Simulation of Real Time Parameter Estimation Algorithms for Time Varying Systems

Abdalla Ayad Ben Zreiba

January 2000

This Thesis is submitted to the Nottingham Trent University in partial fulfilment of the requirements for the degree of Masters of Philosophy.

Abstract

To recognise trends embedded in patterns, a dynamical model of the system generating the patterns can be assumed. A second order model has a wide variety of patterns which can serve well in approximately describing the short-term behaviour of complex physical, financial, societal and biological systems. Apart from initial conditions, the output pattern of a simple second order system is completely defined by 3 parameters: natural frequency (ω), damping ratio (ζ) and external input (u).

Three algorithms are proposed and investigated in this study to estimate the parameters of an equivalent second order system from a given trajectory (in time or space) of the pattern. The algorithms combine successive 1^{st} order filters of specified cut-off frequencies, to provide smoothing and higher order derivative estimation, with non-linear static parameter estimators.

A complete simulation environment is devised enabling the three-parameter estimation algorithms to be tested for 3 categories of parameter sets: constant, variable with 1^{st} order dynamics and variable with 2^{nd} order dynamics. When the parameters have 2^{nd} order dynamics, they themselves may be modelled as having their own unique time varying patterns, i.e. have dynamical behaviour. This leads to a hierarchical parameter estimation process where on-line algorithms are needed to work concurrently with the actual system to provide a continuous estimate of the first level parameters. When these parameters are time varying,

then they in turn are submitted as input to another level of parameter estimation algorithm to estimate the parameters of their own dynamics. This process may be repeated, in theory at least, to as many levels as necessary until a set of parameters is found which is constant.

Accurate estimations of ω, ζ and u were made using non-linear combinations of time derivatives of the measured output of the system. Results of the simulations are presented which show that the algorithms can cope well with variable parameters.

The effect of measurement noise on the estimation accuracy is considered when the incoming trajectories are corrupted with random noise. Noise is simulated using a random number generator with zero-mean and added to the simulated system output. Analysis of the simulation results show varying abilities of the algorithms to cope with the noise perturbations. In some instances high prediction robustness were achieved, in others, simulations showed high sensitivity to noise.

Acknowledgements

I would like to thank the following people for their support during the course of this study. Dr. David Al-Dabass for his support, advice and helpful suggestion throughout the project.

The support staff in the department, both technical and administrative, deserve particular thanks. Furthermore, I would like to thank Dr. Graham Ball for being there when I needed him most.

Finally I would like to express my thanks to my wife, who gave me support during the period of study.

List of symbols

u(t)	Input variable
x(t)	Output variable
ω	Natural frequency
ζ	Damping ratio
G,G1,..G4	Cut-off frequency
Ex	Filter estimate of x
Ex'	Filter estimate of d/dx
Ex''	Filter estimate of d^2/dx^2
Ex'''	Filter estimate of d^3/dx^3
Ex''''	Filter estimate of d^4/dx^4
Eω	Estimated omega
Eζ	Estimated zeta
Eu	Estimated u
ωo	Omega of omega
ζo	Zeta of omega
uo	u of omega
ωz	Omega of zeta
ζz	Zeta of zeta
uz	u of zeta
ωu	Omega of u
ζu	Zeta of u
uu	u of u

Contents

Abstract i

Acknowledgements iii

Contents iv

List of Figures ix

1 INTRODUCTION 1

1.1 Parameter Estimation in General Dynamical Systems 1

1.2 Second Order System 3

1.2.1 Generality of the Second Order System 3

1.2.2 Formulation 3

1.2.3 Parameter Estimation 5

1.3 Hierarchical Models 6

1.4 Main Aims 7

2 LIMITATIONS OF CURRENT TECHNIQUES 8

2.1 Parameter Estimation 8

2.1.1 General Introduction 8

2.1.2 Parameter Estimation of a Second Order System 14

2.1.3 Parameter Estimation of Continuous Linear Systems 16

2.1.4 Parameter Estimation of Dynamic Systems 17

2.2 Hierarchical models 19

2.2.1 Parameter Estimation in Large-scale Systems 20

2.2.2 Parameter Estimation using Hierarchical Structure 22

2.2.3 Parameter Estimation for Distributed Parameter Systems 22

2.3 Summary 23

3 THREE PARAMETER ESTIMATION ALGORITHMS FOR TIME VARYING SYSTEMS 24

3.1 Introduction 24

3.1.1 Definitions of ω, ζ and u 25

3.2 Derivations of Parameter Estimation Algorithms 26

3.2.1 Algorithm-1: Three Points in x, x' and x'' 27

3.2.2 Algorithm-2: Two Points and One Extra Derivative 29

3.2.3 Algorithm-3: One Point and Two Extra Derivatives 31

3.2.4 Derivation of Filter Values 32

3.3 Categories of Parameter sets 32

3.3.1 Constant Parameters 32

3.3.2 Variables with 1^{st} Order Dynamics 32

3.3.3 Variables with 2^{nd} Order Dynamics 33

3.4 Problems Encountered during the Process 34

3.4.1 Parameter Estimation Algorithms without Noise 34

3.4.2 Sensitivity of Parameter Estimation Algorithms to Noise 35

3.4.3 Compensation for Lag and amplitude attenuation caused by Low gain: 35

3.5 Parameter Estimation of Hierarchical Models 37

4 SIMULATION SOFTWARE AND TEST STRATEGY 39

4.1 Simulation Structure 39

4.1.1 Structure Details 39

4.1.2 Initialization 48

4.1.3 Execution 49

4.2 Test Strategy 49

4.2.1 Varying a Single Parameter 49

4.2.2 Varying More than One Parameter 50

4.2.3 Sensitivity to Noise 51

4.3 Parameter Estimation of Hierarchical Models 58

4.3.1 Hierarchical Model for The ω Subsystem 59

4.3.2 Hierarchical Model for The ζ Subsystem 61

4.3.3 Hierarchical Model for The u Subsystem 62

5 SIMULATION RESULTS AND DISCUSSION 63

5.1 Estimating Constant Parameters 63

5.1.1 Algorithm 1 63

5.1.2 Algorithm 2 67

5.1.3 Algorithm 3 70

5.1.4 Discussion 72

5.2 Estimating Parameters with 1^{st} Order Dynamics 73

5.2.1 Algorithm 1 73

5.2.2 Algorithm 2 78

5.2.3 Algorithm 3 81

5.2.4 Discussion 85

5.3 Estimating Parameters with 2^{nd} Order Dynamics 86

5.3.1 Algorithm 1 86

5.3.2 Algorithm 2 89

5.3.3 Algorithm 3 92

5.3.4 Discussion 94

5.4 Parameter Estimation of Hierarchical model 95

5.4.1 Estimation of Variable Parameter ω with 2nd Order Dynamics 95

5.4.2 Estimation of Variable Parameter ζ with 2nd Order Dynamics 96

5.4.3 Estimation of Variable Parameter u with 2nd Order Dynamics 97

5.4.4 Discussion 98

6 CONCLUSIONS AND FURTHER WORK 99

6.1 Conclusions 99

6.1.1 Second Order Systems 99

6.1.2 Parameter Estimation Algorithms 100

6.1.3 Parameter Estimation of Hierarchical Models 102

6.1.4 Noise Sensitivity 103

6.2 Overview 103

6.3 Further Work 104

References 105

Appendix A 112

Appendix B 117

Appendix C 119

List of Figures

Figure 3.1: Three Parameter Estimation Algorithms 27

Figure 3.2: Parameter Estimation Algorithms schematic diagram 36

Figure 3.3: An example of a hierarchical model based on 13 2^{nd} order sub-systems..38

Figure 4.1: The D-vector, containing 7 simultaneous 1^{st} order derivatives with initial conditions in vector x, for the constant parameters case. 41

Figure 4.2: Exact and Estimated outputs and their derivatives for constant parameters (ω =10, ζ =0.1 and u =1; G=1000) 42

Figure 4.3: The D-vector, containing 10 simultaneous 1^{st} order derivatives with initial conditions in vector x, for variables parameters with 1^{st} order dynamics. 44

Figure 4.4: Exact and Estimated outputs and their derivatives for variables parameters with 1^{st} order dynamics (ω: exponentially varying from 10 to 12, ζ =0.1 and u =1; G=5000). 45

Figure 4.5: The D-vector, containing 13 simultaneous 1^{st} order derivatives with initial conditions in vector x for parameters with 2^{nd} order dynamics. 47

Figure 4.6: Exact and Estimated outputs and their derivatives for parameters with 2^{nd} order dynamics (ω: ωo=10, ζo=0.3, uo=20; ζ: ωz=5, ζz=0.3, uz=0.1; u: ωu=10,ζu=0.3 and uu=1). 48

Figure 4.7: The D-vector, containing 10 simultaneous 1^{st} order derivatives, varying ω, ζ and keeping u constant for variable parameters with 1^{st} order dynamics. 51

Figure 4.8: Iterative Euler integrator for the noise source, trajectory generator, the three parameters and time higher derivative estimators. 52

Figure 4.9: An example of random noise of 0.1 53

Figure 4.10: Results of the example before introducing the lag/amplitude compensation to the filter estimation of the simulated system and its 1st derivative. 55

Figure 4.11: Results of the example, after introducing lag and amplitude compensation to the filter estimation of the simulated system and its 1st derivative. 56

Figure 4.12.a: Higher derivative estimation: n=0.1, G=20. 57

Figure 4.12.b: Higher derivative estimation without compensation: n=0.1, G=30. 58

Figure 4.13: Iterative Euler integration for the filter estimate of Eω of the main system and its higher derivative estimators 61

Figure 4.14: Iterative Euler integration for the filter estimate of Eζ of the main system and its higher derivative estimators 62

Figure 4.15: Iterative Euler integration for the filter estimate of Eu of the main system and its higher derivative estimators 63

Figure 5.1: Estimated constant parameters ω , ζ and u using Algorithm-1. 64

Figure 5.2.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, using Algorithm-1, before compensation with G =20, Lag (L) =0 and no noise (n=0). 65

Figure 5.2.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, using Algorithm-1, after compensation with G =20, Lag (L) =32 and no noise (n=0). 65

Figure 5.3.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) = 32. 66

Figure 5.3.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-1 with G = 20, Lag (L) = 32. 66

Figure 5.3.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) = 32. 67

Figure 5.4: Estimated constant parameters ω, ζ and u using Algorithm-2. 68

Figure 5.5.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-2 with G = 20, Lag (L) = 32. 68

Figure 5.5.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-2 with G = 20, Lag (L) = 32. 69

Figure 5.5.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-2 with G = 20, Lag (L) = 32. 69

Figure 5.6: Estimated constant parameters ω =10, ζ =0.1 and u =1 using Algorithm-3 with G= 1000 70

Figure 5.7.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) = 32. 71

Figure 5.7.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-3 with G = 20, Lag (L) = 32. 71

Figure 5.7.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) = 32. 71

Figure 5.8.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-1 (separation time step = 40 points). 74

Figure 5.8.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-1 (separation time step =100 points). 75

Figure 5.9.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-1 75

Figure 5.9.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-1. 76

Figure 5.10: Estimated parameters with 1st order dynamics ω, keeping ζ and u constants using Algorithm-1 with G =20, Lag (L) =32 and no noise (n = 0). 77

Figure 5.11.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) = 32. 77

Figure 5.11.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) = 32. 78

Figure 5.12.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-2 (separation time step =40 points). 79

Figure 5.12.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-2 (separation time step = 100 points). 79

Figure 5.13.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-2. 80

Figure 5.13.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-2. 80

Figure 5.14.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n =0.001) using Algorithm-2 with G = 20, Lag (L) = 32. 81

Figure 5.14.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n =0.1) using Algorithm-2 with G = 20, Lag (L) = 32. 81

Figure 5.15.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-3 with G =1000 82

Figure 5.15.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-3 with G =300 82

Figure 5.16.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-3. 83

Figure 5.16.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-3. 83

Figure 5.17.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) = 32. 84

Figure 5.17.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) = 32. 84

Figure 5.18.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, using Algorithm-1 (separation time step = 40 points). 87

Figure 5.18.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, using Algorithm-1 (separation time step =100 points). 87

Figure 5.19: Estimated parameters with 2nd order dynamics for ω and ζ but keeping u constant using Algorithm-1. 88

Figure 5.20.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) =32. 88

Figure 5.20.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) =32. 89

Figure 5.21.a: Estimated variables with 2nd order dynamics for ω, ζ and u using Algorithm 2 (separation time step =40 points) 89

Figure 5.21.b: Estimated variables with 2nd order dynamics for ω, ζ and u using Algorithm 2 (separation time step = 10 points) 90

Figure 5.22: Estimated parameters with 2nd order dynamics for ω and ζ but keeping u constant using Algorithm-2. 90

Figure 5.23.a: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-2 with G = 20, Lag (L) =32. 91

Figure 5.23.b: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-2 with G = 20, Lag (L) =32. 91

Figure 5.24.a: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u using Algorithm-3 with G = 300 92

Figure 5.24.b: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u using Algorithm-3 with G = 1000 92

Figure 5.25: Estimated parameters with 2^{nd} order dynamics for ω and ζ but keeping u constant using Algorithm-3. 93

Figure 5.26.a: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) =32. 93

Figure 5.26.b: Estimated parameters with 2^{nd} order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) =32. 94

Figure 5.27: Estimated first level parameters (ωo, ζo and uo) of sub-system ω using Algorithm-3. 96

Figure 5.28: Estimated first level parameters (ωz, ζz and uz) of sub-system ζ using Algorithm-3 97

Figure 5.29: Estimated first level parameters (ωu, ζu and uu) of sub-system u using Algorithm-3. 97

CHAPTER 1 - INTRODUCTION

The subject of this project is the study of a parameter estimation problem encompassing a wide range of techniques and algorithms. Three hybrid methods were used for parameter estimation of a second order system. The aim of this Chapter is to introduce the subject of parameter estimation as well as the effects and the methods of modelling which have been used to study the behaviour of physical, biological, economic and social systems.

1.1 Parameter Estimation in General Dynamical Systems

Stimulated by the theory of classical mechanics, an important method interpreting the behaviour of processes has been expressed [D'Azzo et al, 1995], by means of differential equations, in terms of input and output variables u= u(t) and x = x(t), respectively:

$$a_0 \cdot \frac{d^n}{dt^n} x + \dots\dots\dots + a_{n-1} \cdot \frac{d}{dt} x + a_n \cdot x = b_0 \cdot \frac{d^n}{dt^n} u + \dots\dots\dots + b_{n-1} \cdot \frac{d}{dt} u + b_n \cdot u \qquad \dots\dots\dots\dots(1.1)$$

In the linear case the parameters a_i, b_j are independent of u, x and their derivatives. If, in addition, they do not depend on time either, we have a case of constant parameters. This is the most tractable case. These parameters may, however, depend on time. If any a_i or b_j does depend on u, x or their derivatives, the process is non-linear.

The general n^{th} order differential equation can be described in state-space terms as the transformation of n first order simultaneous differential equations. This is interpreted by employing matrix notation:

$$x'(t)=A \cdot x(t)+B \cdot u(t) \qquad y(t)=C \cdot x(t) \qquad \text{.................(1.2)}$$

If a system has r inputs and m outputs, then u (t) is an input column vector containing the r elements u_1 (t), u_2 (t), ...u_r (t) and y (t) is an output column vector containing the m elements y_1 (t), y_2 (t), ...y_m(t). Consequently the A matrix, 'the coefficient matrix of the process', must be of order (n x n), the B matrix, 'the distribution matrix', of order (n x r) and the C matrix, 'the output matrix', of order (m x n).

$$A := \begin{bmatrix} a_{11} & a_{12} & \cdot & a_{1n} \\ a_{21} & \cdot & \cdot & a_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & a_{nn} \end{bmatrix} \quad B := \begin{bmatrix} b_{11} & \cdot & \cdot & b_{1r} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ b_{n1} & b_{n2} & \cdot & b_{nr} \end{bmatrix} \quad C := \begin{bmatrix} c_{11} & \cdot & \cdot & c_{1n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ c_{m1} & c_{m2} & \cdot & c_{mn} \end{bmatrix}$$

The total number of parameters (N) that represent the behaviour of such a system are then: $n^2+n.r+m.n$. Since there are N unknowns, and n initial values (for x), it is necessary to use at least (N + n) points of the trajectories. The aim is to determine values for parameters within the modelled system that produce a closer fit to the measured data. The trajectory of the state variable x can be calculated for any time t, if values for the parameters of n^{th} order equation are known. For example, for a full second order system with two inputs and two outputs there will be '14' unknowns, 12 parameters and 2 initial values.

$$A := \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \qquad B := \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \qquad C := \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

1.2 Second Order System

1.2.1 Generality of the Second Order System

The second order equation is important because it can be used in a wide range of situations as an approximation to the actual process [Anand et al, 1995]. This is especially true for short periods of time. Although the parameters of a basic second order model are constant, it is often applied in situations where one or more of the parameters are time varying [D'azzo, 1995].

1.2.2 Formulation

The position x of an object subjected to an input u can be modelled as a second order system based on the physical characteristics of inertia, damping and elasticity as:

$$\frac{d^2}{dt^2}x + 2\cdot\zeta\cdot\omega\cdot\frac{d}{dt}x + \omega^2\cdot x = \omega^2\cdot u \qquad \text{..............(1.3)}$$

Where ζ relates to the damping of the system, ω relates to the natural frequency with $\omega = 2\pi f$ and u is the external input on the system. Each parameter is assumed to be independent of time.

A general solution of this equation is,

$$x(t)=\frac{u\cdot\omega^2\cdot e^{r_1\cdot t}}{r_1\cdot(r_1-r_2)}-\frac{u\cdot\omega^2\cdot e^{r_2\cdot t}}{r_2\cdot(r_1-r_2)}+\frac{u\cdot\omega^2}{r_1\cdot r_2} \qquad \text{............(1.4)}$$

Where,

$$r_1=\frac{-\zeta\cdot\omega+\sqrt{\zeta^2\cdot\omega^2-4\cdot\omega^2}}{2} \qquad \text{...............(1.5)}$$

$$r_2=\frac{-\zeta\cdot\omega-\sqrt{\zeta^2\cdot\omega^2-4\cdot\omega^2}}{2} \qquad \text{...............(1.6)}$$

Where x_0 (0) and x'_0 (0) are the initial values of x (t) and its first time derivative, i.e. the state variables of the system. The derivation of this solution is given in [Appendix A].

Common examples of a second order system are pendulums and spring & mass systems which move with Simple Harmonic Motion (SHM) about an equilibrium position. In the case of no damping, the angular frequency is the natural frequency ω_n of Equation 1.3. When the damping ratio, ζ, has a non-zero positive value, the amplitude of the oscillation will decrease with time and the actual frequency, ω_a, decreases according to,

$$\omega_a=\sqrt{(1-\zeta^2)}\,.\,\omega_n \qquad \text{..............(1.7)}$$

Negative values of damping ratio cause amplitude to increase and the system is then known to be unstable.

1.2.3 Parameter Estimation

Parameter estimation is a common problem in many areas of process modelling, both in on-line applications such as real time optimisation and in off-line applications such as the modelling of reaction kinetics and phase equilibrium [Isermann et. al, 1992]. The goal is to determine values of model parameters that provide the best fit to measured data. Given values for the parameters of a second order equation, the trajectory of the variable x can be calculated for any time t. This trajectory is defined by the values of the parameters used to create it.

The process of parameter estimation aims to perform the opposite of this scenario. Given a trajectory of x through time, what are the values of the parameters that generated this trajectory?

This is far from a trivial problem. Each point on the trajectory requires the solution of the non-linear Equation 1.4. The values of the parameters, which satisfy all the solutions, must match. Since there are five unknowns, three parameters and two initial values, it is necessary to use at least five points of the trajectory.

1.3 Hierarchical Models

The modelling phase is perhaps the most important phase in any control system design since ultimately, the controllers' effectiveness is limited by the quality of the model used. Modelling is usually done in two parts i.e. first a structural model based on the physical process being modelled is developed. Then, data is used to determine the parameters of the model. Although this parameter estimation problem is of fundamental importance, among the different methods used to deal with these problems are the hierarchical and decomposition techniques [Hassan et al. 1982].

To model the complete behaviour of higher dimensional (>2) systems a simple second order model with time varying parameters may be used. The behaviour of these parameters may be modelled as 1st or 2nd order systems, each with its own dynamical behaviour. This leads to a hierarchical parameter system process where on-line algorithms are needed to work concurrently with the actual system to provide a continuous estimate of the first level parameters. When these parameters are time varying, then they in turn are submitted as input to another level of parameter estimation algorithm to estimate the parameters of their own dynamics. This process may be repeated, in theory at least, to as many levels as necessary until a set of parameters is found which is constant.

1.4 Main Aims

The program of work aims to investigate the problem of parameter estimation of a simple linear second order system. Three hybrid methods of estimating ω, ζ and u are proposed and tested through simulation for 3 categories of parameter sets: constants, variables with 1^{st} order dynamics and variables with 2^{nd} order dynamics.

An important part of this study will examine the performance and robustness of the parameter estimation algorithms when the incoming trajectories of the simulated output are corrupted with noise.

CHAPTER 2 - LIMITATIONS OF CURRENT TECHNIQUES

2.1 Parameter Estimation

System identification or parameter estimation has remained a very active area of research for representing the behaviour of physical, biological, economic and social systems. It has been a fundamental element of engineering and many other fields for many decades, if not hundreds of years. It is defined as the study of determining the model, or structure of a process, using a limited number of input and output data measurements, that may or may not be disturbed by noise. However, in many cases, it is not feasible to assume a knowledge of the process parameters, and moreover, these are often subject to variation based on changing operating conditions. Therefore it is desirable for the values of these parameters to be obtained from input-output measurements on the process. Usually this procedure is called **parameter estimation.**

2.1.1 General Introduction

In this section, an introduction to parameter estimation theory is considered. The major properties of parameter estimation are discussed and illustrated on few examples.

Parameter estimation and system identification are parts of a very complex process. It is seen that the estimation problem is only a small part of the whole identification process, which is preceded and followed, by a number of other important steps. In practice, it is very important to keep this in mind, because the necessary efforts spent in the identification and estimation step is highly dependent upon the choices made in the previous steps.

A first very important choice to be made is the selection of a class of models. The model is a mathematical description of the studied system. It is possible to divide all these models in different categories using some criteria. One criterion is the parametric/non-parametric division.

In parametric models the system is described using a limited number of parameters. For example, the transfer function of a filter, the motion equations of a piston, etc. However, the same filter could also be described by giving its impulse response using a large number of points. This is an example of a non-parametric description.

Another criterion is **white/black box** description.

During the construction of the model, it is possible to use physical laws (Kirchoff laws, the laws of Newton, etc). The use of this knowledge is strongly dependent upon the insight and skill of the researcher. Here, the specialized knowledge of different scientific fields is brought into the identification process. E.g. to model a loudspeaker, it is necessary to have a profound insight in mechanical, electrical and acoustical problems. So, a model is developed based on a good knowledge of the internal working principles of the system. Such a model is called a white box model.

Another approach would be the black box model. Instead of making a detailed study and developing a model based on physical insight and knowledge, a mathematical model is proposed which allows to describe sufficiently well observed input and output measurements. In this way, the modelling effort is reduced significantly. E.g. instead of modelling a loudspeaker using physical laws, an input-output relation is proposed. This can be a transfer function of sufficiently high order [Schoukens et al, 1988].

The choice between these approaches strongly depends upon the aim of the study. If the results must give an insight into the working principles of the system, the white box approach is preferred. If the model will be used for short time prediction, a black box model may be sufficient [Fedorov, 1972].

After the class of model is determined, a specific model has to be selected. A typical example of this problem is choosing the order of a transfer function, what should be the minimal order of the numerator and the denominator to explain all the measured data sufficiently. Another example is the choice between different candidate models which all give a possible explanation of the observations [Schoukens et al, 1988].

This selection has to be made, using limited amount of measurements (information) which are usually disturbed with noise. Due to these noise influences the selection of a model is not unique, and there is an uncertainty on the final choice of the model. Also modelling errors can influence this choice. It is obvious that the design of the experiment used to get the necessary information for the selection process, will influence the uncertainty on the final result. The same will be true for the estimation step. It is possible to minimize the uncertainty on the estimates by an optimization of the input signals. Once the model is selected, its parameters have to be estimated to determine the model completely. Again measurements results are used during this step. Many methods have been presented to minimize the influence of the noise on the estimation. Frequently, the identification and estimation steps are linked together [Astrom, 1980].

When the identification/estimation step is finished, a first check can be made to verify the goodness of the fit and the adequacy of the proposed model. Due to noise, there will be a difference between the measurements and the estimates. These differences, called residuals, possess some known statistical properties. By checking this information (e.g. the mean value, standard deviation, etc) it is possible to conclude if systematic (model) errors still exist for the given input signals. Two algorithms are described here, Least Squares estimation and Maximum Likelihood estimation. A number of other well-known algorithms also exist, for example the Bayes' estimation algorithm and the Markov estimation algorithm (this topic is well covered in many standard textbooks such as Eykhoff (1974)).

Least Squares Estimation:

The principle of Least squares was formulated by Gauss at the end of the eighteenth century for determining the orbits of planets. According to this principle, the unknown parameters of a mathematical model should be chosen in such a way that the sum of the squares of the differences between the parameter values actually observed and their computed values, multiplied by numbers that measure the degree of precision, is minimized. Recently, an improved version of the Least Squares has been developed to handle the problem, where, the Least Squares algorithm can not be used for time varying systems, these include the Least Squares algorithm with selective data weighting, the Least Squares algorithm with covariance resetting and the Least Squares algorithm with covariance modification [Goodwin et al, 1984]. All of these modified algorithms have similar properties to the original Least Squares algorithms.

Maximum Likelihood Estimation

The maximum likelihood estimation method originally developed in statistics can be used to estimate the parameters of the models such as State-Space model (SS), Conditional Markovian Model (CM), Simultaneous Autoregressive model (SARM) or Autoregressive and Moving Average Model (ARMA), [Isermann et al, 1992]. It produces asymptotically consistent and efficient estimates. However, the derivation of the Log-Likelihood function requires knowledge of *a priori* (before the measurement), [Eykhoff, 1974] probability density of the present output and is extremely difficult even if the Gaussian case, and in general the Maximum Likelihood scheme, leads, to non-linear optimization problems. The Maximum Likelihood estimator is one of the best known estimators. A lot of the properties are proven under conditions of independent, identical disturbed noise on the measurements and a Log-likelihood function, which is differentiable twice. For some specific problems it is possible to prove the properties making less restrictive assumptions. However, there is in general no guarantee that the estimator still behaves in the same way if the previous conditions are not met.

Linear State Models

Our attention will be focussing on the important type of state model S(**A**,**B**,**C**,**D**) defined by

$$\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

Where **x** is an n-vector, the input **u** is a r-vector and the output **y** is an m-vector. It is well known that the system A(**A**,**B**,**C**,**D**) and S($\mathbf{TAT^{-1}}$,**TB**,$\mathbf{CT^{-1}}$,**D**) where **T** is non-singular matrix are equivalent in the sense that they have the same input-output relation (a comprehensive study of the state models is provided by Kailath, 1980). It can be verified that the systems S(**A**,**B**,**C**,**D**) and S($\mathbf{A}^\wedge$,$\mathbf{B}^\wedge$,$\mathbf{C}^\wedge$,$\mathbf{D}^\wedge$), "where ^ is the adjoint", are equivalent in the sense that they have the same input-output relation if:

$$\mathbf{D} = \mathbf{D}^\wedge$$

$$\mathbf{C}\,\mathbf{A}^k\mathbf{B} = \mathbf{C}^\wedge\,\mathbf{A}^{\wedge k}\,\mathbf{B}^\wedge \qquad k=0,1,\dots,n$$

The relations between the different representations were clarified by Kalman's work (Kalman, 1963). The impulse response and the transfer function represent only the part of the system S which is completely controllable. It is thus clear that only the completely controllable and completely observable part of a state model S(**A**,**B**,**C**,**D**) can be determined from input-output measurements. The impulse response and the transfer function are easily obtained from the state description. The problem of determining a state model from the impulse response is subtler. The problem of assigning a state model of the lowest possible order, which has a given impulse response has been solved by Ho and Kalman (1966). Again the solution is not unique. The model S(**A**,**B**,**C**,**D**) contains

$$\mathbf{N_1 = n^2 + n.r + n.m + r.m}$$

parameters. The fact that the input-output relation is invariant under linear transformation of the state variables implies that all N_1 parameters cannot be determined from input-output measurements. To obtain unique solutions as well as to be able to construct efficient algorithms, it is of great interest to find representations of the system which contain the smallest number of parameters i.e. Canonical representations. Canonical forms for linear systems were studied by Kailath (1980). If the matrix **A** has distinct eigenvalues, canonical forms can be obtained by a suitable choice of co-ordinates and the matrix **A** can be brought into diagonal form. This representation contain

$$\mathbf{N_2 = n(r+m) + r.m}$$

parameters. Since the system is completely controllable and observable there is at least one non-zero element in each row of the **B** matrix and of each column of the **C** matrix.

2.1.2 Parameter Estimation of a Second Order System

Broadly speaking, two approaches to parameter estimation exist, off-line and on-line. On-line methods will be described later in Section 2.2, Off-line methods use all data available *prior* to analysis. Consequently, they usually give estimates with high precision and set no time limit for the process of analysis. These include Least Squares (Kailath, 1978 and Gersch, 1974), Fourier Transformations (Cawley, 1984), Kalman Filters (Kalman, 1960), etc. In the above, the parameter estimation is achieved by the use of the following steps:

- Solving the differential equation of the given system,
- Approximating the solution of the differential equation by numerical methods,

- Estimating the unknown system parameters in time domain.

Roberto (1990) described a new method of characterising the time domain response of a second order system to step function by dynamical parameter estimation. The fundamental parameters were obtained by comparing the real input signal with a mathematical model using an optimization procedure. The parameters of the model were varied until an adequate match was obtained with the real input signal. This paper showed useful estimates of system parameters that could be produced in only a few iterations.

Pachter et al. (1994) proposed a new technique for parameter estimation of a second-order linear dynamical system model. This approach belongs to the class of classic deterministic identification methods, which assesses the model parameters based on obtaining a few characteristic values directly from the measured unit step response of the real investigated system e.g. rise time, theoretical transient time, position of the inflection point, some ordinates of unit step response and so on.

Goodwin (1997) studied parameter estimation algorithms for second order systems. Conventional approaches (Power series and Fourier transform) were tested for parameter estimation of a stereo camera system. Simulation results showed the estimation of parameters was best achieved with the polynomial least squares fitting algorithm.

Comparable methods have been used for parameter estimation of second order systems as in Meyer et al. (1967), Sundaresan et al. (1978), Hung and Clements (1982).

2.1.3 Parameter Estimation of Continuous Linear Systems

In recent years, polynomial series has been widely used in the analysis, approximation and parameter estimation of control systems [Horng and Chou, 1985]. Although most physical systems are of the continuous-time type, it is more convenient to use the samples of the input-output data and estimate the system model using a digital computer.

Stericker and Sinha (1993) derived algorithms for estimating the parameters of discrete-time model for a continuous-time system using the δ-operator from the samples of the input-output data either for a transfer function model or for a state-space model. Two approaches were considered in this paper to estimate a discrete-time model of a single-input single-output (SISO) system from the samples of the input-output data. In the first, they estimated the transfer function in terms of the δ-operator, while in the second they used a canonical state variable representation. Both of these methods can be readily extended to multivariable systems on the same line as by Sinha and Kszta (1983). Results of simulation indicated that the algorithms work well even if the sampling frequency is ten times the largest undamped natural frequency of the system to be identified, the approach using state-space model would give better results.

Zhihong (1995) developed a novel method to estimate unknown system parameters for SISO linear time invariant systems using functional approximation. The technique is demonstrated by an example which showed that the differential equation of a linear time invariant system can be approximated by a finite Taylor series and the unknown system parameters are then estimated by using the information from Laplace transforms of the input and the output.

Dewolf and Wiberg (1993) developed an ordinary differential equation technique via averaging theory and weak convergence theory to analyze the asymptotic behaviour of continuous-time recursive stochastic parameter estimators. This technique is an extension of Ljung's work in discrete time. The main purpose of this paper is to provide general conditions under which averaging theory can be applied to continuous-time parameter estimators. The theorem is used to prove convergence of continuous-time versions of both gradient and recursive prediction error parameter estimators. The theorem is also used to analyze the asymptotic dynamics of the simplest possible nontrivial parameter estimation problem, thus obtain an indication of the ordering of the dynamical behaviour of the four most common parameter estimators (Extended Kalman Filter, Extended Least Squares, Gradient Algorithm and Recursive Prediction Error).

2.1.4 Parameter Estimation of Dynamic Systems

Many applications in control systems require the estimation of one or more parameters of the plant. Among these are adaptive control algorithms based on plant identification. An example of this class of applications is the estimation of the coefficient of friction and the use of this estimate to generate an opposing force that cancels the friction (Friedland & Park, 1992); (Friedland & Mentzelopoulou, 1992). Friction, however, is not the only parameter that often needs to be estimated; time constants, natural frequency and gains are other parameters that one may desire to estimate for a variety of purposes. The theory developed in Friedland and Park (1992) for friction estimation can be generalised to address the problem of estimation any constant parameters in the system.

Friedland (1997) developed a new algorithm for estimating constant parameters in a dynamical system. The algorithm is a reduced-order observer, having two non-linear functions, one being the Jacobian of the other. If the dynamics of the system are afine in the parameters to be estimated, the error in estimation of these parameters satisfies a linear, time-varying homogeneous differential equation. By proper choice of the non-linear function in the observer, it is possible to achieve asymptotic stability of the estimation error. Although the algorithm was derived on the assumption that the process state can be measured. The method is being successfully used in several applications with real data such as rapid thermal processing application (Belikov et al. 1995).

Ameen (1998) presented a new approach in estimating the parameters of a dynamic system from an initial and relatively inaccurate state-space model. The proposed approach is based on determining an optimal control law that minimizes a quadratic performance index for a free state condition by means of solving a two-point boundary-value problem. The control input to both the model and the physical system were assumed to be the differences between their outputs. The analytical procedure proposes a correction matrix that is subtracted from the dynamics matrix of the initial model to yield the optimal value of the dynamics matrix. Numerical evaluation of this algorithm suggested that this method possesses high accuracy in terms of estimating the parameters of a dynamic system, it was applied to a second order system that had a modelling error of about 40%.

Yarimer and Virgin (1988) identified the non-linear dynamical roll motion equation for a ship using a standard recursive estimation technique (recursive Least Squares scheme). The time series were produced by 4^{th} order Runge-Kutta method. Noise was incorporated into the forcing term at each time step as random pulses from uniform distributions. However, from this investigation point of view, this method will lead to the use of different noise values as the Runge-Kutta scheme uses 4 evaluations of the derivative vector, which results in inconsistent and unpredictable integration errors.

2.2 Hierarchical models

There has been a great deal of research activity in the area of identification of distributed parameter systems over the past two decades. An extensive treatment of off-line schemes (e.g., output least square, equation error, etc.) together with a comprehensive survey of the literature can be found in the monograph by Banks and Kunisch [Banks & Kunisch, 1989]. In the case of on-line, or adaptive, schemes, the available literature is less extensive and more recent [Isermann et al, 1995].

The on-line methods give estimates recursively as the measurements are obtained within the time limit imposed by the sampling period. These include recursive projection algorithm [Baumeister et al, 1997], recursive Least Square algorithm [Glentis et al, 1990], on-line excitation algorithms [Ludwig et al, 1998], etc.

2.2.1 Parameter Estimation in Large-scale Systems

Parameter estimation of low order linear systems has been available for many years; the jump to the case of large-scale systems introduces special difficulties, which are associated with the high dimensionality of interconnected systems. Sage and his co-workers (Arafeh & Sage, 1974) have made important contributions in this area of identification for large-scale systems, such as the Maximum A Posterior approach (MAP). This approach is applicable to both state and parameter estimation, also the method is particularly attractive for parameter estimation since it is a sub-optimal method which converges to the correct parameter/state values near the final time and whilst this may not be acceptable for state estimation, it is usually so for parameter estimation.

An optimal state-estimation method for linear interconnected dynamical systems is the Multiple- Projection approach (Hassan et al. 1982). This technique has an excellent convergence behaviour since the estimator is computed in a number of iterations that is equal to the number of subsystems comprising the overall system (Sultan, 1988). Large-scale least-squares parameter estimation has been studied by Sultan et al. (1983,1985) by developing indirect algorithms using the decomposition-co-ordination approach. In 1988 Sultan, Hassan and Calvet developed two sequential decomposition algorithms for solving the problem of least-squares estimation in large-scale systems. The first algorithm based on the interaction-prediction approach for hierarchical optimization. The second algorithm treats the estimation problem as that of solving a large set of linear algebraic equations using Gauss-Seidal decomposition technique. A recursive version of each algorithm was also developed. The two algorithms were successfully applied to estimate the parameters of a multivariable discrete-time system. The Gauss-Seidal algorithm showed better convergence behaviour than the interaction-prediction algorithm.

Chemouil, Katebi, Sastry and Singh (1981) examined the use of maximum A Posterior (MAP) approach for parameter estimation in large-scale interconnected dynamical systems. The first work in this area was by Sage's group (Guinzy and Sage, 1974) who used the MAP approach in order to convert the estimation problem to an optimization problem, which could subsequently be solved using hierarchical techniques. In an earlier paper Chen and Perlis (1967) considered a hierarchical approach to optimal state estimation. Chemouil, Katebi, Sastry and Singh (1981) converted the approach of Chen and Perlis to the parameter estimation case and applied the co-state prediction approach (Hassan and Singh, 1977) to the parameter estimation problem. Their conclusions were that although the original approach of Sage was sub-optimal, in most cases this sub-optimality was at an acceptable level particularly when they took into account the low computation requirements of the approach.

2.2.2 Parameter Estimation using Hierarchical Structure

Linear time invariant multivariable systems may be considered as an interconnection of other subsystems, which may themselves be considered as an interconnection of other subsystems of lower order (Rosenbrock & Paugh, 1974).

Chang and Rae (1997) proposed a hybrid mapping parameter estimation method using the hierarchical structure in object-oriented coding. The hierarchical structure employed constructs of a low-resolution image. Then six mapping parameters for each object are then estimated from the low-resolution image and these parameter values are verified based on the displaced frame difference (DFD). Computer simulation showed that the estimation method gave a performance similar to that of conventional methods with greatly reduced computational complexity.

2.2.3 Parameter Estimation for Distributed Parameter Systems

Distributed time varying models are generally used for modelling dynamics, which are very complex. With the advent of powerful digital computers it has become possible to implement numerically intensive computations with much ease. Partial differential equations are often used to model the distributed parameter systems (Sunahara, 1982).

Mathew and Jha (1998) dealt with a functional approximation approach for identifying parameters from input output data for a class of systems. The operational properties of Hermit polynomials have been used in formulating an algorithm for identifying a class of time varying linear distributed parameter systems. Other authors have used different sets of orthogonal polynomials and piecewise constant functions such as Walsh functions (Rao & Sivakumar, 1975), Block pulse functions (Hsu and Cheng, 1982), Chebychev series (Horng and Tsai, 1986), Legendre polynomials (Mohan and Dutta, 1988), Laguerre polynomials (Ranganathan and Rajamani, 1987), etc. Mathew and Jha (1998) concluded that estimation of the parameters for the distributed linear time-varying system via Hermit polynomials, under noise free conditions, guaranteed convergence of estimate of the coefficients for Hermit series expansion.

2.3 Summary

This Chapter summarizes the current state of parameter estimation techniques in the literature with particular reference to on-line and off-line. The merits and problems associated with current studies, carried out using each approach, were reviewed. For the choice of parameter estimation method it is of interest to know the conditions under which stable and convergent system behaviour can be attained.

CHAPTER 3 - THREE PARAMETER ESTIMATION ALGORITHMS FOR TIME VARYING SYSTEMS

3.1 Introduction

To recognise trends embedded in patterns, a dynamical model of the system generating the patterns may be assumed. A second order model has a wide variety of patterns which can describe the short-term behaviour of complex physical and biological systems well. Apart from initial conditions, the output pattern of a simple second order system is completely defined by 3 parameters – damping ratio (ζ), natural frequency (ω) and external input (u).

In on-line applications, patterns are collected using devices taking measurements from the real system. The output patterns are fed to an analysis module, which classifies the input. Based on the output, a decision feedback action can make the system output produce the desired input to create a control system in which use is made of estimation techniques for adaptive purposes. Based on these estimates the adaptive module "controller" is adjusted to perform the desired input. The main purpose is to determine the parameters for stability in the modelled system.

Three algorithms are proposed and investigated in this study to estimate three categories of parameters: constant, variables with 1st order dynamics and variables with 2nd order dynamics, of an equivalent second order system from a given trajectory (in time or space) of the pattern. The algorithms combine successive 1st order filters of specified cut-off frequencies, to provide smoothing and higher order derivative filter estimation, with non-linear static parameter estimators.

Another major part of this investigation, the effect of measurement noise on the estimation accuracy is considered when the incoming trajectories were corrupted with random noise.

This chapter describes three hybrid methods for estimating parameters of equivalent second order systems. Section 3.2 gives a detail of breakdown of the proposed algorithms. Section 3.3 describes the categories of parameter sets. In section 3.4, a study of parameter estimation is considered. Section 3.5 gives a description of modelling a hierarchical system based on second order sub-systems.

3.1.1 Definitions of ω, ζ and u

Many physical problems are described by the solution of an initial value problem of the form:

$$\omega^{-2}.x''+2.\zeta.\,\omega^{-1}.\,x'+x=u \qquad (3.1)$$

$$x(0)=x_0, \qquad x'(0)=x'_0$$

Where ω is the natural frequency, ζ is the damping ratio, u is the input and x is the output of the system.. The derivation of the theoretical solution of equation 3.1 is given in Appendix A.

3.2 Derivations of Parameter Estimation Algorithms

Three algorithms are investigated in this study by simulation to estimate the parameters of an equivalent second order system for 3 categories of parameter sets: constants, variables with 1st order dynamics and variables with 2nd order dynamics. These three methods were developed using the following approaches:

- By using three sets of estimated 1st and 2nd time derivatives of the measured output of the system.
- By using two sets of estimated 1st, 2nd and 3rd time derivatives of the measured output of the system.
- By using a single set of estimated 1st, 2nd, 3rd and 4th time derivatives of the measured output of the system.

These are illustrated graphically in Figure 3.1.

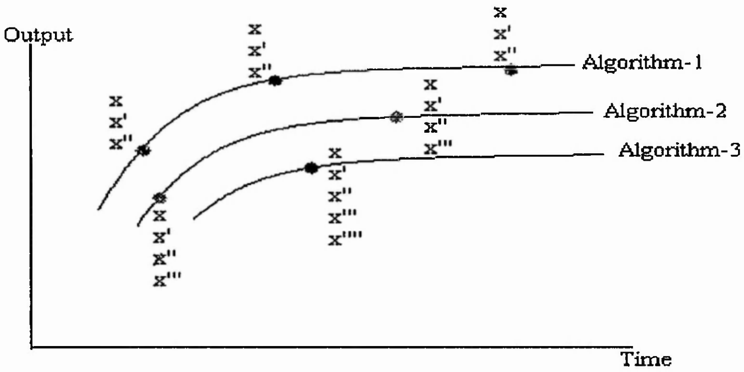

Figure 3.1: Three Parameter Estimation Algorithms

3.2.1 Algorithm-1: Three Points in x, x' and x''

This algorithm uses three points on the time trajectory to provide 3 simultaneous equations in the parameters. The algorithm uses a maximum of two derivatives of x to estimate these parameters. This is in contrast to later algorithms that use more. The algorithm is derived as follows:

Given the second order system:

$$\omega^{-2}.x''+2.\zeta.\,\omega^{-1}.\,x'+x=u \qquad \text{A1.1}$$

$$x(0)=x_0, \quad x'(0)=x'_0$$

Use three points of x, x' and x'' to produce:

$$\omega^{-2} x_1'' + 2.\ \zeta.\omega^{-1}.x_1' + x_1 = u \qquad \text{A1.2}$$

$$\omega^{-2} x_2'' + 2.\ \zeta.\omega^{-1}.x_2' + x_2 = u \qquad \text{A1.3}$$

$$\omega^{-2} x_3'' + 2.\ \zeta.\omega^{-1}.x_3' + x_3 = u \qquad \text{A1.4}$$

Subtract (A1.3) from (A1.2) and subtract (A1.4) from (A1.2) to produce (A1.5) and (A1.6) respectively:

$$\omega^{-2}.(x_1'' - x_2'') + 2.\ \zeta.\omega^{-1}.(x_1' - x_2') + (x_1 - x_2) = 0 \qquad \text{A1.5}$$

$$\omega^{-2}.(x_1'' - x_3'') + 2.\ \zeta.\omega^{-1}.(x_1' - x_3') + (x_1 - x_3) = 0 \qquad \text{A1.6}$$

Divide (A1.5) by $(x_1' - x_2')$ to produce:

$$\omega^{-2}.(x_1'' - x_2'') / (x_1' - x_2') + 2.\ \zeta.\omega^{-1} + (x_1 - x_2) / (x_1' - x_2') = 0 \qquad \text{A1.7}$$

Divide (A1.6) by $(x_1' - x_3')$ to produce:

$$\omega^{-2}.(x_1'' - x_3'') / (x_1' - x_3') + 2.\ \zeta.\omega^{-1} + (x_1 - x_3) / (x_1' - x_3') = 0 \qquad \text{A1.8}$$

Subtract (A1.8) from (A1.7) to produce:

$$\omega^{-2}.[(x_1'' - x_2'') / (x_1' - x_2') - (x_1'' - x_3'') / (x_1' - x_3')] + [(x_1 - x_2) /$$
$$(x_1' - x_2') - (x_1 - x_3) / (x_1' - x_3') = 0 \qquad \text{A1.9}$$

Use the following notations to simplify expressions:

$$\Delta 12 = (x_1 - x_2), \quad \Delta' 12 = (x_1' - x_2'), \quad \Delta'' 12 = (x_1'' - x_2'')$$
$$\Delta 13 = (x_1 - x_3), \quad \Delta' 13 = (x_1' - x_3'), \quad \Delta'' 13 = (x_1'' - x_3'') \qquad \text{A1.10}$$

By which we can get an expression from (A1.9) for estimated ω:

$$E\omega^2 = [\Delta''13.\Delta'12 - \Delta''12.\Delta'13]/[\Delta12.\Delta'13 - \Delta13.\Delta'12] \qquad \text{A1.11}$$

And an expression for estimated ζ using (A1.5) is obtained:

$$E\zeta = [-E\omega^{-2}.\Delta''12 - \Delta12]/[2.\,E\omega^{-1}.\Delta'12] \qquad \text{A1.12}$$

And finally an expression from (A1.2) for estimated u is obtained:

$$Eu = E\omega^{-2}.\,x_1'' + 2.\,E\zeta.\,E\omega^{-1}.\,x_1' + x_1 \qquad \text{A1.13}$$

3.2.2 Algorithm-2: Two Points and One Extra Derivative

This algorithm uses only 2 points from the system output but requires one more time derivative than Algorithm 1, to provide a continuous estimate of the 3 parameters. Algorithm-2 is derived to estimate ω, ζ and u using two values of estimated x, x′, x″ and x‴ as follows:

Given the second order system:

$$\omega^{-2}.x''+2.\zeta.\,\omega^{-1}.\,x'+x=u$$

Use two points of x, x′, x″ and x‴ to produce:

$$\omega^{-2}x_1'' + 2.\,\zeta.\omega^{-1}.x_1' + x_1 = u \qquad \text{A2.1}$$

$$\omega^{-2}x_2'' + 2.\,\zeta.\omega^{-1}.x_2' + x_2 = u \qquad \text{A2.2}$$

Subtract (A2.2) from (A2.1) to produce:

$$\omega^{-2}.(x_1'' - x_2'')/(x_1' - x_2') + 2.\,\zeta.\omega^{-1} + (x_1 - x_2)/(x_1' - x_2') = 0 \qquad \text{A2.3}$$

Divide (A2.3) by $(x_1' - x_2')$ to produce:

$$\omega^{-2}.(x_1'' - x_2'')/(x_1' - x_2') + 2.\ \zeta.\omega^{-1} + (x_1 - x_2)/(x_1' - x_2') = 0 \qquad \text{A2.4}$$

Differentiate (A2.4) with respect to time to eliminate $2.\zeta.\omega^{-1}$ and produce:

$$[\omega^{-2}[(x_1' - x_2').(x_1''' - x_2''')] - (x_1'' - x_2'')^2]/(x_1' - x_2')^2 + 0 + [(x_1' - x_2')^2 - (x_1 - x_2).(x_1'' - x_2'')]/(x_1' - x_2') = 0 \qquad \text{A2.5}$$

Use the following notations to simplify expressions:

$$\Delta 12 = (x_1 - x_2),\quad \Delta' 12 = (x_1' - x_2'),\quad \Delta'' 12 = (x_1'' - x_2'') \text{ and}$$

$$\Delta''' 12 = (x_1''' - x_2''') \qquad \text{A2.6}$$

By which an expression for estimated ω is obtained:

$$E\omega^2 = [(\Delta'12).(\Delta'''12) - (\Delta''12)^2]/[(\Delta'12)^2 - (\Delta 12).(\Delta''12)] \qquad \text{A2.7}$$

Use A2.5 to obtain an expression for estimated ζ:

$$E\zeta = [-E\omega^{-2}.\Delta''12/\Delta'12 - \Delta 12/\Delta'12]/[2.\ E\omega^{-1}] \qquad \text{A2.8}$$

And finally an expression for estimated u is obtained:

$$Eu = E\omega^{-2}.\ x_1'' + 2.\ E\zeta.\ E\omega^{-1}.\ x_1' + x_1 \qquad \text{A2.9}$$

3.2.3 Algorithm-3: One Point and Two Extra Derivatives

This algorithm uses a single time point but two further time derivatives compared to Algorithm-1. Algorithm-3 is derived to estimate ω, ζ and u using estimated higher derivatives of x as follows:

Given the second order system:

$$\omega^{-2}.x''+2.\zeta.\ \omega^{-1}.\ x'+x=u$$

Differentiate (A1.1) with respect to time to produce:

$$\omega^{-2}\ x''' + 2.\ \zeta.\omega^{-1}.x'' + x' = 0 \qquad \text{A3.1}$$

Divide (A3.1) by x″ to produce:

$$\omega^{-2}.\ x'''/x'' + 2.\ \zeta.\omega^{-1} + x'/x'' = 0 \qquad \text{A3.2}$$

Differentiate (A3.2) with respect to time to eliminate $2.\zeta.\omega^{-1}$ and produce:

$$\omega^{-2}.[(x''.\ x'''' - x'''^{2})\ /x''^{2}] + 0 + [(x''^{2} - x'.\ x''')\ /x''^{2}] = 0 \qquad \text{A3.3}$$

By which an expression for estimated ω is obtained:

$$E\omega^{2} = [x''.\ x'''' - x'''^{2}]/[x'.\ x''' - x''^{2}] \qquad \text{A3.4}$$

Using (A3.1) an expression for estimated ζ is obtained:

$$E\zeta = -[E\omega^{-2}x''' + x']/[2.\ E\omega^{-1}x''] \qquad \text{A3.5}$$

And finally an expression for estimated u is obtained:

$$Eu = E\omega^{-2}.x'' + 2.\ E\zeta\ E\omega^{-1}.\ x' + x \qquad \text{A3.6}$$

3.2.4 Derivation of Filter Values

Each of the input/output data sets of the second order system must be conditioned to remove the high frequency content, any initial transients due to initial conditions, and the bias within the signals. One method to reduce the effect of noise 'high frequency components' during continuous estimate of x, x', x'', x''' and x'''' is to use low pass filters. The half power cut-off frequency of these filters is G (which is 1/L in the equivalent inductor/resistor low pass 1st order filter, see Appendix B).

3.3 Categories of Parameter sets

3.3.1 Constant Parameters

This is the most tractable case, when the parameters (ω, ζ and u) of a second order system do not depend on time. Positive and negative values could be estimated using the algorithms derived above over a range of set values.

3.3.2 Variables with 1st Order Dynamics

For the variable parameters case with 1st order dynamics, the parameters of the second order system do depend on time i.e. time varying parameters. Different sets could be estimated using the algorithms derived earlier; these include an exponential, sine wave, ramp or quadratic. When the parameters are set to exponentially varying (say from 0 to A), in this case, instead of using the exponential ($A.e^{-t.A}$) term, it has been modelled by the term derived below. This will make the model easier to implement and faster to execute.

The term is derived as follows:

$$\text{Let } x = -A.e^{-At}+A$$

$$\text{For } t = 0, \quad \text{then} \quad x = 0$$

$$\text{For } t = \infty, \quad \text{then} \quad x = A$$

$$x' = A.\ e^{-At} = A - x$$

Simulation of the three algorithms derived above will be tested for a varying single parameter or for simultaneously varying multiple parameters.

3.3.3 Variables with 2nd Order Dynamics

Variable parameters with 2nd order dynamics may prove to be the most testing to be estimated. The parameters ω, ζ and u are time varying as outputs of second order systems to provide a rigorous test of the algorithms derived earlier. Each parameter sub-system will have its own 3 parameters: for ω, they are ωo, ζo and uo; for ζ, they are ωz, ζz and uz; and for u, they are ωu, ζu and uu. This will lead to the modelling of hierarchical system based on 2nd order sub-systems, which will be discussed in Section 3.5.

3.4 Problems Encountered during the Estimation Process

The goal of parameter estimation in this investigation is to determine values of different model parameters that provide the best fit to the actual system using the algorithms derived earlier. Simulation will be carried out to show:

- The algorithms can cope well for estimating the 3 categories of parameter sets and test the accuracy of the estimation algorithms when the simulated system is assumed to be free of noise.
- The effect on the accuracy of the estimation algorithms when noise is added to the simulated system.

3.4.1 Parameter Estimation Algorithms without Noise

If no noise is present and the simulated system output of the second order system is assumed to be exact, then the filter estimate of the simulated system output and its higher time derivatives can be produced with a very high accuracy. This is achieved by reducing the 2nd order system to 1st order (state space representation), then the system output (x) is evaluated numerically using fourth order Runge-Kutta method with an adaptive step size. To provide continuous estimate of the filter estimate of the simulated system output and its higher derivatives i.e. Ex, Ex$'$, Ex$''$, Ex$'''$ and Ex$''''$, 5 first order low pass filters are needed. The half power cut-off frequency of these filters is G (which is 1/L in as derived in Appendix B). Also these 5 1st order equation are computed numerically using the Runge-Kutta method.

3.4.2 Sensitivity of Parameter Estimation Algorithms to Noise

As mentioned in the aims, an important part of this study is to examine the performance and robustness of the parameter estimation algorithms when the incoming trajectories are corrupted with noise. To make the simulation more realistic simulation runs are made with some noise added, for the three categories of parameters; constants, variables with 1^{st} order dynamics and variables with 2^{nd} order dynamics. Noise is derived from a random number generator of mean zero.

3.4.3 Compensation for Lag and amplitude attenuation caused by Low gain:

Low gain (low cut-off frequency in the filter) is needed to filter out the noise, but low gain introduces lag and amplitude attenuation, which produce erroneous estimates of the parameters. A numerical lag compensation technique need to be introduced which selects progressively distant values of the higher derivatives. Also a numerical amplification factor technique needs to be introduced to overcome the attenuation problem. Simulation examples to investigate these problems and their solution will be presented in Chapters 4 and 5.

Figure 3.2 shows the arrangement of the desired parameter estimation system including the generation of the filter cascade to produce the higher derivative estimations and the three parameters ω, ζ and u estimated using one of the three algorithms.

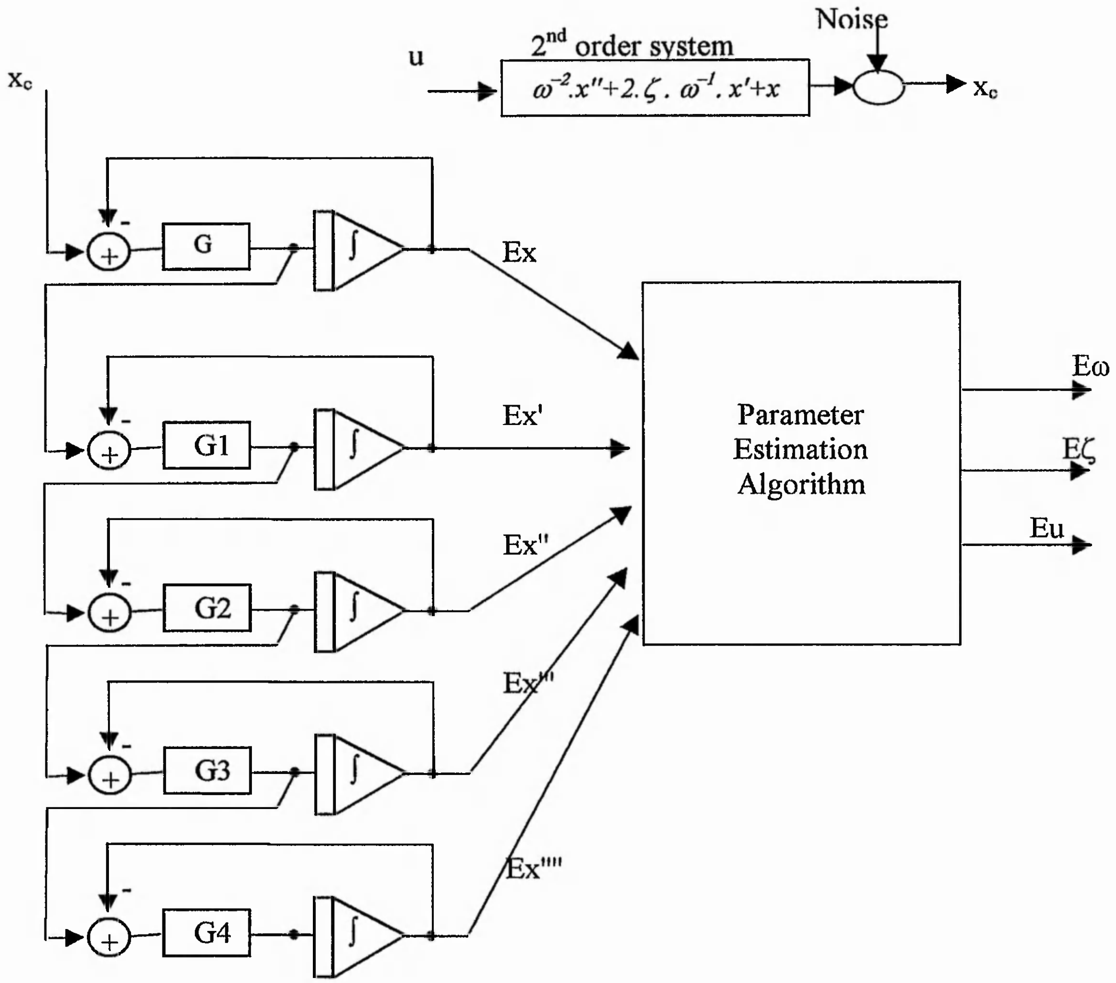

Figure 3.2: Parameter Estimation Algorithms schematic diagram

3.5 Parameter Estimation of Hierarchical Models

A hierarchical model based on second order sub-systems may be well formed by modelling the behaviour of each of the three parameters in the main second order system defined in 3.1.1. The parameters themselves are thus modelled as having their own unique time varying patterns, i.e. have dynamical behaviour. If these parameters are seen as time varying, then they in turn are submitted as input to another parameter estimation algorithm to estimate the parameters of their own dynamics. An example of a hierarchical model based on 12-second order sub-systems is shown in Figure 3.3. The objective of the process of hierarchical parameter estimation is to produce the values of the first level parameters of the sub-systems ω, ζ and u that generate the given trajectory of x through time.

The process of estimating the first level parameters of the sub-systems ω, ζ and u from the observed main system output poses different filtering criteria. Firstly, high gain (G) is needed to produce accurate estimation of time derivatives and hence the the three parameters ω, ζ and u of the main system. Secondly, low gain is needed to produce filter estimate of these and their higher derivatives i.e. Eω, Eω', Eω'', Eω''', and Eω''''; Eζ,, Eζ', Eζ'', Eζ''', and Eζ'''' ; Eu, Eu', Eu'', Eu''', and Eu''''. This low gain is needed because the estimation of the parameters ω, ζ and u is prone to noise and producing the derivatives is also prone to noise in itself. As the gain is reduced, compensation for lag and attenuation need to be introduced. When these derivatives are produced accurately then it is possible to achieve parameter estimates via the three algorithms described in section 3.2.

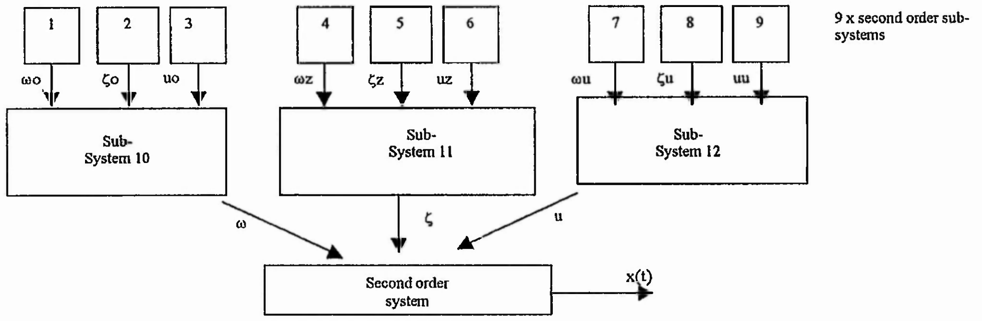

Figure 3.3: An example of a hierarchical model based on 13 2^{nd} order sub-systems.

CHAPTER 4 - SIMULATION SOFTWARE AND TEST STRATEGY

4.1 Simulation Structure

Simulation of the parameter estimation algorithms is implemented using Mathcad 8. This package is a versatile and powerful tool for computing the numerical solution of ordinary differential equations and is very suitable for the simulation of dynamical systems. Numerical system solutions within the Mathcad package will be used to solve the problem described in this thesis after minor modification.

4.1.1 Structure Details

Programs implemented in Mathcad will be presented in this chapter for the three categories of parameters: constant and variable with 1^{st} and 2^{nd} order dynamics. The three algorithms will be applied to each category to test their effectiveness.

Constant Parameters

Consider the case when the parameters of the second order system (defined in 3.1.1) are constants. The actual and estimated outputs of the system and their higher derivatives are computed using the built-in Runge-Kutta routine in Mathcad. This is achieved by reducing the second order system to a first order system as follows:

Let $x_1 = x$ .... (4.1)

$x_1' = x_2 = x'$ .... (4.2)

$x_1'' = x_2' = x''$ .... (4.3)

These expressions are substituted into equation 3.1.1, to get the following:

$x_1' = x_2$ .... (4.4)

$x_2' = -\omega^2.x_1 - 2.\zeta.\omega.x_2 + \omega^2.u$ .... (4.5)

Equations 4.4 and 4.5, as well as five 1st order low pass filters, provide a continuous estimate of x up to the fourth derivative i.e. Ex, Ex', Ex'', Ex''' and Ex''''. These seven simultaneous 1st order equations are contained in the D vector [Figure 4.1], which is solved using the Runge-Kutta routine for the interval t_0 and t_1. The half power cut-off frequency of these filters is G, [Appendix B]. Once the estimate of x and its time derivatives are determined, it will be possible to apply the three algorithms to estimate the constant parameters ω, ζ and u. The Mathcad program that further describes the estimation of ω, ζ and u is shown in Program 1 [Appendix C].

$$x := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad D(t,x) := \begin{bmatrix} x_2 \\ \left(-\omega^2 \cdot x_1 - 2\cdot\zeta\cdot\omega\cdot x_2\right) + \omega^2\cdot u \\ G\cdot\left(x_1 - x_3\right) \\ G\cdot\left[G\cdot\left(x_1 - x_3\right) - x_4\right] \\ G\cdot\left[G\cdot\left[G\cdot\left(x_1 - x_3\right) - x_4\right] - x_5\right] \\ G\cdot\left[G\cdot\left[G\cdot\left[G\cdot\left(x_1 - x_3\right) - x_4\right] - x_5\right] - x_6\right] \\ G\cdot\left[G\cdot\left[G\cdot\left[G\cdot\left[G\cdot\left(x_1 - x_3\right) - x_4\right] - x_5\right] - x_6\right] - x_7\right] \end{bmatrix}$$

x1 and x2 are the system state variables

x_3, x_4, x_5 are estimates of x, x' and x''

x_6 and x_7 are estimates of x''' & x''''

Figure 4.1: The D-vector, containing 7 simultaneous 1st order derivatives with initial conditions in vector x, for the constant parameters case.

When the three parameters of the 2nd order system defined in Section 3.1.1 are set as constants, the estimated output and their derivatives can be compared with the exact solutions [derived in Appendix A]. The results are shown in Figure 4.2. A comparison between the numerical solution of the system, with the filter estimate of system output and its higher derivatives, could also be made from these plots. To provide an accurate estimation of these, a higher gain value (cut-off frequency) was used. The higher the gain the closer the filter estimates were. The gain used ranged from 1,000 to 10,000.

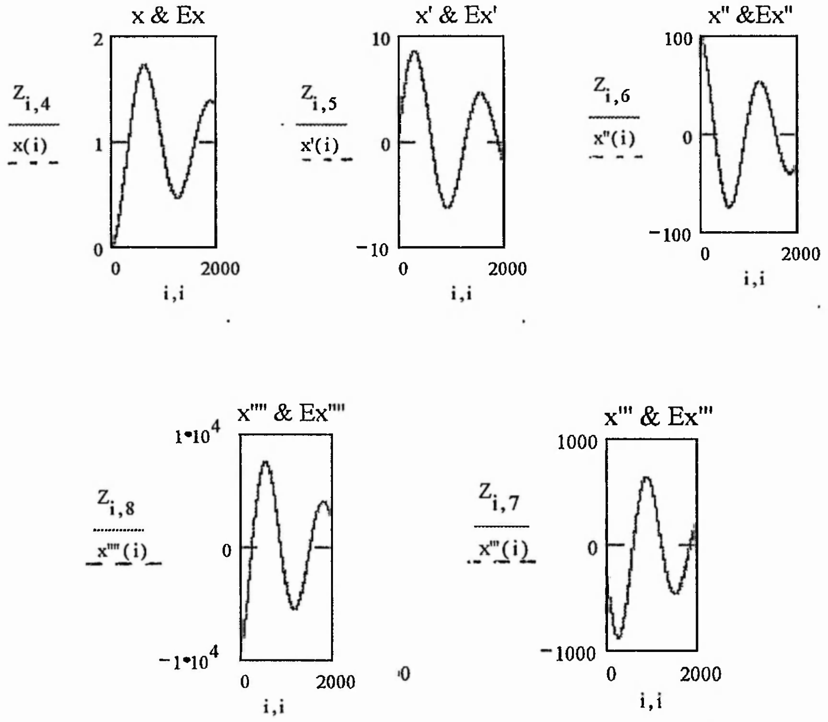

Figure 4.2: Exact and Estimated outputs and their derivatives for constant parameters (ω =10, ζ =0.1 and u =1; G=1000)

Parameters with 1st Order Dynamics

Consider the case when the three parameters (ω, ζ and u) of the 2nd order system (defined in section 3.1.1), are variable with 1st order dynamics; the actual and estimated outputs of the system, as well as their higher derivatives, are computed using the built-in Runge-Kutta routine in Mathcad. The derivatives vector D is modified to include the rate of change of ω, ζ and u. ω', ζ' and u' are set to an exponential ramp, often in the form e^{-t} , which, when coupled with a suitable initial condition, allows these parameters to increases over a wide range in a reasonably short time (about 1 to 3 sec). However, in this case, instead of using the exponential (e^{-t}) term for the solution, the derivative form is used [Figure 4.3]. This makes the model easier to implement and faster to execute.

Once the estimate of x and its time derivatives have been determined, it is possible to apply the three algorithms to estimate the variables with 1st order dynamics (ω, ζ and u). The Mathcad program used for estimating ω, ζ and u is shown in Program 2 [Appendix C].

$$x := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 2 \\ 0.1 \\ 1 \end{bmatrix} \quad D(t,x) := \begin{bmatrix} x_2 \\ -(x_8)^2 \cdot x_1 - 2 \cdot x_9 \cdot x_8 \cdot x_2 + (x_8)^2 \cdot x_{10} \\ G \cdot (x_1 - x_3) \\ G \cdot [G \cdot (x_1 - x_3) - x_4] \\ G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] \\ G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] \\ G \cdot [G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] - x_7] \\ 20 - x_8 \\ 0 \\ 0 \end{bmatrix}$$

x1 and x2 are the system state variables

$[(x_3, x_4)$ and$]$ x_5 are filter estimate of x, x' & x''

x_6 and x_7 are estimates of x''' & x''''

x_8 is ω, x_9 is ζ and x_{10} is u

Figure 4.3: The D-vector, containing 10 simultaneous 1st order derivatives with initial conditions in vector x, for variables parameters with 1st order dynamics.

The exact and filter estimate of outputs and their derivatives are shown in Figure 4.4 when the three parameters of the 2nd order system (defined in 3.1.1) have 1st order dynamics,

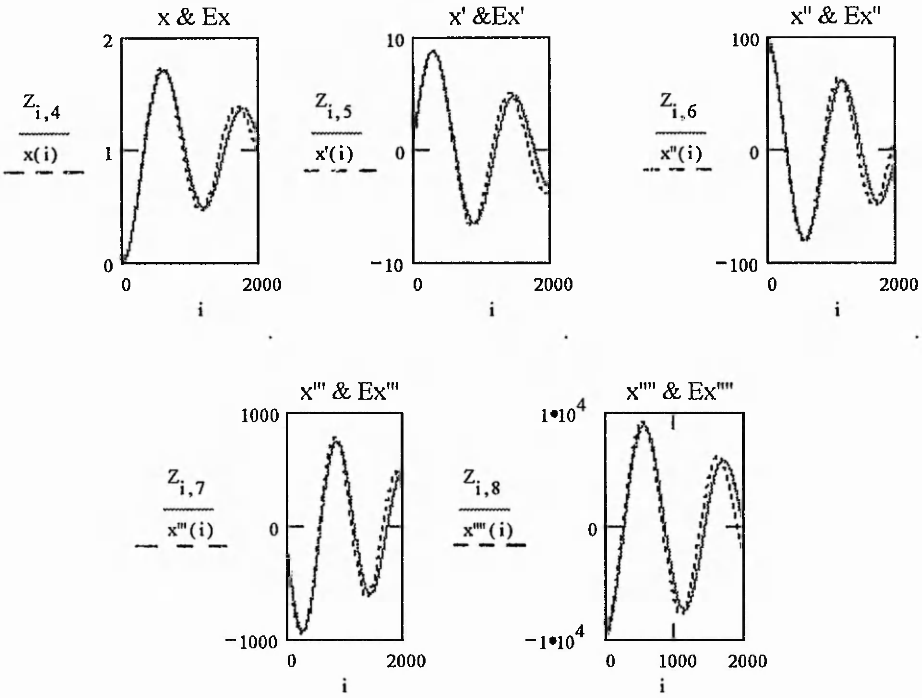

Figure 4.4: Exact and Estimated outputs and their derivatives for variables parameters with 1st order dynamics (ω: exponentially varying from 10 to 12, ζ =0.1 and u =1; G=5000).

Figure 4.4, shows how close the fit of the predictions is. However, this can only be achieved in a short period of time (from 1 to 3 seconds). As time increases the exact solution [derived in Appendix A] will become invalid, due to the assumption that the parameters were time independent.

Parameters with 2^{nd} Order Dynamics

Consider the case when the three parameters (ω, ζ and u) of the 2^{nd} order system (defined in Section 3.1.1) are variable with 2^{nd} order dynamics; the parameters (ω, ζ and u) are time varying as outputs of 2^{nd} order systems, to provide a rigorous test of the algorithms. Each parameter sub-system will have its own 3 parameters: for ω, they are ωo, ζo and uo; for ζ they are ωz, ζz and uz; and for u they are ωu, ζu and uu. The actual and filter estimate of the outputs of the system, and their higher derivatives are computed using the built-in Runge-Kutta routine in Mathcad. The derivatives vector D is modified to include the rate of change of ω, ζ and u (which are reduced from 2^{nd} order form to a 1^{st} order). The Mathcad program used for estimating ω, ζ and u with 2^{nd} order dynamics is shown in Program 3 [Appendix C]. The derivatives vector is shown in Figure 4.5.

$$x := \begin{bmatrix} 0 \\ 0 \\ 10 \\ 0 \\ 0.1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad D(t,x) := \begin{bmatrix} x_2 \\ -(x_3)^2 \cdot x_1 - 2 \cdot x_3 \cdot x_5 \cdot x_2 + (x_3)^2 \cdot x_7 \\ x_4 \\ -\omega o^2 \cdot x_3 - 2 \cdot \zeta o \cdot \omega o \cdot x_4 + \omega o^2 \cdot uo \\ x_6 \\ -\omega z^2 \cdot x_5 - 2 \cdot \zeta z \cdot \omega z \cdot x_6 + \omega z^2 \cdot uz \\ x_8 \\ -\omega u^2 \cdot x_7 - 2 \cdot \zeta u \cdot \omega u \cdot x_8 + \omega u^2 \cdot uu \\ G \cdot (x_1 - x_9) \\ G \cdot \left[G \cdot (x_1 - x_9) - x_{10} \right] \\ G \cdot \left[G \cdot \left[G \cdot (x_1 - x_9) - x_{10} \right] - x_{11} \right] \\ G \cdot \left[G \cdot \left[G \cdot \left[G \cdot (x_1 - x_9) - x_{10} \right] - x_{11} \right] - x_{12} \right] \\ G \cdot \left[G \cdot \left[G \cdot \left[G \cdot \left[G \cdot (x_1 - x_9) - x_{10} \right] - x_{11} \right] - x_{12} \right] - x_{13} \right] \end{bmatrix}$$

x_1 and x_2 are the system state variables

x_3 is ω

x_5 is ζ

x_7 is u

x_9 x_{10} and x_{11} are filter estimate of x, x' and x''

x_{12} and x_{13} are filter estimate of x''' and x'''' respectively

Figure 4.5: The D-vector, containing 13 simultaneous 1st order derivatives with initial conditions in vector x for parameters with 2nd order dynamics.

The exact and filter estimate of outputs and their derivatives, when the three parameters of the 2nd order system (defined in Section 3.1.1) are variables with 2nd order dynamics are shown in Figure 4.6.

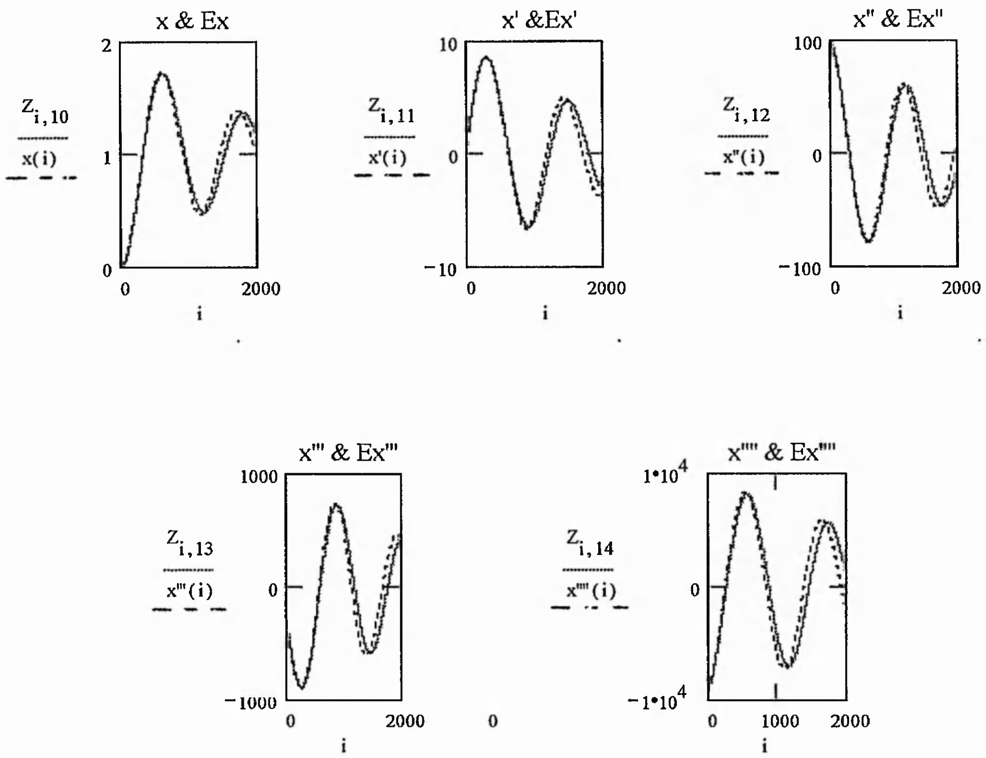

Figure 4.6: Exact and Estimated outputs and their derivatives for parameters with 2nd order dynamics (ω: ωo=10, ζo=0.3, uo=20; ζ: ωz=5, ζz=0.3, uz=0.1; u: ωu=10,ζu=0.3 and uu=1).

4.1.2 Initialization

The time trajectory of a second order model response differs for each set of initial conditions. The most standard is to start with the system at rest. There are sometimes limitations on the parameters of the second order system and initial conditions values, when the solutions of the differential equation do not exist, are required. On the other hand, when the parameters have 1st and 2nd order dynamics they should be used with suitable initial conditions to allow the parameters to increase over a good range.

4.1.3 Execution

In the simulation, each estimation algorithm can be executed separately or combined with other estimation algorithms. Within Mathcad, the execution of the programs is simply carried out by calling the estimation algorithm program. Mathcad then updates results in the worksheet window automatically.

4.2 Test Strategy

4.2.1 Varying a Single Parameter

Many applications in control systems require the estimation of one or more parameters of the system. Among these are adaptive control algorithms based on system identification. This class of applications includes the estimation of the coefficient of friction and the use of this estimate to generate an opposing force that cancels the friction (Freidland & Park, 1992).

The estimation algorithms derived in Chapter 3 can be generalised to address the problem of estimating time varying parameters. By modelling one parameter that varies with time and keeping the other two parameters constant, the estimation algorithms are expected to produce accurate predictions. However, by keeping 2 parameters constant, this system will only be valid for parameters with 1st and 2nd order dynamics. An example of modelling ω (varying with time exponentially), keeping ζ and u constant is shown in Figure 4.3 above. Program 2 [Appendix C] was used and simulation results are presented in Chapter 5.

4.2.2 Varying More than One Parameter

One example of a system described by more than one varying parameter is the DC servomotor modelled as second order system. In the design of a position and speed controller for the DC servomotor, two parameters have to be estimated: the damping ratio and natural frequency. As in the previous example, the algorithms derived in Chapter 3 will be examined in the case of time varying damping ratio and natural frequency. By modelling the two time-varying parameters, and keeping the third parameter constant, the estimation algorithms are expected to produce accurate predictions. The D-vector of modelling ω (exponentially varying from 10 to 12 in 1 sec.), ζ (exponentially varying from 0.1 to 0.9 in 1 sec) and keeping u constant at one is shown in Figure 4.7. When all three parameters vary at the same time at a rate comparable with the system dynamics, invalid estimations results are expected due to the breakdown of the assumptions in the derivation of the three algorithms, i.e. constant parameters. Program 2 [Appendix C] is used to produce the simulation results presented in Chapter 5.

$$x := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 10 \\ 0.1 \\ 1 \end{bmatrix} \qquad D(t,x) := \begin{bmatrix} x_2 \\ -(x_8)^2 \cdot x_1 - 2 \cdot x_9 \cdot x_8 \cdot x_2 + (x_8)^2 \cdot x_{10} \\ G \cdot (x_1 - x_3) \\ G \cdot [G \cdot (x_1 - x_3) - x_4] \\ G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] \\ G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] \\ G \cdot [G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] - x_7] \\ 12 - x_8 \\ 0.9 - x_9 \\ 0 \end{bmatrix}$$

x1 and x2 are the system state variables

$[(x_3, x_4)$ and$]$ x_5 are filter estimate of x, x' & x''

x_6 and x_7 are estimates of x''' & x''''

x_8 is ω, x_9 is ζ and x_{10} is u

Figure 4.7: The D-vector, containing 10 simultaneous 1st order derivatives, varying ω, ζ and keeping u constant for variable parameters with 1st order dynamics.

4.2.3 Sensitivity to Noise

Constant Parameters, Variables with 1st Order Dynamics and Variables with 2nd Order Dynamics

To prepare for system simulation with noise, the Runge-Kutta integration routine had to be abandoned as it uses 4 evaluations of the derivative vector resulting in different noise values being used in each evaluation. A simple iterative Euler integration algorithm with a single evaluation is used instead as shown in Figure 4.8.

Moving from top to bottom of Figure 4.8, the top expression represents the random number generator (n), feeding an integrator whose output N forms the noise source. The 2nd and 3rd expressions represent the state variable derivatives of the 2nd order system, generating the observed trajectory x1 and its derivative x2. Depending on the rate of change of the three parameters, the 4th to 9th expressions represents ω, ζ and u, respectively (Figure 4.8 is an example of ω varying exponentially from 10 to 30 rad/sec in 3 sec). The 10th expression shows a 1st order filter whose input is the noise corrupted trajectory x1+N with a gain G and integration step d. The 11th to 14th expressions show the successive first order filters that generate the progressively higher time derivatives of x, i.e. x', x'', x''' and x''''. A noise source of 10% of the nominal trajectory amplitude is shown in Figure 4.9. The Mathcad Program for this simulation is shown in Program 4 [Appendix C] and the simulation results for the three algorithms are discussed in Chapter 5.

$$\begin{bmatrix} N_{i+1} \\ x1_{i+1} \\ x2_{i+1} \\ \omega 1_{i+1} \\ \omega 2_{i+1} \\ \zeta 1_{i+1} \\ \zeta 2_{i+1} \\ u1_{i+1} \\ u2_{i+1} \\ Ex_{i+1} \\ Exd_{i+1} \\ Exdd_{i+1} \\ Extd_{i+1} \\ Exqd_{i+1} \end{bmatrix} := \begin{bmatrix} N_i + 1000 \cdot \left[\left(\mathrm{rnd}(n) - \frac{n}{2}\right) - N_i\right] \cdot d \\ x1_i + x2_i \cdot d \\ x2_i + \left[u1_i - 2 \cdot \zeta 1_i \cdot \left(\omega 1_i\right)^{-1} \cdot x2_i - x1_i\right] \cdot \left(\omega 1_i\right)^2 \cdot d \\ \omega 1_i + \left(30 - \omega 1_i\right) \cdot d \\ 0 \\ \zeta 1_i + 0 \\ 0 \\ u1_i + 0 \\ 0 \\ Ex_i + G \cdot \left(x1_i + N_i - Ex_i\right) \cdot d \\ Exd_i + G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i\right) - Exd_i\right] \cdot d \\ Exdd_i + G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i\right) - Exd_i\right] - Exdd_i\right] \cdot d \\ Extd_i + G3 \cdot \left[G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i\right) - Exd_i\right] - Exdd_i\right] - Extd_i\right] \cdot d \\ Exqd_i + G4 \cdot \left[G3 \cdot \left[G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i\right) - Exd_i\right] - Exdd_i\right] - Extd_i\right] - Exqd_i\right] \cdot d \end{bmatrix}$$

Figure 4.8: Iterative Euler integrator for the noise source, trajectory generator, the three parameters and time higher derivative estimators.

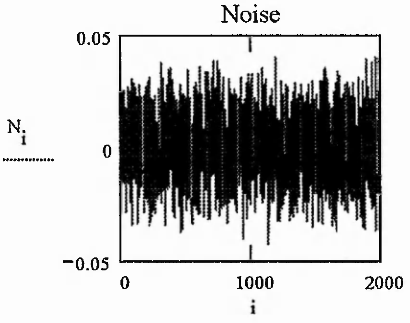

Figure 4.9: An example of random noise of 0.1

Compensation for Lag and Amplitude Attenuation caused by Low gain:

A low gain (low pass filter) is needed to filter out the noise. However, a low gain introduces lag and amplitude attenuation, which produce erroneous parameter estimations. To compensate for this lag, the values of the filter estimate for higher time derivatives are selected by 'shifting' indices. Furthermore, to compensate for the attenuation, the filter estimation of the simulated output and its higher time derivatives are multiplied by the absolute value of the gain factor (A) of the low pass filter as shown in Equation 4.6 below.

$$A := \sqrt{1+\left(\frac{f}{fc}\right)^2} \qquad \text{.................................(4.6)}$$

Where f is the frequency of the main system ($f = \omega/2.\pi$), and fc is the cut-off frequency of the low pass filter ($fc = G/2.\pi$).

For example, a second order system with f = 2 Hz (i.e. $\omega = 4.\pi$), $\zeta = 0.001$ and u =1 and solved in a duration of 3 second using an integration step of 3/T (T is the total number of points, set to 2000 to provide accurate integration results). If the cut-off frequency of the low pass filter is twice f (i.e. $G = 8.\pi$), the results show a lag of about 25 time steps at each stage of the higher derivative estimators, and an attenuation of about 10%, the exact value is A=1.118 from equation 4.6. Figure 4.10 shows sample results of the output compared with its filter estimate, and their 1st time derivatives. Table 4.1a,b shows the lag (L points) caused when the value of G is set to 20 at various values of ω and when the value of w is set to 20 at various values of G respectively.

a.

ω	5	10	15	20	25	30
Lag (L)	34	32	30	29	25	22

b.

G	1000	500	300	100	50	20
Lag (L)	0	1	2	6	13	31

Table 4.1.a, b: Lag caused by different values of ω and G (simulation time = 3 seconds).

The results of introducing of the lag/amplitude compensation for this example are shown in Figure 4.11. After the initial transient period, the filter estimation of the simulated system and its 1st derivative begin to have a close fit to the actual simulated system output and its 1st derivative.

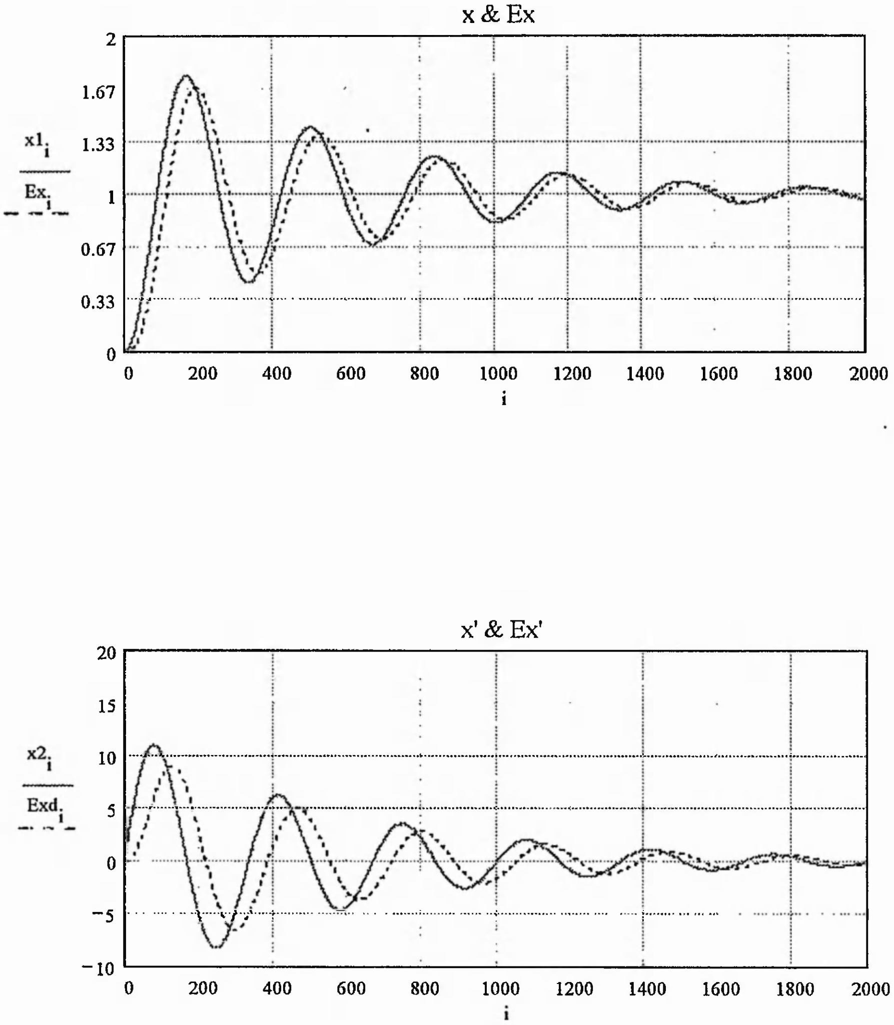

Figure 4.10: Results of the example before introducing the lag/amplitude compensation to the filter estimation of the simulated system and its 1st derivative.

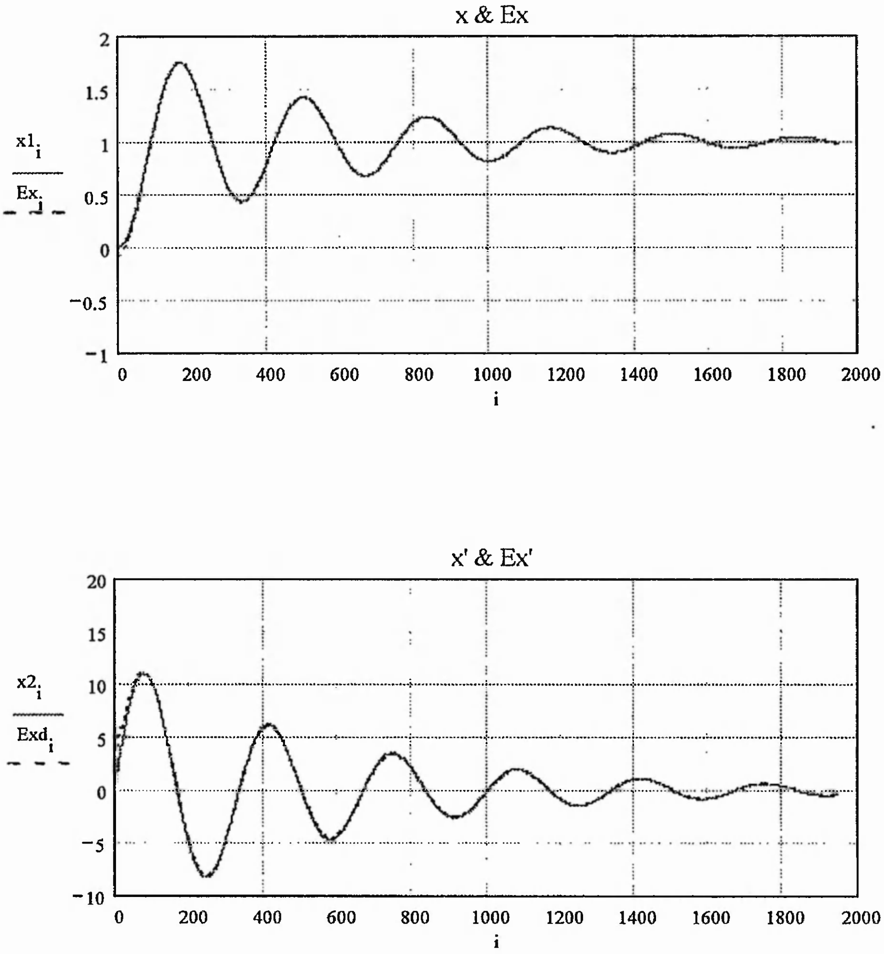

Figure 4.11: Results of the example, after introducing lag and amplitude compensation to the filter estimation of the simulated system and its 1st derivative.

Higher Derivative Estimation:

Noise added to the trajectory undergoes successive filtering as it passes through the stages of the higher time derivatives. Low values of G correspond to low cut-off frequencies in the filter, resulting in smoother derivative estimates. Figures 4.12.a & 4.12.b below show 2 such cases when G=20 and 30 respectively. The heavier filtering effect of G=20 on the derivative trajectories is quite noticeable. The noise level is 10% of the nominal value of x at 1.0.

The effects of noise on the estimation accuracy of the algorithms in the simulation are presented in Chapter 5.

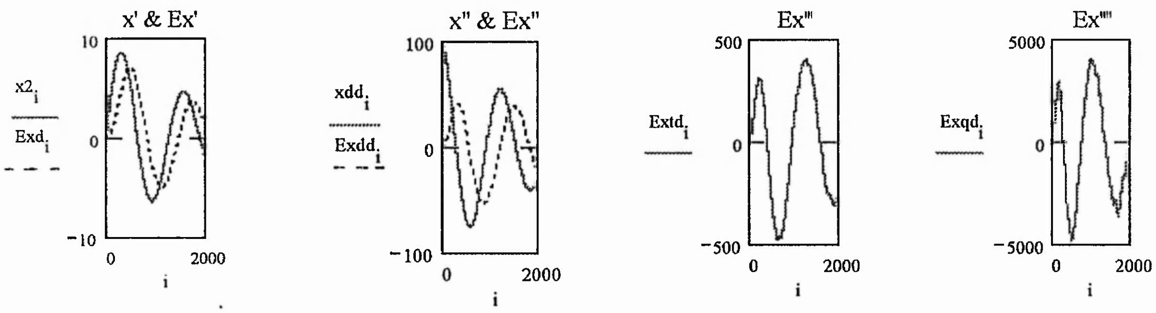

Figure 4.12.a: Higher derivative estimation: n=0.1, G=20.

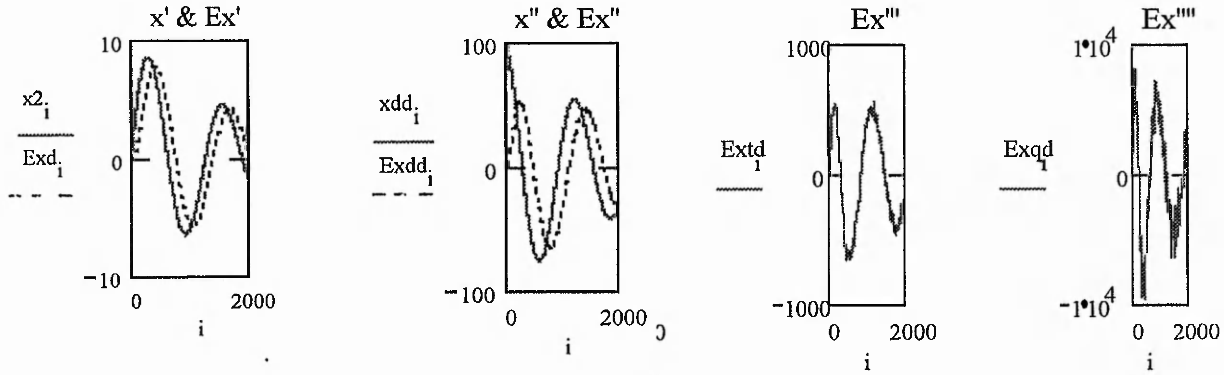

Figure 4.12.b: Higher derivative estimation without compensation: n=0.1, G=30.

4.3 Parameter Estimation of Hierarchical Models

The hierarchical model based on 2nd order sub-systems can be seen when the parameters of the main second order system (defined in 3.1.1) are variables having 2nd order dynamic behaviour. In section 4.1.1 the variables with 2nd order dynamics ω, ζ and u were modelled in the D-vector and the numerical solutions is obtained using the built-in 4th order Runge-Kutta scheme. Deep within the system lie the first level parameters (ωo, ζo, uo; ωz, ζz uz; and ωu, ζu uu) which are constants. In this section, these first level parameters will be estimated using the three algorithms derived in Chapter 3.

To prepare for the simulation of these, the iterative Euler integration (with a single evaluation) is used after estimating the three parameters of the main system (as described in 4.1.1). The Euler method is chosen for the following reasons:

1) Producing estimates of the three parameters of the main system are prone to noise. The first level parameter estimation depends upon the accuracy of the filter estimate of these three parameters and their higher derivatives. Low gain is required and consequently lag problems will occur.

2) Shift indices are required and the built-in Runge-Kutta routine can not be modified to introduce these.

3) The use of a programmed Euler routine permitted better control of the simulation.

4) When the number of rows in the D-vector is increased (adding the filter estimate) in the Runge-Kutta method, the tolerance has to be reduced to complete the simulation within reasonable time duration. This decreased the accuracy of estimations.

4.3.1 Hierarchical Model for The ω Subsystem

Consider the hierarchical model simulation structure for the ω subsystem with 2nd order dynamics. The objective of this simulation is to provide an estimate of the first level parameters (ωo, ζo and uo) of the sub-system ω, from the simulated main system output, using the three algorithms derived in Chapter 3.

The estimate of ω (Eω) of the main system is obtained using the three algorithms described in 4.1.1 (also Program 3, Appendix C). To provide an estimate of the first level parameters, the iterative Euler routine is used to generate the filter estimate of the parameter Eω (FEω) of the main 2nd order system and its higher derivatives i.e. FEω', FEω'', FEω''', and FEω''''. Moving from the top to the bottom of Figure 4.13: the 1st expression generates the filter estimate of the estimated parameter Eω (FEω) of the main system (for example, Eω3 estimates the parameter ω of the main system using Algorithm-3). The 2nd to 5th expressions show the successive first order filters that generate the time derivatives of Eω,, i.e. FEω', FEω'', FEω''', and FEω''''.

The values of the gains (G, G1, G2 to G4) are assigned different letters to allow a number of gain values; the integration step d which is equal to the simulation time divided by the total number of points. Once the time derivatives have been determined, it is possible to apply the three algorithms to estimate the first level parameters (ωo, ζo and uo). The Mathcad program that further describes the process of estimating the first level parameters of the sub-system ω is shown in Program 5 [Appendix C]. Simulation results will be presented in Chapter 5.

$$\begin{bmatrix} FE\omega 3_{i+1} \\ FE\omega d3_{i+1} \\ FE\omega dd3_{i+1} \\ FE\omega td3_{i+1} \\ FE\omega qd3_{i+1} \end{bmatrix} := \begin{bmatrix} FE\omega 3_i + G\cdot\left(E\omega 3_i - FE\omega 3_i\right)\cdot d \\ FE\omega d3_i + G1\cdot\left[G\cdot\left(E\omega 3_i - FE\omega 3_i\right) - FE\omega d3_i\right]\cdot d \\ FE\omega dd3_i + G2\cdot\left[G1\cdot\left[G\cdot\left(E\omega 3_i - FE\omega 3_i\right) - FE\omega d3_i\right] - FE\omega dd3_i\right]\cdot d \\ FE\omega td3_i + G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(E\omega 3_i - FE\omega 3_i\right) - FE\omega d3_i\right] - FE\omega dd3_i\right] - FE\omega td3_i\right]\cdot d \\ FE\omega qd3_i + G4\cdot\left[G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(E\omega 3_i - FE\omega 3_i\right) - FE\omega d3_i\right] - FE\omega dd3_i\right] - FE\omega td3_i\right] - FE\omega qd3_i\right]\cdot d \end{bmatrix}$$

Figure 4.13: Iterative Euler integration for the filter estimate of Eω of the main system and its higher derivative estimators.

4.3.2 Hierarchical Model for The ζ Subsystem

The method described in section 4.3.1 is repeated here. Estimations of the first level parameters (ωz, ζz and uz) of the sub-system ζ are conducted as shown in program 5 [Appendix C]. Estimates of ζ (Eζ) of the main system are obtained as before. Moving from the top to the bottom of Figure 4.14: the 1st expression generates the filter estimate of the estimated parameter Eζ (FEζ) of the main system (for example, Eζ3 estimates the parameter ζ of the main system using Algorithm-3). The 2nd to 5th expressions show the successive first order filters that generate the time derivatives of Eζ,, i.e. FEζ', FEζ'', FEζ''', and FEζ'''.

$$\begin{bmatrix} FE\zeta 3_{i+1} \\ FE\zeta d3_{i+1} \\ FE\zeta dd3_{i+1} \\ FE\zeta td3_{i+1} \\ FE\zeta qd3_{i+1} \end{bmatrix} := \begin{bmatrix} FE\zeta 3_i + G\cdot\left(E\zeta 3_i - FE\zeta 3_i\right)\cdot d \\ FE\zeta d3_i + G1\cdot\left[G\cdot\left(E\zeta 3_i - FE\zeta 3_i\right) - FE\zeta d3_i\right]\cdot d \\ FE\zeta dd3_i + G2\cdot\left[G1\cdot\left[G\cdot\left(E\zeta 3_i - FE\zeta 3_i\right) - FE\zeta d3_i\right] - FE\zeta dd3_i\right]\cdot d \\ FE\zeta td3_i + G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(E\zeta 3_i - FE\zeta 3_i\right) - FE\zeta d3_i\right] - FE\zeta dd3_i\right] - FE\zeta td3_i\right]\cdot d \\ FE\zeta qd3_i + G4\cdot\left[G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(E\zeta 3_i - FE\zeta 3_i\right) - FE\zeta d3_i\right] - FE\zeta dd3_i\right] - FE\zeta td3_i\right] - FE\zeta qd3_i\right]\cdot d \end{bmatrix}$$

Figure 4.14: Iterative Euler integration for the filter estimate of Eζ of the main system and its higher derivative estimators.

4.3.3 Hierarchical Model for The u Subsystem

The methods described in section 4.3.1 are repeated here. Estimations of the first level parameters (ωu, ζu and uu) of the sub-system u are conducted as shown in program 5 [Appendix C]. Estimates of u (Eu) of the main system are obtained as before. Moving from the top to the bottom of Figure 4.15: the 1st expression generates the filter estimate of the estimated parameter Eu (FEu) of the main system (for example, Eu3 estimates the parameter u of the main system using Algorithm-3). The 2nd to 5th expressions show the successive first order filters that generate the time derivatives of Eu,, i.e. FEu', FEu'', FEu''', and FEu''''.

$$\begin{bmatrix} FEu3_{i+1} \\ FEud3_{i+1} \\ FEudd3_{i+1} \\ FEutd3_{i+1} \\ FEuqd3_{i+1} \end{bmatrix} := \begin{bmatrix} FEu3_i + G\cdot\left(Eu3_i - FEu3_i\right)\cdot d \\ FEud3_i + G1\cdot\left[G\cdot\left(Eu3_i - FEu3_i\right) - FEud3_i\right]\cdot d \\ FEudd3_i + G2\cdot\left[G1\cdot\left[G\cdot\left(Eu3_i - FEu3_i\right) - FEud3_i\right] - FEudd3_i\right]\cdot d \\ FEutd3_i + G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(Eu3_i - FEu3_i\right) - FEud3_i\right] - FEudd3_i\right] - FEutd3_i\right]\cdot d \\ FEuqd3_i + G4\cdot\left[G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot\left(Eu3_i - FEu3_i\right) - FEud3_i\right] - FEudd3_i\right] - FEutd3_i\right] - FEuqd3_i\right]\cdot d \end{bmatrix}$$

Figure 4.15: Iterative Euler integration for the filter estimate of Eu of the main system and its higher derivative estimators.

CHAPTER 5 – SIMULATION RESULTS AND DISCUSSION

This chapter presents the results of extensive simulation work and investigations into the parameter estimation algorithms proposed and developed in Chapters 3 and 4. It details each stage of the simulation, describing notable features in the results. The overall test strategy is as described in Chapter 4.

5.1 Estimating Constant Parameters

This section covers parameter estimation of an equivalent second order system using the three algorithms derived in section 3.2 when the parameters are assumed to be constant. Program 1 [Appendix C] is used to get an estimate of the three parameters ω, ζ and u, corresponding to set values of $\omega=10$, $\zeta=0.1$ and $u=1$, for a duration of 3 seconds throughout.

5.1.1 Algorithm 1

Algorithm 1 - Without noise

This algorithm uses three points on the time trajectory to provide 3 simultaneous equations. In each equation the values of its 1st and 2nd time derivatives are obtained from the filter cascade. D is an n-element vector-valued function containing the first derivatives of the system and three 1st order low pass filters to provide a continuous estimate of x, x' and x''. The half power cut-off frequency of these filters is G (which is 1/L in the equivalent inductor/resistor low pass 1st order filter) and is set to 1000 to provide a compromise between good response to systems dynamics and noise filtering. The built-in Runge-Kutta routine was used to find the solution of x for the interval t0 and t1, which ranged between 1 to 3 seconds and T (total number of points) has the value of 2000 for an integration step of t1/T (0.00015). The separation between the 3 points used for the three simultaneous equations clearly affects the accuracy of the estimation. The further they are apart the larger are the differences between the variables and the more accurate the estimation. A separation of 30 to 70 points was found to give good accuracy and rapid convergence to the correct values. Running Program 1 [Appendix C] for $\omega = 10$, $\zeta = 0.1$ and $u = 1$ produced the estimations of ω, ζ and u shown in Figure 5.1.

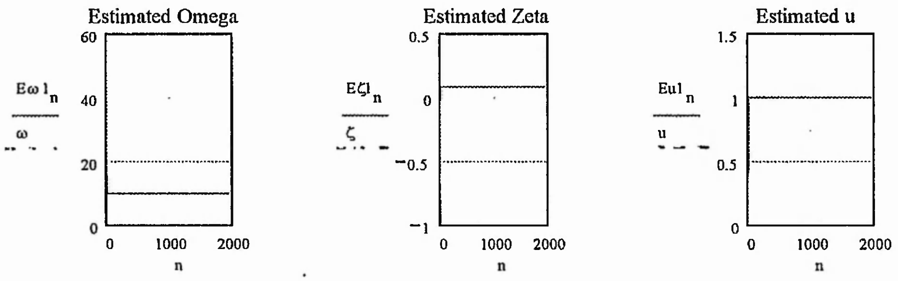

Figure 5.1: Estimated constant parameters ω , ζ and u using Algorithm-1.

Algorithm 1 - Sensitivity to noise

The highest time derivative used in this algorithm is x'' and thus unaffected by errors in estimating x''' and x''''. A low gain (low pass filter) was needed to filter out the noise, but this introduces lag and amplitude attenuation (Table 4.1), which produce erroneous estimates of the parameters. To compensate for this lag, the values of the higher derivatives were selected by shifting indices. For example, produce erroneous estimates of the parameters. To compensate for this lag, the values of the higher derivatives were selected by shifting indices. For example, using the same values for ω, ζ and u as in Figure 5.1, a gain G of 20, and running the simulation of the algorithm without noise for a duration of 3 seconds, produced a lag of about 32 points. The amplitude factor in this example was found to be 1.118. The derivatives estimate was multiplied by this value. Simulation results for this example, before and after introducing the lag/amplitude compensation, are shown in Figures 5.2.a and 5.2.b respectively.

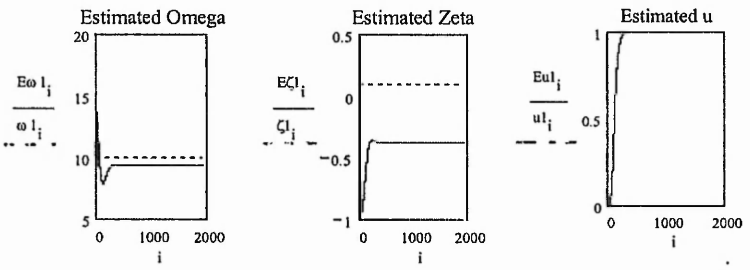

Figure 5.2.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, using Algorithm-1, before compensation with G =20, Lag (L) =0 and no noise (n=0).

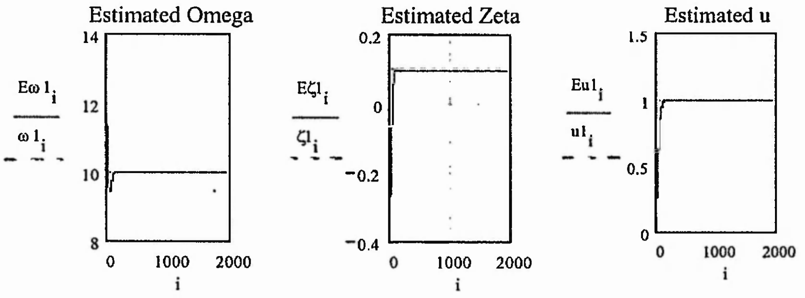

Figure 5.2.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, using Algorithm-1, after compensation with G =20, Lag (L) =32 and no noise (n=0).

The algorithm was tested for increasing noise levels of 0.1%, 1% and 10% of the nominal trajectory amplitude (x). With gains set to 20, noise was injected into the observed trajectory, producing the results shown in Figures 5.3.a, 5.3.b and 5.3.c respectively.

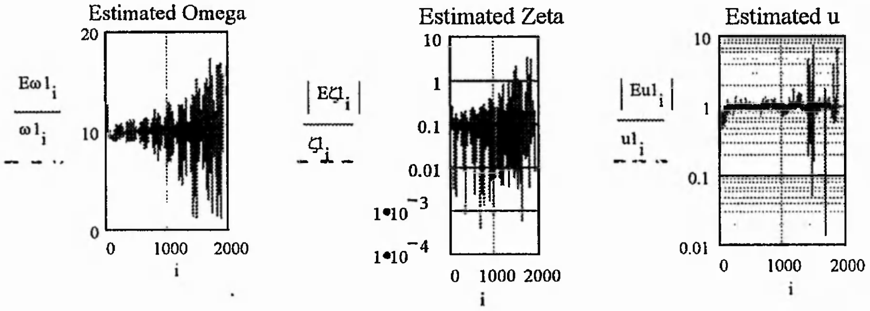

Figure 5.3.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) = 32.

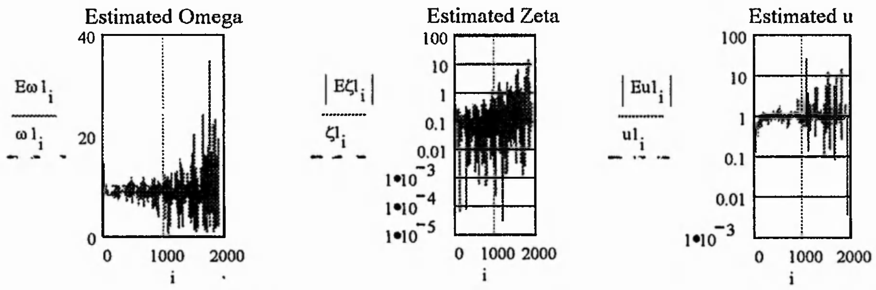

Figure 5.3.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-1 with G = 20, Lag (L) = 32.

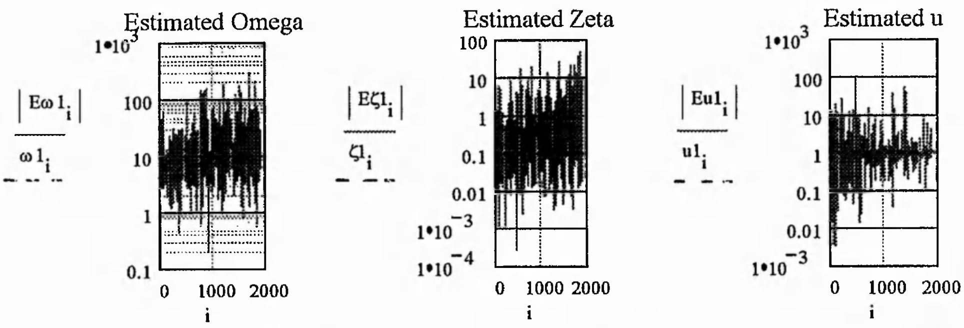

Figure 5.3.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) = 32.

5.1.2 Algorithm 2

Algorithm 2 - Without Noise

This algorithm uses only 2 points from the systems output but requires a further time derivative to provide a continuous estimate of the 3 parameters. The filter cascade is increased by one to yield the extra derivative. The two sets of x, x', x'' and x''' are used in the solution to the 2 simultaneous equations to give the estimated ω, ζ and u. The separation between the two points can be larger than the previous 3-points case to provide better accuracy; a situation helped by the fact that the parameters are constant. Algorithm 2 (in 3.2) was applied and Program 1 for ω = 10, ζ = 0.1 and u = 1 [Appendix C] was run to produce the results shown in Figure 5.4.

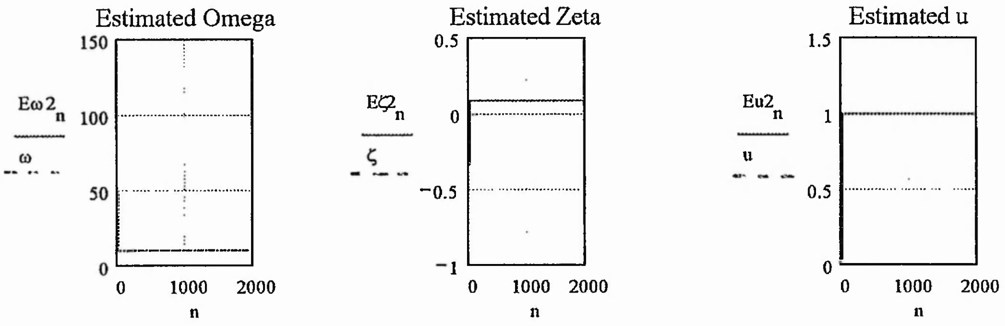

Figure 5.4: Estimated constant parameters ω, ζ and u using Algorithm-2.

Algorithm 2 - Sensitivity to Noise

The simulation presented in 5.1.1 was repeated using Algorithm 2 (section 3.2). Noise (as described in section 5.1.1 Algorithm 1 - sensitivity to noise) was added to the simulated state variable output. By applying the same set values used in Algorithm 2, the results in Figures 5.5.a, 5.5.b and 5.5.c were obtained.

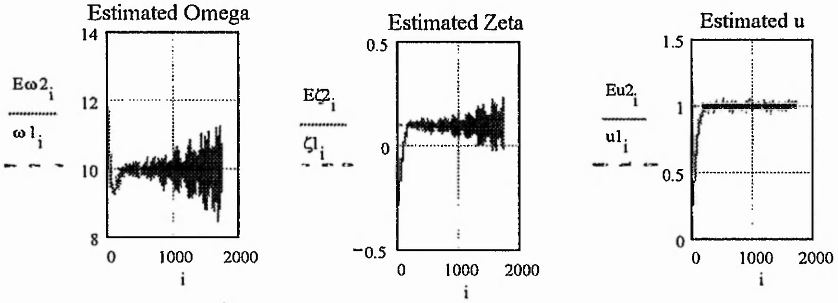

Figure 5.5.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-2 with G = 20, Lag (L) = 32.

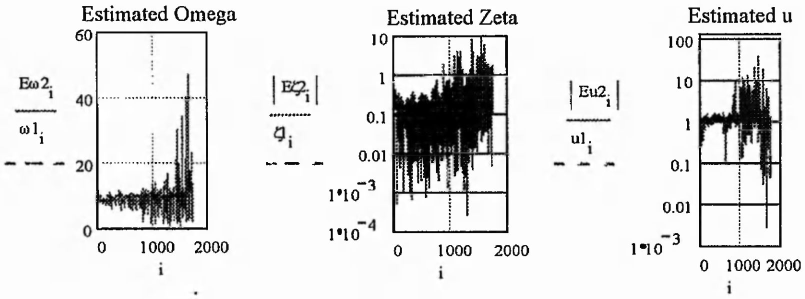

Figure 5.5.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-2 with G = 20, Lag (L) = 32.

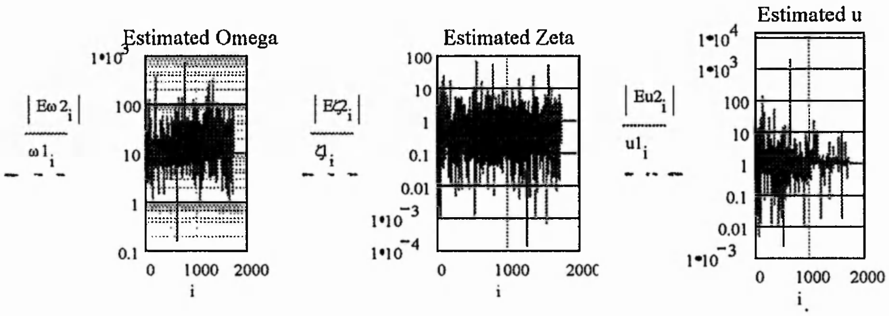

Figure 5.5.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-2 with G = 20, Lag (L) = 32.

5.1.3 Algorithm 3

Algorithm 3 - Without noise

This algorithm uses a single time point but two more time derivatives than Algorithm-1. The filter cascade is again increased by one to provide a continuous estimate of the 4th time derivative x''''. The separation problem now disappears altogether to provide a continuous estimate of all parameters at each point on the trajectory. Program 1 [Appendix C] was run and the results of the estimation are presented in Figure 5.6. Fast and accurate convergence is clearly indicated.

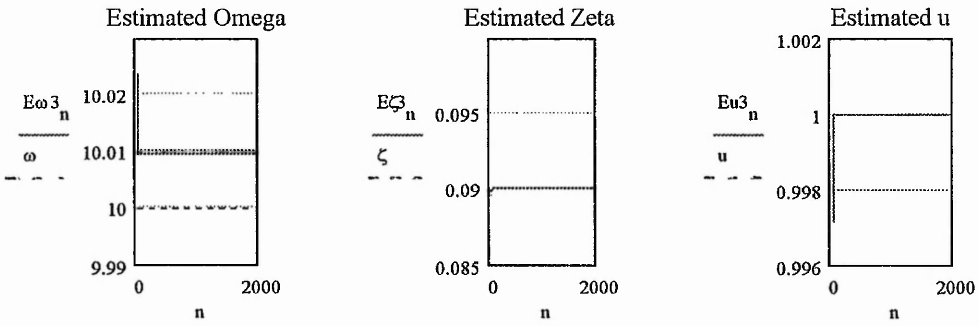

Figure 5.6: Estimated constant parameters ω =10, ζ =0.1 and u =1 using Algorithm-3 with G= 1000.

Algorithm 3 - Sensitivity to noise

The simulation presented in 5.1.1 was repeated again using Algorithm 3. Noise (as described in section 5.1.1) was added to the simulated first state variable (output). By applying the same set values, the results shown in Figures 5.7.a, 5.7.b and 5.7.c were produced.

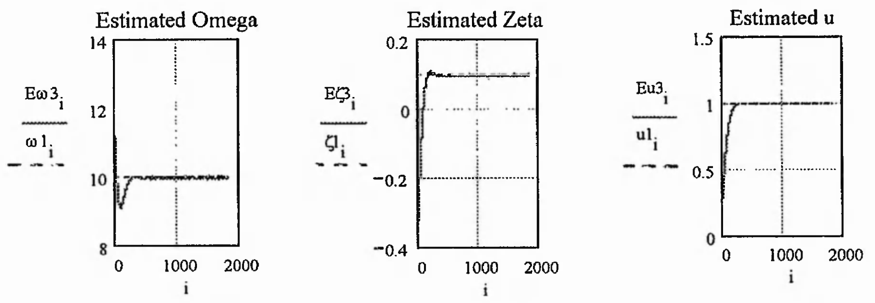

Figure 5.7.a: Estimated constant parameters ω=10, ζ=0.1 and u=1, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) = 32.

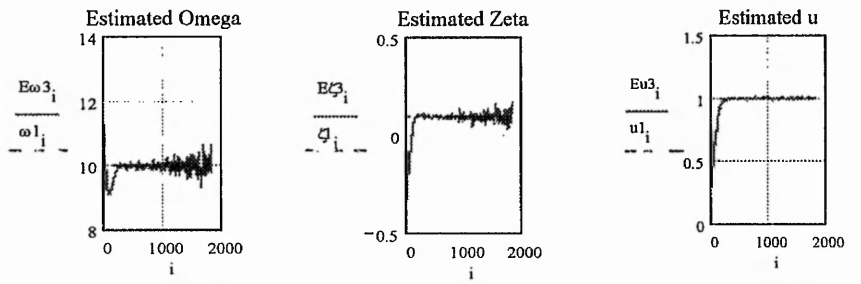

Figure 5.7.b: Estimated constant parameters ω=10, ζ=0.1 and u=1, and medium noise (n = 0.01) using Algorithm-3 with G = 20, Lag (L) = 32.

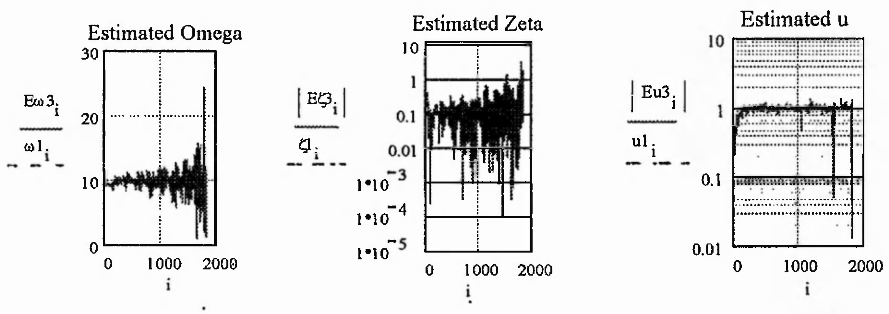

Figure 5.7.c: Estimated constant parameters ω=10, ζ=0.1 and u=1, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) = 32.

5.1.4 Discussion

Estimated values for constant dynamic parameters were close to the desired set values for all three algorithms. The derived algorithms estimated ω, ζ and u for a good range of values: ω from 1 to 20, ζ between ±0.01 and ±1, and u between ±0.5 and ±40. Accurate estimates were produced. Estimation errors decreased as ω increased, particularly when ζ was less than 0.5, where oscillation provided large values and wide variation in the variables. The differences between the (simulated) system time derivatives (x, x' and x'') and their estimates from the filter cascade depended on G (the cut-off frequency). A high value of G provided a more accurate estimation of derivatives but made the algorithms prone to noise and vice versa. Another disadvantage of high G from the simulation point of view is that simulation time increased considerably due to the integration routine adapting to ever-smaller steps. The algorithms provided progressively faster convergence with Algorithm-3 being the fastest to converge.

Furthermore, when noise (derived from a random number generator with a mean of zero) was added to the simulated system output, the lag index shift (L) clearly had a direct effect on the estimation accuracy. It was found that best results were obtained when L was about 32 with a gain (G) of 20. The low value of G used to filter out the noise is related to the value of the natural frequency (ω) of the system (G should be at least equal to four times ω). However, the relationship between G and ω means that when the gain is lower than 4 times ω the accuracy of estimating ω is affected.

The three algorithms were tested for increasing noise levels: low, 0.1%, medium, 1% and high, 10% of the nominal trajectory amplitude. Heavy smoothing is provided by low values of G (cut-off frequencies, G=20), which introduces increasing lags in the successive stages of the higher derivative estimators and amplitude attenuation. Numerical lag and amplitude amplification techniques were introduced which selected progressively distant values of the higher derivatives and calibrated the reduction in the magnitude; these, together with the increased smoothing applied to the higher derivatives gave Algorithm-3 the leading edge in combating noise.

5.2 Estimating Parameters with 1st Order Dynamics

The parameters were set to vary with time. The derivative vector D was modified to include the rate of change of ω, ζ and u. ω' was set to (A-x) to give an exponential profile as derived in section 4.1.1. When coupled with a suitable initial condition this allowed ω to increases over a good range, e.g. 10 to 30 rad/sec in a reasonably short duration (3 sec). ζ' and u' were similarly arranged to vary from 0.01 to 0.9, and 1 to 10 respectively, both in 3 sec. Program 2 was used to get an estimate of the three parameters ω, ζ and u using the three algorithms.

5.2.1 Algorithm 1

Algorithm 1 – Varying a Single Parameter at a Time -Without noise

Three points are used in this algorithm, which immediately raises the issue of the effect of the size of the gap between successive points. Parameters do not change much when the gap is small. Thus, the 3 simultaneous equations are less erroneous in assuming (implying) constant parameters. The larger the gap the less accurate this assumption is and therefore estimation is less accurate. Running Program 2 [Appendix C] when ω was varied from 10 to 30 (setting $\zeta = 0.1$ and $u = 1$); ζ was varied from 0.1 to 0.3 (setting $\omega = 10$ and $u = 1$); and u was varied from 1 to 10 (setting $\omega = 10$ and $\zeta = 0.1$). G was set to 1000 and simulation time to 3 seconds in all three cases. . Figure 5.8.a shows the case when the points were separated by 40 time-steps. Figure 5.8.b shows the case when the points are separated by 100 time-steps, the errors in the estimation are very clear and indicates the extent by which the assumption for constant parameters has broken down in the simultaneous equations. There is a counter effect however, a small gap also means that the variables do not change very much and therefore differences will be prone to numerical errors. It is therefore expected that an optimum separation gap exists.

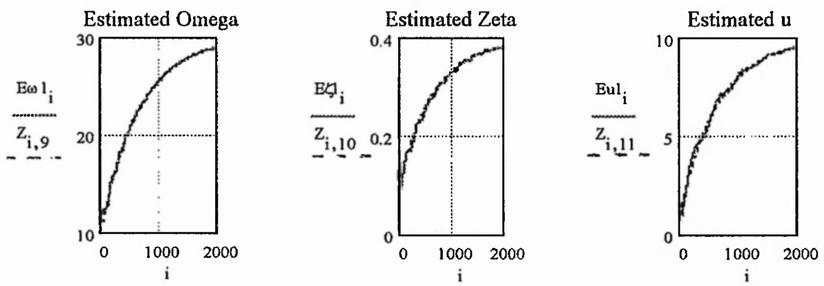

Figure 5.8.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-1 (separation time step = 40 points).

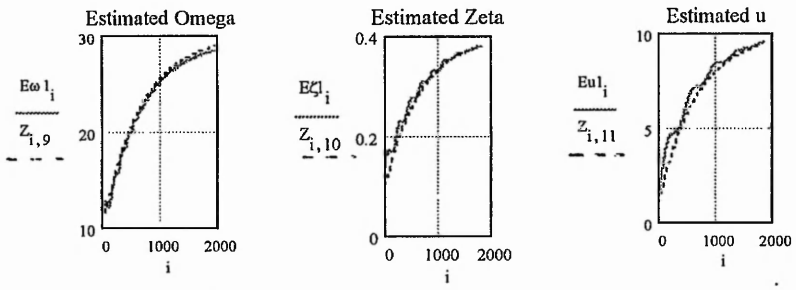

Figure 5.8.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-1 (separation time step =100 points).

Algorithm 1 – Varying more than One Parameter at a Time -Without noise

Simulation results were also obtained using Algorithm 1 for parameters with 1st order dynamics, when more than one parameter was varied. Figure 5.9.a, shows the results obtained for estimating: ω increasing from 10 to 30 in 3 sec, ζ increasing from 0.1 to 0.3 in 3 sec. and keeping u constant at 1. Figure 5.9.b, shows the results obtained for estimating: ω increasing from10 to 30 in 3 sec., u increasing from 1 to 10 in 3 sec. and keeping ζ constant at 0.001.

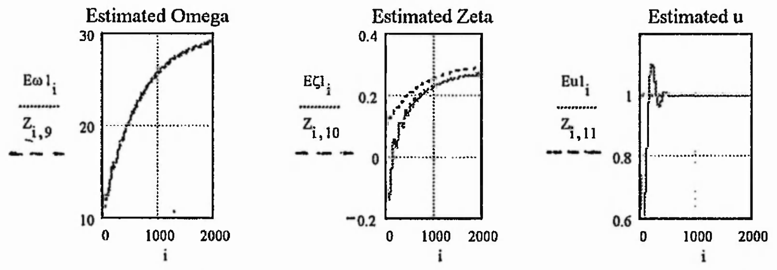

Figure 5.9.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-1.

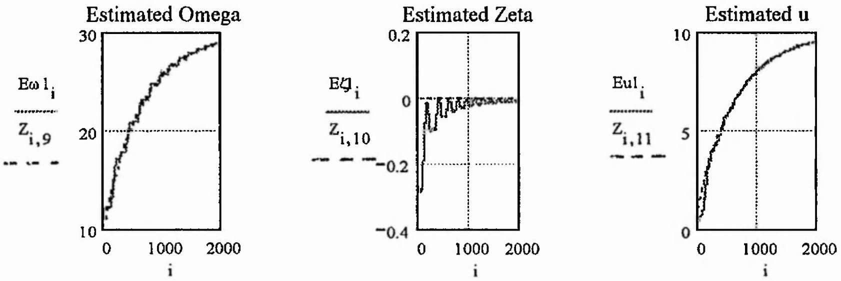

Figure 5.9.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-1.

Algorithm 1 - Sensitivity to noise

Noise added to the trajectory undergoes successive filtering as it passes through the stages of the higher derivatives, immediately raising the issue of the effect of low gain. Low values of the gain G to filter out the noise are needed as discussed in section 5.1.4. By setting the value of G to less than 4 times the value of ω will produce erroneous results in the estimation of ω. An example of estimating ω as it increases from 10 to 30 and keeping ζ and u constants at 0.1 and 1 respectively produces the erroneous result shown in Figure 5.10. From the simulation point of view it was found that when the value of G was set to 20 and L to 32, ω starts to diverge from the set value after it reaches the value 5. This is also true in the constant parameter case when the set value of ω is greater than 5 when G is 20.

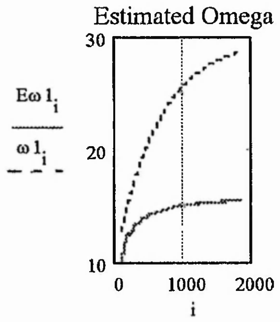

Figure 5.10: Estimated parameters with 1st order dynamics ω, keeping ζ and u constants using Algorithm-1 with G =20, Lag (L) =32 and no noise (n = 0).

Running Program 4 [Appendix C] when ω was varied from 10 to 12 (setting ζ = 0.1 and u = 1); ζ was varied from 0.1 to 0.3 (setting ω = 10 and u= 1); and u was varied from 10 to 12 (setting ω = 10 and ζ =0.1), for noise levels of 0.1% (low) and 10% (high), the results shown in Figures 5.11.a, 5.11.b were produced.

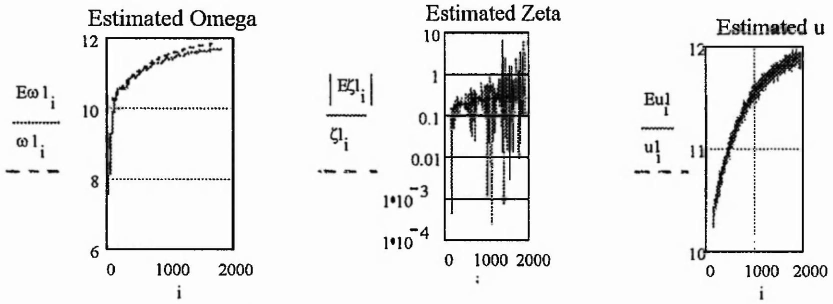

Figure 5.11.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) = 32.

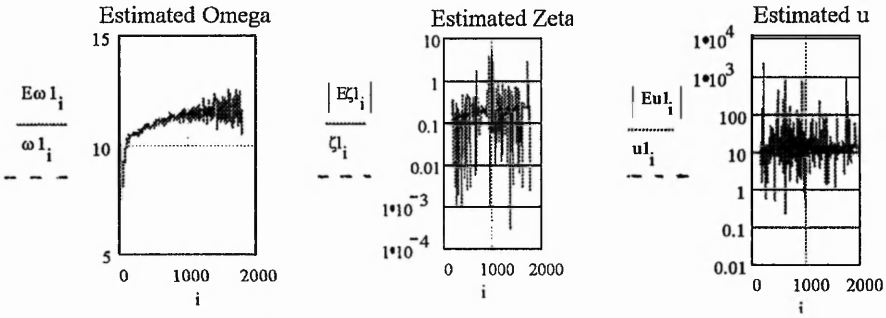

Figure 5.11.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) = 32.

5.2.2 Algorithm 2

Algorithm 2 – Varying a Single Parameter at a Time - Without Noise

Only two points (occupying half the time duration of 3 points) are used here so the constant parameters assumption is less pronounced. This also allows a wider gap to reduce numerical difference errors. Two simultaneous equations are used with two values of x, x', x'' and x''' instead of three values of x, x' and x''. Program 2 [Appendix C] was run to produce the results shown in Figure 5.12: where 5.12.a and 5.12.b correspond to gaps of 40 and 100 time-steps respectively. Overall, the 40 time-steps gap gives better performance despite large initial errors. The noticeable reduction in these errors for the larger gap case is probably due to the reduction in numerical errors as the variables have larger values.

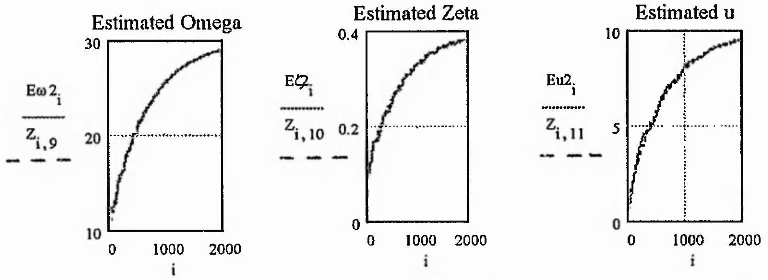

Figure 5.12.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-2 (separation time step =40 points).

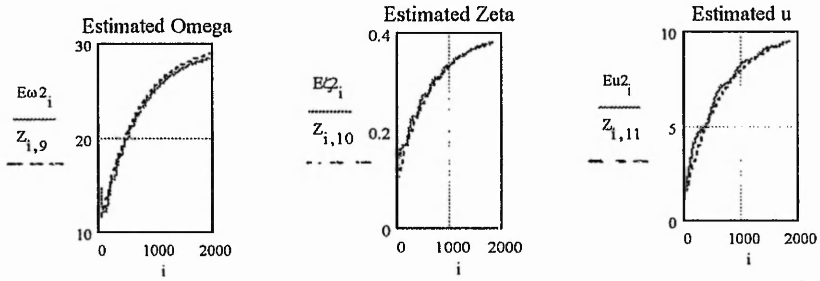

Figure 5.12.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-2 (separation time step = 100 points).

Algorithm2 – Varying more than one Parameter at a Time – Without Noise

The simulation shown in section 5.2.1 for more than one varying parameter was used to estimate the same set values of ω, ζ and u used earlier, but using Algorithm 2 rather than Algorithm 1. Simulation results obtained are shown in Figures 5.13.a and 5.13.b.

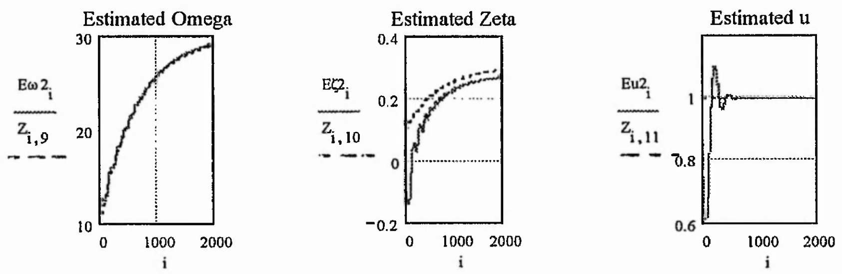

Figure 5.13.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-2.

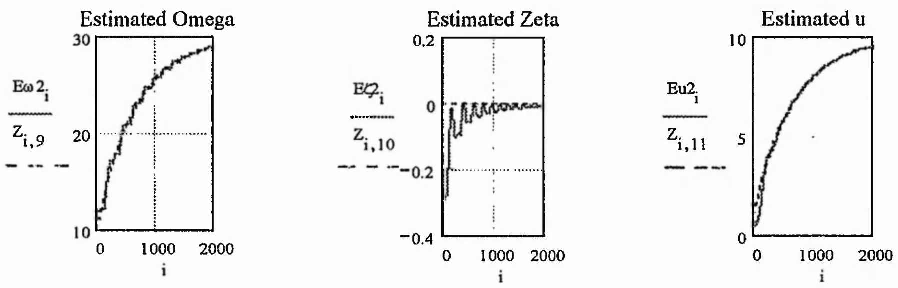

Figure 5.13.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-2.

Algorithm 2 - Sensitivity to Noise

The simulation presented in 5.2.1 was repeated using Algorithm 2. Noise (as described in section 5.2.1) was added to the simulated first state variable (output). By applying the same set values, the results shown in Figures 5.14.a, 5.14.b were produced.

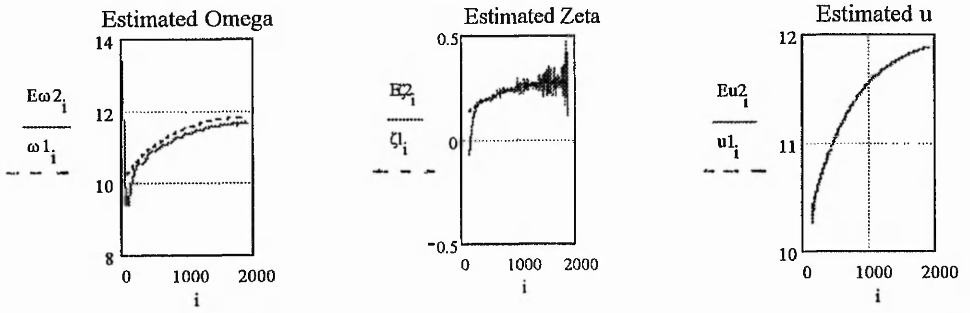

Figure 5.14.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n =0.001) using Algorithm-2 with G = 20, Lag (L) = 32.

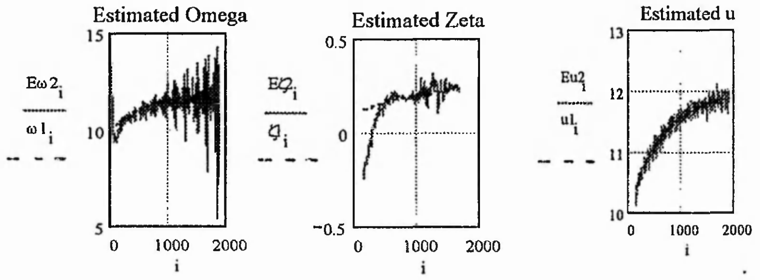

Figure 5.14.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n =0.1) using Algorithm-2 with G = 20, Lag (L) = 32.

5.2.3 Algorithm 3

Algorithm 3 – Varying a Single Parameter at a Time - Without Noise

Just one point is used here but with one extra derivative, x''''. As there are no gap effects, the accuracy of the parameter estimation depends entirely on the estimation accuracy of the time derivatives. These are obtained as by-products of the filter cascade and the accuracy of their instantaneous values will depend on the gain. Larger gain, i.e. smaller filter time constant and shorter delay, gives more accurate derivatives. However, larger gain also means higher cut off frequency and therefore less immunity to high frequency noise. Figures 5.15.a and 5.15.b show two cases with gains of 1000 and 300 respectively.

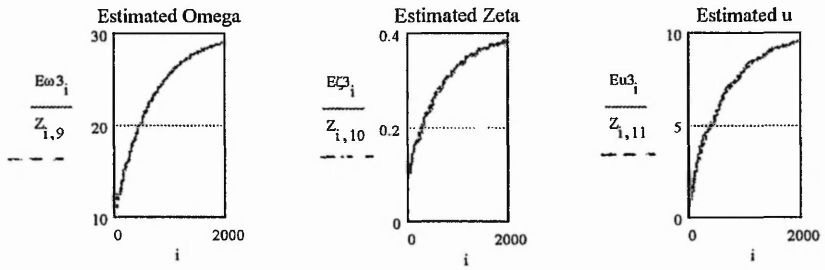

Figure 5.15.a: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-3 with G =1000.

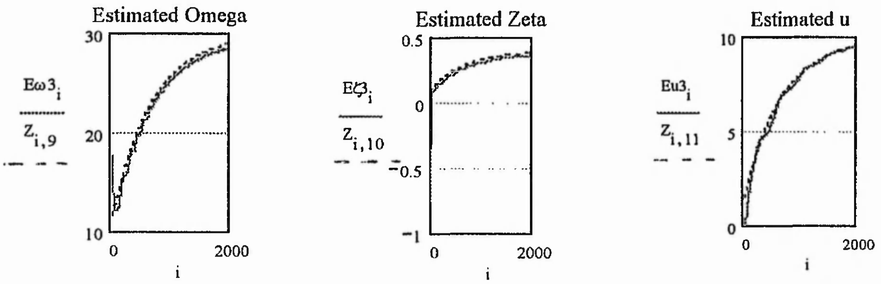

Figure 5.15.b: Estimated parameters with 1st order dynamics for ω, ζ and u using Algorithm-3 with G =300.

Algorithm 3 – Varying more than one Parameter at a Time – Without Noise

The simulation used in 5.2.1 was repeated using Algorithm 3 to estimate the same set values of variables ω, ζ and u used earlier. Simulation results are shown in Figures 5.16.a and 5.16.b.

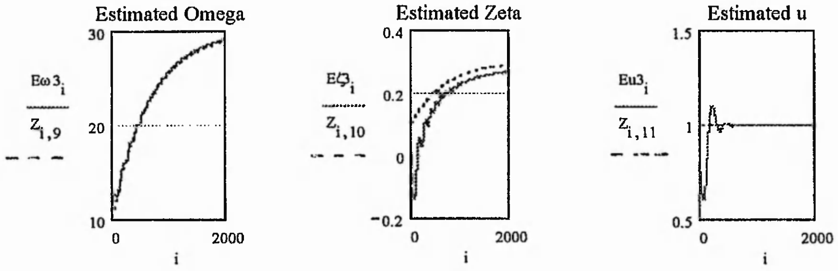

Figure 5.16.a: Estimated parameters with 1st order dynamics for ω and ζ but keeping u constant using Algorithm-3.

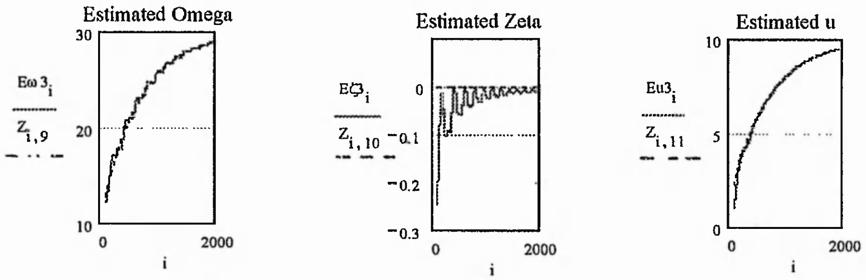

Figure 5.16.b: Estimated parameters with 1st order dynamics for ω and u but keeping ζ constant using Algorithm-3.

Algorithm 3 - Sensitivity to Noise

The simulation presented in 5.2.1 was repeated again using Algorithm 3. Noise (as described in section 5.2.1) was added to the simulated first state variable (output). By applying the same set values, the results shown in Figures 5.17.a, 5.17.b were produced.

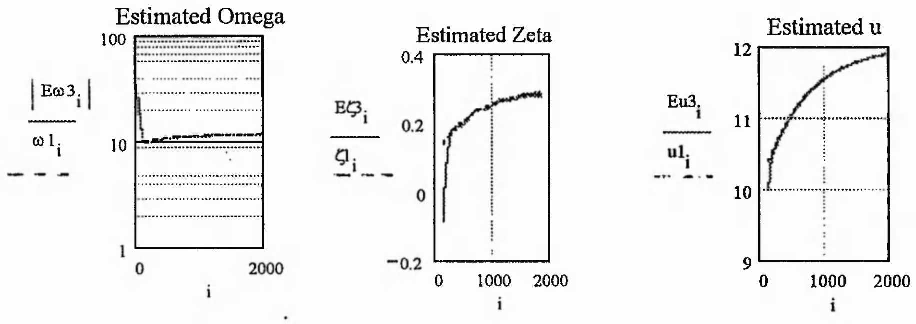

Figure 5.17.a: Estimated parameters with 1st order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) = 32.

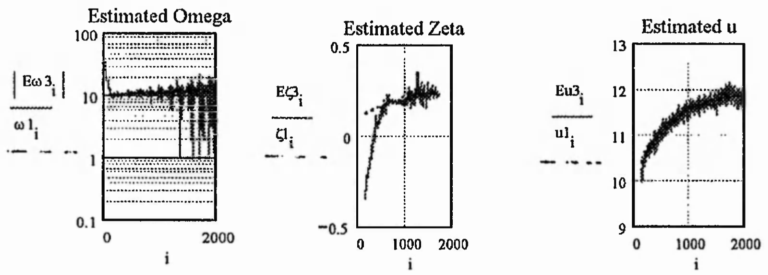

Figure 5.17.b: Estimated parameters with 1st order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) = 32.

5.2.4 Discussion

For the variable parameter case with 1st order dynamics, Algorithm 3 gave better predictions for ω and u than algorithms 1 and 2. For ζ, however, the accuracy of estimation in Algorithm 1 was highest, closely followed by Algorithm 2 then Algorithm 3. On the other hand, the accuracy of the three algorithms depends on the accuracy of the estimated higher derivatives. When the simulated first state variable (position) reaches its maximum, the estimated 1st derivative (velocity) drops to exactly zero and when the estimated velocity reaches its maximum, the estimated 2nd derivative (acceleration) drops to exactly zero. This also occurs for the higher derivatives. As long as this situation exist, with the necessary value of the lag compensator L, then accurate predictions can be expected.

The effects of measurement noise on the estimation accuracy of the three algorithms for parameters with 1st order dynamics were investigated when the incoming trajectories were corrupted with random noise. Two different ranges of random numbers were applied to each algorithm. Simulation results showed highest prediction robustness was achieved by Algorithm-3, followed by Algorithm-2 then Algorithm-1. By simulation, several factors have been seen to affect the estimation accuracy. Firstly, the accuracy of the derivatives is very important and is related to the filter gain. This case showed improvement on the accuracy of the estimation when the simulated output of the system was without noise. However, when noise is present an increase in the gain results in less accurate derivatives obtained and hence produces less accurate estimations. Secondly, if any noise is present, however small, the derivative values become increasingly inaccurate for higher gains and the algorithm that uses fewer derivatives is the most robust. This problem was practically resolved by sufficiently reducing the gain and hence the cut-off frequency (G from 1000 to 20); for simplicity and consistency G was made equal throughout the higher derivative estimators. Shift indices and amplitude magnification techniques were introduced to compensate for the lag and the amplitude attenuation that occur during reduced G values.

5.3 Estimating Parameters with 2nd Order Dynamics

In this section the parameters ω, ζ and u are time varying as outputs of 2nd order systems to provide a rigorous test of the algorithms. Each parameter sub-system will have its own 3 parameters: for ω, they are ωo, ζo and uo; for ζ they are ωz, ζz and uz; and for u they are ωu, ζu and uu. Program 3 [Appendix C] was used to get continuous estimate of the three parameters ω, ζ and u, corresponding to values of: variable ω with (ωo=3, ζo=0.5,uo=30) and ζ=0.001, u=1, variable ζ with (ωz=3, ζz=0.5,uz=0.3) and ω=5, u=1, and variable u with (ωu=3, ζu=0.5,uu=10) and ω=10, ζ=0.001 and G=5000, for a duration of 3 seconds in each case.

5.3.1 Algorithm 1

Algorithm 1 – Varying a single Parameter at a Time- Without noise

Program 3 [Appendix C] was used to get two sets of results corresponding to gaps of 40 and 100 time steps between the points needed for the 3 simultaneous equations in this algorithm, Figures 5.18.a and 5.18.b respectively. As expected, large errors resulted from the larger gap; using a gap of 40 reduced these errors considerably.

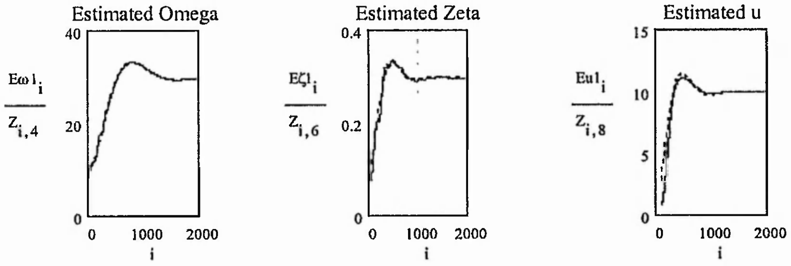

Figure 5.18.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, using Algorithm-1 (separation time step = 40 points).

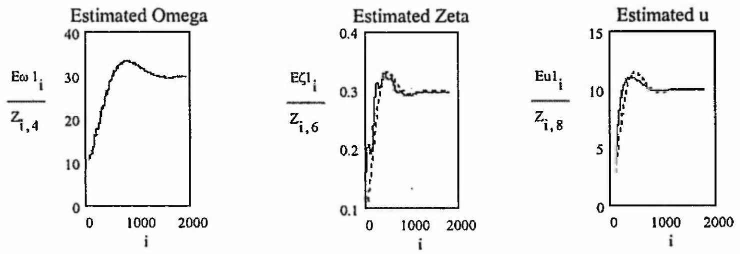

Figure 5.18.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, using Algorithm-1 (separation time step =100 points).

Algorithm 1 – Varying more than one Parameter at a Time – Without Noise

Simulation results were then obtained using Algorithm 1 for parameters with 2nd order dynamics, when varying more than one parameter at a time. Figures 5.19, show the results obtained for estimating: ω (ωo=5, ζo=0.1, uo=11 starting from 5), ζ (ωz=5, ζz=0.1, uz=0.1 starting from 0.01) and u constant at 1.

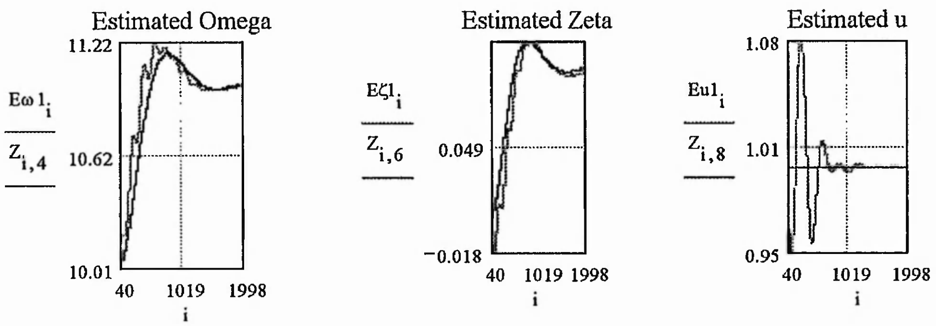

Figure 5.19: Estimated parameters with 2nd order dynamics for ω and ζ but keeping u constant using Algorithm-1.

Algorithm 1 - Sensitivity to Noise

Program 4 [Appendix C] was used to test the estimation accuracy of the algorithm corresponding to two noise levels (low, 0.1% and high, 10%). The simulation results shown in Figures 5.20.a, 5.20.b produced values that agree with the desired set values, i.e. ω with (ωo=10, ζo=0.9, uo=30) and ζ=0.001, u=1, ζ with (ωz=10, ζz=0.9, uz=0.3) and ω=5, u=1 and u with (ωu=10, ζu=0.9, uu=10) and ω=10, ζ=0.001 for a duration of 3 seconds.

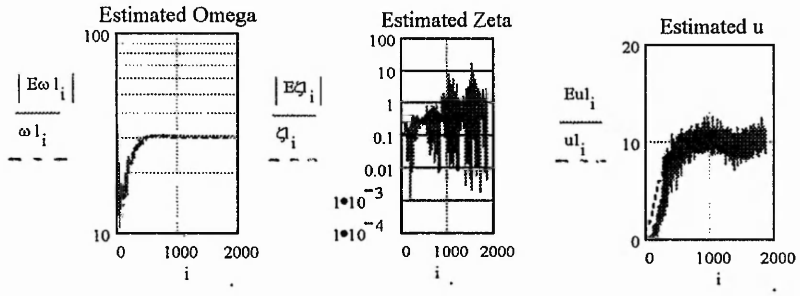

Figure 5.20.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-1 with G = 20, Lag (L) =32.

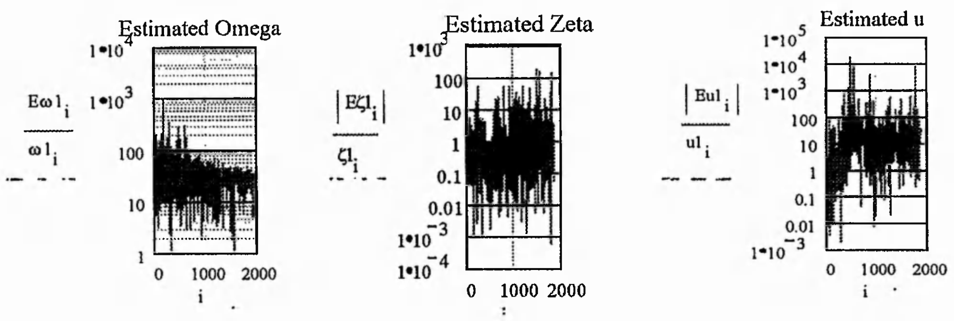

Figure 5.20.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-1 with G = 20, Lag (L) =32.

5.3.2 Algorithm 2

Algorithm 2 – Varying a Single Parameter at a Time - Without noise

Program 3 [Appendix C] was used to estimate the parameters based on two points. Figures 5.21.a and 5.21.b show two sets of results for gaps of 40 and 10 time-steps respectively. The results in Figure 5.21.b are smoother than Figure 5.21.a. For the 40 time-steps gap, the large random errors shown in Figure 5.21.a for ω, ζ have reduced considerably in Figure 5.21.b; for u, however, the improvement is only marginal.

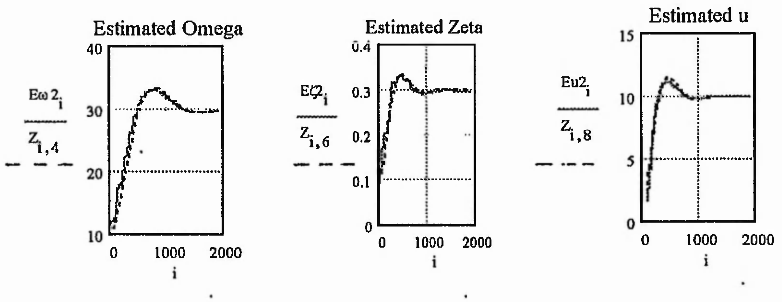

Figure 5.21.a: Estimated variables with 2nd order dynamics for ω, ζ and u using Algorithm 2 (separation time step =40 points).

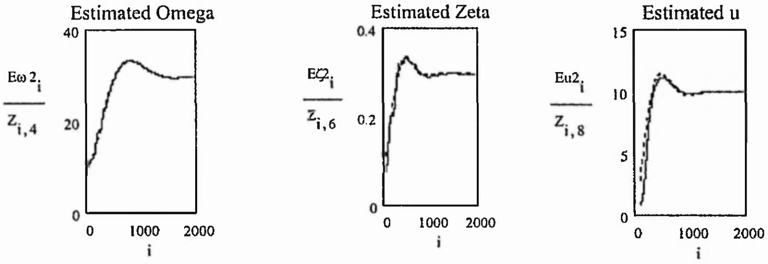

Figure 5.21.b: Estimated variables with 2nd order dynamics for ω, ζ and u using Algorithm 2 (separation time step = 10 points).

Algorithm 2 – Varying more than One Parameter at a Time– Without Noise

Using the same set values as in Section 5.3.1, but using Algorithm 2 rather than Algorithm 1 produced the simulation results shown in Figure 5.22

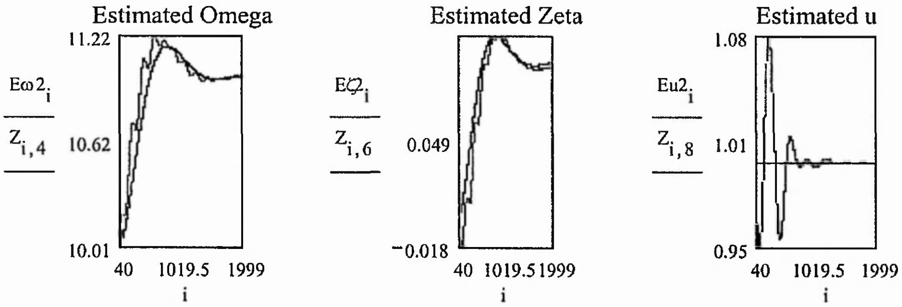

Figure 5.22: Estimated parameters with 2nd order dynamics for ω and ζ but keeping u constant using Algorithm-2.

Algorithm 2 - Sensitivity to Noise

The simulation presented in 5.3.1 was repeated using Algorithm 2. Noise (as described in section 5.3.1) was added to the simulated first state variable (output).

By applying the same set values, the results shown in Figures 5.23.a, 5.23.b were produced.

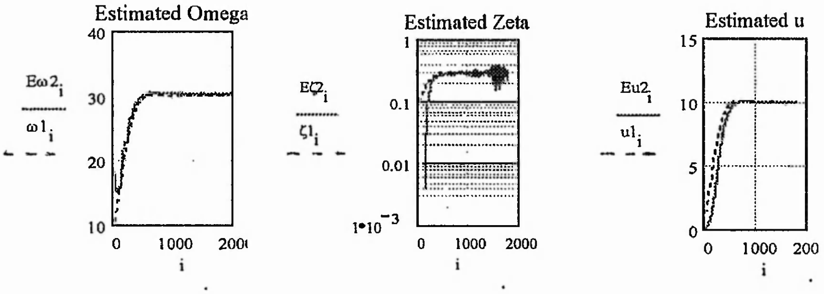

Figure 5.23.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-2 with G = 20, Lag (L) =32.

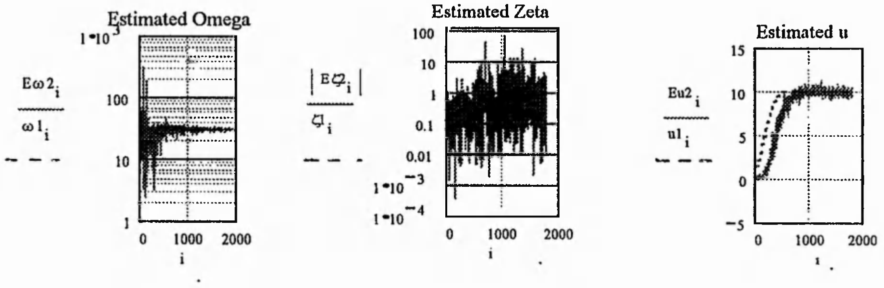

Figure 5.23.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-2 with G = 20, Lag (L) =32.

5.3.3 Algorithm 3

Algorithm 3 – Varying A single Parameter at a Time - Without noise

Program 3 [Appendix C] was used to obtain results for the single point algorithm. The accuracy of the derivatives is very important and is related to the filter gain; three sets of results were obtained to correspond to 2 gain values of 300, 1000 and as shown in Figures 5.24.a, 5.24.b respectively.

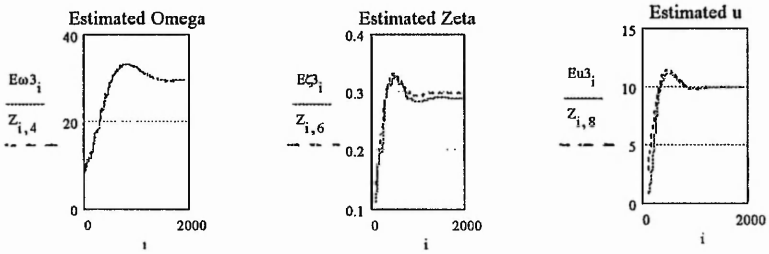

Figure 5.24.a: Estimated parameters with 2nd order dynamics for ω, ζ and u using Algorithm-3 with G = 300.

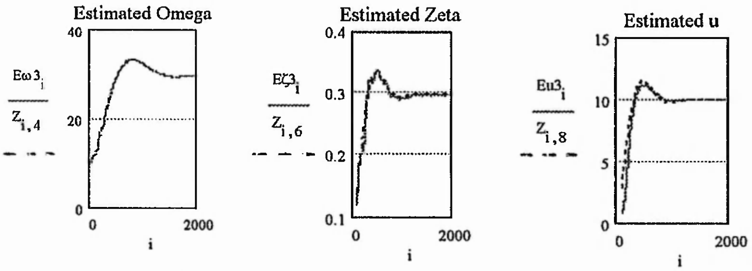

Figure 5.24.b: Estimated parameters with 2nd order dynamics for ω, ζ and u using Algorithm-3 with G = 1000.

Algorithm 3 – Varying more than one Parameter at a Time – Without Noise

The simulation used in 5.3.1 for varying more than one parameter was repeated using Algorithm 3 to estimate the same set values of variables with 2nd order dynamics. The same desired values of the variables ω, ζ and u were used as earlier. Simulation results are shown in Figure 5.25.

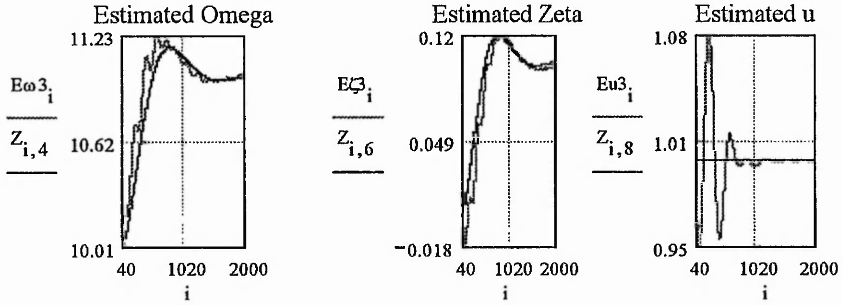

Figure 5.25: Estimated parameters with 2nd order dynamics for ω and ζ but keeping u constant using Algorithm-3.

Algorithm 3 - Sensitivity to Noise

The simulation presented in 5.3.1 was repeated using Algorithm 3. Noise (as described in section 5.3.1) was added to the estimated state variable and its higher derivatives. By applying the same set values, the results shown in Figures 5.26.a, 5.26.b were produced.

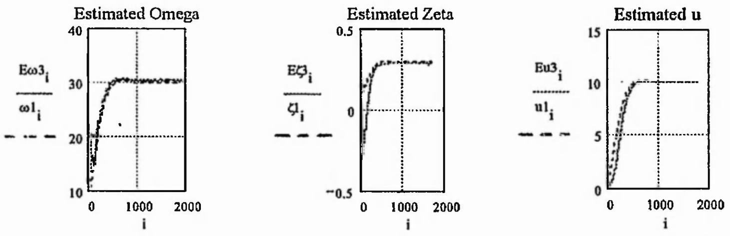

Figure 5.26.a: Estimated parameters with 2nd order dynamics for ω, ζ and u, and low noise (n = 0.001) using Algorithm-3 with G = 20, Lag (L) =32.

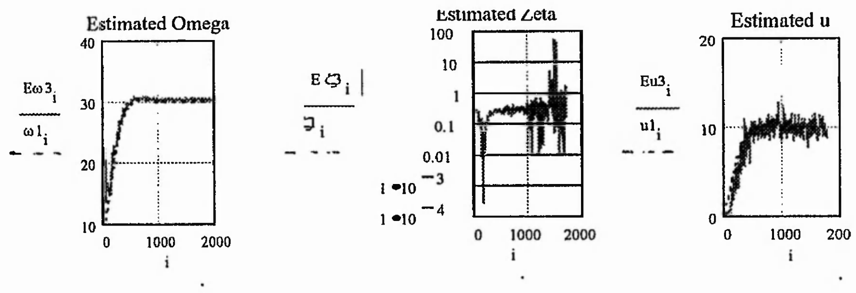

Figure 5.26.b: Estimated parameters with 2nd order dynamics for ω, ζ and u, and high noise (n = 0.1) using Algorithm-3 with G = 20, Lag (L) =32.

5.3.4 Discussion

Variable parameters with 2nd order dynamics proved to be the most testing as expected. Algorithm 3 gave better results for ω and u than algorithms 1 and 2. Estimation of ζ however, proved to be problematic, with Algorithm 1 giving marginally better estimate than Algorithm 2, which in turn was better than Algorithm 3. This is thought to be due to the increasing use of higher derivatives in the latter 2 cases. Different values of gain were tried in Algorithm 3 to improve the accuracy of the higher derivatives, which resulted in marginal improvement as the gain was increased from 300 to 1000. In practice, however, this gain increase will reduce the estimator's noise immunity. There seems to be a clear compromises between these two effects, i.e. estimation accuracy and noise immunity.

5.4 Parameter Estimation of Hierarchical model

A hierarchical model based on second order sub-systems may be formed by modelling the behaviour of each of the three parameters in the main second order system as defined in 3.1.1. The parameters themselves are thus modelled as having their own unique time varying patterns, i.e. have dynamical behaviour. If these parameters are seen as time varying, then they in turn are submitted as input to another parameter estimation algorithm to estimate the parameters of their own dynamics. In section 5.3 the variable parameters with 2^{nd} order dynamics were successfully estimated for all three algorithms. Deep within the system another level of parameters (ωo, ζo, uo; ωz, ζz uz; and ωu, ζu uu) which are constants. In this section, the three algorithms will be tested to provide continuous estimate of these.

5.4.1 Estimation of Variable Parameter ω with 2^{nd} Order Dynamics

In sections 5.3 it was concluded that Algorithm 3 showed a better estimation than Algorithm 1 and, Algorithm 2. Consider the estimation of constants ωo, ζo and uo of the 2^{nd} order dynamic behaviour of the parameter ω using Algorithm 3. To provide an estimate of the first level parameters, the iterative Euler routine was used to generate the filter estimate of the parameter Eω (FEω) of the main 2^{nd} order system and its higher derivatives i.e. FEω', FEω'', FEω''', and FEω''''. Running program 5 to estimate the set values of the first level parameters used in section 5.3 (Algorithm 3- Noise Sensitivity), produced the filter estimate of the first level parameters shown in Figure 5.27.

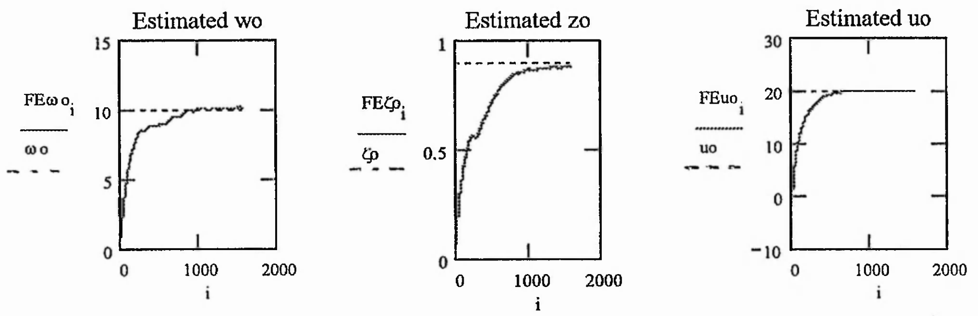

Figure 5.27: Estimated first level parameters (ωo, ζo and uo) of sub-system ω using Algorithm-3.

5.4.2 Estimation of Variable Parameter ζ with 2nd Order Dynamics

In this section the estimation of the first level parameters (ωz, ζz and uz) of the subsystem ζ is considered. The estimated parameter ζ of the main system (in section5.3.3, using Algorithm 3) was fed into a cascade of filters in a similar way to ω in the last section. To provide continuous estimate of the higher derivatives i.e. FEζ', FEζ'', FEζ''', and FEζ''''; the procedure is similar to the one described in section 5.4.1. When these derivatives were obtained, Algorithm 3 was applied. The results of estimating these first level parameters are shown in Figure 5.28.

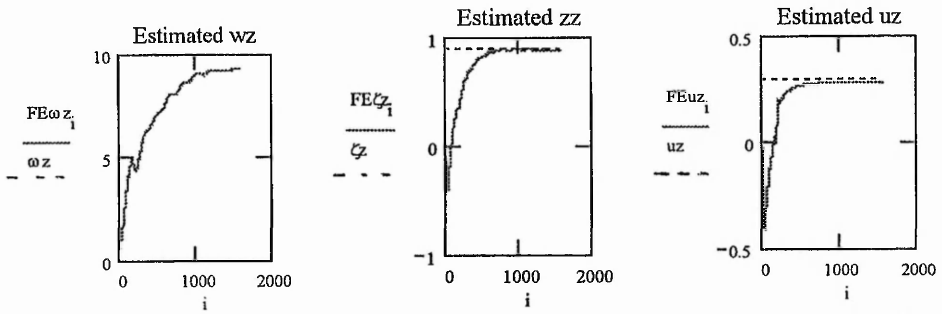

Figure 5.28: Estimated first level parameters (ωz, ζz and uz) of sub-system ζ using Algorithm-3 .

5.4.3 Estimation of Variable Parameter u with 2nd Order Dynamics

Finally the estimation of the first level parameters (ωu, ζu and uu) of the sub-system u is presented in this section. The estimated u was produced after producing the estimated ω and ζ of the main system (as described in section 5.3 using Algorithm 3). The filter estimate of u and its higher derivatives were obtained by the same manner described in section 5.4.1. The result obtained of estimation the first level parameters of the sub-system u is shown in Figure 5.29.

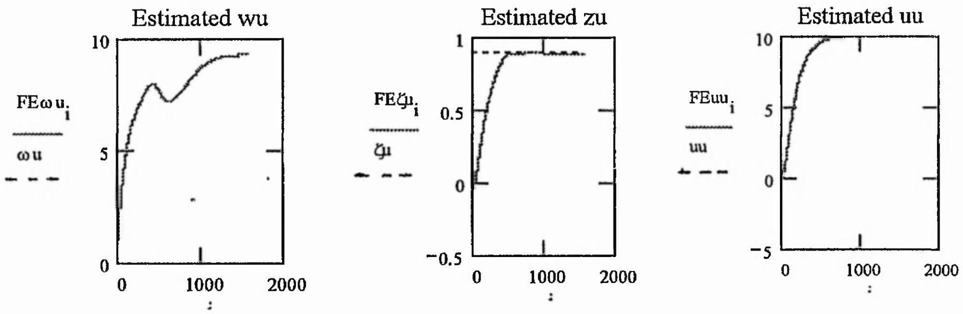

Figure 5.29: Estimated first level parameters (ωu, ζu and uu) of sub-system u using Algorithm-3.

5.4.4 Discussion

Estimated values for constant parameters of the sub-systems with 2nd order dynamic ω, ζ and u were close to the desired set values when Algorithm 3 was used. The estimation accuracy of the parameters depends largely on the following:

-

The estimation accuracy of the 3 sub-systems output ω ζ and u (estimated using Algorithm 3).

The time derivatives (ω, ω', ω'', ω''' and ω''''), (ζ, ζ', ζ'', ζ''' and ζ'''') and (u, u', u'', u''' and u'''') of the sub-systems ω ζ, u.

These in turn depend on the gain value G where, more accurate derivatives are obtained using larger gains. To provide smooth continuous estimate of these derivatives, the values of G had to be reduced (from 1000 to 20) because the estimated (variable) parameters ω, ζ and u are prone to noise and the procedure for producing their derivatives is also prone to noise itself. For low gain lag and amplitude attenuation had to be compensated for using the techniques discussed in section 5.3. Algorithm 3 was applied to estimate the first level parameters of the sub-systems ω, ζ and u and showed accurate estimation.

CHAPTER 6 - CONCLUSIONS AND FURTHER WORK

6.1 Conclusions

6.1.1 Second Order Systems

The three algorithms presented in this work are not limited in their applications to any specific system. Any measurement device that has observed data available can be used. This means that the work conducted here has enormous scope and could be used in a vast range of applications and fields.

Many dynamic systems can be approximated with a second order system. Such a system is sufficiently complex to display the significant features of higher order systems but can be analyzed without excessive computation.

A second order system is determined by three parameters and two variables. The parameters are natural frequency, damping ratio and external input. These parameters can be modelled as constant, variables with 1st order dynamics and variables with 2nd order dynamics. The precise trajectory is determined by the initial values of the signal and its first derivative. This study concentrates on determining the parameter values.

6.1.2 Parameter Estimation Algorithms

Three algorithms were developed for the estimation of natural frequency (ω), damping ratio (ζ) and input (u). These were estimated by using three points of x, x' and x'', two points of x, x', x'' and x''', and one point of x, x', x'', x''' and x''''. The algorithms were tested for 3 categories of parameters: constant, variable with 1st order dynamics and variable with 2nd order dynamics. The algorithms were implemented using Mathcad.

Constant Parameters

Estimated values for constant parameters were very close to the desired set values for all three algorithms. The derived algorithms estimated ω, ζ and u for a good range of values: ω from 1 to 20, ζ between ± 0.01 and ± 1, and u between ± 0.5 and ± 40, and all gave accurate estimates. Estimation errors decreased as ω increased, particularly for small ζ (less than 0.5), where oscillation provided wide variation in the variables. The differences between the (simulated) system time derivatives (x, x' and x'') and their estimates from the filter cascade depended on G (the cut-off frequency). A high G provided a more accurate estimation of derivatives but made the algorithms prone to noise and vice versa. Another disadvantage of high G from the simulation point of view is that simulation time increased considerably due to the integration routine adapting to ever-smaller steps. The algorithms provided progressively faster convergence with Algorithm 3 being the fastest to converge.

Variable Parameters with 1st Order Dynamics

For the variable parameter case with 1st order dynamics, the estimation of the three algorithms were close to the desired set values both when varying a single parameter and when simultaneously varying multiple parameters.

Algorithm 3 gave better predictions for ω and u than algorithms 1 and 2. For ζ, however, the accuracy of estimation in Algorithm 1 was highest, closely followed by Algorithm 2 then Algorithm 3. The accuracy of the three algorithms depended on the accuracy of the estimated higher derivatives. When the first state variable (position) reaches its maximum the 1^{st} derivative (velocity) drops to exactly zero and when the velocity reaches its maximum, the 2^{nd} derivative (acceleration) drops to exactly zero. This also occurs for the higher derivatives. As long as this situation prevails, with suitable lag compensation and amplitude magnification for low G, then accurate predictions can be expected.

Variables Parameters with 2^{nd} Order Dynamics

Variable parameters with 2^{nd} order dynamics proved to be the most testing as expected. The estimation of the three algorithms was again close to the desired set values, both when varying a single parameter and when simultaneously varying multiple parameters.

Algorithm 3 gave better results for ω and u than algorithms 1 and 2; but ζ, however, proved to be problematic, with Algorithm 1 giving marginally better estimate than Algorithm 2 which in turn was better than Algorithm 3. This is thought to be due to the increasing use of higher derivatives in the latter 2 cases. Different values of gain were tried in Algorithm 3 to improve the accuracy of the higher derivatives, which resulted in marginal improvement as the gain was increased from 300 to 1000. In practice, however, this gain increase will reduce the estimator's noise immunity. There seems to be a clear compromise between these two effects.

6.1.3 Parameter Estimation of Hierarchical Models

The three parameters (ω, ζ and u) of the second order system may be modelled as having their own unique time varying patterns, i.e. having dynamical behaviour. When the three parameters (ω, ζ and u) are estimated, they are in turn submitted as input to a second level parameter estimation algorithm to estimate the parameters of their own dynamics (the first level parameters).

The process of estimating of the parameters of the sub-systems ω, ζ and u from the observed main system output requires many different steps. Firstly, high gain (G) was needed to produce accurate higher derivatives and hence estimations of the three parameters ω, ζ and u of the main system. Secondly, low gain was needed to produce a filtered version of ω, ζ and u and then their higher derivatives. This low gain was needed because the estimation of the parameters was prone to noise, as was the production of their derivatives. As low gain was used, compensation of lag and attenuation was essential to produce accurate estimation of the sub-systems parameters.

Algorithm 3 proved to produce accurate estimation of these parameters.

6.1.4 Noise Sensitivity

The effects of measurement noise on the estimation accuracy of the three algorithms for the three categories of parameter sets were investigated when the incoming trajectories were corrupted with random noise. Noise was based on a random number generator of zero mean. The Runge-Kutta integration routine had to be abandoned as it uses 4 evaluations of the derivative vector, which give different noise values in each evaluation and this leading to inconsistent integration results. A simple iterative Euler integration with a single evaluation was used instead.

The three parameter estimation algorithms were tested for an increasing noise levels of 0.1%, 1% and 10% of the nominal trajectory amplitude. Heavy smoothing was provided by low values of G (cut-off frequencies), which introduced increasing lags and amplitude attenuation in the successive stages of the higher derivative estimators. Numerical lag and amplitude compensation techniques were introduced which selected progressively distant values of the higher derivatives and calibrated the reduction in the magnitude; these techniques, together with the increased smoothing applied to the higher derivatives gave Algorithm 3 the leading edge in combating noise.

6.2 Overview

Overall, the estimation of parameters is best achieved with Algorithm 3, which uses the time trajectory and its 1^{st}, 2^{nd}, 3^{rd} and 4^{th} time derivatives. This algorithm is successful because it uses a single point, where changes in the parameters are negligible when the integration step is sufficiently small. Thus, the time derivatives are less erroneous in assuming (implying) constant parameters. In the other two algorithms, the gap between points causes a flaw in this assumption and therefore the estimation is less accurate. Furthermore, Algorithm 3 proved to be the most robust and behaved well when noise was added.

The three algorithms presented in this study showed accurate estimation of different categories of parameter sets. Filter estimates of the time derivatives are obtained using numerical methods. The numerical methods are available in many software packages. Implementation of these algorithms is therefore computationally inexpensive.

6.3 Further Work

In this study only simulation was used. It would be beneficial to see how accurately the algorithms can cope with real data. This would require a real-life situation which can be approximated to a second order system to be identified. Such systems include waveform of human speech, images in stereo camera systems and the non-linear dynamical roll motion for ships. This may well involve adjustment to cope with noise sources with different frequency spectrums. It will also be important to see how well each algorithm copes when the system being investigated is not exactly a second order system, as in the simulations carried out so far.

Estimating natural frequency (ω) is a form of information mining using numerical frequency de-modulation. On the theoretical side, the method need to be generalised in terms of matrix formulation to higher order system dimensions.

References

Al-Dabass D., "Characteristics Estimation of Point Objects using Stereo Vision", *IEE Colloquium on Digital Signal Processing, Savoy Place, London,* **(1985)**.

Al-Dabass D., Zreiba A., Evans D. and Sivayoganathan S., "Simulation of Three Parameter Estimation Algorithms for Pattern Recognition Architecture", *UKSIM99 conference, St. Catherine's College, Cambridge*, 7-9 April, pp. 170-176, **(1999)**.

Al-Dabass D., Zreiba A., Evans D. and Sivayoganathan S., "Simulation of Noise Sensitivity of Parameter Estimation Algorithms", *Simulation99 conference, UCL*, pp. 32-35, **(1999)**.

Ameen H., "A New Approach in Estimating the Dynamics Matrix of a System", *Proc. of the IEEE National Aerospace & Electrical Conf.*, pp. 33-37, **(1998)**.

Anand D. and Zmood R., "Introduction to Control Systems", *Butterworth-Heinemann Ltd*, ISBN 0 7506 2298 9, **(1995)**.

Arafeh S. And Sage A., "Hierarchical System Identification of States and Parameters in Interconnected Power Systems", *Int. J. Systems*, Vol. 9, pp. 817-823, **(1974)**.

Astrom K., "Maximum Likelihood and Prediction Error Methods", *Automatica,* Vol. 16, pp. 551-574, **(1980)**.

Bard Y., "Non-linear Parameter Estimation", Academic Press, New York and London, ISBN 0 12 078250 2, **(1974)**.

Baumeister J., Scondo W., Demetriou M. and Rosen I., "On-Line Parameter Estimation for Infinite-Dimensional Dynamical Systems", *Siam J. Control Optim.*, Vol. 35, No. 2, pp. 678-713, **(1997)**.

Beck J. and Arnold K., "Parameter Estimation in Engineering and Science", Copyright John Wiley & Sons, ISBN 0 471 06118 2, **(1977)**.

Belikove S., Kaplinksy M. and Friedland B., "Parameter Estimation Model for Rapid Thermal Processing; Theory and Experimental Results", *In Proc. American Control Conf., Seattle, WA*, Vol. 1, pp. 7-11, **(1995)**.

Bohuml J. and Radek S., "Correspondence Comments on Second Order System Models of High Order Plants'', *International Journal of Systems Science*, Vol. 27, No. 5, pp. 517-518, **(1996)**.

Cawley P., "The Reduction of Bias Error in Transfer Function Estimates using FFT-based Analysis", *Journal of Vibration, Acoustics, Stress and Reliability in Design*, pp. 29-35, **(1984)**.

Carson E., "System Identification in Biology", *Trans. Of Inst. Of Measurement & Control*, Vol. 4, No. 4, pp. 171-176, **(1982)**.

Chang L. and Rae-Hong P., "Hybrid Mapping Parameter Estimation using Hierarchical Structure in Object-Oriented Coding", *IEEE Trans .On Consumer Electronics*, Vol. 43, No. 4, pp. 1213-1219, **(1997)**.

Chemouil P., Katebi M., Sastry D. and Singh M., "Maximum *a Posterior* Parameter Estimation in Large Scale Systems", *Automatica*, Vol. 17, No. 6, pp. 845-851, **(1981)**.

Desforges M., Cooper J. and Wright J., "Spectral and Modal Parameter Estimation from Output-Only Measurements", *Mechanical Systems and Signal Processing*, Vol. 9, No. 2, pp. 169-186, **(1995)**.

Douglas G. and Donald M., "An ordinary differential Equation Technique for Continuous-Time parameter Estimation", *IEEE Trans. on Automatic Control*, Vol. 38, No. 4, pp. 514-528, **(1993)**.

Eykhoff P., "A Bird's Eye View On Parameter Estimation and System Identification", *Automatisierungstechnik*, Vol. 36, No. 11, pp. 413-420, **(1988)**.

Eykoff P., "System Identification Parameter and State Estimation", Copyright John Wiley & Sons, ISBN 0 471 24980 6, **(1974)**.

Fedorov V., "Theory of Optimal Experiments", Academic Press, New York San Fransico London, **(1972)**.

Friedland B., 'A Non-linear Observer for Estimating Parameters in Dynamic Systems', *Automatica*, Vol. 33, No. 8, pp.1525-1530, **(1997)**.

Friedland B., "A Simple Non-linear Observer for Estimating Parameters in Dynamic Systems", *IFAC 12th Triennial World Congress, Sydney, Australia*, pp. 63-66, **(1993)**.

Horng I and Chou H., "Analysis Parameter Estimation and Optimal Control of Time-Delay Systems Via Chebyshev Series", *Int. J. Control*, Vol. 41, pp. 114-135, **(1985)**.

Gang Z. and Cang-Pu W., "A Recursive Algorithm for the Linear Approximation of Large Scale Interconnected Steady-State Systems", *IEEE*, pp. 788-791, **(1996)**.

Gersch, W., "Least Squares Estimates of Structural System Parameters using Covariance Function Data", *EEE Trans. On auto. Control*, Vol. 19, No. 6, **(1974)**.

Glentis G. and Kalouptsidis, "Efficient Algorithms for Second Order Polynomial System Identification", *Signal processing VI theories & applications Proc. EUSIPCO* 92, Vol. 2, Pp. 1101-1104, **(1992)**.

Goodwin, C., "Real Time Recursive Block Parameter Estimation of Second Order Systems", *PhD Thesis, Dept. of Computing, The Nottingham Trent University, Nottingham*, **(1997)**.

Goodwin G. and Sin K., "Adaptive Filtering, Prediction and Control", Prentice Hall Inc., Englewood Cliffs, New Jersey, **(1984)**

Hwang H., "Identification Techniques of Structure Connection Parameters using Frequency Response Functions", *Journal of Sound and Vibration*, Vol. 212, No. 3, pp. 469-479, **(1998)**.

Huang T. And Clements C., "Parameter Estimation for the Second-Order-Plus-Dead-Time Model", *Industrial and Engineering Chemistry, Process Design and Development*, Vol. 21, pp. 601-603, **(1982)**.

Isermann R., Lachmann K. and Matko D., "Adaptive Control System", Printice Hall International (UK) Ltd., ISBN: 0 13 005414 3, **(1992)**.

Kreyszig E., "Advanced Engineering Mathematics", Seventh Edition, Published by John Wiley & Sons, ISBN 0 471 59989 1, **(1993)**.

Kailath T., "Linear Systems", Printice-Hall, Inc., Englewood Cliffs, N.J, ISBN 0 13 536961 4., **(1980)**

Kalman, R., "A New Approach to Linear Filtering and Prediction Problems", *Tans. of SAME: Journal of Basic Eng., series D*, Vol. 82, pp. 35-45, **(1960)**.

Karampetakis N., Pugh A. and Hayton G., "Notes on A Hierarchical Theory of Systems and Application", *Kybernetika*, Vol. 33, No. 2, pp. 185-201, **(1997)**.

Liu K., "Modal Parameter Estimation using the State Space Method", *Journal of Sound and Vibration*, Vol. 197, No. 4, pp. 387-402, **(1996)**.

Ljung L., " Issues in System Identification", *IEEE Control Systems,* Vol. 11, No.1, pp. 25-29, **(1991)**.

Ludwig E., Crow M., Erickson K. and Shah K., "A Feasibility Study of On-Line Excitation System Parameter Estimation", *IEEE Transaction on Power Systems*, Vol. 13, No. 3, pp. 910-916, **(1998)**.

Mathew A., "On the Use of Hermite Polynomials for The Recursive Identification of a Class of Time Varying Linear Distributed Parameter Systems", *IETE Technical Review*, Vol. 15, No. 4, pp. 229-234, **(1998)**.

Mathsoft, "Mathcad 7 User's Guide", Mathsoft, Inc., **(1991-1997)**.

Meyer R., "Simplifying Process Response Approximation", *Instruments and Control Systems*, Vol. 40, pp. 76-79, **(1967)**.

Nishida N., Hiratsuka S.; Tanaka T. and Okino T., "An Interactive Algorithm for the Parameter Estimation of Complex Systems and its Application to An Ecological Modelling of An Actual Japanese Lake", *Ecological Modelling*, Vol. 32, pp. 85-93, **(1986)**.

Pachter M., D'Azzo J. and Buzogany E., "Second-Order System Models of High Order Plants", *International Journal of Systems Science,* Vol. 25, pp. 1653-1662, **(1994)**.

Press W., Vetterling W., Flannery B. And Teukolsky S., "Numerical Recipes in C: The Art of Scientific Computing:2nd ed.", *Cambridge University Press,* **(1992)**.

Roberto M., "Dynamic Parameter Estimation of a Second Order System to a Step Function", *IEEE Trans. on Instrumentation & Measurement*, Vol. **39,** No. 5, pp. 790-2, **(1990)**.

Seibold S., "Identification of Physical Parameters using an Instrumental Variables Technique", *Mechanical Systems and Signal Processing*, Vol. 11, No. 3, pp. 425-439, **(1997)**.

Singh M. and Titli A., "Handbook of Large Scale System Engineering Applications", North-Holand Publishing Company, ISBN 0 444 85283 2, **(1979)**.

Stericker D. and Sinha N., "Identification of Continuous time systems from samples of the input-output data using the δ-operator", *Control Theory and Advanced Technology*, Vol. 9, No. 1, pp. 113-125, **(1993)**.

Sultan M., Hassan M. and Calvet J., "Parameter Estimation in Large-scale Systems using Sequential Decomposition", *Int. J. System Sci.*, Vol. 19, No. 3, pp. 487-496, **(1988)**.

Sultan M., Hassan M., Calvet J. and Bilal A., "A First-order Parallel Decomposition Method for Parameter Estimation in Large Scale Systems", *Large Scale Systems in Info. & Decision Techn.*, Vol. 9, pp. 51-62, **(1985)**.

Sundaresan KR., Chandra C. and Krishnaswamy P., "Evaluating Parameters from Process Transient", *Industrial Engineering Chemistry, Process Design and Development*, Vol. 17, pp. 237-241, **(1978)**.

Yarimer E. and Virgin L., "Reliability of an Identification System for Predicting Incipient Capsize Due to Chaotic Rolling Motion of A Ship", Proceeding of the Workshop at Lambrecht/Pfalz, pp. 182-194, **(1988)**.

Zhao P. He Z., "On Multidimensional System Identification", *Advances in Modelling & Simulation, ASME press*, Vol. 18, No. 1, pp. 15-26, **(1989)**.

Zhihong M., "Parameter Estimation of Continuous Linear Systems using Functional Approximation", *Computers Elec. Eng.*, Vol. 21, No. 3, pp. 183-187, **(1995)**.

Zmood A., "Introduction to Control Systems", Third Edition, Butterworth-Heinemann, ISBN 0 7506 2298 9, (**1995**).

Appendix A

Derivation of Complex Solution to the Second Order System

This Appendix derives the solution x(t), given the initial conditions $x_0(0)$, $x'_0(0)$ and input u, of the characteristic equation,

$$a \cdot x'' + b \cdot x' + c \cdot x = u \cdot \omega^2 \qquad \text{(A.1)}$$

Where a = 1, b = 2.ζ.ω, c = ω^{-2} and. Each apostrophe given to each x indicates a time derivative.

e.g. $$x'' = \frac{d^2}{dt^2} x$$

The roots of the characteristic equation are:

$$r_1 = \frac{-b + \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \qquad \text{(A.2)}$$

$$r_2 = \frac{-b - \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \qquad \text{(A.3)}$$

Note that,

$$r_1 \cdot r_2 = \frac{-b + \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \cdot \frac{-b - \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} = \frac{b^2 - (b^2 - 4 \cdot a \cdot c)}{4 \cdot a^2} = \frac{c}{a}$$

The free Solution can be written as,

$$x(t) = c_1 \cdot e^{r_1 \cdot t} + c_2 \cdot e^{r_2 \cdot t} \qquad \text{(A.4)}$$

To find the values for c_1 and c_2, we examine the initial conditions for x(t) and x'(t).

The initial conditions are,

$$x_0(0) \quad \text{and} \quad x'_0(0)$$

We can therefore write that,

$$x_0(0)=c_1+c_2 \tag{A.5}$$

Now since,

$$x'(t)=c_1 \cdot r_1 \cdot e^{r_1 \cdot t}+c_2 \cdot r_2 \cdot e^{r_2 \cdot t} \tag{A.6}$$

We can write,

$$x'_0(0)=r_1 \cdot c_1+r_2 \cdot c_2 \tag{A.7}$$

Inserting Equation A.5 into Equation A.7, we get,

$$x'_0(0)=r_1 \cdot \left[x_0(0)-c_2\right]+r_2 \cdot c_2=r_1 \cdot x_0(0)-r_1 \cdot c_2+c_2 \cdot r_2 \tag{A.8}$$

Re-arranging,

$$x'_0(0)-r_1 \cdot x_0(0)=c_2 \cdot \left(r_2-r_1\right) \tag{A.9}$$

Further re-arranging for c2, we get,

$$c_2=\frac{\left[x'_0(0)-r_1 \cdot x_0(0)\right]}{r_2-r_1} \tag{A.10}$$

Now inserting Equation A.10 into Equation A.5,

$$c_1 = x_0(0) - c_2 = x_0(0) - \frac{\left[x'_0(0) - r_1 \cdot x_0(0)\right]}{r_2 - r_1} = \frac{x_0(0) \cdot r_2 - x'_0(0)}{r_2 - r_1} \qquad (A.11)$$

We now calculate the forced solution which is then added to the free solution.

$$x(t) = \omega^2 \cdot \int_0^t \phi(t - \tau) \cdot u(t)\, d\tau \qquad (A.12)$$

Where $\phi(.)$ is a free response of the system $= c_1 \cdot e^{r_1 \cdot t} + c_2 \cdot e^{r_2 \cdot t}$

Therefore, for $u(t) = u$,

$$x(t) = u \cdot \omega^2 \cdot \left[\int_0^t \left(c_1 \cdot e^{r_1 \cdot t} \cdot e^{r_1 \cdot \tau}\right) + \left(c_2 \cdot e^{r_2 \cdot t} \cdot e^{r_2 \cdot \tau}\right) d\tau\right] \qquad (A.13)$$

Where $\phi(t) = c_1 \cdot e^{r_1 \cdot t} + c_2 \cdot e^{r_2 \cdot t}$

we can now deduce $c_1 + c_2$ from $\phi(0) = 0$, and recalling that $\phi'(0) = 1$ (universal initial condition for weight functions.

We can now write,

$$\phi'(t) = c_1 \cdot r_1 \cdot e^{r_1 \cdot t} + c_2 \cdot r_2 \cdot e^{r_2 \cdot t}$$

$$\phi'(0) = 1 = c_1 \cdot r_1 + c_2 \cdot r_2$$

$$1 = c_1 \cdot r_1 - r_2 \cdot c_1 = c_1 \cdot (r_1 - r_2)$$

$$c_1 = \frac{1}{r_1 - r_2} \quad \text{and} \quad c_2 = \frac{-1}{r_1 - r_2}$$

Therefore using the expressions of c_1 and c_2 in Equation A.4,

$$x(t)=\frac{u\cdot\omega^2\cdot e^{r_1\cdot t}}{r_1\cdot(r_1-r_2)}-\frac{u\cdot\omega^2\cdot e^{r_2\cdot t}}{r_2\cdot(r_1-r_2)}-\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}-\frac{1}{r_2\cdot(r_1-r_2)}$$

$$x(t)=\frac{u\cdot\omega^2\cdot e^{r_1\cdot t}}{r_1\cdot(r_1-r_2)}-\frac{u\cdot\omega^2\cdot e^{r_2\cdot t}}{r_2\cdot(r_1-r_2)}+\frac{u\cdot\omega^2}{r_1\cdot r_2}$$

The total solution is then,

$$x(t)=c_1\cdot e^{r_1\cdot t}+c_2\cdot e^{r_2\cdot t}+\left[\frac{u\cdot\omega^2\cdot e^{r_1\cdot t}}{r_1\cdot(r_1-r_2)}-\frac{u\cdot\omega^2\cdot e^{r_2\cdot t}}{r_2\cdot(r_1-r_2)}+\frac{u\cdot\omega^2}{r_1\cdot r_2}\right]$$

$$x(t)=\left[c_1+\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}\right]\cdot e^{r_1\cdot t}+\left[c_2-\frac{u\cdot\omega^2}{r_2\cdot(r_1-r_2)}\right]\cdot e^{r_2\cdot t}+\frac{u\cdot\omega^2}{r_1\cdot r_2} \qquad (A.14)$$

$$x'(t)=\left[r_1\cdot\left[c_1+\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}\right]\cdot e^{r_1\cdot t}+r_2\cdot\left[c_2-\frac{u\cdot\omega^2}{r_2\cdot(r_1-r_2)}\right]\cdot e^{r_2\cdot t}\right]$$

$$x''(t)=\left[(r_1)^2\cdot\left[c_1+\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}\right]\cdot e^{r_1\cdot t}+(r_2)^2\cdot\left[c_2-\frac{u\cdot\omega^2}{r_2\cdot(r_1-r_2)}\right]\cdot e^{r_2\cdot t}\right]$$

$$x'''(t)=\left[(r_1)^3\cdot\left[c_1+\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}\right]\cdot e^{r_1\cdot t}+(r_2)^3\cdot\left[c_2-\frac{u\cdot\omega^2}{r_2\cdot(r_1-r_2)}\right]\cdot e^{r_2\cdot t}\right]$$

$$x''''(t)=\left[(r_1)^4\cdot\left[c_1+\frac{u\cdot\omega^2}{r_1\cdot(r_1-r_2)}\right]\cdot e^{r_1\cdot t}+(r_2)^4\cdot\left[c_2-\frac{u\cdot\omega^2}{r_2\cdot(r_1-r_2)}\right]\cdot e^{r_2\cdot t}\right]$$

Appendix (B)

Derivation of Filter Values (G)

This appendix derives the values of the cut-off frequency (G) in terms of RL low pass filter.

The RL circuit shown in Figure (B.1) is a 1st order low pass filter, whose characteristic equation is:

$$L.\, dI/dt + R.I = V_i \qquad (1)$$

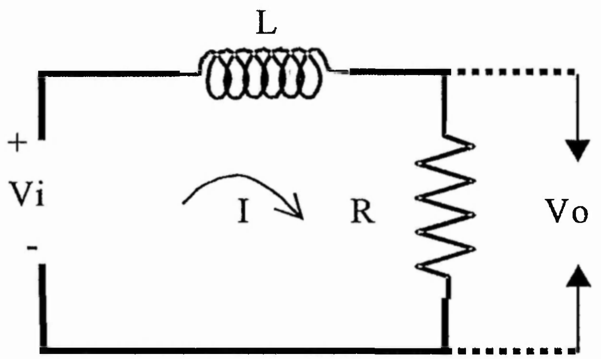

Figure B.1: Simple RL circuit

Total impedance of the RL circuit is: $Z = j\omega L + R$

Total current is: $I = V_i / Z$

The output voltage V_o is: $V_o = I.R$

Thus $V_o = R. [V_i / Z]$

The voltage gain is: $V_o / V_i = 1/[1 + j\omega L/R]$

If $\omega_c = R/L$ then

The voltage gain is: $V_o / V_i = 1/[1 + j\omega/\omega_c]$

When $\omega/\omega_c = 1$, the gain is equal to 0 .707

Where $f_c = \omega_c/2.\ \pi$ is called the cut-off frequency and equal to R/L

To derive an expression for G in terms of R and L is

We have $x'_n = (x_{n-1} - x_n) . G$

And (1) can be rewritten as

$$x'_n = (x_{n-1} - R.x_n) . 1/L$$

When R is equal to one, this gives:

G = 1/L

Appendix C

Simlation Programs of Parameter Estimation Algorithms using Mathcad-8

Program (1)

This program is implemented to test the estimation accuracy of the three algorithms for constants ω, ζ and u.

$G := 1000$ sets the value of the gain (cut-off frequency in Hertz)

$t0 := 0$ $t1 := 3$ Set the simulation time from t0 to t1 in Seconds

$T := 2000$ sets the total number of points Tolerance used =0.0001

$\omega := 10$ sets the desired value of the natural frequency in Hertz

$\zeta := 0.1$ sets the dersired value of the damping ratio

$u := 1$ sets the desired value of the external input

$$x := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad D(t,x) := \begin{bmatrix} x_2 \\ -\omega^2 \cdot x_1 - 2\cdot\zeta\cdot\omega\cdot x_2 + \omega^2\cdot u \\ G\cdot(x_1 - x_3) \\ G\cdot[G\cdot(x_1 - x_3) - x_4] \\ G\cdot[G\cdot[G\cdot(x_1 - x_3) - x_4] - x_5] \\ G\cdot[G\cdot[G\cdot[G\cdot(x_1 - x_3) - x_4] - x_5] - x_6] \\ G\cdot[G\cdot[G\cdot[G\cdot[G\cdot(x_1 - x_3) - x_4] - x_5] - x_6] - x_7] \end{bmatrix}$$

x_1 and x_2 are objects

x_3, x_4, x_5 are filter estimate of x, x' and x"

x_6 and x_7 are filter estimate of x''' & x''''

$Z := \mathrm{Rkadapt}(x, t0, t1, T, D)$ Solving theD-vector using 4th order Runge-Kutta method with adaptive step size.

Note that $Z_{n,4}$ is Ex, $Z_{n,5}$ is Ex', $Z_{n,6}$ is Ex" $Z_{n,7}$ is Ex''' and $Z_{n,8}$ is Ex''''

Algorithm-1: Using Three points of x, x' and x''

$\Delta 1 := 1$ $\Delta 2 := 2$ set the 2[nd] and the 3[rd] points

$i := 1 .. T - \Delta 2$ Sets the number of iterations

$\Delta 12_i := (Z_{i,4} - Z_{i+\Delta 1,4})$ $\Delta' 12_i := (Z_{i,5} - Z_{i+\Delta 1,5})$ $\Delta'' 12_i := (Z_{i,6} - Z_{i+\Delta 1,6})$

$\Delta 13_i := (Z_{i,4} - Z_{i+\Delta 2,4})$ $\Delta' 13_i := (Z_{i,5} - Z_{i+\Delta 2,5})$ $\Delta'' 13_i := (Z_{i,6} - Z_{i+\Delta 2,6})$

$$E\omega 1s_i := \frac{-\left[\left(\frac{\Delta'' 12_i}{\Delta' 12_i}\right) - \frac{\Delta'' 13_i}{\Delta' 13_i}\right]}{\left[\left(\frac{\Delta 12_i}{\Delta' 12_i}\right) - \frac{\Delta 13_i}{\Delta' 13_i}\right]} \qquad E\omega 1_i := \sqrt{E\omega 1s_i}$$

$$E\zeta 1_i := \frac{-\left(\Delta 12_i \cdot E\omega 1_i + \frac{\Delta'' 12_i}{E\omega 1_i}\right)}{2 \cdot \Delta' 12_i}$$

$$Eu1_i := \frac{Z_{i,6}}{E\omega 1s_i} + 2 \cdot \frac{E\zeta 1_i}{E\omega 1_i} \cdot Z_{i,5} + Z_{i,4}$$

Algorithm -2: Using two points and one extra derivative

$\Delta := 1$ sets the 2[nd] point

$i := 10 .. T - \Delta$ Sets the number of iterations

$\Delta 12_i := (Z_{i,4} - Z_{i+\Delta,4})$ $\Delta' 12_i := (Z_{i,5} - Z_{i+\Delta,5})$ $\Delta'' 12_i := (Z_{i,6} - Z_{i+\Delta,6})$ $\Delta''' 12_i := (Z_{i,7} - Z_{i+\Delta,7})$

$$E\omega 2s_i := \frac{-\left[\Delta' 12_i \cdot \Delta''' 12_i - (\Delta'' 12_i)^2\right]}{\left[(\Delta' 12_i)^2 - \Delta 12_i \cdot \Delta'' 12_i\right]}$$ or $$E\omega 2_i := \sqrt{E\omega 2s_i}$$

$$E\zeta 2_i := \frac{-\left(\frac{\Delta 12_i}{\Delta' 12_i} + \frac{\Delta'' 12_i}{E\omega 2s_i \cdot \Delta' 12_i}\right)}{2 \cdot \left(E\omega 2_i\right)^{-1}}$$

$$Eu2_i := \frac{Z_{i,6}}{E\omega 2s_i} + 2 \cdot E\zeta 2_i \cdot \left(E\omega 2_i\right)^{-1} \cdot Z_{i,5} + Z_{i,4}$$

Algorithm -3: Using one point and two extra derivaitves

$i := 1 .. T$ Sets the number of iterations

$$E\omega 3s_i := \frac{\left[Z_{i,6} \cdot Z_{i,8} - \left(Z_{i,7}\right)^2\right]}{Z_{i,5} \cdot Z_{i,7} - \left(Z_{i,6}\right)^2} \quad \text{or} \quad E\omega 3_i := \sqrt{\left|E\omega 3s_i\right|}$$

$$E\zeta 3_i := -\left(E\omega 3_i \cdot \frac{Z_{i,5}}{2 \cdot Z_{i,6}} + \frac{Z_{i,7}}{2 \cdot E\omega 3_i \cdot Z_{i,6}}\right)$$

$$Eu3_i := \frac{Z_{i,6}}{E\omega 3s_i} + 2 \cdot \frac{E\zeta 3_i}{E\omega 3_i} \cdot Z_{i,5} + Z_{i,4}$$

Program (2)

This program is implemented to test the estimation accuracy of the three algorithms for varaibles with 1st order dynamics ω, ζ and u.

$G := 1000$ sets the value of the gain (cut-off frequency in Hertz)

$t0 := 0$ $t1 := 3$ Se the simulation time from t0 to t1 in Seconds

$T := 2000$ sets the total number of points Tolerance used =0.0001

$$x := \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 10 \\ 0.1 \\ 1 \end{bmatrix} \qquad D(t,x) := \begin{bmatrix} x_2 \\ -(x_8)^2 \cdot x_1 - 2 \cdot x_9 \cdot x_8 \cdot x_2 + (x_8)^2 \cdot x_{10} \\ G \cdot (x_1 - x_3) \\ G \cdot [G \cdot (x_1 - x_3) - x_4] \\ G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] \\ G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] \\ G \cdot [G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_3) - x_4] - x_5] - x_6] - x_7] \\ 30 - x_8 \\ 0.4 - x_9 \\ 0 \end{bmatrix}$$

x_1 and x_2 are objects

$[(x_3, x_4)$ and$]$ x_5 are filter estimate of x, x' & x''

x_6 and x_7 are filter estimate of x''' & x''''

x_8, x_9 and x_{10} are desired set value of ω, ζ and u respectively

$Z := \text{Rkadapt}(x, t0, t1, T, D)$ Solving the D-vector using 4th order Runge-Kutta method with adaptive step size.

Note that $Z_{n,4}$ is Ex, $Z_{n,5}$ is Ex', $Z_{n,6}$ is Ex'' $Z_{n,7}$ is Ex''' and $Z_{n,8}$ is Ex''''

Algorithm-1: Using Three points of x, x' and x''

$\Delta 1 := 1$ $\Delta 2 := 2$ set the 2[nd] and the 3[rd] points

$i := 1 .. T - \Delta 2$ Sets the number of iterations

$\Delta 12_i := (Z_{i,4} - Z_{i+\Delta 1,4})$ $\Delta' 12_i := (Z_{i,5} - Z_{i+\Delta 1,5})$ $\Delta'' 12_i := (Z_{i,6} - Z_{i+\Delta 1,6})$

$\Delta 13_i := (Z_{i,4} - Z_{i+\Delta 2,4})$ $\Delta' 13_i := (Z_{i,5} - Z_{i+\Delta 2,5})$ $\Delta'' 13_i := (Z_{i,6} - Z_{i+\Delta 2,6})$

$$E\omega 1s_i := \frac{-\left[\left(\frac{\Delta'' 12_i}{\Delta' 12_i}\right) - \frac{\Delta'' 13_i}{\Delta' 13_i}\right]}{\left[\left(\frac{\Delta 12_i}{\Delta' 12_i}\right) - \frac{\Delta 13_i}{\Delta' 13_i}\right]}$$ or $$E\omega 1_i := \sqrt{E\omega 1s_i}$$

$$E\zeta 1_i := \frac{-\left(\Delta 12_i \cdot E\omega 1_i + \frac{\Delta'' 12_i}{E\omega 1_i}\right)}{2 \cdot \Delta' 12_i}$$

$$Eu1_i := \frac{Z_{i,6}}{E\omega 1s_i} + 2 \cdot \frac{E\zeta 1_i}{E\omega 1_i} \cdot Z_{i,5} + Z_{i,4}$$

Algorithm -2: Using two points and one extra derivative

$\Delta := 1$ sets the 2[nd] point

$i := 10 .. T - \Delta 2$ Sets the number of iterations

$\Delta 12_i := (Z_{i,4} - Z_{i+\Delta,4})$ $\Delta' 12_i := (Z_{i,5} - Z_{i+\Delta,5})$ $\Delta'' 12_i := (Z_{i,6} - Z_{i+\Delta,6})$ $\Delta''' 12_i := (Z_{i,7} - Z_{i+\Delta,7})$

$$E\omega 2s_i := \frac{-\left[\Delta' 12_i \cdot \Delta''' 12_i - \left(\Delta'' 12_i\right)^2\right]}{\left[\left(\Delta' 12_i\right)^2 - \Delta 12_i \cdot \Delta'' 12_i\right]}$$ or $$E\omega 2_i := \sqrt{E\omega 2s_i}$$

$$E\zeta2_i := \frac{-\left(\frac{\Delta 12_i}{\Delta' 12_i} + \frac{\Delta'' 12_i}{E\omega 2s_i \cdot \Delta' 12_i}\right)}{2 \cdot \left(E\omega 2_i\right)^{-1}}$$

$$Eu2_i := \frac{Z_{i,6}}{E\omega 2s_i} + 2 \cdot E\zeta 2_i \cdot \left(E\omega 2_i\right)^{-1} \cdot Z_{i,5} + Z_{i,4}$$

Algorithm -3: Using one point and two extra derivaitves

$i := 1 .. T$ Sets the number of iterations

$$E\omega 3s_i := \frac{\left[Z_{i,6} \cdot Z_{i,8} - \left(Z_{i,7}\right)^2\right]}{Z_{i,5} \cdot Z_{i,7} - \left(Z_{i,6}\right)^2} \quad \text{or} \quad E\omega 3_i := \sqrt{\left|E\omega 3s_i\right|}$$

$$E\zeta 3_i := -\left(E\omega 3_i \cdot \frac{Z_{i,5}}{2 \cdot Z_{i,6}} + \frac{Z_{i,7}}{2 \cdot E\omega 3_i \cdot Z_{i,6}}\right)$$

$$Eu3_i := \frac{Z_{i,6}}{E\omega 3s_i} + 2 \cdot \frac{E\zeta 3_i}{E\omega 3_i} \cdot Z_{i,5} + Z_{i,4}$$

Program (3)

This program is implemented to test the estimation accuracy of the three algorithms for varaibles with 2nd order dynamics ω, ζ and u.

$G := 1000$ sets the value of the gain (cut-off frequency in Hertz)

$t0 := 0$ $t1 := 3$ Set the simulation time from t0 to t1 in Seconds

$T := 2000$ sets the total number of points Tolerance used =0.0001

$\omega o := 3$ $\zeta o := 0.5$ $uo := 30$

$\omega z := 3$ $\zeta z := 0.5$ $uz := 0.3$ set the desired values of the first level parameters of the sub-systems ω, ζ and u.

$\omega u := 3$ $\zeta u := 0.5$ $uu := 1$

$$x := \begin{bmatrix} 0 \\ 0 \\ 10 \\ 0 \\ 0.1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad D(t,x) := \begin{bmatrix} x_2 \\ -(x_3)^2 \cdot x_1 - 2 \cdot x_3 \cdot x_5 \cdot x_2 + (x_3)^2 \cdot x_7 \\ x_4 \\ -\omega o^2 \cdot x_3 - 2 \cdot \zeta o \cdot \omega o \cdot x_4 + \omega o^2 \cdot uo \\ x_6 \\ -\omega z^2 \cdot x_5 - 2 \cdot \zeta z \cdot \omega z \cdot x_6 + \omega z^2 \cdot uz \\ x_8 \\ -\omega u^2 \cdot x_7 - 2 \cdot \zeta u \cdot \omega u \cdot x_8 + \omega u^2 \cdot uu \\ G \cdot (x_1 - x_9) \\ G \cdot [G \cdot (x_1 - x_9) - x_{10}] \\ G \cdot [G \cdot [G \cdot (x_1 - x_9) - x_{10}] - x_{11}] \\ G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_9) - x_{10}] - x_{11}] - x_{12}] \\ G \cdot [G \cdot [G \cdot [G \cdot [G \cdot (x_1 - x_9) - x_{10}] - x_{11}] - x_{12}] - x_{13}] \end{bmatrix}$$

x_1 and x_2 are objects

x_3, x_5 and x_7 are desired set values of ω, ζ and u respectively

$[(x_9, x_{10})$ and$]$ x_{11} are filter estimate of x, x' & x"

x_{12} and x_{13} are filter estimate of x''' & x''''

$Z := \text{Rkadapt}(x, t0, t1, T, D)$ Solving the D-vector using 4th order Runge-Kutta method with adaptive step size.

Note that $Z_{n,10}$ is Ex, $Z_{n,11}$ is Ex', $Z_{n,12}$ is Ex" $Z_{n,13}$ is Ex''' and $Z_{n,14}$ is Ex''''

Algorithm-1: Using Three points of x, x' and x''

$\Delta1 := 1$ $\Delta2 := 2$ set the 2nd and the 3rd points

$i := 1 .. T - \Delta2$ Sets the number of iterations

$\Delta12_i := (Z_{i,10} - Z_{i+\Delta1,10})$ $\Delta'12_i := (Z_{i,11} - Z_{i+\Delta1,11})$ $\Delta''12_i := (Z_{i,12} - Z_{i+\Delta1,12})$

$\Delta13_i := (Z_{i,10} - Z_{i+\Delta2,10})$ $\Delta'13_i := (Z_{i,11} - Z_{i+\Delta2,11})$ $\Delta''13_i := (Z_{i,12} - Z_{i+\Delta2,12})$

$$E\omega1s_i := \frac{-\left[\left(\frac{\Delta''12_i}{\Delta'12_i}\right) - \frac{\Delta''13_i}{\Delta'13_i}\right]}{\left[\left(\frac{\Delta12_i}{\Delta'12_i}\right) - \frac{\Delta13_i}{\Delta'13_i}\right]}$$

or

$$E\omega1_i := \sqrt{E\omega1s_i}$$

$$E\zeta1_i := \frac{-\left(\Delta12_i \cdot E\omega1_i + \frac{\Delta''12_i}{E\omega1_i}\right)}{2 \cdot \Delta'12_i}$$

$$Eu1_i := \frac{Z_{i,12}}{E\omega1s_i} + 2 \cdot \frac{E\zeta1_i}{E\omega1_i} \cdot Z_{i,11} + Z_{i,10}$$

Algorithm -2: Using two points and one extra derivative

$\Delta := 1$ sets the 2nd point

$i := 10 .. T - \Delta$ Sets the number of iterations

$\Delta12_i := (Z_{i,10} - Z_{i+\Delta,10})$ $\Delta'12_i := (Z_{i,11} - Z_{i+\Delta,11})$ $\Delta''12_i := (Z_{i,12} - Z_{i+\Delta,12})$ $\Delta'''12_i := (Z_{i,13} - Z_{i+\Delta,13})$

$$E\omega2s_i := \frac{-\left[\Delta'12_i \cdot \Delta'''12_i - (\Delta''12_i)^2\right]}{\left[(\Delta'12_i)^2 - \Delta12_i \cdot \Delta''12_i\right]}$$

or

$$E\omega2_i := \sqrt{E\omega2s_i}$$

$$E\zeta 2_i := \frac{-\left(\dfrac{\Delta 12_i}{\Delta' 12_i} + \dfrac{\Delta'' 12_i}{E\omega 2s_i \cdot \Delta' 12_i}\right)}{2 \cdot \left(E\omega 2_i\right)^{-1}}$$

$$Eu2_i := \frac{Z_{i,12}}{E\omega 2s_i} + 2 \cdot E\zeta 2_i \cdot \left(E\omega 2_i\right)^{-1} \cdot Z_{i,11} + Z_{i,10}$$

Algorithm -3: Using one point and two extra derivaitves

$i := 1 .. T$ Sets the number of iterations

$$E\omega 3s_i := \frac{\left[Z_{i,12} \cdot Z_{i,14} - \left(Z_{i,13}\right)^2\right]}{Z_{i,11} \cdot Z_{i,13} - \left(Z_{i,12}\right)^2} \quad \text{or} \quad E\omega 3_i := \sqrt{E\omega 3s_i}$$

$$E\zeta 3_i := -\left(E\omega 3_i \cdot \frac{Z_{i,11}}{2 \cdot Z_{i,12}} + \frac{Z_{i,13}}{2 \cdot E\omega 3_i \cdot Z_{i,12}}\right)$$

$$Eu3_i := \frac{Z_{i,12}}{E\omega 3s_i} + 2 \cdot \frac{E\zeta 3_i}{E\omega 3_i} \cdot Z_{i,11} + Z_{i,10}$$

Program (4)

This program is implemented to test the estimation accuracy of the three algorithms when noise (n) is added. The program can be used for the three categories of ω, ζ and u (contants, varaibles with 1st order dynamics or variable with 2nd order dynamics).

$G := 20$ sets the value of the gain (cut-off frequency in Hertz)

$G1 := G$ $G2 := G$ $G3 := G$ $G4 := G$

$time := 3$ Sets the simulation time in Seconds

$T := 2000$ Sets the total number of points Tolerance used =0.000001

$d := \frac{time}{T}$ Sets the integration step size $n := 0.1$ sets the value of the random number

$\omega o := 3$ $\zeta o := 0.5$ $uo := 30$

$\omega z := 3$ $\zeta z := 0.5$ $uz := .1$ set the desired values of the first level parameters of the sub-systems ω, ζ and u.

$\omega u := 3$ $\zeta u := 0.5$ $uu := 1$

Set initial conditions:

$x1_0 := 0$ $x2_0 := 0$ $Ex_0 := 0$ $Exd_0 := 0$ $Exdd_0 := 0$ $Extd_0 := 0$ $Exqd_0 := 0$ $N_0 := 0$

$\omega 1_0 := 10$ $\omega 2_0 := 0$ $\zeta 1_0 := 0.1$ $\zeta 2_0 := 0$ $u1_0 := 1$ $u2_0 := 0$

$i := 0 .. T$ Sets the number of iterations

$$
\begin{bmatrix} N_{i+1} \\ x1_{i+1} \\ x2_{i+1} \\ \omega 1_{i+1} \\ \omega 2_{i+1} \\ \zeta 1_{i+1} \\ \zeta 2_{i+1} \\ u1_{i+1} \\ u2_{i+1} \\ Ex_{i+1} \\ Exd_{i+1} \\ Exdd_{i+1} \\ Extd_{i+1} \\ Exqd_{i+1} \end{bmatrix} := \begin{bmatrix} N_i + 1000 \cdot \left[\left(rnd(n) - \frac{n}{2} \right) - N_i \right] \cdot d \\ x1_i + x2_i \cdot d \\ x2_i + \left[u1_i - 2 \cdot \zeta 1_i \cdot \left(\omega 1_i \right)^{-1} \cdot x2_i - x1_i \right] \cdot \left(\omega 1_i \right)^2 \cdot d \\ \omega 1_i + \omega 2_i \cdot d \\ \omega 2_i + \left[uo - 2 \cdot \zeta o \cdot (\omega o)^{-1} \cdot \omega 2_i - \omega 1_i \right] \cdot (\omega o)^2 \cdot d \\ \zeta 1_i + \zeta 2_i \cdot d \\ \zeta 2_i + \left[uz - 2 \cdot \zeta z \cdot (\zeta o)^{-1} \cdot \zeta 2_i - \zeta 1_i \right] \cdot (\zeta o)^2 \cdot d \\ u1_i + u2_i \cdot d \\ u2_i + \left[uu - 2 \cdot \zeta u \cdot (\omega u)^{-1} \cdot u2_i - u1_i \right] \cdot (\omega u)^2 \cdot d \\ Ex_i + G \cdot \left(x1_i + N_i - Ex_i \right) \cdot d \\ Exd_i + G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i \right) - Exd_i \right] \cdot d \\ Exdd_i + G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i \right) - Exd_i \right] - Exdd_i \right] \cdot d \\ Extd_i + G3 \cdot \left[G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i \right) - Exd_i \right] - Exdd_i \right] - Extd_i \right] \cdot d \\ Exqd_i + G4 \cdot \left[G3 \cdot \left[G2 \cdot \left[G1 \cdot \left[G \cdot \left(x1_i + N_i - Ex_i \right) - Exd_i \right] - Exdd_i \right] - Extd_i \right] - Exqd_i \right] \cdot d \end{bmatrix}
$$

Iterative Euler integrator for, noise source, time trajectory generator and high derivative estimator.

Note: from row 4 to row 9 can be modified to include rate of change of ω, ζ and u.

Now: check is being made to see how much Lag occurs when reducing the cut-off frequency (G).

```
I_x1 := | s←x1_0                         I_Ex := | s←Ex_0
        | for i ∈ 1..T                           | for i ∈ 1..T
        |     | s←x1_i                           |     | s←Ex_i
        |     | return i if s=max(x1)            |     | return i if s=max(Ex)
        | i                                      | i
```

$Lag := I\_Ex - I\_x1$ calculate the lag

$L := Lag$

Algorithm-1: Using Three points of x, x' and x''

$\Delta 1 := 1 \quad \Delta 2 := 2$ set the 2nd and the 3rd points

$i := 0 .. T - \Delta 2 - 3 \cdot L$ Sets the number of iterations

$$\Delta 12_i := \left(Ex_{i+L} - Ex_{i+L+\Delta 1}\right) \quad \Delta' 12_i := \left(Exd_{i+L\cdot 2} - Exd_{i+L\cdot 2+\Delta 1}\right) \quad \Delta'' 12_i := \left(Exdd_{i+3\cdot L} - Exdd_{i+3\cdot L+\Delta 1}\right)$$

$$\Delta 13_i := \left(Ex_{i+L} - Ex_{i+L+\Delta 2}\right) \quad \Delta' 13_i := \left(Exd_{i+L\cdot 2} - Exd_{i+L\cdot 2+\Delta 2}\right) \quad \Delta'' 13_i := \left(Exdd_{i+3\cdot L} - Exdd_{i+3\cdot L+\Delta 2}\right)$$

$$E\omega 1s_i := \frac{-\left[\left(\frac{\Delta'' 12_i}{\Delta' 12_i}\right) - \frac{\Delta'' 13_i}{\Delta' 13_i}\right]}{\left[\left(\frac{\Delta 12_i}{\Delta' 12_i}\right) - \frac{\Delta 13_i}{\Delta' 13_i}\right]} \quad \text{or} \quad E\omega 1_i := \sqrt{E\omega 1s_i}$$

$$E\zeta 1_i := \frac{-\left(\Delta 12_i \cdot E\omega 1_i + \frac{\Delta'' 12_i}{E\omega 1_i}\right)}{2 \cdot \Delta' 12_i}$$

$$Eu1_i := \frac{Exdd_{i+3\cdot L}}{E\omega 1s_i} + 2 \cdot \frac{E\zeta 1_i}{E\omega 1_i} \cdot Exd_{i+L\cdot 2} + Ex_{i+L}$$

Algorithm 2, using two points and one extra derivative

$\Delta := 1$ sets the 2nd point

$i := 0 .. T - \Delta - 4 \cdot L$ Sets the number of iterations

$$\Delta 12_i := \left(Ex_{i+L} - Ex_{i+L+\Delta}\right) \qquad \Delta' 12_i := \left(Exd_{i+L\cdot 2} - Exd_{i+L\cdot 2+\Delta}\right)$$

$$\Delta'' 12_i := \left(Exdd_{i+3\cdot L} - Exdd_{i+3\cdot L+\Delta}\right) \qquad \Delta''' 12_i := \left(Extd_{i+4\cdot L} - Extd_{i+4\cdot L+\Delta}\right)$$

$$E\omega 2s_i := \frac{-\left[\Delta' 12_i \cdot \Delta''' 12_i - \left(\Delta'' 12_i\right)^2\right]}{\left[\left(\Delta' 12_i\right)^2 - \Delta 12_i \cdot \Delta'' 12_i\right]} \quad \text{or} \quad E\omega 2_i := \sqrt{E\omega 2s_i}$$

$$E\zeta 2_i := \frac{-\left(\dfrac{\Delta 12_i}{\Delta' 12_i} + \dfrac{\Delta'' 12_i}{E\omega 2s_i \cdot \Delta' 12_i}\right)}{2 \cdot \left(E\omega 2_i\right)^{-1}}$$

$$Eu2_i := \frac{Exdd_{i+3\cdot L}}{E\omega 2s_i} + 2 \cdot \frac{E\zeta 2_i}{E\omega 2_i} \cdot Exd_{i+L\cdot 2} + Ex_{i+L}$$

Algorithm 3, using one point and two extra derivatives

$i := 0 .. T - 5 \cdot L$ Sets the number of iterations

$$E\omega 3s_i := \frac{\left[Exdd_{i+3\cdot L} \cdot Exqd_{i+5\cdot L} - \left(Extd_{i+4\cdot L}\right)^2\right]}{Exd_{i+L\cdot 2} \cdot Extd_{i+4\cdot L} - \left(Exdd_{i+3\cdot L}\right)^2} \quad \text{or} \quad E\omega 3_i := \sqrt{\left| E\omega 3s_i \right|}$$

$$E\zeta 3_i := -\left(E\omega 3_i \cdot \frac{Exd_{i+L\cdot 2}}{2 \cdot Exdd_{i+3\cdot L}} + \frac{Extd_{i+4\cdot L}}{2 \cdot E\omega 3_i \cdot Exdd_{i+3\cdot L}}\right)$$

$$Eu3_i := \frac{Exdd_{i+3\cdot L}}{E\omega 3s_i} + 2 \cdot \frac{E\zeta 3_i}{E\omega 3_i} \cdot Exd_{i+L\cdot 2} + Ex_{i+L}$$

Program (5)

This program is implemented to provide an estimate of the first level parameters of the sub-systems ω, ζ and u using the three algorithms.

$G := 10$ sets the value of the gain (cut-off frequency in Hertz)

$G1 := G$ $G2 := G$ $G3 := G$ $G4 := G$

$time := 3$ Sets the simulation time in Seconds

$T := 2000$ Sets the total number of points Tolerance used =0.000001

$d := \frac{time}{T}$ Sets the integration step size

$\omega o := 3$ $\zeta o := 0.5$ $uo := 10$

$\omega z := 3$ $\zeta z := 0.5$ $uz := .1$

$\omega u := 3$ $\zeta u := 0.5$ $uu := 1$

set the desired values of the first level parameters of the sub-system ω, ζ and u.

Set initial conditions:

$x1_0 := 0$ $x2_0 := 0$ $Ex_0 := 0$ $Exd_0 := 0$ $Exdd_0 := 0$ $Extd_0 := 0$ $Exqd_0 := 0$

$\omega 1_0 := 10$ $\omega 2_0 := 0$ $\zeta 1_0 := 0.1$ $\zeta 2_0 := 0$ $u1_0 := 1$ $u2_0 := 0$

$i := 0 .. T$ Sets the number of iterations

$$\begin{bmatrix} x1_{i+1} \\ x2_{i+1} \\ \omega 1_{i+1} \\ \omega 2_{i+1} \\ \zeta 1_{i+1} \\ \zeta 2_{i+1} \\ u1_{i+1} \\ u2_{i+1} \\ Ex_{i+1} \\ Exd_{i+1} \\ Exdd_{i+1} \\ Extd_{i+1} \\ Exqd_{i+1} \end{bmatrix} := \begin{bmatrix} x1_i + x2_i \cdot d \\ x2_i + \left[u1_i - 2\cdot\zeta 1_i \cdot (\omega 1_i)^{-1} \cdot x2_i - x1_i\right]\cdot(\omega 1_i)^2 \cdot d \\ \omega 1_i + \omega 2_i \cdot d \\ \omega 2_i + \left[uo - 2\cdot\zeta o\cdot(\omega o)^{-1}\cdot\omega 2_i - \omega 1_i\right]\cdot(\omega o)^2\cdot d \\ \zeta 1_i + \zeta 2_i \cdot d \\ \zeta 2_i + \left[uz - 2\cdot\zeta z\cdot(\zeta o)^{-1}\cdot\zeta 2_i - \zeta 1_i\right]\cdot(\zeta o)^2\cdot d \\ u1_i + u2_i \cdot d \\ u2_i + \left[uu - 2\cdot\zeta u\cdot(\omega u)^{-1}\cdot u2_i - u1_i\right]\cdot(\omega u)^2\cdot d \\ Ex_i + G\cdot(x1_i - Ex_i)\cdot d \\ Exd_i + G1\cdot\left[G\cdot(x1_i - Ex_i) - Exd_i\right]\cdot d \\ Exdd_i + G2\cdot\left[G1\cdot\left[G\cdot(x1_i - Ex_i) - Exd_i\right] - Exdd_i\right]\cdot d \\ Extd_i + G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot(x1_i - Ex_i) - Exd_i\right] - Exdd_i\right] - Extd_i\right]\cdot d \\ Exqd_i + G4\cdot\left[G3\cdot\left[G2\cdot\left[G1\cdot\left[G\cdot(x1_i - Ex_i) - Exd_i\right] - Exdd_i\right] - Extd_i\right] - Exqd_i\right]\cdot d \end{bmatrix}$$

Iterative Euler integrator for, time trajectory generator and time higher derivative estimators.

Note: from row 3 to row 8 can be modified to include rate of change of ω, ζ and u.

Algorithm-1: Using Three points of x, x' and x''

$\Delta 1 := 1 \quad \Delta 2 := 2$ set the 2nd and the 3rd points

$i := 0 .. T - \Delta 2$ Sets the number of iterations

$\Delta 12_i := (Ex_i - Ex_{i+\Delta 1})$ $\quad \Delta' 12_i := (Exd_i - Exd_{i+\Delta 1})$ $\quad \Delta'' 12_i := (Exdd_i - Exdd_{i+\Delta 1})$

$\Delta 13_i := (Ex_i - Ex_{i+\Delta 2})$ $\quad \Delta' 13_i := (Exd_i - Exd_{i+\Delta 2})$ $\quad \Delta'' 13_i := (Exdd_i - Exdd_{i+\Delta 2})$

$$E\omega 1s_i := \frac{-\left[\left(\frac{\Delta'' 12_i}{\Delta' 12_i}\right) - \frac{\Delta'' 13_i}{\Delta' 13_i}\right]}{\left[\left(\frac{\Delta 12_i}{\Delta' 12_i}\right) - \frac{\Delta 13_i}{\Delta' 13_i}\right]}$$

or

$$E\omega 1_i := \sqrt{E\omega 1s_i}$$

$$E\zeta 1_i := \frac{-\left(\Delta 12_i \cdot E\omega 1_i + \frac{\Delta'' 12_i}{E\omega 1_i}\right)}{2 \cdot \Delta' 12_i}$$

$$Eu1_i := \frac{Exdd_i}{E\omega 1s_i} + 2 \cdot \frac{E\zeta 1_i}{E\omega 1_i} \cdot Exd_i + Ex_i$$

Algorithm 2, using two points and one extra derivative

$\Delta := 1$ sets the 2nd point

$i := 0 .. T - \Delta$ Sets the number of iterations

$\Delta 12_i := (Ex_i - Ex_{i+\Delta})$ $\quad \Delta' 12_i := (Exd_i - Exd_{i+\Delta})$

$\Delta'' 12_i := (Exdd_i - Exdd_{i+\Delta})$ $\quad \Delta''' 12_i := (Extd_i - Extd_{i+\Delta})$

$$E\omega 2s_i := \frac{-\left[\Delta' 12_i \cdot \Delta''' 12_i - (\Delta'' 12_i)^2\right]}{\left[(\Delta' 12_i)^2 - \Delta 12_i \cdot \Delta'' 12_i\right]}$$

or

$$E\omega 2_i := \sqrt{E\omega 2s_i}$$

$$E\zeta 2_i := \frac{-\left(\frac{\Delta 12_i}{\Delta' 12_i} + \frac{\Delta'' 12_i}{E\omega 2s_i \cdot \Delta' 12_i}\right)}{2\cdot\left(E\omega 2_i\right)^{-1}}$$

$$Eu2_i := \frac{Exdd_i}{E\omega 2s_i} + 2\cdot\frac{E\zeta 2_i}{E\omega 2_i}\cdot Exd_i + Ex_i$$

Algorithm 3, using one point and two extra derivatives

$i := 0 .. T$ Sets the number of iterations

$$E\omega 3s_i := \frac{\left[Exdd_i\cdot Exqd_i - \left(Extd_i\right)^2\right]}{Exd_i\cdot Extd_i - \left(Exdd_i\right)^2} \quad \text{or} \quad E\omega 3_i := \sqrt{E\omega 3s_i}$$

$$E\zeta 3_i := -\left(E\omega 3_i\cdot\frac{Exd_i}{2\cdot Exdd_i} + \frac{Extd_i}{2\cdot E\omega 3_i\cdot Exdd_i}\right)$$

$$Eu3_i := \frac{Exdd_i}{E\omega 3s_i} + 2\cdot\frac{E\zeta 3_i}{E\omega 3_i}\cdot Exd_i + Ex_i$$

Now: Generation of the higher derivatives of the filter estimate of the sub-system ω:

Set initail conditions:

$EE\omega 3_0 := 0$ $EE\omega d3_0 := 0$ $EE\omega dd3_0 := 0$ $EE\omega td3_0 := 0$ $EE\omega qd3_0 := 0$

$G := 20$ $G1 := G$ $G2 := G1$ $G3 := G2$ $G4 := G3$

$i := 0..T$ Sets the number of iterations

$$\begin{bmatrix} EE\omega3_{i+1} \\ EE\omega d3_{i+1} \\ EE\omega dd3_{i+1} \\ EE\omega td3_{i+1} \\ EE\omega qd3_{i+1} \end{bmatrix} := \begin{bmatrix} EE\omega3_i + G\cdot(E\omega3_i - EE\omega3_i)\cdot d \\ EE\omega d3_i + G1\cdot[G\cdot(E\omega3_i - EE\omega3_i) - EE\omega d3_i]\cdot d \\ EE\omega dd3_i + G2\cdot[G1\cdot[G\cdot(E\omega3_i - EE\omega3_i) - EE\omega d3_i] - EE\omega dd3_i]\cdot d \\ EE\omega td3_i + G3\cdot[G2\cdot[G1\cdot[G\cdot(E\omega3_i - EE\omega3_i) - EE\omega d3_i] - EE\omega dd3_i] - EE\omega td3_i]\cdot d \\ EE\omega qd3_i + G4\cdot[G3\cdot[G2\cdot[G1\cdot[G\cdot(E\omega3_i - EE\omega3_i) - EE\omega d3_i] - EE\omega dd3_i] - EE\omega td3_i] - EE\omega qd3_i]\cdot d \end{bmatrix}$$

Iterative Euler routine to generate higher derivative estimators of Eω3.

Now: check is being made to see how much Lag occurs when reducing the cut-off frequency (G).

```
I_Eω3 := | s ← Eω3_0
         | for i ∈ 1..T
         |   | s ← Eω3_i
         |   | return i if s = max(Eω3)
         | i

I_Eωd := | s ← EEωd3_0
         | for i ∈ 1..T
         |   | s ← EEωd3_i
         |   | return i if s = max(EEωd3)
         | i
```

$Lag := I\_E\omega d - I\_E\omega3$ calculate the lag

$L := Lag$

Agorithm 3 is being used to produce an estimate of the first level parameters of the sub-system ω (ωo, ζo and uo):

$i := 0..T - 5\cdot L$ Sets the number of iterations

$$E\omega o3s_i := \frac{\left[EE\omega dd3_{i+3\cdot L}\cdot EE\omega qd3_{i+5\cdot L} - \left(EE\omega td3_{i+4\cdot L}\right)^2\right]}{EE\omega d3_{i+2\cdot L}\cdot EE\omega td3_{i+4\cdot L} - \left(EE\omega dd3_{i+3\cdot L}\right)^2}$$ or $E\omega o3_i := \sqrt{E\omega o3s_i}$

$$E\zeta o3_i := -\left(E\omega o3_i\cdot\frac{EE\omega d3_{i+2\cdot L}}{2\cdot EE\omega dd3_{i+3\cdot L}} + \frac{EE\omega td3_{i+4\cdot L}}{2\cdot E\omega o3_i\cdot EE\omega dd3_{i+3\cdot L}}\right)$$

$$Euo3_i := \frac{EE\omega dd3_{i+3\cdot L}}{E\omega o3s_i} + 2\cdot\frac{E\zeta o3_i}{E\omega o3_i}\cdot EE\omega d3_{i+2\cdot L} + EE\omega3_{i+L}$$