOR slow and fast). The STD of the coefficients for sub-system 1 were approximately $\sqrt{3}$ times smaller than those of sub-system 2 since there was three times more data available for estimation. However, the STD of system parameters for sub-system 2 were within the 95% confidence bound of the theoretical coefficients. The percent accuracy should be sensitive to the magnitude of c (i.e., coefficient of the squared nonlinear term). The parameter estimates should be less accurate for large c compared to segment length.

In the simulation (and experimental data) the input was a sinusoid. In general, a single sinusoid is a “terrible” input for identification. Therefore, it is tempting to conclude that the ELS results may simply be due to poor input design. The poor results are not due to poor input alone, though it certainly plays a role, but are due

to unmodeled dynamics in the residuals. As the time constant of a mode decreases, switching effects themselves can bias ELS, which MELS takes advantage of despite a poor forcing function. MELS gives good results because by explicitly correcting for impulses it effectively uses a high frequency input. The continuous-time parameters in Table 6.8 (ELS estimates) clearly demonstrate that even when some discrete-time parameters “appear” close to the theoretical values the continuous-time estimates may be far from the true values. This study demonstrates two important cases: (1) it shows when the time constant is large the percent improvement will be small and (2) in the opposite case when the time constant is small, where the bias from ELS can be quite severe (over-estimating the time constant), the percent improvement will be large. These cases illustrate MELS is the preferred tool. Although slow-phase time constants are expected to be large in the normal population, MELS is still a necessary tool in disease subjects since the time constants are often significantly reduced.

6.8.3 Experimental Data

Analysis of a typical experimental VOR data set using MELS showed a better fit for negative eye velocities but at the expense of poorer fits for the positive direction (see Figure 6.7c). This may be because more data was available for negative velocities or it may show that there are unique dynamics in each of the positive and negative directions for slow-phase. This suggests the system may contain a different static nonlinearities for each path and is not described well by a general cascaded system as we have forced here (i.e., Hammerstein structure). This clearly needs to be investigated with more patient data. Overall, experimental analysis of VOR data indicates that hybrid identification may be appropriate for this type of system since it provides an improved data fit, compared to classical linear technique and it gives a better estimate for negative velocity eye dynamics.

6.8.4 Future Work

Although this algorithm yielded good results for estimating the parameters of a multimode system, it is unclear how to determine the model order and structure for these types of systems (i.e., “black-box” identification). Furthermore, the current formulation is only valid for the identification of switched systems with first order dynamics since we have only discussed modeling initial conditions for such systems. The matrix extension (see Equation 6.9) of the MELS algorithm needs additional columns to model initial conditions of a general order dynamic system.

6.8.5 Summary of Findings

Modeling, simulation studies, and analysis of experimental data showed the following.

1. We have demonstrated that the NARMAX polynomial class can be used to model the dynamic behavior of nonlinear hybrid systems.
2. Simulation results showed that our MELS algorithm provides better results for parameter estimation of hybrid systems than applying existing methods which assume continuous smooth behavior.
3. Analysis of experimental data showed that MELS only provides an incremental improvement in data fit over traditional techniques. We hypothesize that this is due to an incorrect model structure used to fit the data (i.e., Hammerstein structure).

Implementation of our MELS algorithm to a simulated model of VOR demonstrated that it is possible to estimate the parameters of switched nonlinear systems. The MELS algorithm takes advantage of the switching effects despite a poor forcing function and gives good results because by explicitly correcting for impulses it effectively uses a high frequency input.

6.9 Conclusions

We have demonstrated that the NARMAX model structure is suited to modeling the dynamics of nonlinear multimode systems. Furthermore, the MELS algorithm is a robust method for estimating the coefficients of such multimode systems. This method provides accurate estimates of parameters since it takes advantage of the entire data record. We also provided an example of application of our hybrid modeling and identification approach on experimental VOR data. The results showed that our technique produced more accurate estimates of data prediction and system parameters than traditional approaches.

These results may have a clinical significance in the analysis of ocular nystagmus. The technique here allows greater insight into the functionality of various reflexes, by providing quantitative measures of both saccadic and slow ocular dynamics from a single experimental record.

Chapter 7

Conclusions

7.1 Introduction

In this thesis, we have developed practical methods for the identification of linear, nonlinear and hybrid (multimode) systems which are applicable under relatively general conditions, i.e when assumptions and conditions of the estimation technique are not violated. Since these algorithms were not designed specifically with any system in mind, they should be applicable to experiments on a variety of systems in many different disciplines.

In this chapter, we state the original contributions made during this thesis work, describe their significance, and give suggestions for further work.

7.2 Statement of Original Contributions

The overall goal of this work was not only for biomedical engineering but to provide an expanded and improved set of tools for the identification of *both* linear and nonlinear systems that fall under the *linear regression* “umbrella”. Results demonstrate that parametric nonlinear identification is a feasible tool for modeling unknown (black-box) systems. Some potential applications for these methods outside the biomedical realm are, for example, efficient control and design for aircraft/spacecraft, communications, analysis of economic trends, analysis of geophysical phenomena, etc. Below is a list

of the original contributions contained in this thesis.

1. **Application of NARMAX Structure to Biological Modeling.** We illustrated that the (polynomial) NARMAX model class is useful for modeling the input-output behavior of block-structured models encountered in biological control. These results suggest that other biological systems may be easily represented as a NARMAX class.
2. **Structure Detection.** We developed a robust algorithm based on bootstrap to compute the structure of linear and nonlinear systems (linear regression models). This method provides accurate estimates of parameter statistics without relying on assumptions made by traditional procedures and yields a parsimonious system description. Convergence results provide an empirical measure for data requirements necessary to achieve a true model structure. The significance of this finding is that it enhances existing methods for structure detection by providing a method for determining structure of highly over-parameterized models.
3. **Model Order Selection.** We provide a robust technique to compute the order of NARMAX models. This algorithm computes a unique minima over a selected dimension in $O = [n_u, n_y, l]$ which provides an unambiguous estimate of model order using the cross-validation cost function. This work contributes to existing methods for model order estimation by providing an algorithm for determining the order of nonlinear systems that are linear-in-the-parameters.
4. **Parameter Estimation of Hybrid Systems.** We demonstrated that the NARMAX model structure is well suited for modeling the dynamics of nonlinear hybrid systems. In addition, we developed a robust MELS algorithm to estimate coefficients of multimode systems. This work contributes to (1) basic understanding of hybrid systems modeling and (2) fundamental algorithmic development for linear and nonlinear hybrid systems by providing a parameter estimation technique for these types of systems.

7.3 Suggestions for Further Research

Much work needs to be done in terms of theoretical analysis, algorithmic development and applications of these techniques to real-life applications. Since the emphasis of this thesis was on algorithmic development, our suggestions for future work are focused on this topic.

7.3.1 Identification of Ankle Dynamics

In Chapter 3 we illustrated the effect(s) of input and output additive noise on a NARMAX representation of ankle dynamics. With the current experimental setup and protocol used in our laboratory, we expect the input to have a SNR approximately in the range of 82 dB – 50 dB (see §3.3.5). With little noise added to the input and noise-free output, the standard deviation of linear path parameters included zero for SNR levels ≤ 55 dB SNR (see §3.3.5). In addition, with a fixed input SNR of 60 dB and noise added to the output, the standard deviation of linear path parameters included zero for output SNR ≤ 20 dB (see §3.3.6). These results showed that, for the given model structure, even if an input is recorded with insignificant noise the least-squares algorithm will not yield an unbiased estimate of model parameters, as it violates the basic assumptions.

Identification of NARMAX models using “standard” least-squares is particularly sensitive to input noise for high-pass systems. However, it is more robust in the low-pass functional form. To address the problem(s) with input noise sensitivity in the high-pass case (i.e., linear path of ankle dynamics) two approaches may be considered. (1) If NARMAX is to be used in the current state for ankle dynamics, then redesign the experimental paradigm. (2) Otherwise, the optimal solution may be to generalize the approach with implementation of total least-squares.

7.3.1.1 Redesign of Experimental Paradigm

Currently, the intended input (noise free input) is not saved for further analysis. Instead, the input used for identification is recorded after being influenced by actuator

dynamics and measurement noise. We recommend that the intended input be saved for identification and an effort be made to model actuator dynamics. This should alleviate the need to record input after being passed through actuator dynamics: hence, reduce the introduction of input noise.

For this paradigm an instrumental variable approach may seem appropriate. However, the IV algorithm only addresses bias, i.e., lagged errors, due to output additive noise (see §2.6.3.4). Extended least-squares solves this bias problem by explicitly modeling lagged error terms. ELS was implemented for the study in Chapter 3 and shown not to provide “good” estimates in the presence of input and output additive noise. This implies that IV is unlikely to provide “better” results since it only addresses bias due to output additive noise and does not account for bias due to input noise.

Nonlinear feedback is a feature of many biomedical systems. The parallel pathway model describing ankle dynamics (see Figure 3.1) can be formulated in closed-loop as shown in Figure 3.19 [78]. For the parallel pathway description, intrinsic components are modeled using derivative operators in the linear path. These derivatives result in an unrealizable model description which also serve as a source of input noise amplification. Formulating this model in nonlinear closed-loop reduces the number of derivative operators (see Figure 3.19) needed to describe the same model and decreases the potential danger of input noise amplification (sensitivity). Note that when this model is posed in feedback, the linear dynamics are modeled as a low-pass system, i.e., compliance dynamics. This model formulation serves to filter high frequency content associated with input, possibly providing a more robust identification in the presence of input noise, without resorting to advanced techniques such as total least-squares. However, this feature comes at the cost of a more complex model description.

7.3.1.2 Total least-squares

In the standard linear model ($Z = \Psi\theta + e$) it is usually assumed that the exact structure of the regressor matrix is known, e is a vector of random errors which are

uncorrelated and have zero means and the same variance, i.e., $E(e) = 0$, $D(e) = \sigma^2 I$. As demonstrated by analysis of experimental ankle data, these assumptions are frequently unrealistic since sampling and/or modeling errors often affect the input, and hence, the regressor matrix Ψ . Therefore, it is necessary to consider methods that also allow for random input errors, such as total least-squares [26, 69]. The premise of total least-squares is that allowing for input as well as output error in the standard linear model above (i.e., modeling input as well as output noise), provides better (hopefully unbiased) parameter estimates.

Application of total least-squares may give insight to determine if parameter estimates for this biological data resemble the currently believed ranges and if the ranges are accurate. Few applications of total least-squares have been published in the literature [43]. Van Huffel and Vandewalle claim that in typical applications (linear systems), gains of 10–15% in accuracy can be obtained by using total least-squares instead of standard least-squares methods [69]. To date there are no known applications of this technique to nonlinear systems likely due to the number of terms needed in the regressor matrix, requiring considerable computational expense. Future work should include a careful development of existence theory for the general NARMAX model, allowing for input additive noise.

7.3.2 Structure Detection

We showed in Chapter 4 that when white assumptions were violated, our BSD technique failed to compute the correct structure. The limiting factor is with the parameter estimation algorithm implemented in our structure detection routine, i.e., ELS, since it requires white, zero-mean conditions. In general, bootstrap does not rely on white assumptions [48, 136]. Therefore, it may be possible to use robust estimation techniques in combination with our BSD method to yield better results under non-white assumptions [35, 122, 147].

7.3.3 Model Order Selection

For the BMOS algorithm we did not study the effect(s) of bandlimited (colored) output additive noise. The properties of this algorithm need to be studied under these conditions to provide a better understanding of its behavior. In addition, we did not compare the performance of our BMOS algorithm against any popular techniques such as AIC or MDL. Future work should include a study of this algorithm's ability to select the correct model order as compared to these classic approaches.

7.3.4 Combined Structure Detection – Model Order Selection

For model order selection the BMOS estimate may not yield the “true” order for all structures since the full model is posed at each step of the search. Consider the model:

$$y(n) = \theta_1 u^3(n-2) + \theta_2 y(n-7). \quad (7.1)$$

A model of this order, i.e., high nonlinear and dynamic order, has many candidate terms. However, the “true” system is described by only two parameters. Systems described by such a high system order may lead to inaccurate estimates of model order. This is because the number of candidate terms grows rapidly as the nonlinear or dynamic order is increased; possibly resulting in highly biased estimates of model errors during the model order search. To overcome the effects of this problem it may be better to compute order and structure simultaneously, i.e., estimate structure at each step of the model order search.

Model parameters and residuals are computed for both the model order selection and structure detection procedures. Instead of discarding information about the parameters and residuals at each step of the model order selection process, this knowledge can be utilized to (1) determine structure and (2) compute the error in prediction. This approach will significantly reduce the time and computational expense required for parametric identification of nonlinear systems.

7.3.5 Hybrid Systems

In Chapter 6 we presented a parameter estimation algorithm for hybrid systems (MELS). The current formulation is only valid for the identification of switched systems with l th order nonlinearity and first order dynamics. Future extensions should include general dynamics for systems of greater complexity. Moreover, these results may be generalized to multiple-input multiple-output (MIMO) nonlinear hybrid systems.

Although our MELS algorithm is a good start for the identification of nonlinear multimode systems much basic work still needs to be done. To perform “black-box” identification of nonlinear hybrid systems it is necessary to develop algorithms to compute model order and structure. This can be done in a similar manner as that employed for single mode systems (see Chapters 4 & 5) but with the extended least-squares algorithm replaced by our modified extended least-squares algorithm.

7.3.6 Application to Real Data

Much can be learned from simulations alone. However, many problems encountered in real situations cannot be duplicated. Some examples are those associated with the finite resolution of A/D converters, and with the finite roll-off of anti-aliasing and reconstruction filters, both of which limit input and output signal bandwidth. In order to demonstrate that these techniques are applicable in real engineering situations, they need to be verified using real data. As a first step, feasibility of these methods to other engineering applications can be established by building several nonlinear circuits, this includes:

1. Analog second order low-pass IIR system preceded by squared nonlinearity
2. Analog second order high-pass IIR system preceded by squared nonlinearity
3. Analog second order bandpass IIR system preceded by squared nonlinearity

The low-pass circuit will establish how these techniques behave with high frequency output noise, the high-pass system will provide insight into the robustness of these

methods in the presence of low frequency output noise and the bandpass model will yield information with the combination of the two.

To justify the selection of these structures, consider the nonlinear-linear (NL) model, a low-pass system preceded by a static nonlinearity, shown in Figure 7.1. This

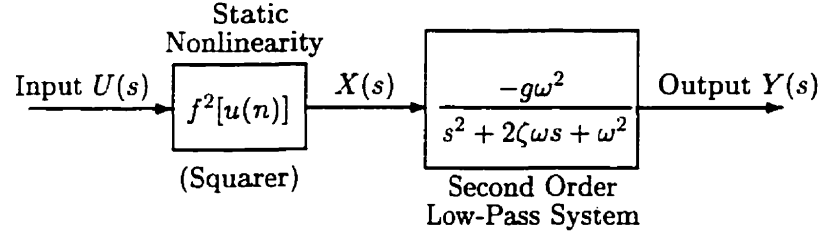


Figure 7.1: Low-pass IIR system preceded by a static nonlinearity (squarer).

model arrangement is described as:

$$y(n) = a_1 y(n-1) + a_2 y(n-2) + b_0 u^2(n) + b_1 u^2(n-1) + b_2 u^2(n-2). \quad (7.2)$$

The NL description of Figure 7.1 yields a NARMAX model (Equation 7.2) with non-linear, current and delayed inputs and linear, delayed outputs. This NL configuration does not result in a NARMAX description with a large number of terms. Therefore, it should be difficult to identify because model errors will be highly biased due to over-parameterization. Moreover, the nonlinearity and system orders should initially be selected as two in an attempt to keep the number of candidate terms manageable, i.e., regressor matrix dimension.

7.4 Discussion

Although our bootstrap approach to the structure detection and model order selection problems are computationally expensive, it may be the only means to solve such complex problems. Many techniques available to researchers in the areas of nonlinear identification and signal processing are purely heuristic. These techniques demand practically full knowledge of the system before identifying it. This naturally poses the question: “*Why identify the system at all if we need such extensive knowledge*

of the system before commencing the identification process?”. Our BSD and BMOS techniques do not require many assumptions and/or extensive *a priori* knowledge of the system. We only make the standard assumptions needed to satisfy conditions in least-squares analysis. We believe that as the power and usefulness of bootstrap is realized by developers and practitioners and as computers become cheaper and more powerful, the bootstrap will quickly become a standard tool in many disciplines. In fact, bootstrap is already an emerging tool in both system identification and signal processing [117, 148, 149, 150, 151, 168].

Theoretical analysis of hybrid modeling and identification of nonlinear MIMO systems may lead to some useful applications. However, to date there are few known “naturally” occurring switched mode systems and hence the practical application or usefulness of such analysis remains speculative. Nevertheless, hybrid control and identification are rapidly developing fields and have recently been gaining a wider appreciation from the controls community [11, 50, 97]. Clearly, this is an emerging field in which much fundamental work remains to be done in terms of analysis, algorithmic development and applications.

As a final remark, we note that only a few simple nonlinear systems were studied to validate our methods. Nevertheless, the systems studied in this thesis are general enough to provide insight into the behavior of our algorithms when applied to more complex systems (see §4.9 & §5.6). In addition, the simplicity of the systems studied was due to certain limitations regarding computing power and accessibility to a wider variety of data from various disciplines. Our algorithms and modeling techniques should be applicable to many systems that fall in the relevant class of models, i.e., linear and nonlinear systems that are linear-in-the-parameters. Furthermore, despite the lack of natural systems known to exist in a hybrid state, we believe that many biological processes may exhibit hybrid behavior or may be postulated as a class of hybrid systems. Study of such biological systems could (1) inspire alternate control strategies in engineering and robotics and (2) lead to the development of better tools for understanding biological control and automating diagnosis.

Appendix A

A Matlab Toolbox for Nonlinear System Identification

A.1 Introduction

This appendix contains a list of the tools developed for the analysis of parametric nonlinear systems (NARMAX models) and gives a brief description of each tool. All routines contained in this appendix were written for the Matlab simulation and development platform.

We first give examples detailing the syntax for using some of our major routines.

A.2 NARMAX Model Simulation

Consider the NARMAX model:

$$y(n) = 0.4u(n-1) + 0.4u^2(n-1) + 0.8y(n-1) - 0.8e(n-1) + e(n). \quad (\text{A.1})$$

Model A.1 is used through out this appendix to illustrate the usage of our Matlab routines.

To simulate this model the procedure is the following.

1. Create a uniform stimulus with a rectangular distribution, for example, as:

```
TMAX=10000;  
STEP_SIZE=1;  
t=(0:STEP_SIZE:TMAX)';  
rand('seed',sum(100*clock));  
u=rand(size(t));  
u=u-mean(u);.
```

2. Next, form a noise sequence with Gaussian distribution for the error terms (e.g.,

```
 $e(n-1)$ ) as  
randn('seed',sum(100*clock));  
noise=randn(size(u));  
noise=noise-mean(noise);.
```

3. To simulate NARMAX model A.1 each term of the model is placed into a cell array as a character string. The syntax is the following:

```
model{1}=['0.4u(n-1)'];  
model{2,:}=['0.4u^2(n-1)'];  
model{3,:}=['0.8y(n-1)'];  
model{4,:}=['-0.8e(n-1)'];  
model{5,:}=['e(n)'];.
```

4. The function used to simulate a NARMAX model is called *simnarmax.m* and it is utilized with the following function call

```
[y]=simnarmax(model,u,noise);.
```

The function inputs are:

“model” – model to be simulated (cell array with appropriate structure),

“u” – input, and

“noise” – noise process.

The function output is:

“y” – simulated output of Model A.1.

In Matlab, to obtain additional help for function *simnarmax.m* type “help simnarmax” or type “simnarmaxdemo” for an on-line demo of this function.

A.3 NARMAX Model Order Selection

Assume that only input-output data is given for Model A.1. Note that for some systems an input or output set alone may be sufficient for order selection.

An estimate of model order can be computed using our bootstrap model order selection routine (see Chapter 5). This routine name is *bmos.m* and its function call is

[RSS,order]=bmos(u,y,N,n,nu,ny,B);

The function inputs are:

“u”, “y” – as defined above,

“N” – number of data points to be used for estimation, e.g., N=9000,

“n” – maximum nonlinearity order, e.g., n=3,

“nu” – maximum input lag order, e.g., nu=2,

“ny” – maximum output lag order, e.g., ny=2

“B” – number to bootstrap replications, e.g., B=100.

The function outputs are:

“RSS” – multidimensional array of bootstrap estimates of the error in prediction.

The first index references array dimension 1, the row. The second index references dimension 2, the column. The third index references dimension 3, the page.

“order” - order=[nu ny l]: estimated order of a NARMAX model, where “nu”, “ny” and “l” are defined above. Note that since we assume output additive noise, ne=ny, i.e., the error lag must equal the output lag order (see Chapter 5).

The order selection procedure starts at a minimum dimension of order=[0 1 1]. This function requires a user to provide integer values for nu, ny and l, defining a maximum search dimension for model order selection. If the maximum search

dimension is large the number of points in the search space will also be very large and require a large number of computations to estimate model order. Moreover, if B is also large the time required to compute an estimate of model order will increase B -fold.

We recommend that this routine only be utilized when the system is known or likely to be nonlinear and basic information about the system under test is not sufficient to build a regressor matrix with the proper dimension, i.e., the system order is unknown. Hence, if the system is linear it may be computationally cheaper to use existing methods within Matlab such as AIC or MDL.

A.4 NARMAX Structure Detection

Assume that input-output data is provided for Model A.1 and the model order is known or given. To compute structure for this model we use our bootstrap structure detection routine (see Chapter 4). The syntax for this function is:

```
[thetaf,vlabel]=bsd(u,y,N,n,nx,ny,iter,B,alpha1,alpha2);
```

The function inputs are:

“u”, “y”, “N”, “n”, “nx”, “ny”, “iter” and “B” – same as defined previously,

“alpha1” – upper confidence bound, e.g., $\alpha_1=0.95$, and

“alpha2” – lower confidence bound, e.g., $\alpha_2=0.05$.

The function outputs are:

“thetaf” – reduced parameter vector (containing only significant terms) and

“vlabel” – parameter labels correspond to the regressor associated with each parameter in thetfa.

The upper and lower confidence bounds, shown above, are typical values used in practice. However, they can be varied depending on the intended application. This function is typically used after the *bmos* routine has provided an estimate of model order or if the system order is known *a priori*. For an on-line demo of this function type “bsddemo”.

A.5 NARMAX Parameter Estimation

Assume that input-output data is given for Model A.1 and the model order and structure are known. In some special cases, input or output alone may be sufficient for parameter estimation. Let the input-output set be given by the simulated model above.

Since this model is described by lagged outputs and lagged error terms we select the extended least-squares algorithm to compute an unbiased estimate of model parameters (see Chapter 2). The function call for our extended least-squares parameter estimation algorithm is:

```
[theta,vlabel,lenu,err,z_hat,PHI]=els(u,y,N,n,nx,ny,iter);
```

The function inputs are:

“u”, “y” – as defined previously,

“N” – number of data points used for estimation, e.g., N=5000,

“n” – nonlinearity order, e.g., n=3,

“nx” – input lag order, e.g., nx=1,

“ny” – output lag order, e.g., ny=1, and “iter” – number of iterations for improving the noise model, e.g., iter=10.

The function outputs are:

“theta” – estimated parameters,

“vlabel” – parameter labels,

“lenu” – number of columns that are dependent only on input,

“err” – residuals or prediction errors,

“z_hat” – predicted output, and

“PHI” – regressor matrix.

Note that “vlabel” contains the corresponding row labels for “theta”, i.e., the regressor associated with each parameter.

If the system is described only by lagged input terms it is more efficient to use ordinary least-squares (ols.m). The syntax for “ols.m” is similar to that of “els.m”. Type “help ols” for more information. For help regarding how to form a regressor

matrix or regressors associated with each column of the matrix type “help rmt2” or “help name2”, respectively. For an on-line demo of this function type “parmdemo”.

A.6 M-files

The first lines of each function describe the purpose of the m-file, as well as the inputs it requires and the outputs it generates. These comments can be accessed within Matlab by typing help and the m-file name.

ank2mat

```
function [Vector]=ank2mat(N,x,y,delay);
% Vector=ank2mat(N,x,y,delay);
%
% This function computes the regressor matrix for
% a 2nd order model of ankle dynamics.
%
% This assumes the general structure of Rob's model
% but uses a 2nd order approximation for the static
% nonlinearity.
%
% N=      number of data points to be used
% x=      measured input
% y=      measured output
% delay=  reflex delay in discrete-time,
%         i.e., DT_delay=CT_delay/sampling_rate
%
% Vector= regressor matrix that contain the correct
%         terms for Rob's model
%
%      Sunil L. Kukreja  8 June 2000
%      Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

ank2mate

```
function [Vector]=ank2mate(N,err);
% Vector=ank2mate(N,err);
%
% This function computes the error regressor matrix
% for a 2nd order model of ankle dynamics.
%
% This assumes the general structure of Rob's model
% but uses a 4th order approximation for the ststic
% nonlinearity.
%
% N=   number of data points to be used
% err= residuals
%
% Vector= regressor matrix that contain the correct
%         terms for Rob's model
%
%   Sunil L. Kukreja  8 June 2000
%   Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

bmos

```
function [RSS,order]=bmos(u,y,N,n,nx,ny,B);
% Bootstrap Model Order Selection
%
% [RSS,order]=bmos(u,y,N,n,nx,ny,B);
%
% u - input
% y - output
% N - number of data points to be used
% n - maximum nonlinerity order
% nx - maximum input lag order
% ny - maximum ouput lag order
% B - number to bootstrap replications
%
% RSS - multidimensional array of bootstrap
%       estimates of the error in prediction.
%       First references array dimension 1, the row
%       Second references dimension 2, the column
%       Third references dimension 3, the page
%
% order - order=[nu ny l]: estimated order of NARMAX
%          model where nu: lag order of input, ny: lag
%          order of output, l: nonlinearity order
%
% NOTE: We assume output additive noise.
%       Therefore, ne=ny.
%       ne: lag order of error
%
% Details can be found in:
% A BOOTSTRAP METHOD FOR NARMAX MODEL ORDER SELECTION
% S.L. Kukreja, R.E. Kearney and H.L. Galiana,
% IFAC-MCBS 2000
%
% Copyright Sunil L. Kukreja 29 February 2000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

bsd

```
function [thetaf,vlabel]=bsd(u,y,N,n,nx,ny,iter,B,alpha1,alpha2);
% Bootstrap Percential Method for Structure Detection
%
% [thetaf,vlabel]=bsd(u,y,N,n,nx,ny,iter,B,alpha1,alpha2);
%
% INPUTS:
% u - input
% y - output
% N - number of data points to be used
% n - nonlinerity order
% nx - input lag order
% ny - ouput lag order
% iter- number of iterations for improving noise model
% B - number to bootstrap replications
% alpha1 - upper confidence bound
% alpha2 - lower confidence bound
%
% thetaf - reduced theta
% vlabel - parameter labels
%
%
% Details can be found in:
% Structure Detection of Nonlinear Dynamic Systems Using
% Bootstrap Methods
% S.L. Kukreja, R.E. Kearney and H.L. Galiana, IEEE-EMBS98,
% October 1998
%
% Structure Detection of NARMAX Models Using Bootstrap Methods
% S.L. Kukreja, H.L. Galiana and R.E. Kearney IEEE-CDC99,
% December 1999
%
% Copyright Sunil L. Kukreja 7 April 1998
% (updated 15 Jan 99)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

els

```
function [theta,vlabel,lenu,err,z_hat,PHI]=els(u,y,N,n,nx,ny,iter);
% Extended least squares
%
% [theta,vlabel,lenu,err,z_hat,PHI]=els(u,y,N,n,nx,ny,iter);
%
% INPUTS:
% u - input
% y - output
% N - number of data points to be used
% n - nonlinearity order
% nx - input lag order
% ny - ouput lag order
% iter- number of iterations for improving noise model
%
% OUTPUTS:
% theta - estimated parameters
% vlabel - parameter labels
% lenu- number of columns that are purely due to input
% err - residuals
% z_hat - predicted output
% PHI - regressor matrix
%
%
% Method based on ELS algorithm in: Dynamic System
% Identification: Experiment Design and Data Analysis
% G.C. Goodwin and R.L. Payne, 1977
%
% Copyright Sunil L. Kukreja 16 November 1999
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

lankid

```
function [theta,z_hat]=lankid(x,z)
% Computes parameters of linear path of parallel pathway
% ankle model
%
%           [theta,z_hat]=lankid(x,z)
%
% This function is meant to be used for iterative
% identification of ankle dynamics.
%
% Inputs:
%       x- input
%       z- output
%
% Outputs:
%       theta- estimated parameters
%       z_hat- predicted out
%
% Copyright Sunil L. Kukreja 9 October 2000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

ls_std

```
function [std_theta] =ls_std(theta,z,z_hat,RZ);
% Standard deviation of parameters using least-squares
% methods.
%
% [std_theta] =ls_std(theta,z,z_hat,R);
%
%
% Inputs:
%
% theta - estimated parameter vector
%      z - measured output
% z_hat - predicted output
%      R - the R matrix in qr factorization of the
%          regressor matrix
%
% Output:
%
% std_theta - standard deviation of parameters
%
%
% Method based on STD computation in: Linear Regression
% Analysis, George A.F. Seber 1977
%
%
%      Sunil L. Kukreja  12 May 1998
%      Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

mgs

```
function [Q,R]=mgs(X);
% QR factorization using the Modified Gram-Schmidt (MGS)
% Algorithm
%
% Given  $X \in \mathbb{R}^{m \times n}$  with  $\text{rank}(X)=n \rightarrow X=Q \cdot R$  where  $Q \in \mathbb{R}^{m \times n}$ 
% has orthogonal columns and  $R \in \mathbb{R}^{m \times m}$  is upper triangular.
%
% Orthogonal-triangular decomposition.
%  $[Q,R] = \text{mgs}(X)$  produces an upper triangular matrix  $R$  of
% the same dimension as  $X$  and a unitary matrix  $Q$  so that
%  $X = Q \cdot R$ .
%
% Method based on MGS algorithm in: Matrix Computations
% Gene H. Golub and Charles F. Van Loan
% 3rd Ed., pp. 232
% The Johns Hopkins University Press, 1996.
%
% also see
%
% Linear Algebra with Applications
% Steven J. Leon, 3rd Ed.,
% pp. 240,
% Macmillan Publishing Co., 1990.
%
% Also see rmmgs.m
%
% Sunil L. Kukreja 08 December 1998
% Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

mod_val

```
function [R_ee,R_ue,R_eeu]=mod_val(x,err,lag);
% [R_ee,R_ue,R_eeu]=mod_val(x,err,lag);
%
% input= the input to the system provided by the user
% residuals= residuals computed from identification
% lag= number of lags correlation is computed to
%
%
% R_ee =auto-correlation of errors
% R_ue =cross-correlation of input & errors
% R_eeu =cross-cross-correlation of errors & error
%
% Function implements method in: Structure detection
% and model validity tests in the identification of
% nonlinear systems
% S.A. Billings and W.F.S. Voon
% IEE Proceedings
% Vol. 130, Pt. D, No. 4, July 1983
%
% SLK 16 November 1997
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

name2

```
function [vlabel,lenu]=name2(n,nx,ny,ne);
% Forms the row names for the prameter vector in
% linear regression.
%
% [vlabel,lenu]=name2(n,nx,ny,ne);
%
%
% This function computes the names of all the parameters
% in the parameter vector.
%
% n= degree of polynimial
% nx= number of lagged inputs
% ny= number of lagged outputs
% ne= number of lagged errors
%
% vlabel - contains the names of all the columns of the
%          regressor matrix
% lenu    - the number of columns depending on only the
%          input and DC term
%
%          Sunil L. Kukreja 12 May 1998 (revised 2 December 1998)
%          Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

nlankid

```
function [theta,z_hat]=nlankid(x,z,delay)
% Computes parameters of nonlinear path of parallel pathway
% ankle model
%
%           [theta,z_hat]=nlankid(x,z,delay)
%
% This function is meant to be used for iterative identification
% of ankle dynamics.
%
% Inputs:
%       x- input
%       z- output
%       delay- discrete-time delay of reflex path
%
% Outputs:
%       theta- estimated parameters
%       z_hat- predicted out
%
% Copyright Sunil L. Kukreja 12 October2000
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

nmse

```
function v = nmse(err,y)
% Computes the Normalized Mean Square Error between
% two signals
%
% v = nmse(err,y)
%
% err=y-y_hat
%
% err are the residuals
% y is the measured signal
%
%
%       Sunil L. Kukreja  25 April 1997
%       Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

ols

```
function [theta,vlabel,PHI,Q,R,err]=ols(u,y,N,n,nx,ny);
% Ordinary least squares
%
% [theta,vlabel,PHI,Q,R,err]=ols(u,y,N,n,nx,ny);
%
% INPUTS:
% u - input
% y - output
% N - number of data points to be used
% n - nonlinearity order
% nx - input lag order
% ny - output lag order
%
% OUTPUTS:
% theta - estimated parameters
% vlabel - parameter labels
% PHI - regressor matrix
% Q - orthogonal decomposition of PHI
% R - triangular decomposition of PHI
% err - residuals
%
% Method based on OLS method in: Linear Regression Analysis
% George A.F. Seber 1977
%
% also see
%
% Dynamic System % Identification: Experiment Design and
% Data Analysis
% G.C. Goodwin and R.L. Payne, 1977
%
%
% Copyright Sunil L. Kukreja 16 November 1999
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

rank2mat

```
function [Vector]=rank2mat(N,x,y,delay);
% Vector=rank2mat(N,x,y,delay);
%
% This function computes the regressor matrix
% for a 2nd order model of ankle dynamics.
%
% This assumes the general structure of Rob's
% model but uses a 2nd order approximation for
% the static nonlinearity.
%
% N=      number of data points to be used
% x=      measured input
% y=      measured output
% delay=  reflex delay in discrete-time,
% i.e., DT_delay=CT_delay/sampling_rate
%
% Vector= regressor matrix that contain a
%          compressed version of the correct
%          terms for Rob's model
%
%
%          Sunil L. Kukreja  8 June 2000
%          Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

rank2mati

```
function [Vector]=rank2mati(N,x);
% Vector=rank2mati(N,x);
%
% This function computes the intrinsic regressor matrix
% for a 2nd order model of ankle dynamics.
%
% This assumes the general structure of Rob's model
% for the intrinsic path
%
% Inputs:
% N=      number of data points to be used
% x=      measured input
%
% Output:
% Vector= regressor matrix that contain a compressed version
%         of the correct terms for Rob's model
%
%
%         Sunil L. Kukreja  11 September 2000
%         Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

rank2matr

```
function [Vector]=rank2matr(N,x,y,delay);
% Vector=rank2matr(N,x,y,delay);
%
% This function computes the reflex regressor matrix
% for a 2nd order model of ankle dynamics.
%
% This assumes the general structure of Rob's model
% for the reflex path but uses a 2nd order approximation
% for the static nonlinearity.
%
% N=      number of data points to be used
% x=      measured input
% y=      measured output
% delay=  reflex delay in discrete-time,
%         i.e., DT_delay=CT_delay/sampling_rate
%
% Vector= regressor matrix that contain a compressed
%         version of the correct terms for Rob's model
%
%
%         Sunil L. Kukreja  11 September 2000
%         Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

rmmgs

```

function [Q,R]=rmmgs(Q,R,A,c);
% QR factorization using the remodified Modified
% Gram-Schmidt (MGS) Algorithm
%
% Given the QR factorization of A_old, and the new
% updated matrix A_new
%
% [Q,R]=modreg(Q,R,A_new,c) computes the fast
% orthogonal-triangular decomposition of A_new
%
% This can be used when the entire matrix, A_new,
% does not need to be re-orthogonalized. The
% "updated or new" columns of A_new are orthogonalized
% relative to the previously orthogonalized and unchanged
% columns of A_old.
%
% The matrix A is assumed to be in two partitions. The
% first, A', does not change while the second partition,
% A'', is updated or new. A=[A' A'']
%
% Q,R - Orthogonal-triangular decomposition of A_old
% A    - A_new    mxn matrix m >= n
% c    - number of columns of A_old that do not change
%
% Q    - an m x n unitary matrix Q so that A_new = Q*R
% R    - an n x n upper triangular matrix
%
% Method based on MGS algorithm in: Matrix Computations
% Gene H. Golub and Charles F. Van Loan
% 3rd Ed., pp. 231-2
% The Johns Hopkins University Press, 1996.
% ALSO SEE
% Linear Algebra with Applications
% Steven J. Leon, 3rd Ed., pp. 240,
% Macmillan Publishing Co., 1990.
%
% Also see mgs.m
%
%      Sunil L. Kukreja  09 December 1998
%      Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

rmt2

```
function Vector=rmt2(N,n,nx,ny,ne,x,y,e);
% Forms a regressor matrix with specified dimensions.
%
%   Vector=rmt(N,n,nx,ny,ne,x,y,e);
%
%   This function computes the regressor matrix (PHI) in
%   a linear regression problem; i.e.  $Y=PHI*THETA$ 
%
%   Y= the output from the system provided by the user
%   PHI= the regressor matrix
%
%   N=  how many rows wanted in the regressor matrix
%   nx= number of lagged inputs
%   ny= number of lagged outputs
%   ne= number of lagged errors
%   n=  degree of polynomial
%
%   The regressor matrix is setup in the following format:
%
%   Vector=[Gu Gzu Gzue Ge] where
%
%   G_zu contains the order in which terms containing
%   z and u terms appear in the first partition of the
%   regressor matrix.
%
%   G_zue contains the order in which terms containing
%   z, u and e appear in the second partition of the
%   regressor matrix.
%
%   G_e contains the order in which terms containing
%   e only appear in the third partition of the regressor
%   matrix.
%
%   Also see name.m
%
%   Sunil L. Kukreja  12 May 1998  (revised 2 December 1998)
%   Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

simnarmax

```
function [output]=simnarmax(model,input,noise);
% Simulate a polynomial NARMAX model
%
% Function: [output]=simnarmax(model,input,noise);
%
% model - polynomial NARMAX model
% The NARMAX model equation must have all coeffs in
% front of the variables without multiplication sign;
% e.g. 0.5u(n-1).
%
% Crossterms are written with a multiplication sign
% between the variables and the coefficient in front;
% e.g., 0.5u(n-1)*y(t-3)
% Do not put a plus in front of leading terms;
% DO NOT DO THIS: e.g. model{2,:}=['+3u^2 (n-3)']
%
% 'u' specifies the input
% 'y' specifies the output
% 'e' specifies the noise
% 'n' specifies the discrete time step
%
% Function Inputs:
% input-system input
% output-system output
% noise-system noise
% model- polynomial NARMAX equation
%
% To simulate without an input declare input=[];
% To simulate without output additive noise declare noise=[];
% e.g.,  $y(n)=5.1u(n-1)+3u^2(n-3)-0.4y(n-4)+0.4y(n-4)*e(n-1)$ 
%  $-0.2e(n-1)+e(n)$  would be specified as follows:
%
% model{1}=['5.1u^2(n-1)'];
% model{2,:}=['3u^2(n-3)'];
% model{3,:}=['-0.4y(n-4)'];
% model{4,:}=['0.4y(n-4)*e(n-1)'];
% model{5,:}=['-0.2e(n-1)'];
% model{6,:}=['e(n)'];
%
% Copyright Sunil L. Kukreja 24 July 1998
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

SWR

```
function [thetan,vlabel]=swr(PHI,y,vlabel);
% Stepwise Regression Algorithm
%
% [theta,vlabel]=swr(PHI,y,vlabel)
%
%     PHI - Regressor matrix
%     y - output
%     vlabel - name vector
%
%     theta - reduced theta
%
% Method based on SWR in: A prediction-error and
% stepwise-regression estimation algorithm for
% nonlinear systems
% S.A. Billings and W.S.F. Voon
% Int. J. Control, vol. 44 No. 3 pp. 803-822 1986
%
% also see
%
% Applied Regression Analysis
% N.R. Draper and H. Smith
% 2nd edition, John Wiley and Sons, 1981
%
%     Copyright Sunil L. Kukreja 11 May 1998
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

t_test

```
function [thetan,vlabel,err,std_theta] =t_test(vlabel,lenu,  
        theta,PHI,std_theta,conf_level,n,nx,ny,ne,x,z,err);  
% t-test Standard deviation of parameters using least-squares  
% methods.  
%  
% Function:[thetan,vlabel,err,std_theta] =t_test(vlabel,lenu,  
%         theta,PHI,std_theta,conf_level,n,nx,ny,ne,x,z,err);  
%  
% Inputs:  
%  
%     vlabel - vector containing the variables associated  
%             with each row of theta (see name.m)  
%     theta - estimated parameter vector  
%     std_theta - standard deviation of parameters (see ls_std.m)  
%     conf_level - standard deviation of parameters  
%             z - output  
%             x - input  
%             err - residuals  
%             ny - output lags  
%             nx - input lags  
%             ne - error lags  
%  
%     conf_level - 80, 90, 95, 97.5, 99 or 99.5 percent  
%  
% Outputs:  
% thetan - parameter vector with significant terms  
%  
% Method based on t-test in: Linear Regression Analysis  
% George A.F. Seber 1977  
%  
%  
%     Sunil L. Kukreja  12 May 1998  
%     Copyright Sunil L. Kukreja  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

vor_high

```

function [std_theta,sys_theta,init_theta,vlabel]=vor_high(u,y,index);
% Identifies the slow phase parameters of the VOR
% system with initial conditions
%
%   [std_theta,sys_theta,init_theta,vlabel]
%   =vor_high(u,y,index);
% This function computes the parameters for a
% high-pass VOR system.
%
% The structure in this function is assumed to
% be known, e.g. n=3, nx=1 and ny=1 which corresponds
% to 3rd order nonlinearity and first order
% dynamics on the input-output terms.
%
% Assumed Model Structure (HIGH-PASS):
%(3rd Order)
% NL Input + |      KS      |
% ----->| -----> Output
%          | tau S + 1 |
%          | -----|
%          Linear System
%
% FUNCTION INPUTS:
% u= the input
% y= the output
% index= the start and stop indices of each segment,
% e.g. |2  345|
%      |678 982|
% where 2 and 678 are the start points and 345 and
% 982 are the stop points of the segments
% FUNCTION OUTPUTS:
% std_theta= std of estimated parameters
% sys_theta= the identified system parameters
% init_theta= the identified initial conditions
% vlabel= the row labels of the theoretical and
%          identified parameters
%
%       Sunil L. Kukreja  20 January 2000
%       Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

vor_low

```

function [std_theta,sys_theta,init_theta,vlabel]=vor_low(u,y,index);
% Identifies the slow phase parameters of the VOR
% system with initial conditions
%
%   [std_theta,sys_theta,init_theta,vlabel]
%   =vor_low(u,y,index);
%   This function computes the parameters for a
%   low-pass VOR system.
%
%   The structure in this function is assumed to
%   be known, e.g. n=3, nx=1 and ny=1 which corresponds
%   to 3rd order nonlinearity and first order
%   dynamics on the input-output terms.
%
%   Assumed Model Structure (LOW-PASS):
%   (3rd Order)
%   NL Input + |      K      |
%   ----->| -----> Output
%           | tau S + 1 |
%           |-----|
%           Linear System
%
%   FUNCTION INPUTS:
%   u= the input
%   y= the output
%   index= the start and stop indices of each segment,
%   e.g. |2   345|
%        |678 982|
%   where 2 and 678 are the start points and 345 and
%   982 are the stop points of the segments
%   FUNCTION OUTPUTS:
%   std_theta= std of estimated parameters
%   sys_theta= the identified system parameters
%   init_theta= the identified initial conditions
%   vlabel= the row labels of the theoretical and
%           identified parameters
%
%   Sunil L. Kukreja  10 September 1999
%   Copyright Sunil L. Kukreja
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Bibliography

- [1] Henery D. I. Abarbanel. *Analysis of Observed Chaotic Data*. Springer-Verlag, New York, first edition, 1996.
- [2] Henery D. I. Abarbanel and Matthew B. Kennel. Local false nearest neighbors and dynamical dimensions from observed chaotic data. *Phys. Rev. E*, 47(5):3057–3068, 1993.
- [3] K.M. Adeney and M.J. Korenberg. Fast orthogonal search for array processing and spectrum estimation. In *IEE Proc. Vision Image Signal Process.*, volume 141, pages 13–18, 1994.
- [4] K.M. Adeney and M.J. Korenberg. Iterative fast orthogonal search algorithm for MDL-based training of generalized single-layer networks. *Neural Networks*, 13:787–799, 2000.
- [5] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, AC-19:716–723, 1974.
- [6] R. Alur and D.L. Dill. Automata for modeling real-time systems. In *Proc. ICALP*. Lecture Notes in Computer Science 443, pages 322–335. Springer-Verlag, 1990.
- [7] R. Alur and T.A. Henzinger. Modularity for timed and hybrid systems. In A. Mazurkiewicz and J. Winkowski, editors, *CONCUR '97: Concurrency Theory*, Lecture Notes in Computer Science 1243, pages 74–88. Springer-Verlag, 1997.

- [8] K.J. Åström. *Computer-Controlled Systems: Theory and Design*. Prentice Hall, Upper Saddle River, N.J., 3rd edition, 1997.
- [9] R.W. Baloh, K.M. Jacobson, K. Beukirch, and V. Honrubia. Horizontal Vestibulo-ocular Reflex after acute peripheral lesions. *Acta Otolaryngology (Suppl.)*, 468:323–327, 1989.
- [10] J.V. Beck and K.J. Arnold. *Parameter Estimation in Engineering and Science*. Wiley series in probability and mathematical statistics. John Wiley & Sons, New York, 1977.
- [11] L. Behera and N. Kirubanandan. A hybrid neural control scheme for visual-motor coordination. *IEEE Control Systems Magazine*, 19(4):34–41, 1999.
- [12] Julius S. Bendat. *Nonlinear Systems Techniques and Applications*. Johns Wiley and Sons, New York, first edition, 1998.
- [13] Julius S. Bendat and Allan G. Piersol. *Random Data, analysis and measurement procedures*. Johns Wiley and Sons, New York, second edition, 1986.
- [14] B. Bérard, P. Gastin, and A. Petit. On the power of non-observable actions in timed automata. In *Actes du STACS '96*, Lecture Notes in Computer Science 1046, pages 257–268. Springer-Verlag, 1996.
- [15] Peter J. Bickel and David A. Freedman. Some asymptotic theory for the bootstrap. *The Annals of Statistics*, 9(6):1196–1217, 1981.
- [16] Peter J. Bickel and David A. Freedman. Bootstrapping regression models with many parameters. Technical Report 7, Department of Statistics, University of California (Berkeley), Berkeley, California, June 1982.
- [17] S.A. Billings and S.Y. Fakhouri. Identification of systems containing linear dynamic and static nonlinear elements. *Automatica*, 18:15–26, 1982.

- [18] S.A. Billings and G.N. Jones. Orthogonal least-squares parameter estimation algorithms for non-linear stochastic systems. *International Journal of Systems Science*, 23(7):1019–1032, 1992.
- [19] S.A. Billings and I.J. Leontaritis. Parameter estimation techniques for nonlinear systems. In *6th IFAC Symposium on Identification & System Parameter Estimation*, volume 6, pages 427–432. IFAC/SYSID, 1982.
- [20] S.A. Billings, Chen. S., and M.J. Korenberg. Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. *International Journal of Control*, 49(6):2157–2189, 1989.
- [21] S.A. Billings and W.S.F. Voon. Structure detection and model validation tests in the identification of nonlinear systems. *IEE Proceedings, Pt. D.*, 130:193–199, 1983.
- [22] S.A. Billings and W.S.F. Voon. Least squares parameter estimation algorithms for non-linear systems. *International Journal of System Science*, 15(6):601–615, 1984.
- [23] S.A. Billings and W.S.F. Voon. Correlation based model validity tests for non-linear models. *International Journal of Control*, 44(1):235–244, 1986.
- [24] S.A. Billings and W.S.F. Voon. A prediction-error and stepwise-regression estimation algorithm for non-linear systems. *International Journal of Control*, 44:803–822, 1986.
- [25] S.A. Billings and Q.M. Zhu. Nonlinear model validation using correlation tests. *International Journal of Control*, 60(6):1107–1120, 1994.
- [26] Åke Björck. *Numerical Methods for Least Squares Problems*. Siam, Philadelphia, first edition, 1996.
- [27] John D. Bomberger and Dale E. Seborg. Determination of model order for NARX models directly from input-output data. *J. Proc. Cont.*, 8(56):459–468, 1998.

- [28] G.E.P. Box and G.M. Jenkins. *Time Series Analysis: Forecasting and Contro*. Holden-Day, San Francisco, first edition, 1970.
- [29] S. Boyd and L.O. Chua. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE transactions on Circuits and Systems*, CAS-32(11):1150–1161, 1985.
- [30] P.J. Brockwell. *Time Series: Theory and Methods*. Springer-Verla, New York, second edition, 1995.
- [31] P.M.T. Broersen and H.E. Wensink. On finite sample theory for autoregressive model order selection. *IEEE Trans. Signal Processing*, 42(1), 1993.
- [32] D.E. Budil, S. Lee, S. Saxena, and J.H. Freed. Nonlinear-least-squares analysis of slow-motion ERP spectra in one and two dimensions using a modified Levenberg-Marquardt algorithm. *J. Magnetic Resonance, Series A*, 120:155–189, 1996.
- [33] Kenneth P. Burnham and David R. Anderson. *Model Selection and Inference: A Practical Information-Theoretic Approach*. Springer-Verlag, New York, first edition, 1998.
- [34] Christopher Chatfield. *The analysis of time series : an introduction*. Chapman & Hall, first edition, 1996.
- [35] S. Chatterjee and M. Machler. Robust regression: A weighted least squares approach. *Communications in Statistics - Theory and Methods*, 26(6):1381–1394, 1997.
- [36] R. Chen and R.S. Tsay. Nonlinear additive ARX models. *J. Am. Stat. Assoc.*, 88(423):955–967, 1993.
- [37] S. Chen and S.A. Billings. Representations of non-linear systems: the NARMAX model. *International Journal of Control*, 49(3):1013–1032, 1989.

- [38] S. Chen, S.A. Billings, and W. Luo. Orthogonal least squares methods and their applications to nonlinear system identification. *International Journal of Control*, 50:1873–1896, 1989.
- [39] A. Choen. *Biomedical Signal Processing*, volume 1, chapter Time and Frequency Domains Analysis. CRC Press, Boca Raton, 1986.
- [40] K.S. Chun and D.A. Robinson. A model of quick phase generation in the vestibulo-ocular reflex. *Biol. Cybern.*, 28:209–221, 1978.
- [41] J.B. Copas. Regression, prediction and shrinkage. *Journal of the Royal Statistical Society B*, 45(2):311–354, 1983.
- [42] B. David and G. Bastin. A maximum likelihood parameter estimation method for nonlinear dynamical systems. In *Proceedings of the 38th IEEE Conference on Decision and Control*, volume 38, pages 612–617, Phoenix, Arizona, December 1999.
- [43] B. de Moor. Total least squares for affinely structured matrices and the noisy realization problem. *IEEE Transactions on Signal Processing*, 42(11):3104–3113. 1994.
- [44] J.L. Douce. Identification of a class of nonlinear systems. In *4th IFAC Symposium on Identification & System Parameter Estimation*, volume 4, pages 1–14. IFAC/SYSID, 1976.
- [45] N.R. Draper and H. Smith. *Applied Regression Analysis*. John Wiley & Sons, New York, second edition, 1981.
- [46] B. Efron. Computer and the theory of statistics: Thinking the unthinkable. *SIAM Review*, 21(4):460–480, 1979.
- [47] B. Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78:316–331, 1983.

- [48] B. Efron and R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, first edition, 1993.
- [49] M.A. Efroymson. *Multiple Regression and Correlation*. in *Mathematical Methods for Digital Computers*. John Wiley & Sons, New York, first edition, 1960.
- [50] J. Eker and J. Malmborg. Design and implementation of a hybrid control strategy. *IEEE Control Systems Magazine*, 19(4):12–21, 1999.
- [51] P. Eykhoff. *System Identification*. John Wiley & Sons, New York, first edition, 1974.
- [52] M. Fliess and D. Normand-Cyrot. On the approximation of nonlinear systems by some simple state-space models. In *Identification and System Parameter Estimation*, volume 6, pages 511–514, Washington, DC, 1982. 6th IFAC/SYSID Symposium.
- [53] G.F. Franklin, J.D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Addison-Wesley, New York, third edition, 1994.
- [54] Donald R. Fredkin and John A. Rice. Method of false nearest neighbors: A cautionary note. *Phys. Rev. E*, 51(4):2950–2954, 1995.
- [55] David A. Freedman. Bootstrapping regression models. *The Annals of Statistics*, 9(6):1218–1228, 1981.
- [56] J.E. Freund. *Mathematical Statistics*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, first edition, 1962.
- [57] H.L. Galiana. A nystagmus strategy to linearize the vestibulo-ocular reflex. *IEEE Transactions on Biomedical Engineering*, 38(6):532–543, 1991.
- [58] H.L. Galiana, H.L. Smith, and A. Katsarkas. Comparing linear and non-linear methods for the analysis of the Vestibulo-ocular Reflex. *Acta Otolaryngology*, 115:585–596, 1995.

- [59] A.H.W. (Ton) Geerts and J.M. Schumacher. Impulsive-smooth behavior in multimode systems Part I: State-space and polynomial representations. *Automatica*, 32(5):747–758, 1996.
- [60] Gene H. Golub. *Matrix computation*. The Johns Hopkins University Press, Baltimore, second edition, 1993.
- [61] G.C. Goodwin and R.L. Payne. *Dynamic System Identification: Experiment Design and Data Analysis*, volume 136 of *Mathematics in Science and Engineering*. Academic Press, New York, 1977.
- [62] G.C. Goodwin and M. Salgado. A stochastic embedding approach for quantifying uncertainty in estimation of restricted complexity models. *Int. J. of Adaptive Control and Signal Processing*, 3:333–356, 1989.
- [63] J.D. Hamilton. *Time Series Analysis*. Princeton University Press, Princeton, N.J, first edition, 1994.
- [64] Michael A. Hanson and Dale E. Seborg, editors. *Nonlinear Process Control*. Prentice-Hall, New Jersey, 1997. ISBN 0-13-625179-X.
- [65] A. C. Harvey. *Time Series Models*. John Wiley & Sons, New York, first edition, 1981.
- [66] S.S. Haykin. A unified treatment of recursive digital filtering. *IEEE Trans. Auto. Control*, 17:113–116, 1972.
- [67] T.A. Henzinger. The theory of hybrid automata. In *Proc. 11th Symp. on Logic in Computer Science*, pages 278–292. IEEE Computer Society Press, 1996.
- [68] Tien C. Hsia. *System Identification: Least-Squares Methods*. Lexington Books, D.C. Heath and Company, Lexington, Massachusetts, 1977.
- [69] Sabine Van Huffel and Joos Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Siam, Philadelphia, first edition, 1991.

- [70] I.W. Hunter and R.E. Kearney. Dynamics of human ankle stiffness: variation with mean ankle torque. *Journal of Biomechanics*, 15(3):753–756, 1982.
- [71] C.M. Hurvich and C. L. Tsai. The impact of model selection on inference in linear regression. *The American Statistician*, 44:214–217, 1990.
- [72] R.L. Kashyap. Inconsistency of the ACI rule for estimating the order of autoregressive models. *IEEE Trans. on Automatic Control*, 25(5):996–998, October 1980.
- [73] A. Katsarkas, H.L. Galiana, and H.L. Smith. Vestibulo-ocular Reflex (VOR) biases in normal subjects and patients with compensated Vestibular loss. *Acta Otolaryngology*, 115:476–483, 1995.
- [74] R.E. Kearney and I.W. Hunter. Dynamics of human ankle stiffness: variation with displacement amplitude. *Journal of Biomechanics*, 15:753–756, 1982.
- [75] R.E. Kearney and I.W. Hunter. System identification of human triceps surae stretch reflex dynamics. *Experimental Brain Research*, 51:117–127, 1983.
- [76] R.E. Kearney and I.W. Hunter. System identification of human stretch reflex dynamics: tibialis anterior. *Experimental Brain Research*, 56:40–49, 1984.
- [77] R.E. Kearney and I.W. Hunter. Nonlinear identification of stretch reflex dynamics. *Annals of Biomedical Engineering*, 16:79–94, 1988.
- [78] R.E. Kearney and I.W. Hunter. System identification of human joint dynamics. *Critical Reviews in Biomedical Engineering*, 18(1):55–87, 1990.
- [79] R.E. Kearney, R.B. Stein, and L. Parameswaran. Identification of intrinsic and reflex contributions to human ankle stiffness dynamics. *IEEE Trans. Biomed. Eng.*, 44(6):493–504, 1997.
- [80] R.E. Kearney, P.L. Weiss, and R. Morier. System identification of human ankle dynamics: intersubject variability and intrasubject reliability. *Clinical Biomechanics*, 5(4):205–217, 1990.

- [81] M. Kendall and J.K. Ord. *Time Series*. Edward Arnold, Great Britain, third edition, 1990.
- [82] M.B. Kennel, R. Brown, and H.D.I. Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A*, 45(6):3403–3411, 1992.
- [83] M.J. Korenberg. Identification of biological cascades of linear and static nonlinear elements. *Proceedings of the Midwest Symposium on Circuit Theory*, 18.2:1–9, 1973.
- [84] M.J. Korenberg. Orthogonal identification of nonlinear difference equation models. *Proceedings of the Midwest Symposium on Circuit Theory*, 1:90–95, 1985.
- [85] M.J. Korenberg. A robust orthogonal algorithm for system identification and time series analysis. *Biological Cybernetics*, 60:267–276, 1989.
- [86] M.J. Korenberg, April 2001. Personal Communications.
- [87] M.J. Korenberg, S.A. Billings, Y.P. Liu, and P.J. McIlroy. Orthogonal parameter estimation algorithm for non-linear stochastic systems. *International Journal of Control*, 48(1):193–210, 1988.
- [88] M.J. Korenberg and I.W. Hunter. The identification of nonlinear biological systems: Wiener kernel approaches. *Annals of Biomedical Engineering*, 18:629–654, 1990.
- [89] R.L. Kosut, M.K. Lau, and S.P. Boyd. Set-membership identification of systems with parameteric and nonparametric uncertainty. *IEEE Trans. Automatic Control*, 37(7):929–942, 1992.
- [90] S.L. Kukreja, H.L. Galiana, and R.E. Kearney. Structure detection of NARMAX models using bootstrap methods. In *Proceedings of the 38th IEEE Conference on Decision and Control*, volume 38, pages 1071–1076, Phoenix, Arizona, December 1999.

- [91] S.L. Kukreja, H.L. Galiana, H.L.H. Smith, and R.E. Kearney. Parametric identification of nonlinear hybrid systems. In *Proc. BMES/IEEE-EMBS*, volume 21, page 991, Atlanta, Georgia, October 1999.
- [92] S.L. Kukreja, R.E. Kearney, and H.L. Galiana. NARMAX representation of a parallel pathway model of ankle dynamics. In *Proc. CMBECC*, volume 24, pages 30–31, Edmonton, Canada, June 1998.
- [93] S.L. Kukreja, R.E. Kearney, and H.L. Galiana. Structure detection of nonlinear dynamic systems using bootstrap methods. In *Proc. IEEE EMBS*, volume 20, pages 3020–3023, Hong Kong, China, October 1998.
- [94] Sunil L. Kukreja. Method for estimation of continuous-time models of linear time-invariant systems via the bilinear transform. Master's thesis, McGill University, Department of Biomedical Engineering, 1996.
- [95] S. Kullback. *Information Theory and Statistics*. John Wiley & Sons, New York, 1959.
- [96] L.H Lee and K. Poolla. On statistical model validation. *Trans. ASME*, 118:226–236, 1996.
- [97] M.D. Lemmon, K.X.He, and I. Markovsky. Supervisory hybrid systems. *IEEE Control Systems Magazine*. 19(4):42–55, 1999.
- [98] I.J. Leontaritis and S.A. Billings. Input-output parametric models for non-linear systems part I: deterministic non-linear systems. *International Journal of Control*, 41(2):303–328, 1985.
- [99] I.J. Leontaritis and S.A. Billings. Input-output parametric models for non-linear systems part II: stochastic non-linear systems. *International Journal of Control*, 41(2):329–344, 1985.
- [100] L. Ljung and L. Guo. Classical model validation for control design purposes. *Mathematical Modelling of Systems*, 3(1):27–42, 1997.

- [101] Lennart Ljung. *System Identification*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, first edition, 1987.
- [102] M. Mantel. Why stepdown procedures in variable selection. *Technometrics*, 12:621–625, 1970.
- [103] P.Z. Marmarelis and V.Z. Marmarelis. *Analysis of Physiological Systems – the White Noise Approach*. Computers in Biology and Medicine. Plenum Press, New York, 1978.
- [104] The MathWorks Inc., Natwick, MA. *Simulink: Dynamic System Simulation for Matlab*, 2 edition, January 1997.
- [105] Jerry M. Mendel. *Lessons in estimation theory for signal processing, communications and control*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, first edition, 1995.
- [106] A.J. Miller. The convergence of Efroymsen’s stepwise regression algorithm. *The American Statistician*, 50(2):180–181, 1996.
- [107] M.M. Mirbagheri, H. Barbeau, and R.E. Kearney. Intrinsic and reflex contributions to human ankle stiffness: Variation with activation level and position. *Experimental Brain Research*, 2000. In Press.
- [108] M.M. Mirbagheri and R.E. Kearney. Mechanisms underlying a third-order parametric model of dynamic reflex stiffness. In *Proc. IEEE EMBS*, volume 22, pages 1241–1242, Chicago, USA, July 2000.
- [109] M.M. Mirbagheri, R.E. Kearney, H. Barbeau, and M. Ladouceur. Parametric modeling of the reflex contribution to dynamic ankle stiffness in normal and spinal cord injured spastic subjects. In *Proc. IEEE EMBS*, volume 17, pages 1241–1242, Montreal, Canada, September 1995.
- [110] R.L. Morier, P.L. Weiss, and R.E. Kearney. Low inertia, rigid limb fixation using glass fibre casting bandage. *Medical and Biological Engineering and Computing*, 28:96–99, 1990.

- [111] B. Ninnes and G.C. Goodwin. Estimation of model quality. *Automatica*, 31(12):1771–1797, 1995.
- [112] M. Noshiro, H. Shindou, Y. Fukuoka, M. Ishikawa, H. Minanitani, K. Sakamoto, A. Tanakadate, and S. Nebuya. NARMAX model of the pCO₂ control system in man estimated by neural computation. In *Proc. IEEE EMBS*, volume 18, pages 999–1000, Amsterdam, The Netherlands, October 1996.
- [113] C.W. Ostrom. *Time Series Analysis: Regression Techniques*. Sage Publications, Newberry Park, Calif., second edition, 1990.
- [114] G.D. Paige. Nonlinearity and asymmetry in human Vestibulo-ocular Reflex. *Acta Otolaryngology*, 108:1–8, 1989.
- [115] S.M. Pandit and S.M. Wu. *Time Series and System Analysis with Application*. John Wiley & Sons, New York, first edition, 1983.
- [116] R.K. Pearson. Nonlinear input/output modelling. *J. Proc. Cont.*, 5(4):197–211. 1995.
- [117] D.N. Politis. Computer-intensive methods in statistical analysis. *IEEE Signal Processing Magazine*, 15(1):39–55, 1998.
- [118] K. Poolla, P.P. Khargonekar, A. Tikku, J. Krause, and K. Nagpal. A time-domain approach to model validation. *IEEE Trans. Automatic Control*, 39(5):951–959, 1994.
- [119] M. Pottmann and R.K. Pearson. Block-oriented NARMAX models with output multiplicities. *AIChE Journal*, 44(1):131–140, 1998.
- [120] M.B. Priestley. *Spectral Analysis and Time Series*. Academic Press, San Diego, first edition, 1981.
- [121] J.G. Proakis and D.G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*. Prentice Hall, New Jersey, third edition, 1996.

- [122] G. Qian. Computations and analysis in robust regression model selection using stochastic complexity. *Computational Statistics*, 14(3):293–315, 1999.
- [123] Claudio G. Rey and Henrietta L. Galiana. Parametric classification of segments in ocular nystagmus. *IEEE Trans. Biomed. Eng.*, 38(2):142–148, 1991.
- [124] Claudio G. Rey and Henrietta L. Galiana. Transient analysis of vestibular nystagmus. *Biol. Cybern.*, 69:395–405, 1993.
- [125] Carl Rhodes and Manfred Morari. The false nearest neighbors algorithm: An overview. *Computers Chem. Engng.*, 21:S1149–S1154, 1997.
- [126] Carl Rhodes and Manfred Morari. False-nearest-neighbors algorithm and noise-corrupted time series. *Phys. Rev. E*, 55(5):6162–6170, 1997.
- [127] Carl Rhodes and Manfred Morari. Determining the model order of nonlinear input/output systems. *AIChE Journal*, 44(1):151–163, 1998.
- [128] J. Rissanen. Modelling by shortest data description. *Automatica*, 14:465–471, 1978.
- [129] D.A. Robinson. The use of control systems analysis in the neurophysiology of eye movements. *Ann. Rev. Neurosci.*, 4:463–503, 1981.
- [130] Wilson J. Rugh. *Nonlinear system theory, the Volterra/Wiener approach*. The Johns Hopkins University Press, Baltimore, 1981.
- [131] Wilson J. Rugh. *Linear System Theory*. Printice Hall, Inc., Englewood Cliffs, New Jersey, first edition, 1993.
- [132] R. Schmidt and F. Lardini. On the predominance of anti-compensatory eye movements in vestibular nystagmus. *Biol. Cybern.*, 23:135–148, 1976.
- [133] George A.F. Seber. *Linear Regression Analysis*. John Wiley & Sons, New York, first edition, 1977.

- [134] B.N. Segal and J.S. Outerbridge. A nonlinear model of semicircular canal primary afferents in bullfrog. *J. Neurophysiol.*, 47:563–578, 1982.
- [135] J. Shao. Bootstrap model selection. *Journal of the American Statistical Association*, 91(434):655–665, 1996.
- [136] Jun Shao and Dongsheng Tu. *The jackknife and bootstrap*. Springer Verlag, New York, NY, first edition, 1995.
- [137] H. Shindou, M. Noshiro, Y. Fukuoka, M. Ishikawa, and H. Minanitani. A new method for parameter estimation in the NARMAX model using neural computation. In *Proc. IEEE EMBS*, volume 17, pages 329–330, Montreal, Canada, September 1995.
- [138] William M. Siebert. *Circuits, Signals, and Systems*. The MIT Press, Cambridge, Massachusetts, first edition, 1986.
- [139] Naresh K. Sinha. Estimation of transfer function of continuous system from sampled data. *Proc. IEE*, 119(5):612–614, 1972.
- [140] N.K. Sinha and B. Kuszta. *Modeling and Identification of Dynamic Systems*. Van Nostrand Reinhold Company Inc., New York, first edition, 1983.
- [141] J. Sjöberg, Q. Zhang, L. Ljung, B. Delyon A. Benveniste, P.Y. Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724, 1995.
- [142] K.W. Smillie. *An Introduction to Regression and Correlation*. Academic Press Inc., London, first edition, 1966.
- [143] H.L.H. Smith and H.L. Galiana. The role of structural symmetry in linearizing ocular reflexes. *Biol. Cybern.*, 65:11–22, 1991.
- [144] Jon M. Smith. *Mathematical Modeling and Digital Simulation for Engineers and Scientists*. John Wiley & Sons, Inc., New York, second edition, 1987.

- [145] R.S. Smith and J.C. Doyle. Model invalidation: A connection between robust control and identification. *IEEE Trans. Automatic Control*, 37(7):942–952, 1992.
- [146] E.D. Sontag. *Polynomial Response Maps*, volume 13 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, Berlin, 1979.
- [147] B.A. Starnes and J.B. Birch. Asymptotic results for model robust regression. *Journal of Statistical Computation and Simulation*, 66(1):19–34, 2000.
- [148] Fredrik Tjärnström. The use of bootstrap in system identification. Technical Report LiTH-ISY-R-2113, Department of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden, Mar 1999.
- [149] Fredrik Tjärnström. Computing uncertainty regions with simultaneous confidence degree using bootstrap. In *Preprints of the 12th IFAC Symposium on System Identification, Santa Barbara, USA*, number FrPM1-6, Jun. 2000.
- [150] Fredrik Tjärnström. Quality estimation of approximate models. Technical Report Licentiate Thesis No. 810, Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden, Feb. 2000.
- [151] Fredrik Tjärnström and Lennart Ljung. Estimating the variance in case of undermodeling using bootstrap. In *Proceedings of the 38th IEEE Conference on Decision and Control*, pages 2394–2399, Phoenix, Arizona, USA, Dec. 1999.
- [152] C. Tomlin, G.J. Pappas, and S. Sastry. Conflict resolution for air traffic management: a case study in multi-agent hybrid systems. Technical Report UCB/ERL/M96/38, Department of Electrical and Computer Science, UC Berkeley, Berkeley, CA, July 1996.
- [153] J.K. Tugnait. Linear model validation and order selection using higher order statistics. *IEEE Trans. Signal Processing*, 42(7):1728–1736, 1994.

- [154] J.K. Tugnait. Model validation and order selection for linear model fitting using third- and fourth-order cumulants. *IEEE Trans. Signal Processing*, 47(9):2433–2443, 1999.
- [155] H. Unbehauen. Some new trends in identification and modeling of nonlinear dynamical systems. *Applied Mathematics and Computation*, 78:279–297, 1996.
- [156] M. Verhaegen. Identification of the deterministic part of MIMO state space models given in innovations form from input-output data. *Automatica*, 30(1):61–74, 1994.
- [157] Michel Verhaegen. Subspace model identification, part 3. analysis of the ordinary output-error state-space model identification algorithm. *Int. J. Control*, 58:555–586, 1993.
- [158] Michel Verhaegen and Patrick Dewilde. Subspace model identification part 1. the output-error state-space model identification class of algorithms. *Int. J. Control*. 56(5):1187–1210, 1992.
- [159] Michel Verhaegen and Patrick Dewilde. Subspace model identification part 2. analysis of the elementary output-error state-space model identification algorithm. *Int. J. Control*, 56(5):1211–1241, 1992.
- [160] B. Wahlberg and L. Ljung. Hard frequency-domain model error bounds from least-squares like identification techniques. *IEEE Trans. Automatic Control*, 37(7):900–912, 1992.
- [161] Bo Wahlberg. The effects of rapid sampling in system identification. *Automatica*, 26(1):167–170, 1990.
- [162] P.L. Weiss, I.W. Hunter, and R.E. Kearney. Human ankle joint stiffness over the full range of muscle activation levels. *J. Biomechanics*, 21(7):539–544, 1988.
- [163] P.L. Weiss, R.E. Kearney, and I.W. Hunter. Position dependence of ankle joint dynamics – I passive mechanics. *J. Biomechanics*, 19(9):727–735, 1986.

- [164] P.L. Weiss, R.E. Kearney, and I.W. Hunter. Position dependence of ankle joint dynamics – II active mechanics. *J. Biomechanics*, 19(9):737–751, 1986.
- [165] J.C. West. Nonlinear signal distortion correlation. *International Journal of Control*, 2:529–538, 1965.
- [166] David T. Westwick, Béla Suki, and Kenneth R. Lutchén. Sensitivity analysis of kernel estimates: Implications in nonlinear physiological system identification. *Annals of Biomedical Engineering*, 26:488–501, 1998.
- [167] D.T. Westwick and R.E. Kearney. Nonparametric identification of nonlinear biomedical systems, part I: Theory. *Critical Reviews in Biomedical Engineering*, 26(3):153–226, 1998.
- [168] A.M. Zoubir and B. Boashash. The bootstrap and its applications in signal processing. *IEEE Signal Processing Magazine*, 15(1):56–76, 1998.

EPILOGUE

To read a poem (so the King of Hearts told the White Rabbit), "begin at the beginning and go on till you come to the end: then stop." Theses, like poems, are sequentially ordered structures, and thus inevitably have a beginning and an end (although very painfully reached); this is the final paragraph of this thesis. But the theory of parametric system identification, as we have seen, is not simply a cascaded arrangement of topics. There are multiple loops and branches, many parallel and crossing paths. Most ideas are linked directly and indirectly to many others. There is no simple step-by-step route by which this multidimensional web can be systematically explored and comprehended. There is really no beginning, and no end. We cannot expect to appreciate one topic fully until we have considered others. And so we must continually circle back to examine earlier concepts from a new vantage point.

- Adapted from Circuits, Signals, and Systems
by William M. Siebert [138]