THE APPLICATION OF H_∞ CONTROLLER SYNTHESIS TO HIGH SPEED INDEPENDENT DRIVE SYSTEMS

ROBERT WILLIAM BEAVEN

Submitted for the Degree of
Doctor of Philosophy

THE UNIVERSITY OF ASTON IN BIRMINGHAM

August 1995

THE UNIVERSITY OF ASTON IN BIRMINGHAM

THE APPLICATION OF $H\infty$ NORM CONTROLLER SYNTHESIS TO HIGH SPEED INDEPENDENT DRIVE SYSTEMS.

by ROBERT WILLIAM BEAVEN

Doctor of Philosophy, 1995

SUMMARY

This thesis describes work completed on the application of $H\infty$ controller synthesis to the design of controllers for single axis high speed independent drive design examples. $H\infty$ controller synthesis was used in a single controller format and in a self-tuning regulator, a type of adaptive controller.

Three types of industrial design examples were attempted using $H\infty$ controller synthesis, both in simulation and on a Drives Test Facility at Aston University. The results were benchmarked against a Proportional, Integral and Derivative (PID) with velocity feedforward controller (VFF), the industrial standard for this application. An analysis of the differences between a $H\infty$ and PID with VFF controller was completed.

A direct-form $H\infty$ controller was determined for a limited class of weighting function and plants which shows the relationship between the weighting function, nominal plant and the controller parameters. The direct-form controller was utilised in two ways. Firstly it allowed the production of simple guidelines for the industrial design of $H\infty$ controllers. Secondly it was used as the controller modifier in a self-tuning regulator (STR). The STR had a controller modification time (including nominal model parameter estimation) of 8ms.

A Set-Point Gain Scheduling (SPGS) controller was developed and applied to an industrial design example.

The applicability of each control strategy, PID with VFF, $H\infty$, SPGS and STR, was investigated and a set of general guidelines for their use was determined.

All controllers developed were implemented using standard industrial equipment.

Keywords: $H\infty$ Controller Synthesis; Self-Tuning Regulator; High-Speed Independent Drives

ACKNOWLEDGEMENTS

I would like to thank Professor M.T.Wright for his excellent supervision of this project. My kind thanks go to my associate supervisor Dr. S.D.Garvey and Dr. R. Johnson for his assistance.

Thanks to Ms L.D.Wilkes and her associate supervisor Dr. M.I.Friswell for their assistance on the identification section of this thesis.

Thanks to Mr J.Jeffs, Mr P.Roach and Mr G. Morgan for their assistance in the practical parts of this thesis.

Thanks must also go to the and Engineering and Physical Sciences Research Council and Molins plc. for funding this research. Special thanks, in particular to the following members of Molins plc., Mr M.Cahill, Dr.D.R.Seaward and Mr P.Thorton for their assistance in determining realistic industrial design examples and invaluable suggestions.

LIST OF CONTENTS Page

LIST OF TABLES

Table Page

Chapter 7

LIST OF FIGURES

Chapter 6

Chapter 7

ABBREVIATIONS AND NOTATION

A	...	Gain of Controller Position Loop
ACSL	...	Advanced Continuous Simulation Language.
B_v	...	Viscous Friction (Nm/rads^{-1}).
CM	...	Covariance Matrix.
DM	...	Drive Module.
EMF	...	Electro-Motive Force.
EMI	...	Electro-Magnetic Interference.
EP	...	Estimated Parameters.
FF	...	Forgetting Factor.
FFW	...	Forgetting Factor Weight.
G	...	Plant Transfer Function.
G_a	...	Gain of Controller Velocity Loop
G_o	...	Nominal Model (Identified Model).
I_{rms}	...	Root Mean Square Current (A).
J_l	...	Load Inertia (kgm^2).
J_m	...	Motor Inertia (kgm^2).
J_t	...	Total Driven Inertia (kgm^2).
K_r	...	Controller Transfer Function.
K_m	...	Gain Margin
K	...	Ratio of Natural to Sampling Frequency
K_{sp}	...	Torsional Spring Constant (Nm/Rad)
K_t	...	Motor Torque Constant (Nm/A).
L	...	Open-Loop Gain - GK
L_o	...	Nominal Open-Loop Gain - G_oK
LHP	...	Left half-plane.
LQG	...	Linear Quadratic Gaussian.
LTR	...	Loop Transfer Recovery.
L_{∞}	...	Control Hamiltonian Matrix.
MIMO	...	Multiple-input, Multiple-output.
P	...	Power (W).
P(s)	...	Augmented State-Space Problem Representation.
PID	...	Proportional, Integral and Derivative
RHP	...	Right half-plane
R_{ϕ}	...	Motor Armature Resistance Per Phase (Ω)
R_{TH}	...	Thermal Resistance (°CW^{-1}).
S	...	Sensitivity Function.
S_{id}	...	Ideal Sensitivity Function.

S_o	...	Nominal Sensitivity Function.
SPGS	...	Set-Point Gain Scheduling.
SISO	...	Single Input Single Output System.
SSE	...	Strict System Equivalence (transformation).
STR	...	Self-Tuning Regulator.
T	...	Complementary Sensitivity Function.
T_{id}	...	Ideal Complementary Sensitivity Function.
T_o	...	Nominal Complementary Sensitivity Function.
T_g	...	Torque generated by the motor (Nm).
T_f	...	Friction Torque (Nm).
T_l	...	Load Torque (Nm).
T_{rms}	...	Root Mean Square Torque.
T_Δ	...	Temperature Rise(°C).
VFF	...	Velocity Feed-Forward.
W_1	...	Performance Weighting Function for design of a $H\infty$ controller.
W_2	...	Robustness Weighting Function for design of a $H\infty$ controller.
$X\infty$	...	Control Riccati Equation Solution
$Y\infty$	...	Observer Riccati Equation Solution.
$Z\infty$	...	Observer Hamiltonian Matrix.
iff	...	if and only if.
mmf	...	Magnetomotive Force.
p	...	Number of pulses per revolution from an encoder
rms	...	Root Mean Square
sup	...	supremum.
t	...	Time (in sampling intervals).
e(t)	...	Error between plant output and reference input at time t.
r(t)	...	Reference input to closed-loop system at time t.
u(t)	...	Output from the controller at time t.
y(t)	...	Output from the plant at time t.
x_t	...	Set of plant inputs and outputs for use in identification algorithm.
Δ_a	...	Additive Perturbation.
Δ_o	...	Multiplicative Perturbation.
$\Delta_n\Delta_m$	...	Coprime Factorisation Perturbations.
ΔT	...	Encoder pulse count sampling time.
ω	...	Frequency (Rads^{-1}).
ω_r	...	Resonant Frequency (Rads^{-1}).
$\overline{\sigma}$	...	Maximum Singular Value.
$\underline{\sigma}$	...	Minimum Singular Value.

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

This thesis investigates the applicability of $H\infty$ controller synthesis to the individual axes of high speed machinery with the objective of improving the performance beyond that achievable with a benchmark industrial control algorithm (A Proportional, Integral and Derivative (PID) position controller with Velocity Feedforward (VFF)). The thesis details how the technique was applied to industrial design examples (all with incremental demand profiles) with a single controller and with a self-tuning regulator. Additionally, a set-point gain scheduling controller was developed for single axis high-speed independent drives. To allow the direct transfer of any control strategies developed, all controller implementation was completed using standard industrial control equipment and the controller design techniques were developed to be at least moderately user friendly.

Traditional high speed machines utilise a single power source to drive a number of separate motions via mechanical transmissions such as gears, belts and cams. The relative movements of the axes are usually co-ordinated through fixed mechanical interfaces. The approach has the inherent problem of inflexibility, and leads to restrictive design procedures. Seaward [1989] demonstrated that the alternative approach of using independent drives in high-speed machinery was a viable option.

Current state of the art high-speed machinery often utilises independent drives where the relative and absolute position of each axis are determined through software controllers, rather than via stiff mechanical linkages. The benefits of using independent drives in high-speed machinery are well established. They allow much more rapid development of the machines themselves, they facilitate testing and maintenance through their modularity and they provide a degree of configurability on the final machine, through software, which could not be contemplated in a machine with purely mechanical linkages. Additionally acoustic noise, maintenance and running costs are usually reduced.

The control of individual axes is one of the factors which limits the performance of high-speed independent drive machinery. The absolute positional accuracy of an individual axis limits the relative positional accuracy of machine axes and hence the overall machine performance. The characteristics of brushless dc servo systems, used as independent drives, together with the application dynamics require that the machines be treated as time-varying non-linear systems.

The response of dc servo systems, used as high-speed independent drives, is characterised by a number of features [Seaward 1989]:-

1) The achievable system bandwidth is limited by time delays inherent to dc servo systems.
2) The achievable system bandwidth is limited by the response of the internal digital controllers within the servo systems.
3) The behaviour of internal constraints (e.g. current limit, voltage limit etc.) usually causes the dc servo systems to operate in a non-linear manner. This is particularly true of high-performance servos which are normally required to operate at, or near, maximum outputs for substantial portions of the event cycle.
4) The dynamics of dc servo systems alter dramatically with environmental variations (i.e. temperature change or under forcing conditions).
5) dc servo systems have inherent limits on their achievable accuracy, owing to the limited resolution of the optical encoder, and physical limits on achievable acceleration, velocity and position response.

6) The response of a dc servo system is position dependent, due to saliency effects.

The high speed machinery, in which brushless dc servos are most frequently used, is characterised by a number of features:-

1) System dynamics vary considerably over each event cycle.
2) Event cycles have short-time periods, usually measured in 10's of milliseconds.
3) Required tracking error is often extremely low; for example, an axis accelerating at 60 rads^{-2} may have a continuous position tracking error requirement of less than 0.004 rads.
4) To meet modern manufacturing requirements, machine processes must exhibit a high degree of repeatability and reliability.
5) To allow rapid changes in the manufacturing process, machine processes must be modular.
6) Strict size, weight and cost criteria usually apply.
7) Control system design targets, in terms of individual axis interrelation and tracking error requirements may be known only provisionally.
8) Mechanical resonances are common (effectively limiting the obtainable bandwidth).
9) To achieve the required load positional accuracy torsionally-stiff direct coupling between the motor and load is often used, rendering the absolute and relative positional accuracy of each shaft crucial (to ensure no damage to the drive).

For a controller to effectively control high speed machinery it must be able to cope with a high level of unknown system parameter variation within the event cycle (for example due to load variation) and also over longer time periods (due for example, to temperature variation). In addition, unknown sets of disturbances must be coped with, while still addressing performance criteria. Two key features of $H\infty$ controller synthesis (discussed further in section 1.3) address these requirements and make the technique appropriate for this application:

1) a sufficient condition for robust performance against a set of plant perturbations can be determined as the solution of a single problem.
2) the system inputs (including disturbances) do not have to be known deterministically.

$H\infty$ controller synthesis can be used to determine a stabilising controller for a system if the error in the initial modelling of the system and any variation within the system dynamics can be estimated. However if the variation in the plant dynamics are large a robustness weighting function will have to be selected such that a large stability margin will be produced. The use of such a robustness weighting function will limit the attainable performance of the system (i.e. the inherent trade-off between performance and stability which is present in all controlled systems). Also as the plant dynamics vary the closed-loop dynamics will alter and hence the performance of the system may deteriorate. In such cases an adaptive controller may improve the system performance. The aim of adaptive control is to improve the performance of a system by adapting controller parameters as plant dynamics vary. A single axis self-tuning regulator (STR), was developed to investigate whether any performance improvement could be achieved for this type of application. The self-tuning regulator was developed using simplified $H\infty$ controller synthesis together with recursive least squares parameter estimation (with variable forgetting factor) algorithms. A set-point gain scheduling (SPGS) controller was developed to improve single axis performance. Both the STR and SPGS controller were developed to cope with system variations within the event cycle.

The aims of this thesis are to investigate the applicability of $H\infty$ controller synthesis and a self-tuning regulator to high speed independent drive machinery, using single axis industrial design examples.

1.2 THESIS ORGANISATION

This thesis is organised into 9 chapters which are summarised below

Chapter 1 Introduction

This chapter describes the motivation and contributions of the thesis. An extensive literature review is given placing the completed research into context.

Chapter 2 $H\infty$-norm Controller Synthesis and Parameter Estimation.

The basic theory of $H\infty$ controller synthesis is examined and explained. The theory of unstructured and structured uncertainty and forms of perturbation (i.e. additive, multiplicative and coprime factor) is given. A simple proof of the guaranteed phase and gain margins is given. The state-space controller determination algorithm [Glover and Doyle 1988 and Doyle *et al* . 1989] in conjunction with the simplifying transformations detailed by Safonov *et al.* [1989] is given. The theory of least squares parameter estimation is briefly outlined.

Chapter 3 System Limitations and Simulations.

A description is given of the test equipment used. The inherent limitations on the system performance due to non-linearities, time delays, quantisation errors and sampling effects are described. A comparative study of the ACSL simulations (used in the initial controller design work) and the actual test rig is given. The use of a second order model to approximate the dc servo systems is justified.

Chapter 4 Initial Design Example Using H∞ Controller Synthesis.

An industrial design example is detailed. A H∞ controller designed to meet the performance requirement, both in simulation and on a practical system, is given. The H∞ controller results are compared to a traditional PID with VFF controller. An analysis of the differences between the two controllers is given.

Chapter 5 Simplifications of H∞ Controller Synthesis and Recursive Least Squares Parameter Estimation Algorithms.

This chapter details the determination of the direct-form H∞ controller. The direct-form H∞ controller shows the relationship between the nominal plant, weighting functions and controller parameters. The steps used to determine the direct-form controller are enumerated. These consist of the initial establishment of the augmented plant state-space representation; implementing the simplifications suggested by Safonov *et al.* [1989]; the formation of the required Hamiltonian matrices; the solution of the Riccati Equations, using the eigenvalue method, and the final redetermination of the controller. The conditions for one of the Riccati equation solutions to be the zero matrix (which provides a dramatic reduction in computational requirements) are given. Valid simplifications made to reduce the complexity of a recursive least squares parameter estimation technique with variable forgetting factor, are also developed.

Chapter 6 Weighting Function Selection.

An analysis of performance weighting function selection, for the direct-form H∞ controller, using an endpoint error term, is given. Guidelines are given for the redetermination of the performance weighting function to achieve specified closed-loop performance. Two design case studies are analysed. These high-light the theoretical aspects of weighting function selection. A computer program developed, to search for optimal (limiting) weighting function parameters, is described. A three term time-series controller is also described.

Chapter 7 Adaptive Control.

The theory of the set-point gain scheduling controller and its application to an industrial design example is given. The direct-form H∞ controller is combined with a simplified recursive least squares parameter estimation technique to produce a self-tuning regulator (STR). The STR is implemented on an industrial design example. A discussion of the stability and performance of the self-tuning regulator is given. The identifier performance is discussed in terms of two identification error terms the relative magnitude error (RME) and absolute phase shift error (APE).

Chapter 8. Controller Application.

The four control techniques (PID with VFF, H∞ control, set-point gain scheduling and a self-tuning regulator) which are implemented in this thesis are reviewed together with a three term time-series controller. Guidelines on the applicability and limitations of each controller are given in terms of industrial applications.

Chapter 9. Conclusions and Suggestions For Further Work.

The chapter contains concluding remarks and suggestions for further research.

1.3 RELEVANT LITERATURE

During the literature review no author was found who had attempted to apply H∞ controller synthesis to high speed independent drive systems, either as a single controller or as a self-tuning regulator. This research does however overlap into three areas of research: dc servo systems, $H\infty$ controller synthesis and adaptive control systems. Only the relevant contributions of each research area, in the context of this thesis, are discussed.

1.3.1 dc Servo Systems

Since the work of Seaward [1989] the use of dc servo systems as independent drives in high speed machinery has become commonplace. Seaward identified the possible benefits of this approach as a reduction in costs (maintenance, running and manufacturing), increased flexibility, reduced acoustic emissions, overall energy savings, reduced design time and the opportunity for an effective overall control strategy. These possible benefits could lead to a competitive advantage and increase in profits for the machine manufacturer and user. Seaward analysed the inherent limitations on dc servo performance, for example those due to sampling delays (all the limitations identified by Seaward are discussed in Chapter 3). Seaward also developed a series of Advanced Continuous Simulation Language (ACSL) [Mitchell and Gauthier 1991] simulations (used for the initial testing of controllers and control strategies) of the Electro-Craft™ Bru 500 series of dc servo systems.

The brushless dc servo systems consist of a power supply module, a personality module and a permanent magnet synchronous motor (with rare earth neodymium-iron-boron magnets) [Electro-Craft™ 1987]. The power supply module contains an electronic inverter which takes input power from a dc source and supplies polyphase alternating currents to the stator (armature) windings. Sensors signal the position of the rotor (field) of the synchronous machine to the personality module which controls the electronics controlling the switching elements in the inverter [McPherson and Laramore 1990]. As Tomasek [1989] argued, some confusion exists over the classification of servo systems as being ac or dc since, at least theoretically, the motor voltages and currents are sinusoidal. However Kusko and Peeran [1988] developed the following definition of a brushless dc motor (and hence brushless dc servo system):

> A motor having a stator (armature) winding and a permanent-magnet (PM) or salient-pole soft-iron rotor. The stator windings are supplied from a primary dc supply through a matrix of solid state switches, which are controlled by a rotor shaft position sensors and logic. In the absence of a regulator, the motor speed is approximately proportional to the primary dc voltage.

The above definition ignores the speed or torque regulator and other features which are not necessary to distinguish the dc motor from other motor types. The last part of the definition, referring to the motor speed being approximately proportional to the primary dc voltage, of course implies that the motor will run up to a speed such that the back EMF is equal to the primary dc voltage (motor back EMF is proportional to speed for a given motor). Hence the servo systems used in this thesis are clearly definable as brushless dc servo systems.

The advantages of brushless dc servo systems are:-

1) Long life and high level of reliability.
2) Ability to operate in a contaminated environment or in a vacuum.
3) High speed operation.
4) Low rotor inertia (hence potentially high acceleration rates).
5) No power limit due to mechanical commutation (limit due to power electronics or stator windings).
6) No electromagnetic noise associated with brushes.
7) No rotor winding resistance losses or magnetising mmf requirement.
8) Ease of control (compared with dc machines).
9) Bandwidth well in excess of most conventional systems.

10) Peak torque is 2-3 times continuous torque (as allowed by the motor manufacturers, the actual value may be higher).

11) Deceleration torque is enhanced because back EMF and supply voltage are additive in the deceleration mode.

12) Motor size and weight are low compared to commutated dc machines.

A servo system monitors the condition of an output variable by comparing it to an input command. The system attempts to adjust the output such that there is zero error. The most common quantities used as controlled variables are position, speed or torque. For the particular application of motion control, position is of primary interest. Servo systems generally operate in one of two modes: incremental or continuous [Seaward 1989]. In incremental mode, the demand changes in discrete steps with a static condition between steps (often referred to as a dwell). In continuous mode the command variable remains constant or changes smoothly. In this thesis dc servo systems are used in incremental mode. Seaward [1989] suggested that incremental profiles should usually be smoothly curved to minimise energy dissipation, in order to prevent possible overheating of the motor.

The absolute positional accuracy and relative position of each independent drive is critical, in terms of ensuring both product fidelity and non-failure of the machine due to axis collision. The requirement that the motion of several axes be synchronised dictates that either a multi-input multi-output (MIMO) controller is used or that single axes have local single-input single-output (SISO) position control loops which receive reference signals from a higher level controller. High performance MIMO controllers are extremely difficult to design for synchronised applications, with non-linear time varying characteristics, and, in general, an acceptable solution can only be found using local position control loops if the performance of each controller, and hence each axis, is of a high order. Consequently the control of individual axes is of crucial importance. The standard industrial method for the commissioning of a multi-axis control scheme is to optimise the control of each individual axis (in terms of positional tracking error). A local control scheme, such as the master / slave scheme, is then used to synchronise the axes [Tal 1989].

1.3.2 The Design of Motion Control Systems Using Servo Drives.

The standard control technique normally used for individual drive axes is PID with VFF (this form of control will be treated as the benchmark). Proportional control is used to increase the closed-loop system bandwidth by feeding back an actuating signal into the plant which is proportional to the system error. Integral control is routinely used in conjunction with proportional control to reduce the steady-state error by including a proportion of the integral of the system error in the actuating signal. The derivative term is used to increase the speed of response of the system by including a term proportional to the rate of change of system error in the actuating signal. VFF is used to 'drive' the system during demanded positional movements by providing an additional component to the actuation proportional to the rate of change of required system output. VFF does not affect closed-loop stability and behaves like an external disturbance which enhances performance. To avoid overshoot a typical VFF gain value chosen for rapid incremental movements is 0.75 (which for the system set-up used at Aston University would be in volts per demanded velocity in encoder counts). If the priority is to reduce steady-state positional tracking error a value of unity is often selected. The use of VFF has been found to dramatically increase the performance of dc servo systems [Seaward and Vernon 1991].

The PID parameters can be determined using the Ziegler-Nichols [Ziegler and Nichols 1942] tuning method. Typically in practice, however, the performance specifications are defined and then the controller parameters are modified iteratively until the best compromise controller is achieved. Every iteration of controller parameter adjustment is followed by some analysis based on experience and empirical rules of thumb, which are used to determine the next iteration. Stability analysis on the system normally entails the frequency domain analysis techniques of Nyquist and Bode [Nyquist 1932, Bode 1947]. However increased demands for absolute and relative positional accuracy have exposed inadequacies in the current practice of iteratively adjusting the controller parameters to obtain the required performance, in certain instances [Seaward 1993]. Additionally the normal process of iterative tuning of a PID with VFF controller does not address stability and performance criteria in a unified manner. More advanced control techniques are therefore relevant.

The most influential control technique researched through the 1970's was Linear Quadratic Gaussian (LQG), which Newton *et al.* [1957] used to directly obtain the "optimal" controller parameters as an alternative to the previously used method of trial and error. This approach has inherent problems however: disturbances must be quantified accurately, a controller is produced which minimises the rms error of control signals (which is inadequate when maximum error is of key importance, as in many servo drive application) and the controllers produced using this technique do not have a guaranteed stability margin which makes them inadequate for some applications [Chiang and Safonov 1992]. Zames [1981] suggested using a $H\infty$-norm to determine a controller which simultaneously satisfies both stability and performance criteria. The method [Zames 1981, Zames and Francis 1983] considers the disturbances to be a set of norm bounded finite energy signals, avoiding the necessity for accurate quantification of disturbances. Zames [1981] argued that minimisation of the worst case energy gain ($H\infty$-norm), from the set of disturbances through to the output, was more appropriate than the minimisation of the worst average error achieved by LQG. Han and Hsia [1990] further argued that the unified approach to stability and performance, which is implicit in $H\infty$ controller synthesis, overcomes the lack of robustness in LQG without introducing the problems of noise amplification endemic in the Loop Transfer Recovery (LTR) technique, which is used to introduce robustness into the LQG design.

The theory of H∞ provides an intuitive and simple extension to the classical control technique concepts of gain and distance to the Nyquist point [Foo 1986].

The original methods of determining the solution to the H∞ design process (i.e. the determination of H∞ controllers) [Francis 1987 and Vidyasagar 1985] were extremely complex requiring the solution of a Nehari distance problem and coprime factorisations. Glover and Doyle [1988] and Doyle *et al.* [1989] developed a conceptually and numerically simpler, iterative state-space method for the design problem of a H∞ controller which required finding the positive semi-definite solutions of a pair of Riccati Equations (assuming such solutions exist). The level of analytical complexity in this method is such that it has the advantage of being "readily" understood. The state-space method required a number of assumptions to be satisfied by the augmented state-space representation of the problem (the initial representation used in the controller design). Safonov *et al.* [1989] used a series of strict system equivalence (SSE) transformations together with general invertible changes of variables which allow the simple state-space method to be applied to a general class of plants and weighting functions, including those discussed in this thesis. Hvostov [1990] demonstrated that the solution to one of the necessary Riccati Equations was the zero matrix provided that the plant was open-loop stable. In such a case only one Riccati Equation solution is required, which substantially reduces the computational requirement. This has been found to be of key importance for the production of adaptive controllers which require "within the event cycle" controller modification times (the time taken for an adaptive controller to determine and implement the appropriate controller values) .

The design of H∞ controllers requires the determination of a number of weighting functions which are related to required robustness and performance criteria. Methods for the determination of the robustness weighting function are well known [e.g. Francis 1987]. A number of methods have been suggested for the determination of the performance weighting function such as those of Francis [1990], Lundstöm *et al.*

[1991] and Piché *et al.* [1991]. However, as Lundstöm *et al.* [1991] have noted, although simple guidelines exist for the determination of the performance weighting functions the specific design process is still both subjective and implicit in nature. The prospect of varying the performance weighting function to achieve specific closed-loop performance is not addressed explicitly in the literature.

Tsai *et al.* [1992] and Sefton and Glover [1990] completed a general analysis of the relationship between the plant, weighting functions and controller poles and zeros but the exact effect of adjustment of the frequency domain weights on the final closed-loop performance was not reconciled.

The H∞ controller design technique has been previously applied to a number of design, e.g. large space structures [McFarlene and Glover 1992]. The application areas tend, in general, to be slowly varying linear systems, which use highly complex nominal models to produce highly complex controllers. Liu and Liu [1990] described the use of H∞ controller synthesis for ac (dc) servo systems and completed a simple comparison of the responses of a benchmark plant controlled by a H∞ controller and by a PID controller. The research reported by Liu and Liu [1990] was essentially a precursor to the work discussed in this thesis. In the work of Liu and Liu [1990] H∞ controller synthesis was not applied to a high speed independent drive design example. Moreover only a simple PID controller was used (i.e. no VFF) and no guidelines on the determination of weighting functions (and hence controller) were given. The major contribution of this paper, however, is that the authors presented a persuasive argument for the proposition that motion driver servo systems may be modelled as second order systems. In general the simpler the system model, the simpler the resulting final controller [Chiang and Safanov 1992]). If the controller is too complex (i.e. more complex than the benchmark PID with VFF controller) an increase may occur in the controller update time leading to a degradation in dynamic performance due to sampling effects (i.e. a lack of information about the system between update times) [Morari and

Zafiriou 1989], together with the introduction of an additional time delay in the system, hence reducing the achievable system bandwidth [Seaward 1989] (due to the introduction of an additional phase shift into the system). The design of H∞ controllers for sampled data systems [Bamieh and Pearson 1992] (a dc servo system is an example of a sampled data system) has been shown to be possible.

The H∞ controller is to be implemented digitally since digital controllers are reliable, flexible, can implement complex control algorithms and facilitate the on-line adjustment of control parameters. Gu *et al.* [1989] and Postlethwaite *et al.* [1989] developed discrete-time H∞ controller determination algorithms, but these algorithms have not yet, to the authors knowledge, been developed into reliable software which could be used as a benchmark for any analysis or development of the algorithm. Igelias and Glover [1990] noted that, although the discrete-time state-space approach was analogous to continuous time approach, the solution of the discrete-time state-space H∞ controller synthesis problem is much more difficult. Since the transformation between the discrete and continuous domains can be readily carried out [Hewlett-Packard 1989] the standard state-space algorithm was used in this thesis.

1.3.3 Adaptive Control

The possibility of extending the performance of a robust controller by the use of the controller determination algorithm as a controller modifier in a STR (i.e. producing a robust adaptive controller) was suggested by Lewis *et al.* [1993] and Bitmead *et al.* [1990]. This approach was used by Grimble [1987a and 1987b] who developed a STR based on a H∞ controller design method. The design method embedded the H∞ controller synthesis into a fictitious LQG problem to allow easier solution of the design problem. The H∞ controller synthesis method was combined with a recursive least squares algorithm to produce a self-tuning regulator. Fairbairn and Grimble [1990] demonstrated the use of this STR on a ship steering simulation with a controller modification time of 4 seconds (it should be noted that the controller modification time was estimated and is 500 times that achieved for the system described in this thesis). The work of Grimble [1987a and 1987b] and Fairbairn and Grimble [1990] demonstrated that the construction of a STR based on H∞ controller synthesis was possible but the calculated computational requirements were so excessive as to provide no real value for the present application where, for the adaptation to occur within the event cycle, the controller modification would have to be achieved in a number of milliseconds.

Bitmead *et al.* [1990] discussed and analysed the interrelationship between the identifier and the controller in a self-tuning regulator. Bitmead *et al.* [1990] noted that the controller modification algorithm could induce instability by altering the frequency of the input signal to the identifier causing a drift in the identified parameters into regions of instability. The suggested solution to this problem was the use of an identifier filter to ensure that the nominal model of the system is bandwidth limited. Other papers concerned with adaptive control (using either dc servo systems or H∞ controller synthesis) include Tal and Baron [1987] who showed that an intelligent motion controller with self-tuning feedforward parameters could maintain system performance against significant variation. Hyde and Glover [1990] demonstrated that a controller switching strategy based on H∞ techniques was viable in a simulated aircraft model. Hashim and Grimble [1990] described an algorithm for an implicit H∞ self-tuning control scheme, obtained using a recursive least squares algorithm, although no examples of its use were given.

1.4 THESIS CONTRIBUTIONS.

This thesis is considered to demonstrate the first application of H∞ controller synthesis to industrial examples of single axis high speed independent drive systems. The author considers the following to be specific contributions to the utilisation of H∞ controller synthesis, and to the general field, of high-speed independent drive systems control:-

1) The application of H∞-norm optimisation to real-world industrial design examples is developed and demonstrated. In one design example a H∞ controller was designed that produced a closed-loop system performance which met a performance criteria which had not been achieved using a benchmark PID with VFF controller.

2) Determination of a direct-form of the state-space solution to the H∞ problem for a limited class of weighting and plant transfer functions. The transform has been used in two distinct areas of research :-
    - i) Analysis of the relationship between the theoretical plant, weighting functions and actual system response, yielding simple guide-lines for weighting function determination and tuning.
    - ii) The direct-form of the controller has been used to develop a self-tuning regulator, with an 8 ms controller modification time, which has been implemented on a practical system.

3) A set-point gain scheduling controller has been developed and implemented on the Drives Test Facility at Aston University.

4) Guidelines have been produced for the application of H∞ controllers, set-point gain scheduling controllers and a self-tuning regulator to single axis of high-speed independent drive systems.

5) An analysis of a benchmark PID with VFF controller and a H∞ controller has been completed.

6) An alternative approach to the traditional γ iteration method, used to determine optimal $H\infty$ controllers, was developed using a weighting function parameter scanning program.

7) A relative magnitude and absolute phase shift error value, determined in the frequency domain, were developed to enable the comparison of identification errors for different combinations of forgetting factor and forgetting factor weights in the least squares parameter estimation with variable forgetting factor technique.

8) The STR produced demonstrated the use of identified parameter limits, used instead of an identification signal filter, to overcome the problem of controller / identifier interaction causing system instability.

Additionally six research papers have been completed, of which four have been accepted for publication or published, and three technical (based on the work completed in this thesis) have been supplied to our industrial sponsor (with all appropriate software).

CHAPTER 2.

H∞ CONTROLLER SYNTHESIS AND PARAMETER ESTIMATION.

2.1 BASIC CONCEPTS.

The underlying concept of H∞ controller synthesis is that the H∞ norm of a combination of transfer functions can be used to determine a controller which addresses a performance criterion and is stable against a set of perturbations and disturbances.

2.1.1 Mathematical Background

Zames [1981] suggested using H∞-norms to produce controllers which satisfy both performance and stability criteria. The controller synthesis is completed over the Hardy space H∞ which consists of all complex-valued functions F(s) of a complex variable which are analytic and bounded in the open right half-plane, Re s > 0. "Bounded" means there exists a real number, b, such that

$$| F(s) | \leq b , \quad \mathrm{Re}\, s > 0. \tag{2.1}$$

The least such bound b is the H∞-norm of F, denoted $\| F \|_\infty$. The standard definition of the H∞-norm of a transfer function (or matrix of transfer functions), F, is given in terms of the supremum (sup), the least upper bound, of the maximum singular values, $\bar{\sigma}$, of a function over the entire frequency range, denoted by

$$|| F ||_\infty = \sup_{\omega} \bar{\sigma}(F(j\omega)) \tag{2.2}$$

In H∞ controller synthesis the input (including disturbances) and output signals are indeterminate (which is the case in the majority of practical systems) and belong to a class of signals with L_2 norm less than or equal to 1 (The properties of norms hold in that the signal can be scaled by a factor, up to infinity, but the signal is usually referenced back to unity [Maciejowski 1993]). The L_2 norm of a signal, u(t), is given by

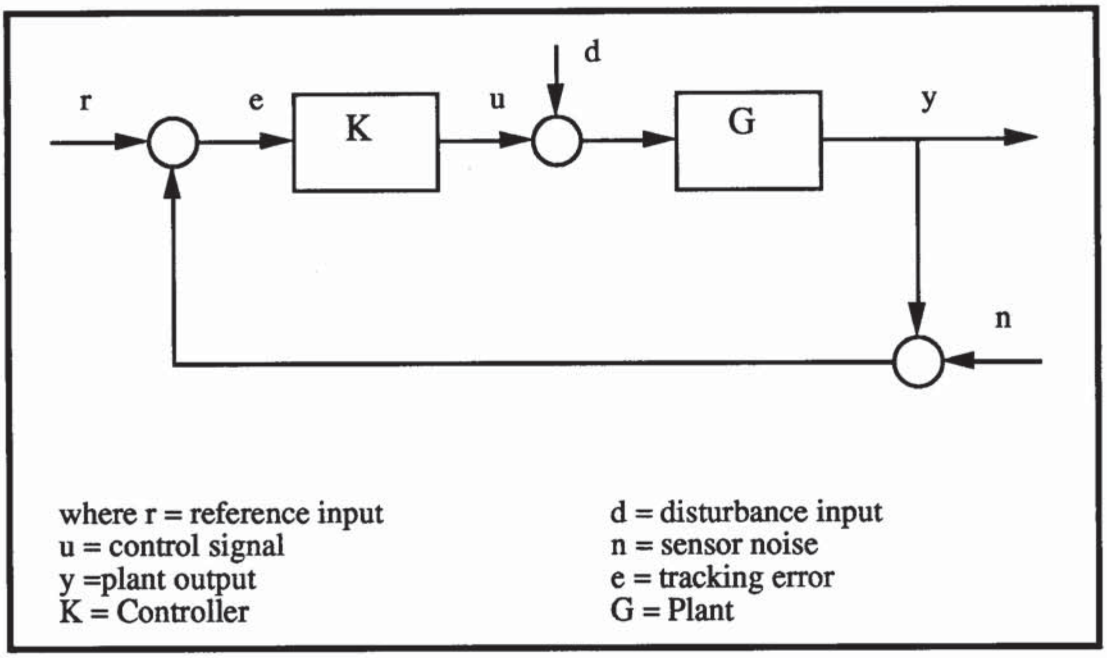
$$\| u(t) \|_2 := \left(\int_{-\infty}^{\infty} | u(t) |^2 dt \right)^{1/2} \qquad (2.3)$$

Minimising the H∞-norm of a transfer function F is equivalent to minimising the L_2 norm of the output, y, for any input, u,

$$\| F \|_\infty = \sup\{ \| y \|_2 : \| u \|_2 \leq 1 \} \qquad (2.4)$$

(i.e. the minimal output energy for a unit energy input).

This property can be applied to control problems such as disturbance attenuation or minimisation of tracking errors. Two transfer function are commonly used in the controller synthesis; the sensitivity function, S, used for system performance and the complementary sensitivity function, T, used for system stability.



where r = reference input
u = control signal
y =plant output
K = Controller

d = disturbance input
n = sensor noise
e = tracking error
G = Plant

Figure 2.1 Basic Feedback Loop

2.1.2 Stability Measure.

Referring to Figure 2.1 the complementary sensitivity function, T, is given by

$$T = GK / (1 + GK) \tag{2.5}$$

Then using the Generalised Nyquist stability theorem [Vidyasagar 1985] the H∞ norm of the complementary sensitivity function can be used to determine the level of internal stability of a system (since the H∞-norm of T is simply the maximum closed-loop system gain). Internal stability for Figure 2.1 means that the transfer functions from the exogenous inputs r, d, n to the internal signals e, u are stable.

2.1.3 Performance Measure.

Minimising the H∞ norm of the sensitivity function, S, given by

$$S = 1 / (1 + GK) \tag{2.6}$$

will minimise the tracking error, e. The distance from the critical Nyquist point ,-1, to the Nyquist plot of GK equals 1/||S||∞ consequently minimising the H∞-norm of S increases the system stability

$$\begin{aligned}\text{distance from -1 to Nyquist plot} &= \inf_{\omega} |-1 - L(j\omega)| \\ &= \inf_{\omega} |1 + L(j\omega)| \\ &= \left[\sup_{\omega} \frac{1}{|1 + L(j\omega)|} \right]^{-1} \qquad (2.7) \\ &= ||S||_{\infty}^{-1}\end{aligned}$$

The complementary sensitivity function, T, is used as a stability margin measure in preference to S, since S does not include frequency domain information (as can be seen in the above equations 2.7): better stability margins are obtained by taking explicit frequency-dependent perturbation models (i.e. the multiplicative perturbation model) which are related to the complementary sensitivity function [Francis 1990], see Section 2.2.

2.2 STABILITY SPECIFICATION.

A perfect plant model, G, is assumed to exist in (2.5) and (2.6). In reality the nominal and actual plant dynamics will differ since:-

1) The nominal model must be simple enough to be mathematically manipulated and therefore usually includes only dominant system modes (A trade-off occurs between the simplicity and accuracy of the model).
2) A number of errors may occur in the identification process (i.e. due to numerical accuracy or linearisation of non-linear dynamics).
3) The plant dynamics may vary after the nominal model has been determined.

The design procedure is required to produce a controller which copes with these variations as well any errors that occur in the controller determination or implementation. The disparities between the real and nominal system are referred to as system uncertainty (or modelling errors).

Two different types of plant uncertainty exist: structured and unstructured. Structured uncertainty consists of a finite number of plant parameter variations or a discrete set of plants. Unstructured uncertainty consists of an infinite number of plants in a region around the nominal model. Structured uncertainty can be described by a diagonal matrix of perturbations [Doyle 1982], used in μ analysis. Structured uncertainty places less restriction on system performance since the system is not constrained by the highest level of uncertainty of any system component as in $H\infty$ controller synthesis [Maciejowski 1993]. This thesis deals with single-input single output (SISO) systems where only general frequency domain information is available and where the additional complexity of controllers determined by μ analysis [Chiang and Safonov 1992] means that the use of structured uncertainty is not viable. The added complexity of a controller determined by μ analysis (approximately three times the complexity of a $H\infty$ controller [Chiang and Safonov 1992]) would lead to an increase in the controller update time and hence a decrease in system performance due to both sampling effects and an increase in the system time delays.

System uncertainty is modelled by a perturbation. Three forms of perturbations are commonly discussed: additive [Chen and Deseor 1982], multiplicative [Doyle and Stein 1981] and coprime factorisation [Vidysagar 1985]. The form of each is given in equation form below and diagramatically overleaf:-

The form of an additive perturbation is $G(s) = G_0(s) + \Delta_A(s)$

The form of a multiplicative perturbation is $G(s) = [I + \Delta_0(s)]G_0(s)$

The form of a coprime perturbation is $G(s) = (M + \Delta_m)/(N + \Delta_n)$

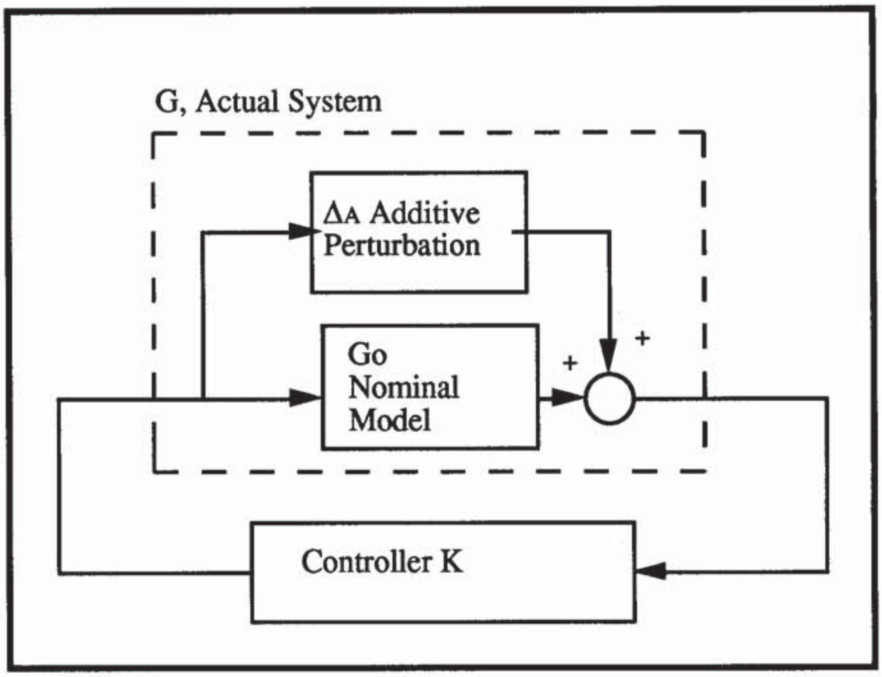


Figure 2.2 Additive Perturbation Model

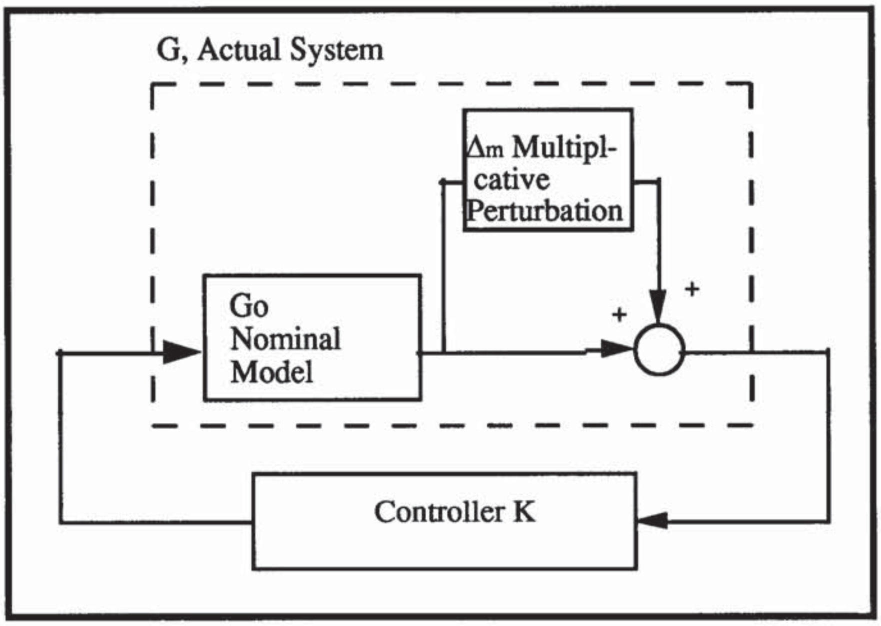


Figure 2.3 Multiplicative Perturbation Model

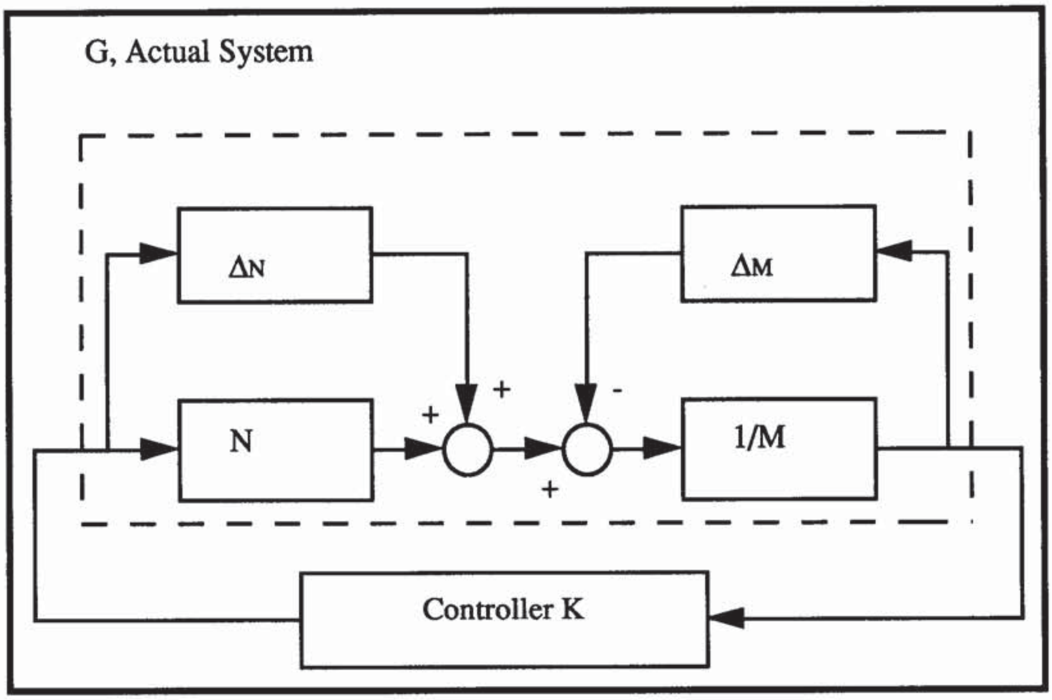


Figure 2.4 Coprime Factorisation Perturbation Model

Vidyasagar [1985] showed that the coprime factor approach for scalar systems precludes the possibility of common right half plane (RHP) zeros for stable factors (i.e. the coprime factors of a transfer function will contain no unstable hidden plant modes). The coprime factorisation approach does not require the perturbations to be constrained to preserve the number of RHP plant poles. This allows a wider plant perturbation class to be considered and a greater degree of confidence to be placed in the robust stability condition. The perturbations Δ_m and Δ_n are both asymptotically stable functions. The greater complexity of the coprime factorisation approach, although it has theoretical advantages, limits its use for high speed application. This thesis therefore uses the standard approach of combining multiplicative and additive perturbations into a fictitious multiplicative perturbation .

The multiplicative perturbation, denoted Δ_o, must be allowable (a variable stable transfer function with $H\infty$ norm strictly less than 1). The perturbation is combined with a frequency-dependent weight $W_2(j\omega)$ which provides an uncertainty profile, usually increasing in magnitude with increasing frequency, to more effectively represent the system uncertainty.

The robustness weighting function is determined using (for unity magnitude multiplicative perturbation)

$$\left| \frac{G(j\omega)}{G_0(j\omega)} - 1 \right| \leq | W_2(j\omega) | \qquad (2.8)$$

This inequality describes a disk in the complex plane, as shown in Figure 2.5.

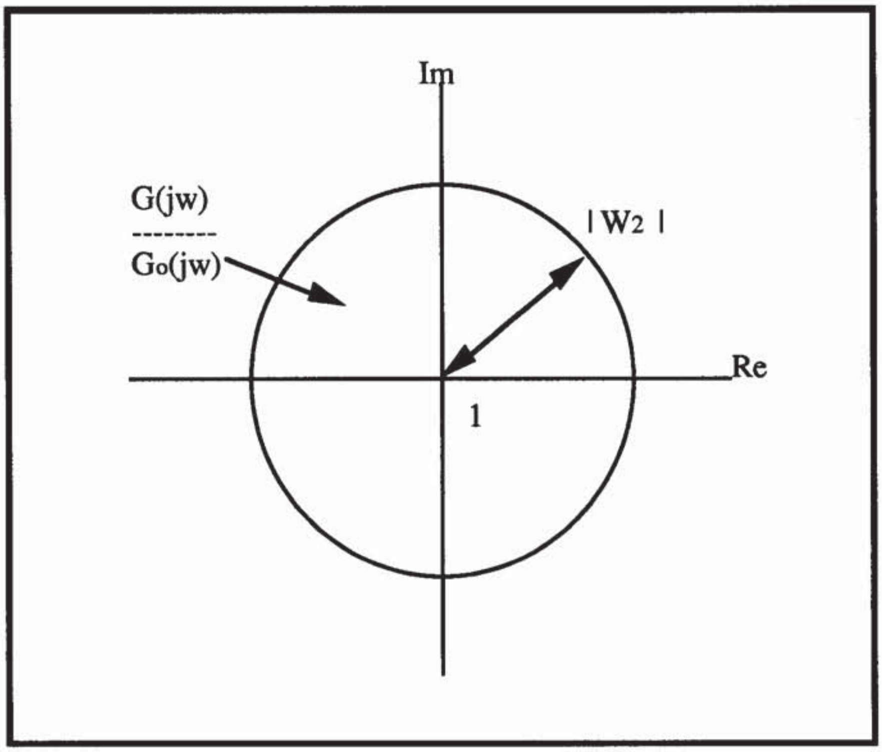


Figure 2.5 Uncertainty Disk in Complex Space

At each frequency the point I $G(j\omega)/G_o(j\omega)$ I lies in the disk, centre 1, radius I W_2 I. Typically, I $W_2(j\omega)$ I is an increasing function of ω: uncertainty increases with increasing frequency. Note that the weighting function includes expected variations in the plant, uncertainty caused by model simplifications and modelling errors (i.e. G is estimated to be in a general region using experimental data or previous "knowledge").

For a multiplicative uncertainty, stability is guaranteed if

$$\| W_2 \dot{T_0} \|_\infty \le \| \dot{\Delta_0} \|_\infty^{-1} \tag{2.9a}$$

This leads to a conceptually useful graphical representation. If the perturbation Δ_0 is of unity magnitude the following relationship, with $L_0 = G_0K$, holds

$$\begin{aligned} \|W_2T_o\|_\infty < 1 &\Leftrightarrow \left|\frac{W_2(j\omega)L_0(j\omega)}{1 + L_0(j\omega)}\right| < 1 \;\; \forall\omega \\ &\Leftrightarrow |W_2(j\omega)L_0(j\omega)| < |1 + L_0(j\omega)| \;\; \forall\omega \end{aligned} \tag{2.9b}$$

The last inequality describes a system for which at every frequency the critical (Nyquist) point, -1, lies outside the disk of centre $L_0(j\omega)$, radius $|W_2(j\omega)L_0(j\omega)|$, this is shown in Figure 2.6. The actual open-loop plant gain, GK, exists in the region bounded by $|W_2(j\omega)L_0(j\omega)|$ (by the definition of the robustness weighting function). Figure 2.6 is a fuzzy Nyquist plot [Safonov 1980]. (A proof of stability for multiplicative uncertainty is given in Appendix A).

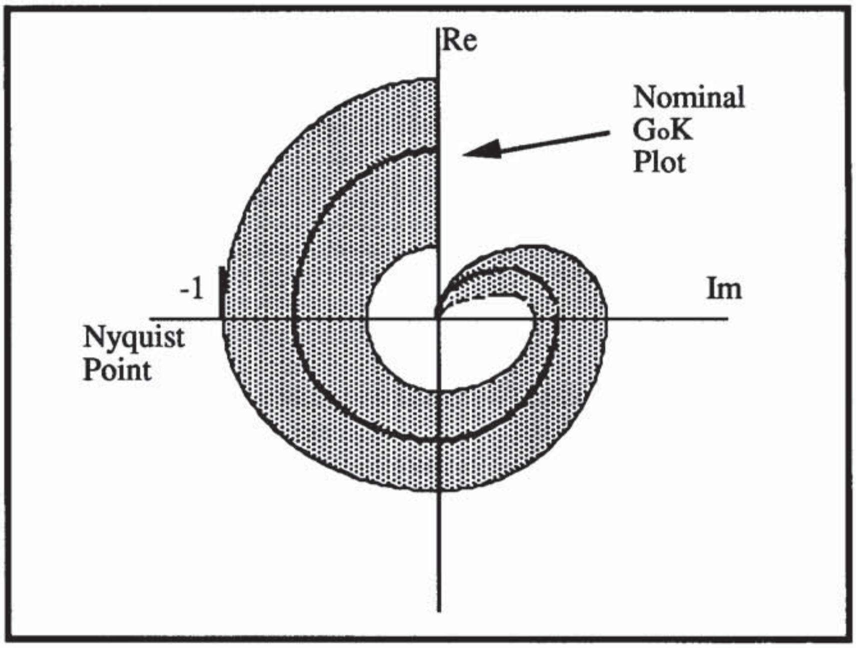


Figure 2.6 Fuzzy Nyquist Plot.

The weighting function W_2 limits the nominal complementary sensitivity function

$$\overline{\sigma}\,(T_o(j\omega)) \leq |\, W_2^{-1}(j\omega)\,| \tag{2.10}$$

This relationship can be used to determine a guaranteed nominal gain and phase margin (considering the perturbation to be of magnitude 1).

2.2.1 Relationship Between H∞-norm and Classical Phase and Gain Margin.

The H∞ norm of the complementary sensitivity function can be directly related to the traditional gain and phase margins by

$$\begin{aligned}&\text{Guaranteed Gain Margin} = 1 \pm \frac{1}{\|T\|_\infty}\\ &\text{Guaranteed Phase Margin} = \pm 2\sin^{-1}\frac{1}{2\|T\|_\infty}\end{aligned} \tag{2.11}$$

2.2.2. Proof of Guaranteed Phase and Gain Margins

This proof is included here, even though it is often referred to as a standard formula, since the proof was not found in the literature reviewed. From the theory of M-circles [Maciejowski, 1989] $\|T\|_\infty$ describes the least radius M-circle with centre

$$x = \frac{-\|T\|_\infty^2}{\|T\|_\infty^2 - 1} \quad ; \quad y = 0 \quad \text{and radius} \quad r = \left| \frac{\|T\|_\infty}{\|T\|_\infty^2 - 1} \right| \tag{2.12}$$

from Figure 2.3 it can be seen that the gain margin, km, is given

$$\frac{-1}{km} = \frac{-\|T\|_\infty^2 \pm \|T\|_\infty}{\|T\|_\infty^2 - 1} \Rightarrow km = \frac{\|T\|_\infty^2 - 1}{\|T\|_\infty^2 \pm \|T\|_\infty} = \frac{\|T\|_\infty \pm 1}{\|T\|_\infty} = 1 \pm \frac{1}{\|T\|_\infty} \tag{2.13}$$

the guaranteed phase margin can be determined by simply geometry. Using the cosine rule

$$\|T\|_\infty^2 = (\|T\|_\infty^2)^2 + (\|T\|_\infty^2 -1)^2 - 2(- \|T\|_\infty^2)(\|T\|_\infty^2 -1)\cos\theta \tag{2.14}$$

$$\cos\theta = \frac{2\,\|T\|_\infty^4 + 3\,\|T\|_\infty^2 - 1}{2\,\|T\|_\infty^4 - 2\,\|T\|_\infty^2} = \pm\left(1 - \frac{2}{4\,\|T\|_\infty^2}\right) \tag{2.15}$$

$$1 - 2\sin^2\frac{\theta}{2} = \cos\theta = \pm\left(1 - \frac{2}{4\,\|T\|_\infty^2}\right) \tag{2.16}$$

$$\sin\frac{\theta}{2} = \pm\,\frac{1}{2\|T\|_\infty} \quad \text{therefore} \quad \theta = \pm\, 2\,\sin^{-1}\left(\frac{1}{2\,\|T\|_\infty}\right) \tag{2.17}$$

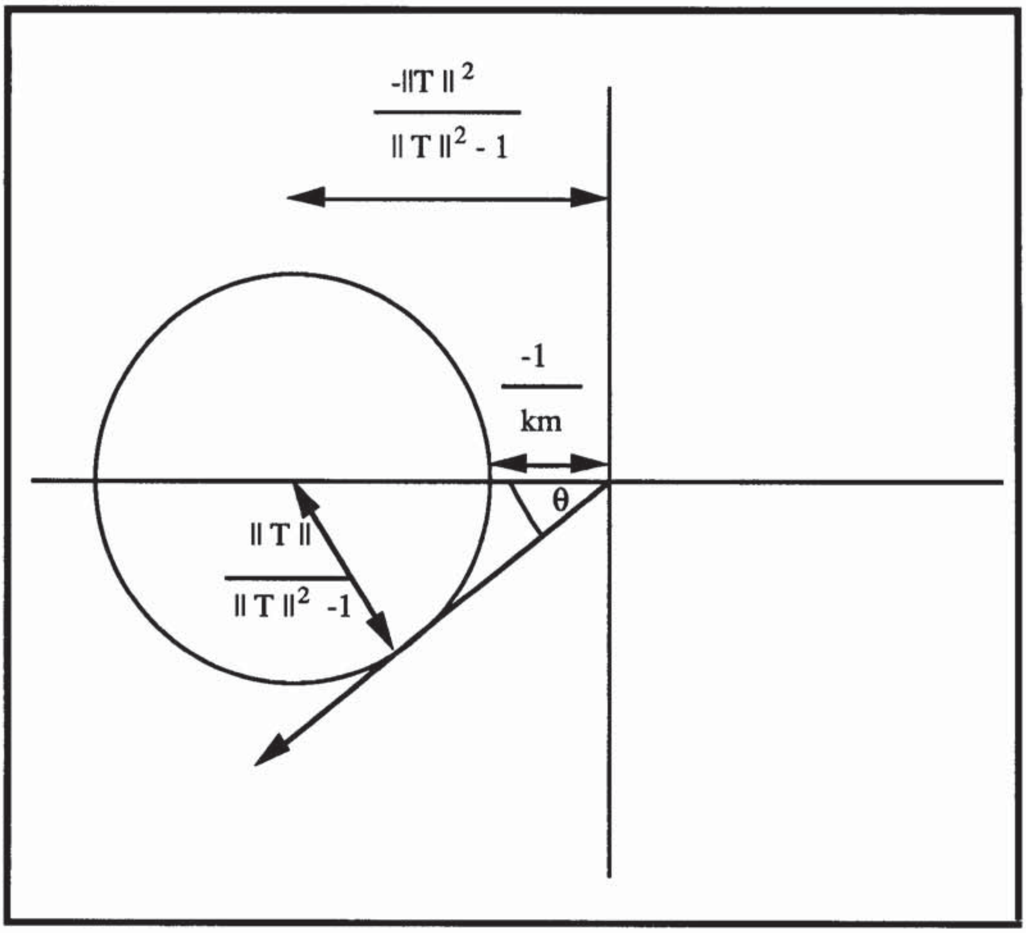


Figure 2.7 M-Circles Diagram.

2.3 ROBUST PERFORMANCE.

A prerequisite of a robust controller is that it maintains system stability, achieved by satisfying (2.7). Achieving system stability alone will not usually meet all the performance requirements placed on a system (i.e. usually a level of performance is required) therefore the stability criterion is combined with a performance criterion. The performance criterion is used to determine a frequency weighting function, W_1, which shapes the sensitivity function response. That is W_1 limits the magnitude of the tracking error over a particular part of the frequency range. The performance requirement is

$$\| W_1 S \|_\infty \leq 1 \qquad (2.18)$$

Robust performance means that internal stability and performance, of a specified type, should hold for any plant G in the region described by

$$G(s) = [I + \Delta_0(s)]G_0(s) \qquad (2.19)$$

(for multiplicative perturbations).

Recall that the nominal feedback system is internally stable, the nominal performance condition is $\|W_1 S\|_\infty \leq 1$ and the robust stability condition is that $\|W_2 T\|_\infty \leq 1$. If G is perturbed to $(1 + \Delta W_2)G$, S is perturbed to

$$\frac{1}{1 + (1 + \Delta W_2)L} = \frac{S}{1 + \Delta W_2 T} \qquad (2.20)$$

Clearly the robust performance condition should therefore be

$$\| W_2 S \|_\infty \leq 1 \quad \text{and} \quad \left\| \frac{W_1 S}{1 + \Delta W_2 T} \right\|_\infty \leq 1, \quad \forall \Delta \qquad (2.21)$$

A sufficient and necessary condition for robust performance is

$$|| \, |W_1S| + |W_2T| \, ||_\infty < 1 \qquad (2.22)$$

The equation means that the sum of the weighted sensitivity functions is less than unity over the entire frequency range, which can only be possible if the controller (if it exists) has been shaped to meet the stability and performance requirements placed on the nominal model. This, (2.22), is represented in the augmented plant shown in Figure 2.8 which combines the performance and stability requirements into one unified problem, with P(s) given by

$$P(s) = \left[\begin{array}{c|c} W_1 & -W_1G \\ 0 & W_2G \\ \hline I & -G \end{array}\right] \qquad (2.23)$$

The problem is to determine a controller such that $\| T_{y1u1} \|_\infty \leq 1$. It is possible to extend the problem to more than 2 sensitivity functions [Chiang and Safonov 1992].

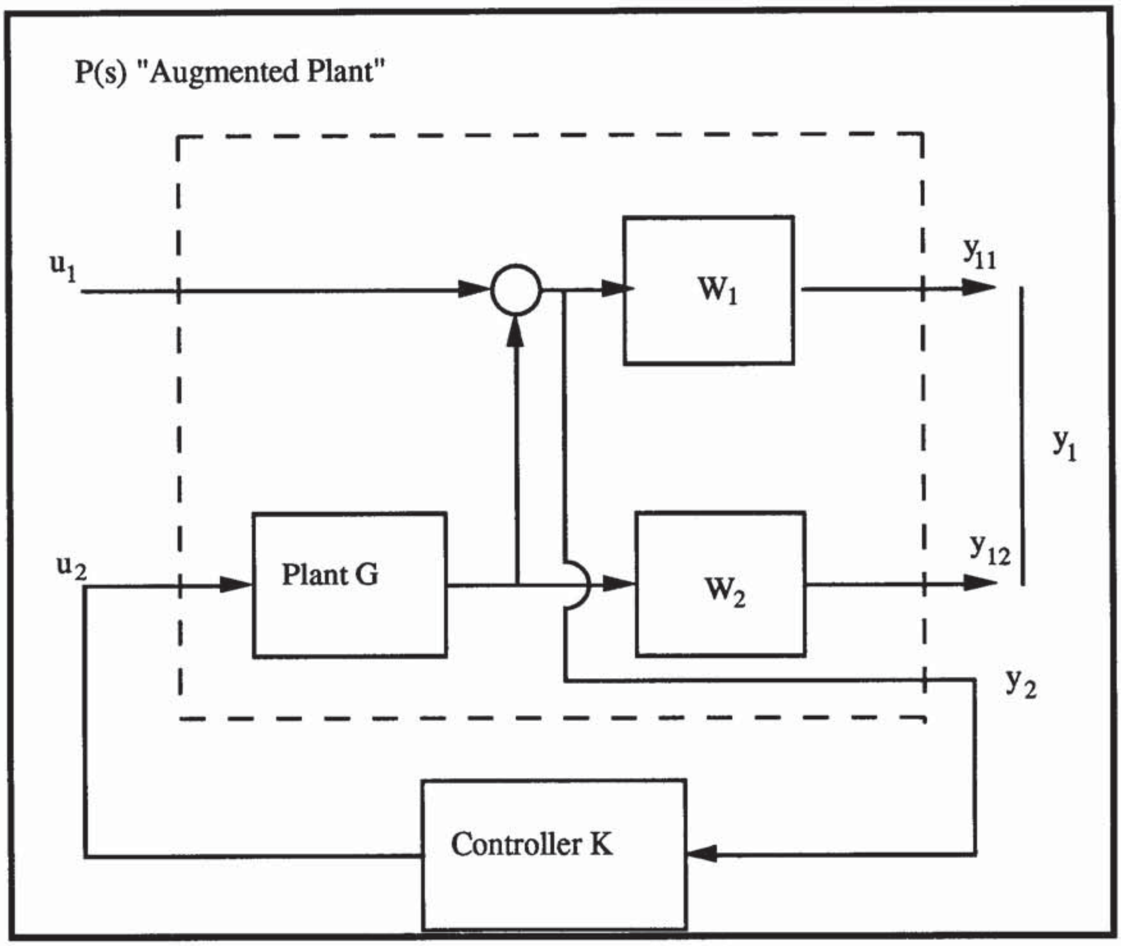


Figure 2.8 Standardised Mixed H∞ Design Problem.

2.4. DOYLE AND GLOVERS STATE SPACE METHOD.

Doyle and Glover suggested a state-space method of determining a solution to the H∞ design process (i.e. determining a H∞ controller). The weighting functions and nominal plant model are combined into a state space representation of the augmented plant P(s) and two Hamiltonian matrices, L_∞ and Z_∞, are determined (from P(s)).

Doyle *et al.* [1989] proved that if the eigenvalues of L_∞ and Z_∞ are not on the imaginary axis then $\| T_{y1u1} \|_\infty \leq \gamma$ (γ is usually unity). Therefore a simple test to find an optimal value of γ (or of the weighting function parameters) is to examine the eigenvalues of L_∞ and Z_∞ and adjust appropriately (i.e. adjust γ is until a value is found such that $\gamma + \delta\gamma$ gives no imaginary eigenvalues while $\gamma - \delta\gamma$ gives imaginary eigenvalues, for an appropriately small value of δ, a similar technique can be used for weighting function parameters). Once an optimal γ value has been determined an optimal controller can be produced via the solution of two Algebraic Riccati Equations (ARE) formed from L_∞ and Z_∞.

If the state-space representation of the augmented plant is

$$P(s) = \left[\begin{array}{c|cc} A_1 & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \tag{2.24}$$

and the following six assumptions are satisfied

$$D_{22}=0\,;\; D_{12}=\begin{bmatrix} 0 \\ I \end{bmatrix};\; D_{21}=[\;0 \quad I\;]\,;\; D_{11}=0\,;\; D_{12}^T=0\,;\; B_1 D_{21}^T=0 \tag{2.25}$$

then the two required Hamiltonian matrices are given by

$$L_\infty := \begin{bmatrix} A & \gamma^2 B_1 B_1^T - B_2 B_2^T \\ -C_1^T C_1 & -A^T \end{bmatrix} \tag{2.26}$$

$$Z_\infty := \begin{bmatrix} A^T & \gamma^2 C_1^T C_1 - C_2^T C_2 \\ -B_1 B_1^T & -A \end{bmatrix} \tag{2.27}$$

There then exists an admissible controller such that $\| T_{y1u1} \|_\infty < \gamma$ iff the following three conditions hold

(i) $L_\infty \in \mathrm{dom(Ric)}$ and $X_\infty := \mathrm{Ric}(L_\infty) \geq 0$

(ii) $Z_\infty \in \mathrm{dom(Ric)}$ and $Y_\infty := \mathrm{Ric}(Z_\infty) \geq 0$

(iii) $\rho(X_\infty Y_\infty) < \gamma^2$ (where ρ denotes the spectral radius of $X_\infty Y_\infty$ i.e. the largest eigenvalue of $X_\infty Y_\infty$)

Conditions (i) and (ii) mean that L_∞ and Z_∞ are members of the domain of Ric (Hamiltonian matrices with Riccati Equation solutions) which have two properties:

a) they have no eigenvalues on the imaginary axis

b) that the Hamiltonian matrices span two complementary subspaces.

Conditions (i) and (ii) also require that positive semi-definite solutions, X_∞ and Y_∞, of two Riccati Equation exist (positive semi-definite means that all the eigenvalues of the solution are zero or positive [Fraleigh and Beauregard 1987]). Condition (iii) means that the largest eigenvalue of the product of the two Riccati equation solutions is less than γ.

When these conditions hold, one such controller is

$$K_{sub}(s) = \left[\begin{array}{cc} A_\infty & -M_\infty L_\infty \\ F_\infty & 0 \end{array} \right] \tag{2.28}$$

where $A_\infty = A + \gamma^2 B_1 B_1^T X_\infty + B_2 F_\infty + M_\infty L_\infty C_2$

$$F_\infty = -B_2^T X_\infty, \qquad L_\infty := -Y_\infty C_2^T, \qquad M_\infty := (I - \gamma^2 Y_\infty X_\infty)^{-1}$$

If the assumption that $D_{11} = 0$ is not satisfied then the maximum singular value of a partitioned form of D_{11} must be strictly less than γ [Glover and Doyle 1988]. This is necessary since the maximum singular value of D_{11} is the value of the H_∞-norm of the augmented state-space representation of the plant at infinite frequency.

Problems exist with this technique. Although an optimal γ may exist the Riccati equations may be insoluble (due to numerical problems) and therefore no controller is produced [Safonov *et al.* 1989]. The method has the advantage that the solution is readily obtainable and that the performance can be developed as required.

Other methods are available for solving H∞ controller synthesis problems, such as the polynomial or J-Spectral factorisation approaches [Stoorvogel 1992]. The state-space method was used in this thesis for two reasons, firstly it is the simplest technique [Lundström *et al.* 1991] and secondly it is the most thoroughly developed.

2.5 SIMPLIFYING H∞ CONTROLLER SYNTHESIS.

The simplified controller determination outlined in Doyle *et al.* [1989] showed that complete derivations could be achieved for the controller if six assumptions (2.25) hold. These assumptions can be forced upon the general form by using a set of equivalent relationships.

The first constraint $D_{22} = 0$ can be lifted by using a change of variable.

$$D_{22}u_2 + y_2 \rightarrow y_2 \tag{2.29}$$

which has the effect of zeroing out the D_{22} element of the plant and wrapping a feedback element of $-D_{22}$ around the controller.

The constraints $D_{12} = [\ 0\ I\ ]^T$ and $D_{21} = [0\ I]$ can be lifted by completing a singular-value decomposition of D_{12} and D_{21} as

$$\begin{aligned} D_{12} &= U_{D_{12}} \begin{bmatrix} 0 \\ \Sigma_{D_{12}} \end{bmatrix} V_{D_{12}}^T \\ D_{21} &= U_{D_{12}} \begin{bmatrix} 0 & \Sigma_{D_{12}}^T \end{bmatrix} V_{D_{21}}^T \end{aligned} \tag{2.30}$$

where U_{D12}, U_{D21}, V_{D12} and V_{D21} are unitary matrices. The P(s) matrix is scaled to

$$P(s) = \begin{bmatrix} A & B_1 V_{D21} & B_2 V_{D12} \Sigma_{D12}^{-1} \\ U_{D21}^T C_1 & U_{D21}^T D_{11} V_{D21} & \begin{bmatrix} 0 \\ I \end{bmatrix} \\ \Sigma_{D21}^{-1} U_{21}^T C_2 & [\,0 \;\; I\,] & 0 \end{bmatrix} = \begin{bmatrix} A^1 & B_1^1 & B_2^1 \\ C_1^1 & D_{11}^1 & D_{12}^1 \\ C_2^1 & D_{21}^1 & 0 \end{bmatrix} \quad (2.31)$$

(This change of variable has the effect of scaling u_1, u_2, y_1 and y_2).

The constraint $D_{11} = 0$ may be lifted by two sets of variable changes. D_{11} is partitioned as

$$D_{11} = \begin{bmatrix} D_{1111} & D_{1112} \\ D_{1121} & D_{1122} \end{bmatrix} \quad (2.32)$$

with

$$\mathrm{size}(D_{1122}) = \dim(u_2) \times \dim(y_2) \quad (2.33)$$

The D_{11} term is now fixed using a variable, $K\infty$. Where $K\infty$ is a constant matrix chosen to minimise the greatest singular value of the resulting D_{11} matrix i.e.

$$D_{11} = \begin{bmatrix} D_{1111} & D_{1112} \\ D_{1121} & D_{1122} + K\infty \end{bmatrix} \quad (2.34)$$

The requirement for γ to be greater than the maximum singular value of D_{11} still holds even though the D_{11} term is to be zeroed out.

From Parrott's theorem [Parrott 1978] a minimising $K\infty$ is

$$K\infty = -[\, D_{1122} + D_{1121} (I - D_{1111}^T D_{1111})^{-1} D_{1111}^T D_{1112}] \quad (2.35)$$

The elements of P(s) are first scaled using K∞

$$P(s) = \begin{bmatrix} A^1 + B_2^1 K\infty C_2^1 & B_1^1 + B_2^1 K\infty D_{21}^1 & B_2^1 \\ C_1^1 + D_{12}^1 K\infty C_2^1 & D_{11}^1 + D_{12}^1 K\infty D_{21}^1 & D_{12}^1 \\ C_2^1 & D_{21}^1 & 0 \end{bmatrix} = \begin{bmatrix} A^2 & B_1^2 & B_2^1 \\ C_1^2 & D_{11}^2 & D_{12}^2 \\ C_2^2 & D_{21}^2 & 0 \end{bmatrix} \tag{2.36}$$

Next the D_{11} element of P(s) is zeroed using a series of Strict System Equivalence (SSE) transformations.

Let $X = D_{11}^2$

$$P(s) = \begin{bmatrix} A^2 + B_1^2 X^T(I - XX^T)^{-1} C_1^2 & B_1^2(I - X^T X)^{-1/2} & B_2^2 + B_1^2 X^T(I - XX^T)^{-1} D_{12}^2 \\ (I - XX^T)^{-1/2} C_1^2 & 0 & (I - X^T X)^{-1/2} D_{12}^2 - 1 \\ C_2^2 + D_{21}^2 X^T(I - XX^T)^{-1} C_1^2 & D_{21}^2(I - X^T X)^{-1/2} & D_{21}^2 X^T(I - XX^T)^{-1} D_{12}^2 \end{bmatrix}$$

$$= \begin{bmatrix} A^3 & B_1^3 & B_2^3 \\ C_1^3 & D_{11}^2 & D_{12}^3 \\ C_2^3 & D_{21}^3 & D_{22}^3 \end{bmatrix} \tag{2.37}$$

The D_{22} term is re-zeroed and a second singular-value decomposition is required to satisfy the D_{12} and D_{21} criteria

$$D_{12}^3 = \widetilde{U}_{D12} \begin{bmatrix} 0 \\ \widetilde{\Sigma}_{D12} \end{bmatrix} \widetilde{V}_{D12}^T \tag{2.38}$$

$$D_{21}^3 = \widetilde{U}_{D21} \begin{bmatrix} 0 & \widetilde{\Sigma}_{D12} \end{bmatrix} \widetilde{V}_{D12}^T \tag{2.39}$$

$$P(s) = \begin{bmatrix} A^3 & B_1^3\tilde{V}_{D21} & B_2^3\tilde{V}_{D12}\tilde{\Sigma}_{D12}^{-1} \\ \tilde{U}_{D21}^T C_1^3 & 0 & \begin{bmatrix} 0 \\ I \end{bmatrix} \\ \tilde{\Sigma}_{D21}^{-1}\tilde{U}_{21}^T C_2^3 & [\,0 \;\; I\,] & 0 \end{bmatrix} = \begin{bmatrix} A^4 & B_1^4 & B_2^4 \\ C_1^4 & 0 & D_{12}^4 \\ C_2^4 & D_{21}^4 & 0 \end{bmatrix} \quad (2.40)$$

Safonov *et al.* [1989] showed that the final two restrictions on the augmented state-space plant representation are not required. If the requirements required by the Doyle and Glover state-space method have been satisfied and the three requirements on the Hamiltonian matrices and Riccati equation solutions have been satisfied the state-space representation of the controller is given by

$$K(s) = \begin{bmatrix} A_k & B_{k1} & B_{k2} \\ C_{k1} & 0 & I \\ C_{k2} & I & 0 \end{bmatrix} \qquad (2.41)$$

where $A_k = -Is + E^{-1}\hat{A}$; $B_{k1} = -E^{-1}G$; $B_{k2} = -E^{-1}(B_2^4 + Y\infty C_1^{4T}D_{12}^4)$ (2.42)

and $C_{k1} = F$; $C_{k2} = -(C_2^4 + D_{21}^4 B_1^{4T} X\infty)$ (2.43)

$$F = -(B_2^{4T}X\infty + D_{12}^{4T}C_1^4) \qquad (2.44)$$

$$G = -(Y\infty C_2^{4T} + B_1^4 D_{21}^{4T}) \qquad (2.45)$$

$$X\infty = \mathrm{Ric}\begin{bmatrix} A^4 - B_2^4 D_{12}^{4T} C_1^4 & B_1^4 B_1^{4T} - B_2^4 B_2^{4T} \\ -\tilde{C}_1^{4T}\tilde{C}_1^T & (A^4 - B_2^4 D_{12}^{4T} C_1^4)^T \end{bmatrix} \qquad (2.46)$$

$$Y\infty = \mathrm{Ric}\begin{bmatrix} (A^4 - B_1^4 D_{21}^{4T} C_2^4)^T & C_1^{4T}C_1^4 - C_2^{4T}C_2^4 \\ -\tilde{B}_1^4\tilde{B}_1^{4T} & A^4 - B_1^4 D_{21}^{4T} C_2^4 \end{bmatrix} \qquad (2.47)$$

$$\hat{A} = E(A^4 + B_2^4 F + B_1^4 B_1^{4T} X\infty) + G(C_2^4 + D_{21}^4 B_1^{4T} X\infty) \qquad (2.48)$$

$$E = I - Y_\infty X_\infty \tag{2.49}$$

$$\widetilde{C}_1 = (I - D^4_{12}D^{4T}_{12})C^4_1 \tag{2.50}$$

$$\widetilde{B}_1 = B^4_1(I - D^{4T}_{21}D^4_{21}) \tag{2.51}$$

Equivalent retransformations have to be completed on the controller elements, i.e. the reverse of the previous steps, to obtain the final controller. Chiang and Safonov [1992] used this approach in the Matlab Robust Control Toolbox script Hinf.m.

2.6 RICCATI EQUATIONS.

The previous section shows that a stabilising controller exists if the solutions to two Riccati equations are positive semi-definite and the spectral radius of the product of the two Riccati equations is less than γ^2. Therefore the theory and solution of Riccati equations is of key importance.

Consider the Algebraic Riccati Equation (ARE)

$$A^T X + XA + XRX - Q = 0 \tag{2.52}$$

with the associated Hamiltonian matrix

$$H = \begin{bmatrix} A & R \\ Q & -A^T \end{bmatrix} \tag{2.53}$$

A solution X of the ARE with all of the eigenvalues of A + RX in the open left hand-plane (LHP) is called the stabilising solution. Doyle *et al.* [1989] showed that a stabilising solution, X, exists if H has no imaginary eigenvalues and that the stabilising solution must be positive semi-definite (i.e. all eigenvalues of $X \geq 0$).

2.6.1 Proof of Equivalence of $\| P \|_\infty < \gamma$ and H has no eigenvalues on the imaginary axis. (A full proof appears in Doyle *et al.* [1989]).

Assume without loss of generality that $\gamma = 1$ and that the state-space of the augmented plant is

$$P = \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} \tag{2.54}$$

$$(I - P^T P)^{-1} = \begin{bmatrix} A & BB^T & B \\ -C^T C & -A^T & 0 \\ 0 & B^T & I \end{bmatrix} \tag{2.55}$$

so H is the A-matrix of $(I - P^T P)^{-1}$. H has no eigenvalues on the imaginary axis iff (I - $P^T P$) has no poles there. Therefore $(I - P^T P)$ has no zeros there, that is $| P(j\omega) | \neq 1$ (The modulus of P cannot be 1 since $(I - P^T P)^{-1}$ would not exist). Since P is strictly proper this implies that $| P(j\omega) | < 1$. If $\|P\|_\infty \geq 1$ then at some frequency this would equal 1 and $(I - P^T P)^{-1}$ would not exist. That is $\| P \|_\infty < 1$ and H has no eigenvalues on the imaginary axis are equivalent.

2.6.2 Proof That X is Positive Semi-Definite

The solution of the associated Riccati equation X satisfies :

a) X is symmetric;

b) X satisfies the algebraic Riccati equation (2.52)

c) A + RX is stable.

Suppose H has no imaginary eigenvalues, X is either positive semidefinite or negative semidefinite, and (A,R) is stabilisable. Then H is member of domain (Ric).

Suppose H has the form

$$H = \begin{bmatrix} A & -BB' \\ -C'C & -A' \end{bmatrix} \tag{2.56}$$

with (A,B) stabilisable and (C,A) detectable. Then H is a member of dom(Ric) $X = Ric(H) \geq 0$ (i.e. positive semi-definite). Which follows from standard Lyapunov equation results [Doyle *et al.* 1989].

2.6.3 Solution of Algebraic Riccati Equation.

Algebraic Riccati Equations (AREs) can be solved using a number of difference techniques such the eigenvector or Schur method [Laub 1979]. In this thesis the eigenvector approach was used since it is an adequate, simple and well documented technique. The technique requires the determination of the eigenvalues ($\lambda_1, \lambda_2, ..., \lambda_n$) of H in order that the corresponding eigenvectors, V, can be determined

$$H = \begin{bmatrix} A & R \\ Q & -A^T \end{bmatrix} = V \begin{bmatrix} -\Lambda & 0 \\ 0 & \Lambda \end{bmatrix} V^{-1} \tag{2.57}$$

where $\Lambda = \text{diag}(\lambda_1, \lambda_2, ..., \lambda_n)$ and $V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$

The first n columns of (V_1) of the matrix V form the stable eigenspace of H and also provide the ARE solution :

$$X = V_{21}V_{11}^{-1} \tag{2.58}$$

2.7 PARAMETER ESTIMATION TECHNIQUES.

A step or swept sine wave test can be used to identify a second order system relatively easily [Raven 1987]. However such an approach is not applicable to the on-line identification of model parameters (for use in an adaptive control system).

Many different parameter estimation techniques exist although most are variations to a few core algorithms. A survey of the main parameter estimation methods [Wilkes 1995] showed that least-squares parameter estimation requires the least computation making it the easiest and fastest to implement for the majority of practical applications [Hsia 1977, Isermann *et al.* 1973, and Lamanna *et al.* 1981]. The estimation is achieved by minimising the error in the discrete equation of motion of the system. The presence of non-Gaussian noise leads to biased, and therefore inaccurate, parameter estimates. The level of parameter inaccuracy is dependent on the noise properties.

More complex identification schemes such as ARMAX or the Box-Jenkins method attempt to improve the parameter accuracy by reducing the effect of any noise bias. The methods increase accuracy by characterising the noise, but this requires more computational effort and hence compromises any benefits achieved through reduction in computational delays for real time applications. A comparative study of estimation methods showed that least squares parameter estimation required 40% of the computation of instrumental variables, the next simplest technique. Any estimation scheme requires a sufficient number of points, to allow noise averaging, to distinguish between system output noise and parameter variation.

Parameter estimation techniques can be utilised in batch or recursive forms. In batch form a set of parameters is retrospectively estimated over a given time period. In recursive form parameter changes are estimated as they occur, albeit with some degree of time delay. Recursive formulations utilise a forgetting factor term to ensure that the identifier receives relevant information [Fortescue *et al.* 1981 and Zarrop 1983]. The forgetting-factor term, FF (which is limited to between zero and strictly less than unity), is usually an exponential weighting function with a sample constant, N, given by

$$N = \frac{1}{1 - FF} \tag{2.59}$$

Measurements samples that occurred more than N samples previously are included in the criterion with a weight of 36% of the current sample [Soderstrom and Stoica 1989]. Therefore the higher the FF the greater the emphasis on previous samples and vice-versa. The use of forgetting factors introduces a trade off between noise rejection and the quickest of system variation tracking. If the forgetting factor is too low the estimated parameter will simply follow a noise or input signal pattern. If the forgetting factor value is too high system parameter changes will not be tracked quickly enough. The use of a constant forgetting factor can lead to covariance "blow-up" where insufficient excitation in the system causes the data supplied to the identifier to be non-varying which causes numerical instability in the identification algorithm and hence inaccurate parameter estimates (a thorough explanation of this is given by Stripada and Fisher [1987]).

These problems can be overcome by using a variable forgetting factor which allows the forgetting factor to vary depending on system dynamics [Fortescue *et al.* 1981]. The variable forgetting factor works on the principle that the error between estimated model and the real system can provide enough information to determine a suitable FF value. If the error is small the conclusion drawn is that either the process has not been excited or that the system has been excited and the estimated parameters are nearly correct or that the estimator is sufficiently sensitive to reduce the parameter error. In all these cases a reasonable strategy would be to retain as much information as possible by choosing a higher forgetting factor. If the error is large the estimator sensitivity should be increased by choosing a lower forgetting factor, shortening the effective memory length of the estimator until the parameters are readjusted and the error becomes smaller. This treatment assumes that any lack of system excitation can be determined before covariance-variance "blow-up" occurs making the error between nominal output and actual large.

Wilkes [1995] completed a review of parameter estimation techniques and determine that a recursive least squares parameter estimation technique was most applicable to this application since:-

1) It is conceptually simple and is generally understood.
2) Recursive identification does not require the storage of large amounts of data.
3) A variable forgetting factor avoids numerical problems, such as covariance "blow-up", and allows the identifier properties to modified depending on the system variation.
4) The technique is 2.5 times faster than the nearest alternative technique.

A brief summary of the recursive least squares parameter estimation technique is given below (further details can be found in Soderstrom *et al.* [1976], Ljung [1987], Soderstrom and Stoica [1989] and Hsia [1977])

$$EP(t) = EP(t\text{-}1) - \frac{CM(t\text{-}1)x_t\ [x_t^T EP(t\text{-}1) - y(t)]}{g} \qquad (2.60)$$

$$CM(t) = \frac{1}{FF}[\ CM(t\text{-}1) - \frac{CM(t\text{-}1)x_t\{x_t^T CM(t\text{-}1)\}}{g}\ ] \qquad (2.61)$$

$$g = [\ FF + x_t^T CM(t\text{-}1)\ x_t] \qquad (2.62)$$

$$FF(t) = 1 - \frac{[y(t) - x_t^T EP(t-1)]^2}{FFW\ g_f} \qquad (2.63)$$

In the above equations, y(t) is the plant output at time t, x_t is a set of past system inputs and outputs (the form of which depends on the model to be identified), EP(t) is a vector of the current parameter estimate and FF is the forgetting factor. The CM(t) is the covariance matrix at time t. FFW is the forgetting factor weight, which determines the rate of change of FF, it should be "low" for rapid parameter change and high for slowly changing systems [Fortescue *et al.* 1981].

2.9 SUMMARY.

The basic theory of H∞ controller synthesis has been explained and the state-space solution method of Doyle *et al.* [1989] and Doyle and Glover [1988] has been shown. The general simplifications of Safanov *et al.* [1989] have been briefly outlined. The eigenvector approach to the solution of an ARE has been shown. The least squares parameter estimation technique with variable forgetting factor has been given.

CHAPTER 3

SYSTEM LIMITATIONS AND SIMULATION

3.1. DRIVE TEST FACILITY.

This chapter describes the equipment used and the initial studies completed on the servo systems. The limitations on servo system performance, as identified by Seaward [1989], are discussed. Details of a comparative study undertaken to examine differences between the response of the simulations used for the initial controller verification and the real system are given. The justification for using a second order nominal model for use in the controller design is presented and the design of digital controllers is reviewed.

3.1.1 Test Facility Configuration.

The Drives Test Facility at Aston University Drives consists of Electro-Cratft™ Bru 200 and 500 brushless dc servo systems [Electro-Craft™ 1987] coupled via torsionally stiff, laterally flexible couplings to eddy current couplings, mechanical mechanisms or dynamometers (which act as variable loads). Torque transducers can be fitted to the shaft coupling to produce a fully instrumented test rig. Each brushless dc servo system comprises a control unit, a power supply module and a brushless neodymium-boron-iron permanent magnet synchronous motor. The motors have optical encoders which produce 8000 quadrature pulses per revolution to provide position / velocity feedback information.

3.1.2 Servo System Internal Control Loops.

The servo system's control unit contains control loops for the correct functioning of the motor (for example determining the correct supply to each motor phase). The servo systems have two control modes, "velocity" and "torque". In torque mode an analogue torque input signal is converted to the required current magnitude and inputted into three internal current regulators (one for each phase of the motor) which control the motor current. Torque mode is essentially current mode since the control variable is the current supplied to the motor. In velocity mode, an analogue velocity demand input signal is converted into an equivalent digital velocity demand signal which is compared to the motor velocity signal, taken from the motor's optical encoder. The difference between the two signals is the error signal used in the systems internal Proportional, Integral and Derivative (PID) control loop.

The servo system's internal variable parameter Proportional Integral and Derivative (PID) velocity control loop operates on the velocity error signal, with an update time of 1ms. Proportional control provides a control signal proportional to the velocity error. Integral control is used, in conjunction with proportional control, to reduce steady-state errors by adding a signal proportional to the integral of the error to the control signal. Integral action tends to make the system unresponsive and decreases the maximum possible system gain before instability occurs. Derivative control, used in conjunction with proportional control, increases the speed of response and extends the system bandwidth by adding a signal proportional to the rate of change of the error signal. In practice derivative control also acts as a noise amplifier, particularly in sampled data systems [Tal 1989]. The control module contains a variable low-pass filter situated after the control output used to limit the effects of disturbances in the frequency bandwidth of the system (i.e. mechanical resonances).

3.1.3 Outer Positional Control Loop.

An outer positional control loop is provided by a Themis TSVME440 controller card [Themis 1988] mounted in a VME rack with a MVME147 Motorola MC68030 card. The themis card runs assembler and 'C' code which enables a standard PID with VFF controller or any generalised control algorithm (the general control software was developed by the author) to be implemented. The system allows any digital demand profile to be implemented and data logging of critical signals such as command, position errors, desired position and actual position. The TSVME440 implements control algorithms with an update time of 1ms and has the ability to receive higher level commands from an external processor, via a VME bus, to alter the control algorithms on-line. The system works under the OS9 operating system. The MC68030 is used to complete data processing and determine high level control action. The logged data is transferred via an ethernet network to either a PC or a Sun Workstation for data analysis.

3.1.4 Selection of Velocity Mode For Servo System Testing.

The servo system's were always operated in velocity mode since:-

1) Experience has shown that they are easier to control in velocity mode.
2) The industrial sponsor uses them in velocity mode, therefore to aid the direct transfer of algorithms and allow a comparison of results the same mode of operation was selected.
3) The majority of the research completed in thesis describes a comparative study, with only the outer positional loop being adjusted, therefore the servo system's mode and settings were "immaterial" provided they were kept constant between tests.

4) Using the servo system's in a single mode allowed a degree of "experience" to be developed with the servo system.

3.2 SYSTEM STUDIES AND PERFORMANCE LIMITATIONS.

3.2.1 Initial Studies

Discussions with Electro-Craft™ and initial studies completed on the servo systems yielded a number of crucial points. The internal velocity control loop has an update time of 1 ms and the internal current loop has an update time of 250 µs. A 1 ms time delay is introduced into the system by the analogue to digital converter, based on a voltage controlled oscillator, on the servo system input. The outer position control loop introduces a further 1ms time delay into the system resulting from the computational delay of the Themis card.

3.2.2 Torque and Current Limits

The torque and current limits for each motors and drive module combination are readily available [Electro-Craft 1987] and can be used to determine initial limits on the motor/drive performance. The airgap torque from the servo motor has to be shared out between different torque "consumers" as 3.1 indicates.

$$T_g = T_f + J_t\frac{d\omega}{dt} + T_l + B_v\omega \qquad (3.1)$$

where T_g = Torque generated by the motor (Nm)

T_f = Friction Torque (Nm)

T_l = Load Torque (Nm)

B_v = Viscous Friction (Nm/rads^{-1})

J_t = Total driven inertia (kgm^{-2})

It can clearly be seen that there is a limit on the ability of the independent drive to follow a given profile due to the limited amount of torque available from the motor (effectively a limit on the attainable acceleration). This is not usually a limiting factor since the sizing of motors and choice of optimal gearing ratio is reasonably well understood [Tal 1989], a brief resume of motor selection and optimal gear ratios is given in Appendix B.

3.2.3 Effects of Sampling.

Time delays are introduced to the system due to the digital nature of the internal servo system velocity control loop and the outer positional control loop. Considering a simplified system representation, see Figure 3.1., the accumulated time delay is (at least) 3ms (the time delays correspond to those described in Section 3.2.1).

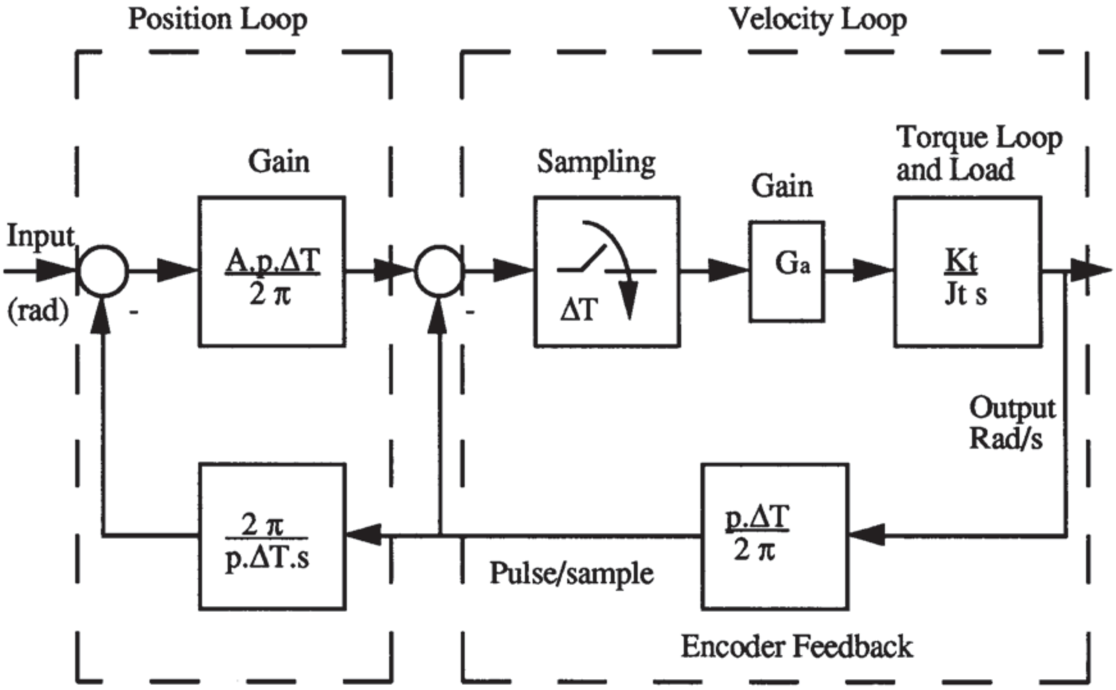


Figure 3.1 Position-Loop Bandwidth.

These time delays limit the ability of the system to track position profiles for two reasons. Firstly due to the short cycle times in this application (under 10ms in some cases) the time delay effectively reduces the time available for the increment of an axis (i.e. requiring the servo system to accelerate at a higher rate, which may not be possible). Secondly the phase shift introduced by the sampling delays limits the controller gains which can be used before instability occurs. Using standard control techniques Seaward [1989] developed the following gain and phase equation, corresponding to Figure 3.1, (for the outer positional loop)

$$\text{Gain} = \frac{A/\omega}{\sqrt{(1 + \tau\omega\cos(\pi/2 + 2\pi\tau\omega/K_r))^2 + (\tau\omega\sin(\pi/2 + 2\pi\tau\omega/K_r))^2}} \tag{3.2}$$

$$\text{Phase} = -\pi/2 - 2\pi\tau\omega/K_r - \tan^{-1}\left(\frac{\tau.\omega\sin(\pi/2 + 2\pi\tau\omega/K_r)}{1 + \tau\omega\cos(\pi/2 + 2\pi\tau\omega/K_r)}\right) \tag{3.3}$$

where $K_r = (2\pi\tau) / \Delta T$.

Seaward demonstrated that the value of G_a, the internal velocity-loop gain, can be set at a maximum value such that the velocity-loop bandwidth is one quarter of the sampling rate. If this limit is used to determine the limit on the position-loop gain, A, it is found that the value of A for a gain margin of 8dB and a phase margin of 59^o the closed-loop system would have a position gain of -6dB at 14.8Hz. Therefore the usable bandwidth of the system, where the input to output ratio is unity, would, probably, be less than 14.8Hz (dependent on the acceptable level of error). A study of the actual system shows that the phase shift is greater than used in this simple treatment and that a more realistic usable closed-loop bandwidth would be of the order of 10Hz (for the outer positional loop).

Shannon's Sampling Theorem (Nyquist limit), as described by Lynn and Fuerst [1993], states that the maximum signal frequency that can be accommodated by a digital system is 1/2 of the sampling rate (before aliasing occurs). In this case the Nyquist limit would be at a frequency of 500Hz, since the controller samples every millisecond (a 1 KHz sampling rate). But a study of the command output from the Themis card demonstrated, due to its digital nature, that a more realistic frequency limit was 1/5 of the sampling rate or in this case 200Hz, still much higher than the achievable system position loop bandwidth.

3.2.4 Quantisation Effects.

The optical encoder resolution limits the system performance for two reasons. Firstly the encoder resolution limits the absolute achievable positional accuracy to ±1 pulse, for the motor optical encoders used this is equivalent to ±0.045°. Secondly when the axis is required to hold position (which is common for this application) a quantisation ripple is seen on the error signal, as the motor shifts between optical encoder pulses. The open-loop gain is restricted to a level such that the quantisation ripple does not cause the axis to go out of specification, i.e. a level such that the quantisation ripple becomes to large.

3.2.5 Mechanical Resonances.

Mechanical resonances, excited by the quantisation ripple inducing mechanical excitation, limit the attainable bandwidth since the controller is required to be set-up such that the mechanical resonances are not persistently excited. The effect of mechanical resonances can usually be removed using the software filter in the servo system, but this effectively reduces the system bandwidth. Resonances cause system oscillations which often cause an axes to go out of specification and in general hamper system performance. A complete analysis of mechanical resonances is difficult and beyond the scope of this thesis, but the approximate resonance of a two inertia mechanical system can be determined using the ratios of the load and motor inertia and the torsional spring constant, K_{sp}, of the connecting shaft. The approximate resonant frequency (in rad/s) is

$$\omega_r = \sqrt{\frac{K_{sp}(J_m + J_l)}{J_m \, J_l}} \tag{3.4}$$

3.2.6 Effects of Backlash.

The effect of backlash is to disconnect the motor and load momentarily. The system may become unstable during this time period due to the increase in the open-loop gain caused by the reduction in inertia. Once coupled again the system should become stable. The effect is more pronounced when the ratio of motor to load inertia is large and in this instance the system will vigorously "jitter" within the backlash zone. This can damage gearing, cause overcurrents and cause the system to emit high levels of acoustic noise. To enable satisfactory performance in the presence of backlash the system must be stable with only the motor and input gearing inertia. Seaward showed that if the system is set up to be stable for the motor and input gearing only, the system performance will be degraded by the ratio of the inertias. Additionally the decoupling of the load and motor can lead to load positional inaccuracy (the position information comes from the motor shaft) since the load can move independently of the motor. In this application the load is directly coupled using taper locks which should overcome the problem of backlash.

3.2.7 Thermal Limitations.

The servo systems contain thermal sensors which cause shut down if the motor overheats. The motor temperature limits can be identified for a motor and profile combination [Barber 1984]. The root mean square torque can be calculated from

$$T_{rms}^2 = \frac{1}{\text{Time Period}} \int_0^{\text{Time Period}} T_g^2 \, dt \qquad (3.5)$$

and since $I_{rms}^2 = \frac{T_{rms}^2}{K_t}$ and power $P = I_{rms}^2 R_\phi$, the temperature rise can be calculated from $T_\Delta = R_{TH}P$ where R_{TH} is the thermal resistance.

3.2.8 Electromagnetic Interference.

The servo systems emit Electro-magnetic interference (EMI - electrical noise believed to be produced by the power modules pulse width modulation unit) which hampers the measurement of analogue quantities such as torque.

3.2.9. Previous Work on Overcoming System Limitations.

Seaward [1989] analysed all of the inherent limitations on the dc servo systems and eliminated the time delay on the Bru 500 servo system input to increase the achievable system bandwidth. This approach was not repeated since:-

1) The study was designed to be a comparison of control techniques.
2) The results had to directly transferable to a standard industrial servo system.

3.3 SYSTEM SIMULATION.

Simulations are a valuable design tool in controller determination allowing a greater understanding of the system to be obtained and allow easy and rapid testing of different system configurations. Seaward and Vernon [1991] used simulation analysis to increase the velocity loop bandwidth of a system from 20 to 90 Hz. The simulations produced by Seaward [1989] were readily available and were used instead of developing independent simulations, after verifying their validity, to avoid the repetition of work.

Two Advanced Continuous Simulation Language (ACSL) [Mitchell and Gauthier 1991] simulations were available. The first was relatively simple, based on Figure 3.1. The simulation included all non-linearities seen in a real servo system, such as current and voltage limits, friction torque, the servo system control filter, a digital internal velocity PID controller, time delays due to sampling and computation and data sampling through an optical encoder (a listing of this simulation is given in Appendix C). The second simulation was more complex and had the additional features of back EMF, motor resistance and inductance, and three phase torque generation (both simulations are shown schematically and detailed in Seaward [1989]). A series of simulated and real frequency and step responses were taken for different combinations of motor and drive. An example of the results can be seen in Figures 3.2 a and b. which show a comparison of the frequency response of the two simulations and an actual S-4075 and DM-25 combination.

Both simulations give reasonable approximations to the real system (solid lines). The simple simulations (dashed lines) gave an average magnitude error of 2.4805 dB and phase error of 22.89° (for the frequency points tested). The more complex simulation (dotted lines) gave an average magnitude error of 1.2295 dB and phase error of 22.89° (for the frequency points tested). The frequency response shows that the system is non-minimum phase (as expected since the system contains time delays [Golten and Verwer 1991]), and the gain fall off (for the practical system) of approximately 40 dB/dec (34.85 dB/dec) would suggest a second order system.

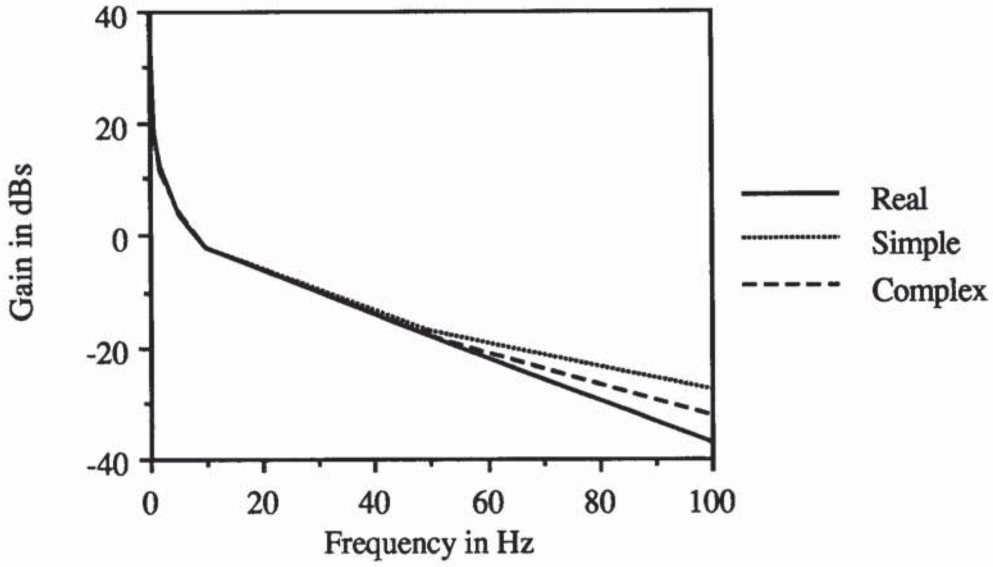


Figure 3.2a Gains For Simulated and Real System.

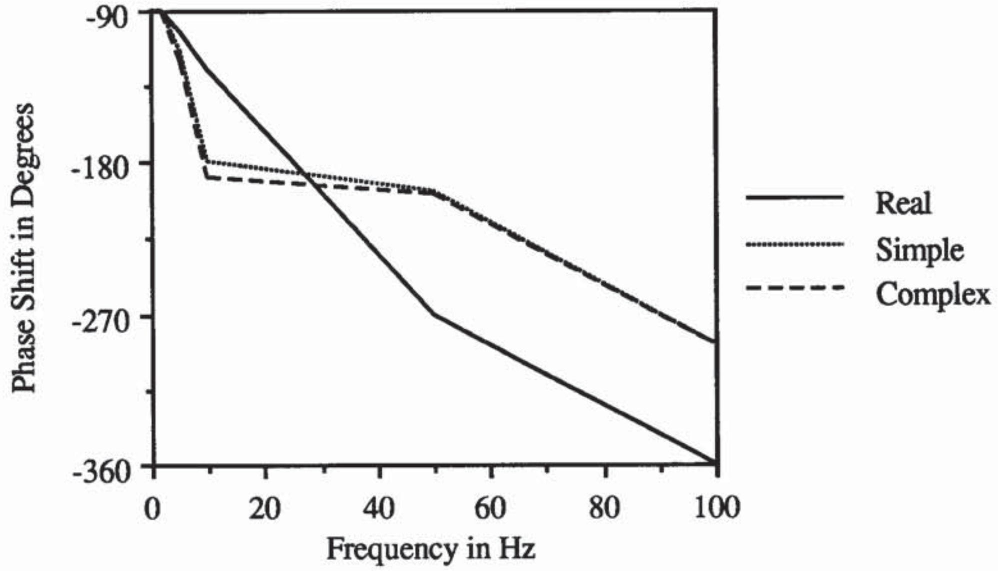


Figure 3.2b Phase Shifts for Simulated and Real System.

The simplified simulation was used for the basic studies of control algorithms and implementations since:-

1) The simpler simulations were easier to manipulate, understand and had a shorter run-time.

2) The ready availability of a fully instrumented test rig meant that simulations were only required to debug and gain experience of control algorithms and strategies.

3) Due to variations in the dynamics of "identical" dc servo systems the extra accuracy achieved by using a more complex simulation was considered unnecessary.

The simulations were enhanced by inserting a data logging routine, which saved the data in the correct format for data analysis.

3.4 SECOND ORDER SYSTEM MODEL.

3.4.1 Theoretical Perspective.

Liu and Liu [1990] completed a study of an ac servo system (dc servo systems are often referred to as ac servo system since permanent magnet synchronous motors are used) and determined that it was possible to construct a fairly accurate approximate dynamic second order model for the drive. The model for the permanent magnet synchronous motor drive was based on the field orientation and current feedback control of the inverter. The model was successfully used to design a $H\infty$ controller for an ac (dc) servo system. However an analysis of the frequency response, see Figures 3.2 a and b, of the servo systems show that they are not simple second order systems, clearly seen by the fact that they exhibit a phase shift greater than 180^o (meaning that they are either higher order or non-minimum phase systems).

3.4.2 Justification For Using Second Order Model.

The work of Liu and Liu [1990] demonstrated that a second order model was adequate for the design of a H∞ controller for a dc servo system, even though a dc servo system is not a second order system. The robustness weighting function defines a stability margin around the nominal system, such that the critical (Nyquist) point, -1, lies outside the disk of centre $L_0(j\omega)$ and radius $|W_2(j\omega)L_0(j\omega)|$ (for a multiplicative perturbation of unitary magnitude, see Figure 2.6, Chapter 2). The definition of the robustness weighting function (Equation 2.8, Section 2.2, Chapter 2) is such that the actual system is within the bounds defined by the robustness weighting function of the nominal system. Therefore the actual system will be within the bounds specified by the stability margin and a stable system will be produced, assuming that the robustness weighting function has been properly defined.

A second order system is used, instead of a third order system, since a simple controller is required, the use of a third order system model would increase the controller complexity. A lower order system model, for example a first order model, is not used since this would require an increase in the robustness weighting function order, due to the greater level of model uncertainty, which would lead to the production of a controller as complex as that produced using a second order model. Therefore the approach of Liu and Liu [1990] of using a second order model was used.

3.5 DIGITAL CONTROLLER DESIGN.

3.5.1 Introduction.

The system under study is a continuous sampled data system with a digital controller. Digital control systems are used since they are reliable, flexible, can implemented complex control algorithms and the on-line adjustment of control parameters is possible. There are essentially two approaches to digital controller design [Chen and Francis 1991] :

1) Convert the sampled data system (model) into a continuous system and use a continuous domain approach (such as H∞ controller synthesis). However the analogue performance can only be regained as the sampling period tends to zero and can never be expected to be bettered.

2) A digital system model can be determined and a digital controller determined directly. This approach has two main disadvantages. Intersample behaviour is ignored and continuous time specification may be difficult to translate into discrete time.

(The third and 'standard' method is to use trial and error, as used in the tuning of a traditional PID with VFF controller).

This thesis uses method (1), where the model is determined discretely, converted to the continuous frequency domain, where the problem is solved and the controller is then digitised. This approach was selected since:-

1) Frequency domain H∞ algorithms are readily available.
2) Guidelines are available for frequency domain weighting functions selection.
3) A number of frequency domain design examples were available.
4) Methods to transform between the continuous frequency and discrete time domain are available. The sampling rate is high compared to the expected system bandwidth frequency, therefore distortions due to conversion between the s-domain and z-domain should not be a problem.

3.5.2 Conversions Between Discrete and Continuous Time Domains.

A number of methods are available for the conversion from s to z domains (and vice-versa). Two main types of transformation are discussed. The first involves matching time waveforms, usually the system impulse response or the step response [Hewlett Packard 1989]. The second type involves rational fraction approximations such as the bilinear transformation [Chiang and Safonov 1992].

It was envisaged that the design process would become automated (i.e. an adaptive controller) and therefore the conversion between discrete and continuous time was required to be automatic, which effectively rules out matching waveforms. Since matching waveforms would probably require some form of look-up table which would be computationally expensive. Therefore a rational fraction approximation was used. A study of transformations showed that the Tustin bilinear transform was adequate for the study purposes since for the sampling rate used little frequency distortion was seen in the usable bandwidth of the system. The bilinear transform is obtained from the quotient of two first order polynomials in s

$$z = e^{sT} = (e^{sT/2}) / (e^{-sT/2}) \tag{3.5}$$

If the first two terms of the Taylor's series expansion of the numerator and the denominator are used, then z can be approximated by

$$z = (1 + sT/2) /(1 - sT/2) \tag{3.6}$$

this can be inverted to obtain the Tustin transform

$$s = (2/ T) (z-1)/(z+1) \tag{3.7}$$

Note that (3.7) is used to obtain the digital controller from the continuous controller.

3.6 CONCLUSION

This Chapter has described the Drives Test Facility, at Aston University, and examined the inherent limits placed upon the performance of dc servo systems due to torque and current limits, sampling delays, thermal effects, backlash, mechanical resonances, quantisation effects and sampling rate limits. An example of a comparative study, undertaken to show the validity of dc servo system simulations, was detailed. A justification for using a second order nominal model was given. Methods of digital controller design were described.

CHAPTER 4

INITIAL DESIGN EXAMPLE USING H∞ CONTROLLER SYNTHESIS.

4.1 INTRODUCTION

In order to assess the viability of using H∞ controller synthesis for high speed independent drives a single axis industrial design example was studied. The design example was selected for the following reasons

1) The performance required was within the physical limitations of the system.
2) Two independent research groups had unsuccessfully attempted the problem using a traditional PID with VFF controller, both on a simulated and real system.

Therefore a well defined and previously unsolved design problem was available. The results clearly demonstrate the improvement, in terms of reduced cycle time, achievable using a H∞ controller (for the particular design example) and are a justification for the work completed in the remainder of the thesis.

4.2 SIMULATED DESIGN EXAMPLE.

4.2.1 Cold Forming Design Example.

Initially the design example was carried out in simulation for a single-axis, non-linear, time-varying system: a cold forming machine, derived from a real case study. The ACSL simulation very accurately models the response of the real servo system (see Chapter 3, section 3.5). The simulation models all of the non-linearities seen in a real servo system, such as current and voltage limits, friction torque and data sampling through an optical encoder. The controller is modelled in discrete time with an update time of 0.33 ms whereas the servo system is modelled as a continuous process with a digital controller (with a 1ms update time). Time delays due to sampling and control law computation are included. The simulation also incorporates a 204 Hz mechanical resonance (as exhibited on the real system) which imposes limits on the controller gains, thereby limiting the achievable performance. The simulated system represents a BRU 500 dc servo system consisting of a DM150 drive module with a S-6300 motor.

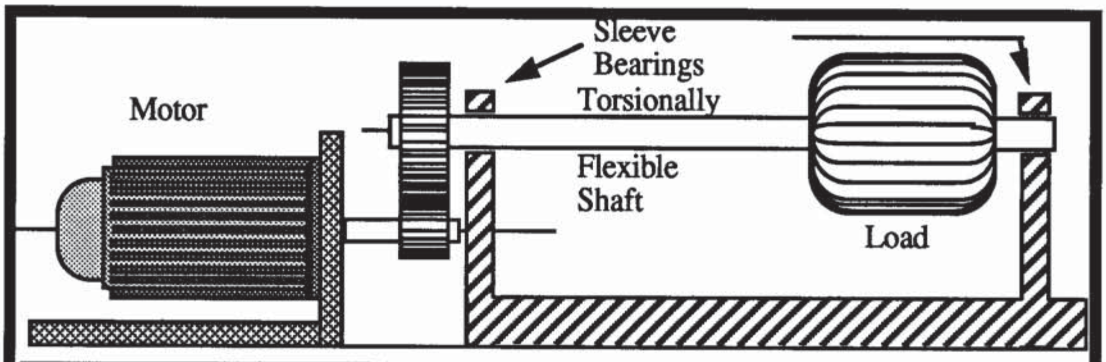
Depending on the characteristics of the product under process, the total inertia of the drive rotor and the load (referred to the motor shaft) can take any value in the range 0.006 - 0.01 kgm^2 and the control system (not necessarily a single controller) is expected to deal with all values of inertia. The referred inertia was set at its maximum value for the analysis, since this is considered to present the most onerous control problem. The total static friction torque is 3 Nm and the viscous damping coefficient acting on the load is 0.022 Nm/rad s^{-1} (referred to the motor shaft).

The load is required to perform a repeated incremental duty cycle with cycle time 150 ms. The duty cycle requires the motor to index through 144° in 50 ms at the start of the cycle and then to hold its end position to ±0.1° for the remaining 100 ms. The actual motion of shaft within the first 50 ms is irrelevant and for the purposes of this study, the reference input, r(t), (angular position of motor shaft in degrees) was given as

$$\begin{aligned}&\text{If } t < 40\text{ms} \ldots r(t) = r(0) + 72^{\circ} \times \{1 - \cos\left(\frac{\pi t}{40 \times 10^{-3}}\right)\}\\ &\text{If } t \geq 40\text{ms} \ldots r(t) = r(0) + 144^{\circ}\end{aligned} \qquad (4.1)$$

which allows 10 ms initial settling time. The reference input, r(t), was determined such that the rate of change of demanded acceleration was sinusoidal (i.e. contain no sudden changes in demanded acceleration which may have caused the servo drive to hit its current limit and shut down).

The angular position of the motor shaft is measured by an optical encoder and the requirement to hold position to within ±0.1° translates into ±5 encoder pulses. A drive system was selected capable of supplying enough peak and continuous torque for the required levels of speed and acceleration. The optimum gearing ratio was fixed using the established principle of inertia matching [Tal 1989]. Figure 4.1 illustrates the set-up and summarises the key parameters.



Motor Constant	= 0.7 Nm/A.	Shaft Encoder	=18000
Rotor + gears Inertia	= 0.00373 kgm2	pulses/rev	
Max.Load Inertia (ref.d)	= 0.0066 kgm2	Dry friction per	
Torsional Stiffness of		sleeve bearing (ref.d)	= 1.5 Nm
Load Shaft (ref.d)	= 3.575 kNm/rad	Backlash (ref.d)	= 0.004 rads
Load Visc.Dmp.(ref.d)	= 0.022 Nms/rad		

Figure 4.1 Schematic of Motor and Load with Key Parameters

The control of the motor and load combination was achieved using the drive package supplied with the motor which has an inbuilt PID control loop for velocity and automatic current limiting to protect the motor (as described in Chapter 3)

4.2.2 Identification of Nominal Model and Determination of Robustness Weighting Function.

A random noise signal (equivalent in magnitude to the demanded position increment) was applied to the dc servo system simulation and the least squares parameter estimation method was applied in the time domain to determine a nominal model of the form:

$$y(t) = a\,u(t) - (b1\,y(t\text{-}1) + b2\,y(t\text{-}2)) \qquad (4.2)$$

where $u(t)$ = system input at time t.

$y(t)$ = system output at time t.

a, b1, b2 = respective time series coefficients.

The coefficients a, b1, b2 were determined as 35.0928, –1.5036 and 0.5036 respectively. The discrete time model was converted to the frequency domain model representation, using a bilinear transform (Equation 3.7, Chapter 3, section 3.5), of

$$\text{Nominal Model} = \frac{5.3898\text{x}10^{8}}{s^{2} + 2.2886\text{x}10^{3}s + 892.2749} \qquad (4.3)$$

The frequency response of the simulated system was obtained using a swept-sine test (from controller output to motor position output). Figures 4.2a and b show the nominal model (dotted line) and simulated system (solid line) frequency response.

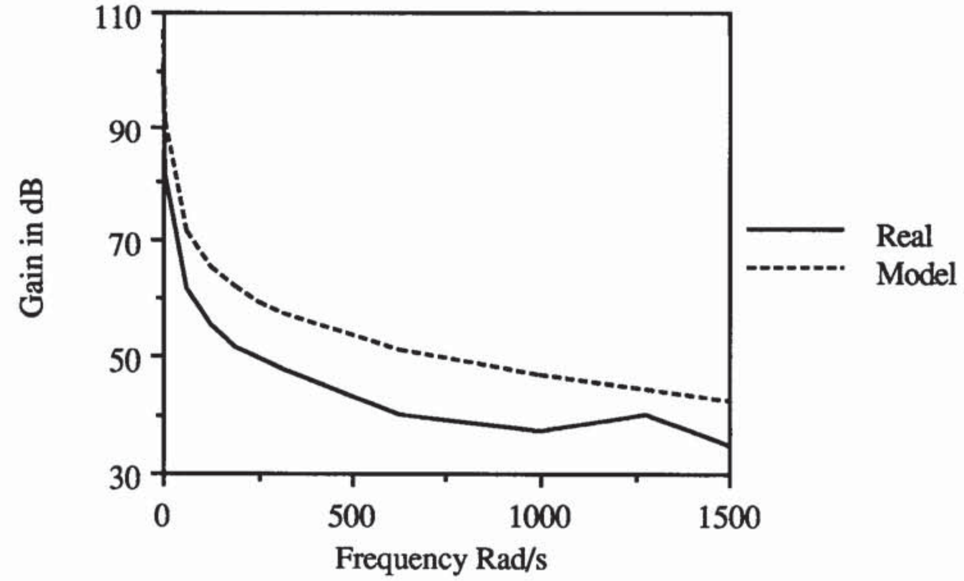


Figure 4.2a Gains For the Nominal and "Real" System

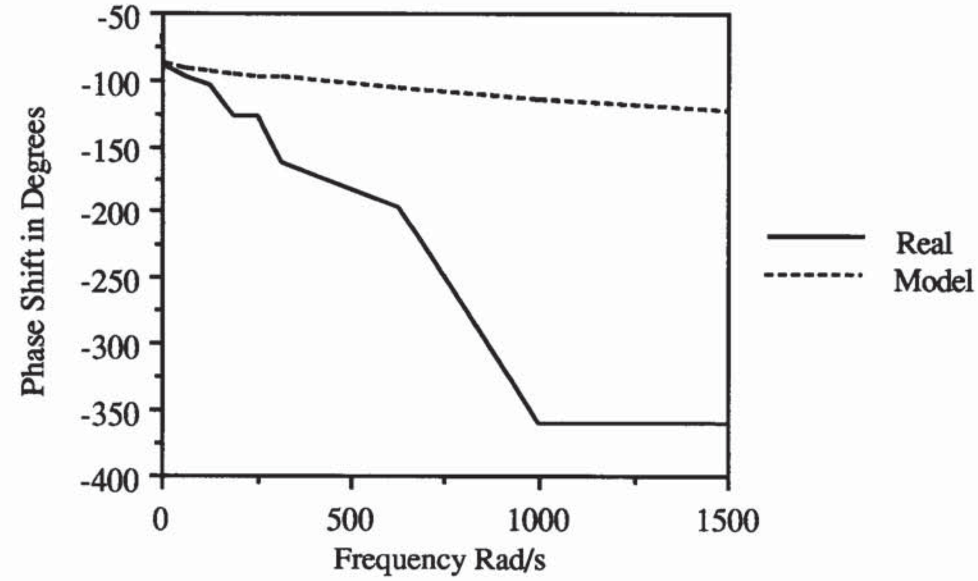


Figure 4.2b Phase Shifts for Nominal and "Real" System

The robustness weighting function was determined, using the multiplicative perturbation model formula, as

$$W_2 = \frac{s^2 + 1.5x10^5 s + 1.9x10^{1C}}{4x10^{10}} \qquad (4.4)$$

To reduce the number of states of the augmented state-space plant an improper robustness weighting function was used. The lower the number of augmented state-space plant states the lower the controller complexity [Chiang and Safonov 1992]. The weighting function was selected to guarantee a minimum nominal gain margin of 3.12 dB and a minimum nominal phase margin of 25° (see section 2.4). For a practical system a robustness weighting function with at least a 6 dB gain margin and a phase margin of at least 60° would be used, since real systems exhibit greater parameter variations than the simulation, due to unmodelled dynamics (e.g. thermal effects).

4.2.3 Performance Weighting Function Selection.

A performance weighting function was determined using the method suggested by Francis [1990]. To begin the performance weighting function determination the ideal closed loop system response, denoted T_{id}, is selected. The selection is based on any performance criteria which are available such as a particular bandwidth or a tracking error less than a given value. The ideal sensitivity function, S_{id}, response is then calculated from

$$S_{id} = 1 - T_{id} \tag{4.5}$$

The performance weighting function, W_1, was determined as the inverse of the ideal sensitivity function:

$$W_1 = 1 / S_{id} \tag{4.6}$$

In this case the performance weighting function was determined as

$$W_1 = \frac{0.99\ s + 53.5}{s + 0.01} \tag{4.7}$$

which corresponds to a step response time to steady-state of 10 milliseconds

The performance specification was so demanding that a number of iterations were required to determine this performance weighting function (effectively tuning the controller). This is due to the H∞ controller synthesis producing a controller satisfying nominal performance requirements, with inaccuracies in the nominal model producing a disparity between the expected and actual performance. The selection of performance weighting functions is discussed further in chapter 6. The performance weighting function was readjusted and the design procedure repeated, instead of simply tuning the controller parameters, to ensure the integrity of the controller (in terms of theoretical stability and performance criteria). If the controller parameters had been adjusted using a trial and error method, any possible benefits of using the technique would have been corrupted (i.e. it would have been the same as tuning a PID controller, although the controller would have a different structure). Analysis of the frequency response of the final performance weighting function shows that it is a low-pass filter with a 0dB crossover point of approximately 64Hz, effectively constraining the controller to filter out the mechanical resonance frequency.

4.2.4 Controller Implementation.

The nominal model, robustness weighting function and performance weighting function were combined into an augmented state-space representation using the Matlab script augtf.m and a controller produced using the Matlab script hinf.m [Chiang and Safonov 1992]. The controller was determined as

$$K(s) = \frac{0.8623\ s^2 + 1.9734 \times 10^3\ s + 768.94}{s^2 + 7.5901 \times 10^5\ s + 7.5901 \times 10^3} \qquad (4.8)$$

The controller was converted, using a bilinear transform, to the digital time series

$$u(t) = c1\ e(t) + c2\ e(t-1) + c3\ e(t-2) + d1\ u(t-1) + d2\ u(t-2) \qquad (4.9)$$

where u(t) is the controller output.

e(t) is the positional error.

c1, c2, c3, d1 and d2 are the controller coefficients (respectively 0.0095, 0.0052, -0.0043 –1.9694 and –0.9699).

The controller was implemented as a positional time series controller around the inner velocity loop of the complex non-linear simulation.

4.2.5 Simulated Results.

The system responses obtained using the two controllers can be seen in Figures 4.3 (for the sake of clarity only the crucial 100 ms holding errors are shown). Even after lengthy tuning, the combination of the traditional PID position control loop with VFF and the inner velocity control loop was unable to meet the stringent performance criteria, achieving an indexing time of 62 ms (against a requirement of 50 ms) for the highest value of load inertia (0.0066 kgm^2 referred to the motor shaft), shown in Figure 4.3 (dotted line). An extensive period of tuning was completed on the PID and velocity feedforward coefficients to check that they were optimal, in terms of time to reach the required holding error. The H∞ controlled system met the specification in 50 ms, Figure 4.3 (solid line).

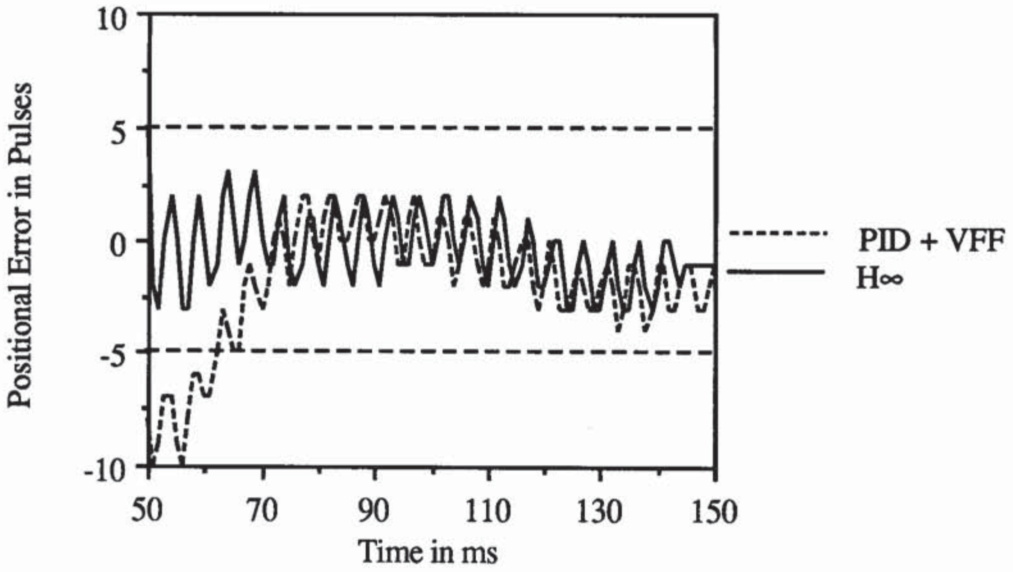


Figure 4.3 Holding Errors For PID with VFF and H∞ Controlled Systems.

4.2.6 Robustness Tests on Simulated System.

The design and initial comparison of the two controllers was for the particular case of maximum inertia with no parameter variation (except between static and viscous friction). An examination of the effect of parameter variation on the system was completed to assess the level of robust performance of the two controllers. The average absolute error over the holding period (the 100 ms period) was taken as a performance measure since:

1) The time to meet specification can be distorted by transients in the simulation.
2) Similarly the maximum error variance can be distorted by system transients.
3) The average absolute error per sample combines both the variance and the time to meet specification into a single value.

The torque constant, k_t, was varied from 0.65 to 0.75 Nm/A (its initial value was 0.7 Nm/A), the referred viscous damping, B_v, was varied between 0.017 and 0.027 Nm /

Rad s^{-1} (its initial value was 0.022 Nm / Rad s^{-1}) and the referred inertial load was varied between 0.006 and 0.007 kgm^2 (its initial value was 0.0066 kgm^2). The average error per sample period can be seen in Figures 4.4, 4.5 and 4.6 (the solid line is for the $H\infty$ controlled system while the dotted line is for the PID with VFF controlled system). Over the variation in the torque constant, the $H\infty$ controlled system showed a 43.6% less variation in average absolute error per sample than the traditionally controlled system. For the variation in the viscous damping the $H\infty$ controlled system exhibited a 54.7% less variation than the traditionally controlled system. For the inertial variation the $H\infty$ controlled system displayed a 59.5% less variation in the average absolute error per sample.

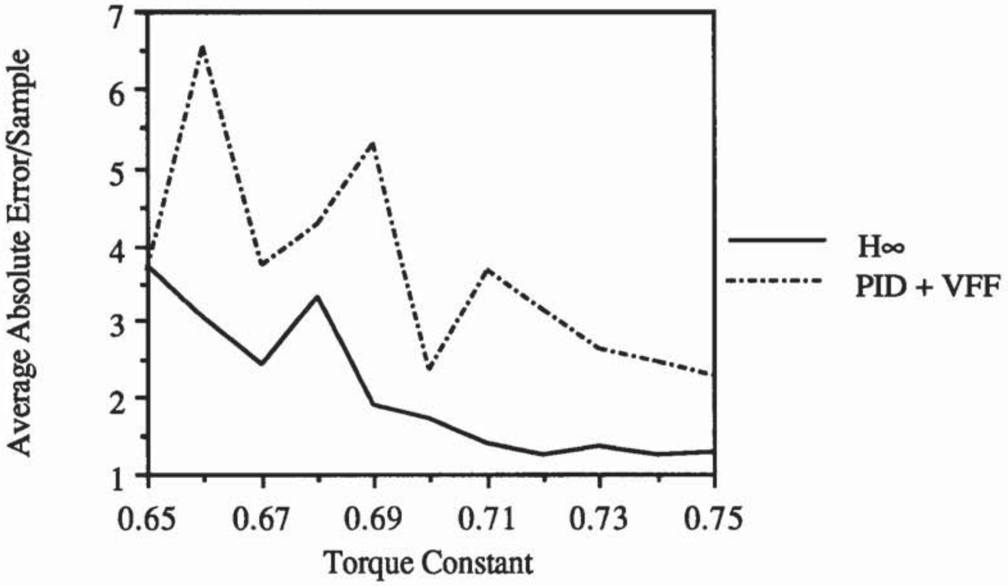


Figure 4.4 Average Absolute Holding Error with Varying Torque Constant

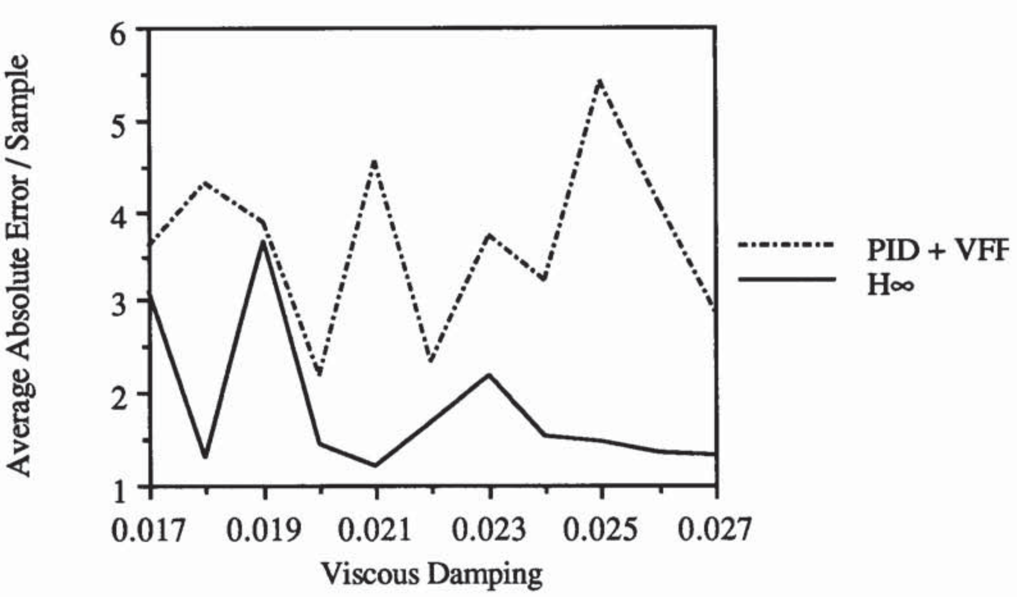


Figure 4.5 Average Absolute Holding Error with Varying Viscous Damping

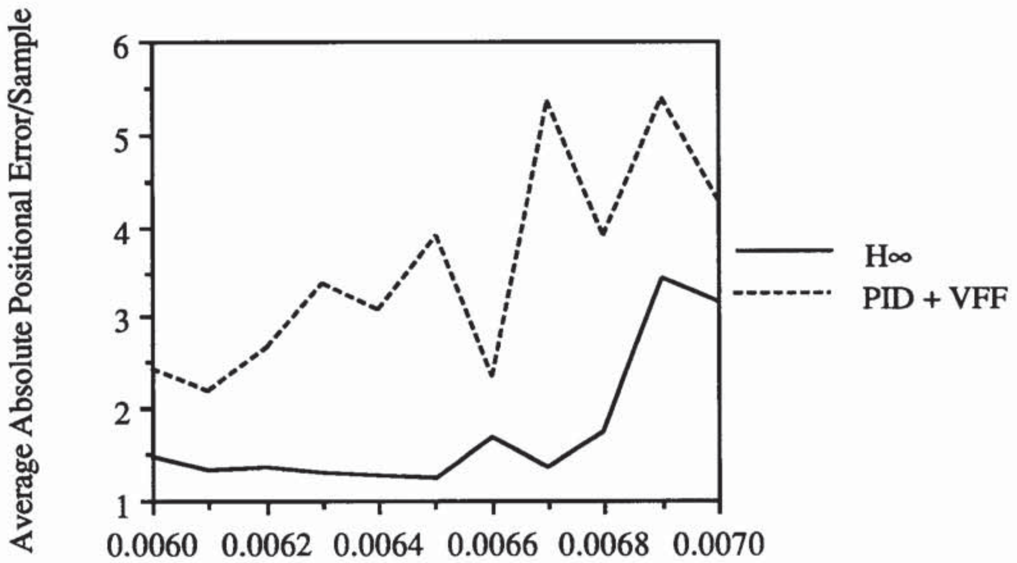


Figure 4.6 Average Absolute Holding Error with Varying Load

The Bode plots for the open-loop plant and the two controllers, Figures 4.7a and 4.7b, show that the H∞ controlled system had a gain margin of 7.13 dB and a phase margin of 78°. The traditional PID controller had a gain margin of 6.38 dB and a phase margin of 72°. The PID controller was analysed without VFF since the performance specification is in terms of holding error and when the rate of change of the demanded position is zero, the VFF controller component has no effect. Additionally the VFF controller component acts as an external controller signal which improves performance but does not affect the system stability.

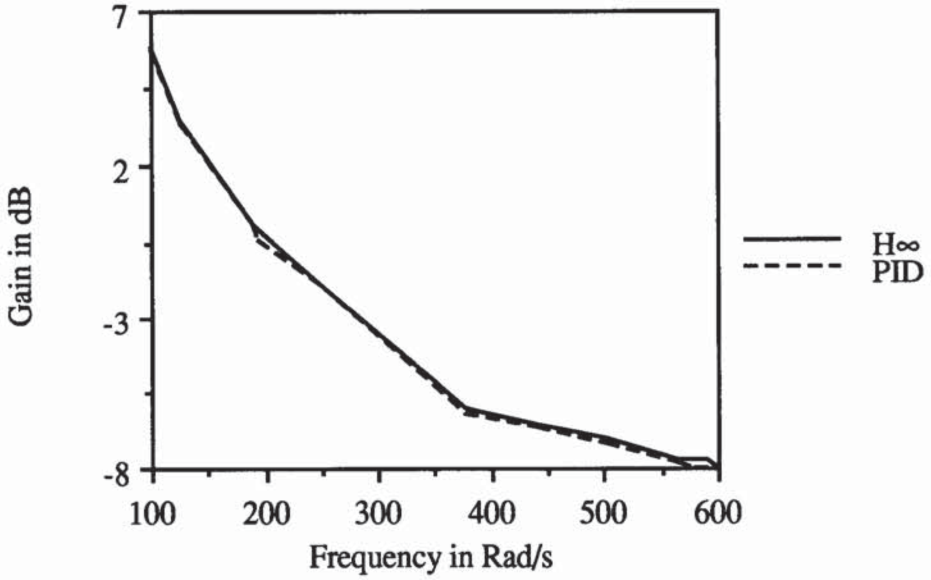


Figure 4.7a Gains For Open-Loop Controllers and Plant.

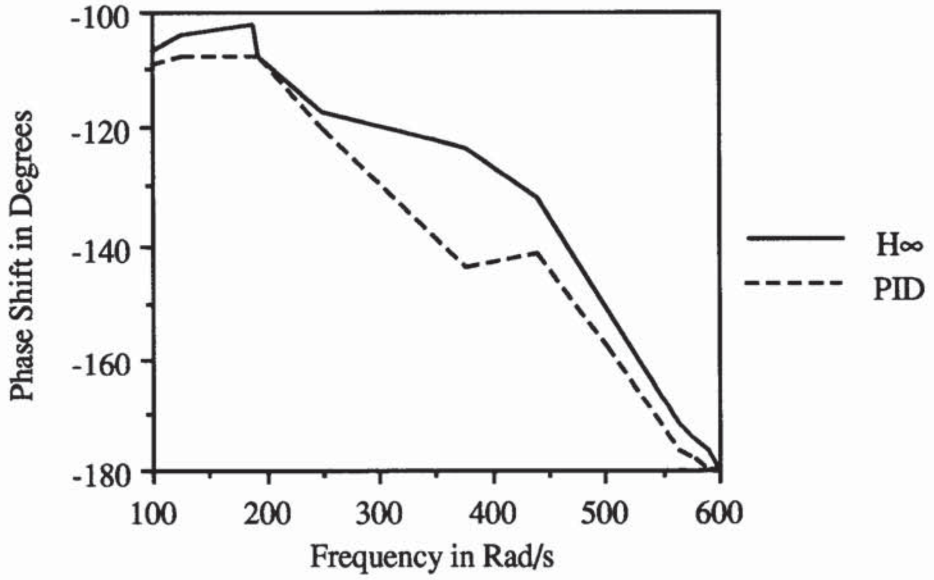


Figure 4.7b Phase Shifts For Open-Loop Controllers and Plant.

4.2.7 Analysis of H∞ Controller and PID Controller.

4.2.7.1 Frequency Response Analysis.

A study of the digital controllers and the closed-loop systems was undertaken to determine the reason for the better performance achieved by the H∞ controlled system. The closed-loop frequency responses, Figures 4.8a and 4.8b, show that at lower frequencies the H∞ controlled system (solid line) has a lower gain than the PID controlled system (dotted line). At higher frequencies the H∞ controlled system has a higher gain than the PID controlled system.

The controller frequency responses, Figures 4.9a and 4.9b, show the H∞ controller (solid line) acts as a lead controller with a high gain at high frequencies and a lower gain at lower frequencies (compared to the PID controller, dotted line). The H∞ controller has a frequency response that has been shaped to meet the required performance. This is an example of the loop shaping property of H∞ controller synthesis [McFarlene 1992].

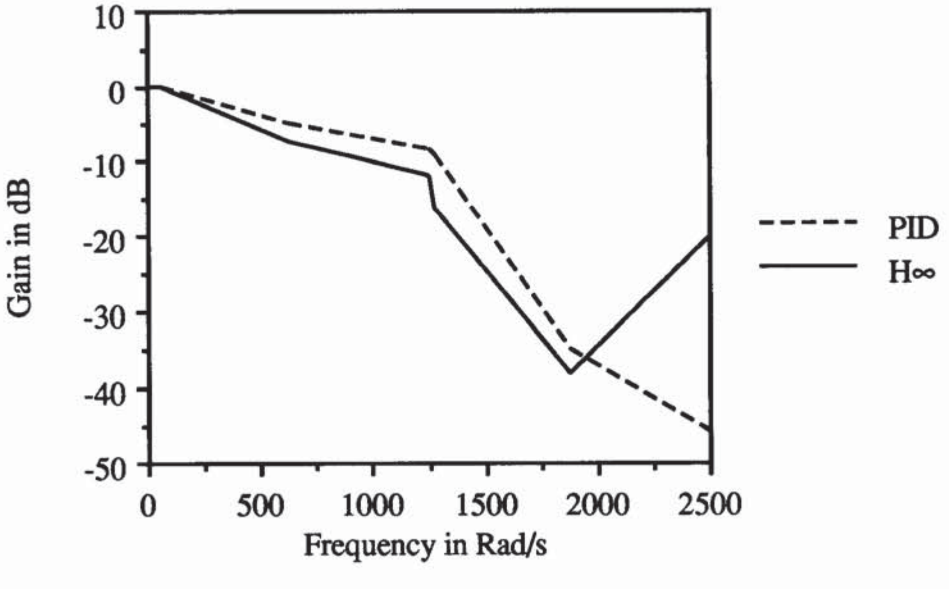


Figure 4.8a Closed-Loop Gains For H∞ and PID Controlled Systems.

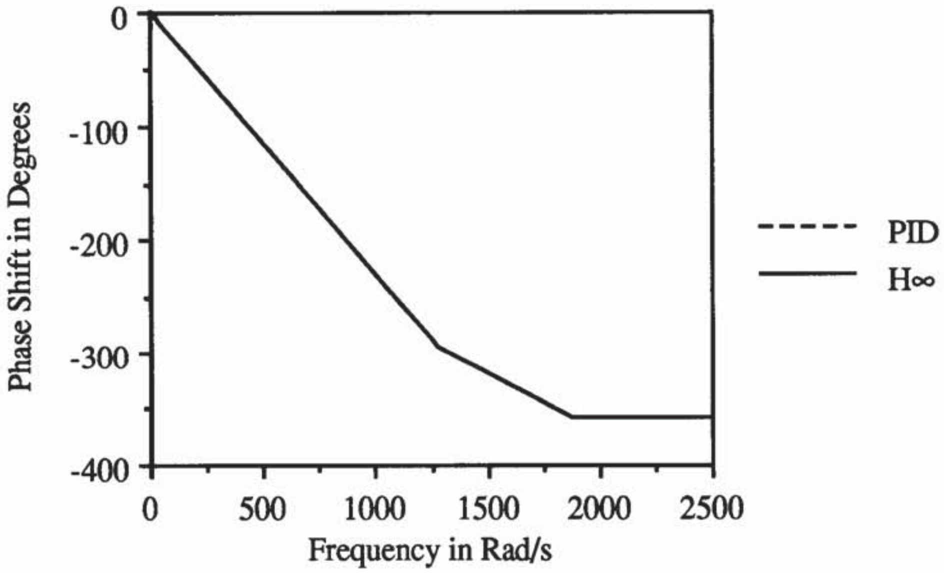


Figure 4.8b Closed-Loop Phase Shifts For H∞ and PID Controlled Systems.

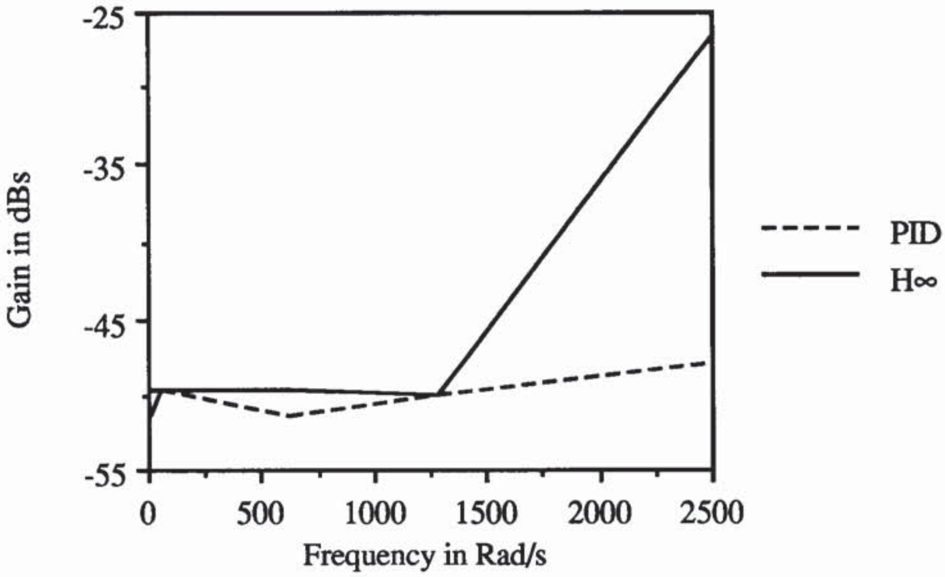


Figure 4.9a H∞ and PID Controller Gains.

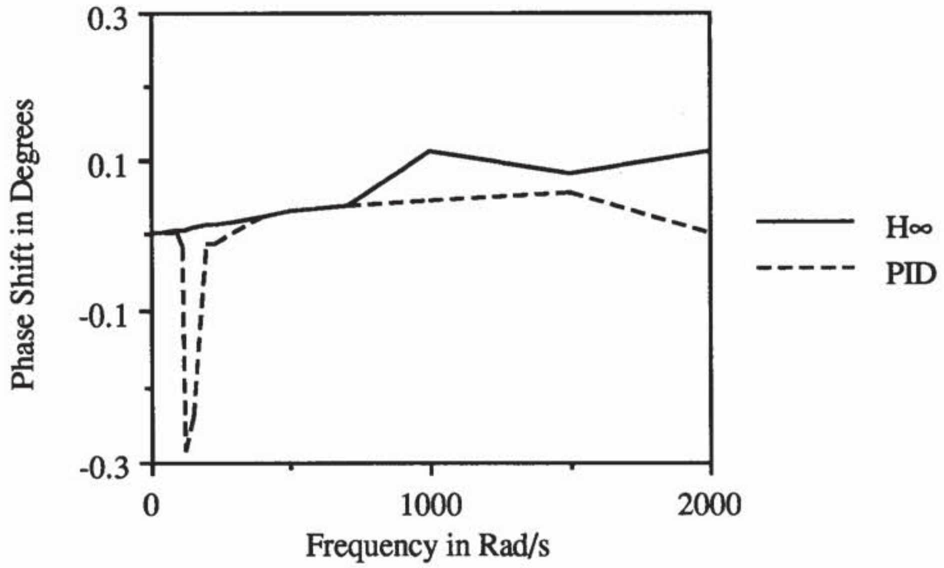


Figure 4.9b H∞ and PID Controller Phase Shifts.

Note that the H∞ controller displays a slight phase advance at higher frequencies (the controller acting as a lead compensator), while the PID controller displays a slight phase delay at low frequencies and phase advance at higher frequencies (the controller is acting as a lead-lag compensator). If the integral wind-up protection in the PID controller is removed a 90° phase shift occurs at low frequencies (as predicted theoretically) and the low frequency gain controller is 30dB higher (i.e. the closed-loop system would have stability problems).

An example of the steady state response of the controllers is shown in Figure 4.9c, with the solid line corresponding to the controller input (error) signal (frequency 150 Rad/s), the dotted line corresponding to the H∞ controller output and the dashed line corresponding to the PID controller output. The phase advance of the H∞ controller output can be seen, with a peak at 0.0518383 seconds, while the controller input (error) signal has a peak at 0.0525249 seconds and the PID controller has a peak at 0.0532115. Note for ease of comparison the H∞ controller signal was multiplied by 455 and the PID controller signal was multiplied by 341.

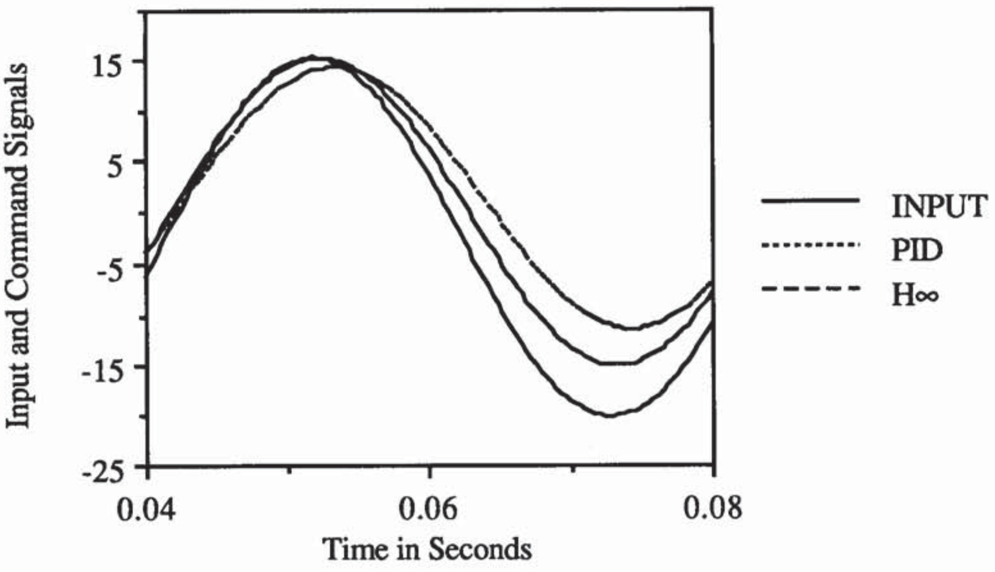


Figure 4.9c Example of Input and Controller Signals

4.2.7.2 Analysis of Controller Structure.

Analysis of the digital PID (in time series format) shows it has the structure

$$u(t) =(P+I+D)e(t) - (2D+P)e(t-1) + De(t-2) + u(t-1) \tag{4.10}$$

while the H∞ controller has the form given in (4.9).

It can be seen that the structure of the H∞ controller is more flexible, in that the digital H∞ controller has variable poles (the u(t) terms), whereas the PID controller has a static controller pole, at z = 1. This is believed to be the reason for the better performance achieved in the design example by the H∞ controlled system, the fact that its frequency response can be "shaped". If the integral component of the PID controller is removed a cancellation occurs removing the time series PID controller pole (the u(t-1) term). Thus the integral controller component causes a controller pole on the unit circle in the z-domain, equivalent to a pole on the imaginary axis in the s-domain, which could cause stability problems and limits the value of P and D that can be used (A limit is placed on the contribution of the integral term to the control signal to stop the phenomena known as integral wind-up, where the integral contribution makes the system unstable).

This analysis also shows how a PID controller can be calculated using fewer mathematical operations. The standard PID controller requires 7 mathematical operations while the time series version (Equation 4.10) requires 6 (if the P+I+D etc., summation are precalculated). This may be of use in systems with limited computational abilities.

4.3 PRACTICAL DESIGN EXAMPLE.

4.3.1 Design Example Details.

To verify the results produced in simulation a comparative study was undertaken on the Drives Test Facility at Aston University. The test rig comprised an Electro-Craft™ BRU200 servo system, consisting of a DM25 drive module and a S-4075 motor (rotor inertia 6.8×10^{-4} kg m^2 and with an 8000 pulse per revolution optical encoder). The motor was coupled to a load inertia of 3.473×10^{-3} kgm^2 via a shaft with torsional stiffness of 523.6 Nm/rad which produced a system with a 172 Hz mechanical resonance. The mechanical resonance limits the allowable gains in the internal velocity loop and the outer positional control loop effectively limiting the achievable system bandwidth and hence performance. The ratio of motor inertia to load inertia is 5.1:1 which is at the ratio limit for a practical system, but was deemed allowable since this was a comparative study. The system has a static friction of 0.1 Nm and viscous damping of 1.33×10^{-3} Nm/rads. The update time of both the outer positional loop and the internal servo system velocity control-loop was 1 ms. The performance specification conceived for this design example was similar to that used in the simulated example. The load was required to be indexed through 180° in 50 ms holding its final position to within ±5 encoder pulses (equivalent to ±0.225°) for a further 100 ms. Note this is a more onerous control problem than the simulated design example since the load inertia is greater (compared to the rotor inertia), the static friction and viscous damping are lower (meaning that the system will be less stable), the mechanical resonance is at a lower frequency (effectively putting a greater restriction on the achievable system bandwidth) and the controller update time is lower (reducing the achievable system bandwidth).

4.3.2 Robustness Specification and Nominal Model.

A nominal model was determined by feeding a random noise signal into the BRU200 via the Themis card (using software developed by the author) and recording the output position of the system. Least squares parameter estimation was used to determine a digital nominal model which was converted, using a bilinear transform, to

$$G_o = \frac{104490}{s^2 + 112.31s + 524.78} \qquad (4.11)$$

The frequency response of the system was determined using a swept-sine test and the multiplicative robustness weighting function was determined as

$$W_2 = \frac{s^2 + 1.5x10^5\ s + 1.9x10^{10}}{2x10^{10}} \qquad (4.12)$$

which corresponds to a guaranteed nominal gain margin of 5.9 dB and a phase margin of 58°.

4.3.3 Performance Weighting Function.

The performance weighting function was determined using method described by Francis [1990] (as in the simulated design example). A number of iterations were required to determine the performance weighting function

$$W_1 = \frac{0.99\ s + 356.1}{s + 23.8} \qquad (4.13)$$

The controller was determined and a pole/zero cancellation removed. The controller was converted to the digital time series

$$u(t) = c1\ e(t) + c2\ e(t\text{-}1) + d1\ u(t\text{-}1) \qquad (4.14)$$

where u(t) is the controller output.

e(t) is the positional error.

c1, c2 and d1 are the controller coefficients (respectively 0.92089, -0.89844 and 0.97656).

4.3.4 Actual Results and Simple Robustness Test.

The results for the system controlled by the two controllers can be seen in Figure 4.11 (for the sake of clarity only the crucial 100 ms holding errors are shown). The system controlled with a PID with VFF position controller met the specification in 81 ms shown in Figure 4.10 (dotted line). An extensive period of tuning was completed on the PID with VFF coefficients to ensure that they were optimal, in terms of time to reach the required holding error. The H∞ controlled system produced the results shown in Figure 4.10 (solid line) with the specification met in 50 ms. If the original specification was used, a holding error of ±0.1°, the H∞ system met the specification in 100 ms, while PID with VFF controlled system met the specification in 105 ms.

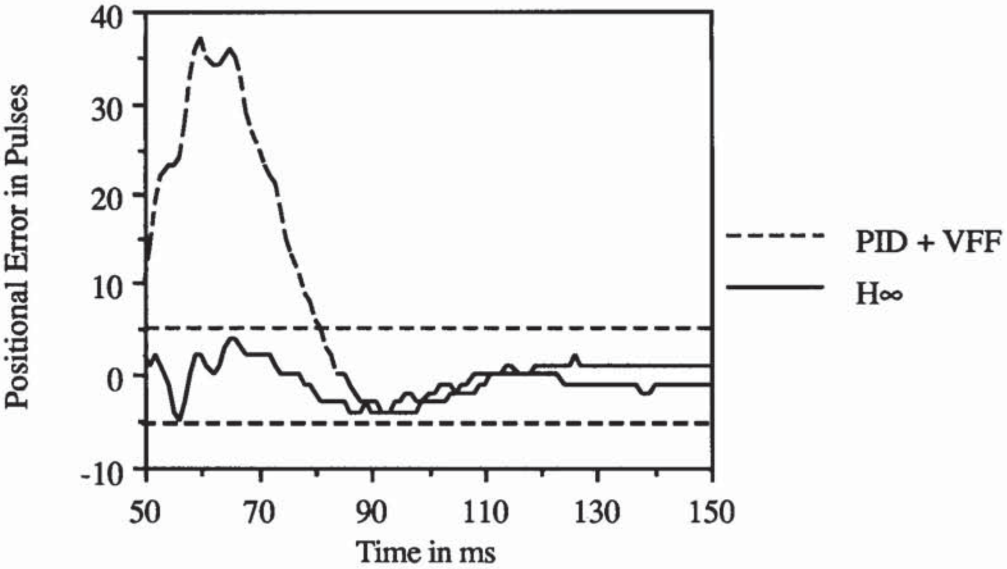


Figure 4.10 Holding Errors For H∞ and PID with VFF Controlled System.

In order to test the robustness of the H∞ controlled system the load disk was removed, which has an equivalent effect to that of backlash (i.e. decoupling the motor and load) on the system (see Chapter 3, Section 3.2.6). It can be seen, Figure 4.11, that the H∞ controller (solid line) achieved the performance specification in 92 ms while the traditional PID with VFF controller (dotted line) achieved the specification in 100 ms.

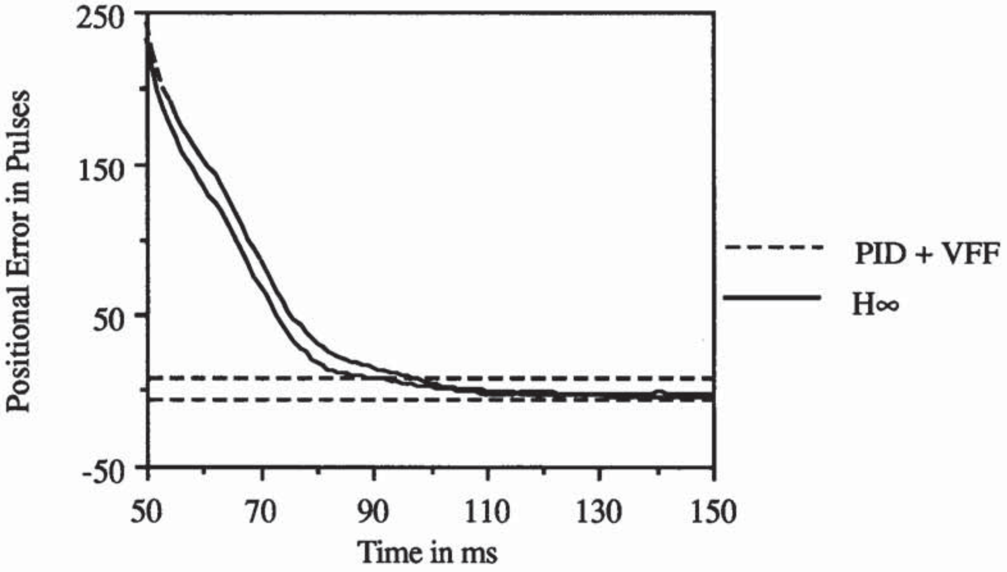


Figure 4.11 Holding Errors For Reduced Inertia System.

4.3.5 Analysis of Two Controllers.

The H∞ controller determined in this example is simpler than the traditional PID with VFF controller. The H∞ controller (of form 4.14) requires a total of 5 mathematical operations while the PID with VFF requires a total of 10 mathematical operations. A reduction in the computational requirements can reduce the calculation time of the outer positional control-loop which would reduce sampling delays, increasing the maximum possible system bandwidth and improving the system performance (see Chapter 3). The analysis of the open-loop, closed-loop and controller frequency response was repeated. The H∞ and PID controllers showed the same features as the simulated design example with the H∞ controller acting as a lead controller with a higher high frequency gain and a lower low frequency gain than the PID controller.

4.3.6 Repeatability Tests.

In order to assess the repeatability of the system when controlled by either controller the error analysis was completed for four additional cycles. This was completed since the real system shows variations between cycles which are not demonstrated by the simulated system. The H∞ controlled system showed less variation between cycles and always produced a system which meet the specification in less time than the PID with VFF controlled system (The H∞ controlled system met the specification in 53ms, 72ms, 68ms and 62ms). The PID with VFF controlled system failed to meet the specification for three cycles and meet the specification in 85ms on the final cycle. The variation between cycles was believed to be due to the inherent variation in the dc servo systems between each cycle and the fact that the timing of the two control loops is not synchronised.

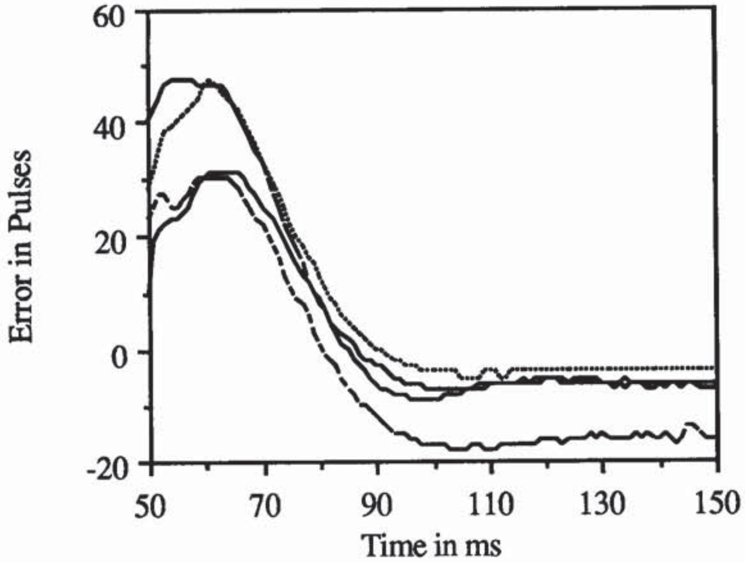


Figure 4.12a Repeatability Test on PID with VFF Controlled System.

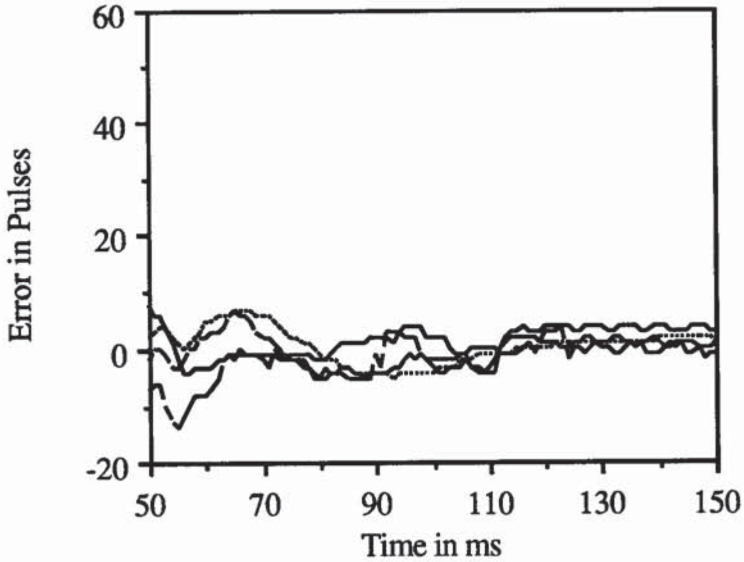


Figure 4.12b Repeatability of H∞ Controlled System.

4.4 CONCLUSIONS

This chapter has illustrated the following points, albeit for a single (type of) design example:-

1) It has shown the validity of using $H\infty$ controller synthesis for high-speed independent drive system.

2) An analysis of the differences in structure between the $H\infty$ controller and the traditional PID controller has been detailed.

3) A $H\infty$ controller produced a solution to an industrial design example which could not be solved using a traditional PID with VFF controller (with the $H\infty$ controller always producing better results than the PID with VFF controlled system, although the level of variation between cycles was unacceptable).

4) The validity of using a second order system model for a servo system, identified using least squares parameter estimation has been demonstrated.

5) The correlation between the simulation used and an actual system has been further demonstrated.

Points 1, 2, 3 and 4 have not previously been demonstrated, to the authors knowledge. The results presented in this chapter have been published in the I.Mech.E Journal of Systems and Control Engineering [Beaven *et al.* 1994a].

CHAPTER 5

SIMPLIFICATION OF H∞ CONTROLLER SYNTHESIS AND RECURSIVE LEAST SQUARES PARAMETER ESTIMATION ALGORITHMS.

5.1 INTRODUCTION

The research detailed in Chapter 4 demonstrated that if a nominal plant model and weighting functions of the form

$$G_o(s) = \frac{\alpha}{s^2 + \beta s + \delta} \; ; \; W_1 = \frac{\varepsilon s + \theta}{s + \mu} \; ; \; W_2 = \frac{s^2 + \psi s + \Omega}{\rho} \qquad (5.1)$$

were used then H∞ controller synthesis could be applied to single axis high-speed independent drives. To extend this research a detailed study of state-space H∞ synthesis controller algorithm of Doyle *et al.* [1989] with the simplification detailed by Safonov *et al.* [1989] was undertaken. The study consisted of analysing and simplifying the controller synthesis and was completed for two reasons:-

1) To gain an understanding of the relationship between the weighting functions, plant and actual system performance. This would allow simple guidelines to be produced for weighting functions selection.

2) If the algorithm can be reduced to a sufficiently simple enough form, it could be used as the controller modifier in a self-tuning regulator (STR), a form of adaptive controller.

The simplifications are detailed, following the steps described in Chapter 2. They are briefly outlined to allow the reproduction of the results but detailed analysis is not given since many of the files generated symbolically, in particular those concerned with the eigenvector and eigenvalue determination, are too extensive to be included (i.e. of the order of megabytes of hard disk space). The simplifications were completed using the symbolic manipulators MapleV [Char *et al.* 1991] and Mathematica [Wolfram 1991]

5.2 SIMPLIFICATION OF H∞ CONTROLLER SYNTHESIS ALGORITHM.

5.2.1 Initial Set-Up of Augmented State-Space Plant.

If the nominal plant and weighting functions are of the form shown in (5.1) and the augmented plant state-space representation is

$$P(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \tag{5.2}$$

its elements are given by

$$A = \begin{bmatrix} -\beta & -\delta & 0 \\ 1 & 0 & 0 \\ 0 & -\alpha & -\mu \end{bmatrix} \; ; \; B_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \; ; \; B_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 0 & -\varepsilon\alpha & \theta - \varepsilon\alpha \\ \frac{(\psi-\beta)\alpha}{\rho} & \frac{(\Omega-\delta)\alpha}{\rho} & 0 \end{bmatrix} \; ; \; C_2 = [\, 0 \; -\alpha \; 0 \,] \tag{5.3}$$

$$D_{11} = \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix} \; ; \; D_{12} = \begin{bmatrix} 0 \\ \frac{\alpha}{\rho} \end{bmatrix} \; ; \; D_{21} = [1] \; ; \; D_{22} = [0]$$

It can be clearly seen that the augmented plant representation is almost in the format required for the simplified controller determination of Doyle *et al.* [1989]. This form shows that the robustness weighting function, for the above form of plant and weighting functions, cannot be set to zero since D_{12} would be of the incorrect form (i.e. it would be a zero matrix). Additionally ε should be less than 1, for this form of plant and weighting functions, since ε is the maximum singular value of D_{11} which must be less than γ (which in this case is unity).

5.2.2 Zeroing D_{22} and Scaling D_{12} and D_{21}.

The D_{22} element does not requires zeroing since it is already zero. The scaling of D_{12} and D_{21} transforms the augmented state-space representation to

$$A=\begin{bmatrix} -\beta & -\delta & 0 \\ 1 & 0 & 0 \\ 0 & -\alpha & -\mu \end{bmatrix} ; \; B_1=\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} ; \; B_2=\begin{bmatrix} \frac{\rho}{\alpha} \\ 0 \\ 0 \end{bmatrix}$$

$$C_1=\begin{bmatrix} 0 & \varepsilon\alpha & \varepsilon\alpha-\theta \\ \frac{-(\psi-\beta)\alpha}{\rho} & \frac{-(\Omega-\delta)\alpha}{\rho} & 0 \end{bmatrix} ; \; C_2=[\,0 \;\; -\alpha \;\; 0\,] \qquad (5.4)$$

$$D_{11}=\begin{bmatrix} -\varepsilon \\ 0 \end{bmatrix} ; \; D_{12}=\begin{bmatrix} 0 \\ 1 \end{bmatrix} ; \; D_{21}=[1] \; ; \; D_{22}=[0]$$

5.2.3 Zeroing of D11 Element Via Loop-Shifting.

In this case the K_∞ value, selected to minimise the greatest singular value of the D_{11}, is zero. Therefore the maximum singular value of D_{11} is ε which is therefore required to be less than unity (in this case). The zeroing of D_{11} produces the following elements

$$A = \begin{bmatrix} -\beta & -\delta & 0 \\ 1 & 0 & 0 \\ 0 & \frac{-\alpha}{1-\varepsilon^2} & \frac{\mu+\theta}{\varepsilon^2-1} \end{bmatrix} ; \; B_1 = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\sqrt{1-\varepsilon^2}} \end{bmatrix} ; \; B_2 = \begin{bmatrix} \frac{\rho}{\alpha} \\ 0 \\ 0 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 0 & \frac{\varepsilon\alpha}{\sqrt{1-\varepsilon^2}} & \frac{\varepsilon\alpha-\theta}{\sqrt{1-\varepsilon^2}} \\ \frac{-(\psi-\beta)\alpha}{\rho} & \frac{-(\Omega-\delta)\alpha}{\rho} & 0 \end{bmatrix} ; \; C_2 = \begin{bmatrix} 0 & \frac{-\alpha}{1-\varepsilon^2} & \frac{\theta-\varepsilon^2\mu}{1-\varepsilon^2} \end{bmatrix} \quad (5.5)$$

$$D_{11} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} ; \; D_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} ; \; D_{21} = \begin{bmatrix} \frac{1}{\sqrt{1-\varepsilon^2}} \end{bmatrix} ; \; D_{22} = [0]$$

5.2.4 Re-zeroing the D_{22} term and Rescaling D_{12} and D_{21}.

Again the D_{22} term is zero and therefore does require transformation. The scaling of D_{12} and D_{21} produces the following elements

$$A = \begin{bmatrix} -\beta & -\delta & 0 \\ 1 & 0 & 0 \\ 0 & \frac{-\alpha}{1-\varepsilon^2} & \frac{\mu+\theta}{\varepsilon^2-1} \end{bmatrix} ; \; B_1 = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{\sqrt{1-\varepsilon^2}} \end{bmatrix} ; \; B_2 = \begin{bmatrix} \frac{\rho}{\alpha} \\ 0 \\ 0 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 0 & \frac{\varepsilon\alpha}{\sqrt{1-\varepsilon^2}} & \frac{\varepsilon\alpha-\theta}{\sqrt{1-\varepsilon^2}} \\ \frac{-(\psi-\beta)\alpha}{\rho} & \frac{-(\Omega-\delta)\alpha}{\rho} & 0 \end{bmatrix} ; \; C_2 = \begin{bmatrix} 0 & \frac{-\alpha}{\sqrt{1-\varepsilon^2}} & \frac{\theta-\varepsilon^2\mu}{\sqrt{1-\varepsilon^2}} \end{bmatrix} \quad (5.6)$$

$$D_{11} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \; ; \; D_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \; ; \; D_{21} = [1] \; ; \; D_{22} = [0]$$

5.2.5 Formation of the Hamiltonian Matrices.

Now the Hamiltonian Matrices L_∞ and Z_∞ are formed

$$L_\infty = \begin{bmatrix} -\psi & -\Omega & 0 & \frac{-\rho^2}{\alpha^2} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\alpha}{\varepsilon^2-1} & \frac{\mu-\varepsilon\theta}{\varepsilon^2-1} & 0 & 0 & \frac{1}{1-\varepsilon^2} \\ 0 & 0 & 0 & \psi & -1 & 0 \\ 0 & \frac{\varepsilon^2\alpha^2}{1-\varepsilon^2} & \frac{\varepsilon\alpha(\varepsilon\mu-\theta)}{1-\varepsilon^2} & \Omega & 0 & \frac{-\alpha}{\varepsilon^2-1} \\ 0 & \frac{\varepsilon\alpha(\varepsilon\mu-\theta)}{1-\varepsilon^2} & \frac{(\varepsilon\mu-\theta)^2}{\varepsilon^2-1} & 0 & 0 & \frac{\varepsilon\theta-\mu}{\varepsilon^2-1} \end{bmatrix}$$

$$Z_\infty = \begin{bmatrix} -\beta & 1 & 0 & \frac{(\psi-\beta)^2\alpha^2}{\rho^2} & \frac{(\psi-\beta)\alpha^2(\Omega-\delta)}{\rho^2} & 0 \\ \delta & 0 & 0 & \frac{(\psi-\beta)\alpha^2(\Omega-\delta)}{\rho^2} & \frac{\alpha^2(\delta^2-2\delta\Omega+\Omega^2-\rho^2)}{\rho^2} & 0 \\ 0 & 0 & -\mu & 0 & 0 & (\theta-\varepsilon\mu)^2 \\ 0 & 0 & 0 & \beta & \delta & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \qquad (5.7)$$

5.2.6 Determination of Eigenvalues and Eigenvectors For Riccati Equation Solution.

The required Riccati solution, X_∞ and Y_∞, can be determined using the eigenvectors of the Hamiltonian matrices, L_∞ and Z_∞, as discussed in Chapter 2. The eigenvalues of L_∞ can be determined symbolically using

$$\det(L_\infty - \lambda I) = 0 \tag{5.8}$$

The resulting characteristic equation has the form

$$Ew^6 + Fw^4 + Gw^2 + I = 0 \tag{5.9}$$

which can, by substitution, be given as

$$x^3 + rx^2 + sx + t = 0 \tag{5.10}$$

This equation can be put into reduced form [example in Gellert 1977] using the substitutions

$$y^3 + py + q = 0 \quad \text{where } p = s - \frac{r^2}{s} \; ; \; q = \frac{2r^3}{27} - \frac{rs}{3} + t \tag{5.11}$$

The eigenvalues (roots) x_1, x_2 and x_3 can then be determined by

$$r = \sqrt{\frac{-p^3}{27}} \; ; \; \theta = \cos^{-1}\left(\frac{3q}{2\sqrt{\frac{p^3}{3}}}\right)$$

$$x_1 = 2\sqrt{\frac{a}{3}}\cos(\theta) - \frac{p}{3} \; ; \; x_2 = 2\sqrt{\frac{a}{3}}\cos(\theta + \frac{2\Pi}{3}) - \frac{p}{3} \; ; \; x_3 = 2\sqrt{\frac{a}{3}}\cos(\theta + \frac{4\Pi}{3}) - \frac{p}{3} \tag{5.12}$$

Once the eigenvalues of L_∞ are known, the corresponding eigenvectors follow relatively simply and hence X_∞ can be derived (as shown in Section 2.6.3).

The second Riccati solution, Y_∞, is the zero matrix if the plant is stable [Hvostov 1990]. This greatly simplifies the problem since only one Riccati equation needs to be solved.

5.2.7 Re-Transformation of Controller Elements.

The steps in sections 5.2.2, 5.2.3 and 5.2.4 are reversed to retrieve the controller. The controller is given by

$$K(s) = \frac{-X_{\infty 13}\,\rho^2(s^2 + \beta s + \delta)}{(\mu+s)(\alpha^2 s^2 + (\psi\alpha^2 + X_{\infty 11}\,\rho^2)s + (\Omega\alpha^2 + X_{\infty 12}\,\rho^2)} \qquad (5.13)$$

where $X_{\infty 11}$, $X_{\infty 12}$, $X_{\infty 13}$ are elements of the Riccati Equation solution, $X\infty$.

The simplified H∞ controller synthesis algorithm required 271 floating point operations against the standard Matlab function which required 13932 floating point operations for a standard plant and set of weighting functions (measured using the Matlab function flops). The number of floating point operations for the simplified algorithm can be reduced further if the weighting function parameters are kept constant.

5.3 LEAST SQUARES PARAMETER ESTIMATION.

5.3.1 Introduction.

For a self-tuning regulator to be a viable for a high speed system, where cycle times and hence parameter variations are measured in 10's of milliseconds, a parameter identifier is required which is quick and accurate to within certain limits. A review completed on parameter estimation techniques [Wilkes 1995] suggested that recursive least squares parameter estimation with a variable forgetting factor was the most applicable technique.

A brief review of the recursive least squares parameter estimation with variable forgetting technique is given in Section 2.7 of this thesis. A detailed description of can be found in Wilkes [1995].

5.3.1 Simplification of Recursive Least-Squares Parameter Estimator with Variable Forgetting Factor.

A series of simplifications, using a substitution, were completed on the recursive least squares parameter estimator with variable forgetting factor. The simplifications reduced the number floating point operations from 175, for the standard algorithm, to 58 for the simplified algorithm.

The simplifications are made using the substitution of a constant for $CM(t)x_t$.(see Chapter 2) and the fact that the covariance matrix CM is symmetrical.

5.4 CONCLUSION.

In this chapter a simplified H∞ controller synthesis algorithm has been developed which requires 2% of the computations required by the standard Matlab H∞ controller synthesis routine (calculated using the Matlab function flops). The simplified H∞ controller synthesis algorithm will allow two different strands of research to be completed. Firstly the selection of weighting functions, discussed in Chapter 6. Secondly the application of a self-tuning regulator to high speed independent drive systems, discussed in Chapter 7. The simplified H∞ controller synthesis algorithm additionally has the advantage that the reduced number of mathematical operations required will reduce any inaccuracies in the controller determination algorithm due to round-off errors. The simplification used to reduce the complexity of the recursive least squares parameter estimator with variable forgetting factor (developed by Wilkes [1995]) has been given.

CHAPTER 6

WEIGHTING FUNCTION SELECTION

6.1 INTRODUCTION.

Considerable progress has been made in the synthesis of H∞ controllers since the original work of Zames [1981]. The technique has been applied to numerous design examples, both theoretical and practical [Postlethwaite *et al.* 1986, Limbeer and Kasenally 1986]. Two broad criticisms, each consisting of many individual aspects, are levelled against the technique. Firstly, although simple guidelines exist for the selection of weighting functions, no objective set of criteria is available. Secondly, the relationship between the frequency-dependent weighting functions and the actual closed-loop time response is difficult to establish.

This chapter discusses the initial selection of weighting functions and adjustments of the performance weighting function parameters to achieve specific closed-loop performance. The discussion is limited to a single input single output second order system (Chapter 4 demonstrated that the use of such a model for the design of a H∞ controller for servo system applications was satisfactory). Many of the points made are demonstrated for dc servo systems but are considered valid for general control applications.

This chapter extends previous research completed on weighting selection in two respects. Firstly, it uses a direct-form of the controller (determined in Chapter 5), albeit for a simple second order system, to enhance the understanding and analysis of the initial weighting function selection. Secondly, suggestions are made for the adjustment of the performance weighting function, after the controller has been implemented, to achieve the desired closed-loop performance. The intuitively simple approach of adjusting the performance weighting function to obtain the required closed-loop performance is completed, instead of adjusting the nominal model or robustness weighting function, although adjustment of the nominal model or robustness weighting function may be required in certain circumstances. The theory is illustrated by application of the techniques to a dc servo system.

This chapter also discusses a computer program, written in 'C', to determine optimal weighting function parameters. Details are briefly given (since it is outside the general scope of this thesis) of a three term time series controller developed as a software test for the practical part of this thesis.

6.1.1 Problems in Weighting Function Selection.

Lundström *et al.* [1991] discussed two problems in weighting function selection:

1) In many real applications the performance specification is not detailed before the design starts, i.e. the best possible performance is required.
2) Alternative methods exist for weighting function selection since the problem can be formulated using different methods (for example using different uncertainty models) with several physical interpretations of the $H\infty$-norm (i.e. a classical frequency transfer function bound or an induced norm on input / output power spectrum).

Additional difficulties in weighting function selection include:

1) Which criterion is to be used to select the optimal controller to be selected, since different combinations of weighting functions produce different "optimal" controllers.

2) The controller synthesis ensures that the performance specification is achieved for the nominal model although the actual closed loop performance may be unacceptable (i.e. the tracking error maybe too large). Guidelines on the adjustment of the weighting function parameters in such a situation are unavailable (to the authors knowledge) .

3) Non-linearities are difficult to include in the design process.

4) Weighting functions are difficult to select for mixed demand profiles, such as the step and dwell profiles commonly used in the process industry (i.e. multi-frequency demand profiles).

5) Many practical systems require controllers with limited complexity (due to the limited time/computing facilities available). In general an increase in the update time of a process leads to a decrease in performance [Morari and Zafiriou 1989].

6) Limitations due to sampling effects, both inherent to the system and controller, are difficult to include in the design process.

7) Undefined time delays, both in the implementation of the controller and inherent in the system, are difficult to include in the design process.

8) Controllers are usually implemented digitally; the effects of digitisation (i.e. time delays and numerical round-off) are difficult to incorporate into the design process.

9) The effects of controller reduction, which is often required to produce implementable controllers, is difficult to quantify and hence difficult to incorporate into the design process.

The difficulties in weighting function selection (1)-(9) mean that not only is initial weighting function selection an onerous task but also, almost certainly, some tuning of the weighting functions will be required to obtain the desired closed-loop performance.

6.2 DIRECT-FORM CONTROLLER FOR WEIGHTING FUNCTION SELECTION.

The direct-form of the controller, if it exists, is given by

$$K(s) = \frac{-(s^2 + \beta s + \delta)\ \rho^2 X\infty_{13}}{(\mu+s)(\alpha^2 s^2 + (\rho^2 X\infty_{11} + \psi\alpha^2)s + \alpha^2\Omega + \rho^2 X\infty_{12})} \tag{6.1}$$

where $X\infty_{11}$, $X\infty_{12}$ and $X\infty_{13}$ are elements of the Riccati equation solution $X\infty$ (as detailed in Chapter 5).

The direct-form controller is used to determine two closed-loop system forms. One of these is for the initial determination of the weighting functions. The second is for alteration of the weighting functions, after the controller has been implemented, to achieve the specified closed-loop performance, if this is possible.

If the actual plant and identified model are equivalent, the closed loop system transfer function is

$$T = \frac{CL1}{CL2\ s^3 + CL3\ s^2 + CL4\ s + CL5} \tag{6.2}$$

where $CL1 = \rho^2\ \alpha X\infty_{13}$, $CL2 = -\alpha^2$, $CL3 = (-\psi\alpha^2 - \alpha^2\mu - \rho^2 X\infty_{11})$

$CL4 = (-\psi\alpha^2\mu - \alpha^2\Omega - \rho^2\mu X\infty_{11} - \rho^2 X\infty_{12})$

and $CL5 = (-\alpha^2\mu\Omega - \rho^2\mu X\infty_{12} + \rho^2\alpha X\infty_{13})$

If the actual plant is

$$\frac{A}{s^2 + Bs + C} \tag{6.3}$$

The closed loop system response T(s) is

$$\frac{CL1Rs^2 + CL2Rs + CL3R}{CL4Rs^5 + CL5Rs^4 + CL6Rs^3 + CL7Rs^2 + CL8Rs + CL9R} \qquad (6.4)$$

where $CL1R = X_{\infty 13}\rho^2 A$; $CL2R = X_{\infty 13}\rho^2 A\beta$; $CL3R = X_{\infty 13}\rho^2 A\delta$

$CL4R = -\alpha^2$; $CL5R = -\alpha^2(B + \psi + \mu) - X_{\infty 11}\rho^2$

$CL6R = -\alpha^2(C + B\psi + B\mu + \psi\mu + \Omega) - X_{\infty 11}\rho^2(B + \mu) - X_{\infty 12}\rho^2$

$CL7R = -\alpha^2(C(\psi+\mu) + B(\psi\mu+\Omega) + \mu\Omega) - X_{\infty 11}\rho^2(C+B) - X_{\infty 12}\rho^2(B+\mu) + AX_{\infty 13}\rho^2$

$CL8R = -\alpha^2(C\mu\psi+C\Omega) + B(\mu\Omega+\Omega)) - X_{\infty 11}\rho^2 C\mu - X_{\infty 12}\rho^2(C+B\mu) + AX_{\infty 13}\rho^2$

$CL9R = -C\mu(\Omega\alpha^2 + \rho^2 X_{\infty 12}) + A\delta X_{\infty 13}\rho^2$

Note that the analysis can be extended to a use a more complex physical system (i.e. a third or higher order system), but for the sake of clarity only a simple second order model is used.

6.3 NOMINAL MODEL.

The nominal model can be determined experimentally using a swept sine (or step response) test or a more complex identification method such as least squares parameter estimation. Alternatively the nominal model can be constructed theoretically using readily available information such as torque constants, rotor inertia, etc. It has been shown, for dc servo systems [Beaven *et al.* 1994a], that an adequate model for H∞ controller synthesis can be determined using least squares parameter estimation or alternative model estimation techniques, such as a swept sine test. The nominal model should be restricted to its simplest form since the more complex the nominal model, the more complex the controller [Chiang and Safonov 1992].

6.4 ROBUSTNESS WEIGHTING FUNCTION DETERMINATION.

The robustness weighting function is constructed to allow for two types of disparity between the nominal model and actual system. Firstly, that due to inaccuracies in the nominal model, such as unmodelled dynamics. Secondly, disparity due to changes in the dynamics of the system. Both elements of the robustness specification can be estimated by a comparison of the actual and theoretical response of the system over a period of time.

An analysis of the limitations placed on the controller, by the necessity for the Riccati equation solution, X_∞, to be positive semi-definite, shows that the 0dB crossover frequency of W_2^{-1} constrains the 0dB crossover frequency achievable by W_1, in effect limiting the performance specification (this was noted by Chiang and Safonov [1992]). Therefore the inverse of the robustness weighting function, W_2^{-1}, should have as high a 0dB crossover frequency as possible. A second argument for making W_2^{-1} a high bandwidth transfer function is that the robustness weighting function zeros are transformed, with some modification, to the poles of the controller. Therefore high frequency W_2^{-1} poles produce high frequency controller poles which may be removed in limited bandwidth applications (if the poles are outside the maximum system bandwidth) and hence reduce the controller complexity.

A number of simple rules are available for the determination of the robustness weighting function:-

1) An improper robustness weighting function, W_2, should be used such that GW_2 is proper (since GW_2 is used in the controller synthesis), in order to limit the controller complexity.

2) The estimation of W_2 can be made by comparison of the response of the nominal model and actual system using a number of different frequency input signals, using Equation 2.8, Section 2.2, Chapter 2.

3) The robustness weighting function should be selected such that W_2^{-1} has the highest possible 0dB crossover frequency. This will reduce any unnecessary restriction on the performance specification. Additionally it will induce high frequency controller poles which may allow a reduction in the controller complexity, by pole removal, in limited bandwidth systems.

6.5 PERFORMANCE WEIGHTING FUNCTION SELECTION.

The literature discusses three strategies for performance weighting functions selection. The first method (only briefly mentioned here since it was used in Chapter 4) uses the ideal closed-loop response, T_{id}, to determine the ideal sensitivity function, S_{id}. The performance weighting function is then determined as the inverse of the ideal sensitivity function [Francis 1990].

6.5.1 Classical Frequency Domain Specification.

Lundström *et al.* [1991] used a classical frequency domain specification with the following criteria

1) Steady-state offset less than ϕ;
2) Closed-loop bandwidth higher than ω_B;
3) Amplification of high frequency noise less than a factor M;

Then the performance weighting function W_1 is of the form

$$W_1 = \frac{s + M\omega_B}{M (s + \phi\omega_B)} \qquad (6.5)$$

6.5.2 Alternative Frequency Domain Specification.

Piché *et al.* [1991] used a performance weighting function of the form

$$W_1 = \frac{\eta s + \lambda}{s + \xi\lambda} \tag{6.6}$$

The parameter η, which is limited to $0 \leq \eta < 1$, is the high-frequency limiting value of W_1 and serves to constrain the maximum resonance peak in the ideal sensitivity function frequency response. The parameter $\lambda > 0$ gives the 0dB crossing of W_1 (and thus of S) when ξ is small. Piché *et al.* [1991] suggested the parameter λ was closely related to the steady state error of the system. Note the form of weighting function used by Piché, (6.6), is the same as that used by Lundström, (6.5), with the weighting function constants simply specified in a different manner.

6.5.3 Direct-Transform Approach.

The performance weighting function used in this analysis has the same form as the weighting function used by Lundström *et al.* [1991] and Piché *et al.* [1991]

$$W_1 = \frac{\varepsilon s + \theta}{s + \mu} \tag{6.7}$$

An analysis of the initial augmented state-space representation (Chapter 5) shows that the value of ε should be restricted as $0 \leq \varepsilon < 1$, to satisfy the constraint that the maximum singular value of D_{11} be strictly less than γ (where γ, in this case, is unity). The value of ε should be set relatively high to constrain the controller gain so that the closed-loop system does not have an excessive overshoot. Experience has shown that a reasonable value for ε is 0.99.

The closed-loop form in (6.2), used for the initial estimate of the weighting functions, shows that the steady-state system response, for a step input (the majority of high speed independent drive motions require a rapid increment followed by a holding period, which can be approximated by a step input), is

$$\frac{CL1}{CL5} = \frac{\alpha X\infty_{13}\ \rho^2}{-\alpha\mu\Omega - \mu X\infty_{12}\ \rho^2 + \alpha X\infty_{13}\ \rho^2} \tag{6.8}$$

Thus the higher the value of μ, the greater will be the steady state error of the system. This was indicated by Piché *et al.* [1991] but the relationship between μ (λ) and the steady-state error was not quantified. An increase in the value of θ leads to an increase the numeric value of the Riccati equation solution, $X\infty$, and its elements contained in the controller (i.e. $X\infty_{11}$, $X\infty_{12}$, $X\infty 13$). This leads to an increase in the frequency of the poles in the closed-loop system and an increase in the controller gain, increasing the system bandwidth and the speed of system response [Raven 1987]. Note however that this ignores the effect of high frequency noise amplification. This concurs with classical control theory since an increase in the value of θ is equivalent to a decrease in the tracking error over a particular frequency. An examination of the Riccati equation solution, $X\infty$, (i.e. is it positive semi-definite) demonstrates that a trade-off exists between the allowable values of θ and μ, with an increase in μ leading to a possible increase in θ and vice-versa.

Assuming that the motor has been correctly selected and a nominal model and robustness weighting function have been determined, the strategy suggested for the initial selection of performance weighting function is to use (6.8) and the acceptable level of endpoint error to determine an initial value of μ. Note μ is not set to zero since this would place a controller pole on the imaginary axis limiting the system speed of response and introduce a 90^o phase shift, which could limit the system performance (see Chapter 3). The scanning program (see Section 6.7) can then be used to determine the limiting θ value.

If the closed-loop response of the system is unsatisfactory the performance weighting function parameters θ and μ should be tuned. Analysis of (6.4) yields the steady-state system response

$$\frac{X_{\infty 13}\, \rho^2 A\, \delta}{X_{\infty 13}\, \rho^2 A \delta - \mu\, (\alpha^2 \Omega + \rho^2\ X_{\infty 12})\, C} \qquad (6.9)$$

This can be used to re-determine the value of μ. That is μ is altered until the steady-state offset of the system is as required (if this is an appropriate performance measure). The higher the value of μ the greater the steady-state error (i.e. the system response is effectively shifted downwards). The θ value is altered until the optimal actual response is achieved. A decrease in θ leads to a decrease in the responsiveness of the system but also to a decrease in the amount of overshoot. Since errors will occur in the determination of the controller, due to digitisation effects for example, the performance weighting function may have to be adjusted more than once. The values of μ and θ can also be used to remove the effect of any system disturbances (i.e. mechanical resonances) simply by determining the frequency response of W_1 such that the controller gain falls off at a lower frequency than the disturbance (if possible). If the required performance cannot be achieved then the nominal model or robustness weighting function may need to be redetermined or the physical system or performance requirements altered.

6.6 DESIGN EXAMPLES

To demonstrate the determination of the performance weighing function two design examples were completed, both in simulation (details of the implementation of the second design example on a practical system are given in Chapter 7).

6.6.1 Simple Design Example.

If the plant and nominal model are determined as the transfer functions

$$\text{Actual Plant} = \frac{4070}{s^2 + 145s + 545}\ , \quad \text{Nominal Model} = \frac{4050}{s^2 + 125s + 525} \qquad (6.10)$$

The performance specification is for a step response with an endpoint error of less than 0.1% within 40 ms. A comparison of the actual and nominal frequency responses yields the robustness weighting function, W_2, given below. Using the steady-state error value for the nominal model, an initial value of μ is determined as 0.0001. The corresponding optimal θ value is 82.2. Thus the weighting functions are

$$W_1 = \frac{0.99s + 82.2}{s + 0.0001} \quad , \quad W_2 = \frac{s^2 + 9.5\text{x}10^4 s + 1.9\text{x}10^8}{2\text{x}10^8} \qquad (6.11)$$

These weighting functions were used to produce the following controller

$$\frac{6.97\text{x}10^3\, s^2 + 8.71\text{x}10^5\, s + 3.66\text{x}10^6}{s^2 + 1.07\text{x}10^5\, s + 10.72} \qquad (6.12)$$

Figure 6.1 shows that the nominal closed-loop response meets the specification in 25ms (determined assuming that the actual and identified models are equivalent). The actual system response, Figure 6.2, is out of specification since the endpoint error is 0.158%. Using (6.8) the performance weighting function parameter μ is redetermined as 0.2, effectively shifting the step response downwards. Decreasing the value of θ

did not reduce the amount of endpoint error. The corresponding optimal θ value is 83.3. The performance weighting function becomes

$$W_1 = \frac{0.99s + 83.3}{s + 2} \tag{6.13}$$

This was used to redetermine the H∞ controller

$$\frac{6.97 \times 10^3 s^2 + 8.71 \times 10^5 s + 3.66 \times 10^6}{s^2 + 1.066 \times 10^5 s + 2.13 \times 10^4} \tag{6.14}$$

When this controller is applied to the actual plant, it gives a closed-loop response which meets the specification in 40 ms, Figure 6.3.

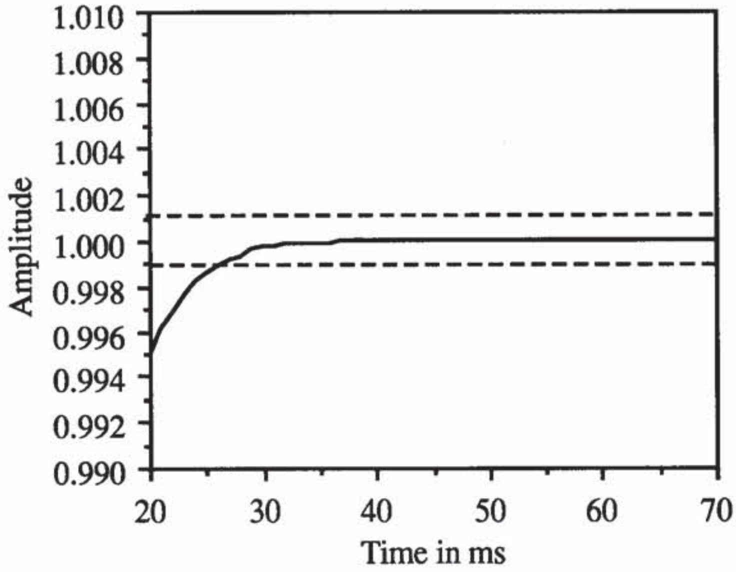


Figure 6.1 Closed-Loop Response for Nominal Model.

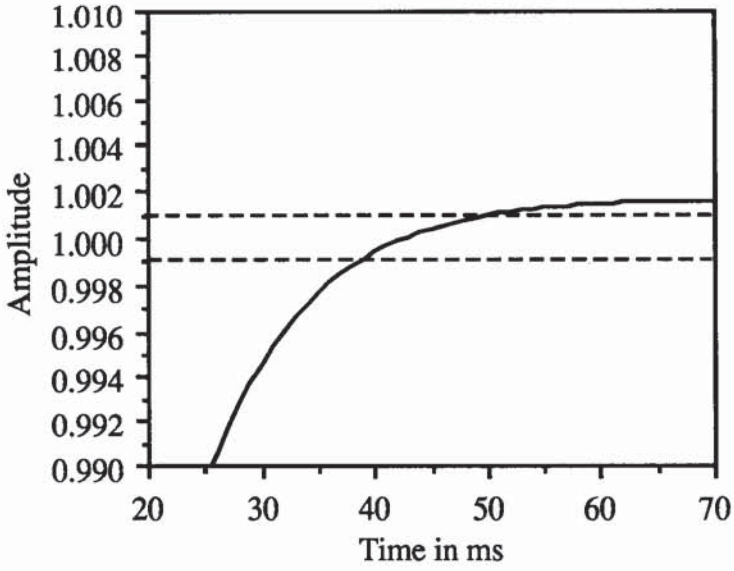


Figure 6.2 Actual Closed-Loop Response.

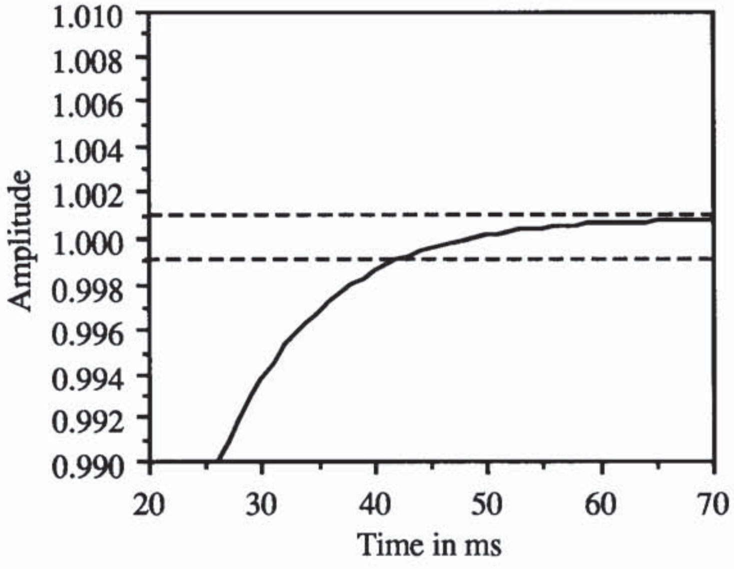


Figure 6.3 Adjusted Closed-Loop Response.

6.6.2 More Complex Design Example.

An ACSL simulation was constructed for an industrial design example (the simulation is as detailed in Chapter 3). Both the inner velocity control loop and outer position control loop have an update time of 1 ms.

The design problem is as follows: an Electro-Craft™ BRU200 DM-25 drive module controlling a S-3016 brushless dc motor (rotor polar moment of inertia 8.3x10^{-5} kgm^2) is required to rotate an axis with a total referred inertial load of 1.718x10^{-4} kgm^2 through 105° in 20 ms with an endpoint error not greater than ±1.5°. The motor has a 8000 pulse per revolution optical encoder and therefore the motion corresponds to moving through 2334 pulses in 20 ms with a maximum endpoint error of ±33 pulses.

The frequency response of the system, Figure 6.4, was determined using a swept sinewave test (solid line gain, dotted line phase). Yielding the nominal model

$$G_o(s) = \frac{4x10^7}{s^2 + 1001s + 100} \qquad (6.15)$$

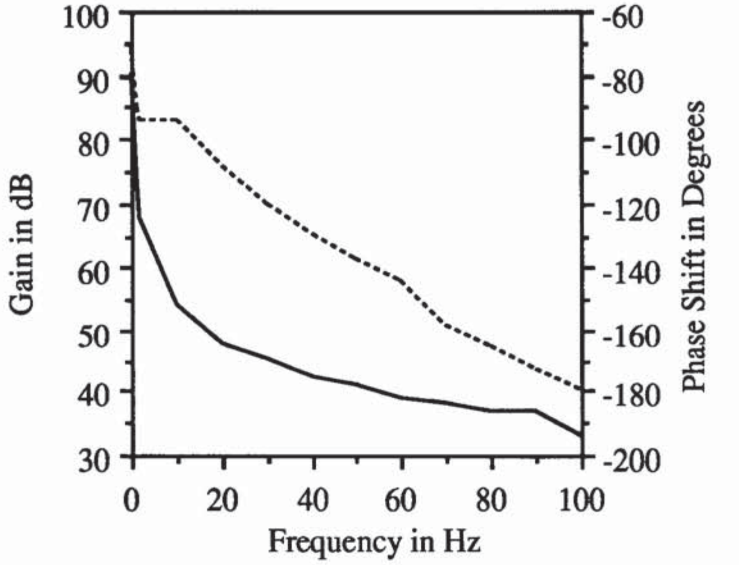


Figure 6.4 Frequency Response of Plant.

The weighting functions were determined as (using the same method as the first example)

$$W_1 = \frac{0.99s + 82.2}{s + 0.0001} \quad , \quad W_2 = \frac{s^2 + 9.5x10^4s + 1.9x10^8}{2x10^8} \qquad (6.16)$$

The controller was digitally implemented in the outer position control loop. The discrete-time H∞ controller was

$$\begin{aligned} u(t) &= 0.0194*e(t) - 0.0258*e(t-1) + 0.0065*e(t-2) \\ &\quad + 0.0366*u(t-1) + 0.9634*u(t-2) \qquad (6.17) \end{aligned}$$

with e(t) = positional error at time t

u(t) = controller signal at time t.

The system response was out of specification with an initial overshoot to 3121 pulses and a settling time to within ±33 of 2334 pulses in 48 ms, see Figure 6.5. The system response showed a positive steady-state offset therefore the μ value was increased. Additionally the initial overshoot indicates the controller gain is too high and therefore θ was reduced. Using (6.8) to re-determine the performance weighting function parameters yielded

$$W_1 = \frac{0.99s + 47}{s + 1} \qquad (6.18)$$

This produced the digital H∞ controller (after cancellation)

$$u(t) = 0.0036*e(t) - 0.003599*e(t-1) + 0.999*u(t-1) \qquad (6.19)$$

(notation as Equation 6.17)

This produced a system with response shown in Figure 6.6 (demanded position dotted line, actual position solid line), where the system settles to within ±33 of 2334 pulses in 19 ms.

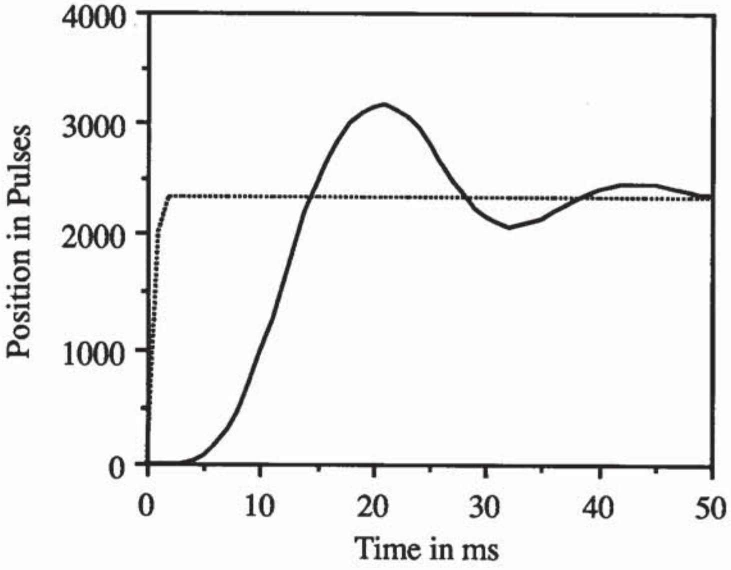


Figure 6.5 Initial Closed-Loop Response.

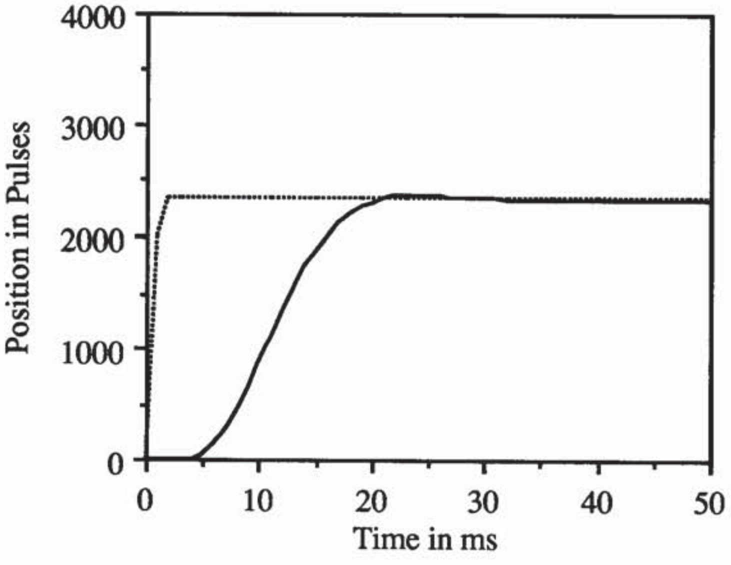


Figure 6.6 Final Closed-Loop Response.

6.6.3 Summary of Controller Determination.

The determination of a H∞ controller for this application can be broken down into a number of discrete steps.

1) Determine a nominal model using swept sinewave test or other identification method, remembering that the simpler the model the simpler the controller, which for digital control systems is of key importance.

2) Determine an improper robustness weighting function with as high a 0 dB crossover point as possible. The robustness weighting function must reflect the level of system uncertainty and/ or expected system variation and must be such that GW_2 is proper.

3) Assume that the nominal model is equivalent to the actual system dynamics and use the formula (6.8) to determine an initial value of μ (but not zero since this will introduce a open-loop phase shift of 90^o degrees into the system). The value of ε should be set at 0.99 to limit any overshoot in the closed-loop system. The optimal value of θ should be determined by use of a scanning function (such as that in Appendix C). If the closed-loop response is unsatisfactory a number of steps can be taken:-

a) If the system shows a positive steady-state offset increase μ.

b) If the system shows a negative steady-state offset decrease μ.

c) If the system response is not rapid enough increase θ. If this is not possible (as determined using the scanning function) then decreasing the value of ε will allow the value of θ to be increased.

d) If the system response shows too much overshoot, decrease θ.

6.7 WEIGHTING FUNCTION SCANNING PROGRAM.

The work completed in Chapter 5, for the simple set of plants and weighting functions described, yielded a controller which was dependent on the solution, X_∞, of a single Riccati equation being positive semi-definite. A program was developed, in 'C', to enable the determination of optimal (limiting) weighting function parameters. The program requires that all but one of the weighting function parameters are set. The program determines the optimal (limiting) value of the unknown weighting function parameter by scanning through the possible values of the unknown weighting function parameter and determining if the Riccati equation solution, X_∞, is positive semi-definite. The program reduces the time required to determine the limiting values of weighting function parameters by 98% (if the reduction in the required number of flops is related directly to the time taken to determine optimal weighting function parameters). The program is given in the Appendix D. The program was used determine if a positive semi-definite solution existed for the range of model parameters used in the STR (detailed in Chapter 7).

6.8 THREE TERM TIME SERIES CONTROLLER.

During the study of the relationship between weighting function parameters and the closed-loop system responses, it was noted that many of the H_∞ controllers could be reduced to three term controllers, via zero/pole cancellation, of the form

$$u(t) = A*e(t) - B*e(t-1) + C*u(t-1) \tag{6.20}$$

where e(t) = positional error at time t

u(t) = control signal at time t.

A study of the controller response yielded the following points:-

1) The value of B should always be less than the value of A.

2) The absolute value of C should be less than unity.

3) Increasing the value of A in relation to the value of B increases the speed of response of the system, but also the level of overshoot.

4) Increasing the value of C leads to an increase in the speed of response and a reduction in the steady-state error.

5) Increasing the value of B in relation to the value of A decreases the speed of response and the amount of overshoot.

6) Increasing A and B by proportional amounts leads to an increase in the speed of response and the amount of overshoot.

The three term controller can be related to the traditional PID controller, since a PD controller is given by

$$u(t) = P*e(t) + D*[e(t) - e(t-1)] \qquad (6.21)$$

which is equivalent to the time series controller (6.20) with the values of A and B set as A = P+D and B = D. The integral term in a PID controller is the summation of the previous errors. The C term in the three term controller is equivalent to the integral of the sum of the previous and the differential of the previous error terms. Therefore the close relationship between the two controllers can be seen. The three term controller requires 5 mathematical operations against the 7 required by the PID controller. Therefore the time series controller could be calculated in a reduced time frame, leading to a reduced sampling delay, which could lead to an increase in the possible system bandwidth (Chapter 3). Additionally the system is inherently more flexible since it does not have a fixed pole on the unity circle in the z-domain (note the three term time series can be expanded to include an additional error term to make it equivalent to the traditional PID controller).

The three term controller detailed here is simply a lead or lag compensator and is included only for completeness.

6.9 CONCLUSIONS.

This chapter has addressed the problem of performance weighting function selection in H_∞ controller synthesis. An endpoint error value criteria has been suggested as a basis for the structured selection of performance weighting function parameters. Two distinct parts of the design synthesis problem have been distinguished - designing the initial controller and refining the controller, via performance weighting function adjustment, to match the desired closed-loop system response. A direct transformation between the parameters of the nominal model, weighting functions and the resultant controller has been used in both cases. The method has been demonstrated on simulated design examples. The method is fundamentally different from the approach used by Lundström *et al.* [1991] and Piché *et al.* [1991], in that the actual effect that a change in the performance weighting function has on the controller, and hence the closed-loop system, can be seen and quantified and is not simply given in general terms. Details of a scanning function, used to determine limiting weighting function parameters, has been given. The effect of varying the parameters in a three term time series controller has been given.

CHAPTER 7.

ADAPTIVE CONTROL.

7.1 INTRODUCTION.

This chapter details the development of a single axis self-tuning regulator (STR). The STR was developed to overcome the effects (in terms of reduced performance) of variations in the dynamics of high speed independent drive systems (the dynamic variations are due to the characteristics of the loads and the dc servo systems used and occur both within the event cycle and over longer periods). Also a set-point gain scheduling (SPGS) controller, developed to increase the performance of a single axis (by switching between different controllers during the event cycle) is detailed. The SPGS controller is included in this thesis, even though outside the mainstream of the research completed, since it produced a significantly reduced cycle time for a design example.

7.2 BASIC THEORY OF ADAPTIVE CONTROL.

An adaptive controller alters the controller parameters on-line in order to improve system performance as measured by performance criteria (the two standard performance criteria for this application are defined in terms of endpoint and average position errors).

For an adaptive controller to function correctly a number of criteria must be satisfied:

(1) Some form of identification of either the plant dynamics or plant state must be available within certain time and accuracy limits.

(2) A controller modifier must be available to alter the controller in an appropriate fashion and to ensure that the system is not consistently unstable.

(3) The entire adaptive process must operate within a time frame such that the dynamics of the real system are, within certain limits, equivalent to those of the identified system.

7.3 SET-POINT GAIN SCHEDULING.

7.3.1 General Theory of Set-Point Gain Scheduling.

The concept of SPGS is that separate controllers can be implemented within the event cycle, dependent on the demanded system dynamics, to enhance performance. For example using a high-gain controller when a quick response is required and using a lower gain controller to reduce overshoot. The use of this controller requires that the duty cycle be repetitive in nature with a "known" variation. The SPGS controller developed switched between two proportional controllers (although more and more complex controllers could be used). In general the effect of a high gain proportional controller is to increase the speed of the system response and the amount of overshoot while a low gain proportional controller decreases the speed of system response and overshoot. A mixture of the two controllers could produce a system with a rapid transient response and a low level of overshoot, which would be an "ideal" response for high speed machinery. For the simple SPGS controller developed, two system responses were recorded, using a high and low gain proportional controller. The controller switch-point is determined by superimposing the output responses, with the low gain system output shifted backward in time, relative to the high gain system output, to produce the "ideal" system response. The crossover point of the traces is then used to determine the switchpoint by subtracting the effect of the system time delay, which is inherent in the servo system, from the crossover point on the high gain system response (demonstrated in the design examples). The switch-point is dependent, via the output position, on the system dynamics. In practice either the switch-point, or the positional (proportional) gains, require tuning to achieve the desired performance.

7.3.2 Simulated Results.

An ACSL simulation was constructed, based on an industrial design problem (using the simulation out-lined in Chapter 3). The SPGS controller was implemented in the outer position loop with an update time of 1 ms (corresponding to that of the Themis motor-controller card). The design example required the indexing an inertial load of $2x10^{-4}$ kgm^2 through 70°, in 20ms or less, using a motor with a rotor inertia $8.3x10^{-5}$ kgm^2 with an endpoint accuracy of ±1°. The servo system has a continuous stall torque rating of 2.26 Nm and a peak torque rating of 4.97 Nm. The optimal gear ratio was determined as 1.5:1 so that the referred load inertia to the motor is $8.888x10^{-5}$ kgm^2 and the required motor index is 105° with an endpoint accuracy of ±1.5°. Since the optical encoder on the drive had 8000 pulses per revolution the required index was 2334 pulses with an endpoint error of ±33 pulses. This design problem relates to the limiting axis on a process machine where all other axes easily meet their specifications.

Two proportional controllers were implemented (a high gain of 0.00478 and a low gain of 0.00123) and the system responses recorded. The high gain proportional controller was selected to produce a system with a rapid transient response without instability. The low gain value was selected to produce a system response with no overshoot. The two system responses were superimposed with the low gain controller (solid line) response shifted 40ms backwards in time, relative to the high gain (dotted line) system response, in order to produce the "ideal" output response, shown in Figure 7.1. The switchpoint is determined by examining the crossover point of the two traces, an output position of 2100 pulses (in this case) and taking into account the time delays that are inherent in the dc servo system. The system response shows a time delay of 4ms. Therefore the switchpoint is determined as the crossover point of the output responses minus 4ms (on the high gain system response) which is an output position between 500 and 690 pulses. During the tuning period of this controller the switch point was determined as an output position of 600 pulses.

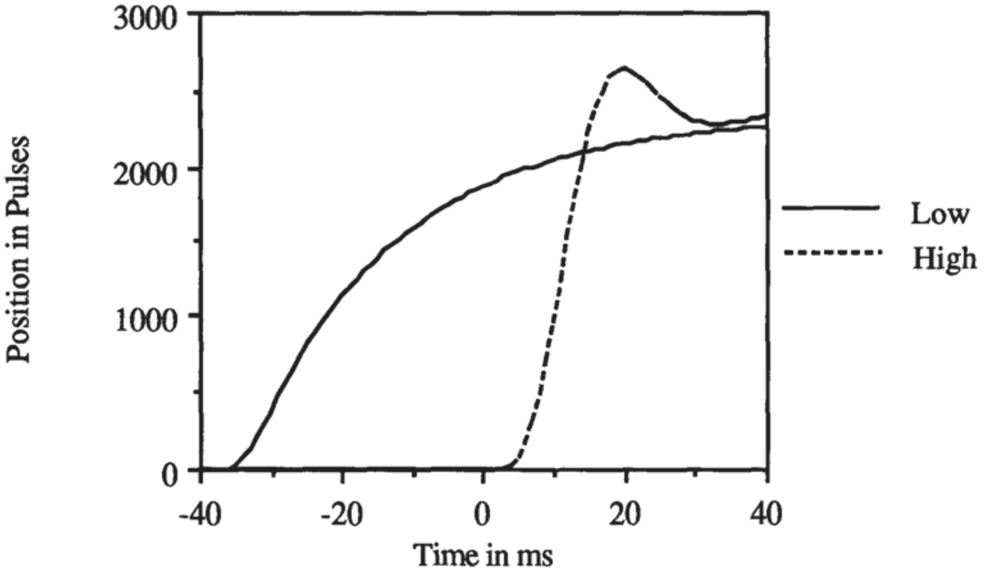


Figure 7.1 Determination of Switchpoint for SPGS Controller.

Three controllers were implemented. A PID with VFF controlled system met the specification in 19ms, Figure 7.2. (For clarity all graphs show only the positional errors after 15ms). The PID with VFF controller was optimised using the standard tuning process. The PID with VFF controller parameters were a P gain of 0.00339, a VFF value of 0.03204 (the I and D were both zero). An H∞ controller was determined (as outlined in chapter 6) and when implemented produced a system which met the specification in 19ms, Figure 7.3. The H∞ controller was

$$u(t) = 0.00361*e(t) - 0.00352*e(t-1) + 0.972*u(t-1) \tag{7.1}$$

where u(t) is the command signal at time t

e(t) is the error signal at time t

A SPGS controller, consisting of two proportional controllers and one switch-point (determined as outlined in 7.3.1) was implemented and produced a system which met the specification in 17ms, Figure 7.4. The SPGS controller parameters were a high gain of 0.00478, a low gain of 0.00123 and a switchpoint of 600 pulses (output position).

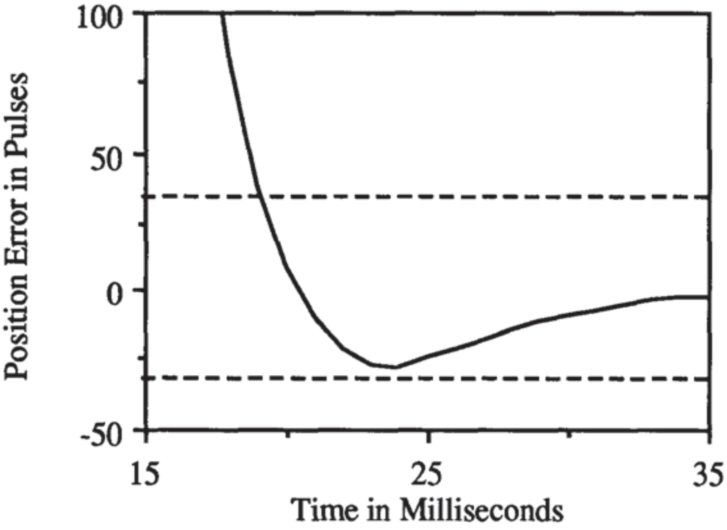


Figure 7.2 PID with VFF Controlled System.

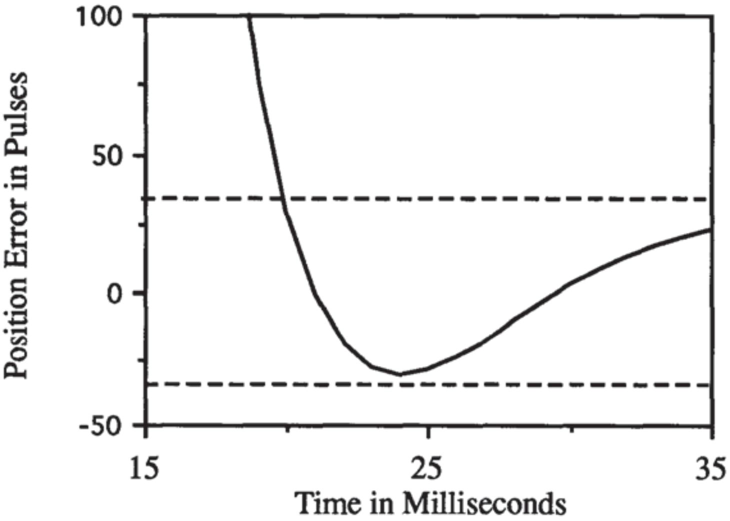


Figure 7.3 H∞ Controlled System.

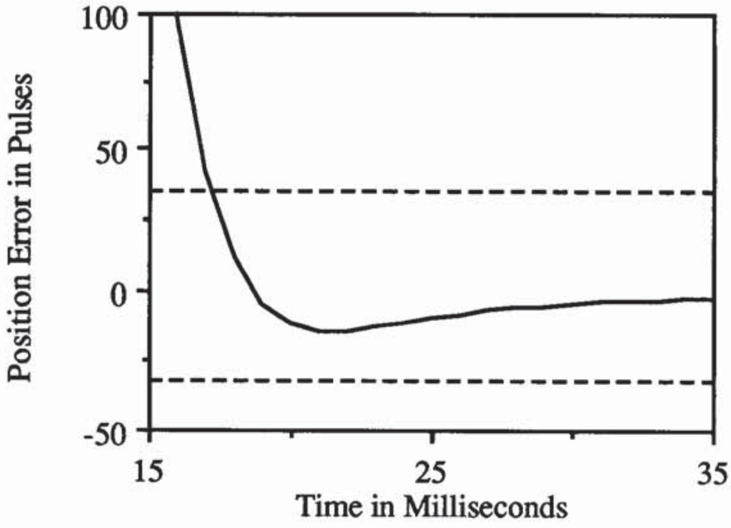


Figure 7.4 SPGS Controlled System.

To determine the reason for the better performance achieved produced by the SPGS controller the SPGS (dotted line) and PID with VFF (solid line) control signals were examined, Figure 7.5.

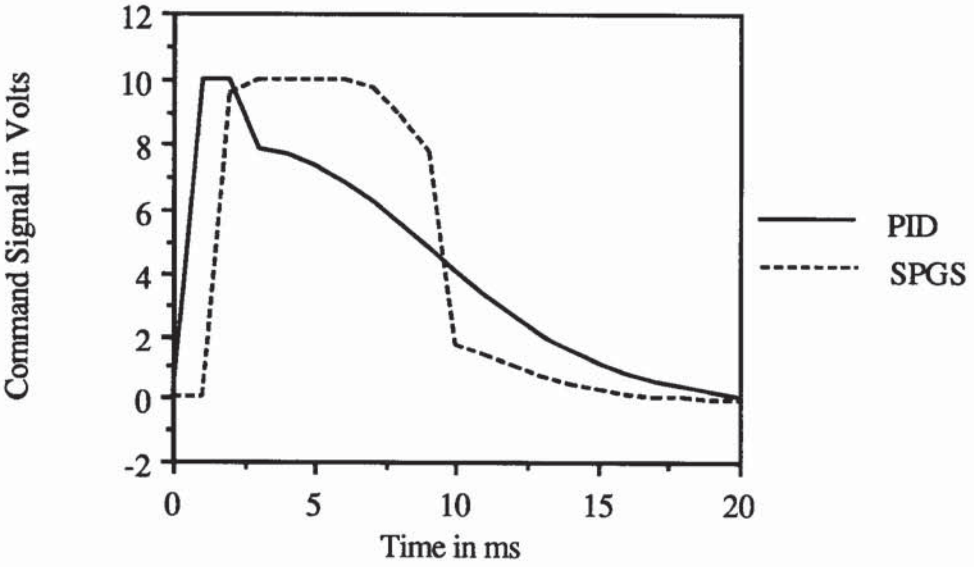


Figure 7.5 Command Signals

It can readily be seen that the SPGS control signal is consistently high during the initial stages of the cycle, in effect forcing the motor to move, whereas the PID with

VFF controller signal peaks and then drops below the SPGS control signal. The SPGS control signal drops off near the end of the increment thereby reducing the level of overshoot.

7.3.3 Practical Results.

To validate the simulated results a scaled version of the design example was constructed using the Drives Test Facility at Aston University. The design example consists of an Electro-Craft™ BRU200 DM-30 drive module with a S-4075 motor (rotor inertia of 6.8×10^{-4} kgm^2) directly coupled to a non-varying inertial load of 4.472×10^{-4} kgm^2. The servo system has a continuous stall torque rating of 10.2 Nm and a peak torque rating of 19.7 Nm. The load is required to be indexed through 105° in 28ms (since the motor has a lower torque to inertia ratio, than the motor in the simulated example, and therefore has a lower possible travel distance in a particular time) or less to an endpoint accuracy of ±1.5°. The motor shaft optical encoder has 8000 pulses per revolution the index is 2334 pulses with an endpoint error of ±33 pulses. The BRU 200 internal velocity control-loop and outer positional control-loop had an update time of 1ms. The SPGS controller was implemented in the outer positional control-loop only (on the Themis controller card, using software developed by the author).

To determine the SPGS controller parameters a high gain proportional controller, gain 2.44, and a low gain proportional controller, gain 0.854, were implemented and the corresponding system responses were recorded. As in the simulated design example the high gain controller was selected to produce a rapid transient response without system instability and the low gain controller was selected to produce a closed-loop system with no overshoot. The two responses were superimposed with the low gain system response (solid line) shifted backwards in time by 5ms, relative to the high gain controller response (dotted line), in order to produce the "ideal" system response. The switchpoint was determined by subtracting the servo system time delay from the high gain controlled system as an output position value of between 200 and 400 pulses.

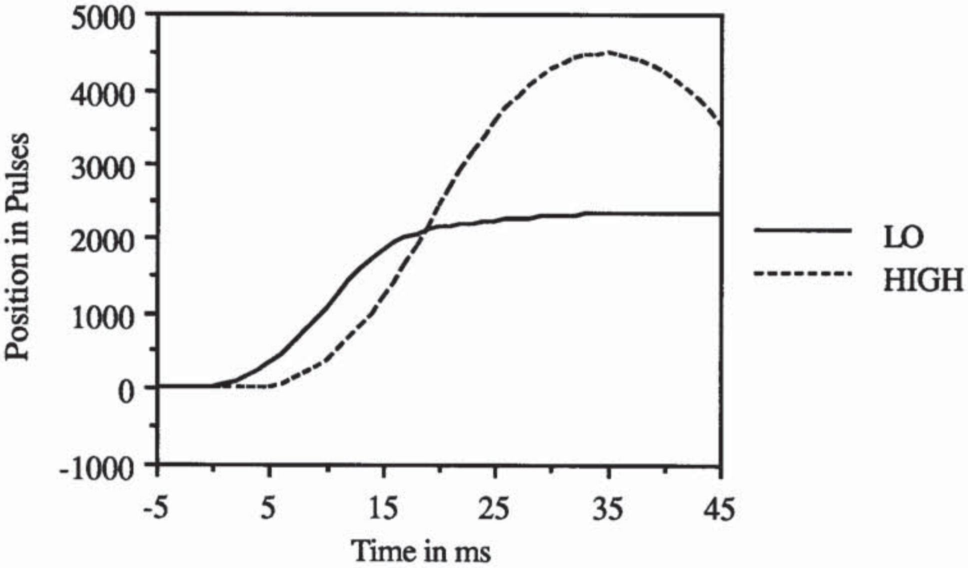


Figure 7.6 Switchpoint Determination for SPGS Controller.

The PID with VFF controlled system met the specification in 29ms, Figure 7.7. The PID with VFF controller values were a P value of 0.879 and a VFF value of 0.9 (both I and D were set to zero). The H∞ controlled system met the specification in 29ms, Figure 7.8. The H∞ controller used was

$$u(t) = 0.9209*e(t) - 0.8936*e(t-1) + 0.9766*u(t-1) \qquad (7.2)$$

notation as in (7.1).

The SPGS controlled system met the specification in 25ms, Figure 7.9. The SPGS controller values were a low gain of 0.854, a high gain of 2.44 and a switchpoint of 350 pulses.

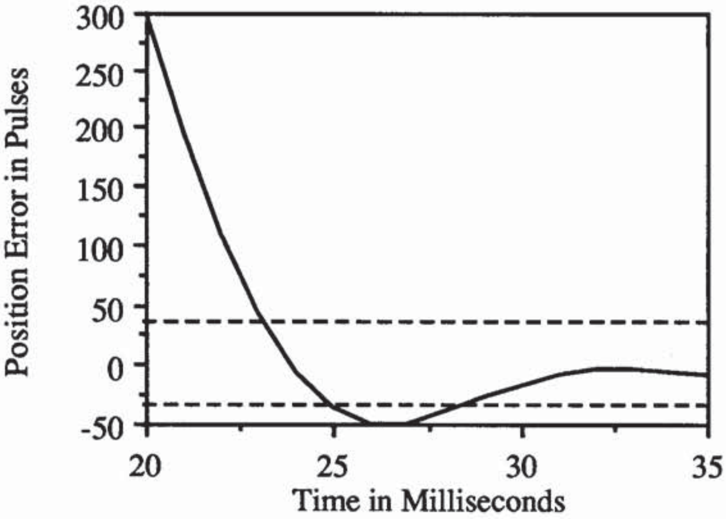


Figure 7.7 PID with VFF Controlled System.

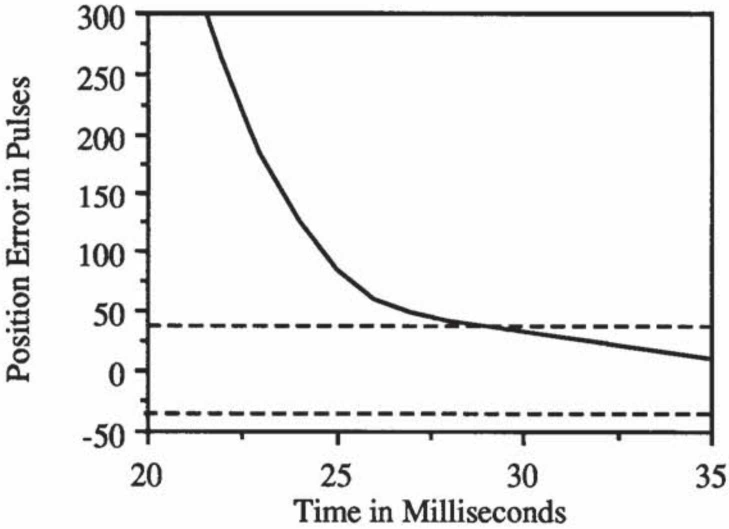


Figure 7.8 H∞ Controlled System.

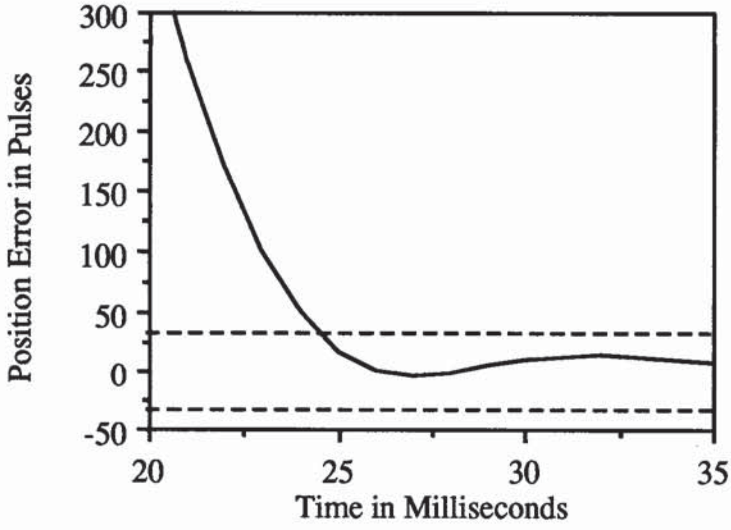


Figure 7.9 SPGS Controlled System.

7.3.4 Controller Action.

In order to determine the reason for the better performance, in terms of reduced increment time, achieved by the SPGS controller an examination was completed of the controller output signal, Figure 7.10 (the dotted line is the PID with VFF controller output, the solid line is the SPGS controller output). This shows, as expected, that the SPGS controller gain is high during the initial period of the demanded increment but low at the end of the increment (to stop any overshoot). The effect of the velocity feedforward controller component on the control signal (for the PID with VFF controller) can be seen, during the initial the increment part of the cycle where the control signal is high.

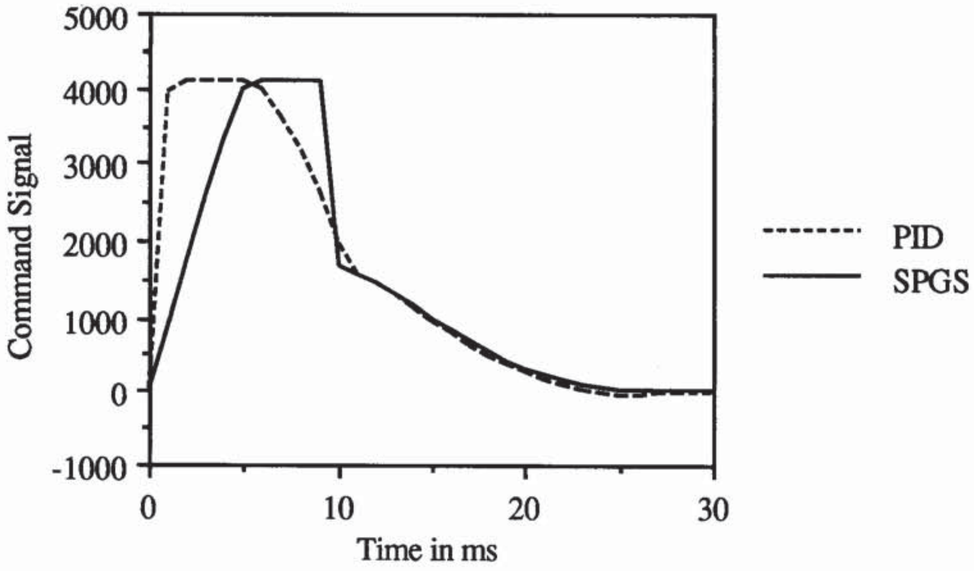


Figure 7.10 Control Signal From SPGS and PID with VFF Controllers.

7.3.5 Repeatability of SPGS Controller.

To assess repeatability and reliability, the SPGS controller was implemented over twenty cycles. It can be seen, Figure 7.11, that the time to meet specification varies between 28ms and 23ms with the controller always producing a system meeting the specification. The PID with VFF and H∞ controlled systems showed similar levels of variation, never producing a response which meet the performance requirement.

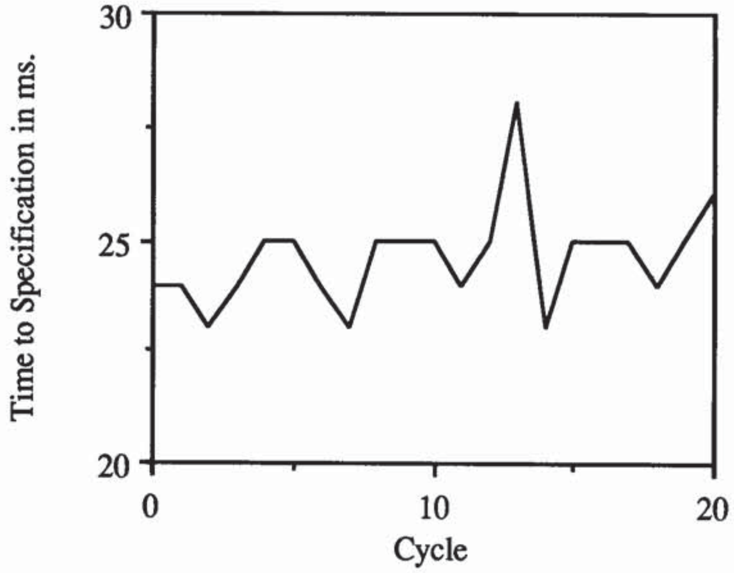


Figure 7.11 Repeatability of SPGS Controlled System.

7.3.6 SPGS Concluding Remarks.

The SPGS controller produced a 10.5% reduction in increment time in simulation and a 13.8% reduction in increment time for a design example implemented on a practical system. The technique is simple, quick and easy to implement and appears to offer substantial benefits for industrial applications of high-speed independent drives (although no exhaustive study was completed on the technique). However the technique is limited to systems where a large degree of information is available, such as systems with repetitive cycles and / or load variations. The results contained in section 7.3 have been presented at an IEE Colloquium on Precision Motion Control [Beaven *et al.* 1994b] and a full version of the work is to be published in Control Engineering Practice [Beaven *et al.* 1995a].

7.4 SELF-TUNING REGULATOR.

7.4.1 Fundamentals of a Self-Tuning Regulator.

Chapter 4 demonstrated that H∞ controller synthesis (in conjunction with least-squares parameter estimation methods) can be applied to determine controllers for high-speed independent-drive systems.

In an attempt to improve the control of independent drives, two alternative control avenues are open: robust control and adaptive control. Robust control, implemented via such methods as H∞-norm optimisation, involves the implementation of a single controller which is "immune" to plant variation. Adaptive control [Åström and Wittenmark 1989] aims to maintain a required level of performance by compensating for plant parameter variations by appropriate controller adjustment. The two concepts have been combined to produce robust adaptive controllers [Lewis *et al.* 1993].

One form of adaptive controller is the self-tuning regulator, shown schematically in Figure 7.12. In this form of controller, the adaptation problem is subdivided into two distinct parts [Tal 1989]: plant identification and controller modification. The parameter identifier supplies information on the state of the plant and the controller modification algorithm adjusts the controller function accordingly.

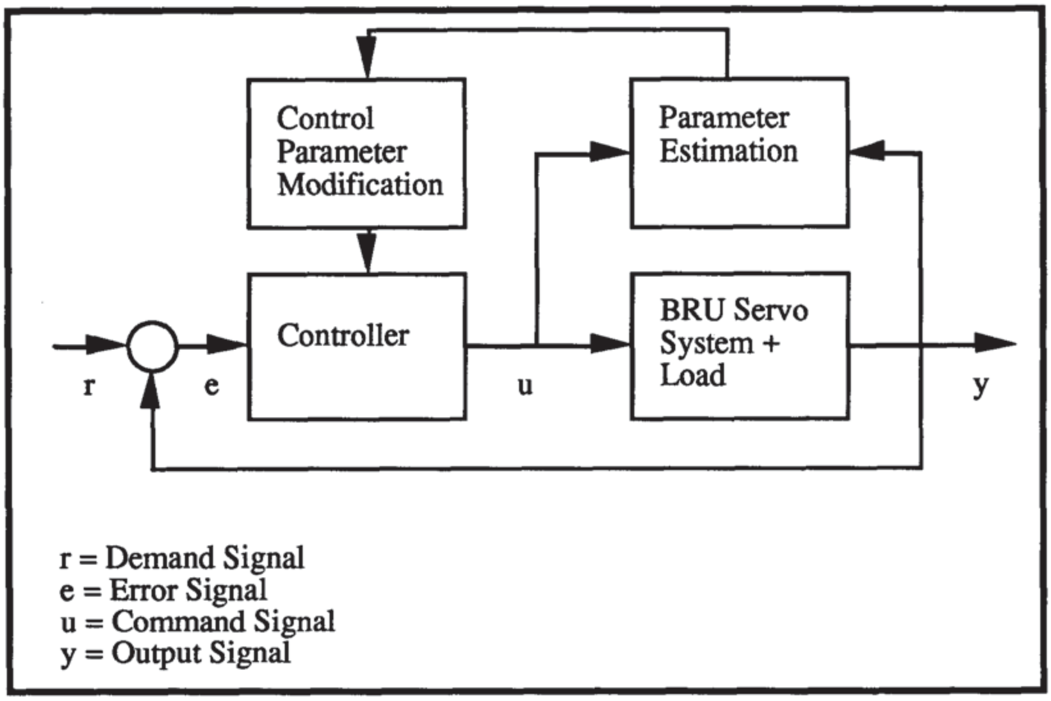


Figure 7.12 Schematic of a Self-Tuning Regulator.

This form of adaptive control is a logical extension of the research completed in the Chapter 4. However three potential problems have been identified with this approach [Åström and Wittenmark 1989] :

(a) To ensure that the estimated parameters converge to the true system parameters it is necessary that the process input be persistently exciting and that the model structure be appropriate.

(b) The map from plant parameters to controller parameters may have singular points if the estimated process has coincident poles and zeros.

(c) Stability analysis is complex.

In the application area under study, however, the self-tuning regulator approach is still applicable: the general form of reference motion profile is usually a series of indexes or steps with a quantisation ripple (on the error signal and hence on the control signal) always present, yielding a process input which is persistently exciting; for dc servo drive systems, appropriate forms of model are available which preclude coincidental poles and zeros (see Chapter 3). Stability analysis remains a difficulty, although it has been demonstrated that, if the controller can be switched, between alternative stabilising controllers, within a short enough time frame, instability may be avoided [Hyde and Glover 1990].

A STR was produced by combining the simplified recursive least squares parameter estimation technique with the simplified H_∞ controller synthesis. The STR was implemented on the MVME147 Motorola MC68030 card (using software developed by the author). The controller modification time was determined as 8ms.

7.4.2 The Performance Problem.

Whether an adaptive controller will produce an increase in system performance is of key importance. Chapter 6 demonstrated that the performance weighting function parameters could be adjusted to achieve (or at least move towards) specified closed-loop performance. The controller modifier (devised by the author) uses constant performance and robustness weighting functions parameters throughout the adaptation process (reducing the computational requirement and adaptation time to a reasonable level). Therefore the ability of the identifier to track parameter variation determines whether the adaptive controller will produce an increase in system performance.

The weighting function parameters are selected so that the required performance (assuming it is possible) is obtained for a non-varying system, effectively compensating for any identification errors. For the adaptation process to keep the system dynamics constant, effectively cancelling out the system variation, the identified model must maintain the magnitude ratio of the identified and real system dynamics constant. That is, it must have the same relative level of identification error and it must maintain the same absolute value of phase shift. However if the magnitude ratio and absolute phase shift are not constant then the system performance may increase or decrease.

The argument that the closed-loop system performance remains constant if the relative error magnitude and the absolute value of the phase shift remains constant is as follows. For stable systems, the controller is a function of the weighting functions (as seen in the direct-form of the controller). Hence if the weighting functions are kept constant any alteration in the identified model will relate directly to the determined controller. In terms of magnitude, if the actual plant and its associated nominal model are perturbed to

$$G = G(1+\Delta) \text{ and } G_o = G_o(1+\Delta) \tag{7.1}$$

Then the controller, determined from the nominal model, will be perturbed to

$$K(s) = K(s)/(1+\Delta) \tag{7.2}$$

The perturbation in the controller will cancel the effect of the perturbation in the actual plant, causing the closed-loop performance to remain constant. Therefore for the adaptive controller to maintain constant performance the ratio of the magnitude of the identified and real plants must remain constant. (Note the system performance may be improved if this ratio is not maintained). For phase shift, a similar argument holds, except that since phase shift is a cumulative effect, the absolute phase shift is important. However the theoretical phase advance and delay displayed by $H\infty$ controllers does not occur in practice, for single shot motion increments, therefore any change in phase would require an equal and opposite change in phase of the identified system. This could conflict with the stability criteria as determined by the robustness weighting function. However the majority of design examples are concerned with holding error (i.e. low frequency), where the phase shift is due to the time delays inherent in the system which do not alter (see Chapter 3). Thus the phase shift requirement is as suggested theoretically (to meet stability requirements).

7.4.3 Stability of Adapting System.

The system stability is guaranteed if the actual plant is within the stability margin defined by the robustness weighting function (see Chapter 3). However as Bitmead *et al.* [1990] noted, the controller has the effect of increasing the system bandwidth which can lead to an inexorable drifts towards high frequency model fits which can produce unstable controllers. Bitmead *et al.* [1990] suggested using an identifier filter (which filters the signals sent to the identifier) to overcome this problem. This thesis uses a similar approach of limiting the identified plant parameters (discussed in Section 7.5.2) to overcome the problem of interaction between the identifier and controller modifier causing system instability.

7.5 RECURSIVE LEAST-SQUARES PARAMETER ESTIMATION.

7.5.1 Parameter Estimation Errors

The nominal model exhibits two forms of identification errors. Firstly those inherent to the identification process and secondly those due to the time delay between data-point input and the controller modification (the STR's mathematical operations require a finite length of time). In this case, since the STR has a modification time of 8ms, the controller implemented uses the nominal system model determined 8ms previously.

The time delay between the data input and controller modification (i.e. the use of the model) causes two forms of identification errors:-

1) The system dynamics will alter over the time period between the data input and the controller modification.
2) The identifier will require a number of cycles to adjust to a change in the system parameters, i.e. if the algorithm takes 10 times as long it will take 10 times as long to adapt.

7.5.2 Study of Parameter Variation.

The research of Wilkes [1995] demonstrated that recursive least squares parameter estimation with variable forgetting factor was the most suitable identifier, for this application due to the computational and time limitations. The recursive least squares parameter estimation technique (with simplifications) was detailed in Chapter 5. A series of simulations were completed, using ACSL, to determine the ability of the algorithm to track parameter changes. A simulation of a BRU200 system with a step change in inertial load referred to the motor from 16.1x10^{-5} kgm^2 on the forward stroke to 9x10^{-5} kgm^2 on the reverse stroke (the inertial variation represents removal of a work component) was constructed. The simulation contains the fully implemented STR (i.e. the controller parameters are constantly being adjusted) and the time delay in controller modification is incorporated into the simulation.

7.5.2.1 Model Error Determination.

The direct-form controller shows that the absolute or relative value of each individual identified parameter, or any model error term based on individual model parameters, was invalid for determining the model validity for this application (performance enhancement using a STR). An absolute or relative change in identified parameters, except α, does not cancel the corresponding change in the actual system and therefore would not keep the closed-loop response and hence the system performance constant. Therefore a Relative Magnitude Error (RME) and Absolute Phase Error (APE) frequency term were developed for the purpose of evaluating different identification strategies.

$$\mathrm{RME} = \sum_{\omega=0}^{1000} \left| \frac{G_o(j\omega)}{G(j\omega)} - 1 \right| , \quad \mathrm{APE} = \sum_{\omega=0}^{1000} |\,\mathrm{Phase}(G\text{-}G_o)| \qquad (7.3)$$

The relative magnitude error term determines the ratio between a real and identified system over the frequency bandwidth of the system (note the swept-sine test, using appropriate magnitude signal, is used to determine the "real" frequency response and thus this error analysis provides only guidance to the validity of the identifier). The absolute phase error determines the absolute phase shift between a real and identified system over the frequency bandwidth of the system

7.5.3 Simulated Estimation Examination.

The frequency error terms allows an accuracy comparison to be made for different cycle time, different minimum forgetting factors (FFs) and different variable forgetting factor weight (FFWs).

Four cycle times were used in the analysis, 100ms, 200ms, 300ms and 400ms. Four minimum FF values were used 0.1, 0.5, 0.9 and 0.95 (shown in the tables 7.1a to 7.4b, the maximum FF value was set at 0.999 in all cases). Five values of FFW were used 1, 10, 100, 1000 and 10000. The average frequency error identification term for each cycle can be seen in Tables 7.1a, 7.1b, 7.2a, 7.2b, 7.3a, 7.3b, 7.4a and 7.4b. The nominal model relative magnitude error values for the loaded and unloaded system were determined as 0.2309 and 0.0691 respectively. The absolute phase error values were 0.2786 and 0.2961. Therefore for the identifier to maintain the same system performance as the non-varying system, the relative magnitude error should be 0.2309 and the absolute phase error should be 0.2786.

Table 7.1a Relative Magnitude Errors For Cycle Time of 100ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.6974	0.7490	0.6589	0.4801	0.4417
0.5	0.5165	0.5801	0.4868	0.3683	0.9695
0.9	0.5630	0.5631	0.5634	0.5639	0.5655
0.95	0.6144	0.6126	0.6112	0.6050	0.5714

Table 7.1b Absolute Phase Errors For Cycle Time of 100ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.7605	0.8806	0.5822	0.5879	0.7464
0.5	0.6560	0.6039	0.5701	0.5500	0.8440
0.9	0.9254	0.9254	0.9254	0.9252	0.8375
0.95	0.7375	0.7376	0.7375	0.7376	0.6510

Table 7.2a Relative Magnitude Errors For Cycle Time of 200ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.5169	0.5431	0.6826	0.4999	0.8399
0.5	0.4738	0.4005	0.3451	0.4094	0.4371
0.9	0.6379	0.3868	0.4141	0.3129	0.4890
0.95	0.4222	0.4236	0.4794	0.4918	0.8895

Table 7.2b Absolute Phase Errors For Cycle Time of 200ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.5809	0.5791	0.5379	0.3816	1.4203
0.5	0.4061	0.4728	0.4802	0.4854	0.4893
0.9	0.5143	0.5528	0.5521	0.5715	0.4873
0.95	0.5323	0.5479	0.5042	0.4095	1.3169

Table 7.3a Relative Magnitude Errors For Cycle Time of 300ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.6599	0.6618	0.7299	0.6877	0.7017
0.5	0.6772	0.6737	0.6857	0.8853	0.8322
0.9	0.6927	0.6825	0.5171	0.6340	0.7203
0.95	0.6995	0.8531	0.7850	0.7243	0.6399

Table 7.3b Absolute Phase Errors For Cycle Time of 300ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.7623	0.7422	0.9771	0.6492	1.2861
0.5	0.6144	0.7532	0.6413	0.8215	1.4303
0.9	1.2529	1.1823	1.3672	1.4101	1.5090
0.95	1.7918	1.7536	1.7564	1.7613	1.7560

Table 7.4a Relative Magnitude Errors For Cycle Time of 400ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.7446	0.6713	0.7493	0.7552	0.7215
0.5	0.7018	0.8844	0.7869	0.7091	0.6659
0.9	0.2916	0.2985	0.3344	0.7815	0.3767
0.95	0.5338	0.5653	0.6978	0.5449	0.2319

Table 7.4b Absolute Phase Errors For Cycle Time of 400ms.

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	1.0456	1.1033	0.9432	1.1635	1.0147
0.5	1.0714	1.0784	0.7719	0.8915	0.8802
0.9	0.5879	0.5867	0.5998	0.9628	0.5522
0.95	0.7593	0.7735	0.8506	0.6070	0.5839

The comparison shows that the identification technique (for the selected cycle times, FF and FFW values) in standard format is inadequate for use in a STR (using the performance theory developed in 7.4.2). This was substantiated by a comparison of the position errors produce by the system controlled by a STR and a single controller, with the system controlled by a single controller producing better results. This can be seen in Figure 7.13 with the absolute endpoint errors for the first ten cycles correspond to a single H∞ controller and the final twenty correspond to a STR implemented with no identified parameter limits. In Figure 7.13 the solid line corresponds to the forward motion while the dotted line corresponds to the reverse motion, the cycle time is 100 ms the forgetting factor weight is 1000 and the minimum forgetting factor is 0.5. The STR results initially showed worse performance than a single controller and then drifted into instability.

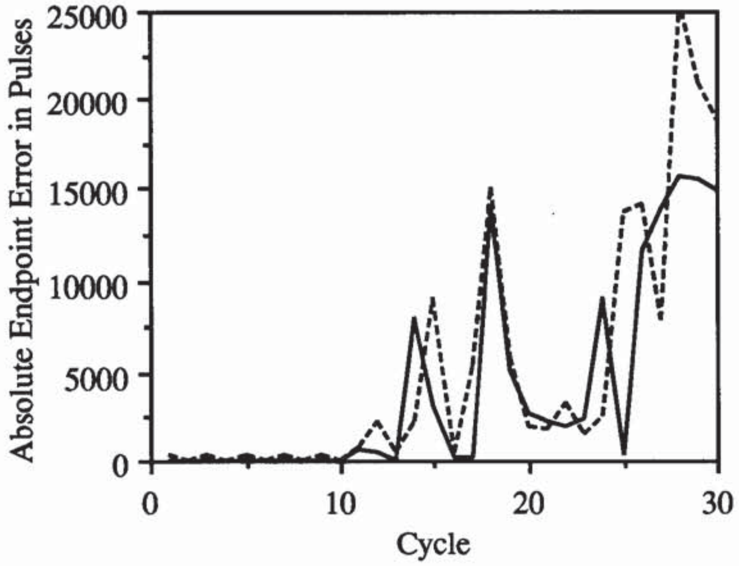


Figure 7.13 Endpoint Errors No STR Parameter Limits

A study of the identified model parameters, Figures 7.14a, 7.14b and 7.14c, showed that when the inertial variation occurs the estimated parameters oscillate (note a test was completed to ensure that the parameter variation was not due to the identifier tracking the input signal) and over a longer period the identified parameters drift causing instability in the system. To overcome the problem of transient "jumps" and drift in the estimated parameters limits were placed on the identified model outputs.

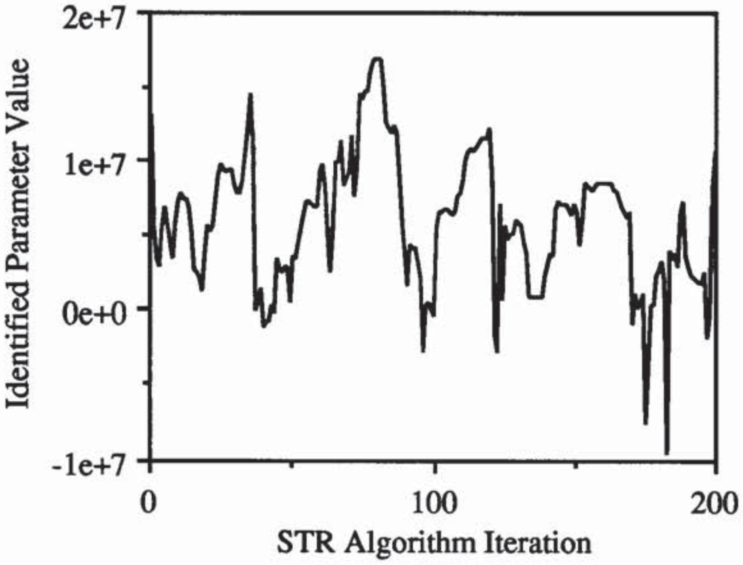


Figure 7.14a Variation in α Identified Parameter

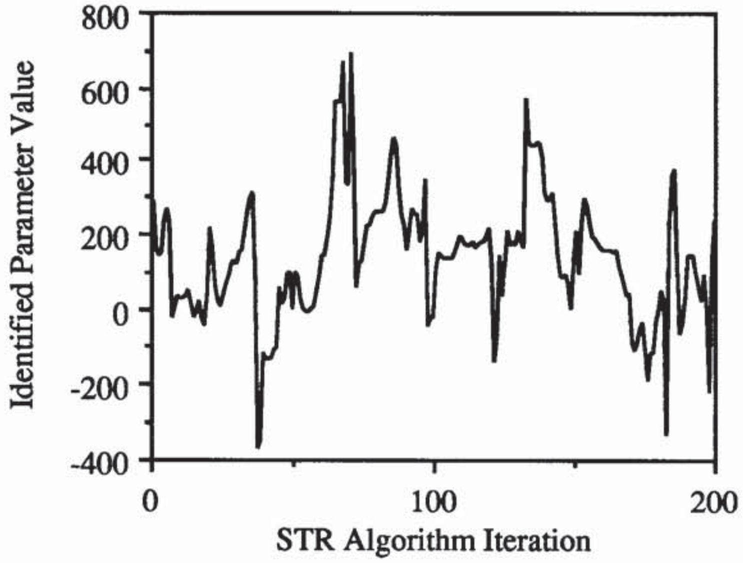


Figure 7.14b Variation in β Identified Parameter

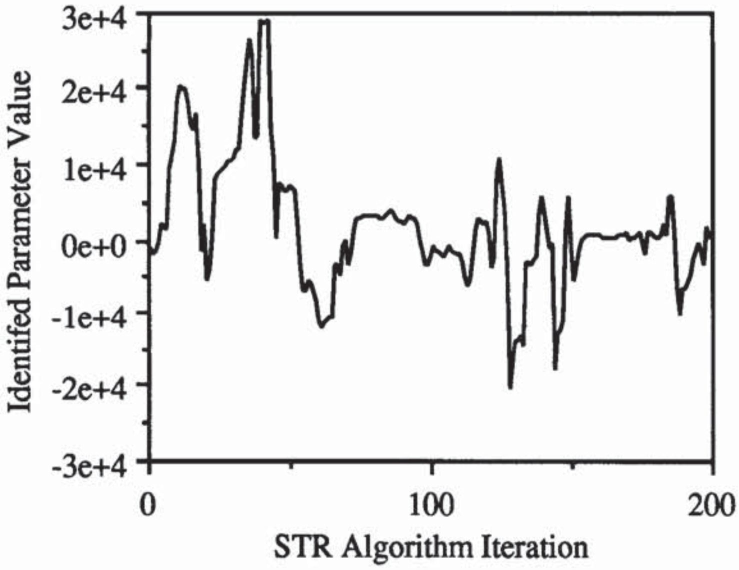


Figure 7.14c Variation in δ Identified Parameter

Bitmead *et al.* [1990] noted a similar phenomena where the interaction of the controller modifier and identifier caused a degradation in system performance. Bitmead *et al.* [1990] used an identifier filter, which limits the frequency content of data input to the identifier and hence limits the estimated model parameters, to overcome this problem. This is similar to the approach used in this thesis, where limits are placed directly onto the identified model parameters. Placing limits directly onto the identified plant parameters overcomes the possible problem of data corruption due to filtering. However, as with filtering the signal, a number of additional parameters (identified model limits) have to be determined. The identified model parameter limits were set at ±10% of the nominal model values. The simulations were repeated and the results in Tables 7.5a, 7.5b, 7.6a, 7.6b, 7.7a, 7.7b, 7.8a and 7.8b obtained. These results show that for certain combinations of cycle time, FF and FFW values the average relative magnitude and absolute phase errors are at a level where the adaptive controller should be expected to maintain the performance to non-varying levels. It also demonstrated that the use of limits on the identified model stops the problem of parameter drift causing system instability.

Table 7.5a Relative Magnitude Errors Bandlimited Identifier 100ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.2409	0.2421	0.2488	0.2403	0.2113
0.5	0.2076	0.2081	0.2321	0.2321	0.2542
0.9	0.2475	0.2475	0.2475	0.2475	0.2779
0.95	0.2427	0.2447	0.2447	0.2447	0.2779

Table 7.5b Absolute Phase Errors Bandlimited Identifier 100ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.2847	0.2852	0.2904	0.2847	0.2936
0.5	0.3008	0.3020	0.2859	0.2915	0.2807
0.9	0.2779	0.2779	0.2779	0.2779	0.2779
0.95	0.2779	0.2779	0.2799	0.2779	0.2779

Table 7.6a Relative Magnitude Errors Bandlimited Identifier 200ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.1748	0.2021	0.1738	0.1514	0.1514
0.5	0.2004	0.1672	0.1738	0.1843	0.1794
0.9	0.1728	0.1886	0.1840	0.1886	0.1515
0.95	0.1422	0.1840	0.1546	0.1576	0.1886

Table 7.6b Absolute Phase Errors Bandlimited Identifier 200ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.2952	0.2928	0.3011	0.3028	0.3028
0.5	0.2960	0.3043	0.2974	0.2956	0.2973
0.9	0.3068	0.2933	0.2953	0.2933	0.3068
0.95	0.3068	0.2953	0.3037	0.3068	0.2933

Table 7.7a Relative Magnitude Errors Bandlimited Identifier 300ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.1954	0.2074	0.1951	0.1911	0.1893
0.5	0.2034	0.2128	0.2128	0.1881	0.1627
0.9	0.1344	0.1343	0.1543	0.1406	0.1496
0.95	0.1935	0.1942	0.1982	0.1963	0.1861

Table 7.7b Absolute Phase Errors Bandlimited Identifier 300ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.2916	0.2938	0.2913	0.2843	0.2907
0.5	0.2820	0.2948	0.2948	0.3093	0.3069
0.9	0.3063	0.3061	0.3012	0.3039	0.3006
0.95	0.2924	0.2920	0.2920	0.2894	0.2933

Table 7.8a Relative Magnitude Errors Bandlimited Identifier 400ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.1889	0.1889	0.2134	0.2132	0.2378
0.5	0.2324	0.2390	0.2179	0.2145	0.1930
0.9	0.1933	0.1933	0.1933	0.1886	0.1625
0.95	0.1914	0.1909	0.1887	0.1071	0.1071

Table 7.8b Absolute Phase Errors Bandlimited Identifier 400ms Cycle Time

Forgetting	Forgetting Factor Weight				
Factor	1	10	100	1000	10000
0.1	0.2936	0.2936	0.2976	0.2983	0.3033
0.5	0.3025	0.3015	0.2953	0.2933	0.2809
0.9	0.2796	0.2796	0.2796	0.2933	0.3228
0.95	0.2866	0.2882	0.2930	0.3259	0.3259

The study was not a critical test of the identification process since the system inputs were constantly varying, due to the controller modification and difference in cycle times, and the point at which adaptation occurred for each test was not varied but it was a realistic comparison in terms of the expected application.

The effects of an increase in the modification time can be seen for the case where the cycle time = 100ms, minimum FF value = 0.5 and FFW = 100 (with the same limits on the identifier outputs as before). The phase and magnitude frequency errors are 0.1691 and 0.3002 which imply that the controller gains would be too high. An analysis of the system performance shows this to be the case as the system does not meet the specification. This shows that the set-up of the STR is crucial and that any change in its parameters (i.e. controller modification time) leads to a degradation in performance.

7.6 IMPLEMENTATION OF STR.

7.6.1 Simulated Design Example.

For a set of plants, weighting functions can be determined such that X_∞ is always positive semi-definite and that, consequently, an internally-stabilising controller always exists. The key assumption is that the plant can be represented as an equivalent second order system with β, δ, and α always positive (Chapter 4 showed this to be a valid assumption) and that the variation in the system will occur between estimated limits.

An ACSL simulation was constructed for an industrial design example (The simulation is as detailed in Chapter 3). The velocity and position control loops both have an update time of 1ms, while the STR loop has a controller modification time of 8 ms.

The update and modification times correspond to measurements taken on a TSVME440 card and MVME147 Motorola MC68030 microprocessor. The simulation is considered to be a valid vehicle for testing the proposed adaptation strategy, since it had been demonstrated previously (See Chapters 3 and 4) to give results which correlate well with experimental data.

The design problem is as follows: an Electro-Craft™ BRU200 DM-25 drive module controlling a S-3016 brushless dc motor (rotor inertia, 8.3 x10^{-5} kgm^2) is required to rotate an axis with total (average) referred inertial load of 5.8 x10^{-5} kgm^2, forwards through 157.5^o in 50 ms and then return the load to its original position in a further 50 ms. The endpoint error for each directional motion must not be greater than ±1.8^o. The motor has a 8000 pulse per revolution encoder and each directional motion therefore corresponds to moving through 3500 pulses in 50 ms with a maximum endpoint error of ± 40 pulses.

The problem is complicated by a number of features:

(a) the total referred inertial load is 5.8 x10^{-5} kgm^2 on the forward stroke but is reduced, by a step function, to 0.7 x 10^{-5} kgm^2 on the return stroke, due to component removal at the end of the forward stroke.

(b) due to component variation the forward stroke inertial load can vary by ± 1% between each cycle.

(c) the simulation also incorporate drift in the torque constant of the motor to replicate dynamic drift observed in practice.

These complications mean that simple "open-loop" adaptive strategies, such as set-point gain scheduling (Section 7.2) are not applicable.

Initially the transfer function of the plant was determined using a swept sine wave test. The design procedure followed is briefly outlined below :

1) system model parameters are estimated using the identifier as described in Chapter 2.

2) an estimate is made of the modelling error, by comparing the frequency responses, of the actual and identified plants.

3) the estimate is used to determine the robustness weighting function.

4) a performance weighting function is determined such that X_∞ (and hence a stabilising controller) exists, for the possible range of plants.

5) the controller is tested on the non-varying system to ascertain whether the performance specification is met. If not step(4) is repeated until satisfactory performance is achieved.

6) the identifier is tested on the varying system and appropriate values of FF, FFW and model bounds are determined.

7) the full STR is implemented, if the performance specification is not met then the process is repeated from step (4).

Initially, a H_∞ controller design was executed with non-varying inertia to mimic the actual commissioning procedure where the drive control parameters are assigned with the axis unloaded (done to protect the process machine). A position PID with VFF controller was determined as a "standard system" benchmark. Axis motions were simulated for both controllers for a period of 2 seconds (20 cycles). The results for the traditional controlled system are shown in Figure 7.15. The results for the single H_∞ controlled system can be seen in Figure 7.16. For clarity, only absolute endpoint errors, for both the forward (solid line) and reverse (dotted line) motions are shown. Both types of controller easily meet the specification.

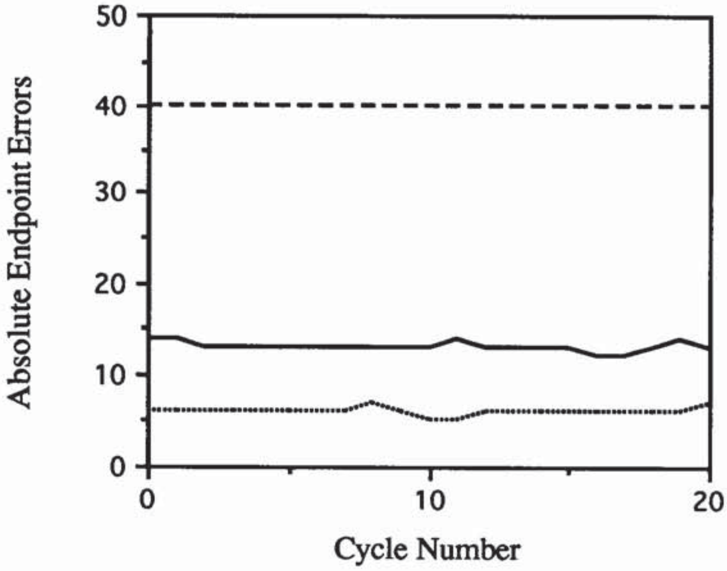


Figure 7.15 PID with VFF Controlled Non-Varying System.

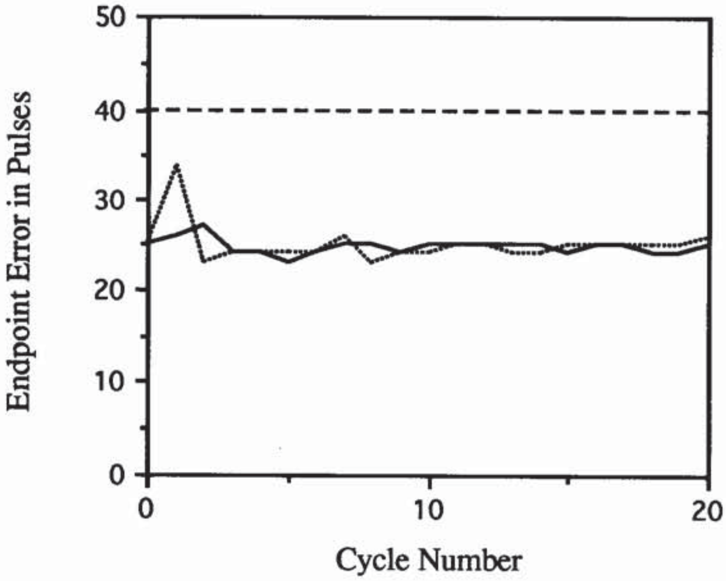


Figure 7.16 H∞ Controlled Non-Varying System.

When the plant variations were introduced neither controller was capable of meeting the specification although the performance of the H∞ controller was found to be clearly superior to the benchmark PID with VFF controller. The results for the traditional controller can be seen in Figure 7.17, whilst the results for the single H∞

controller can be are shown in Figure 7.18. Each set of results is for the same pattern of prescribed plant variations.

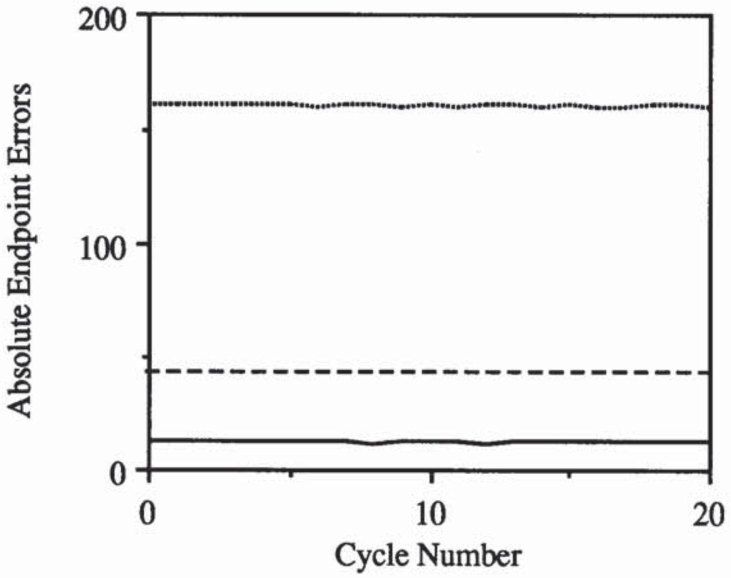


Figure 7.17 PID with VFF Controlled Varying System.

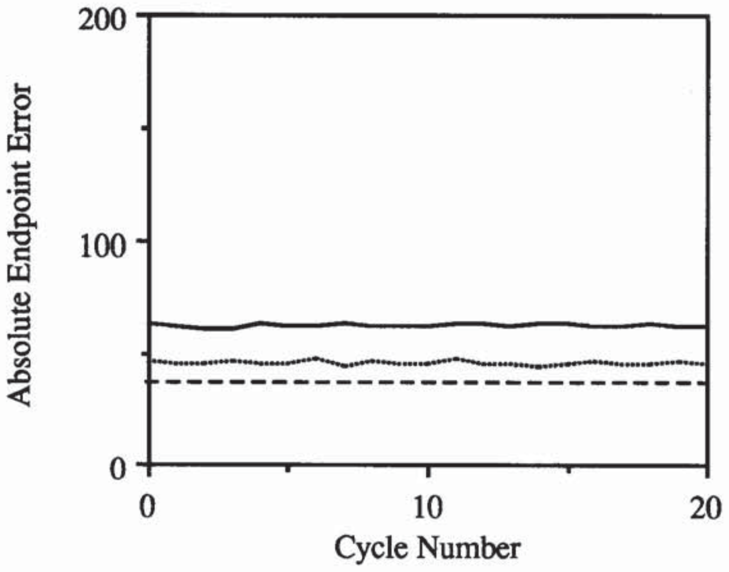


Figure 7.18 H∞ Controlled Varying System.

When the STR was activated the results shown in Figure 7.19 were obtained. These results clearly show, that for this design example, a significant improvement in terms of endpoint errors was achieved using the adaptation algorithm (which used constant performance and robustness specifications).

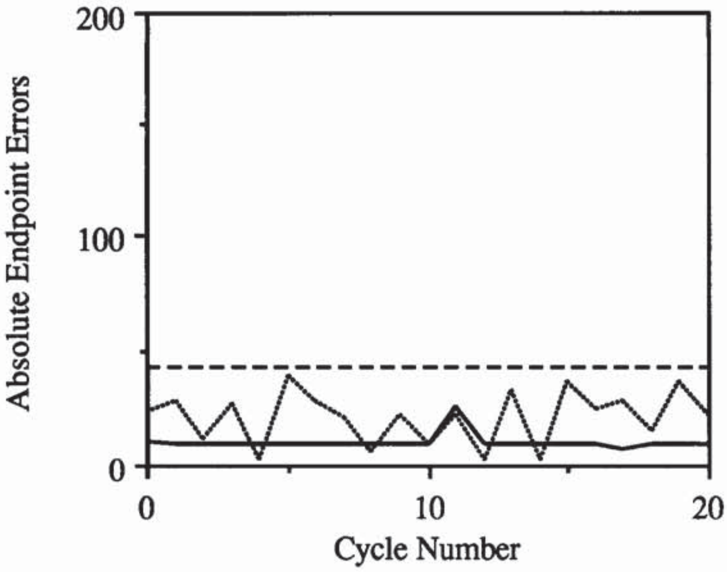


Figure 7.19 STR Controlled Varying System.

If the controller modification time is extended to 9ms the results seen in figure 7.20 are obtained where the specification is not met (This shows that the controller set-up is sensitive to any variation, i.e. a change in update time).

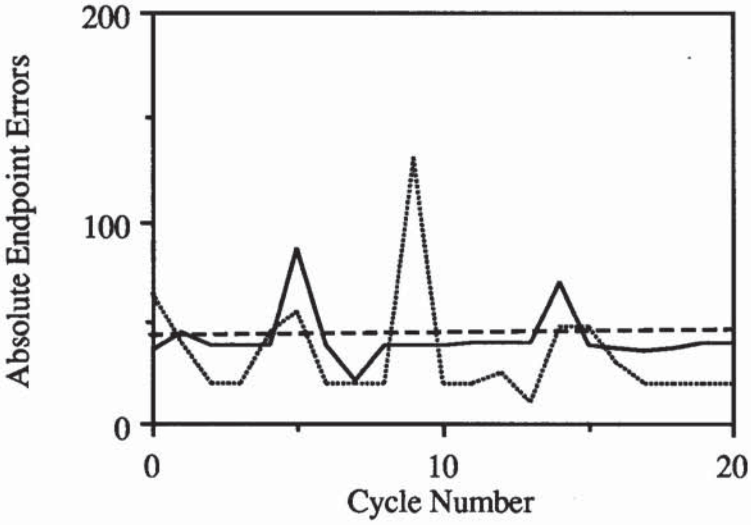


Figure 7.20 STR Controlled Varying System with 9ms Modification Time.

7.6.2 Practical Design Example.

The design example is as follows: an Electro-Craft™ BRU500 DM-50 drive module controlling an S-6100 motor (rotor inertia of 6.4×10^{-3} kgm^2) is required to rotate a non-symmetrical inertial load of 0.0137 kgm^2 forwards through 180° in 200ms and then return the load to its original position in a further 200ms. The endpoint error for each directional motion must not be greater than ±1.25°. The motor has a 8000 pulse per revolution encoder and each directional motion corresponds to moving through 4000 pulses in 200ms with a maximum endpoint error of ±27 pulses. The non-symmetrical load causes a ±0.3 Nm torque disturbance over each cycle of the shaft which is added to the inherent variation of the system (due to the non-linear nature of the servo systems).

Three different controllers were implemented: a PID with VFF, a H∞ controller and a STR controller. The results shown are absolute endpoint position error over 15 cycles. The PID with VFF controlled system did not met the specification, Figure 7.21. The H∞ controlled system did not meet the specification, Figure 7.22. The STR controlled system produced results which met the specification, Figure 7.23.

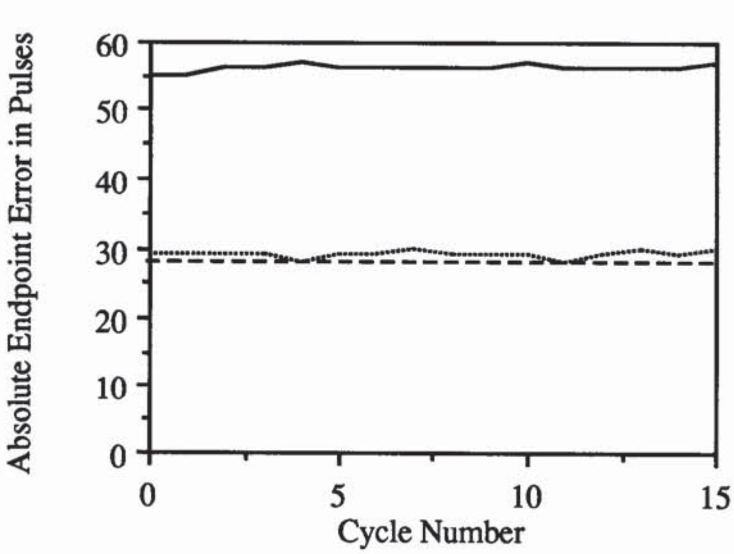


Figure 7.21 PID with VFF Controlled System.

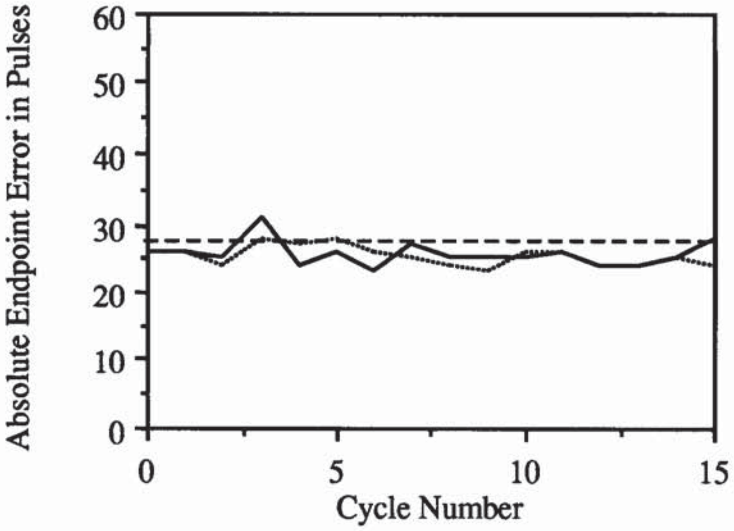


Figure 7.22 H∞ Controlled System.

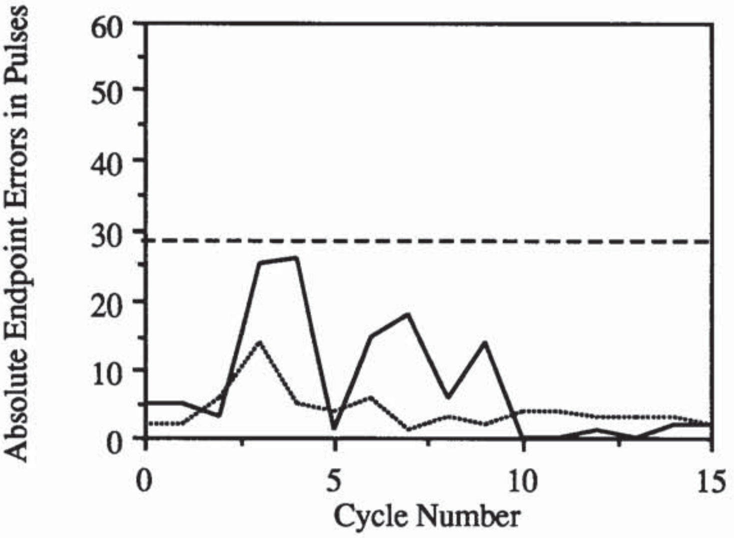


Figure 7.23 STR Controlled System.

7.7 CONCLUSIONS.

This chapter has detailed work completed on two forms of adaptive control system. The adaptive controllers have been demonstrated to yield improvements over both traditional PID with VFF and H∞ controllers. The SPGS controller produced a system with a 10.5% reduction in increment time in simulation and a 13.8% reduction in increment time on a practical system. However the SPGS controller technique is limited to systems with repetitive cycles and / or load variations. The STR has been demonstrated to give better results than a standard PID with VFF and a H∞ controller both on a simulated and real system. However the additional complexity of the STR, both in terms of computational requirement and difficulties in determining appropriate values for the identified model limits, FF and FFW values, means that its application will be severely limited.

CHAPTER 8.

CONTROLLER APPLICATION.

8.1. INTRODUCTION.

This chapter examines the application and limitations of the control strategies developed in this thesis. The controllers examined include PID with VFF, the standard industrial control algorithm for this application, H∞-norm optimisation, set-point gain scheduling, a three term time series controller and a self-tuning regulator.

Since the work of Seaward [1989] the use of independent drives in high speed machinery has become commonplace with all the benefits such an approach provides [Seaward and Vernon 1991]. The control of dc servo systems, used as independent drives, is still predominated by the traditional PID with Velocity Feed-Forward (VFF) controller. This chapter examines alternative methods of control emphasising the limitations and applicability of each control technique.

8.1.1 PID with VFF Control.

Proportional control is to used increase the closed-loop system bandwidth by feeding back into the plant a control signal which is proportional to the error between the desired and actual response. Integral control is routinely used in conjunction with proportional control to reduce the steady-state offset by including a proportion of the integral of the error into the control signal. The derivative term is used to increase the speed of response of the system by including in the control signal a term proportional to the rate of change of the error signal. The PID parameters can be determined using Ziegler-Nichols [Ziegler and Nichols 1942] tuning but usually an optimisation routine, based on an analysis of the closed-loop response, is required. The terms are usually optimised in the order proportional, derivative and integral. This process is repeated (i.e. each term is constantly readjusted until a satisfactory response is achieved) until the controller satisfies both transient and steady-state specifications, if possible.

VFF is used to 'drive' the system during demanded positional movements. VFF does not affect closed-loop stability and behaves like an external disturbance which enhances performance. To avoid overshoot a typical value for rapid incremental moves is 0.75, but to reduce steady-state positional tracking error it is often set to unity. The use of VFF has been found to dramatically increase the performance of dc servo systems [Seaward 1991].

The advantages of PID with VFF controllers are:

1) Both the theory and software have a long history and are well understood.
2) The controllers have been demonstrated to work on actual systems.

The disadvantages are:

1) There is a limited degree of flexibility in the controller.
2) The optimisation routine can produce controllers which are far from optimal.
3) A degree of experience is needed to tune the controller effectively .

8.1.2 H∞ Controller Synthesis.

The theory of H∞-norm optimisation is well known [Piché *et al.* 1991 and Francis 1987]. It has been shown to be applicable to dc servo systems (Chapter 4 and Liu and Liu [1991]). H∞ controller synthesis (as used un this thesis) utilises two frequency-dependent weighting functions, which correspond to a performance and robustness requirement, and a nominal model of the plant to attempt to determine an internally stabilising controller. If the controller exists it should yield both a nominal level of robustness and performance (determined by the selected weighting functions). H∞ controller synthesis conceptually has a number of advantages:

1) The technique offers a unified approach to the problems of stability and performance.
2) Determination of limited complexity controllers, which can be implemented on standard digital controller cards without any increase in the update time, is possible (An increase in the update time in general leads to a decrease in system performance [Morari and Zafiriou 1989]).
3) Simple guidelines exist for the determination of the weighting functions and the nominal model which combined with the readily available H∞ controller synthesis software means that the technique can be easily applied.
4) The controllers are inherently flexible in structure.
5) The technique uses available information, the nominal model etc., as a starting point to the controller design process.

The technique has a number of disadvantages:-

1) The theoretical background of the technique is complex.
2) The technique requires a nominal model which, in many industrial application, is difficult if not impossible to determine.
3) It is difficult to reconcile the nominal model and frequency dependent weights with the actual time response of the system.
4) Important limitations of practical systems, such as non-linearities or time delays, are omitted from the standard texts.
5) The technique requires a degree of experience to be applied effectively since an inappropriate model or weighting function selection can produce an undesirable system response.

8.1.3 Three Term Time-Series Controller.

The three term controller is detailed in chapter 6. The perceived advantages of this controller are:-

1) It is less complex than a PID controller and could lead to a reduction in the controller sample delays and hence lead to an increase in the system bandwidth (see chapter 3).
2) The controller is conceptually simple, with the controller being tuned in a similar fashion to a PID controller.
3) The controller is inherently flexible.

The disadvantages of this controller are:-

1) Stability criteria is not included in the design process (as detailed in this thesis).
2) The controller has no proven track record (i.e. not used by the author or industrial sponsor).

8.1.4. Set-Point Gain Scheduling.

Set-point gain scheduling is one of the simplest form of adaptive control. It is usually applied to time invariant or deterministic systems. The method utilises *a priori* knowledge to determine a number of controllers and switch points. Each controller is switched on-line at the appropriate point (see chapter 7). The principle is similar to that of forcing profiles or profile adaptation where the demand profile and hence the control signal is altered, depending on the point reached in the event cycle, in order to enhance performance.

The technique has a number of advantages:

1) In certain application it can reduce cycle times.

2) The technique is conceptually simple and easy to apply.

3) The technique has been shown to work on a real design example.

The disadvantages are:

1) The use of the technique for stochastic systems is severely limited.

2) Stability analysis is difficult.

3) In many applications it is difficult or impossible to implement different controllers and examine their time responses.

8.1.5. Self-Tuning-Regulator.

A more complex form of adaptive controller is the self-tuning regulator (STR) [Åström and Wittenmark 1989]. In this form of adaptive controller the adaptation problem is subdivided into two distinct parts [Tal 1989]: plant identification and controller modification (i.e. the parameter identifier supplies information on the state of the plant and the controller modification algorithm adjusts the controller function accordingly). The STR used in Chapter 7 is based on a combination of simplified variable forgetting factor least squares parameter identification and $H\infty$ controller synthesis algorithms.

The advantage of this control strategy is that a reduction in tracking error may be achieved for systems with varying dynamics.

The disadvantages of this control strategy are:

1) Inappropriate controller modification, due to incorrect identification or excessive time delays in the implementation of the controller, can lead to a degradation of the system performance.

2) The algorithm has no proven track record and due to its complexity limits the computing power available for other system requirements.

3) The identifier and controller modifier have a number of pre-set coefficients which are difficult to determine.

4) Stability analysis is difficult.

5) For the identifier to function correctly the system input signal must be multi-frequency.

8.2 GUIDELINES FOR CONTROLLER USE.

Experience has shown:

1) In 95% of applications satisfactory results can be obtained with standard PID with VFF controllers.

2) $H\infty$ controllers are suitable for systems that have lightly damped resonances because of their inherently more stable nature. Additionally the inherent flexibility of these controllers means that their frequency response can be "shaped" [McFarlene and Glover 1992] to attenuate the effects that any resonances may have on the closed-loop system.

3) SPGS controllers are applicable to systems with non-varying loads which have repetitive cycles. Their main advantage over other forms of adaptive control is that no on-line identification is required so the input signal does not have to be multi-frequency. Additionally the simplicity of these controllers makes them attractive for industrial applications.

4) STR control is applicable to systems with multi-frequency input signals (i.e. a series of steps or similar) and has been shown to produce improved results for a particular varying system.

8.3 CONCLUSIONS.

In this chapter the controller strategies examined in this thesis have been, very briefly, discussed. The applicability and limitations of each control system have been given (in terms of the design examples studied). The conclusion drawn is that each type of controller will be applicable in different situations (as common sense would suggest). The work completed in this chapter has been accepted for publication in the IFAC Journal, Control Engineering Practice [Beaven *et al.* 1995b].

CHAPTER 9

CONCLUSIONS AND FURTHER WORK

9.1 CONCLUSIONS.

This thesis has, for the first time, demonstrated the applicability of $H\infty$ controller synthesis to high speed independent drive systems. Real time digital control was realised using standard industrial control equipment. The thesis has covered a number of areas and has made the following contributions, in the author's opinion, to the control of high-speed independent drive systems:-

1) $H\infty$ controller synthesis has been applied to a previously unsolved industrial design example, both in a complex non-linear simulation and on a test rig. On a test rig the $H\infty$ controlled system had a cycle time 17% which was less than the "optimal" PID with VFF controlled system.

2) A direct-transform of a $H\infty$ controller has been determined for a simple class of plants and weighting functions. The direct-form $H\infty$ controller relates the plant and weighting function parameters to the controller parameters. The direct-form $H\infty$ controller requires 271 floating point operations (flops) while the standard algorithm required 13932 flops. The direct-transform $H\infty$ controller has allowed two separate strands of research to be completed.

    (i) The determination of an objective set of criteria, determined using endpoint errors, for the initial selection of the performance weighting function and the retuning of the performance weighting function to obtain a required level of closed-loop performance.

    (ii) A self-tuning regulator has been produced, based on recursive least squares parameters estimation with a variable forgetting factor and the direct transform H∞ controller. The STR's controller modification time is 8 ms (compared to 4 seconds for a STR implemented by Fairbairn and Grimble [1990]). The STR has been tested on an industrial design example (using standard industrial equipment) and it was shown to produce significantly less endpoint error than either a H∞ controller or a PID with VFF controller. Limits were placed on the identified model parameters (produced by the STR's identifier) to overcome system stability problems caused by controller modifier and identifier interaction instead of using an identifier filter (which performs a similar function).

4) A comparative study of the differences between a H∞ controller and a traditional PID controller has been completed, with the conclusion that the H∞ controller has an inherently more flexible structure. This is believed to be the reason for the improved control demonstrated by the H∞ controller.

Additionally the following have been examined :-

1) The theory and application of a SPGS controller to high speed independent drive examples has been described. The technique has been applied to an industrial design example and has shown a 10.5% reduction in increment time in simulation and a 13.8% reduction in increment time for a real system.
2) A three term time-series controller (a lead or lag compensator) has been suggested and the relationship to a traditional PID controller has been explained.
3) Two frequency domain modelling error terms were developed to allow an examination of different least squares parameter estimation strategies (in terms of cycle times, minimum forgetting factors and forgetting factor weights)

4) A review of the controller strategies used in this thesis has been given suggesting the applicability and limitations of each method.

9.2 FURTHER WORK.

In the authors opinion the following areas of research should be completed or extended :-

1) A form of adaptive controller should be developed based on the three term controller developed in Chapter 6. The adaptive controller should set up the initial controller and adapt the controller parameters on-line.
2) The direct-transform should be extended to higher order and unstable systems in order that a better understanding of H∞ controller synthesis can be gained.
3) The STR should be extended along the lines of (2) and should be developed so that the controller modification can also be adapted depending on the error state of the system.
4) The theory of SPGS should be developed using more and more complex controllers.
5) A review should be completed of the controllers used in this thesis and other possible control strategies, such as the Minimal Controller Synthesis (MCS [Benchoubane and Stoten 1990]), to be build up a bank of knowledge (an expert system) on the applicability and application of various control techniques to allow the design of high-speed independent drive controllers (or any system controller) to become standard instead of ad hoc as present.
6) Both the H∞ controller synthesis and SPGS controller could be reformulated to include VFF terms.

REFERENCES.

Åström, K.J. and Wittenmark, B., 1989, "Adaptive Control", Addison-Wesley Publishing Company, New York, U.S.A..

Bamieh, B.A. and Pearson, J.B., 1992, "A General Framework for Linear Periodic Systems with Applications to H∞ Sampled-Data Systems", IEEE Transactions on Automatic Control, Vol. 37, No. 4, April 1992.

Barber, N.T., 1984, "Benefits of Using Brushless Drives on High Performance Robots", 7th British Robot Association Ann. Conf., May, p95-105.

Beaven, R.W., Wilkes, L.D., Wright, M.T., Garvey, S.D. and Friswell, M.I., 1994a, "An Adaptive Control System Incorporating H∞ Concepts For High-Speed Machinery With Independent Drives", Proceedings I.Mech.E., Part I, Journal of Systems and Control Engineering, Vol 208, pp43-52.

Beaven, R.W., Wilkes, L.D., Wright, M.T., Garvey, S.D. and Friswell, M.I., 1994b, "The Application Of Setpoint Gain Scheduling To Improve The Performance Of High Speed Independent Drives", Presented at the IEE Colloquium on Precision Motion Control, Savoy Place, London, 22nd November 1994.

Beaven, R.W., Wilkes, L.D., Wright, M.T., Garvey, S.D. and Friswell, M.I., 1995a, "The Application of Setpoint Gain Scheduling to High-Speed Independent Drives", Control Engineering Practice, A Journal of IFAC, The International Federation of Automatic Control, Vol. 3, No. 8, pp 1059-1065.

Beaven, R.W., Wilkes, L.D., Wright, M.T., Garvey, S.D. and Friswell, M.I., 1995b, "Application Of Four Control Strategies To High-Speed Independent Drive Systems", Control Engineering Practice, A Journal of IFAC, The International Federation of Automatic Control, Vol. 3, No. 11, pp 1581-1585.

Benchoubane, H. and Stoten, D.P., 1990, "Convergence Rates Of An Adaptive Control Algorithm With Application To The Speed Control Of A DC Machine", IECON'90, 16th Annual Conference of IEEE Electronics, 27-30 Nov. 1990, Part 1, pp 390-395.

Bitmead, R.R., Gevers, M. and Wertz, V., 1990, "Adaptive Optimal Control: The Thinking Man's GPC", Prentice Hall, Victoria, Austrailia.

Bode, H.W., 1947, Network Analysis and Feedback Amplifier Design, Fourth Printing, D. Van Nostrand Company Inc., New York, U.S.A.

Char, B.W., Geddes, K.O., Gonnet, G.H., Leong, B.L., Monagan, M.B. and Watt, S.M., 1991,MapleV Library Reference Manual, Springer-Verlag, New York, U.S.A.

Chen, M.J. and Desoer, C.A., 1982, "Necessary and Sufficient Conditions for Robust Stability of Linear Distributed Feedback Systems", Int. J. Control, Vol. 35, No. 2, p255-267.

Chen, M.J. and Francis, B.A., 1991, "Linear Time-varying H_2-Optimal Control of Sampled-Data Systems", Automatica, Vol. 27, No. 6, pp963-974.

Chiang, R.Y. and Safonov, M.G., 1992, "Robust Control Toolbox", The Maths Work Inc., Published by The Maths Works, Inc., Massachusetts 01760, U.S.A.

Doyle, J.C., 1982, "Analysis of Feedback Systems With Structured Uncertainty", IEE Proc., Pt. D, Vol. 129, No. 6, pp. 242-250.

Doyle, J.C. and Stein, G., 1981, "Multivariable Feedback Design: Concepts For A Classical/ Modern Synthesis", IEEE Trans. Aut. Contr., Vol. 26, No. 1, pp4-16.

Doyle, J.C., Glover, K., Khargonekar, P and Francis, B., 1989, "State-Space Solutions to Standard H_2 and H-infinity control problems", IEEE Transaction on Automatic Control, Vol 34, No. 8, pp831-847.

Electro-Craft, 1987, "BRU-200 Servo Drives Instruction Manual", Published by Reliance Electric, Crewe, U.K..

Fairbairn, N.A. and Grimble, M.J., 1990, "H_∞ Robust Controller for Self-Tuning Control Applications Part 3. Self-tuning Implementation", International Journal of Control, Vol. 46, No. 5, pp 1819-1840.

Foo, Y.K., 1985, "Robustness of Multivariable Feedback Systems: Analysis and Optimal Design", PhD Thesis, Oxford University, Oxford, England.

Fortescue, T.R., Kershenbaum, L.S., Ydstie, B.E., 1981, " Implementation of self-tuning regulators with variable forgetting-factors ", Automatica, Vol. 17, No 6, pp 831-835.

Francis, B.A., 1987, A Course in H_∞ Control Theory, Springer-Verlag, Berlin, Germany.

Francis, B. A., 1990, "Lecture on H_∞ - control and sampled data systems", In H_∞ control theory, Lecture Notes in Mathematics 1496, Eds E. Mosca and L. Pandolfi, 1990, (Springer - Verlag, Berlin) pp 37-105.

Fraleigh, J.B. and Beauregard, R.A., 1987, "Linear Algebra", Addison-Wesley Publishing Company, Massachusetts, U.S.A..

Gellart, W. , Kustner, H. , Hellwich. M. and Kastner. H., 1977, Concise Encyclopaedia of Mathematics, Van Nostrand Reinhold Company, New York.

Glover, K. and Doyle, J.C. State-space formulae for all stabilising controllers that satisfy an H∞-norm bound and relations to risk sensitivity. Systems and Control Letters, 1988, 11, 167-172.

Golten, J. and Verwer, A., 1991, "Control System Design and Simulation", McGraw-Hill Book Company (UK) Ltd., London, U.K.

Grimble, M.J., 1987a, "H∞ Robust Controller for Self-Tuning Control Applications Part 1. Controller Design", International Journal of Control, Vol. 46, No. 4, pp 1429-1444.

Grimble, M.J., 1987b, "H∞ Robust Controller for Self-Tuning Control Applications Part 2. Self-tuning and Robustness", International Journal of Control, Vol. 46, No. 5, pp 1819-1840.

Gu, D.W., Tsai, M.C., O'Young, S.D. and Postlethwaite, I., 1989, "State-space Formulae for Discrete-time H∞ Optimisation", Int. J. Control, Vol. 52, No. 1, pp 15 - 36.

Hashim, R.A. and Grimble, M.J., 1990, "An Implicit Quadratic H∞ Self-Tuning Controller", Proceedings of the 1990 American Control Conference, May23-25 1990, Calofornia, U.S.A., pp1088-1090.

Han, K.C. and Hsia, T.C., 1990, “On Reducing Compensator Bandwidth of LQG/ LTR Control: An H∞-Optimisation Approach”, Proceedings of the 1990 American Control Conference, May 23-25, Vol. 1, pp924-929, Calofornia, U.S.A..

Hsia, T.C. System Identification, Least-Squares Methods. 1977, Lexington Books, London, England.

Hyde, R.A. and Glover, K., 1990, "VSTOL Aircraft Flight Control System Design Using H∞ Controllers And A Switching Strategy", Proceedings of the 20th IEEE Conference on Decision and Control, Honolulu Hawaii, December 1990, pp 2975-2980.

Hvostov, H.S., 1990, "Simplifying H∞ Controller Synthesis Via Classical Feedback System Structure", IEEE Transactions on Automatic Control, Vol.35., No 4, April 1990, p485-488.

Iglesias, P.A. and Glover, K., 1990, “State-Space Solutions to H∞ Control Problems at Optimality: Full Information Problem”, Proceedings of the 29th Conference on Decision and Control, Honolulu, Hawaii, pp1032-1033.

Isermann, R., Baur, U., Bamberger, W., Kneppo, P. and Siebert, H. Comparison and evaluation of six on-line identification and parameter estimation methods with three simulated processes. 1973, Third IFAC Symposium on Identification, The Hague, paper E-1, pp. 1061-1080.

Hewlett Packard, 1989, "Fundamentals of the z-Domain and Mixed Analog/Digital Measurements", Application Note 243-4, Published by Hewlett Packard Co., Geneva, Switzerland.

Kusko, A and Peeran, S.M., 1988, "Definition of the Brushless dc Motor", Conference Records of the 23rd IEEE Industry Applications Society Annual Meeting, Pittsburgh: IEEE Industry Applications Society, pp 20-22.

Kwakernaak, H., 1993, "Robust Control and H∞-Optimization - Tutorial Paper", Automatica, Vol. 29, No 2, pp255-274.

Lamanna De R, R., Padilla, R.A., Uria De C, M., 1981, " On-line selection and parameter estimation - an experimental application ", IEEE Control Systems, Vol. 1, No 3, pp 6-14.

Laub, A.J., 1979, "A Schur Method For Solving Algebraic Riccati Equations", IEEE Trans. Autom. Control, AC-24, pp. 913-921.

Lehtomaki, N.A., Sandell, N.R.Jr, and Athans, M. "Robustness results in Linear-Quadratic Gaussian based multivariable control designs", IEEE Transactions on Automatic Control, 1981, AC-26(1), 75-92.

Lewis, F.L., Abdallah, C.T. and Dawson, D.M., 1993, "Control of Robot Manipulators", Macmillan Publishing Company, New York, pp311-318.

Limbeer, D.J.N. and Kasenally, E., 1986, "H∞ optimal control of a synchronous turbo-generator" in Proc. IEEE 25th Conf. Decision Contr. (Athens, Greece), Dec. 1986, pp62-65.

Liu Tian-Hua and Liu Chang-Huan, 1990, "Implementation of AC Servo Controllers Employing Frequency-Domain Optimisation Techniques", IEEE Transaction on Industrial Electronics, Vol. 37, No 4, August 1990.

Ljung, L. System identification: theory for the user. 1987, Prentice-Hall International, New Jersey, U.S.A..

Loh, A.P., 1986, "Uncertainty Estimation and Multivariable Control System Design", PhD Thesis, Oxford University, Oxford, U.K.

Lundström, P., Slogestad, S. and Wang, Z-Q., 1991, "Performance Weight Selection For H-infinity and μ-control Methods", Transaction Institute of Measurement and Control, Vol. 13 No 5, pp241-252.

Lynn, P.A. and Fuerst, W., 1993, "Introductory Digital Signal Processing with Computer Applications", John Wiley & Sons Ltd., Coventry, U.K.

Maciejowski, J.M., 1993, Multivariable Feedback Design, Addison-Wesley Publishing Company, Wokingham, U.K.

McFarlene, D., 1988, "Robust Controller Design Using Normalised Coprime Factor Plant Descriptions", PhD Thesis, Queens College Cambridge, Cambridge University, England.

McFarlene, D. and Glover, K., 1992, "A Loop Shaping Design Procedure Using H∞ Synthesis", IEEE Transactions on Automatic Control, Vol. 37, No. 6.

McPherson, G. and Laramore, R.D., 1990, An Introduction to Electrical Machines and Transformers, 2nd Edition, John Wiley and Sons Inc, Singapore.

Mitchell and Gauthier, 1991, Advanced Continuous Linear Simulation Language, Refernce Manual, Edition 10, Published by Mitchell and Gauthier Associates (MGA) Inc., Concord MA01742, U.S.A.

Morari, M. and Zafiriou, E. Robust Process Control, 1989, Prentice-Hall International Editions, New York, U.S.A.

Newton, G.C., Gould, L.A., Kaiser, J.F., 1957, Analytical Design of Linear Feedback Controls, John Wiley and Sons Inc., New York, U.S.A.

Nyquist, H., 1932, "Regeneration Theory", Bell Systems Technical Journal, No. 11, pp126-147.

Parrott, S., 1978, "On A Quotient Norm and the Sz.-Nagy-Foias Lifting Theorem", Journal of Functional Analysis, Vol. 30, pp.311-328.

Piché, R., Pohlolainen, S. and Virvalo, T., 1991, "Design of Robust Controllers For Position Servos Using H-infinity Theory", I.Mech.E. Part I, Journal of Systems and Control Engineering, Vol 205, No. 14, pp299-307.

Postlethwaite, I., D-W. Gu., S.D.Young and M.S. Tombs, 1986, "Industrial Control System Design Using H∞ optimisation", Proc. IEEE 25th Conf. Decision Contr. (Athens, Greece), Dec. 1986, pp12-13.

Postlethwaite, I., Tsai, M.C., Gu, D.-W., 1989, "State-Space Approach to Discrete-Time Super-Optimal H∞ Control Problems", International Journal of Control, Vol. 49, No. 1, pp 247-268.

Raven, F.H., 1987, Automatic Control Engineering: Fourth Edition, McGraw-Hill International Editions, New York, U.S.A..

Safonov, M.G., 1980, “Stability and Robustness of Multivariable Feedback Systems” MIT Press, Massachusetts, U.S.A.

Safonov, M.G., Chiang, R.Y. and Flashner, H., 1988, "H_∞ robust control synthesis for a large space structure", American Control Conference, Georgia, USA, June 1988, pp. 2038-2043.

Safonov, M.G., Limbeer, D.J.N. and Chiang, R.Y., 1989, "Simplifying the H_∞ Theory Via Loop-shifting, Matrix-Pencil and Descriptor Concepts.", Int. J. Control, Vol 50, No. 6. pp2467-2488.

Seaward, D. R., 1989, "Continuous Phase Synchronised Drives (For A Rod Making Machine)", PhD Thesis, The University of Aston in Birmingham, Birmingham, England.

Seaward, D.R. and Vernon, G.W., 1991, "Using High Performance drives Technology in High-Speed Machinery - An Original Equipment Manufacture's Point of View", Eurotech Direct 91, I.Mech.E, pp73-78.

Seaward, D. R., 1993, "Study of Independently Driven Cold Forming Machine", Internal Company Report, Molins Plc, Coventry, U.K.

Sefton, J. and Glover, K., 1990, "Pole / Zero Cancellation In The General H_∞ Problem With Reference To A Two Block Design", Systems and Control Letters, Vol. 14, pp 295-306.

Soderstrom, T., Stoica, P., 1989, " System Identification ", Prentice Hall International (UK) Ltd., London, U.K.

Soderstrom, T., Ljung, L., Gustavsson, I., 1976, "A theoretical analysis of recursive identification methods", Automatica, Vol. 14, pp 231-244

Stoorgovel, A., 1992, "The H∞ Control Problem: A State-Space Approach", Prentice Hall Series in Systems and Control Engineering, London, U.K..

Sripada, N.R., Fisher, D.G., 1987, " Improved least squares identification for adaptive controllers ", Proceedings of the 1987 American Control Conference, June 1987.

Tal, J. and Baron. W, 1987, "Intelligent Motion Controllers with Variables - A New Concept", Power Conversion and Intelligent Motion, Vol. 13, pp36-45.

Tal, J., 1989, Motion Control Applications, Published by Galil Motion Control, 575 Maude Court, Sunnyvale, CA 94086, U.S.A.

Tomasek, J., 1989, "Structural and Performance Fundamentals of Sinewave-Controlled Brushless Servo Systems", IEEE Applied Electronics Conference (APEC '89) in Baltimore, Maryland, USA, March 1989.

Themis. TSVME440 axis controller board DOCA246-03 and TSVME servo drive software A285-04 card manual. 1988, Published by Themis, 38610 Gieres, France.

Tian-Hua Liu and Chang-Huan Liu, 1990, "Implementation of ac Servo Controllers Employing Frequency-Domain Optimization Techniques", IEEE Transaction on Industrial Electronics, Vol. 37., No 4, August 1990, p275-282.

Tsai, M.C., Geddes, E.J.M and Postlethwaite, I., 1992, "Pole-Zero Cancellation and Closed-Loop Properties of an H∞ Mixed Sensitivity Design Problem", Automatica, Vol. 28 , No 3, pp519-531.

Vidyasagar, M., 1985, “Control Synthesis: A Coprime Factorization Approach”, MIT Press, London, England.

Wilkes, L.D., 1995, "Comparison of Identification Methods For High-Speed Independent Drive Systems" Research Thesis, The University of Aston in Birmingham, Birmingham, England.

Wolfram, S., 1991, Mathematica: A System for Doing Mathematics by Computer, 2nd Edition, Addision-Wesley Publishing Company, Reading, Massachusetts, U.S.A..

Zames, G., 1981, "Feedback of Optimal Sensitivity: Model Reference Transformations, Multiplicative Semi-norms, and Approximate Inverses", IEEE Trans. AC, Vol. AC-26, 1981, pp301-320.

Zames, G. and Francis, B.A., 1983, "Feedback Minimax Sensitivity and Optimal Robustness", IEEE Trans. Auto. Contr., Vol AC-28, pp585-601.

Zarrop, M.B., 1983, "Variable forgetting-factors in parameter estimation", Automatica, Vol. 19, No 3, pp 295-298.

Ziegler, J.G. and Nichols, N.B., 1942, "Optimum Settings for Automatic Controllers", Transactions of the A.S.M.E., November 1942, 64, pp759-768.

APPENDIX A

Proof of Stability for Multiplicative Uncertainty

For the sake of simplicity W_2 is considered to be unity for all frequencies. If it is assumed that the nominal plant G_0 and the perturbed plant G have the same number of unstable poles and that the compensated system GK is stable. Then the closed loop system will remain stable provided the number of encirclements of -1 by the characteristic loci of GK, K being the controller, remains unchanged, which it will if no locus passes through -1 as G varies. In other words

$$\det[I + G(jw)K(jw)] \neq 0 \qquad \text{(A1)}$$

which is equivalent to

$$\underline{\sigma}[I + G(jw)K(jw)] > 0 \qquad \text{(A2)}$$

for all ω where $\underline{\sigma}$ is used to indicate the smallest singular value. Thus

$$\underline{\sigma}[I + G_0K_0 + \Delta_0 G_0 K] > 0 \qquad \text{(A3)}$$

and factorising this gives

$$\underline{\sigma}\{[(\Delta_0 G_0 K)^{-1} + \Delta_0^{-1} + I]\, \Delta_0 G_0 K] > 0 \qquad \text{(A4)}$$

The above equation holds if

$$\underline{\sigma}\{[(G_0K)^{-1} + I]\, \Delta_0^{-1} + I\,] > 0 \qquad \text{(A5)}$$

and this in turn is true if

$$\underline{\sigma}\{[(G_0K)^{-1} + I]\,\Delta_0^{-1}\} > 1 \tag{A6}$$

which is equivalent to

$$\underline{\sigma}(\Delta_0^{-1})\underline{\sigma}[(G_0K)^{-1} + I] > 1 \tag{A7}$$

Hence

$$\bar{\sigma}(\Delta_0^{-1})\bar{\sigma}\{[(G_0K)^{-1} + I]^{-1}\} < 1 \tag{A8}$$

or

$$\bar{\sigma}\{G_0K[(G_0K) + I]^{-1}\} < \frac{1}{\bar{\sigma}(\Delta_0^{-1})} \tag{A9}$$

APPENDIX B

RESUME OF MOTOR / DRIVE SIZING AND OPTIMAL GEAR RATIO.

This appendix contains a resume on the theory of motor / drive sizing (selection) and the determination of the optimal gear ratio, the reader is directed to Tal [1989] for a full explanation of these theories.

B1. Motor / Drive Selection

Motor selection involves two important parameters, the required continuous and peak torque values. The peak torque is the highest torque required (usually required during periods of acceleration). The continuous torque is the level of torque that the motor must supply continuously without overheating; it is needed to overcome the friction and to drive the load. If the system operates in a cyclic manner, where the torque varies periodically, the required continuous torque is the average torque value or more precisely the RMS (root mean square) value of the torque.

To determine the size of motor required the designer should ideally know the following parameters :-

1) Moment of inertia of the load - J_l
2) Moment of inertia of the motor - J_m
3) Maximum acceleration rate - $d\omega/dt$.
4) Friction torque - T_f

The peak torque, T_p, can be calculated from

$$T_p = (J_m + J_l).d\omega/dt + T_f \qquad \text{(B1)}$$

The continuous torque, T_c, can be computed as the RMS value of the required torque. Often the continuous torque equals the friction or gravatational torque. The reader is directed to Tal [1989] for methods of directly measuring the peak and continuous torque levels.

Once the required torque levels have been determined a motor / drive combination can be selected. The important features of the motor are the torque constant, K_t, the peak torque rating (the maximum torque the motor can produce) and its continuous torque rating (the torque the motor can produce continuously). The important features of the drive are its peak current and continuous current ratings. The drive must be able to supply the motor with appropriate continuous current and peak currents. The relationship between the motor current, I, and torque, T_g, is given by the torque constant, i.e. the torque produced is the current times the torque constant

$$T_g = K_t I \tag{B2}$$

Therefore it can easily be determined if the motor / drive combination will be able to produce the continuous and peak torque values required.

Besides torque capabilities and torque constant, size and weight limitations, as well as speed and reliability requirements are important. Such considerations also influence the choice of motor and drive.

Optimal Gear Ratio

Often the designer has freedom in selecting the coupling between the motor and the load. The motor may be geared down by a gear box, a timing belt or by other means. The selection of the optimal gear ratio is as follows.

If the motor speed is geared down by a factor of N times. The gear reduction increases the motor output torque and equivalently reduces the values of the load inertia and torque as reflected to the motor. The disadvantage of gearing is that the motor must travel N times further than the of the load. Seaward [1989] has shown that the optimal gear ratio, that will allow accelerating a given load at a specified rate while requiring minimum torque, is that which causes the reflected inertia of the load to equal the inertia of the motor. That is if J_l is the load inertia and J_m is the motor inertia the optimal gear ratio is

$$N = \sqrt{\frac{J_l}{J_m}} \tag{B3}$$

While the gear reduction improves the power efficiency of the system, it may introduce significant backlash that can lead to position errors or oscillations. Therefore, it is often desirable to use direct drive, although it may reduce the system efficiency.

APPENDIX C

```
"PROGRAM BRU500 SIMULATION OF PROPORTIONAL INTEGRAL"
"DERIVATIVE CONTROL by R.W.Beaven (Based on the research and"
"computer simulations completed by Dr D.R.Seaward 1992). "
"SIMULATION OF THE ELECTROCRAFT BRU500 SERIES OF BRUSHLESS"
"MACHINES TO BE USED WITH ITS DATA FILE FOR ENKOTEC PROBLEM"
"INCLUDES THE LM628 CONTROLLER MODIFIED WITH OPTIONAL"
"VELOCITY AND/OR CURRENT FEEDFORWARD"

"THIS PROGRAM SIMULATES ELECTROCRAFT PROPORTION /INTEGRAL"
"/ DERIVATIVE SOFTWARE SCHEME. "

"THE SWITCHES SHORT OUT A BLOCK WHEN SET TO ONE"
"THE SWITCH NUMBER REFERS TO THE VARIABLE IT SWITCHES"
"IE. S2 SWITCHES OUT THE BLOCK BETWEEN Y1 AND Y2"

"DEFAULT COMMUNICATION INTERVAL"
CINTERVAL CINT = 0.001
"DEFAULT MAXIMUM STEP LENGTH"
MAXTERVAL MAXT = 5E-5
"DEFAULT NUMBER OF INTEGRATION STEPS PER COMMUNICATION
INTERVAL"
NSTEPS NSTP = 1

"DEFAULT ALGORITHM IS RUNGE-KUTTA FOURTH ORDER"
"IALG = 1 IS ADAMS-MOULTON, VARIABLE STEP, VARIABLE ORDER"
"IALG = 2 IS GEARS STIFF, VARIABLE STEP, VARIABLE ORDER"
"IALG = 3 IS RUNGE KUTTA FIRST ORDER OR EULER"
"IALG = 4 IS RUNGE KUTTA SECOND ORDER"
"IALG = 5 IS RUNGE KUTTA FOURTH ORDER"
"IALG = 7 IS USER SUPPLIED ROUTINE"
"IALG = 8 IS RUNGE-KUTTA-FEHLBERG SECOND ORDER"
"IALG = 8 IS RUNGE-KUTTA-FEHLBERG FIFTH ORDER"
ALGORITHM IALG = 5
```

```
INITIAL

"INITIAL SECTION REQUIRED TO RUN THIS PROGRAM IN ISOLATION"
JEQ = JMOT + JLOAD

"INITIAL SECTION REQUIRED BY THIS PROGRAM"
"FEEDBACK SCALED IN PULSES"
KENC = 2*ENC/PI
K2ENC = 2*ENC2/PI
"SOFTWARE INPUT FILTER"
TFPLE = 1/(2*PI*FBAND)

VELFB     = 0
VELDEM    = 0
PERROR    = 0
DELTA     = 0
DIST      = 0
VEL       = 0
X         = 0
LMIN      = 0
LMZ1      = 0
LMZ2      = 0
LMZ3      = 0
LMZ4      = 0
LMZ5      = 0
LMZ6      = 0
LMZ7      = 0
LMZ8      = 0
PIDZ7     = 0
PIDZ1     = 0
PIDZ2     = 0
PIDZ3     = 0
PIDZ4     = 0
PIDZ5     = 0
PIDZ6     = 0
PIDZ7     = 0
PIDZ8     = 0
```

```
PIDZ9       = 0
PIDZ11      = 0
PIDZ12      = 0
PIDZ13      = 0
PIDZ14      = 0
PIDZ15      = 0
Y1          = 0
IINPUT      = 0
IFEED       = 0
ONPOS       = 0
LT          = 0
LT1         = 0
LT2         = 0
YT          = 0
YT1         = 0
YT2         = 0
YTP1        = 0
YTP2        = 0
NOISE       = 0

END $"OF INITIAL"

DERIVATIVE

" ADDITIONAL SECTION TO OPEN FILES TO SAVE DATA"
IF(T.LE.0.001)OPEN(9,FILE='LMZ1')
IF(T.LE.0.001)REWIND(9)
IF(T.LE.0.001)OPEN(10,file='Y')
IF(T.LE.0.001)REWIND(10)

"MOTOR/ LOAD SECTION CONVERTS 0-10V INTO A VELOCITY"
"DEFAULT IS S6300 & DM150 SERVO WIH LARGEST MATRIX LOAD"
CONSTANT JLOAD = 0.0066, KT = 0.7, DM = 150, JMOT = 0.0034, VISC=0.022,
CONSTANT FRICT = 3.0 , TRAN2=5.9E-07,TRAN1 =7.6E-07

"CURRENT FEEDFORWARD SECTION - NOT USED"
"DUM1 = BOUND(-ILIM,ILIM,IINPUT*DM/10 + IFEED)"
```

```
DUM1 = BOUND(-ILIM,ILIM,IINPUT*DM/10)
TORQ1 = KT*DUM1

PROCEDURAL(TORQ=TORQ1,FRICT)
TORQ = TORQ1 - SIGN(FRICT,TORQ1)
IF(ABS(TORQ1).LE.FRICT)TORQ=0
END

DUM2 = 1/VISC*REALPL(JEQ/VISC,TORQ,0.0)
OMEGA = CMPXPL(TRAN2,TRAN1,DUM2,0.0,0.0)

"POSITION IN RADIANS"
THETA = INTEG(OMEGA,0.0)

"ALL CONSTANTS REQUIRED BY THIS PROGRAM"
CONSTANT PI = 3.141593, RADRPM = 9.55, TSTOP = 0.1, ILIM = 150

"DEFAULT SWITCH SETTINGS"
CONSTANT XS1 = 0.0, XS2 = 0.0, XS3 = 0.0, XS4 = 0.0, S1 = 0.0, S2 = 0.0

"DEFAULT MOTOR CONSTANTS FOR BRU S-6300 MOTOR AND DM-150"
CONSTANT ENC = 5000, INPLE = 0.0008, MOT = 6300
CONSTANT INLIM = 10.0, QUANT = 1

"SERVO SET UP CHARACTERISTICS DEFAULT"
CONSTANT PGAIN = 200, IGAIN = 0, FBAND = 300
CONSTANT VSCALE = 240, DGAIN = 0, ww = 6.28

"LM628 SERVO CONSTANTS"
CONSTANT KD = 10, KP = 8, KI = 22, KILIM = 12, ENC2 = 4500
CONSTANT KFF = 87, KIFF = 0, ACCUR = 5, GEAR = 1

"HINF CONTROLLER CONSTANTS"
CONSTANT AD = 0.0093, BD = -0.0135, CD = 0.0042, ED = -0.0156, FD = -0.9844

"SIGNAL GENERATION FOR CLOSED-LOOP FREQUENCY RESPONSE"
"AND IDENTIFICATION"
NOISE = OU(10000,0,0.1)
```

```
"INPUT POLE DUE TO ANALOGUE FILTERING"
"LMZ7 IS PID WITH VFF CONTROLLER, YT IS HINF CONTROLLER"
"Y1 = RSW(S1.EQ.1,LMZ7,REALPL(INPLE,LMZ7,0.0))"
Y1 = RSW(S1.EQ.1,YT,REALPL(INPLE,YT,0.0))

"INPUT VOLTAGE LIMIT..USUALLY 10VOLTS"
Y2 = RSW(S2.EQ.1,Y1,BOUND(-INLIM,INLIM,Y1))

"VOLTAGE SCALING INTO R/MIN"
Y3 = VSCALE*Y2

"INPUT IN PULSES"
Y4 = 4*ENC/60*Y3

"CURRENT LOOP INPUT DELIVERED FROM DIGITAL ALGORITHM"
IINPUT = 10*REALPL(TFPLE,PIDZ15,0.0)/32768

"POSITION IN PULSES"
ENCODE = KENC*THETA
LMIN = K2ENC*THETA/GEAR

"VARIABLES OF INTEREST"
"OUTPUT SCALED IN RPM FROM DIGITAL TACHO"
YRPM = OMEGA*RADRPM
"INPUT TO DIGITAL SECTION IN PULSES"
DEMAND = Y4 TERMT(T.GE.TSTOP)
END $"OF DERIVATIVE"

DISCRETE CNTRL
INTERVAL SMP=0.001
PROCEDURAL
PIDZ1 = QNTZR(QUANT,ENCODE)
PIDZ3=PIDZ1 -PIDZ2
PIDZ5 = 4*PIDZ3
PIDZ4 = 256*PIDZ5
PIDZ7 = PIDZ5 - PIDZ6
```

```
PIDZ8 = PIDZ4 + 4*DGAIN*PIDZ7
PIDZ9 = QNTZR(QUANT,DEMAND*SMP*256*4)
PIDZ10 = 40*(PIDZ9 - PIDZ8)
PIDZ11 = 4*PGAIN*PIDZ10/65536
PIDZ12 = IGAIN*PIDZ10/65536
PIDZ13 = PIDZ12 + PIDZ14
"THIS ALGORITHM NEES INTEGRAL WINDUP PROTECTION"
"USE AN IF PIDZ13 GREATER THAN JUMP"
PIDZ15 = PIDZ11 +PIDZ13
PIDZ2 = PIDZ1
PIDZ6 = PIDZ7
PIDZ14 = PIDZ13
END $"OF PROCEDURAL"

"END OF PROPORTIONAL INTEGRAL ALGORITHM MAIN PROGRAM"

END $"OF DISCRETE"

DISCRETE LM628
INTERVAL LMSMP=0.0003343
PROCEDURAL

"DETERMINATION OF POSITION ERROR INPULSES"
LMZ1 = QNTZR(QUANT,X)- QNTZR(QUANT,LMIN)

"SIGNAL GENERATION FOR OPEN-LOOP BODE PLOT"
"LMZ1 = 100*SIN(WW*T)"

LT2   = LT1
LT1   = LT
LT    = LMZ1
YT2   = YT1
YT1   = YT

"GENERALISED HINF CONTROLLER"
YTP1 = AD*LMZ1 + BD*LT1 + CD*LT2
YTP2 = -ED*YT1 - FD*YT2
```

```
YT = YTP1 + YTP2

ONPOS=0 IF(ABS(LMZ1).LE.ACCUR)ONPOS=1

"PID WITH VFF CONTROLLER"
LMZ2 = KP*10*LMZ1/32768
LMZ3 = (KD*10/32768)*(LMZ1-LMZ4)
LMZ5 = KI*10*LMZ1/8388608 + LMZ6
LMZ6 = BOUND(-KILIM/1000,KILIM/1000,LMZ5)
LMZ4 = LMZ1
"QUICK FUDGE TO PUT CORRECT MOTION PROFILE IN"
IF(DELTA.LE.60) DIST = DELTA*DELTA
IF(DELTA.GT.60) DIST = 7200 - (120-DELTA)*(120-DELTA)
IF(DELTA.GE.120)DIST = 7200
IF(DELTA.GE.450)DIST = 7200 + (DELTA-449)*(DELTA-449)
IF(DELTA.GT.510)DIST = 14400 - (569-DELTA)*(569-DELTA)
IF(DELTA.GE.570)DIST = 14400
IF(DELTA.GE.898)DIST = 14400 + (DELTA-897)*(DELTA-897)
IF(DELTA.GT.958)DIST = 21600 - (1017-DELTA)*(1017-DELTA)
IF(DELTA.GE.1018)DIST= 21600

"QUICK FUDGE TO PUT CURRENT FEEDFORWARD IN"
IF(DELTA.LE.60) IFEED = KIFF
IF(DELTA.GT.60) IFEED = -KIFF
IF(DELTA.GE.120)IFEED = 0

" DATA STORAGE SECTION"
WRITE(9,11) (LMZ1)
11..FORMAT(f16.6)
WRITE(10,12) (LMIN)
12..FORMAT(f16.6)

VEL = DIST - X
X = DIST
DELTA = DELTA + 1
"VELOCITY FEEDFORWARD TERM ADDED"
LMZ7 = BOUND(-INLIM,INLIM,(LMZ2 + LMZ3 + LMZ6 +VEL*KFF*10/32768))
END $"OF PROCEDURAL"
```

```
"END OF PROPORTIONAL INTEGRAL ALGORITHM MAIN PROGRAM"

END $"OF DISCRETE"

END $"OF PROGRAM"
```

APPENDIX D

```
/*
* A C PROGRAM TO CALCULATE THE OPTIMAL PARAMETERS OF
* A WEIGHTING FUNCTION GIVEN THE VALUES OF THE PLANT
* AND THE MAJORITY OF THE WEIGHTING FUNCTION PARAMETERS
*/

double p,q,q1,q2,r,a,b,AA,BB,nnn,ddd,dud,p11,p12,p13,p14,p15,p16,p17,p18;
double p19,p21,p22,p23,P1,P2,P3,pbot,A,B,C,D,E,F,G,ae,bbe,i,BBeP;
double pe,qe,re,AAe,BBe,nnne,dude,ddde,p24,p25,p26,p27,p28,p29,P4,P5,P6,th;

float ep = 0.99;
float fork = 95000;
float ohm = 190000000;
float mu = 10;
float gaa = 6.9e6;
float rho = 190005000;
float be = 152.27;
float del = 538;

#include <dos.h>
#include <stdio.h>
#include <float.h>
#include <math.h>
#include <time.h>

main()
{
for(i = 1; i <= 10; i++){
th = 60 + 3.5*i;

/* DETERMINATION OF EIGENVALUES OF HAMILTONIAN MATRIX   */

p = (fork*fork*ep*ep - 2*ohm*ep*ep + 2*ohm - fork*fork + th*th - mu*mu)/(1 -
ep*ep);
q1 = (-ohm*ohm*ep*ep + ep*ep*rho*rho + ohm*ohm + 2*ohm*th*th -
fork*fork*th*th);
```

```
q2 = (fork*fork*mu*mu - 2*ohm*mu*mu);
q = (q1 + q2)/(1 - ep*ep);
r = (ohm*ohm*mu*mu + rho*rho*th*th - ohm*ohm*th*th)/(ep*ep - 1);

a = (3*q - p*p)/3;
b = (2*p*p*p - 9*p*q + 27*r)/27;

AA = 2*sqrt(-a/3);
BB = acos( (3*b)/(a*AA))/3;

nnn = -sqrt( 2*sqrt(-a/3)*cos(BB) - p/3);
dud = -sqrt( 2*sqrt(-a/3)*cos(BB + 2*M_PI/3) - p/3);
ddd = -sqrt( 2*sqrt(-a/3)*cos(BB + 4*M_PI/3) - p/3);

/*
*printf("nnn %lf ", nnn);
*printf("dud %lf ", dud);
*printf("ddd %lf ", ddd);
*/

/* NEXT CALCULATE REQUIRED EIGENVECTORS TO DETERMINE P */

/* p1 */

p11 = (nnn*nnn*ep*ep -th*th - nnn*nnn + mu*mu)*nnn/gaa;
p12 = (ddd*ddd*ep*ep -th*th - ddd*ddd + mu*mu)*ddd/gaa;
p13 = (dud*dud*ep*ep -th*th - dud*dud + mu*mu)*dud/gaa;
p14 = (nnn*nnn*ep*ep -th*th - nnn*nnn + mu*mu)/gaa;
p15 = (ddd*ddd*ep*ep -th*th - ddd*ddd + mu*mu)/gaa;
p16 = (dud*dud*ep*ep -th*th - dud*dud + mu*mu)/gaa;
p17 = -(mu - nnn);
p18 = -(mu - ddd);
p19 = -(mu - dud);

/* p2 */

p21 = -gaa*(nnn*nnn*nnn*fork*ep*ep + nnn*nnn*ohm*ep*ep +
nnn*nnn*nnn*nnn*ep*ep -ohm*th*th - nnn*nnn*nnn*nnn + nnn*nnn*mu*mu -
nnn*nn n*th*th - nnn*nnn*ohm - nnn*fork*th*th - nnn*nnn*nnn*fork +
ohm*mu*mu + nnn*fork*mu*mu)/(rho*rho);
p22 = -gaa*(ddd*ddd*ddd*fork*ep*ep + ddd*ddd*ohm*ep*ep +
ddd*ddd*ddd*ddd*ep*ep -ohm*th*th - ddd*ddd*ddd*ddd + ddd*ddd*mu*mu -
ddd*dd d*th*th - ddd*ddd*ohm - ddd*fork*th*th - ddd*ddd*ddd*fork +
ohm*mu*mu + ddd*fork*mu*mu)/(rho*rho);
p23 = -gaa*(dud*dud*dud*fork*ep*ep + dud*dud*ohm*ep*ep +
dud*dud*dud*dud*ep*ep -ohm*th*th - dud*dud*dud*dud + dud*dud*mu*mu -
dud*du d*th*th - dud*dud*ohm - dud*fork*th*th - dud*dud*dud*fork +
ohm*mu*mu + dud*fork*mu*mu)/(rho*rho);

p24 = -gaa*(nnn*nnn*ohm*ep*ep + nnn*nnn*nnn*fork*ep*ep +
nnn*nnn*nnn*nnn*ep*ep - nnn*nnn*th*th - nnn*nnn*ohm - ohm*th*th - nnn*nnn*n
nn*fork - nnn*fork*th*th + ohm*mu*mu + nnn*nnn*mu*mu - nnn*nnn*nnn*nnn +
nnn*fork*mu*mu)*(fork-nnn)/(rho*rho);
p25 = -gaa*(ddd*ddd*ohm*ep*ep + ddd*ddd*ddd*fork*ep*ep +
ddd*ddd*ddd*ddd*ep*ep - ddd*ddd*th*th - ddd*ddd*ohm - ohm*th*th - ddd*ddd*d
dd*fork - ddd*fork*th*th + ohm*mu*mu + ddd*ddd*mu*mu - ddd*ddd*ddd*ddd +
ddd*fork*mu*mu)*(fork-ddd)/(rho*rho);
p26 = -gaa*(dud*dud*ohm*ep*ep + dud*dud*dud*fork*ep*ep +
dud*dud*dud*dud*ep*ep - dud*dud*th*th - dud*dud*ohm - ohm*th*th - dud*dud*d
ud*fork - dud*fork*th*th + ohm*mu*mu + dud*dud*mu*mu - dud*dud*dud*dud +
dud*fork*mu*mu)*(fork-dud)/(rho*rho);

p27 = (ep*nnn + th)*(ep*mu - th);
p28 = (ep*ddd + th)*(ep*mu - th);
p29 = (ep*dud + th)*(ep*mu - th);

/* NEXT CALCULATE P1,P2,P3 etc (REQUIRED ELEMENTS OF P) */

pbot = p13*p15*p17 - p12*p16*p17 - p13*p14*p18 + p11*p16*p18 + p12*p14*p19
- p11*p15*p19;
P1 = (p16*p18*p21 - p15*p19*p21 - p16*p17*p22 + p14*p19*p22 + p15*p17*p23 -
p14*p18*p23)/pbot;
P2 = (p13*p18*p21 - p12*p19*p21 - p13*p17*p22 + p11*p19*p22 + p12*p17*p23 -
p11*p18*p23)/(-pbot);
P3 = (p13*p15*p21 - p12*p16*p21 - p13*p14*p22 + p11*p16*p22 + p12*p14*p23 -
p11*p15*p23)/pbot;
```

```
P4 = (p13*p18*p24 - p12*p19*p24 - p13*p17*p25 + p11*p19*p25 + p12*p17*p26 -
p11*p18*p26)/(-pbot);
P5 = (p13*p15*p24 - p12*p16*p24 - p13*p14*p25 + p11*p16*p25 + p12*p14*p26 -
p11*p15*p26)/pbot;
P6 = (p13*p15*p27 - p12*p16*p27 - p13*p14*p28 + p11*p16*p28 + p12*p14*p29 -
p11*p15*p29)/pbot;

/* CALCULATE EIGENVALUES OF RICCATI SOLUTION */

pe = -(P1+P6+P4);
qe = -(P2*P2-P1*P4-P1*P6-P4*P6+P3*P3+P5*P5);
re = -(P1*P4*P6-P3*P3*P4-P2*P2*P6-P1*P5*P5+2*P2*P3*P5);

ae = (3*qe - pe*pe)/3;
bbe = (2*pe*pe*pe - 9*pe*qe + 27*re)/27;

AAe = 2*sqrt(-ae/3);
BBeP = (3*bbe)/(ae*AAe);
if(BBeP = 1){
BBe = 0; }
else{ BBe = acos( BBeP)/3; }

nnne = 2*sqrt(-ae/3)*cos(BBe) - pe/3;
dude = 2*sqrt(-ae/3)*cos(BBe + 2*M_PI/3) - pe/3;
ddde = 2*sqrt(-ae/3)*cos(BBe + 4*M_PI/3) - pe/3;

/*
* PRINTOUT REQUIRED VALUES TO DETERMINE IF RICCATI EQUATION
* POSITIVE SEMI-DEFINITE FOR SELECTED WEIGHTING FUNCTION
* PARAMETER VALUE
*/

printf("nn= %f ", nnne);
printf("dd= %f", ddde);
printf("du= %f ", dude);
printf("th= %f\n", th);
}

return(0); }
```

APPENDIX E

SUPPORTING PUBLICATIONS