Power system optimisation and stability studies using real-time simulation

Williams, S. K.

Award date:
1986

Awarding institution:
University of Bath

POWER SYSTEM OPTIMISATION AND STABILITY STUDIES USING REAL-TIME SIMULATION

Submitted by

S. K. Williams, B.Sc.(Hons.)

for the degree of

Doctor of Philosophy

of the University of Bath

1986

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

CONTENTS

ABSTRACT

A multi-processor Real-Time Power System Simulator is used to optimise the transient performance of power systems. Optimal control theory is reviewed and the form of a sub-optimal controller is inferred. Several simulation techniques are discussed and a discrete time domain representation for the various components of a power system is given.

The Real-Time Power System Simulator is described in terms of its hardware and software, with particular attention given to the Monitor task which manages the collection and use of data generated by the simulator. Within the Monitor task, mechanisms have been developed to compare the dynamic and transient stability limits of power systems with different controller settings. These mechanisms are then used to compare the dynamic and transient stability limits changes brought about by the optimisation of the transient performance of the power system.

The hardware developed by the author has been used in conjunction with his specifically written software to study, in real time, the dynamic and transient performance of several different power system models. Significant improvements in the transient performance of power systems by the use of sub-optimal controllers have been demonstrated.

ACKNOWLEDGMENTS

The author is indebted to his supervisor, Mr. A. R. Daniels, for his help and guidance throughout this work. He would also like to thank Professor J. F. Eastham, Head of the Power & Control Group, and Professor T. E. Rozzi, Head of School, for provision of the facilities of the School of Electrical Engineering. He is grateful to Dr. I. A. Erinmez of the CEGB and to Lt. Cmdr. B. E. Hill of the RNEC Manadon, who have assisted by supplying plant data for the simulations presented within this thesis. His thanks are extended to the SERC for funding throughout the period of this work.

He would also like to express his thanks to Dr. B. A. White, Mr. V. S. Gott and to the postgraduates of the School, both past and present, who have at various times involved themselves in his activities. Finally, he would like to thank Miss P. Hammond for the time and effort she has expended converting the handwritten draft into this thesis.

LIST OF PRINCIPAL SYMBOLS

The principal symbols used in this thesis are listed as follows. Symbols which do not appear in this list will be defined in the test as they are introduced.

V_d, V_q = stator voltages in d and q axes respectively

V_{fd} = field voltage

V_f = field voltage referred to the armature circuit

i_d, i_q = stator currents in the d and q axes respectively

i_{fd}, i_{1d}, i_{1q} = current in the field winding and the d and q axes damper windings respectively

r_a = stator resistance in d and q axis windings

r_{fd}, r_{1d}, r_{1q} = resistances in the field winding and the d and q axes damper windings respectively

ψ_d, ψ_q = stator flux linkages in d and q axes respectively

ψ_{fd}, ψ_{1d}, ψ_{1q} = flux linkages in the field winding and the d and q axes damper windings respectively

x_{ffd}, x_{11d}, x_{11q} = total reactances of the field winding and the d and q axis damper windings respectively

x_{ad}, x_{aq} = mutual reactances between any pair of d axis windings and between any pair of q axis windings respectively

x_d, x_q = synchronous reactances in d and q axes respectively

x'_d = d axis transient reactance

x''_d, x''_q = d and q axis subtransient reactances respectively

e'_q = q axis voltage behind transient impedance

e''_d, e''_q = d and q axis voltage behind subtransient impedances respectively

T'_{do}, T'_d = d axis transient open and short circuit time constants respectively (sec)

T''_{do}, T''_d = d axis subtransient open and short circuit time constants respectively (sec)

T''_{qo}, T''_q = q axis subtransient open and short circuit time constants respectively (sec)

T'_{qo}, T'_q = q axis transient open and short circuit time constants respectively (sec)

Pe = electrical power at the generator terminals

Telec = airgap torque

Tout = mechanical torque

K_d = damping factor

H = inertia constant (kWs/kVA)

M = $H/\pi f_o$

ω_o, f_o = rated frequency (rad/s and Hz respectively)

ω = instantaneous angular velocity or rotor (rad/s)

ω_{err} = governor system speed error (rad/s)

γ = rotor slip speed (rad/s)

δ = rotor angle (rad)

$\delta(t)$ = an impulse function

V_b, V_t = voltage of infinite busbar and machine terminals respectively

R_t, x_t	=	transfer resistance and reactance respectively
R_l, x_l	=	transmission line resistance and reactance respectively
R_x, x_x	=	generator transformer resistance and reactance respectively
p	=	differential operator d/dt
s	=	Laplace operator
t	=	time (sec)
z^{-i}	=	Z transform delay operator
Vref	=	excitation system reference
V_i	=	extra feedback signal to excitation system
V_s	=	excitation system stabilising voltage
Gov_{ref}	=	governor system reference
T_i	=	extra feedback signal to governor system
T_a, T_b, T_c	=	governor system time constants (sec)
T_e, T_s	=	excitation system time constants
Δ	=	prefix to denote a deviation about an initial operating point
o	=	subscript to denote an initial value
$	=	prefix to indicate a hexadecimal value
$\underline{x}^T$	=	transpose of vector $\underline{x}$
A^T	=	transpose of matrix A

CHAPTER 1

INTRODUCTION

1.1 The Evolution of Power Systems

The evolution of generating plant has been stimulated since its earliest days by an ever increasing demand for electrical power. Early generating plants were both small and thermally inefficient by today's standards, and economic pressures gradually demanded an increase in plant efficiency. As the power system in this country evolved, other pressures were brought to bear, particularly environmental pressures, restricting the siting of new generating plant and the amount of overhead transmission network available to transport electrical power from centres of generation to centres of load. Taken together, these pressures favour large, thermally efficient generating plants, coupled together by the minimum number of interconnections. Today, a modern, thermally efficient power station, consisting of four generating sets, is capable of generating in excess of 2000MW of power. In the case of small power systems, such as those encountered on small islands, oil rigs and naval vessels, these pressures have resulted in large, efficient plant, that under light loading conditions may reduce the power system to a single item of generating plant.

The first high speed alternating current generating plant was installed in 1888 by C. A. Parsons (1-1) and used a steam turbine prime mover. This generator was fitted with a rotating armature

winding and a static field winding. This arrangement was soon discarded in favour of a static armature winding and a rotating field winding, due to the difficulties experienced in removing high levels of power from the machine using slip rings. In 1912 the basic form of the turbogenerator plant was established, again by C. A. Parsons, with the installation of a 25MW turbogenerator set. Since then, there has been a steady increase in the power output available from a single generator set.. This increase in plant capacity can be attributed to improvements in engineering techniques rather than to fundamental design changes in the form of the turboalternator configuration. In particular, improved cooling schemes have allowed large increases in the electrical power output of generating units. During the 1950's, hydrogen cooling of the machine rotor appeared and, subsequently, water cooled stator windings were introduced.

This increase in the power output of generating plant has not been accompanied by a proportional increase in the inertia of the rotor shaft, and has resulted in a reduction of the short circuit ratio. Both these trends are detrimental to power system stability. With smaller values for the inertia constant, H, less mechanical power is required to increase the machine load angle under transient conditions, and the smaller short circuit ratio reduces the capability of the machine to transmit power for a given field current by increasing the synchronous impedance.

1.2 Power System Stability

A power system generally consists of synchronous generators and loads, interconnected by a transmission system. This is a dynamic

system, and the stability of a power system is the ability of a synchronous generator to regain and maintain synchronism with the rest of the power system to which it is connected, following a disturbance (1-2, 1-3). The steady-state stability of a power system reflects its ability to regain and maintain synchronism following small, slow disturbances (such as those caused by normal load fluctuations), while transient stability reflects the ability of a power system to regain and maintain synchronism following large and sudden disturbances (such as those caused by system faults or line switching). The stability limit of a generator is the maximum power that the generator can deliver without losing synchronism. In this way, the steady-state stability limit is defined as the maximum power that a generator can deliver without losing synchronism when subjected to a small, slow disturbance, and the transient stability limit is defined as the maximum power that a generator can deliver without losing synchronism when subjected to a large, sudden disturbance. The maximum power that a generator may deliver is governed by its capacity and the conditions under which it operates. When operating in the lagging power factor region of the operating chart, the maximum power output of the generator is limited by the maximum power output of the prime mover, and by the heating of rotor and stator windings. When operating in the leading power factor region, the maximum power output of a generator is limited by stability considerations. The steady-state stability limit, which depends on the presence of small, slow disturbances, is well defined, while the transient stability limit depends on the type and severity of the disturbance used to define the stability limit. The transient stability criterion usually adopted is the ability of the generator to regain and maintain synchronism following a three-phase short

circuit fault of a standard duration, close to the generator terminals. When transient stability is defined in this manner, operation in the leading power factor region of the operating chart is usually restricted by the transient stability limit, which is generally lower than the steady-state stability limit.

The steady-state stability limit may be discussed in terms of synchronising and damping torque components (1-4). Synchronising torques are in phase with the load angle and tend to restore the machine load angle following a disturbance, while damping torques are in phase with the rotor slip speed, and tend to damp out rotor oscillations about the synchronous reference frame. In order to maintain steady-state stability, both synchronising and damping torques must be positive. This ensures that a small increase in electrical load, resulting in a small increase in load angle, will be compensated for by an increase in electrical power output, and that rotor oscillations will be damped out. Steady-state stability can be lost if either of these torque components is missing, or negative. A lack of synchronising torque is characterised by a monotonic increase in load angle following a small disturbance, while a lack of damping torque is characterised by load angle oscillations of increasing amplitude. The steady-state stability limit of a synchronous generator can be extended by the use of a continuous, fast-acting, automatic voltage regulator (AVR), which allows stable operation at load angles well in advance of those possible using fixed excitation. The region of the operating chart between the fixed excitation steady-state stability limit and the steady-state stability limit obtained using a modern AVR, is generally referred to as the dynamic stability region, and the later stability limit is referred to as the

dynamic stability limit, to distinguish it from the fixed excitation steady-state stability limit. Improvement of the steady-state stability limit by means of excitation control using various extra feedback signals has been investigated by many authors (1-4 to 1-8). Practical extra feedback signals include shaft speed (1-4,1-5), terminal power (1-6,1-7) and frequency deviation (1-8).

When the system is disturbed by a three-phase, short circuit fault, close to the high voltage terminals of the generator transformer, the voltage at this point collapses, and prevents the transfer of power between the generator and the rest of the power system. While the fault is present, the power balance between mechanical power input, and electrical power output is upset, and the excess power must either be dissipated in increased losses, or stored in increased magnetic fluxes and as kinetic energy by increasing the rotor speed. The high level of stator currents at fault inception result in excess electrical losses which may give rise to an initial retarding torque and, consequently, an angular backswing. The magnitude of this backswing is generally small and it is usually neglected. Most of the excess input energy is absorbed as additional kinetic energy in the rotor shaft. The increased rotor speed in turn increases the angle of the machine e.m.f. relative to the rest of the power system. When the fault is cleared, the transmission of power to the rest of the system is resumed, possibly at a new power level, depending on the state of the power system following fault clearance. The generator will generally maintain synchronism if the excess kinetic energy can be transferred to the electrical system as electrical power. This introduces the concept of first swing transient stability, which may be considered as a measure of the

likelihood that the machine will not pole slip on the first rotor swing.

Common methods of comparing the transient stability of power systems, based on a first swing criterion are pre-fault/post-fault impedance plots and critical fault clearance times. First swing stability gives no insight into the behaviour of the transient following the first swing in the load angle excursion and, in order to obtain a comparison of the subsequent behaviour, some form of performance criterion must be defined. This criterion would generally be some function of the terminal voltage and load angle, designed to favour the early damping of rotor angle oscillations and the rapid recovery of the terminal voltage.

Modern, fast acting, high ceiling AVRs offer an improvement in the transient stability limit over a fixed excitation system. The terminal voltage error during the fault is sufficient to drive the excitation voltage to its positive ceiling, which maximises the power transfer during and after the fault, so reducing the size of the load angle overshoot. However, simple terminal voltage feedback, while increasing the synchronising torque component, reduces the damping torque component (1-4), and so is not very effective at damping rotor oscillations following the first load angle swing. In general, the terminal voltage error keeps the field voltage in positive saturation for too long, resulting in excessive power transfers between the mechanical and electrical systems, thereby giving an overshoot in the load angle recovery. Supplementary feedback signals (1-4 to 1-8), fed into the excitation system, can be used to give more control over the field voltage response, without degrading the terminal voltage

recovery.

When considering the governing system, only small speed deviations are present during the fault and these alone will cause little governing action. Modern, fast acting, electrohydraulic governing systems are capable of significantly reducing the prime mover output torque during a fault, thus improving first swing stability and subsequent damping, provided a suitable control strategy can be employed. Since power balance is the ultimate consideration for the stability of a generator, the most direct solution to the stability problem is to maintain the power balance, either by reducing the mechanical input power to the generator, or by increasing the electrical power output during the fault. This gives rise to such techniques as fast governing (1-9,1-10), fast valving (1-11, 1-12) reduction of the fault clearance time, braking resistors (1-13) and fast excitation with high field voltage ceiling (1-14).

In recent years, interest has been shown in the application of optimal control theory to the problem of stabilising power systems (1-15 to 1-17) by the use of extra control signals, fed into the excitation and governing systems. Most early studies used linearised system representations, and controls obtained by the solution of the matrix riccatti equation require access to all the system states. Such controls are not easily applied in practice since some of the system states will be inaccessible. Sub-optimal controls have been obtained using non-linear methods (1-18) and selected system states (1-19, 1-20). It is reasonable to assume that, in control schemes requiring access to all the system states, some states will dominate the control law. States making little contribution to the control

scheme may be eliminated from the control law (1-19,1-20) and it is an objective of the work presented in this thesis to improve the transient performance of a power system by obtaining sub-optimal controls which employ the feedback of a limited number of extra control signals. Several authors (1-21 to 1-23) have considered the reduction of the number of extra feedback signals necessary to produce significant improvements in the transient performance of a synchronous generator connected to an infinite bus bar, and the most popular of these signals are the terminal power, Pe, the direct axis current, I_d, and the rotor acceleration, $p^2\delta$. The use of extra control signals in improving the transient performance of a small power system comprising a single synchronous generator and an inductive load is also investigated. In this case the operating power factor is determined entirely by the impedance angle of the load, and it is not possible for synchronism to be lost. The performance objectives in this case are the rapid recovery of terminal voltage and the maintenance of the power system frequency following a fault or a load change.

1.3 Power System Simulation

The simulation of power systems is necessary to study their transient behaviour under controlled and repeatable conditions. This is particularly true when the transient performance of a power system is to be optimised with respect to some performance criterion. Analogue computer simulations are cumbersome, and incapable of automatically adjusting control parameters to improve transient performance as an optimisation proceeds. The general trend in simulation studies has been towards numerical solution of the system

equations using digital computers. As it becomes necessary to simulate larger and more complex power systems, increasing demands are placed on the processing power of the computing systems. Traditionally these computing systems have been large mainframe computers, which are unable to guarantee the speed of response given to any individual user, and it is uneconomic to comtemplate using such computer hardware as a single user power system simulator.

In recent years microprocessor parts have appeared on the market, and the latest of these devices offer processing power which rivals that of mini-computers such as the PDP 11/34. Indeed, the divisions between so-called micro, mini and mainframe computer systems are no longer clear cut, as integration techniques evolve and permit more complex devices to be fabricated using the same area of semiconductor. The latest generation of microprocessor devices offers cheap computing power which may be placed at the disposal of a single user. Large-scale power system simulations can then be undertaken, with guaranteed response times, at speeds which, from the user's point of view, are far in excess of those obtainable using either mini or mainframe computer systems. It is now possible to economically develop sufficient computing power to run a transient simulation of a power system in parallel with the real power system.

1.4 Digital Systems

Since the first transistors appeared in the 1950's, the field of digital electronics has evolved at a tremendous pace. The fabrication of more than one transistor on a piece of semiconductor soon became commonplace and, as new, more complex and more powerful

semiconductor devices were developed, the cost of these components fell. Similarly, the cost of constructing computer systems from these new devices has been falling and the reliability of the resulting systems has been improving. Semiconductor integration technology has evolved through large-scale integration (LSI) in the late 1960's which gave rise to devices containing of the order of one thousand transistors, to today's very large-scale integration (VLSI), where an integrated device may contain in excess of 70,000 transistors. Single component microcomputer systems are now available, containing random access memory (RAM), read only memory (ROM), input/output and a central processing unit (CPU), all fabricated on a single piece of semiconductor. Although not very powerful as a general purpose computing system, these devices are much used in dedicated applications such as the control of consumer goods, (e.g. microwave ovens and washing machines) and are rapidly finding new applications in the field of vehicle electronics (e.g. self tuning ignition timing, anti-skid braking systems, and vehicle instrumentation). A further order of magnitude increase in the number of transistors per device will mean that large scale computing systems may be constructed from a handful of components.

At the end of the 1960's, when LSI technology emerged, further advances were temporarily blocked by a lack of application. Apart from memory devices, such large-scale devices would be too specialised to be of any use. The development of the pocket calculator revived the situation by providing a device which required the use of LSI technology and was extremely marketable, so giving rise to the volume production necessary to sustain the semiconductor industry. It was a short step from the fixed function calculator to

the programmable calculator, and from there the next step was the design of a general purpose microprocessor. The use of microprocessor based solutions to engineering problems greatly reduces the component count, and improves the flexibility of the solution so that design changes may be made easily. Today the problem of device specialisation has been solved in two ways. First, the microprocessor route, which provides a general purpose device, and secondly, the advent of gate array technology. Gate arrays are LSI and VLSI devices which are made up of uncommitted logic gates. It is then a relatively cheap process to customise the device by designing the metalisation layer which connects these logic gates together and produces a dedicated device. Gate arrays, however, are still an expensive solution when very small production quantities are required.

The earliest microprocessors were not very powerful and could not be considered to be general purpose processing devices. The number of registers, the width of the data and address paths, and the instruction set were restricted by the technology available, since only a limited number of active components could be fabricated on a single integrated circuit. These microprocessors found limited application in such fields as industrial controls, logic sequencing and as general purpose logic devices. The second generation microprocessors (I8080, 6502, MC6800, Z80, etc.) evolved from a realisation that, given a suitable architecture, the microprocessor could compete with the traditional computing elements. These second generation devices were fabricated using metal oxide silicon technology (MOS), which gave rise to the transition from LSI to VLSI. As a result of the overall reduction in size of these devices, speed

and power dissipation have improved due to a reduction in the gate capacitance and the cell volume.

As microprocessor devices have become more complex, their designs have become more regular. Early microprocessors were so-called 'random logic' designs, whereas today's third generation microprocessors, such as the MC68000, are built around microcoded sequencers, resulting in a large and regular instruction set. The Motorola MC68020, which has just appeared on the market, represents the state of the art in microprocessor technology. The instruction stream is fed to the execution sequencer via an instruction prefetch pipeline. This pipeline, and the execution unit both make requests on a bus controlling sequencer which is responsible for moving data between the microprocessor and memory. Internal to the device there is also an instruction cache and, using this architecture, it is possible for an instruction to have an execution time of zero. A register to register instruction, which has already been prefetched into the instruction pipeline, may be executed simultaneously with the return of the result of the previous operation to memory by the bus controlling sequencer, and it becomes impossible to attribute any execution time to this instruction. The MC68020 also fully supports virtual memory by instruction continuation, and has been labelled a 'microframe' since its architecture closely resembles that of a mainframe computer.

1.5 Multiprocessor Systems

Although third generation microprocessors have resulted in large increases in the computational power available from a low cost

computer system, this power is limited. A typical third generation microprocessor has an instruction throughput of one million instructions per second (1MIP), which, although large, represents an absolute limit on the computational speed of a single processor. A means of bypassing this limit is the development of multiprocessor systems. These systems, in general, consist of a number of co-operating microprocessor devices, either tightly or loosely coupled together. The wide range of multimicroprocessor configurations available take their lead from modern mainframe architecture. Here a number of tightly coupled CPU's perform the main task execution, while slightly more loosely coupled front end processors manage the input/output functions, such as disc storage and interaction with the user. These front end processors may in turn be connected directly to the user's terminal, or loosely coupled to other computer systems, remote from the main installation, which provide terminal concentration and multiplexing facilities for remote users.

Distributed microprocessing gives the user several advantages. The reliability and fault tolerance of such systems is improved by the distribution and duplication of processing subsystems. More processing power in terms of instruction throughput is available to the user, and the system may be expanded in a modular fashion to solve more complex problems due to the modular nature of the system architecture.

1.6 Real Time Simulation Studies

With so much processing power now available, and the increasing

demand for more complex power system simulation, with a faster response to the user's commands, it became a logical step to seek a microprocessor based solution on which to perform power system simulation. This thesis details the development of a real-time power system simulator (Chapter 4) and its use in the optimisation of the transient performance of power systems (Chapter 6). Chapter 3 details the power systems studied and the simulation methods used. Other studies have been conducted using the simulator (Chapter 5) which permit quantative comparisons to be made between the dynamic and transient stability limits obtained by the use of various control strategies. Transient stability limits are compared by the use of pre-fault/post-fault impedance charts, while the dynamic stability comparisons are made by plotting the dynamic stability limit in the leading power factor region of the operating chart.

The objective of this work is to apply optimal control theory (Chapter 2) to the improvement of the transient performance of power systems by the use of a real-time simulator, and to demonstrate the effect the use of optimised control strategies has on both the transient and dynamic stability limits of the generating plant.

References

1-1 Vickers, V.J.: "Recent trends in turbogenerators", Proc.IEE, Vol.121, No.11R, Nov.1974, pp1273-1306.

1-2 Cary, S.B.: "Power system stability", Vol.1, John Wiley, New York, 1954.

1-3 Kimbard, E.W.: "Power system stability", Vols.1 & 3, John Wiley, New York, 1948 & 1956.

1-4 Demello, F.P. and Concordia, C.: "Concepts of synchronous machine stability as affected by excitation control", Trans.IEEE, Vol.PAS-88, No.4, April 1969, pp316-329.

1-5 Watson, W. and Manchur, G.: "Experience with supplementary damping signals for generator static excitation systems", Trans.IEEE, 1973, PAS-92, pp.199-204.

1-6 Bayne, J.P., Lee, D.C. and Watson, W.: "A power system stabiliser for thermal units based on derivation of accelerating power", Trans.IEEE, 1977, PAS-96, pp1777-1783.

1-7 Demello, F.P., Hannett, LN.N and Undrill, J.M.: "Practical approaches to supplementary stabilising from accelerating power", Trans.IEEE, 1978, PAS-97, pp1515-1522.

1-8 Keay, F.W. and South, W.H.: "Design of power system stabiliser sensing frequency deviation", Trans.IEEE, 1971, PAS-90, pp707-713.

1-9 Hughes, F.M.: "Improvement of turbogenerator transient performance by control means", Proc.IEE, Vol.120, 1973, pp233-240.

1-10 Sullivan, A.C. and Yee, M.: "Fast governing of turbogenerators during transients", Proc.IEE, Vol.120, 1973,

pp371-378.

1-11 Gushing, E.W., Drechsler, G.E., Killgoar, W.P., Marshall, H.G. and Steward, M.R.: "Fast valving as an aid to power system transient stability and prompt resynchronisation and rapid reload after full load rejection", Paper 71TP705-PWR presented at 1971 IEEE-ASME Power Generation Conference, St. Louis, Mo., Sept. 19-23, 1971.

1-12 Aanstand, O.J. and Lokay, H.E.: "Fast valve control can improve turbine generator response to transient disturbances", Westinghouse Engineer, July 1970.

1-13 Ellis, H.M., Hardy, J.E., Blythe, A.L. and Skooglund, J.W.: "Dynamic stability of the Peace River Transmission System", Trans.IEEE, PAS-85, 1966, pp586-600.

1-14 Dineley, J.L., Morris, A.J. and Preece, C.: "Optimised transient stability from excitation control of synchronous generators", Trans.IEEE, PAS-87, 1968, pp1696-1705.

1-15 Iyer, S.N. and Cory, B.J.: "Optimisation of turbogenerator transient performance by differential dynamic programming", Trans.IEEE, 1971, PAS-90, pp2149-2157.

1-16 Iyer, S.N. and Cory, B.J.: "Optimal control of a turbogenerator including an exciter and governor", Trans.IEEE, 1971, PAS-90, pp2142-2148.

1-17 Yu, Y.N., Voncuriya, K and Wedman, L.N.: "Application of an optimal control theory to a power system", Trans.IEEE, 1970, PAS-89, pp55-62.

1-18 Daniels, A.R., Davis, D.M. and Pal, M.K.: "Linear and nonlinear optimisation of power system performance", Trans.IEEE, 1975, PAS-94, pp810-818.

1-19 Lee, Y.B.: "Sensitivity and optimal control studies of power systems", PhD Thesis, University of Bath, 1975.

1-20 Lu, H.: "Optimisation studies of a single machine power system", PhD Thesis, University of Bath, 1979.

1-21 Daniels, A.R. and Lu, H.: "Nonlinear single variable optimisation studies of a.c. turbogenerator performance", Proc.IEE, Vol.126, No.5, May 1979.

1-22 Dineley, J.L. and Mikhail, S.E.: "Effect of feedback signals in excitation and governor/turbine systems on transient and dynamic stability", IEEE, PES Meeting Paper A76, 113-1, Jan. 1976.

1-23 Hazell, P.A.: "Co-ordinate excitation control and governing of turbogenerators", PhD Thesis, University of Bath, 1981.

CHAPTER 2

OPTIMAL CONTROL

2.1 Introduction

This Chapter introduces the concepts behind optimal control theory. These concepts are then developed to obtain a set of necessary conditions upon which the design of an optimal controller may be based. The significance of these various conditions is discussed in order to assess how each will influence the solution of an optimal control. A number of solution methods are then considered and their suitability with regard to deriving optimal controls for a power system is discussed.

The objective of an optimal controller is to predict a continuous control function of time, or a sequence of controls, which, when applied to a plant during a specified interval of time, will cause the plant to behave in some optimum manner. The term 'optimum manner' must be put into context and, in order to do this, it is necessary to establish the concepts of performance measure and performance index.

A performance measure is some arbitrary scalar valued function of the plant state variables, the plant output variables and the plant control variables. The objective of the performance measure is to provide an instantaneous scalar quantity which may be used to assess the instantaneous quality of the system. The performance measure itself gives no explicit information about the state of the system,

rather it indicates how close the system states or outputs are to their desired operating point, and how much control effort is currently being expended.

A performance index is the time integral of a performance measure, taken over the closed control interval, $t_o \leq t \leq t_f$, for which the optimal control is required, where t_o and t_f denote the beginning and end of the control interval respectively. A plant is said to behave in an optimum manner when the control applied to it extremalises (i.e. maximises or minimises) the performance index.

Controls generated by a conventional controller are functions of the instantaneous state of the system and the desired state of the system. As such, a conventional controller may be thought of as extremalising (minimising) a performance measure, and no account is taken of the possibility of future changes in the desired system state, or the possibility that the control required to minimise the performance measure at this instant may give rise to larger performance measures at future instants. In extremalising a performance index, an optimal controller generates controls not only for the current instant but for the entire control interval over which the performance index is evaluated.

Without losing generality only the process of minimisation will be considered in this Chapter, since a maximisation problem may be transformed into a minimisation problem by simply negating the value of the performance index that is to be maximised.

2.2 Performance Measures

It is clear that an optimal control derived by the minimisation of a performance index will be heavily dependent on the performance measure chosen. Therefore, it is important that a performance measure should embody the objectives of the control scheme. It is also important that during the integration process, which evaluates the performance index, the value of the performance measure at one instant in time should not cancel out its value at another instant. For this reason, it is desirable that the performance measure should yield values which are either all positive and zero, or all negative and zero. Consider a single output system in which the difference between the actual output and the desired output is given by:

$$e = x - x_d \tag{2.1}$$

where x is the system output and x_d is its desired value. The performance measure $L(x,x_d) = e$ is not suitable in the context of optimal control, since it is possible for it to take on both positive and negative values, which may cancel out when the performance index

$$I = \int_{t_o}^{t_f} L(x,x_d)\, dt \tag{2.2}$$

is evaluated.

Suitable performance measures based on the output error of equation (2.1) are $L(x,x_d) = |e|$, $L(x,x_d) = e^2$ and $L(x,x_d) = K^e$ where

K is any positive number except unity.

In most regulator problems, the performance index is based on the output error of the system. A performance measure of quadratic form for a system with n outputs and m controls may in general be represented by:

$$L(\underline{x}(t),\underline{x}_d(t),\underline{U}(t),t)$$

$$= [\underline{x}(t)-\underline{x}_d(t)]^T Q_1(t)[\underline{x}(t)-\underline{x}d(t)] + \underline{U}(t)^T Q_2(t)\underline{U}(t) \qquad (2.3)$$

where $\underline{x}(t)$ is an nxl vector representing the plant outputs, $\underline{x}_d(t)$ is an nxl vector representing the desired plant outputs, $\underline{U}(t)$ is an mxl vector representing the control inputs, $Q_1(t)$ is an nxn time varying positive, semi-definite matrix of weighting coefficients, and $Q_2(t)$ is an mxm time varying positive semi-definite matrix of weighting coefficients. This expression may be simplified slightly by redefining the performance measure in terms of the nxl output error vector, $\hat{\underline{x}}$, such that:

$$\hat{\underline{x}}(t) = \underline{x}(t) - \underline{x}_d(t) \qquad (2.4)$$

and

$$L(\hat{\underline{x}}(t), \underline{U}(t),t) = \hat{\underline{x}}^T(t)Q_1(t)\hat{\underline{x}}(t)+\underline{U}^T(t)Q_2(t)\underline{U}(t) \qquad (2.5)$$

The first term of the performance measure takes into account the plant output error, while the second term provides a means by which

the control inputs may contribute to the performance index. If this second term is ignored, it is possible that the minimisation process which generates the optimum control function will require infinitely large magnitude control signals. Incorporating this second term in equations (2.3) and (2.5) allows a trade off to be made between output errors and control amplitude.

The matrices Q_1 and Q_2 have been shown as time varying. This permits the output errors and control magnitudes to make time varying contributions to the performance index, with the effect that output error and control amplitude variations may be made more important at differing times. For example, when a synchronous generator is subjected to a 3-phase symmetrical short-circuit fault close to its terminals, very little may be done to restore the terminal voltage while the fault is present. However, it is important that, following fault clearance, the terminal voltage is restored quickly to its pre-faulted value. In this case, an ordinary linear time weighting on the terminal voltage error term will cause terminal voltage errors during and shortly after the fault to be much less significant than those some time after the fault has been cleared. This will encourage rapid terminal voltage recovery following fault clearance. The performance measures used in Chapter 6 contain linear time weighted terms to encourage rapid recovery of both the terminal voltage and load angle deviations.

Two special cases of time weighting exist. Consider first a matrix $Q_1(t)$ in which all the elements are zero except at the terminal time $t=t_f$. Such a matrix may be written with the aid of the delayed impulse function, $\delta(t-t_f)$, to give the following performance index:

$$I = \int_{t_o}^{t_f} \underline{\hat{x}}(t)^T \delta(t-t_f) Q_1 \underline{\hat{x}}(t) + \underline{U}^T(t) Q_2(t) \underline{U}(t)\, dt \qquad (2.6)$$

Integrating the first term under the integral of equation (2.6) gives:

$$I = \underline{\hat{x}}^T(t_f) Q_1(t_f) \underline{\hat{x}}(t_f) + \int_{t_o}^{t_f} \underline{U}^T(t) Q_2(t) \underline{U}(t)\, dt \qquad (2.7)$$

It can be seen that minimising this performance index will involve minimising the output error of the system at the terminal time $t=t_f$. This type of performance index would be used when system output error is required to be near zero at the terminal time.

Strictly speaking, the second special case is not an example of time weighting. The performance measure used to perform what is known as time optimal control is unity. Thus the performance index is given by

$$I = \int_{t_o}^{t_f} 1\, dt \qquad (2.8)$$

Minimising this performance index will minimise the transit time of the system between two operating points, specified at the start and end of the control interval.

Although it is possible to invent an infinite variety of performance measures, those of time varying quadratic form are of most interest to this investigation.

2.3 Functional Minimisation

In the discussion so far, no consideration has been given to the constraint imposed on the minimisation process by the dynamic system. This constraint is the relationship between the state of the plant and the plant input which can be expressed by the vector valued function $\underline{f}$, such that

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t),\underline{U}(t),t) \tag{2.9}$$

where, for an nth order system with m inputs, $\underline{x}(t)$ and $\dot{\underline{x}}(t)$ are nx1 vectors representing the system state and rate of change of state, $\underline{U}(t)$ is an mx1 vector representing the control input, and $\underline{f}$ is an nx1 vector function which may be both non-linear and time varying. The objective of this section is to derive necessary conditions such that

$$I = \min_{\underline{U}(t)} \int_{t_o}^{t_f} L(\underline{x}(t),\underline{U}(t),t)\, dt \tag{2.10}$$

subject to the constraint that:

$$\underline{f}(\underline{x}(t),\underline{U}(t),t) - \dot{\underline{x}} = \underline{0} \tag{2.11}$$

where L is the scalar valued performance measure and $\underline{0}$ is a null vector.

Adopting the calculus of variation approach, this constrained minimisation problem may be converted into an unconstrained

minimisation problem by the use of Lagrange multipliers (2-1). A new constrained performance measure may be defined such that:

$$H(\underline{x}(t),\dot{\underline{x}}(t),\underline{U}(t),\underline{\lambda}(t),t)$$

$$= L(\underline{x}(t),\underline{U}(t),t)+\underline{\lambda}^{T}(t)[\underline{f}(\underline{x}(t),\underline{U}(t),t)-\dot{\underline{x}}(t)] \qquad (2.12)$$

where $\underline{\lambda}(t)$ is an nx1 vector of Lagrange multipliers. The problem now becomes that of finding necessary conditions for:

$$I_C = \min_{\underline{U}(t)} \int_{t_o}^{t_f} H(\underline{x}(t),\dot{\underline{x}}(t),\underline{U}(t),\underline{\lambda}(t),t)\, dt \qquad (2.13)$$

where I_C is the minimised value of the new performance index. Note that the value of both of the performance measures H and L is identical since the constraint of equation (2.11) applies. Thus, any condition necessary to solve the minimisation problem of equation (2.13) will also be necessary to solve the minimisation problem of equation (2.10).

Before deriving the necessary conditions for solving equation (2.13), some notation must be defined for the partial derivatives of both a scalar and a vector function with respect to a vector. Consider the nx1 vector $\underline{x}$, the scalar function $g(\underline{x})$ and the mx1 vector function $\underline{r}(\underline{x})$ then by definition:

$$\frac{\partial g}{\partial \underline{x}} = \left[\frac{\partial g}{\partial x_1} \cdots\cdots \frac{\partial g}{\partial x_n} \right] \qquad (2.14)$$

and

$$\frac{\partial \underline{r}}{\partial \underline{x}} = \begin{bmatrix} \frac{\partial r_1}{\partial x_1} & \cdots\cdots & \frac{\partial r_1}{\partial x_n} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \frac{\partial r_m}{\partial x_1} & \cdots\cdots & \frac{\partial r_m}{\partial x_n} \end{bmatrix} \qquad (2.15)$$

where $x_i, (1 \leq i \leq n)$ is the ith element of $\underline{x}$ and r_q, $(1 \leq q \leq m)$ is the qth element of $\underline{r}$.

For convenience in the minimisation problem, the arguments of the functions $\underline{x}$, $\dot{\underline{x}}$, $\underline{U}$, $\underline{\lambda}$, $\underline{f}$, L and H will be dropped except where necessary for clarity. If the functions which yield the required minimum in the performance index are denoted $\underline{x}^*$, $\dot{\underline{x}}^*$, $\underline{U}^*$ and $\underline{\lambda}^*$, then any arbitrary values of these functions may be written as:

$$\underline{x} = \underline{x}^* + \delta\underline{x}$$

$$\dot{\underline{x}} = \dot{\underline{x}}^* + \delta\dot{\underline{x}} \qquad \text{where } \delta\dot{\underline{x}} = \frac{d}{dt}[\delta\underline{x}] \qquad (2.16)$$

$$\underline{U} = \underline{U}^* + \delta\underline{U}$$

and

$$\underline{\lambda} = \underline{\lambda}^* + \delta\underline{\lambda}$$

Where the functions ($\delta\underline{x}$, $\delta\dot{\underline{x}}$, $\delta\underline{u}$ and $\delta\underline{\lambda}$) represent deviations from an optimal trajectory. Evaluation of a function along the optimal trajectory, i.e. where $\underline{x} = \dot{\underline{x}}^*$, $\underline{x} = \underline{x}^*$, $\underline{U} = \underline{U}^*$ and $\underline{\lambda} = \underline{\lambda}^*$ will be indicated by $|_*$. Assuming that the vector valued functions $\delta\underline{x}$, $\delta\dot{\underline{x}}$, $\delta\underline{U}$ and $\delta\underline{\lambda}$ represent arbitrary, small deviations from the optimal trajectory, then an expression for the value of the performance index in the neighbourhood of the optimal trajectory may be written by first order Taylor series expansion, and the corresponding small change in the value of the performance index, δI, is:

$$\delta I = \int_{t_o}^{t_f} \left[\frac{\partial H}{\partial \underline{x}}\bigg|_* \delta\underline{x} + \frac{\partial H}{\partial \dot{\underline{x}}}\bigg|_* \delta\dot{\underline{x}} + \frac{\partial H}{\partial \underline{U}}\bigg|_* \delta\underline{U} + \frac{\partial H}{\partial \underline{\lambda}}\bigg|_* \delta\lambda \right] dt \qquad (2.17)$$

Since each of the small change functions is arbitrary, equation (2.17) may be partitioned into three integrals, each of which must be zero independently in order for the change in performance index, δI, to be zero. Thus, necessary conditions required for the solution of equation (2.13) are:

$$\int_{t_o}^{t_f} \left[\frac{\partial H}{\partial \underline{x}}\bigg|_* \delta\underline{x} + \frac{\partial H}{\partial \dot{\underline{x}}}\bigg|_* \delta\dot{\underline{x}} \right] dt = 0 \qquad (2.18)$$

$$\int_{t_o}^{t_f} \frac{\partial H}{\partial \underline{U}}\bigg|_* \delta\underline{U}\, dt = 0 \qquad (2.19)$$

and

$$\int_{t_o}^{t_f} \left.\frac{\partial H}{\partial \underline{\lambda}}\right|_{*} \delta\underline{\lambda} = 0 \qquad (2.20)$$

As $\delta\dot{\underline{x}} = \frac{d}{dt}[\delta\underline{x}]$, equation (2.18) may be integrated by parts to give:

$$\int_{t_o}^{t_f} \left[\left[\frac{\partial H}{\partial \underline{x}} - \frac{d}{dt}\left[\frac{\partial H}{\partial \dot{\underline{x}}}\right]\right] \Big|_{*} \delta\underline{x}\right] dt + \left[\left.\frac{\partial H}{\partial \dot{\underline{x}}}\right|_{*} \delta\underline{x}\right]_{t_o}^{t_f} = 0 \qquad (2.21)$$

Also, since $\delta\underline{x}$, $\delta\lambda$ and $\delta\underline{U}$ are all arbitrarily small but finite, and because the various trajectories in the interval $t_o < t < t_f$ are independent of the values at the extremes of the control interval, $t = t_o$ and $t = t_f$, the necessary conditions of equations (2.18) to (2.20) become:

$$\left.\frac{\partial H}{\partial \underline{x}}\right|_{*} - \frac{d}{dt}\left[\left.\frac{\partial H}{\partial \dot{\underline{x}}}\right|_{*}\right] = \underline{0} \quad \text{(Euler Lagrange Equation)} \qquad (2.22)$$

$$\left.\frac{\partial H}{\partial \underline{U}}\right|_{*} = \underline{0} \qquad \text{(Control Equation)} \qquad (2.23)$$

$$\left.\frac{\partial H}{\partial \underline{\lambda}}\right|_{*} = \underline{0} \qquad \text{(State Equations)} \qquad (2.24)$$

$$-\left[\underline{\lambda}^{*T}.\delta\underline{x}\right]_{t_o}^{t_f} = 0 \qquad \text{(Transversality Condition)} \qquad (2.25)$$

These, then, are the necessary conditions which must be satisfied by the solution of equations (2.10) and (2.13).

Before discussing the implications of these necessary conditions, another form for these conditions was established by Pontryagin (2-2) et al, along with the Minimum Principle. Pontryagin showed that the solution to equation (2.10) requires that the Hamiltonian function

$$H_p(\underline{x},\underline{U},\underline{\lambda},t) = L(\underline{x},\underline{U},t) + \underline{\lambda}^T \underline{f}(\underline{x}\ \underline{U}\ t) \qquad (2.26)$$

must be minimised. The necessary conditions of equations (2.22) - (2.25) may be expressed in terms of the Hamiltonian of equation (2.26), by noting that:

$$H = H_p - \lambda^T \dot{\underline{x}} \qquad (2.27)$$

Thus, Pontryagin's equations are:

$$\frac{\partial Hp}{\partial \underline{x}}\Big|_* + \dot{\lambda}^{*T} = \underline{0} \qquad \text{(Adjoint or Co-state Equation)} \qquad (2.28)$$

$$\frac{\partial Hp}{\partial \underline{U}}\Big|_* = \underline{0} \qquad \text{(Control Equation)} \qquad (2.29)$$

$$\frac{\partial Hp}{\partial \underline{\lambda}}\Big|_* - \dot{\underline{x}}^{*T} = \underline{0} \qquad \text{(State Equation)} \qquad (2.30)$$

$$-[\underline{\lambda}^{*T}.\delta\underline{x}]_{t_o}^{t_f} = 0 \qquad \text{(Transversality Condition)} \qquad (2.31)$$

The essential difference between Pontryagin's formulation of the

necessary conditions and that given previously is the statement of the minimum principle which requires that the optimal control, $\underline{U}^*(t)$, must be a member of the set of admissable controls. If one or more of the elements $U_i(t)$, $1 \leq i \leq m$, of the control vector is constrained, such that $U_{min} \leq U_i \leq U_{max}$, and the necessary conditions, equations (2.28) to (2.31), require the control $U_i(t)$ to lie outside this range, or no control information is available from the control equation, (2.29), then the minimum principle requires that $U_i(t)$ takes on the value in the admissable range which minimises the Hamiltonian, equation (2.26). This is summarised by the statement that:

$$H_p|_* = \inf_{\underline{U} \in U} \left\{ H_p(\underline{x}^*(t), \underline{U}(t), \underline{\lambda}^*(t), t) \right\} \qquad (2.32)$$

where inf denotes the greatest lower bound or infimum and U is the set of admissible controls. As a direct consequence of the minimum principle, should the necessary conditions require that the optimal control $U_i(t)$ lie outside its permitted range, it will take on the value U_{min} or U_{max}, whichever minimises the Hamiltonian, equation (2.26). The derivation of equations (2.22) to (2.25) also made use of a Taylor series expansion which requires that the first partial derivative of the state equation with respect to the control vector $\underline{U}$ is continuous. This requirement is relaxed by the use of Pontryagin's Minimum principle.

2.4 Implications of the Necessary Conditions

In the next section the application of the necessary conditions,

equations (2.28) to (2.31), will be illustrated by deriving an optimal control for a linear system subjected to a quadratic performance measure. Prior to that, some discussion is necessary with regard to the implications of equations (2.28) to (2.31). The comments which follow apply equally to the necessary conditions of equations (2.22) to (2.25), since it has already been shown that both sets of equations are closely related.

2.4.1 The Transversality Condition

This condition is defined by equation 2.31 as:

$$-[\underline{\lambda}^{*T}.\delta\underline{x}]_{t_o}^{t_f} = 0 \qquad (2.31)$$

which implies:

$$\underline{\lambda}^{*T}(t_o)\,\delta\underline{x}(t_o) - \underline{\lambda}^{*T}(t_f)\,\delta\underline{x}(t_f) = 0 \qquad (2.33)$$

The vector function $\delta\underline{x}(t)$ is entirely arbitrary in the interval $t_o < t < t_f$ and the conditions at either end of the interval, $\delta\underline{x}(t_o)$ and $\delta\underline{x}(t_f)$, are independent of each other, but depend on the specification of initial and terminal conditions. Consequently, in order that equation (2.31) is satisfied, each of the terms of equation (2.33) must be zero, independently. This means that the vectors $\underline{\lambda}(t_o)$ and $\delta\underline{x}(t_o)$ must be orthogonal, as must the two vectors $\underline{\lambda}(t_f)$ and $\delta\underline{x}(t_f)$. If an initial condition is given for one of the system states, say $x_i(t_o) = x_{io}$ then $\delta x_i(t_o)$ must be zero, since by equation (2.16) $x_i = x_i^* + \delta x_i$ and $x_i(t_o)$ must be on the optimal

trajectory. If $\delta x_i(t_o)$ is zero, then $\delta \underline{x}(t_o)$ and $\underline{\lambda}(t_o)$ will be orthogonal for all values of $\lambda_i(t_o)$. Conversely, if no initial condition is given for x_i, then $\delta x_i(t_o)$ is entirely arbitrary and $\lambda_i(t_o)$ must be zero to guarantee that the two vectors $\delta \underline{x}(t_o)$ and $\underline{\lambda}(t_o)$ are orthogonal. Similar considerations may be applied to the terminal conditions, $\delta \underline{x}(t_f), \underline{\lambda}(t_f), \delta x_i(t_f)$ and $\lambda_i(t_f)$. These considerations are summarised in Table 2.1.

$x_i(t_o)$	$x_i(t_f)$	$\delta x_i(t_o)$	$\delta x_i(t_f)$	$\lambda_i(t_o)$	$\lambda_i(t_f)$
FREE	FREE	FREE	FREE	0	0
x_{io}	FREE	0	FREE	FREE	0
FREE	x_{if}	FREE	0	0	FREE
x_{io}	x_{if}	0	0	FREE	FREE

Table 2.1

It can be seen from this Table that every row contains one initial condition at time t_o and one terminal condition at time t_f. The presence of the transversality condition therefore creates a Two Point Boundary Value Problem (TPBVP).

2.4.2 The Control and State Equations

Equation (2.29) has been labelled the control equation. Application of this equation to the Hamiltonian of equation (2.26) gives:

$$\frac{\partial}{\partial \underline{U}} \left[L(\underline{x}^*, \underline{U}^*, t) + \underline{\lambda}^{*T} \underline{f}(\underline{x}^*, \underline{U}^*, t) \right] = \underline{0} \qquad (2.29)$$

i.e.

$$\frac{\partial L}{\partial \underline{U}}\bigg|_{*} + \underline{\lambda}^{*T} \frac{\partial \underline{f}}{\partial \underline{U}}\bigg|_{*} = \underline{0} \qquad (2.34)$$

Provided that either the performance measure, L, or the system function $\underline{f}$ is non-linear in the control vector $\underline{U}$, then the respective partial derivatives in equation (2.34) will be functions of $\underline{U}$ and the equation may then be rearranged to express the control $\underline{U}^*$ as a function of the co-state vector $\underline{\lambda}^*$. If both L and $\underline{f}$ are linear in $\underline{U}$ then the control equation yields no information regarding the optimal control $\underline{U}^*$, since neither of the partial derivatives are functions of the control vector. In this case the control $\underline{U}^*$ must be obtained by direct minimisation of the Hamiltonian according to Pontryagin's maximum principle. For example, consider the dynamic equation described by the state equation:

$$\dot{\underline{x}} = \underline{f}(\underline{x},t) + B(\underline{x},t).\underline{U} \qquad (2.35)$$

where $\underline{f}$ is an nx1 vector function which may be time varying and non-linear, and B is an nxm matrix function which may also be non-linear and time varying. This state equation is linear in $\underline{U}$, as is the performance measure:

$$L(\underline{x},\underline{U},t) = L_1(\underline{x},t) + \underline{L}_2^{T}(\underline{x},t).\underline{U} \qquad (2.36)$$

where L_1 is a scalar function and $\underline{L}_2$ is an mx1 vector valued function, both of which may be time varying and non-linear. The Hamiltonian that arises by use of the performance measure of equation (2.36) and

the state equation of (2.35) is:

$$H_p = L_1 + \underline{L}_2^T.\underline{U} + \underline{\lambda}^T[\underline{f} + B\underline{U}] \qquad (2.37)$$

Applying the control equation (2.29) to equation (2.37) gives:

$$\frac{\partial H_p}{\partial \underline{U}}^T\Big|_* = \underline{L}_2 + B^T\underline{\lambda} = \underline{0} \qquad (2.38)$$

which conveys no information about the control $\underline{U}^*$. In order to minimise the Hamiltonian according to the minimum principle, the terms in $\underline{U}$ may be collected together and, since these are independent of any of the other terms making up the Hamiltonian, they may be minimised independently of the rest of the Hamiltonian. Thus, the optimal control $\underline{U}^*$ is the control which minimises the term:

$$(\underline{L}_2^T + \underline{\lambda}^T B)\underline{U}^* \qquad (2.39)$$

The term $(\underline{L}_2^T + \underline{\lambda}^T B)$ is a 1xm vector which is dependent on the system state $\underline{x}$, the co-state $\underline{\lambda}$ and time. Defining a new mx1 vector function:

$$\underline{K}(\underline{x},\underline{\lambda},t) = \underline{L}_2 + B^T\underline{\lambda} \qquad (2.40)$$

then each of the optimal controls U_i^*, $1 \leq i \leq m$, must be such as to minimise the term $K_i.U_i$. Thus, when $K_i > 0$, U_i will take on as large a negative amplitude as is admissible and when $K_i < 0$, U_i will take on as large a positive amplitude as is admissible. When $K_i = 0$ no

control action will directly minimise the Hamiltonian and the problem becomes a singular optimal control problem.

One of the necessary conditions, equation (2.30), requires that the state equation is satisfied by the optimal control. Applying equation (2.30) to the Hamiltonian of equation (2.26) gives:

$$\frac{\partial}{\partial \underline{\lambda}} \left[L(\underline{x}^*, \underline{U}^*, t) + \underline{\lambda}^T \underline{f}(\underline{x}^*, \underline{U}^*, t) \right] - \dot{\underline{x}}^{*T} = \underline{0} \qquad (2.30)$$

$$\text{i.e. } \dot{\underline{x}}^* = \underline{f}(\underline{x}^*, \underline{U}^*, t) \qquad (2.41)$$

which is precisely the state equation of (2.8) in the specific instance that $\underline{U} = \underline{U}^*$. This condition arises because the use of Lagrange multipliers in the derivation of the necessary conditions, turns the constrained problem described by equations (2.9) and (2.10) into an unconstrained problem, equation (2.12). Thus the state equation is not a constraint of the minimisation described by equation (2.12), but it is a necessary condition for $\underline{U}^*$ to be the optimal control.

2.4.3 The adjoint or Co-State Equation

The necessary condition of equation (2.28) may be applied to the Hamiltonian of equation (2.26) as follows:

$$\frac{\partial H_p}{\partial \underline{x}} \Big|_* = -\dot{\underline{\lambda}}^{*T} \qquad (2.28)$$

$$\text{i.e. } \dot{\underline{\lambda}}^* = -\left[\frac{\partial \underline{f}^T}{\partial \underline{x}}\Big|_* \underline{\lambda}^* + \frac{\partial L^T}{\partial \underline{x}}\Big|_*\right] \qquad (2.42)$$

The form of equation, (2.42) is that of a state equation, say $\dot{\underline{\lambda}} = \underline{g}(\underline{\lambda}, \frac{\partial L}{\partial \underline{x}}\Big|_*)$ where $\underline{\lambda}$ is the state vector and $\frac{\partial L}{\partial \underline{x}}\Big|_*$ corresponds to the input or forcing term, similar in function to the control vector $\underline{U}$ of the plant state equation (2.8). In order to solve the TPBVP posed by the transversality condition, both equations (2.42) and (2.8) must be integrated. It was noted in the previous section that the forcing term $\underline{U}^*$ for the plant equations is generally a function of the co-state vector $\underline{\lambda}$. Due to the transversality condition, if an initial condition is available for the state x_i then there is no constraint on the initial condition for the co-state λ_i. As a result, there is no general solution to the TPBVP since simultaneous integration of the state and co-state variables either forwards from initial conditions or backwards from terminal conditions, is impossible. Possible numerical methods of solution will be dealt with in a later section.

2.5 Application of Pontryagin's Equations to a Linear System with Quadratic Performance Measure

Consider a linear dynamic system described by the dynamic equation:

$$\dot{\underline{x}} = A\underline{x} + B\underline{U} \text{ with } \underline{x}(t_o) = \underline{x}_o \qquad (2.43)$$

where A is an nxn matrix which may vary with time and B is an nxm

matrix which may also be time variant. Also consider the quadratic performance measure:

$$L(\underline{x},\underline{U},t) = \underline{x}^T(Q_1 + \delta(t-t_f)Q_3)\underline{x} + \underline{U}^TQ_2\underline{U} \qquad (2.44)$$

where Q_1, Q_2 and Q_3 are respectively nxn, mxm and nxn symmetrical positive definite matrices, Q_1 and Q_2 may be time varying and $\delta(t-t_f)$ is the delayed impulse function. The corresponding Hamiltonian is:

$$H_p = \underline{x}^T(Q_1 + \delta(t-t_f)Q_3)\underline{x} + \underline{U}^TQ_2\underline{U} + \underline{\lambda}^T[A\underline{x} + B\underline{U}] \qquad (2.45)$$

By transversality, the various initial conditions are $\underline{x}(t_o) = \underline{x}_o$, $\underline{x}(t_f)$ is free, $\underline{\lambda}(t_o)$ is free and $\underline{\lambda}(t_f) = \underline{0}$. Application of the control equations gives:

$$\left.\frac{\partial H_p^T}{\partial \underline{U}}\right|_* = 2Q_2\underline{U}^* + B^T\underline{\lambda}^* = 0$$

so that, provided the inverse of Q_2 exists:

$$\underline{U}^* = (-1/2)Q_2^{-1}B^T\underline{\lambda}^* \qquad (2.46)$$

is the required control law. Applying the adjoint equation to the Hamiltonian gives:

$$\dot{\underline{\lambda}}^* = -A^T\underline{\lambda}^* - 2(Q_1 + \delta(t-t_f)Q_3)\underline{x}^* \qquad (2.47)$$

for which terminal conditions exist at time t_f. The integration of equation (2.47) in reverse time from $t = t_f$ towards $t = t_o$ will

immediately cross on the impulse function at time $t = t_f$. This corresponds to a step in the $\underline{\lambda}$ versus time response from $\underline{\lambda} = \underline{0}$ at $t = t_f$ to $\underline{\lambda} = 2Q_3\underline{x}$ at time $t = t_f^-$ where t_f is the last instant prior to the impulse in forward time. Thus, equation (2.47) may be rewritten and the boundary condition adjusted to incorporate the impulse function. This gives:

$$\dot{\underline{\lambda}}^* = -A^T\underline{\lambda}^* - 2Q_1\underline{x}^* \text{ with } \underline{\lambda}^*(t_f) = 2Q_3\underline{x}^*(t_f) \qquad (2.48)$$

The control law of equation (2.46) describes an open loop control and it is desirable that a closed loop control be found. The adjoint equation (2.48) is linear in the state vector $\underline{x}$, so it would be reasonable to propose that the co-state vector $\underline{\lambda}$ is some linear time varying function of the state vector. This may be represented by the equations:

$$\underline{\lambda}^* = 2P\underline{x}^*, \text{ and } \dot{\lambda}^* = 2P\dot{\underline{x}}^* + 2\dot{P}\underline{x}^* \qquad (2.49)$$

where P is an nxn time varying matrix. The control law can then be written as:

$$\underline{U}^* = -Q_2^{-1}B^TP\underline{x}^* \qquad (2.50)$$

which is now in a closed loop form. Rearranging equations (2.49) and (2.48) and substituting for $\dot{\underline{x}}$ and, for $\underline{U}$, using equations (2.43) and (2.50) results in the Matrix Riccatti equation:

$$\dot{P}\underline{x} = -[PA + A^TP + Q_1 - PBQ_2^{-1}B^TP]\underline{x} \qquad (2.51)$$

with $P(t_f) = Q_3$. For non-trivial solutions to equation (2.51):

$$\dot{P} = -[PA + A^T P + Q_1 - PBQ_2^{-1} B^T P],\ P(t_f) = Q_3 \qquad (2.52)$$

This is a non-linear equation which governs the dynamics of the matrix P and may be solved by integrating in reverse time over the closed interval $t_f \geq t \geq t_o$. P(t) may then be used to generate the controls of equation (2.50) when the state equation is integrated forward in time.

2.6 Dynamic Programming

An alternative approach to the calculus of variation methods used above, is to use dynamic programming (2-1,2-3,2-4) to establish necessary conditions for the solution of the optimal control problem. Dynamic programming requires the definition of a Minimum Error Function, which is an explicit function of the plant state variables and time. The value of this function is the minimum value that the performance index can achieve during the remainder of the control interval. Thus, for $t_o \leq t \leq t_f$:

$$E(\underline{x}(t),t) = \min_{\underline{U}(\sigma)\in U} \int_t^{t_f} L(\underline{x}(\sigma),\underline{U}(\sigma),\sigma)d\sigma \qquad (2.53)$$

where σ is a dummy variable for the purposes of integration, L is the performance measure as before, U is the set of admissible controls and E is the minimum error function. If the closed interval $[t,t_f]$ is partitioned into two closed intervals $[t,\tau]$ and $[\tau,t_f]$, then the minimum error function can be written as:

$$E(\underline{x}(t),t) = \min_{\substack{\underline{U}(\sigma)\in U \\ [t,\tau]}} \min_{\substack{\underline{U}(\sigma)\in U \\ [\tau,t]}} \left[\int_t^{\tau} L(\underline{x}(\sigma),\underline{U}(\sigma),\sigma)d\sigma + \int_{\tau}^{t_f} L(\underline{x}(\sigma),\underline{U}(\sigma),\sigma)d\sigma \right] \tag{2.54}$$

Since the control applied during the later interval $[\tau,t_f]$ can have no effect on the contribution made to the minimum error function during the first interval $[t,\tau]$, equation (2.54) may be written as:

$$E(\underline{x}(t),t) = \min_{\substack{\underline{U}(\sigma)\in U \\ [t,\tau]}} \left[\int_t^{\tau} L(\underline{x}(\sigma),\underline{U}(\sigma),\sigma)d\sigma + E(\underline{x}(\tau),\tau) \right] \tag{2.55}$$

Provided τ is very close to t, it is possible to assume that $\underline{x}(\sigma)$ and $\underline{U}(\sigma)$ have approximately constant values during the interval $t \leq \sigma \leq \tau$, these values being approximately $\underline{x}(t)$ and $\underline{U}(t)$ respectively. An approximation for the integral on the righthand side of equation (2.55) may then be made, so that:

$$\int_t^{\tau} L(\underline{x}(\sigma),\underline{U}(\sigma),t)d\sigma \simeq (\tau - t)L(\underline{x}(t),\underline{U}(t),t) \tag{2.56}$$

Likewise, provided τ is close to t, an approximation for the minimum error function to the righthand side of equation (2.55) may be made such that:

$$E(\underline{x}(\tau),\tau) \simeq E(\underline{x}(t),t) + (\tau - t)\frac{d}{dt}[E(\underline{x}(t),t)] \tag{2.57}$$

Equations (2.55) to (2.57) may then be combined to give:

$$E(\underline{x}(t),t)$$

$$\simeq \min_{\substack{\underline{U}(\sigma)\in U \\ [t,\tau]}} \left[(\tau-t)L(\underline{x}(t),\underline{U}(t),t) + (\tau-t)\frac{d}{dt}[E(\underline{x}(t),t)] + E(\underline{x}(t),t)\right] \tag{2.58}$$

The total time derivative in these last two equations may be expanded as:

$$\frac{d}{dt}\left[E(\underline{x}(t),t)\right] = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial \underline{x}}\ \dot{\underline{x}} \tag{2.59}$$

$$= \frac{\partial E}{\partial t} + \frac{\partial E}{\partial \underline{x}}\ \underline{f}(\underline{x},\underline{U},t)$$

by substitution of the state equation (2.8). Incorporating equation (2.59) into equation (2.58) leaves only two terms which are subject to the minimisation process, $\frac{\partial E}{\partial \underline{x}}\ \underline{f}(\underline{x},\underline{U},t)$ and $L(\underline{x},\underline{U},t)$. Rearranging (2.58) then results in the dynamic programming condition for minimum performance index:

$$-\frac{\partial E}{\partial t} = \min_{\underline{U}(t)\in U}\left[\ L(\underline{x}(t),\underline{U}(t),t) + \frac{\partial E}{\partial \underline{x}}\ \dot{\underline{x}}\ \right] \tag{2.60}$$

It naturally follows that the optimal control $\underline{U}^*(t)$ satisfies this condition, so:

$$-\frac{\partial E}{\partial t}\Big|_* = L(\underline{x}^*,\underline{U}^*,t) + \frac{\partial E}{\partial \underline{x}}\Big|_*\ \underline{f}(\underline{x}^*,\underline{U}^*,t) \tag{2.61}$$

which is known as the Hamilton–Jacobi equation (2-3). If the minimum on the righthand side of equation (2.60) exists for an admissible control, then an expression for the optimal control $\underline{U}^*$ is given by equating the partial derivatives of the righthand side of equation (2.60) with respect to $\underline{U}$, to zero. This gives the control law equation as:

$$\frac{\partial L}{\partial \underline{U}}\bigg|_* + \frac{\partial E}{\partial \underline{x}}\bigg|_* \frac{\partial \underline{f}}{\partial \underline{U}}\bigg|_* = \underline{0} \qquad (2.62)$$

The functions $\underline{f}$ and L are known since they are part of the specification of the problem and so their partial derivatives, with respect to the control vector, can be evaluated. The function $E(\underline{x}(t),t)$ is not given explicitly and the partial derivatives $\frac{\partial E}{\partial t}$ and $\frac{\partial E}{\partial \underline{x}}$ cannot be evaluated. Consequently, there is no general solution to the Hamilton–Jacobi equation, although it may be solved for specific instances where a general form can be specified for E in terms of $\underline{x}(t)$ and t. In particular, if $E(\underline{x}(t),t)$ is given the quadratic form:

$$E(\underline{x}(t),t) = \underline{x}^T K(t)\, \underline{x}(t) \qquad (2.63)$$

where K(t) is an nxn time varying matrix, then for a linear system:

$$\dot{\underline{x}} = A\underline{x} + B\underline{U} \qquad (2.43)$$

as in section 2.5, and a quadratic performance measure:

$$L(\underline{x}\ \underline{U}\ t) = \underline{x}^T(Q_1 + \delta(t-t_f)Q_3)\underline{x} + \underline{U}^T Q_2 \underline{U} \qquad (2.44)$$

such that:

$$E(\underline{x}(t),t)$$

$$= \min_{\substack{\underline{U}(\sigma)\in U \\ [t,t_f]}} \left[\underline{x}^T(t_f)Q_3\underline{x}(t_f) + \int_t^{t_f} [\underline{x}^T(\sigma),Q_1(\sigma)\underline{x}(\sigma) + \underline{U}^T(\sigma),Q_2(\sigma),\underline{U}(\sigma)]d\sigma\right] \qquad (2.64)$$

It can be shown that K(t) in equation (2.63) is identical to the P(t) which is the solution to the Matrix Riccatti equation (2.52). This result may be immediately applied to finding the minimum value of the performance index:

$$I = \min_{\underline{U}(t)\in U} \left[\int_{t_o}^{t_f} (\underline{x}^T Q_1 \underline{x} + \underline{U}^T Q_2 U)dt + \underline{x}^T(t_f)Q_3\underline{x}(t_f) \right]$$

since

$$I = E(\underline{x}(t_o),t_o) = \underline{x}(t_o)^T P(t_o)\ \underline{x}(t_o) \qquad (2.65)$$

It can also be shown (2-3) that the dynamic programming approach to optimal control is equivalent to the calculus of variation approach; in fact, the Euler Lagrange equation (2.22) may be derived from the Hamilton-Jacobi equation (2.61).

2.7 Solution Methods

It has been noted in previous sections that the presence of the

transversality condition gives rise to a TPBVP which in general makes simultaneous integration of the state and co-state equations impossible. In general, for an nth order system, the combination of state and co-state equations results in 2n simultaneous differential equations with n initial conditions at the beginning of the control interval and n terminal conditions at the end of the control interval. When the initial state $\underline{x}(t_o)$ is completely specified and the terminal state $\underline{x}(t_f)$ is unspecified, the state equations may be integrated forward in time from t_o to t_f and since, by transversality, $\underline{\lambda}(t_f)$ is known, the co-state equations may be integrated in reverse time from t_f to t_o. Integration of this type must be performed numerically on a digital computer since it is necessary to store state information from the forward integration to be used in the forcing function for the reverse integration of the co-state equations. Initially, some arbitrary choice must be made for the control function $\underline{U}(t)$ over the entire control interval. Using this initial control, the state equations are integrated forward in time and the state trajectory stored for the subsequent integration of the co-state equations. After the first iteration, provided the control equation permits the control to be described as a function of the co-state variables, $\underline{\lambda}$, a new control may be derived from the co-state trajectory stored during integration. Repeated iterations of the above procedure are stable and convergent (2-3). However, this technique, known as control function iteration, may only be used when the control equation yields a control law and the boundary conditions are such that $\underline{x}(t_o)$ and $\underline{\lambda}(t_f)$ are completely specified.

2.7.1 Relaxation

Relaxation methods (2-3) are applicable to a wide class of optimal control problems and will always converge to a local minimum in the performance index value. Consider the nth order non-linear system:

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{U}, t) \tag{2.66}$$

where $\underline{x}$ is an nx1 state vector, $\underline{U}$ is an mx1 control vector and $\underline{f}$ is an nx1 vector valued function. Consider also the performance index:

$$I = \phi(\underline{x}(t_f)) + \int_{t_o}^{t_f} L(\underline{x}, \underline{U}, t)\, dt \tag{2.67}$$

where ϕ is a scalar function of the terminal state and L is some performance measure. If $\tilde{\underline{x}}(t)$ and $\tilde{\underline{U}}(t)$ are nominal state and control trajectories which yield a performance index value of $\tilde{I}$, a small change in the nominal control trajectory, $\delta\underline{U}(t)$, will result in a small change in the state trajectory $\delta\underline{x}(t)$ and a small change in the performance index value δI. The changes $\delta\underline{x}$ and $\delta\underline{U}$ are related by the linearised state equation:

$$\delta\dot{\underline{x}} = \frac{\partial \underline{f}}{\partial \underline{x}}\,\delta\underline{x} + \frac{\delta \underline{f}}{\partial \underline{U}}\,\delta\underline{U} \text{ with } \delta\underline{x}(0) = \underline{0} \tag{2.68}$$

and the corresponding change in performance index is:

$$\delta I = \frac{\partial \Phi}{\partial \underline{x}} \delta \underline{x}(t_f) + \int_{t_o}^{t_f} \left[\frac{\partial L}{\partial \underline{x}} \delta \underline{x} + \frac{\partial L}{\partial \underline{U}} \delta \underline{U} \right] dt \qquad (2.69)$$

The term in $\delta \underline{x}$ may be eliminated from the integral by use of the co-state equation (2.42):

$$\frac{\partial L}{\partial \underline{x}} = -\dot{\underline{\lambda}}^T - \underline{\lambda}^T \frac{\partial \underline{f}}{\partial \underline{x}} \quad \text{with } \underline{\lambda}^T(t_f) = \frac{\partial \Phi}{\partial \underline{x}} \Big|_{t = t_f} \qquad (2.70)$$

Substituting equation (2.70) into (2.69), then integrating the term in $\dot{\underline{\lambda}}$ by parts and finally substituting equation (2.68) for $\delta \dot{\underline{x}}$ gives:

$$\delta I = \left[\frac{\partial \Phi}{\partial \underline{x}} - \underline{\lambda}^T(t_f) \right] \delta \underline{x}(t_f) + \underline{\lambda}^T(t_o) . \delta \underline{x}(t_o) + \int_{t_o}^{t_f} \left[\underline{\lambda}^T \frac{\partial \underline{f}}{\partial \underline{U}} + \frac{\partial L}{\partial \underline{U}} \right] \delta U \, dt \qquad (2.71)$$

Substitution of the initial conditions into (2.71) make all the terms outside the integral equal zero, thus:

$$\delta I = \int_{t_o}^{t_f} \left[\underline{\lambda}^T . \frac{\partial \underline{f}}{\partial \underline{U}} + \frac{\partial L}{\partial \underline{U}} \right] \delta \underline{U} \, dt \qquad (2.72)$$

If $\delta \underline{U}$ is considered to be an impulse function $\delta \underline{U}^\tau$ at some time τ in the closed interval $t_o \leq \tau \leq t$ then:

$$\frac{\partial I}{\partial \underline{U}^\tau} = \lambda^T(\tau) . \frac{\partial \underline{f}}{\partial \underline{U}} \Big|_{t=\tau} + \frac{\partial L}{\partial \underline{U}} \Big|_{t=\tau} \qquad (2.73)$$

The term on the lefthand side of equation (2.73) gives the

gradient of the performance index with respect to the control vector at the instant $t = \tau$. This gradient may be used to minimise the performance index with respect to the control $\underline{U}(t)$ at each instant in time by one of numerous gradient methods. Functional minimisation by gradient methods is discussed in Chapter 6. The iterative procedure adopted by a relaxation method is:

1. Assume an initial control $\tilde{\underline{U}}(t)$ for the interval $t_o \leq t \leq t_f$.
2. Numerically integrate the state equations forward in time storing the state $\tilde{\underline{x}}(t)$ at each instant in time and evaluate the performance index $\tilde{I}$.

3. Integrate the adjoint equations in reverse time, calculating and storing $\partial I/\partial \underline{U}^T$ at each instant using equation (2.73).

4. Use a gradient technique to improve the control $\tilde{\underline{U}}$ and reduce the performance index $\tilde{I}$. For example, the use of a steepest descent technique would require repeated forward integration of the state equations with the modified control vector:

$$\underline{U}(t) = \tilde{\underline{U}}(t) - C \frac{\partial I}{\partial \underline{U}^T}\Bigg|_{\tau=t}^{T} \qquad (2.74)$$

to establish the value of the scalar C which minimises $\tilde{I}$.

5. On conveyance of the gradient technique, the state equations have just been integrated forward in time. The procedure now returns to step 3 and the integration of the co-state equations in

reverse time in order to establish a new set of partial derivatives $\partial I/\underline{U}^T$ from which the next search direction of the gradient minimisation will be constructed.

In accordance with the minimum principle, controls must be kept within their admissible range during step 4. The relaxation method described above requires the initial conditions for the state equation at time t_o and the terminal conditions for the co-state equations at time t_f to be completely specified in order that the forward and reverse integrations can be performed. In common with the 'control function iteration' method described, the optimal controls derived by this method are open loop and, as such, are applicable only to the particular initial conditions $\underline{x}(t_o)$ and the control interval for which the optimal control has been calculated. This type of control law is unsuitable for the control of generating plant subjected to large external disturbances.

2.7.2 Numerical Solution of the Matrix Riccatti Equation

In Section 2.5, Pontryagin's equations, (2.28) to (2.31) were used to derive an optimal control for the linear system described by equation (2.43) in association with the performance measure of equation (2.44). The formulation of this optimal control forces a closed loop solution by the definition of a linear relationship between the state and co-state vectors

$$\underline{\lambda} = 2P\underline{x} \qquad (2.49)$$

The nxn time varying matrix P was shown to be governed by the

matrix riccatti equation:

$$\dot{P} = -[PA + A^TP + Q_1 - PBQ_2^{-1} B^TP] \text{ with } P(t_f) = Q_3 \qquad (2.52)$$

Given equation (2.52) and its terminal condition, it is possible to obtain $P(t)$, $t < t_f$, by integration of the matrix riccatti equation in reverse time. This results in the closed loop control described by equation (2.50):

$$\underline{U}^* = -Q_2^{-1}B^TP\underline{x} = -K\underline{x} \text{ with } K = Q_2^{-1}B^TP \qquad (2.50)$$

which, in general, requires full state feedback using the time varying gain matrix, K.

Much has been written regarding the solution of the matrix riccatti equation (2-5) to (2-9) and, in particular, the solution of the so-called stationary matrix riccatti equation. This stationary solution arises when the terminal time, t_f, of the control interval is extended towards infinity, with a time invariant plant and constant matrices Q_1 and Q_2. The matrix P becomes time invariant so equation (2.52) reduces to:

$$PA + A^TP + Q_1 - PBQ_2^{-1}B^TP = 0 \qquad (2.75)$$

Provided the matrix pair $[Q_1, A]$ are completely detectable, that is, all unstable modes are observable, integration of equation (2.52) in reverse time is completely stable and will converge upon the unique non-negative definite solution (2-6) of equation (2.75). If the

pair[Q_1,A] are completely observable this solution will be positive definite.

Potter (2-5) showed that the solution of equation (2.75) could be inferred from the eigenvalues of the 2nx2n matrix:

$$E = \begin{bmatrix} A & : & -BQ_2^{-1}B^T \\ \cdots\cdots & : & \cdots\cdots\cdots \\ -Q_1 & : & -A^T \end{bmatrix} \qquad (2.76)$$

provided these eigenvalues are distinct. If λ_i is an eigenvalue of E and $\underline{a}_i$ is the corresponding eigenvector which may be partitioned as:

$$\underline{a}_i = \begin{bmatrix} \underline{b}_i \\ \underline{c}_i \end{bmatrix} \qquad (2.77)$$

where $\underline{b}_i$ and $\underline{c}_i$ are each nx1 vectors, then it can be shown (2-5 and 2-6) that:

$$P = [\underline{c}_1 \cdots \underline{c}_n]\,[\underline{b}_1 \ldots \underline{b}_n]^{-1} \qquad (2.78)$$

is a solution of the stationary matrix riccatti equation (2.75). The 'n' eigenvectors $\underline{a}_1$ to $\underline{a}_n$ used to establish this solution may be chosen arbitrarily from the 2n eigenvectors of E. However, it can also be shown that corresponding eigenvalues λ_1 to λ_n are also eigenvalues of the closed loop system matrix.

$$G = A - BQ_2^{-1}B^TP \qquad (2.79)$$

Further, if λ_i is an eigenvalue of E, then so is $-\lambda_i$ and if λ_i is

complex then λ_i^* and $-\lambda_i^*$ are also eigenvalues of E, where λ_i^* is the complex conjugate of λ_i. In order that the closed loop system matrix, G of equation (2.79) is stable, the eigenvalues λ_i to λ_n included in the solution, equation (2.78) must have negative real parts and since only n of the 2n possible eigenvalues have negative real parts (assuming there are no entirely imaginary eigenvalues) there is only one way to choose P such that G is stable. Martensson (2-6) relaxed the requirement that E must have distinct eigenvalues by the introduction of generalised eigenvectors. These generalised eigenvectors are defined such that if λ_i is an eigenvalue of E with multiplicity h then

$$(E-\lambda_i I)\underline{a}_i = \underline{0} \qquad \text{defines the rank one eigenvector}$$

$$(E-\lambda_i I)\underline{a}_{i+1} = \underline{a}_i \qquad \text{defines the rank two eigenvector}$$

$$(E-\lambda_i I)\underline{a}_{i+h-1} = \underline{a}_{i+h-2} \qquad \text{defines the rank h eigenvector} \qquad (2.80)$$

where I is the nxn identity matrix. If it is required to include the eigenvector $\underline{a}_{i+r-1}$ of rank r, $1 < r \leq h$, in the solution, then all eigenvector of rank less than r corresponding to λ_i must also be included in the solution. If λ_i is a stable, unobservable mode of the matrix pair $[Q_1, A]$ then the solution matrix P is not positive definite, but is non-negative definite. If λ_i is an unstable, unobservable mode of $[Q_1, A]$ then the solution matrix is non-negative definite, but is no longer unique. In general, if there are m unstable, unobservable modes, then there are 2^m possible non-negative definite solutions to equation (2.75). In all cases, the matrix pair

[A,B] must be stabilisible, that is, all unstable modes must be controllable, otherwise it will not be possible to choose the eigenvalues such that the matrix $[b_1 \ . \ . \ . \ b_n]$ is non-singular and so, by equation (2.78) a solution will not exist. The insight this gives to the solution of the stationary matrix riccatti is that, where a mode of the open loop system is unobservable in the matrix pair $[Q_1,A]$, i.e. it makes no contribution to the performance index, there is no positive definite solution. Further, if this unobservable mode is unstable, the solution is not unique. The various non-negative definite solutions that arise with unobservable, unstable modes correspond to optimal controls which may stabilise some or all of these modes. Each of these solutions is a local minimum in the minimisation process. Stabilisation of unstable modes will clearly require more control effort on the part of the controller and the solution with the largest performance index value, $\underline{x}(t_o)^T P \underline{x}(t_o)$ by equation (2.65) corresponds to the solution with all the closed loop modes stabilised. Thus, depending on the observability characteristics of the pair $[Q_1,A]$, it is possible to perform integration of equation (2.52) in reverse time to converge upon one of a number of distinct non-negative definite solutions. Martensson (2-6) shows that the choice of the terminal condition, $P(t_f) = Q_3$, influences which of these solutions this integration will converge upon and that, if Q_3 is an identity matrix, the integration will converge on the largest solution, i.e. the one that stabilises all the closed loop modes.

An alternative means of solving the stationary matrix riccatti equation is by iterative methods (2-8 and 2-9). Equation (2.75) may be rearranged as follows:

$$[A-BQ_2^{-1}B^TP]^T P+P [A-BQ_2^{-1}B^TP] = -(Q_1 + PBQ_2^{-1}B^TP) \quad (2.81)$$

$$A_1^TP + PA_1 = -(Q_1 + KQ_2K) = -Z \quad (2.82)$$

where $A_1 = A+BK$, $K = -Q_2^{-1}B^TP$ and $Z = Q_1 + KQ_2K$.

An initial value for K is estimated and, from this, values for A_1 and Z are calculated. With these values for A_1 and Z, equation (2.82) is solved to give the matrix P, from which a new value for K may be calculated and the process repeated until the solution converges. It can be shown (2-10) that, subject to similar observability and controllability conditions as those above, if the initial gain matrix K_0 is chosen such that A_1 is a stable closed loop system matrix, the sequence of values of the matrix P, known as Newton's sequence, will converge to the unique solution of equation (2.75).

Stabilisation of the system:

$$\dot{\underline{x}} = (A+BK)\underline{x} \quad (2.83)$$

prior to the iterative solution of the stationary matrix riccatti equation may be achieved in a number of ways. In the trivial case where the matrix A is already stable, the matrix K may be set to zero. A stabilisation algorithm due to Kleinman (2-10) applicable to only completely observable and controllable linear systems was extended by Sandell (2-9) to include systems which are only stabilisible. These two algorithms are particular cases of an algorithm attributed to Bass (2-11) and extended by Armstrong (2-12) so that the system need only be stabilisible.

These algorithms are based on the theorem that for any constant nxn matrix A, the matrix $-(A+\beta I)$ is a stability matrix provided $\beta > ||A||$, where $||A||$ is any matrix norm of A which has a consistent vector form and I is the nxn identity matrix. By Lyapunov's stability theory (2-13) for any constant nxn matrix A, there is a positive definite solution, Z, to the Lyapunov matrix equation:

$$AZ + ZA^T = -Q \tag{2.84}$$

if and only if, A is a stable system matrix, where Q is any nxn positive definite matrix. For a completely controllable matrix pair [A,B], the Bass (2-11) algorithm states that the control law:

$$\underline{U} = K\underline{x} \text{ with } K = -B^T Z^{-1} \tag{2.85}$$

will stabilise the system, equation (2.43), where $Z = Z^T > 0$ is the positive definite solution to the Lyapunov equation:

$$[-(A+\beta I)]Z + Z[-(A+\beta I)]^T = -2BB^T \tag{2.86}$$

and $\beta > ||A||$. Armstrong's modification relaxes this requirement for complete controllability to a requirement for stabilisability, this results in a modification to the gain matrix K which becomes:

$$K = -\beta^T Z^+ \tag{2.87}$$

where $Z = Z^T > 0$ is the unique positive definite solution of equation (2.86) and Z^+ is the pseudo inverse of Z. Equation (2.86) and the modified matrix riccatti equation (2.82) are of the same form, each

requiring the solution of n(n+1)/2 linear algebraic equations, since both P and Z are unknown, symmetrical nxn matrices.

Many authors (2-14 to 2-18) have used a linear model of the generating plant with quadratic performance indices to obtain optimal controls by the solution of the stationary matrix riccatti equation. This type of solution results in a feedback controller with constant coefficients. However, in general, it is necessary to feedback all the states used to represent the system. These solutions can only be said to be optimal for small disturbances that do not cause large deviations about the operating point for which the system was linearised. In general, those authors obtained improvements in the transient performance of optimally controlled machines over conventionally controlled machines when small disturbances were applied to linearised models. When the same control law is applied to a non-linear representation of the generating plant, some form of sub-optimal performance is expected. However, it has been shown (2-16 and 2-18) that use of the optimal control for the linearised system can improve the transient performance of the corresponding non-linear system when subjected to large disturbances. In order that a useful optimal solution is obtained, it is necessary to formulate the system representation in terms of system states that may be measured, otherwise the control derived may not be applied to a real plant. Even then some of the states may be inaccessible and a modification can be made (2-18) to the matrix riccatti equation such that the control law becomes:

$$\underline{U}^*(t) = -Q_2^{-1}B^T P M \underline{x} \qquad (2.88)$$

where M is an nxn diagonal matrix in which the element M_{ii}, $1 \leq i \leq n$,

is either 1 if the state x_i is available to the controller, or zero if it is not, x_i being the ith element of the state vector. The modified matrix riccatti equation then becomes:

$$-\dot{P} = PA + A^TP + Q_1 + MPBQ_2^{-1}B^TP(M-2I) \qquad (2.89)$$

which may be solved by integration in reverse time as before.

The matrix riccatti equation can, and has been, applied successfully to the optimal control of generating plant. The controls obtained are only optimal for small deviations about a particular operating point. However, as sub-optimals controls, performance improvements can still be obtained in the presence of large disturbances.

2.7.3 The Minimum Principle and Singular Solutions

Some mention has already been made in section 2.4.2 concerning the application of Pontryagin's minimum principle in situations where the control equation (2.29) yields no information about the optimal control. In general, if the control vector $\underline{U}$ appears linearly in both the performance measure and the state equations and, in addition, each element of the optimal control is bounded, then the optimal control is of a Bang-Bang nature. In practice, there are always finite limits imposed on control magnitudes so this latter condition is always present. Thus, even though the control law may not be given explicitly by application of the control equation, its form may be inferred via the minimum principle. Chana (2-19) inferred that the

optimal excitation control of a synchronous generator may be 'bang-bang' in nature and used direct minimisation (described in the next section) to determine the optimum times at which to switch the field voltage of the generator from its positive to its negative ceiling and vice versa. Following these investigations, Chana (2-19) used a similar technique to investigate a closed loop bang-bang control.

It is also possible that the coefficient of the control vector in the Hamiltonian is zero for finite periods of time during the control interval. If this occurs, the minimum principle can offer no help since it is not possible to infer the effect of the control vector on the Hamiltonian by direct inspection. The solution to this type of problem is known as a Singular control. Thus, an optimal control may either be soluble via Pontryagin's equations, bang-bang by application of the minimum principle when the control equation yields no information about the control vector, singular when the minimum principle cannot be applied, or some combination of bang-bang and singular.

Consider the Hamiltonian:

$$H_p(\underline{x},\underline{U},\underline{\lambda},t) = L_1(\underline{x},\underline{\lambda},t) + \underline{\phi}^T(\underline{x},\underline{\lambda},t).\underline{U} \qquad (2.90)$$

where the terms in the control vector $\underline{U}$ have been gathered together into the mx1 vector valued function $\underline{\phi}$. If $\underline{\phi}$ is a null vector for any period of time then the optimal control problem becomes singular and, when it is non-zero, the minimum principle may be applied. While the problem is singular:

$$\underline{\Phi} = \dot{\underline{\Phi}} = \ddot{\underline{\Phi}} \cdots = \underline{0} \qquad (2.91)$$

and expanding the first time derivative of $\underline{\Phi}$ gives:

$$\dot{\underline{\Phi}} = \frac{\partial \underline{\Phi}}{\partial t} + \frac{\partial \underline{\Phi}}{\partial \underline{x}} \dot{\underline{x}} + \frac{\partial \underline{\Phi}}{\partial \underline{\lambda}} \dot{\underline{\lambda}} = \underline{0} \qquad (2.92)$$

Depending on the nature of $\underline{\Phi}$, this may or may not result in the reintroduction of $\underline{U}$ into an equation which can be rearranged into a control law via the state and co-state equations. There is no guarantee that this or higher time derivatives of $\underline{\Phi}$ will yield a control law and, even if a control law is obtained, it is not necessarily the optimal control. As yet, there is no general set of necessary conditions to determine the optimality of a singular control but an optimal singular control must satisfy both the generalised Legendre Clebtch condition and the Jacobson condition (2-20).

2.7.4 Direct Minimisation or Specific Optimal Control

It has already been noted that, in general, the solution of Pontryagin's equations (2.28) to (2.31) leads to an open loop optimal control, which is undesirable. The matrix riccatti equation leads to closed loop optimal controllers which, in general, have time varying coefficients and this time variation is itself open loop. In the particular case of linear time invariant systems, matrix riccatti solutions result in time invariant closed loop controllers.

It is also possible to assume some general form of control law

which is a function of the measurable outputs of the system and some as yet unspecified parametric values, say, parameter vector $\underline{s}$. The resultant controller must be sub-optimal since the form of the controller is assumed and the minimised value of the performance index may not be the smallest possible by control means. However, such a controller is in a desirable closed loop form and is constrained to use plant variables which are measurable. Numerical techniques for performing the minimisation of the performance index of equation are discussed further in Chapter 6.

2.8 Summary

Having introduced the fundamental concepts of performance measure and performance index, conditions necessary for a control to be optimal have been derived. These conditions are not sufficient in themselves and there exists no general solution to the optimal control problem, due to the presence of the Two Point Boundary Value Problem (TPBVP) imposed by the transversality condition. The most general application of the necessary conditions, whether in the form of Pontryagin's equations, or as the dynamic programming conditions, leads to open loop control laws. In many cases it is impossible to derive a practical control law, since solution of the necessary conditions either fails to yield an explicit control law, or the TPBVP prevents simultaneous solution of the state and adjoint equations. Except in the case where the matrix riccatti equation may be applied, the main value of the conditions necessary for optimal control are that they indicate the form of the control. If the control equation yields a control law, then an optimal control may be expressed in terms of the state and co-state variables. Otherwise it

may be posssible to invoke the minimum principle to obtain a bang-bang controller. Only in the case of a linear, time invariant plant, optimised for a quadratic performance index, over an infinite control interval, is it possible to obtain a linear, time invariant, optimal feedback controller.

The matrix riccatti equation may not be applied to a non-linear plant, such as a power system. However, the type of solution obtained from matrix riccatti suggests that a sub-optimal control may be obtained in the form of a state feedback controller. Unfortunately, not all the system states are measurable and some are therefore not available for control purposes. This motivates the search for a sub-optimal controller which uses the minimum number of feedback signals. Such sub-optimal controllers may be established by the use of 'specific' optimal control techniques, where the form of the controller is predetermined and a parametric optimisation is required to establish the specific values of the controller parameters. The reduction in the number of feedback signals used is discussed in Chapter 6.

Finally, where a control is bounded, for example, the ceiling voltage of an AVR, or the rate limits on governor value actuation, application of Pontryagin's minimum principle suggests that it is likely that the optimal control will, to some extent, be bang-bang in nature. Thus, when a power system is subjected to a severe transient, it may be expected that an optimal control will drive the field voltage rapidly between its upper and lower ceilings.

References

2-1 Merriam, C.W.: "Optimisation theory and the design of feedback control systems", McGraw-Hill, 1964.

2-2 Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V. and Mishchenko, E.F.: The mathematical theory of optimal processes", Interscience, New York, 1962.

2-3 Speedy, C.B., Brown, R.F. and Goodwin, G.C.: "Control theory: Identification and optimal control", Oliver & Boyd, Edinburgh 1970.

2-4 Bellman, R.: "Dynamic programming", Princeton University Press, 1957.

2-5 Potter: "Matrix quadratic solutions", SIAM, J.Appl.Math, Vol.14, pp.496-501, 1966.

2-6 Martensson, K.: "On the matrix riccatti equation", Information Sciences, Vol.3, pp.17-49, 1971.

2-7 Kalman, R.E.: "When is a linear control system optimal?", J.Basic Engineering, Vol.86, pp.51-60, 1967.

2-8 Kleinman, D.L.: "On an iterative technique for the solution of riccatti equation computations", Trans.IEEE, Vol.AC-13, pp.114-115, 1968.

2-9 Sandell, N.R., Jr.: "On Newton's method for riccatti equation solution", Trans,IEEE, Vol.AC-19, pp.254-255, 1974.

2-10 Kleinman, D.L.: "An easy way to stabilise a linear constant system", Trans,IEEE, Vol.AC-15, pp.692-693, 1970.

2-11 Bass, R.W.: "Lecture notes on control synthesis and optimisation", presented at NASA Langley Research Center, Hempton, Va, Aug. 21 to Sept. 2, 1961.

2-12 Armstrong, E.S.: "An extension of Bass' algorithm for stabilising linear control systems", Trans,IEEE, Vol.AC-20, pp.153-154, 1975.

2-13 Kalman, R.E. and Bertram, J.E.: "Control system analysis and design via the 'second method' of Lyapunov. Continuous time systems", J.Basic Engineering, pp.371-393, 1960.

2-14 Yu, Y.N., Vonsuriya, K and Wedman, L.N." "Application of an optimal control theory to a power system", Trans.IEEE, PAS-89, pp.55-62, 1970.

2-15 Yu, Y.N. and Siggers: "Stabilisation and optimal control for a power system", Trans,IEEE, PAS-90, pp.1469-1481, 1971.

2-16 Anderson, J.H.: "The control of a synchronous machine using optimal control theory", Proc.IEE, Vol.59, pp.25-35, 1971.

2-17 Ramamoorty, M and Arymugam, M.: "Design of optimal constant

output feedback controllers for a synchronous machine", Proc.IEE, Vol.119, pp.257-260, 1972.

2-18 Humpage, W.D., Smith, J.R. and Rogers, G.J.: "Application of dynamic optimisation to synchronous generator excitation controllers", Proc.IEE, Vol.120, pp.87-93, 1973.

2-19 Chana, G.S.: "Optimal control studies of power system stability", PhD Thesis, University of Bath, 1977.

2-20 Bell, D.J. and Jacobson, D.H.: "Singular optimal control problems, Academic Press, 1975.

CHAPTER 3

SYSTEM REPRESENTATION

3.1 Introduction

Two power system configurations are studied in this dissertation. These correspond to, first, a single synchronous generator connected to an infinite busbar via a generator transformer and a transmission line and, secondly, a finite busbar system consisting of a single synchronous generator, connected to an inductive load. Three different prime movers have also been used in these studies, namely, a steam driven turbine, a water driven turbine and a diesel engine.

Three sets of plant data have been used and these may be found in Appendix A. Two sets of plant data were supplied by the CEGB and correspond to a typical 500MW turbogenerator such as those installed at Pembroke Power Station and a 330MW hydrogenerator such as those installed at the Dinorwig Pump Storage scheme. The data used for the studies carried out on the finite busbar system was supplied by the Royal Naval Engineering College (RNEC) at Manadon and is based on values typical of those found in the power system of a warship.

This Chapter develops the dynamic and algebraic equations associated with the synchronous machine, the excitation system, the prime movers and governors and the transmission system. Consideration is then given to numerical solution of these equations and the plant equations are finally transformed into a suitable form

for numerical solution.

3.2 Synchronous Machine Representation

The synchronous machine model used in these studies is based on Park's two-axes equations for an idealised machine (3-1). These equations are obtained by performing an axis transformation on the original three-phase equations. This transformation is known as Parks transform and resolves the stator quantities onto a set of rotating axis which lie along, and perpendicular to, the direction of the main field flux vector. These axes are known as the direct and quadrature axis respectively. Zero sequence components are not considered in the representation given here since, for the purposes of this study, the synchronous machine is assumed to be on balanced three-phase operation, so the zero sequence components are zero in value. The effect of eddy current losses in the rotor iron and damper windings are considered to be adequately represented by one short circuit winding in each of the rotor axes. The idealised machine is further subject to the following assumptions (3-2):

(i) The current flowing in any winding is assumed to set up an m.m.f. which has a sinusoidal distribution in space around the air gap.

(ii) The effects of hysteresis may be ignored.

(iii) A component of m.m.f. acting in the direct axis only, is assumed to give rise to a sinusoidally distributed flux wave in the direct axis only. Likewise, an m.m.f. acting

in the quadrature axis only, is assumed to give rise to a sinusoidally distributed flux wave in the quadrature axis only.

Fig. 3.1 illustrates schematically the layout of the generator windings. The algebraic signs shown represent generator action and direct and quadrature axes flux linkage and voltage equations may be written as follows.

The flux linkage equations:

$$\psi_{fd} = x_{ffd}.i_{fd} - x_{ad}.i_d + x_{ad}.i_{1d} \qquad (3.1)$$

$$\psi_d = x_{ad}.i_{fd} - x_d.i_d + x_{ad}.i_{1d} \qquad (3.2)$$

$$\psi_{1d} = x_{ad}.i_{fd} - x_{ad}.i_d + x_{11d}.i_{1d} \qquad (3.3)$$

$$\psi_q = -x_q.i_q + x_{aq}.i_{1q} \qquad (3.4)$$

$$\psi_{1q} = -x_{aq}.i_q + x_{11q}.i_{1q} \qquad (3.5)$$

The voltage equations:

$$V_{fd} = P\psi_{fd}/\omega_o + r_{fd}.i_{fd} \qquad (3.6)$$

$$V_d = P\psi_d/\omega_o - \omega\psi_q/\omega_o - r_a.i_d \qquad (3.7)$$

$$0 = P\psi_{1d}/\omega_o + r_{1d}.i_{1d} \qquad (3.8)$$

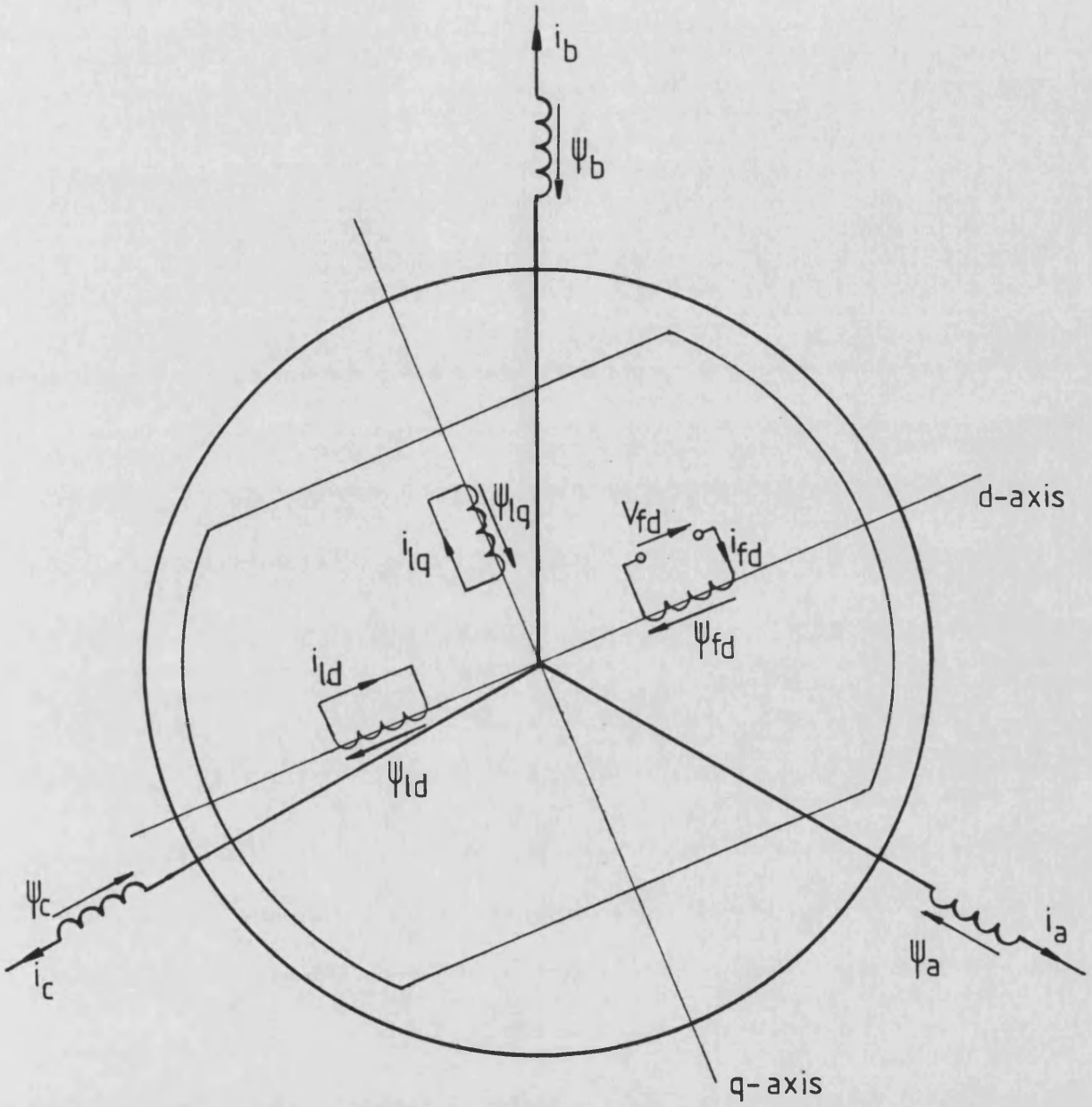


Fig 3.1 Generator Windings

$$V_q = P\psi_q/\omega_o + \omega\psi_d/\omega_o - r_a.i_q \qquad (3.9)$$

$$0 = P\psi_{1q}/\omega_o + r_{1q}.i_{1q} \qquad (3.10)$$

The air gap torque is given by:

$$T_{elec} = \psi_d.i_q - \psi_q.i_d \qquad (3.11)$$

All the above equations are given in per-unit form. The magnitude of the terminal voltage may be obtained from:

$$V_t = \sqrt{(V_d^2 + V_q^2)} \qquad (3.12)$$

Associated with the synchronous machine are the mechanical dynamics of the rotor shaft which are described by:

$$MP^2\delta = T_{out} - T_{elec} - K_d\, P\delta/\omega_o \qquad (3.13)$$

where K_d is a constant to account for mechanical damping, and T_{out} is the prime mover output torque.

It has been shown (3-3) that iron saturation has little effect on the transient stability of the machine and so such effects have been neglected. The above equations give a seventh order representation of the machine (five from the electrical equations and two from the rotor dynamics). This representation takes into account the decay of stator flux linkages which result in power frequency oscillations in the machine axis voltage and currents after a severe disturbance. It also takes into account the initial braking torque, caused by the

direct current induced in the armature, which decays quickly with the armature time constant. This torque can result in an initial decrease in load angle prior to the large increase resulting from the sudden loss of demand for power caused by the fault. Unfortunately, since power frequency terms are present in the axis quantities using this representation, the time steps for numerical integration must be small enough to simulate these terms accurately. This obviously increases the overall computing time necessary to simulate the effect of a disturbance and, when the simulation has to be performed repeatedly in order to perform an optimisation of the system transient response, the computing effort will be appreciable. Also, the state information generated by integration of equations (3.6) to (3.13), along with the solution of the algebraic equations (3.1) to (3.5), are flux linkages, which are not readily measurable as feedback signals. These two drawbacks motivate a rearrangement of these equations into operational form by eliminating the rotor currents. This, followed by some algebraic manipulations, results in the following set of equations(3-3):

$$Pe''_d = [(x_q - x''_q).i_q - e''_d]/T''_{qo} \qquad (3.14)$$

$$Pe'_q = [V_f - (x_d - x''_d).i_d - e'_q]/T'_{do} \qquad (3.15)$$

$$Pe''_q = [e'_q-(x'_d-x''_d).i_d - e''_q - \{T_1(e''_q-x''_d.i_d)+T_2.x_d.i_d-T_{kd}.V_f\}/T'_{do}]/T''_{do} \qquad (3.16)$$

$$V_d = P(e''_q-x''_d.i_d)/\omega_o+ \omega(e''_d+x''_q.i_q)/\omega_o - r_a.i_d \qquad (3.17)$$

$$V_q = P(-e''_d-x''_q.i_q)/\omega_o + \omega(e''_q-x''_d.i_d)/\omega_o - r_a.i_q \qquad (3.18)$$

$$T_1 = (x_{1d}+x_{ad})/(\omega.r_{1d}) \qquad (3.19)$$

$$T_2 = (x_{1d}+x_{ad}.x_a/(x_a+x_{ad}))/(\omega.r_{1d}) \qquad (3.20)$$

$$T_{kd} = x_{1d}/(\omega.r_{1d}) \qquad (3.21)$$

These equations (3.14) to (3.20), take into account all the effects described by equations (3.1) to (3.10). In the above equations, the terms $1/\omega_o$ P i_d and $1/\omega_o$ P i_d are usually negligible compared with the other terms. By neglecting these two terms and other less significant terms, equations (3.14) to (3.18) may be written as:

$$Pe''_d = [(x_q-x''_q).i_q-e''_d]/T''_{qo} \qquad (3.22)$$

$$Pe'_q = [V_f-(x_d-x'_d).i_d-e'_q]/T'_{do} \qquad (3.23)$$

$$Pe''_q = [e'_q - (x'_d-x''_d).i_d-e''_q]/T''_{do} \qquad (3.24)$$

$$e''_d = V_d + r_a.i_d - x''_q.i_q \qquad (3.25)$$

$$e''_q = V_q + r_a.i_q + x''_d.i_d \qquad (3.26)$$

When combined with the rotor dynamic equations, these equations give a fifth order machine representation. The air gap torque in this case becomes:

$$T_{elec} = e''_d.i_d + e''_q.i_q - (x''_d - x''_q).i_d.i_q \qquad (3.27)$$

Further simplification may be achieved by neglecting the effects of both the direct axis and quadrature axis damper windings. This results in the following third order machine representation:

$$Pe'_q = [V_f - (x_d - x'_d).i_d - e'_q]/T'_{do} \quad (3.28)$$

$$0 = V_d + r_a.i_d - x_q.i_q \quad (3.29)$$

$$e'_q = V_q + r_a.i_q + x'_d.i_q \quad (3.30)$$

$$T_{elec} = e'_q.i_q - (x'_d - x_q)i_d.i_q \quad (3.31)$$

The fifth order synchronous machine representation of the synchronous generator is used throughout the work presented in this thesis.

3.3 The Excitation System

The primary objective of any excitation control system is to maintain the voltage at the machine terminals within specified limits. The simplest form of excitation system is to apply a constant voltage to the field winding. The voltage regulation of such a system is extremely poor and, since no control signals are used to vary the field voltage, terminal voltage recovery following a fault is likely to be poor.

Terminal voltage regulation may be improved by the use of an automatic voltage regulator (AVR), which drives the field voltage

from the difference between the terminal voltage and a reference voltage, through a gain, K, as shown in Fig. 3.2. In order for such a system to maintain the terminal voltage to within 0.5% of the reference voltage, V_{ref}, the actual value of the gain, K, must be in the order of 200 (assuming a unity gain through the generator and terminal voltage rectifier). Unfortunately, such a simple system is impractical since no consideration has been given to the dynamic terms of the generator itself, or to the mechanism by which the field voltage, V_f, is produced.

On large generator sets the excitation system must supply a large amount of power to the field circuit of the machine (typically about 5% of the machine's rated output power), and thus the gain, K, is not easily synthesised. In fact, the most widely used strategy for generating the field voltage is to mount a second machine on the generator shaft; this is known as an exciter. The exciter has its field circuit mounted on the machine stator and its armature coils wound on the machine rotor, so that control of the voltage generated by the exciter may be exercised without the use of slip rings. The voltage generated by the exciter is firstly rectified by a diode bridge mounted on the generator shaft and is then fed into the main generator field winding. Fig. 3.3 shows such an arrangement. In some cases the amplification stage, K_1, is provided by further exciter stages.

At this stage, some dynamic consideration may be given to the system shown in Fig. 3.3. By representing both the exciter and the generator as simple time constants and neglecting the time constants associated with the rectifiers, the system of Fig. 3.4 approximates the dynamics of the excitation system shown in Fig. 3.3, with the

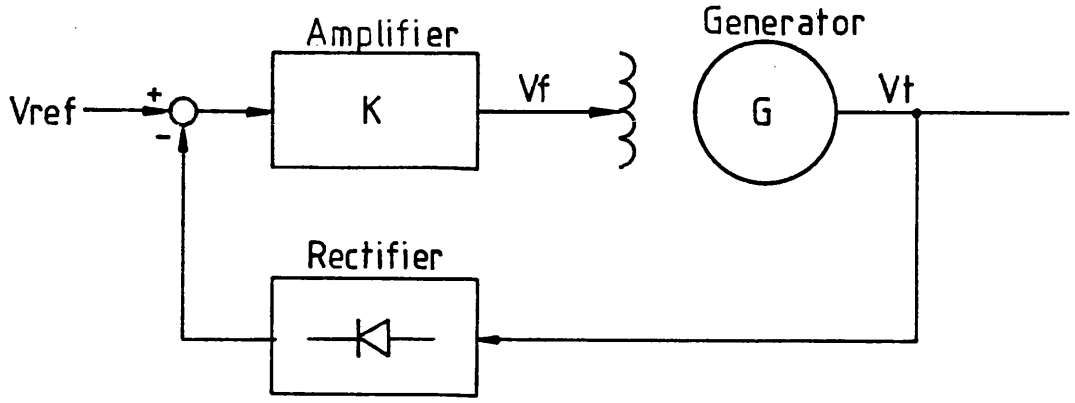


Fig 3.2 A Simple Excitation System

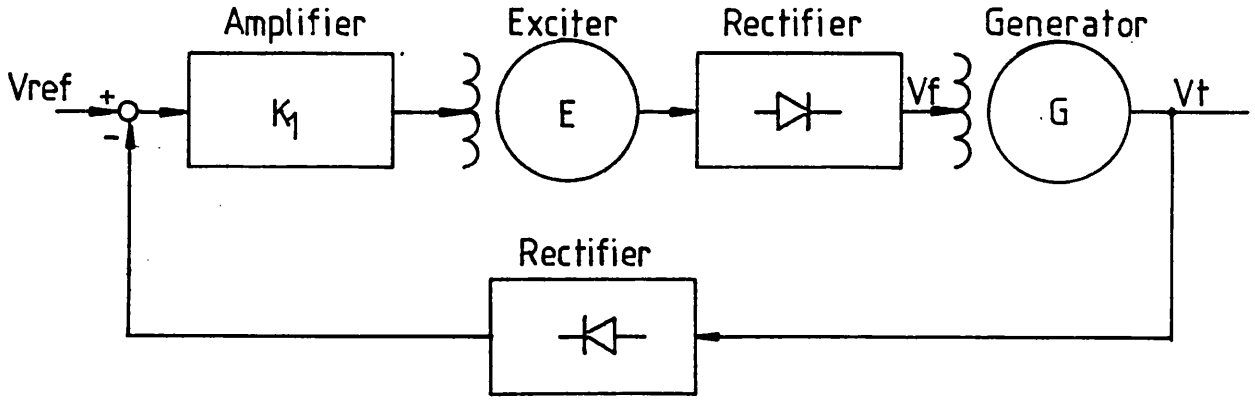


Fig 3.3 An Excitation System with an AC Exciter

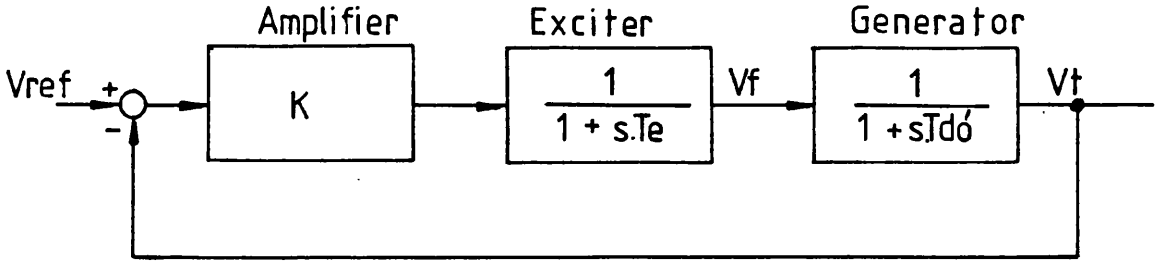


Fig 3.4 A Simple Dynamic Model of an Excitation System

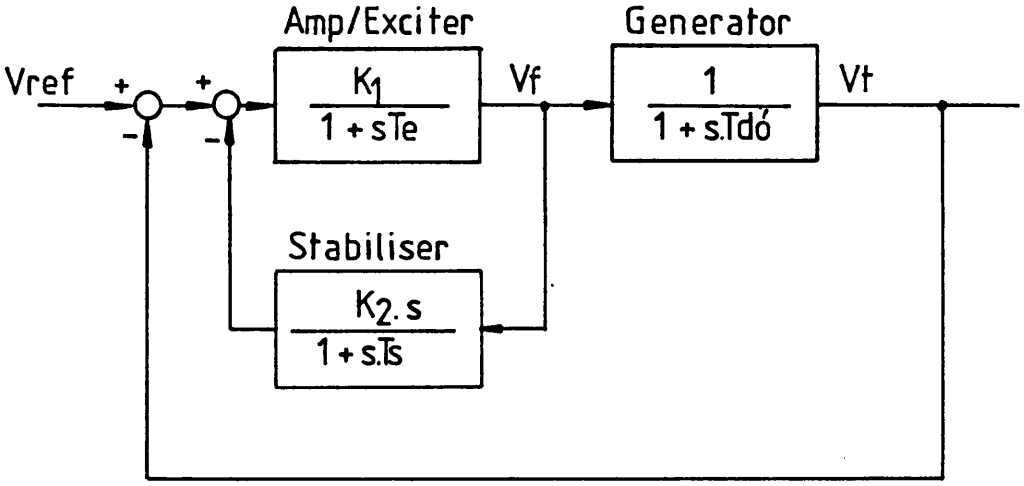


Fig 3.5 A Stabilised Excitation System

generator operating with its terminals open circuit.

It was mentioned earlier that the gain, K, of the amplifier/exciter must be in the order of 200 to achieve good steady state terminal voltage regulation. The time constants T_e and T'_{do} are typically in the range 0.5 to 2.0 seconds and 2.0 to 15 seconds respectively. Taking typical values for these time constants as T_e = 1.0 seconds and T'_{do} = 5.0 seconds, substituting these in the characteristic equation:

$$S^2 + \left[\frac{T_e + Td_o'}{Td_o'.T_e} \right] S + \frac{K_1}{Td_o'.T_e} = 0 \qquad (3.32)$$

gives a natural frequency of ω_n = 6.32 radians/sec and a damping ratio of η = 0.095 which results in a very oscillatory system. Clearly, this is an unsatisfactory situation. To improve matters, a stabilising feedback signal is introduced into the excitation system. This signal is the time derivative of the field voltage, passed through a low pass filter, in the manner shown in Fig. 3.5 to give some low frequency phase advance to the loop around the exciter. This may be considered to reduce the forward path gain under transient conditions and gives a better damped response.

The stabiliser time constant, T_s, is generally in the range 0.5 to 2.0 seconds, which permits the maximum phase advance to be moved to the region of the gain crossover and so give maximum damping.

Thus, the component parts of a conventional AVR have been introduced, namely a summing junction and amplifier/exciter and a stabiliser. There are, however, two further considerations which

have so far been neglected. First, the iron of the exciter field circuit may be driven into saturation, thereby placing a natural limit on the magnitude of the field voltage. This is to some extent a desirable feature as it is necessary to limit the power dissipated in the field circuit of the generator to avoid overheating. Secondly, the effect of the rectifier which supplies the field voltage from the exciter, to the generator field winding has not been considered. It is clear that the presence of the rectifier will prevent the field voltage from being forced in a negative direction and that any attempt to reduce the field voltage will cause the field current to flywheel through the rectifier. This flywheeling action continues until the field current has decayed to a level supported by the new, lower, field voltage. So the action of the rectifier is such that the field voltage may not be driven negative and, since the resistance of the field winding is low, the time constant associated with the decay of field current will be long due to the flywheel effect.

More recently, the trend has been towards the use of thyristor exciters. These consist of a thyristor amplifier, fed from a three-phase supply which is, in general, independent of the machine terminal voltage. This type of exciter has two principle advantages over the exciter described previously. First, the field voltage may now be reversed and secondly, the thyristor is much faster to respond to changes in the control input, so increasing the speed of the excitation system. Such a system is shown in Fig. 3.6 and is very similar in configuration to that of Fig. 3.3.

Again, due to the presence of time lags in the system and the high gain necessary for good steady state terminal voltage regulation, a

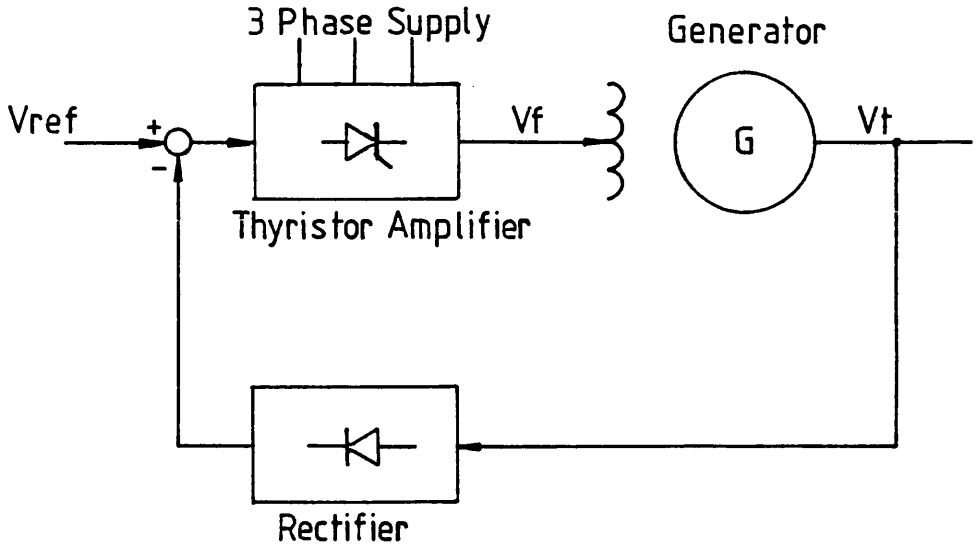


Fig 3.6 A Thyristor Excitation System

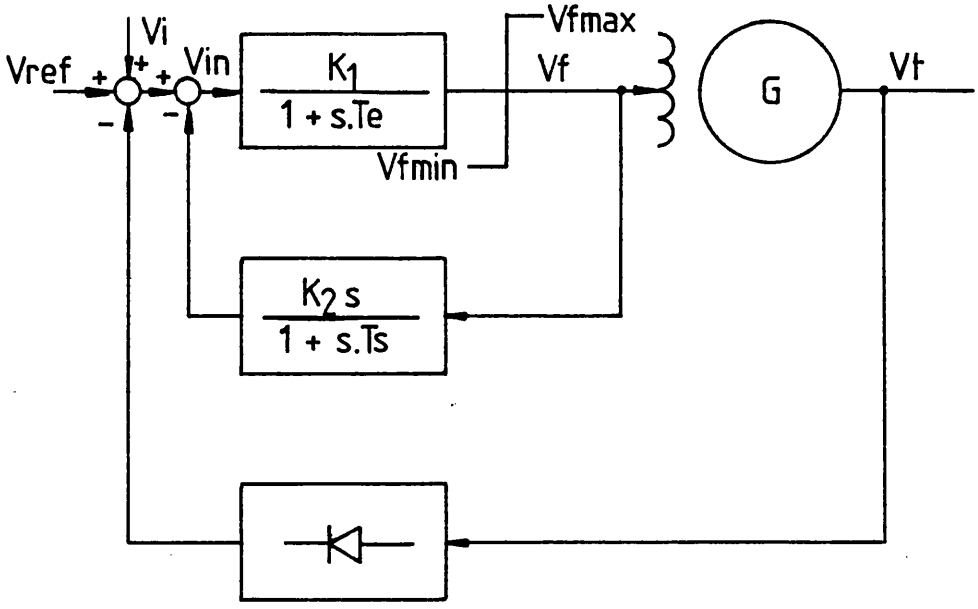


Fig 3.7 A Generalised Excitation System

stabiliser must also be fitted to this type of excitation system.

A general form of excitation system representation used throughout this thesis is shown in Fig. 3.7. An extra input, V_i, is shown, which permits the injection of extra control signals into the excitation system.

This form of representation may be used for both types of excitation system with appropriate changes in the time constants, T_e and T_s, and by adjusting the lower field voltage limit, V_{fmin}. The following assumptions are made by adoption of the representation.

1. The parameters of the exciter are constant throughout any transient behaviour.

2. The characteristics of the thyristor exciter and any rectifiers present in both types of exciter are linear.

3. Saturation of the exciter output may be represented by a simple limit on the output voltage.

4. There is no reactive power limiter fitted to the excitation system.

The dynamic equations for the excitation system which may be written from Fig. 3.7 are:

$$PV_f = [K_i.V_{in} - V_f]/T_e \qquad (3.33)$$

$$PV_s = [K_2.PV_f - V_s]/T_s \qquad (3.34)$$

$$V_{in} = V_{ref} + V_i - V_s - V_f \qquad (3.35)$$

Equation 3.33 may be substituted in equation (3.34) to give:

$$PV_s = \left[\frac{K_1.K_2.V_{in}}{T_e} - \frac{K_2.V_f}{T_e} - V_s \right] /T_s \qquad (3.36)$$

Care must be taken in the application of the field voltage limit. If equation (3.34) is used to integrate the stabilising voltage, V_s, then the term PV_f must be set to zero when the limit is applied. Alternatively, if using equation (3.36) then the voltage V_{in} must be adjusted according to equation (3.37) in order to obtain the correct stabilising voltage.

$$V_{in} = \frac{V_f}{K_1} \qquad (3.37)$$

Equation (3.37) applies only while the field voltage is in limit.

3.4 The Steam Turbine Prime Mover and Governor

The objectives of governor action when applied to a generating plant are two fold and, to a certain extent, depend on the size of the plant in relation to the rest of the total generating capacity of the power system. These two objectives are the regulation of the power system frequency and the control of the amount of power generated by

the power system as a whole. These two objectives are linked. If the power system in total generates more power than is required to supply all the loads in the system and all the associated transmission losses, then the system frequency will rise as the excess power accelerates the various inertias in the system.

Two modes of governing are often discussed: 'droop' mode, or free governing; and 'pressure' mode governing. Pressure mode governing relies on the synchronous nature of the plant to maintain the system frequency, and the governor controls the torque output by the prime mover and hence (at constant speed) the power output by the generator. Free, or droop mode governing is applied to a number of large generating sets in order to control the overall system frequency by allowing large fluctuations in the power output by the generator. Typically a droop setting of 4% is used by the CEGB, which means that a 4% speed change (expressed as a percentage of the nominal generator speed) will fully open the turbine steam valves from a fully closed position.

The original Watt type governor was a droop mode type governor. This type of governor, based on a mechanical flyball device, remained the principal method of governing steam turbines until the mid-twentieth century. Its popularity was due to a proven reliability and its ability to support acceptable load sharing between a number of generators in parallel operation. Generators of different capacities operating in parallel will share load fluctuations in proportion to their capacity if they all have the same droop setting. Speed changes detected by the flyball mechanism control hydraulic servos which position the main steam valves. Such

governors have dead bands, their droop settings are fixed and there is no facility for introducing extra control signals into the governor control loop. These limitations, coupled with the need for faster responses from governors due to the gradual increase in the power output of new generating plant and the reduction in the shaft inertia constant, have led to the adoption of electrical means for measuring shaft speed and governing. Electronic governors do not suffer from wear and vibration which can cause the failure of a mechanical governor. Further electrical signals, representing other measurements made on the plant may be easily introduced into the control loop, and the slip speed gain, which controls the droop setting, may be adjusted easily. Actuation of the steam valves is still performed by hydraulic amplifiers and the Watt type governor is still present on the turbine as part of the overspeed trip safety mechanism. Schematically the turboalternator system is represented in Fig. 3.8.

As can be seen from Fig. 3.8, the boiler/turbine/reheater/ condenser arrangement is quite complicated. The system represented is a three stage axial flow turbine. Superheated steam is passed from the boiler through the emergency valves and the governing valves to the high pressure stage. Approximately 25%–30% of the energy supplied by the generator is derived from the high pressure (HP) stage. After passing through the high pressure stage, the steam is returned to the boiler to be reheated, so boosting the thermal energy of the steam. The reheated steam then passes through a further set of emergency valves and governing valves (known as intercept valves) to the intermediate pressure (IP) stage of the turbine. The intermediate pressure stage directly feeds the low pressure (LP)

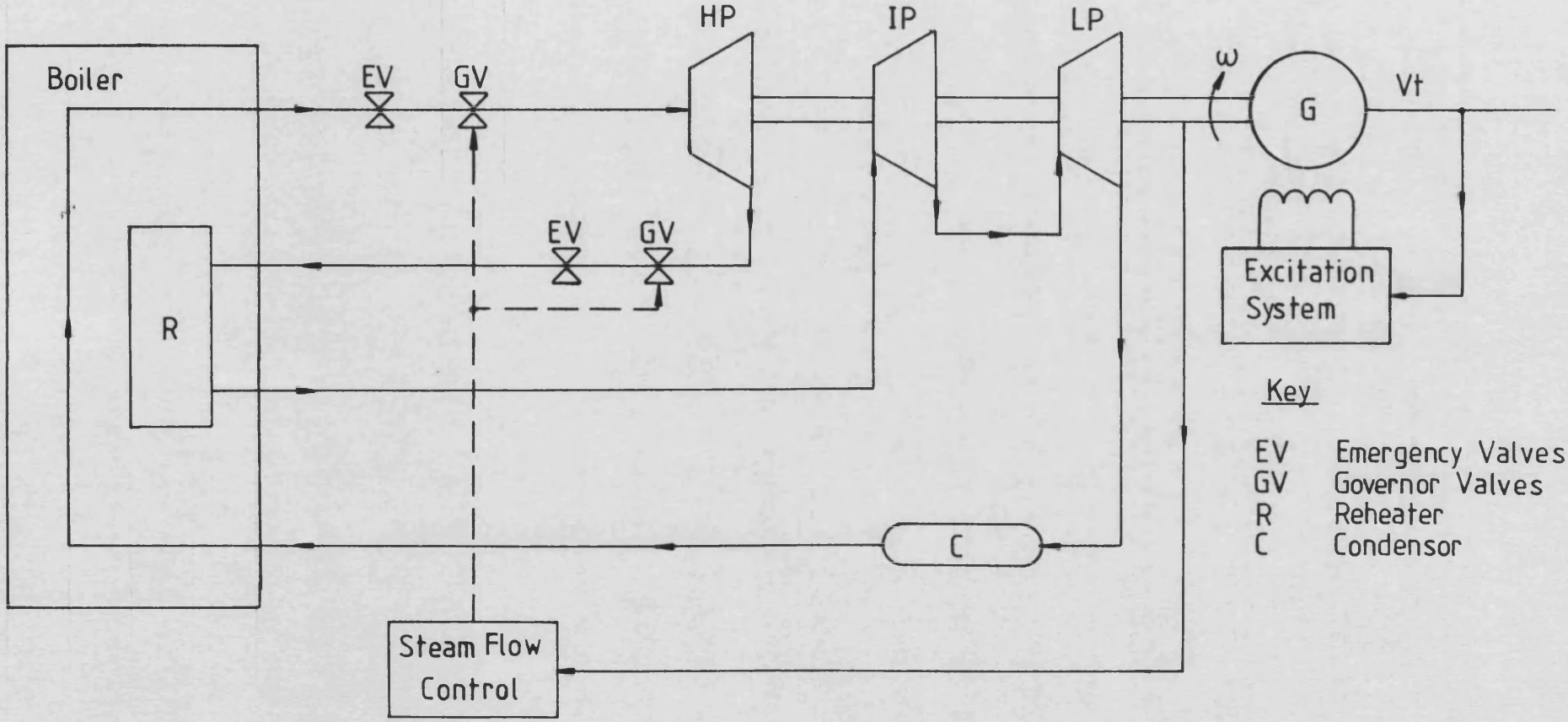


Fig 3,8 General Arrangement of a Turboalternator

stage which, in turn, exhausts under the vacuum created by the condensation of steam in the condenser prior to its return to the boiler.

Clearly, any control aimed at improving the first swing in load angle following a severe fault must have influence over a large portion of the turbine output power, so it is assumed for the purposes of this study that both the intercept valves and the high pressure governing valves are controlled. It is further assumed that these valves are controlled together and, as a result, all three stages of the turbine are amalgamated into a single stage with a single control input. These assumptions were justified by Hazell (3-4) in a reduction of a detailed non-linear representation given by Ham (3-5). This results in the turbine/governor representation of Fig. 3.9.

Fig. 3.9 also shows an auxiliary input, T_i, to the governor which may be used to feedback extra control signals. The following assumptions are made by adoption of the plant representation in Fig. 3.9.

1. Both the superheater and the reheater deliver steam at a constant temperature and pressure, i.e. they may be considered to be infinite steam sources.

2. Both the high pressure control valves and the intercept valves operate in unison and are subject to the same limits.

3. The output torque, T_{out}, is a linear function of the valve displacement, T_{in}.

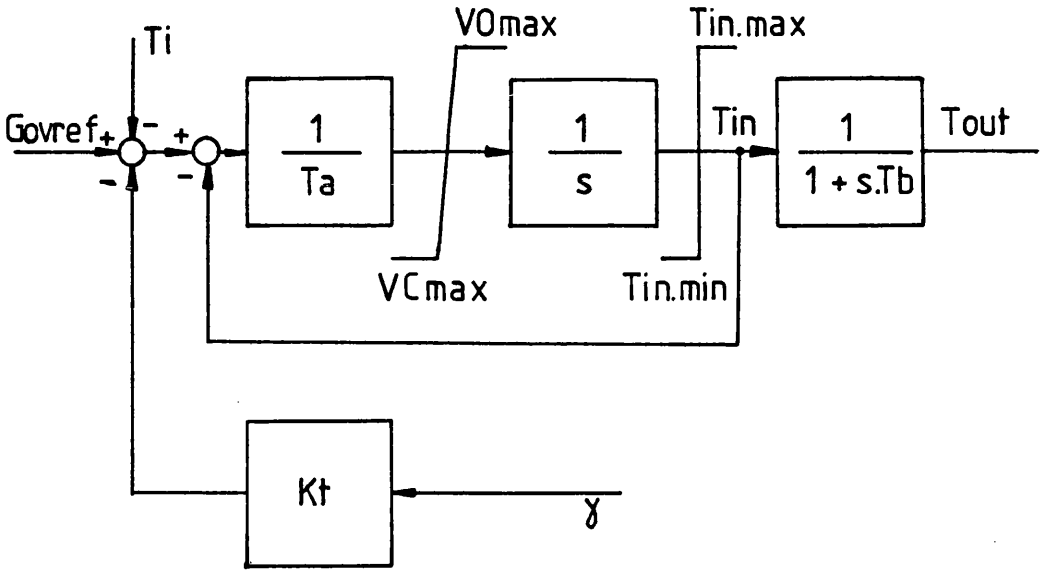


Fig3.9 Governor and Turbine Representation

4. Turbine efficiency is constant over small speed changes.

5. The dynamics of the speed transducer are negligible.

The various limits shown in Fig. 3.9 represent the limit on valve position, T_{in}, and the asymmetric rate limit on the valve velocities. Valve closing rates are generally faster than valve opening rates due to spring assistance. The closed loop around the time constant T_a and the associated integrator represent the valve positioning servo-mechanism, while the time constant T_b represents the lag between valve position and output torque.

From Fig. 3.9 the following dynamic equations may be written to represent the prime mover and governor.

$$PT_{out} = (T_{in} - T_{out})/T_b \qquad (3.38)$$

$$PT_{in} = (Gov_{in} - T_{in})/T_a \qquad (3.39)$$

$$\text{where } Gov_{in} = Gov_{ref} - T_i - K_T Y \qquad (3.40)$$

$$\text{and } Y = \omega - \omega_o = P\delta \qquad (3.41)$$

3.5 The Hydro-Turbine Prime Mover and Governor

The plant data used to simulate a water driven synchronous generator is based on the Dinorwig Pump Storage Scheme sited in the Snowdonia National Park, North Wales. Dinorwig has a subterranean machine hall which contains six machine sets, each rated at 330MW.

The Dinorwig machine sets are mounted vertically and each is capable of both pumping water during off-peak periods or generating during peak periods. Dinorwig is unique in this respect, since other pump storage schemes, such as the small Ffestiniog installation, use separate machine sets for pumping and generating operations. Installations such as Dinorwig are necessary in order that nuclear generation may be used to supply more than the base load power requirements. Nuclear plant may not be run up or down quickly and, as a result, other types of generation, such as oil fired, coal fired or gas turbine, must be used to cater for the daily peaks and troughs in the demand for power. The use of pumping action at Dinorwig during off-peak periods artificially increases the system load and the energy stored by this action is used during peak periods to supplement the output from other generating plant. Thus Dinorwig may be used to smooth the daily fluctuations in the demand for power by artificially increasing the base load requirements during off-peak periods and supplementing generating capacity during peak periods. Dinorwig may also be turned around very quickly if generating capacity is lost from the National Grid. Unlike most other installations, Dinorwig requires very little in the way of power in order to run the station. Coal fired power stations require large amounts of power, typically 10% of the station output, to supply the various pieces of plant and machinery, such as coal crushers, forced draught fans, and conveyors, all of which supply the coal/air mixture to the boiler furnaces. Dinorwig has been designed to be capable of being run up from 'cold', without any support from the grid. In the event that the grid as a whole were to be run down then the presence of Dinorwig allows a much more rapid recovery of the power system. Once Dinorwig itself has been run up, it could supply the auxiliary

power necessary to bring up another large piece of generating plant and so on until sufficient of the network is operational for load centres to be reconnected.

The transmission lines which connect Dinorwig to the rest of the National Grid run close to the north coast and are subjected to severe weather conditions. Conductor clashes are not uncommon and so disturbances to the plant are not infrequent. When generating, water is taken from an existing lake at Marchlyn Mawr and discharged some 500 metres below at Llyn Peris (3-15). The main high pressure water tunnels are 9.5 metres in diameter and deliver the water to a manifold from which high pressure, steel penstock pipes direct the water through the main inlet valves to each machine set. The opening and closing rates of the main inlet valves are very slow and it takes of the order of 15 seconds to fully open or close these valves. Such slow valve rates dominate the dynamics of the governor loop and very little governor action can occur in the short transient period associated with faults such as overhead conductor clashes. The prime mover/governor arrangement has therefore been represented by the same structure as the steam turbine, Fig. 3.9. Again, the loop around the time constant T_a represents the position servomechanism of the control valves, while the time constant, T_b, represents the lag between valve position, T_{in}, and prime mover output torque, T_{out}. The valve opening and closing rates have been set at ±0.0833 p.u. per second which requires 12 seconds to fully open the control valves from the fully closed position or vice versa.

3.6 The Diesel Engine Prime Mover and Governor

Diesel engine prime movers are often used in small finite busbar power systems, such as those found on board ships and on small islands, or used as backup facilities for equipment normally supplied from the National Grid or some other electricity supply utility. A full dynamic simulation of a diesel engine requires representation of the fluid and thermodynamics of the engine, where these are treated as a series of control volumes, interconnected by orifices of varying geometry (valves, etc.), which are coupled by heat, mass and work transfers (3-6). Such a representation leads to a high order dynamic model which requires a large amount of plant data. An alternative approach is to use a quasi-steady (3-6) representation in which the model employs the steady state characteristics of the engine. This approach requires steady state experimental data for each of the active components of the prime mover. Models of this nature are essentially algebraic and ignore the dynamic effects involved in mass and heat transfers within the engine. Again, a large amount of data is required to describe the diesel engine.

A simple linear representation, used by Wyatt (3-7) relies on the fact that the diesel engine may be expected to operate close to the electrical supply frequency at a nominally constant speed. In this mode of operation, the output torque of the diesel engine may be considered to be proportional to the fuel rack position and delayed by the transport lag of the fuel system. At constant speed the output torque limit may be implemented simply as a constant saturation limit, rather than as a complex speed dependent function. Fig. 3.10 shows a schematic diagram of the diesel engine and governor

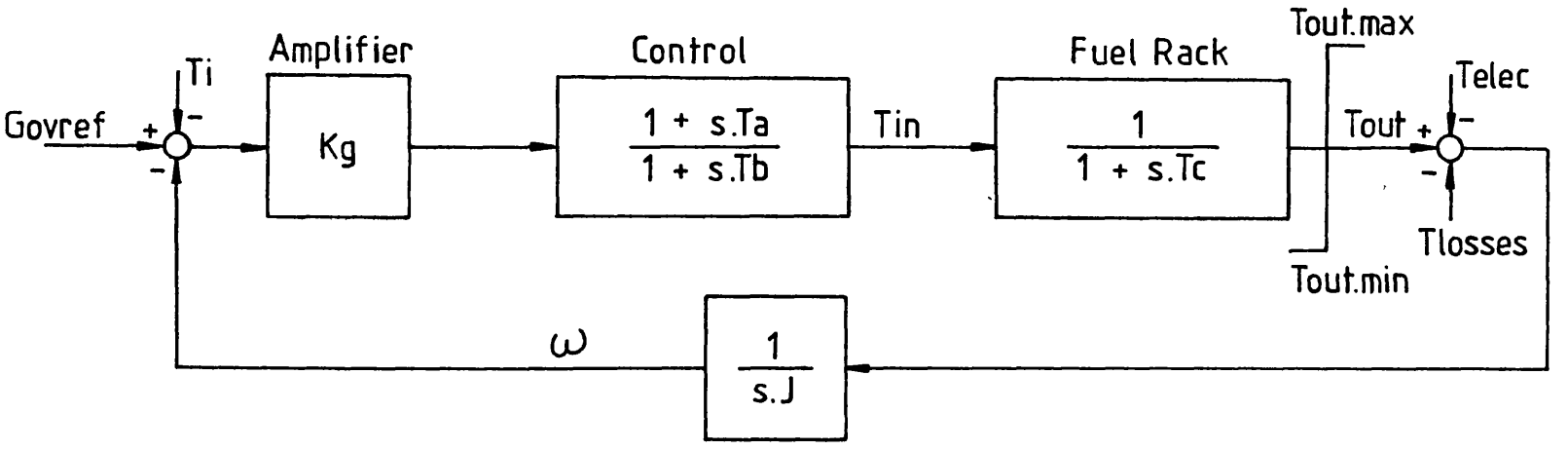


Fig 3.10 Diesel Engine and Governor Representation

representation used in the finite busbar optimisation studies presented in Chapter 6.

The time constant, T_c, represents the dynamics of the fuel rack and the lead/lag network, T_a and T_b, is introduced by the control amplifiers used to actuate the fuel rack. An auxiliary input, T_i, is available to feed extra control signals into the governor system. The transport lag of the fuel system is not represented. From Fig. 3.10 the following equations may be written to represent this prime mover/governor arrangement.

$$pT_{out} = [T_{in} - T_{out}]/T_c \qquad (3.42)$$

$$pT_{in} = [K_g(1+pT_a).Gov_{in} - T_{in}]/T_b \qquad (3.43)$$

$$p\omega = [T_{out} - T_{elec} - T_{losses}]/J \qquad (3.44)$$

where

$$Gov_{in} = Gov_{ref} - \omega - T_i \qquad (3.45)$$

3.7 The Transmission System and the Finite Load

The equations governing the transmission line dynamics may be obtained by applying the Parks transform to the three phase transmission line equations. The following assumptions are made in formulating these equations.

1. The magnetising currents of the generator transformer and the line charging currents may be neglected.

2. The transformer and transmission lines may be represented together by a lumped series inductance and resistance.

Application of the Parks transform yields the following equations in the direct and quadrature axes of the generator.

$$V_d = V_b \sin\delta + \frac{X_T . P i_d}{\omega_o} + R_T . i_d - \frac{\omega}{\omega_o} X_T \, i_q \qquad (3.46)$$

$$V_q = V_b \cos\delta + \frac{X_T P i_q}{\omega_o} + R_T . i_q + \frac{\omega}{\omega_o} . X_T . i_d \qquad (3.47)$$

By neglecting the effects associated with frequency variation and the rate of change of axis currents, then these equations may be simplified and written as:

$$V_d = V_b \sin\delta + R_T . i_d - X_T . i_q \qquad (3.48)$$

$$V_q = V_b \cos\delta + R_T . i_q + X_T i_d \qquad (3.49)$$

where X_T and R_T represent the lumped transformer/transmission line reactance and resistance.

An inductive load, connected to a single generator in a small finite busbar system may be represented by considering the busbar voltage, V_b, to be zero. This gives the following equations which represent an inductive load connected to a simple finite busbar power system.

$$V_d = R_L.i_d - X_L.i_q \qquad (3.50)$$

$$V_q = R_L i_q + X_L.i_d \qquad (3.51)$$

where R_L and X_L represent the load resistance and reactance respectively. Equations (3.48) to (3.51) are steady state equations and their use simulates the transmission line by stepping from one steady state condition to the next.

3.8 Numerical Solution

The plant equations given in the preceding sections are given in a continuous form and, as such, are suitable for solution by an analogue computer. The solution obtained by numerical techniques on a digital computer will give the system response at discrete instants in time. In general, the mathematical description of a plant consists of a set of simultaneous differential equations and a set of simultaneous algebraic equations. These two groups of equations may be represented as:

$$\dot{\underline{x}} = \underline{f}(\underline{x},\underline{U}) \qquad (3.52)$$

and

$$\underline{0} = \underline{g}(\underline{x},\underline{y}) \qquad (3.53)$$

where $\underline{x}$ is a vector of dynamic state variables which may be generated by the integration of the vector function $\underline{f}$ with respect to time. The vector $\underline{y}$ is a vector of algebraic variables which are obtained by

solving the algebraic equations represented by the vector function $\underline{g}$. The vector $\underline{U}$ contains a subset of the algebraic variables that make up the vector $\underline{y}$, which are known as the interface variables. These interface variables provide the coupling between the dynamic and algebraic equations. For example, using a plant description made up of the fifth order machine representation, with automatic excitation control, prime mover and governor, and a load, the vectors $\underline{x}$, $\underline{y}$ and $\underline{U}$ are:

$$\left.\begin{aligned} \underline{x}^T &= [e''_d,\ e'_q,\ e''_q,\ \delta,\ P\delta,\ T_{out},\ T_{in},\ V_f,\ V_s] \\ \underline{y}^T &= [i_d,\ i_q,\ T_{elec},\ Gov_{ref},\ V_{ref},\ V_T,\ V_d,\ Vq] \\ \underline{U}^T &= [i_d,\ i_q,\ T_{elec},\ Gov_{ref},\ V_{ref},\ V_T] \end{aligned}\right\} \qquad (3.54)$$

The objective of solving these equations is to compute the dynamic response of the system with sufficient accuracy for engineering purposes. The solution method must also be reliable and stable.

3.8.1 Integration Techniques

There are two main categories of integration technique, explicit or implicit integration and each of these categories can be subdivided into single-step and multi-step algorithms. Explicit integration techniques involve explicit evaluation of the system function $\underline{f}$, whilst implicit techniques involve some form of rearrangement of the dynamic equations into a set of simultaneous algebraic equations, so that the derivative $\dot{\underline{x}}$ is not explicitly calculated. This latter type of technique is more complicated, but leads to greater numerical stability (3-8). Single step techniques

require no information about the solution prior to the beginning of each integration step. As such, they are self-starting, which is convenient when discontinuities are present. Multi-step algorithms require the storage of previous values of the state variables and/or their time derivatives. In principle, such algorithms are more efficient. However, most multi-step algorithms must be restarted when a discontinuity is encountered. Most multi-step algorithms are based on open and/or closed difference equations.

It is also necessary to recognise the possible sources of numerical error which contribute to the total error encountered during a given integration step. Since most integration algorithms are obtained from truncated series, errors may be introduced due to the missing terms. Further errors may be introduced by the numerical representation since, in general, it is not possible to represent the full significance of a numerical value and arithmetic operations will cause a rounding error. If equations (3.52) and (3.53) cannot be solved simultaneously at each integration step, an interfacing error may be introduced. Errors may also be introduced by approximations regarding the behaviour of variables (through interpolation or extrapolation) or by assuming the linearity of equations over the integration step. The application of limits can cause problems due to the finite nature of the time increments. Care must also be taken when a limit is encountered. Any time derivatives of limited variables which are explicitly calculated must be forced to behave correctly. Thus the error in the solution at the end of an integration step is some function of the error incurred during the integration step and the accumulated error passed on from the previous integration step. An integration technique becomes unstable

when these errors accumulate and hide the true solution.

3.8.2 Explicit Integration Techniques

Perhaps the most simple of all integration methods is the explicit Euler method. This algorithm calculates the system state at the end of an integration step by simply adding to the state at the beginning of the integration step the time derivative of the state at the beginning of the integration step multiplied by the integration step length. Thus, by Euler:

$$\underline{x}_n = \underline{x}_{n-1} + h\underline{f}(\underline{x}_{n-1}, \underline{U}_{n-1}) \qquad (3.55)$$

where h is the integration step length and the subscripts n-1 and n indicate the begining and end of the integration step respectively. This method is equivalent to a Taylor series truncated after the first order term. This is a single step explicit integration method since equation (3.52) is explicitly evaluated and only the current values of $\underline{x}$ and $\dot{\underline{x}}$ are used to calculate the new value of $\underline{x}$. Euler is not a very stable algorithm so the integration steps must be short in order to ensure that the error introduced into the solution is small.

Open multi-step methods are similar to Euler. In essence, they are the predictor part of a predictor-corrector algorithm and differ from Euler solely in the use of stored values of the time derivative of the system state, $\underline{x}$.

Predictor-Corrector methods consist of two equations both of

which are multi-step. The open multi-step predictor equation predicts a value for the state at the end of the integration step, $\underline{x}_n$, based on the state at the beginning of the integration step, $\underline{x}_{n-1}$, and the time derivatives stored backward in time from $\dot{\underline{x}}_{n-1}$. The predicted value for $\underline{x}_n$ then becomes the initial estimate in the closed multi-step corrector algorithm which is iterated to convergence. For example, the following Adams pair (3-15) may be used to implement a predictor-corrector algorithm. First, the derivative $\dot{\underline{x}}_{n-1} = \underline{f}(\underline{x}_{n-1}, \underline{U}_{n-1})$ is calculated and used in the predictor equation

$$\underline{x}_n = \underline{x}_{n-1} + h(23\dot{\underline{x}}_{n-1} - 16\dot{\underline{x}}_{n-2} + 5\dot{\underline{x}}_{n-3})/12 \qquad (3.56)$$

to obtain an estimate for $\underline{x}_n$. The corresponding value of the interface vector, $\underline{U}_n$, is then calculated by solution of the algebraic equations (3.53) and the time derivative $\dot{\underline{x}}_n = \underline{f}(\underline{x}_n, \underline{U}_n)$ is calculated. The closed corrector equation:

$$\underline{x}_n = \underline{x}_{n-1} + h(5\underline{f}(\underline{x}_n, \underline{U}_n) + 8\dot{\underline{x}}_{n-1} - \dot{\underline{x}}_{n-2})/12 \qquad (3.57)$$

is applied to modify the value of the state at the end of the integration step. A new solution to the algebraic equations is then calculated and the correction process repreated to convergence. Economies may be made in the use of this algorithm by only solving the algebraic equations once, at the point between prediction and the first correction and using this value for $\underline{U}_n$ throughout the correction steps. This introduces an interface error, which may or may not be significant, depending on the integration step length. If

the algebraic equations are solved each time around the correction loop, then the interface error is eliminated, even if they were not solved between the prediction step and the first correction step. In this latter case, an extrapolated value for $\underline{U}_n$ is used in the first correction step. When iterated to convergence, the solution provided is independent of the performance of the predictor which merely provides initial values for the corrector. Corrector convergence is slow for stiff problems, that is, for problems in which the ratio of the largest time constant to the smallest time constant is high and these algorithms must be restarted following a discontinuity. The iterative nature of the corrector steps make it difficult to guarantee the execution time of the algorithm which is a severe disadvantage when the algorithm is required to operate in real time.

The Runge Kutta method is another single step explicit integration technique. This method has been widely used for the solution of the dynamic responses of power systems (3-8 to 3-10) and the fourth order verison is commonly used. This algorithm is summarised by the following equations:

$$\underline{K}_1 = h\underline{f}(\underline{x}_{n-1}, \underline{U}_{n-1}) \qquad (3.58)$$

$$\underline{K}_2 = h\underline{f}(\underline{x}_{n-1} + 0.5\underline{K}_1, \underline{U}_a) \qquad (3.59)$$

$$\underline{K}_3 = h\underline{f}(\underline{x}_{n-1} + 0.5\underline{K}_2, \underline{U}_b) \qquad (3.60)$$

$$\underline{K}_4 = h\underline{f}(\underline{x}_{n-1} + \underline{K}_3, \underline{U}_c) \qquad (3.61)$$

$$\underline{x}_n = \underline{x}_{n-1} + 1/6[\underline{K}_1 + 2\underline{K}_2 + 2\underline{K}_3 + \underline{K}_4] \qquad (3.62)$$

Some variations to the algorithm are possible due to the handling of the interface vector, $\underline{U}$, during the integration step. In order for the differential and algebraic equations to be rigorously interfaced to each other, $\underline{U}_a$, $\underline{U}_b$ and $\underline{U}_c$ must be obtained by solution of the algebraic equations between each evaluation of the function $\underline{f}$. Thus, $\underline{U}_a$ must be obtained by the solution of:

$$\underline{0} = \underline{g}(\underline{x}_{n-1} + 0.5\underline{K}_1, \underline{y}_a) \qquad (3.63)$$

prior to the evaluation of $\underline{K}_2$, remembering that $\underline{U}_a$ is a subset of $\underline{y}_a$. Likewise, $\underline{U}_b$ and $\underline{U}_c$ must be calculated prior to the evaluation of $\underline{K}_3$ and $\underline{K}_4$ and, finally, $\underline{U}_n$ must be calculated prior to the beginning of the next integration step. Alternatively, it is possible to assume that $\underline{U}$ is constant over the integration step so that $\underline{U}_a = \underline{U}_b = \underline{U}_c = \underline{U}_{n-1}$, or it is possible to extrapolate values for these intermediate values of $\underline{U}$ based on the previous integration steps. Neither of these last two methods provides a rigorous interface between the two sets of equations and it is possible to improve on the solutions obtained in these instances by reintegration of the time step using values for $\underline{U}_a$, $\underline{U}_b$ and $\underline{U}_c$ which are interpolated from the $\underline{U}_{n-1}$ and $\underline{U}_n$ values. This reintegration process may be repeated until values for $\underline{x}_n$ and $\underline{U}_n$ converge. Iterative solution in this manner is only of advantage if the time taken to converge is less than the time taken by a single vigorously interfaced integration step. It is notable that the fourth order Runge Kutta method requires four evaluations of the dynamic equations and up to four evaluations of the algebraic equations for each integration step. Thus, although Runge Kutta may be used with longer integration step lengths than Euler, a 4:1 increase in the step length must be obtained before the Runge Kutta

algorithm becomes computationally more efficient. Numerical stability problems arise when using Runge Kutta to integrate stiff equations. With stiff equations, a relatively unstable integration method requires a shorter integration step length in order to accurately track changing components in the state vector and to maintain introduced errors at a low level. The range of time constants encountered in a power system can be large and the only solution to numerical instability when using Runge Kutta is to reduce the integration step length.

3.8.3 Implicit Integration Techniques

An implicit multi-step integration method can be based on the corrector equation of a predictor-corrector pair. This equation, for example, equation (3.57), is a simultaneous equation in $\underline{x}_n$ with $\underline{U}_n$ being the only other unknown. All closed loop multi-step corrector equations may be expressed in the form:

$$\underline{x}_n = Kh\underline{f}(\underline{x}_n, \underline{U}_n) + \underline{C} \qquad (3.64)$$

where $\underline{C}$ is a constant vector during each integration step, made up of some weighted sum of previous $\underline{x}$ and $\dot{\underline{x}}$ vectors, and K is some constant coefficient. If the function $\underline{f}$ can be expressed in the linear form:

$$\dot{\underline{x}} = A\underline{x} + B\underline{U} \qquad (3.65)$$

where the matrices A and B are constant over the integration step then equation (3.64) may be written as:

$$[I - KhA]\underline{x}_n = KhB\underline{U}_n + \underline{C} \qquad (3.66)$$

where I is an identity matrix. Provided that the matric [I - KhA] is non-singular, $\underline{x}_n$ can then be expressed in terms of $\underline{U}_n$. Iterative solution of equations (3.52) and (3.53) is then possible. From an initial estimate of $\underline{U}_n$, either by extrapolation from previous values or by application of a predictor equation to estimate $\underline{x}_n$ and the subsequent solution of the algebraic equations, a new value of $\underline{x}_n$ may be obtained by application of equation (3.66). This value of $\underline{x}_n$ is then used to resolve the algebraic equations and a new value for $\underline{U}_n$ calculated so that the process may be repeated to convergence. Although this process may be slightly more complicated than the predictor corrector method, interface error is eliminated and larger integration steps are possible with stiff equations, so the overall computation effort may be reduced (3-8).

Another implicit integration method arises from dynamic equations represented in the form of equation (3.65). Equation (3.65) may be integrated analytically over a single time step in which case the solution is expressed as a matrix exponential. This results in an expression for the new state $\underline{x}_n$ in terms of the state $\underline{x}_{n-1}$ and the interface variables $\underline{U}_n$ and $\underline{U}_{n-1}$, of which $\underline{U}_n$ is again the only other unknown. Thus:

$$\underline{x}_n = \Phi\underline{x}_{n-1} + W_1\underline{U}_{n-1} + W_2\underline{U}_n \qquad (3.67)$$

where

$$\Phi = I + A.G.h \qquad (3.68)$$
$$G = I + A.F \qquad (3.69)$$

$$F = \sum_{K=0}^{\infty} A^K h^{K+1}/(K+2)! \qquad (3.70)$$

$$W_2 = F.B \qquad (3.71)$$

and

$$W_1 = Gh - W_2 \qquad (3.72)$$

The form of equation (3.67) is similar to that of equation (3.66) since it is possible to express $\underline{x}_n$ as a function of $\underline{U}_n$, the remaining terms being constant across the integration step. This equation may also be solved by iterative means to avoid interfacing error. The infinite series of equation (3.70) may be truncated after a few terms to give, say, an mth order integration formula, where m-1 is the upper limit of the summation. Integration by this method is extremely stable (3-8) and, provided that the matrices A and B do not vary with time or operating point, the matrices ϕ, W_1 and W_2 need only be calculated once.

Implicit integration by use of the Z-transform (3-14) requires the transformation of the plant differential equations into the discrete time domain (the Z domain). The Z-transform is used to describe sampled data systems. In general, a system is considered to act on a train of scaled impulses (samples), which represent the input, to produce a train of scaled impuses at the output. The Z-transform of a continuous function of time, f(t), is defined as:

$$Z[f(t)] = \sum_{K=0}^{\infty} f(KT)Z^{-K} \qquad (3.73)$$

where Z means "take the Z transform of", is the Z-transform operator and Z^{-1} represents a delay of one sample period, T. The transfer function of a system may be represented by the ratio of the input and output Z-transforms. For example, the impulse response, y(t), of a first order lag described in terms of the Laplace domain transfer function:

$$\frac{Y(s)}{X(s)} = \frac{1}{s+a} \qquad (3.74)$$

is well known to be an exponential with a time constant of 1/a. Thus the Z-transform of the output, $y(t) = e^{-at}$, is:

$$Y(Z) = \sum_{K=0}^{\infty} e^{-a(TK)}.Z^{-K} = \sum_{K=0}^{\infty} [e^{-aT}Z^{-1}]^K = \frac{1}{1-e^{-aT}Z^{-1}} \qquad (3.75)$$

The Z-transform of the impulse $x(t) = \delta(t)$ is unity, i.e. $X(Z) = 1$, so the discrete time domain transfer function for the first order lag of equation (3.74) is:

$$\frac{Y(Z)}{X(Z)} = \frac{1}{1-e^{-aT}Z^{-1}} = \frac{Z}{Z-e^{-aT}} \qquad (3.76)$$

This transfer function may be implemented by rearranging equation (3.76) so that:

$$Y(Z) = X(Z) + e^{-aT}Z^{-1}Y(Z) \qquad (3.77)$$

which may be programmed simply as a multiply and add loop, since e^{-aT} is a constant and $Z^{-1}Y(Z)$ is the previous sample value of y(t). Note

that in steady state $Z^{-1}Y(Z) = Y(Z)$, so at steady state for a unit step input where all sample values are unity,

$$Y(Z) = \frac{1}{1-e^{-aT}} X(Z) \qquad (3.78)$$

This shows that the discrete time domain transfer function requires scaling in order to produce the correct amplitude output. In the Laplace domain the steady state value is given by:

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} s.Y(s) \qquad (3.79)$$

which, in this case, for a step input gives:

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} s.\frac{1}{s+a}\frac{1}{s} = \frac{1}{a} \qquad (3.80)$$

Likewise, in the discrete time domain the steady-state value of a Z-transform is given by:

$$\lim_{K\to\infty} y(KT) = \lim_{Z\to 1} (Z-1)\,Y(Z) \qquad (3.81)$$

which, in this case, for a step input gives:

$$\lim_{K\to\infty} y(KT) = \lim_{Z\to 1} (Z-1)\frac{Z}{Z-e^{-aT}}\frac{Z}{Z-1} = \frac{1}{1-e^{-aT}} \qquad (3.82)$$

Thus, in this case, the normalised discrete time domain transfer function for the first order lag of equation (3.74) is given by:

$$\frac{Y(Z)}{X(Z)} = \left[\frac{1-e^{-aT}}{a}\right]\frac{Z}{Z-e^{-aT}} \qquad (3.83)$$

It is notable that no explicit evaluation of the time derivative of y(t) is made. An important consideration with this type of integration method is the sample period, T. If the sampling frequency ($1/T$ H_Z) is less than twice the frequency of the highest frequency component of the transformed functions, then an aliasing error may occur. This is illustrated in Fig. 3.11.

Although the sample values of the two curves in Fig. 3.11 are coincident, the two curves are completely different. If it can be said that the function, y(t), contains no frequency components higher than $1/(2T)H_Z$ then it can be shown that there is a unique function y(t) which passes through all the sample values, i.e. the low frequency curve in Fig. 3.11.

Timings taken by Tanner (3-11) show that the Z-transform integration method is approximately four times as fast as the fourth order Runge Kutta method and that the method is numerically stable enough to use an integration step length that is twice as long as that used with Runge Kutta.

The Z-transform method has been chosen as the method by which simulation of the plant dynamic equations is implemented. Transformation of the various plant equations into the discrete time domain will be conducted in a later section.

3.9 Numerical Representation

The choice of numerical representation is an important decision

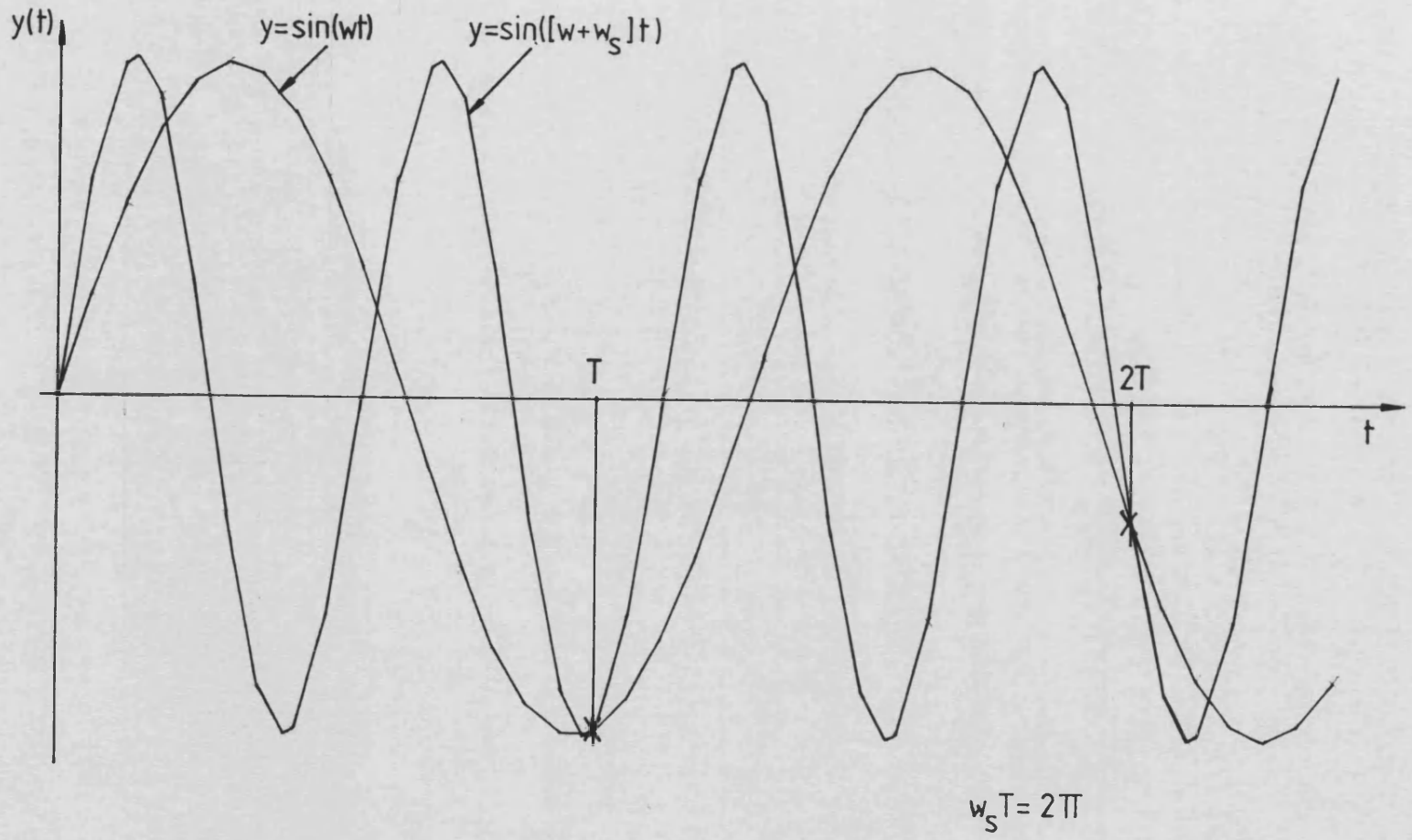


Fig 3.11 Aliasing Error

when considering numerical simultation. There are two distinct methods of representing numerical data when finite sized number representations are used. These two methods of representation are fixed point (scaled integer) and floating point.

Fixed point representations of numerical data are essentially integer representations with a fixed position for the binary point (either inside or outside the width of the number representation). Most, if not all, currently available microprocessor integrated circuits are equipped with instructions for performing arithmetic operations on integer data. More recent microprocessors such as the Motorola MC68000 (3-12) are capable of handling two's complement binary arithmetic using word lengths up to 32 bits. Additions and subtraction of fixed point values with aligned binary points may therefore be performed by a single machine instruction. Multiplication and division of fixed point numbers may also be carried out using the standard integer mechanisms. However, the binary point of the product or quotient must be re-aligned so that the result remains consistent with the particular fixed point representation. Again, most currently available microprocessors facilitate such realignment with a comprehensive set of shift instructions. If a 32 bit two's complement binary representation is used to represent fixed point numbers, then a dynamic range of ±2147483658 to 1, i.e. 2^{31} to 1, is possible. Despite this large dynamic range, should the dynamic range over which a particular value varies be large, then problems arise due to the precision to which this number may be represented. Truncation due to arithmetic operations can force small but finite values to zero purely because there are insufficient bits to the right of the binary point to

represent the finite value. Thus a fixed point representation of a value is equivalent to quoting that value to a fixed number of decimal places.

Conversely, a floating point representation is equivalent to quoting a value to a fixed number of significant figures. A floating point value is kept in two parts, a mantissa and an exponent. The mantissa of a floating point value represents the significant figures of the number, while the exponent indicates the position of the binary point. In order that the mantissa gives the most precise representation of the numerical value, the mantissa is generally normalised so that the most significant bit of the mantissa is always set and the exponent adjusted accordingly. One example of a 32 bit floating point representation is shown in Fig. 3.12.

The 24 bit mantissa, from bits 23 to 0, is unsigned and the binary point is deemed to be immediately to the left of bit 23. This means that, by using a normalised mantissa, the value of the mantissa, m, is such that $1 > m \geq 0.5$. The exponent, E, is a 7 bit two's complement value in the field between bits 30 and 24, so the exponent is such that $63 \geq E \geq -64$. Bit 31 gives the sign of the mantissa, 0 for a positive mantissa, so that the sign of a floating point value may be determined in the same manner as the sign of a two's complement representation. The decimal value, D, represented by this representation is $D = (-1)^S.m.2^E$, such that $9.2234 \times 10^{18} > |D| \geq 2.7105 \times 10^{-20}$. This gives this particular floating point representation a dynamic range of $\pm 3.4028 \times 10^{38}$ to 1, i.e. 2^{126} to 1. Using this representation, zero is represented by a 32 bit word with all its bits cleared since it is not possible to

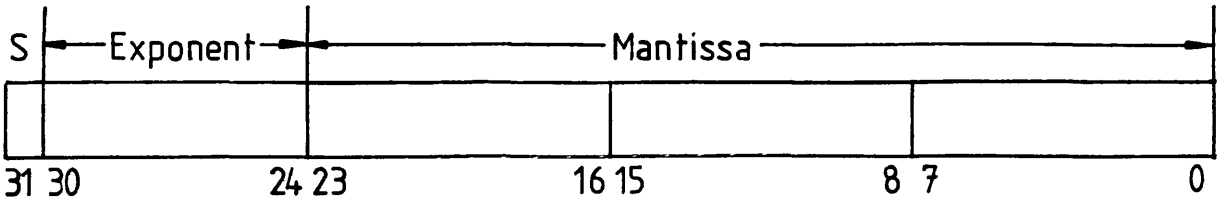


Fig 3.12 Floating Point Representation

normalise a zero valued mantissa. Alternative floating point representations, such as that recommended by the IEEE (3-13) recognise that, in general, the most significant bit of the mantissa is set by the normalisation process and so need not be represented. The extra bit left over by the removal of the so-called 'hidden bit' is then available for use either to increase the precision of the mantissa or, more usefully, to increase the dynamic range of the representation by extending the exponent field. Exponents may also be represented by unsigned values since no sign extension is then required on an exponent extracted from a floating point value. For example, the IEEE recommended representation of a single precision 32 bit floating point number (3-13) uses an 8 bit exponent, from bits 30 to 23 inclusive, represented excess 127, i.e. the true exponent is obtained by subtracting 127 from the unsigned 8 bit exponent field. In order to accommodate this 8 bit exponent field and maintain 24 bit precision for the mantissa field, the most significant mantissa bit is taken to be set, unless the whole floating point word is zero and the binary point is taken to be immediately to the right of the hidden bit. Thus the mantissa is normalised such that $2 > m \geq 1.0$ and the exponent field represents values for the exponent E such that $128 \geq E \geq -127$. Use of the maximum exponent value, E=128, is reserved for representing infinity and other exceptional conditions, so the range of decimal values which are represented is $2x2^{-127} > |D| \geq 1x2^{-127}$, i.e. $3.4028x10^{38} > |D| \geq 5.8775x10^{-39}$, giving a dynamic range of $\pm 5.7896x10^{76}$ to 1, i.e. 2^{255} to 1.

A variety of number representations are used throughout the real time simulation software. By far the majority of arithmetic performed in the simulation uses fixed point arithmetic with the

binary point between bits 15 and 16 of a 32 bit value. In order to extend the precision to which certain small numerical values are represented at critical points in the simulation, many of the constant multipliers used in the simulation are represented by fixed point values with the binary point placed between bits 23 and 24. In one case, an exceptionally small multiplier is represented by a fixed point number with the binary point to the left of bit 31. These fixed point formats correspond to representations with 16, 24 and 32 significant binary digits to the right of the binary point and care must be taken to align the binary points correctly during arithmetic operations and to preserve the low order bits where truncation would otherwise occur. The fixed point representation has been chosen because of the superior speed of arithmetic operations. Floating point calculations involve an extra implied operation since each value is handled as a mantissa and an exponent and results must be renormalised following each arithmetic operation. The type of number representation used in a calculation will be indicated in the sections which follow.

3.10 The Z-Transformed Power System Model

The various power system component blocks were discussed earlier in this chapter. Each of these blocks will now be considered in turn and the corresponding Z-transformed equations derived. In the following subsections, two additional values will be associated with any time constant, T_a, which will be denoted α_a and β_a where $\alpha_a = e^{-T/Ta}$, $\beta_a = 1-\alpha_a$ and T is the sampling period.

3.10.1 The Generator

A fifth order representation of the synchronous generator, equations (3.22) to (3.26) was chosen for this simulation and these equations may be rearranged into the form of transfer functions:

$$e''_d(s) = \frac{1}{1+sT''_{qo}} [(x_q - x''_q) i_q(s)] \qquad (3.84)$$

$$e'_q(s) = \frac{1}{1+sT'_{do}} [V_f(s) - (x_d - x'_d) i_d(s)] \qquad (3.85)$$

$$e''_q(s) = \frac{1}{1+sT''_{do}} [e'_q(s) - (x'_d - x''_d) i_d(s)] \qquad (3.86)$$

Each of these equations may then be transformed in the manner of equation (3.83), such that:

$$e''_d(Z) = \frac{\beta''_{qo}}{1-\alpha''_{qo} Z^{-1}} [(x_q - x''_q) i_q(Z)] \qquad (3.87)$$

$$e'_q(Z) = \frac{\beta'_{do}}{1-\alpha'_{do} Z^{-1}} [V_f(Z) - (x_d - x'_d) i_d(Z)] \qquad (3.88)$$

$$e''_q(Z) = \frac{\beta''_{do}}{1-\alpha''_{do} Z^{-1}} [e'_q(Z) - (x'_d - x''_d) i_d(Z)] \qquad (3.89)$$

These equations may then be rearranged into a form where the current values of e''_d, e'_q and e''_q are expressed in terms of their previous

values and the current values of V_f, i_d and i_q. Thus:

$$e''_d(Z) = \beta''_{qo} [(x_q - x''_q)\, i_q(Z)] + \alpha''_{qo}\, Z^{-1}\, e''_d(Z) \qquad (3.90)$$

$$e'_q(Z) = \beta'_{do} [V_f(Z) - (x_d - x'_d) i_d(Z)] + \alpha'_{do} Z^{-1}\, e'_q(Z) \qquad (3.91)$$

$$e''_q(Z) = \beta''_{do} [e'_q(Z) - (x'_d - x''_d) i_d(Z)] + \alpha''_{do} Z^{-1}\, e''_q(Z) \qquad (3.92)$$

Providing current values are available for i_d, i_q and V_f, the equations (3.90) to (3.92) may be evaluated in order to provide current values for e''_d, e'_q and e''_q. Equations (3.90) to (3.92) may be written as:

$$e''_d(Z) = G_1 i_q(Z) + G_2 Z^{-1} e''_d(Z) \qquad (3.93)$$

$$e'_q(Z) = G_3 V_f(Z) + G_4 i_d(Z) + G_5 Z^{-1} e'_q \qquad (3.94)$$

$$e''_q(Z) = G_6 e'_q(Z) + G_7 i_d(Z) + G_8 Z^{-1} e''_q \qquad (3.95)$$

where $G_1 = \beta''_{qo}(x_q - x''_q)$, $G_2 = \alpha''_{qo}$, $G_3 = \beta'_{do}$, $G_4 = -(x_d - x'_d)G_3$, $G_5 = \alpha'_{do}$, $G_6 = \beta''_{do}$, $G_7 = -(x'_d - x''_d)G_6$, and $G_8 = \alpha''_{do}$. The multipliers $G_1 - G_8$ are represented by 32 bit fixed point binary values with the binary point to the left of bit 23.

3.10.2 The Excitation System

Equations (3.33) to (3.35) describe the excitation system

representation. These equations may be expressed in the form of transfer functions as:

$$V_f(s) = \frac{K_1}{1+sT_e} V_{in}(s) \qquad (3.96)$$

and

$$Vs(s) = \frac{K_2 s}{1+sT_s} V_f(s) \qquad (3.97)$$

which may be transformed to:

$$V_f(Z) = \frac{K_1\beta_e}{1-\alpha_e Z^{-1}} V_{in}(Z) \qquad (3.98)$$

and

$$V_s(Z) = \frac{K_2(1-Z^{-1})\beta_s}{T(1-\alpha_s Z^{-1})} V_f(Z) \qquad (3.99)$$

using $(1-Z^{-1})/T$ as an approximation for the operator s on the top of equation (3.97). Rearranging equations (3.98) and (3.99) and substituting equation (3.35) for V_{in} gives:

$$V_f(Z) = \beta_e.K.[V_{ref}(Z) + V_i(Z) - V_s(Z) - V_t(Z)] + \alpha_e Z^{-1} V_f(Z) \qquad (3.100)$$

and

$$V_s(Z) = \frac{\beta_s K_2}{T} V_f(Z) + \alpha_s Z^{-1} V_s(Z) - \frac{\beta_2 K_2}{T} Z^{-1} V_f(Z) \qquad (3.101)$$

Substituting equation (3.101) into (3.100) to remove the term in V_s gives:

$$V_f(Z) =$$

$$\beta_e K_1 [V_{ref}(Z)+V_i(Z)-V_t(Z)- \frac{(1-Z^{-1})\beta_s K_2}{T} V_f(Z)-\alpha_s Z^{-1}V_s(Z)]+\alpha_e Z^{-1}V_f(Z) \quad (3.102)$$

Collecting terms in V_f yields:

$$V_f(Z)\left[1+\frac{K_1 K_2 \beta_s \beta_e}{T}\right] = \beta_e K_1 [V_{ref}(Z)+V_i(Z)-V_t(Z)] - \beta_e K_1 \alpha_s Z^{-1}V_s(Z) + \left[\alpha_e+\frac{K_1 K_2 \beta_s \beta_e}{T}\right]Z^{-1}V_f(Z) \quad (3.103)$$

so that:

$$V_f(Z) = A_1[V_{ref}(Z) + V_i(Z) - V_t(Z)] + A_2 Z^{-1}V_s(Z) + A_3 Z^{-1}V_f(Z)$$

where $A_1 = K_1\beta_e/(1+a)$, $A_2 = -\alpha_s A_1$, $A_3 = (\alpha_e+a)/(1+a)$ (3.104)

and $a = K_1 K_2 \beta_s \beta_e / T$

Having calculated $V_f(Z)$ from equation (3.104), then equation (3.101) may be used to calculate $V_s(Z)$. Care must be taken when the field voltage reaches either of its limits to ensure that the term $(1-Z^{-1})V_f(Z)$ in equation (3.99) becomes zero so that the rate of change of field voltage, which this term represents, is zero. Equation (3.101) may be written as:

$$V_s(Z) = A_4(1-Z^{-1})V_f(Z) + A_5 Z^{-1}V_s(Z) \quad (3.105)$$

where $A_4 = \frac{\beta_g K_2}{T}$ and $A_5 = \alpha_g$. The multipliers A_1-A_3 are represented by 32 bit fixed point binary values with the binary point to the left of bit 23, while the multipliers A_4 and A_5 are represented by 32 bit fixed point binary values with the binary point to the left of bit 15.

3.10.3 Generator Dynamics

The dynamics of the generator associated with the inertia of the generator shaft are represented by equation (3.13), which may be written as:

$$\delta(s) = \frac{1}{s} \Upsilon(s) \qquad (3.106)$$

and

$$\Upsilon(s) = \frac{1}{Ms} [T_{out}(s) - T_{elec}(s) - T_{losses}] \qquad (3.107)$$

Each of these equations may be transferred and rearranged so that:

$$\delta(Z) = Z^{-1}\delta(Z) + T\Upsilon(Z) \qquad (3.108)$$

$$\Upsilon(Z) = Z^{-1}\Upsilon(Z) + T/M[T_{out}(Z) - T_{elec}(Z) - T_{losses}(Z)] \qquad (3.109)$$

Equation (3.109) must be evaluated before equation (3.108), and a small amount of interfacing error is introduced into the equation by the various torques, T_{out}, T_{elec} and T_{losses}, since they depend on quantities evaluated in the current sample period. By comparison to the electrical dynamics of the machine, the dynamics of the machine rotor are very slow and so these interfacing errors are small and may be ignored. The time step, T, and the constant T/M, are both

represented by 32 bit fixed point binary values with the binary point to the left of bit 23.

3.10.4 The Turbine Prime Mover and Governor

Both the steam and hydro turbines are represented by equations (3.38) to (3.41), which may be rearranged as:

$$T_{out}(s) = \frac{1}{1+sT_b} T_{in}(s) \qquad (3.110)$$

and

$$T_{in}(s) = \frac{1}{1+sT_a} [Gov_{ref}(s) - T_i(s) - K_T Y(s)] \qquad (3.111)$$

Each of these equations may be transformed and rearranged to give:

$$T_{out}(Z) = \beta_b T_{in}(Z) + \alpha_b Z^{-1} T_{out}(Z) \qquad (3.112)$$

and

$$T_{in}(Z) = \beta_a [Gov_{ref}(Z) - T_i(Z) - K_T Y(Z)] + \alpha_a Z^{-1} T_{in}(Z) \qquad (3.113)$$

Limits on both the size of T_{in} and the rate of change of T_{in} are applied in order to implement the position and rate limits on the valve gear. The constants α_a, β_a, α_b and β_b are all represented by 32 bit fixed point binary values with the binary point to the left of bit 23. The interfacing error present in the generator dynamics mentioned earlier may be reduced by integrating equations (3.113) and

(3.112) before equations (3.109) and (3.108). This does not completely remove the interfacing error since the current value of T_{in} is dependent on the current value of Υ and the current value of Υ depends on the current value of T_{in}. This ambiguity could be resolved by substituting equations (3.112) and (3.113) into equation (3.109) in place of the term in T_{out}. However, it is not felt that the improvement in the accuracy of the simulation justifies the extra computation involved.

3.10.5 The Diesel Engine Prime Mover and Governor

The diesel engine and governor are represented by equations (3.42) to (3.45), which may be rearranged as:

$$T_{out}(s) = \frac{1}{1+sT_c} T_{in}(s) \qquad (3.114)$$

and

$$T_{in}(s) = \frac{K_g(1+sT_a)}{1+sT_b} [Gov_{ref}(s) - \omega(s) - T_i(s)] \qquad (3.115)$$

Equation (3.109) may be written in terms of the rotor speed, ω, instead of the rotor slip speed, Υ, so that:

$$\omega(Z) = Z^{-1}\omega(Z) + T/M[T_{out}(Z) - T_{elec}(Z) - T_{losses}(Z)] \qquad (3.109a)$$

Equations (3.114) and (3.115) may be transformed to give:

$$T_{out}(Z) = \frac{\beta_C}{1-\alpha_C Z^{-1}} T_{in}(Z) \qquad (3.116)$$

and

$$T_{in}(Z) = \frac{K_g\beta_b(1+\frac{1-Z^{-1}}{T}T_a)}{1-\alpha_b Z^{-1}}\,[Gov_{ref}(Z)-\omega(Z)-T_i(Z)] \qquad (3.117)$$

These equations may be rearranged so that:

$$T_{out}(Z) = \beta_c T_{in}(Z) + \alpha_c Z^{-1} T_{out}(Z) \qquad (3.118)$$

and

$$\begin{aligned} T_{in}(Z) &= K_g\beta_b(1+\frac{1-Z^{-1}}{T}T_a)[Gov_{ref}(Z)-\omega(Z)-T_i(Z)] + \alpha_b Z^{-1} T_{in}(Z) \\ &= K_g\beta_b(1 + T_a/T)[Gov_{ref}(Z)-\omega(Z)-T_i(Z)] - \\ &\quad K_g\beta_b\, T_a/T\, Z^{-1}[Gov_{ref}(Z)-\omega(Z)-T_i(Z)] + \alpha_b Z^{-1} T_{in}(Z) \end{aligned} \qquad (3.119)$$

The expression for T_{in} may not be calculated at this point since it contains a term in the current rotor speed which, in turn, contains a term in the current prime mover output, T_m. Substituting equation (3.118) into equation (3.109a) gives:

$$\omega(Z) = Z^{-1}\omega(Z) + T/M[(\beta_c.T_{in}(Z)+\alpha_c.Z^{-1}.T_{out}(Z))-T_{elec}(Z)-T_{losses}(Z)] \qquad (3.120)$$

which may be substituted into equation (3.119) to replace the term in ω, giving:

$$T_{in}(1+K_g\beta_b[1+T_a/T](T/M)\beta_c) = K_g\beta_b[1+\frac{(1-Z^{-1})}{T}T_a][Gov_{ref}(Z)-T_i(Z)] +$$
$$K_g\beta_b[1+T_a/T](T/M)[T_{elec}(Z) + T_{losses}(Z)] -$$
$$K_g\beta_b[1+T_a/T](T/M)\alpha_3 Z^{-1}T_{out}(Z) -$$
$$K_g\beta_b Z^{-1}\omega(Z) + \alpha_b Z^{-1}T_{in}(Z) \qquad (3.121)$$

Thus, the diesel engine and governor are simulated by the following equations:

$$T_{in}(Z) = D_1 [Gov_{ref}(Z) - T_i(Z)] + D_2 Z^{-1} [Gov_{ref}(Z) - T_i(Z)] +$$
$$D_3 [T_{elec}(Z) + T_{losses}(Z)] + D_4 Z^{-1} T_{out}(Z) + D_5 Z^{-1}\omega(Z) +$$
$$D_6 Z^{-1}T_{in}(Z) \qquad (3.122)$$

and

$$T_{out}(Z) = D_7 T_{in}(Z) + D_8 Z^{-1}T_{out}(Z) \qquad (3.123)$$

where $D_1 = K_g\beta_b[1 + T_a/T]/D_9$, $D_2 = -K_g\beta_b(T_a/T)/D_9$, $D_3 = D_1\ T/M$, $D_4 = -D_3\ \alpha_3$, $D_5 = -K_g\beta_b/D_9$, $D_6 = \alpha_2/D_9$, $D_7 = \beta_3$, $D_8 = \alpha_3$ and $D_9 = [1 + K_g\beta_b\beta_c[1 + T_a/T]T/M]$. The constants D_1-D_8 are represented by 32 bit fixed point binary values with the binary point immediately to the left of bit 23. Limits may be applied to T_{in} and T_{out} as appropriate.

3.10.6 Algebraic Equations

So far, only the dynamic plant equations have been considered. A further set of algebraic equations exist which must be solved simultaneously with the dynamic equations. These equations give

expressions for electrical torque, T_{elec}, the voltages V_d, V_q and V_t, the currents i_d and i_q, and the extra control signals, rotor acceleration ($p^2\delta$) and transient electrical power (P_e), which are made available to the excitation and governing systems. The equation for the air gap torque, equation (3.27), was given earlier in this chapter. The four equations, (3.25), (3.26), (3.48) and (3.49), relate the direct and quadrature axis voltage and currents, and the load angle δ. These equations may be combined, to eliminate V_d and V_q and expressed in matrix form as:

$$\begin{bmatrix} e''_d - V_b\sin\delta \\ e''_q - V_b\cos\delta \end{bmatrix} = \begin{bmatrix} (r_a + R_T) & -(X_T + x''_q) \\ (X_T + x''_d) & (r_a + R_T) \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} \qquad (3.124)$$

which may be inverted to express the currents i_d and i_q in terms of the busbar voltage V_b, the subtransient voltages e''_d and e''_q, and the load angle δ. Thus:

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \frac{1}{(r_a+R_T)^2+(X_T+x''_q)(X_T+x''_d)} \begin{bmatrix} (r_a + R_T) & (X_T + x''_q) \\ -(X_T + x''_d) & (r_a + R_T) \end{bmatrix} \begin{bmatrix} e''_d - V_b\sin\delta \\ e''_q - V_b\cos\delta \end{bmatrix}$$

(3.125)

and the currents so obtained may then be substituted into equations (3.48) and (3.49) to obtain the axis components of the terminal voltage, V_d and V_q. These voltage components are then used to obtain the terminal voltage magnitude according to equation (3.12).

$$V_t = \sqrt{V_d^2 + V_q^2} \qquad (3.12)$$

This calculation is the only part of the simulation that is performed using floating point arithmetic, V_d and V_q are both converted from fixed point to floating point, squared, summed, square rooted and then converted back to a fixed point representation of the terminal voltage magnitude V_t.

The rotor acceleration is calculated from the difference in two speed values, rather than using the expression for the rate change in rotor speed given by rearranging equation (3.107). This action more closely corresponds to the action of the slotted disc type speed and acceleration transducer used by Hazell (3-4) and introduces the type of phase lag associated with such a transducer. Thus:

$$\hat{p^2\delta}(Z) = \frac{(1 - Z^{-2})\, Y(Z)}{2T} \qquad (3.126)$$

is used to estimate the rotor acceleration, $\hat{p^2\delta}$.

The electrical power at the machine terminals is calculated from the standard equation for the terminal power of an electrical machine.

$$P_e = V_d . i_d + V_q i_q \qquad (3.127)$$

In practice there will be some dynamics associated with the transducers measuring the terminal power. For the purposes of this investigation, it is assumed that electrical power can be accurately measured at the machine terminals and that this power is given by equation (3.127).

3.11 Summary

The dynamic and algebraic equations used to simulate the component parts of a power system have been described. A number of numerical integration techniques were then considered, and of these an implicit integration method, based on the Z-transform was chosen to implement the numerical solution of the plant equations. The plant equations were then transformed into the discrete time domain and written in a form suitable for numerical integration. As presented here, there is a small amount of interfacing error associated with the coupling of the electrical and mechanical systems. At the start of each integration loop, all the algebraic equations may be solved by using the dynamic quantities from the previous integration step. This introduces a one sample interface error, which is insignificant provided that the integration step length is short.

The prime mover, governor and rotor dynamic equations may then be applied to step the machine speed, ω, or slip speed, Y, and the load angle, δ, forward by one integration step length. Next the excitation system equations may be applied to establish the field voltage for the current step, which in turn is used to evaluate the direct and quadrature axis transient and subtransient voltages prior to the next iteration of the simulation.

References

3-1 Park, R.H.: "Two-reaction theory of synchronous machines; Generalised Methods of Analysis - Part 1", Trans. Amer. Inst. Elect. Engrs., 1929, Vol.48, pp716, Pt.2, 1933, Vol.52, pp352.

3-2 Humpage, W.D. and Saha, T.N.: "Digital-computer methods in dynamic-response analysis of turbogenerator units", Proc.IEE, Vol.114, 1967, pp1115.

3-3 Pal, M.K.: Mathematical methods in power system stability studies", PhD Thesis, University of Aston in Birmingham, Dec. 1971.

3-4 Hazell, P.A.: "Co-ordinate excitation control and governing of turbogenerators", PhD Thesis, University of Bath, 1981.

3-5 Ham, P.A.L., Jenkins, K. and Mikhal, S.E.: "Performance and control capabilities of electrohydraulic governing systems for steam turbine generators", Reyrolle Parsons Review, Vol.2, No.5, Summer 1976.

3-6 Stronach, A.F.: "The analysis and experimental study of the transient behaviour of marine and off-shore electromechanical systems", PhD Thesis, University of Aberdeen, 1980.

3-7 Wyatt, T.G.: "Micro-machine simulation of a marine electrical power supply system", MSc Thesis, University of

Bath, 1981.

3-8 Stott, B.: "Power system dynamic response calculations", Proc.IEE, Vol.67, No.2, February 1979.

3-9 Humpage, W.D., Wong, K.P. and Lee, Y.W.: "Solution methods for transient and dynamic stability", Proc.IEEE, Vol.62, pp.951-958, July 1974.

3-10 Dandeno, P.L. and Kandur, P.: "A non-iterative transient stability program including the effects of variable load-voltage characteristics", IEEE Trans., PAS-90, pp.96-101, Jan/Feb 1971.

3-11 Tanner, D.G.: "Real-time simulation of power systems", PhD Thesis, 1982, University of Bath.

3-12 "MC68000 16 Bit Microprocessor Users Manual", 2nd Edition, Motorola Inc., 1980.

3-13 "A Proposed Standard for Binary-Floating Point Arithmetic", (Draft of IEEE Task P754), Computer, Vol.14, No.3, May 1981, pp.51-62.

3-14 "Theory and Application of the Z-transform Method", E. I. Jury, Pub. by Robert E. Krieger, 1964.

3-15 "Modern Numerical Methods for Ordinary Differential Equations", G. Hall & J. M. Watt, Clarendon Press, Oxford, 1976.

CHAPTER 4

THE REAL-TIME SIMULATOR

4.1 Introduction

The increasing complexity and speed of digital integrated circuits, in particular, memory devices and the latest generation of 16-bit microprocessors, make it possible to consider their use in solving large computational problems which have previously been confined to solution by mainframe computer due to their size in terms of storage and processing requirements. Dedicated processing systems may be designed relatively cheaply from these circuits and, in this particular study, the objective of such a system is to solve, in real-time, the dynamic and algebraic equations (developed in Chapter 3) which describe a small power system. The term 'real-time' is widely used in many contexts but for the purposes of this investigation 'real-time' is taken to mean that the time taken to solve all the equations associated with the plant is less than the time step of the particular integration method used. This means that the simulation must be capable of generating data at the same rate as the real plant which the computer system is simulating. The power systems considered consist of a prime mover with governor control, a synchronous generator fitted with automatic excitation control and a load. These components have already been discussed in Chapter 3 and methods for numerical integration of the dynamic equations have been described.

This Chapter describes the development of both the hardware and the software of a Real-Time Power System Simulator, which combine to solve the power system equations in real-time. The hardware requirements for the simulator and various possible configurations are discussed, with a detailed description of the particular hardware used. The software for the simulator may be considered in three groups: the system software, which provides the environment in which the simulation is executed; the application software, which comprises of the simulation software itself and the necessary monitoring software; and the support software which eases the debugging, loading and modification of the real-time simulation. Finally, the operation of the hardware and software components together as a simulation is discussed.

4.2 Simulator Hardware

A variety of possible solutions exists to the problem of choosing hardware capable of simulating the power system equations given in Chapter 3. The equations could be left in the operational form in which they were introduced, to be solved using analogue (4-1 to 4-3) or hybrid analogue/digital computers. An analogue computer simulation results in a very inflexible solution which is very labour intensive to set up or modify. The results from such a simulation may be saved by plotting directly onto paper or by recording on magnetic tape. Part of the motivation for the real-time simulation is to enable stability and optimisation studies to be performed, rapidly, on the system under investigation. These studies require various parameter changes based on the results of previous simulations, and the manual modification of the various potentiometer settings on an

analogue computer becomes a tedious process. It is possible to program a hybrid computer to make the necessary parameter modifications as the study progresses. However, the cost of a hybrid computer is prohibitive and an all digital solution to the hardware problem has been chosen. Three main digital solutions are available, namely, array processors, mainframe computers, or some form of multiple microprocessor architecture.

An array processor consists of a single execution unit controlling an array of arithmetic and logic units (ALUs). As such, a single program instructions is executed simultaneously on several sets of data (Single Instruction Multiple Data, SIMD). This type of processing hardware is ideal for handling matrix calculations, where, for example, in a matrix multiply, each ALU is responsible for calculating a single element of the product matrix. In this case, each ALU would have to multiply a row of the lefthand matrix by a column of the righthand matrix and each of these operations is entirely independent of the other row and column products. Thus, the product matrix elements may be calculated simultaneously by the use of an array processor which performs the same operation on multiple sets of data. There is obviously a finite limit to the number of ALUs built into an array processor and such a processor is at its most efficient when all its ALUs are in use. In the matrix multiply example, should there be more elements in the product matrix than there are ALUs, then a number of passes over the data will be necessary, each pass calculating new elements of the product matrix. The flow of data to and from an array processor is very complicated as separate pathways between the system memory and each ALU must exist. For this reason, array processors are generally slave processors

which increase the performance of a host processor. The host is then responsible for retrieving data from the host memory and supplying it to pipelines associated with each ALU. When each ALU requires a new operand it removes a data item from the front of its pipeline. It is the responsibility of the host to ensure that this data is presented in the correct order. The results of intermediate calculations are held in registers internal to each ALU. When the calculation is complete, it is the responsibility of the host to retrieve each result from the ALUs and return it to host memory. Clearly, large speed improvements are possible with such architectures, where a number of similar, independent, calculations may be performed simultaneously. Despite this speed improvement, an array processor is not a stand-alone device, and may be considered to be a hardware 'add on' which improves the performance of the host. The array processor is purely a 'number cruncher' which makes no decisions regarding program flow based on the calculations it performs. It merely handles arithmetic calculations on complex data structures. Programming a combination of host and array processor is not a simple matter. Apart from the host software needed to steer data to and from the array processor, there is also the software/firmware required by the array processor to determine the function it performs on the data arriving via its pipelines. The cost of array processing is high, not just in terms of the cost of hardware (the array processor and the host) but in terms of the software effort necessary to program the device. Expansion of such a device to handle more complex problems at the same execution speed requires either the use of more ALUs or replication of the entire array processor/host combination. Array processing is therefore not an attractive solution when the cost of the system is important.

The traditional approach to the numerical solution of power system equations and those of other large engineering systems has been to use the facilities of a mainframe computer, such as those found serving Universities or industrial concerns. In general, the simulation is written in a high level language and, once compiled, it may be run as a single task amongst the many tasks such a computer is executing on behalf of its many users. The raw processing power of a mainframe, typically in excess of 10 million instructions per second (MIPS), is more than adequate for the real-time simulation of many large engineering systems. However, exclusive use of a mainframe computer is prohibitively expensive. In general, such computing systems are equipped with large quantities of random access memory (RAM) and disc storage in order to service a large number of users. The user environment set up by the system software must protect each user from the other users and it is generally not possible to write low level machine code to improve the execution speed of a task. Due to the time-sharing nature of such a system it is impossible to guarantee a 'real-time' response from the system. This and the cost of exclusive use of a mainframe computer make it an unattractive solution.

Distributed computing systems in general contain more than one processing unit. The term 'distributed computing' is used loosely here to encompass a whole variety of architectures based around the use of many processing elements, working together in some manner to solve a problem. Distributed processing schemes increase the instruction throughput of a computing installation since a computational problem may be divided into a number of tasks and each task may be executed on a separate processor. Each of these tasks

would possibly wait for data that is the result of a computation taking place in another task. The use of multiple processing elements has some similarities with the use of array processors, in that truly simultaneous calculations can take place. However, in this case, the nature of the simultaneous calculations may be entirely different (Multiple Instruction Multiple Data, MIMD), since each ALU has its own execution unit. Thus, associated with each processing element is an 'intelligence' not present in an array processor, which allows each element to make decisions and steer its own execution, without disturbing the execution of other calculations elsewhere in the system. The individual manner in which each of the processing elements may perform its calculations eases the problem of debugging the application software that runs on the system. Also, depending on the specific architecture chosen, expansion of the system to solve more complex problems at the same speed may simply require the addition of further processing units. This type of approach will encourage the use of modular hardware and software, whereby a computation may be distributed in some 'optimum' fashion around a number of identical pieces of hardware. If the problem expands, requiring the use of more hardware to achieve the required computational speed, new program modules may be written and all the modules may then be redistributed around the new hardware configuration.

With the falling price and increasing complexity of currently available microprocessor integrated circuits and their supporting devices, it was decided that the simulator should be some form of distributed microprocessing system.

4.3 A Microprocessor Based Distributed Computing System

One of the first choices that has to be made when opting for a microprocessor based solution to a problem is which microprocessor to use. Four main contenders were considered: the Texas Instruments TMS9900; the Intel I8086; the Motorola MC68000; and the Zilog Z8000; all of which have at least 16-bit arithmetic capabilities. As a result of benchmark tests by Tanner (4-4) it was decided to use the Motorola device. This decision is based on the speed with which the MC68000 can perform arithmetic operations, its large instruction set and the potentially fast bus access speed of its asynchronous bus interface.

The Motorola MC68000 is a powerful 16-bit microprocessor (4-5) which can directly access an address space of 16 Megabytes. It supports operands of five different types; bit; byte (8 bits); word (16 bits); longword (32 bits); and binary coded decimal (BCD). The instruction set of the MC68000 contains instructions to perform; data movement; two's complement integer arithmetic; logical operations; shift and rotate operations; bit manipulation; branch and jump operations; and system control and privileged operations. The programmers model of the MC68000 consists of a 16-bit status register and eighteen 32-bit registers comprising eight general purpose data registers, seven general purpose address registers, two stack pointers and a program counter. The processor supports two levels of privilege and seven levels of prioritised, vectored interrupt. Exception processing is directed via 256 exception vectors, most of which are available for handling vectored interrupt exceptions. Some of the exception vectors are reserved for use by such exceptions as

reset, bus error (when an accessed location fails to respond), address error (when a word or longword access is attempted at an odd valued address) and a variety of trap instructions and conditions such as illegal instruction, divide by zero, privilege violation and a trace exception which allows software single stepping. In order to ease the design procedure, the MC68000 supports two types of bus interface. The asynchronous bus interface is a high performance means of accessing memory devices and peripheral devices of the MC68000 family. The synchronous bus interface may be used to directly interface the MC68000 to the large range of peripheral devices belonging to the Motorola MC6800 family of support devices (4-6). The MC68000 contains bus arbitration logic which aids in the transfer of control of the microprocessor buses to multiple bus masters, with the possibility of bus contention being resolved while the bus is still active and assigned to another bus master. Thus, the MC68000 is capable of supporting a large range of devices and peripherals such as terminals, rotating disc storage, memory management hardware and direct memory access controllers. These features greatly simplify the task of the system designer.

4.3.1 System Specification

Once the choice of microprocessor device has been made, it is necessary to specify the requirements of the system into which it is to be built. The fundamental requirement is that each of the MC68000 based processing units should be capable of operating in a multiprocessor configuration, with high speed transfer of data between the processing units. In order to make the system flexible, it was considered that some form of open ended architecture should be

used to facilitate easy expansion. On these grounds, it was felt that a 'star' connected processing system, with communication between all processors directed through a central, intelligent, bus controller, was undesirable. In its simplest form such a star connected system could consist of a number of identical processing units, each of which supports at least one serial port (for example, an RS232-C asynchronous port), all coupled to a further, possibly similar, processing unit fitted with sufficient serial ports to allow individual connection of each of the other processing units. The processing unit at the centre of the star connection represents a common mode of failure should it break down and it can be identified as a potential bottleneck should the volume of communication traffic become large. Expansion is also restricted to the number of serial ports available on the controlling unit at the centre of the star.

Fig. 4.1 shows a number of bus orientated multiprocessor configurations. Fig. 4.1(a) shows a tightly coupled system where all the system resources, memory, disc controllers and input/output devices reside on the system bus. When multiple bus masters, such as central processing units (CPUs) or disc controllers fitted with direct memory access (DMA) facilities, contend for the ownership of the bus in order to exchange data, the bus controller must resolve which of the requesting bus masters actually acquires the bus. On the surface such a system appears to meet the specification. It is expandable and, in theory, more memory, CPUs or peripheral devices may be connected to the bus, with high speed communication between CPUs accomplished simply by data movement. There are, however, practical limits to the number of devices that can be connected to the bus and a much more serious limitation to the overall expansion of the

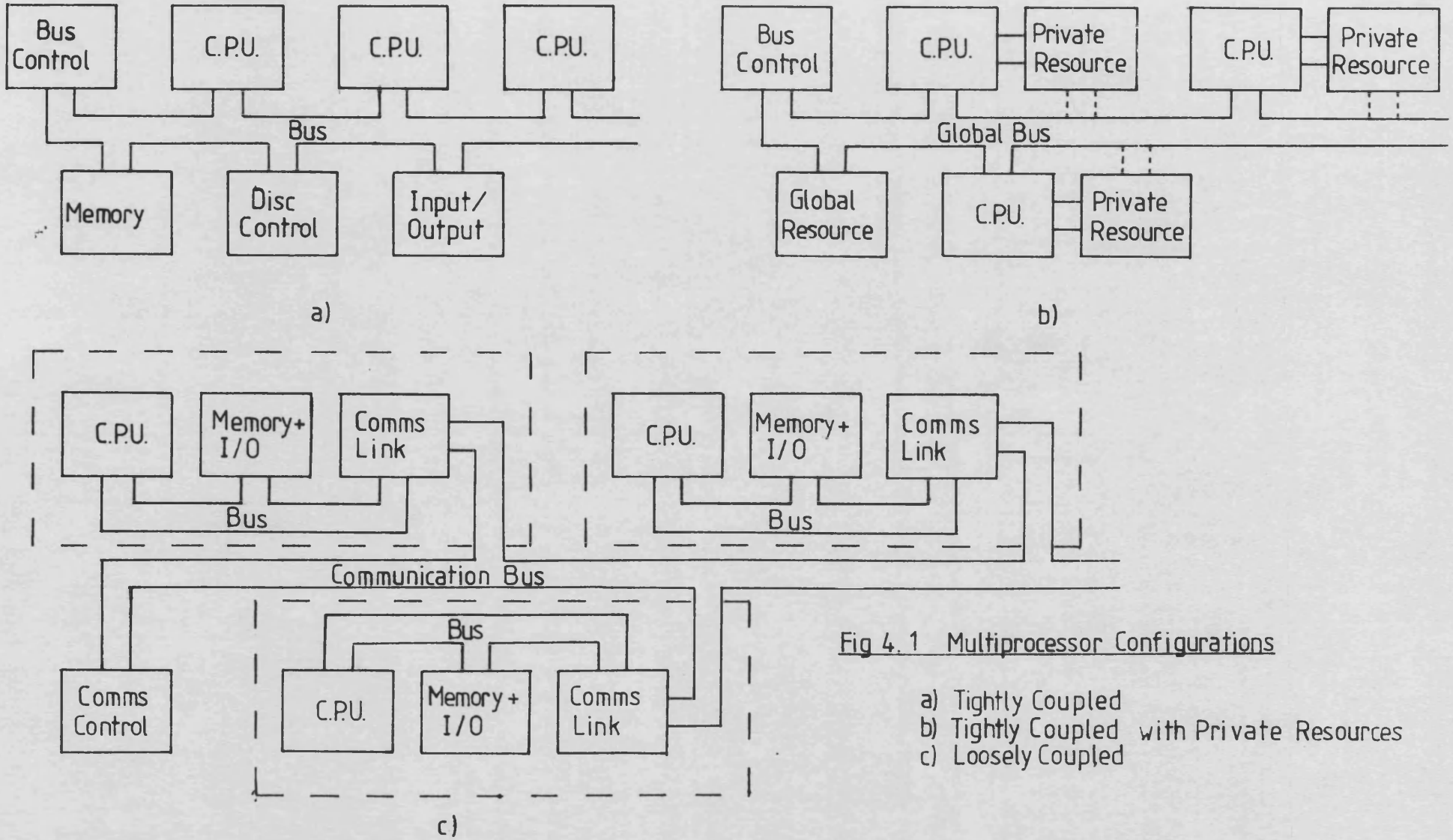


Fig 4.1 Multiprocessor Configurations

a) Tightly Coupled
b) Tightly Coupled with Private Resources
c) Loosely Coupled

system is imposed by the bus bandwidth. In order to operate, each CPU must fetch instructions and data from the memory attached to the bus. This has both advantages and disadvantages. The major advantage is that if separate CPUs are identical, they may share both program and data areas. This means that any sub-task of the overall computation may be executed by any CPU. The decision as to which CPU executes a particular sub-task can be made dynamically as the computation proceeds and the computation load varies. In a multitasking environment, a suspended task may be resumed by a different processor to that which suspended the task. The disadvantage of this architecture is that each processor must acquire ownership of the bus for every read or write cycle. For the worst case MC68000 instruction, the actual instruction fetch will require 5 bus cycles (the longest MC68000 instruction format is ten bytes long) and, if both source and destination operands are long words held in memory, a further 6 bus cycles are required to obtain both operands and return the result of a computation to memory. Thus, each CPU generates a large volume of traffic on the bus and, in the limit, each CPU will be kept waiting prior to each bus cycle, while the bus cycles of other bus masters (CPUs, DMACs, etc.) are completed. Despite the attractive nature of this regular architecture which permits elegant software to be written, the overall instruction throughput of the system is limited by its bus bandwidth.

A modification to this configuration is shown in Fig. 4.1(b). Associated with each CPU are some private, or dedicated, resources. Aside from some form of local memory, these dedicated resources may include peripheral devices, such as disc controllers, terminal ports, timers and parallel ports. The local memory may be organised in two

forms, either as cache memory or as conventionally addressed, linear memory. A cache memory intercepts read cycles to the global bus to determine whether or not the particular memory location has already been read by its own CPU and whether a valid value for that location is available from the cache memory. If a cache 'hit' occurs, i.e. the cache does in fact contain valid data for the required memory address, then no cycle takes place on the global bus and the data is supplied locally to the CPU by the cache. If the cache 'misses', then the read cycle proceeds as normal by acquiring the global bus and reading from the global memory (part of the global resource). The data so read is captured both by the CPU and by the cache memory. The cache then tags the stored value with the address from which it came, so that a subsequent access to the same location will result in a cache hit. Problems arise from the use of cache memory in a multiprocessor environment where the global bus is shared. First, the cache itself is a complicated piece of hardware and it requires time to respond at the start of a CPU read cycle, so lengthening the time taken to acquire the data if that data is to be acquired from the global bus. A more serious problem arises in a multiprocessor environment if a cached valued is invalidated by a write to the corresponding global address by a different CPU. Caching may be restricted to instruction fetches only in order to avoid this latter problem since it is generally accepted that it is bad practice to write software in such a manner that it is modified during its execution. This 'instruction only' restriction reduces the effectiveness of using cache memory since not all read cycles are subject to the possibility of a cache hit. However, use of instruction cache does reduce much of the traffic on the global bus, so increasing overall instruction throughput beyond the limit

previously imposed by the bus bandwidth, whilst at the same time maintaining the elegant software flexibility offered by the tightly coupled architecture of Fig. 4.1(a).

The use of linearly addressed private memory also offers a reduction in bus traffic. The linear memory forms part of the CPU memory map and both program and data segments may reside there. Such private memory may be very closely coupled to the CPU and it is possible to achieve full speed operation of memory cycles using this arrangement. Only software resident in the global memory may be shared, but making use of shared software in this fashion results in the bus bandwidth problems previously encountered with the arrangement of Fig. 4.1(a). The memory of the global resource is used for the exchange of messages or data while the memory of the private resource is used for program execution and the storage of all the data necessary to make use of the global bus infrequent. Common subroutines and modules must be duplicated throughout the private resources and the possibility of one CPU taking up the execution of a task suspended by another CPU is complicated by the fact that the appropriate software and data segments are not resident on the CPU taking up task execution. In many cases, the private resources (particularly the memory) of a CPU are accessible from the global bus as indicated by the broken lines in Fig. 4.1(b). The addresses, at which access to this memory is made via the global bus, are different for each CPU board and different to that used by the local CPU to make private accesses. Care is therefore necessary if a suspended task is to be resumed by a different CPU and, in general, it is simpler if each task remains resident on one CPU throughout its execution. With the configurations of Fig. 4.1(a) and (b), the bus interface is a

complex section of hardware and many contingencies must be taken into account. For example, a 'deadly embrace' can occur with the configuration of Fig. 4.1(b) if one CPU attempts to access the private resources of a different CPU whilst this second CPU is attempting to access the global bus. The first CPU has access to the global bus, but may not access the private resources of the second CPU until its internal buses are free, which cannot happen until it has completed the access it requires to make to the global bus. Hardware mechanisms can be incorporated to handle such problems so that the programmer remains unaware of their existence.

The configuration shown in Fig. 4.1(c) is a loosely coupled system. Each CPU becomes an entire subsystem, consisting of the processor itself, memory, input/output devices and a communication link. Each subsystem, enclosed by broken lines in Fig. 4.1(c), is capable of functioning independently of the others. On the surface, this arrangement appears to be similar to the arrangement of Fig. 4.1(b). However, the important difference is that the communication bus has no private resources associated with it. Access to the communication bus is governed by the communication controller. A CPU deposits a message for another processor subsystem in its own communication link and it may then continue processing. When directed by the communication controller, the communication link then transfers the message to the communication link of the destination subsystem. The arrival of the message at the destination subsystem then causes an interrupt at the associated CPU. This communication is therefore asynchronous to the software running on both the source and destination subsystems and results in larger transmission delays than those encountered in the previous two arrangements, although the

bandwidth of the communication bus may in fact be higher than that of the system bus of either of the tightly coupled systems.

There is a clear evolutionary trend through these three possible multiprocessor configurations and the choice of one of these configurations in particular is not made easily. It is questionable whether the tightly coupled configuration of Fig. 4.1(a) is capable of meeting the speed requirement in terms of instruction throughput and there is certainly an upper limit placed on this by the bus bandwidth. It is not possible for the CPU elements of Fig. 4.1(a) to function independently since they have no private resources. The use of private resources, Fig. 4.1(b), relieves the bandwidth limitation on instruction throughput and, with the linearly addressed private memory, it is possible with careful design for the processing units to function in a stand-alone mode, independently of the global bus and the other processing units. The loosely coupled architecture of Fig. 4.1(c) is more flexible since each of the subsystems may be tailored to its function. Each subsystem may be rack based so that only the necessary board assemblies for each subsystem may be included, whereas the private resources associated with each CPU in Fig. 4.1(b) often reside on the same board as the CPU and so are present whether they are used or not.

The configuration chosen for the real-time simulator described in this Chapter is that of Fig. 4.1(c). Each subsystem is housed in a standard 19 inch rack, fitted with a 9 or a 22 slot backplane capable of accepting standard and extended double Eurocard sized boards. The entire system is expandable by the use of extra subsystems, and each subsystem has expansion capabilities in terms of the peripherals and

memory that it uses.

4.4 The Microprocessor Based Sub-System

The minimum hardware requirements for each processing subsystem are that it should contain: one MC68000 based CPU; a minimum of 256Kbytes of dynamic random access memory (DRAM); monitor and/or bootstrapping firmware, stored in eraseable programmable read only memory (EPROM); two RS232-C asynchronous serial ports for terminal connections; a real-time counter/timer device; a front panel offering reset and non-maskable interrupt facilities in addition to status display; hardware enhancement of the CPU arithmetic capabilities; and a communication link. An expanded subsystem may also require direct memory access (DMA);' rotating disc storage; and high resolution graphics facilities. Fig. 4.2 shows a block diagram of an extended processing subsystem where the components enclosed inside the broken line are not required for the minimum configuration.

As far as possible, each of the functions within the subsystem block diagram of Fig. 4.2 has been implemented on a single, extended double Eurocard. Subsystems may then be assembled from a series of standard boards. In the subsections which follow, a brief description is given of each of these boards.

4.4.1 The Central Processing Unit

A simplified schematic diagram of the CPU board is shown in Fig. 4.3. The major functional blocks enclosed by the broken line are only available on the latest version of this board. The principle

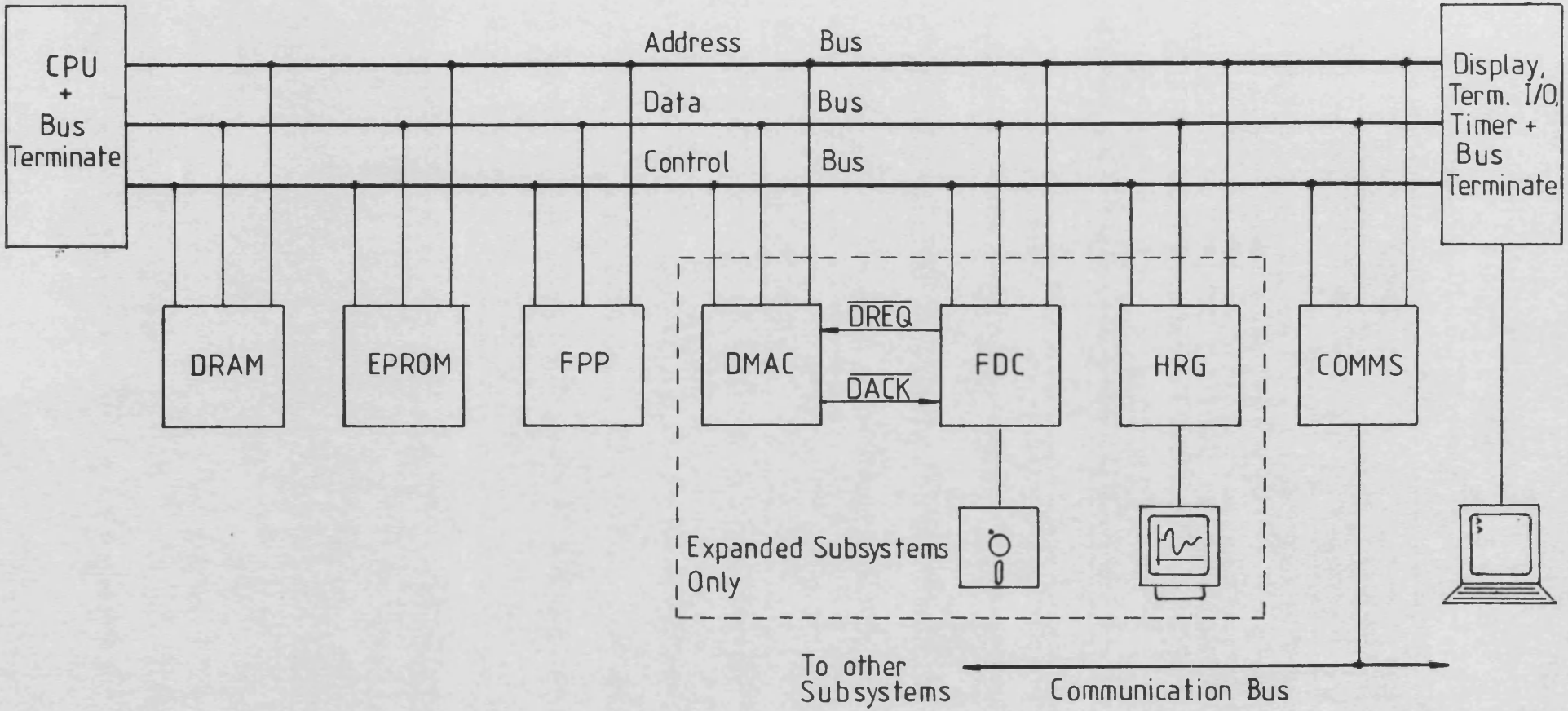


Fig 4.2 Processing Subsystem Architecture

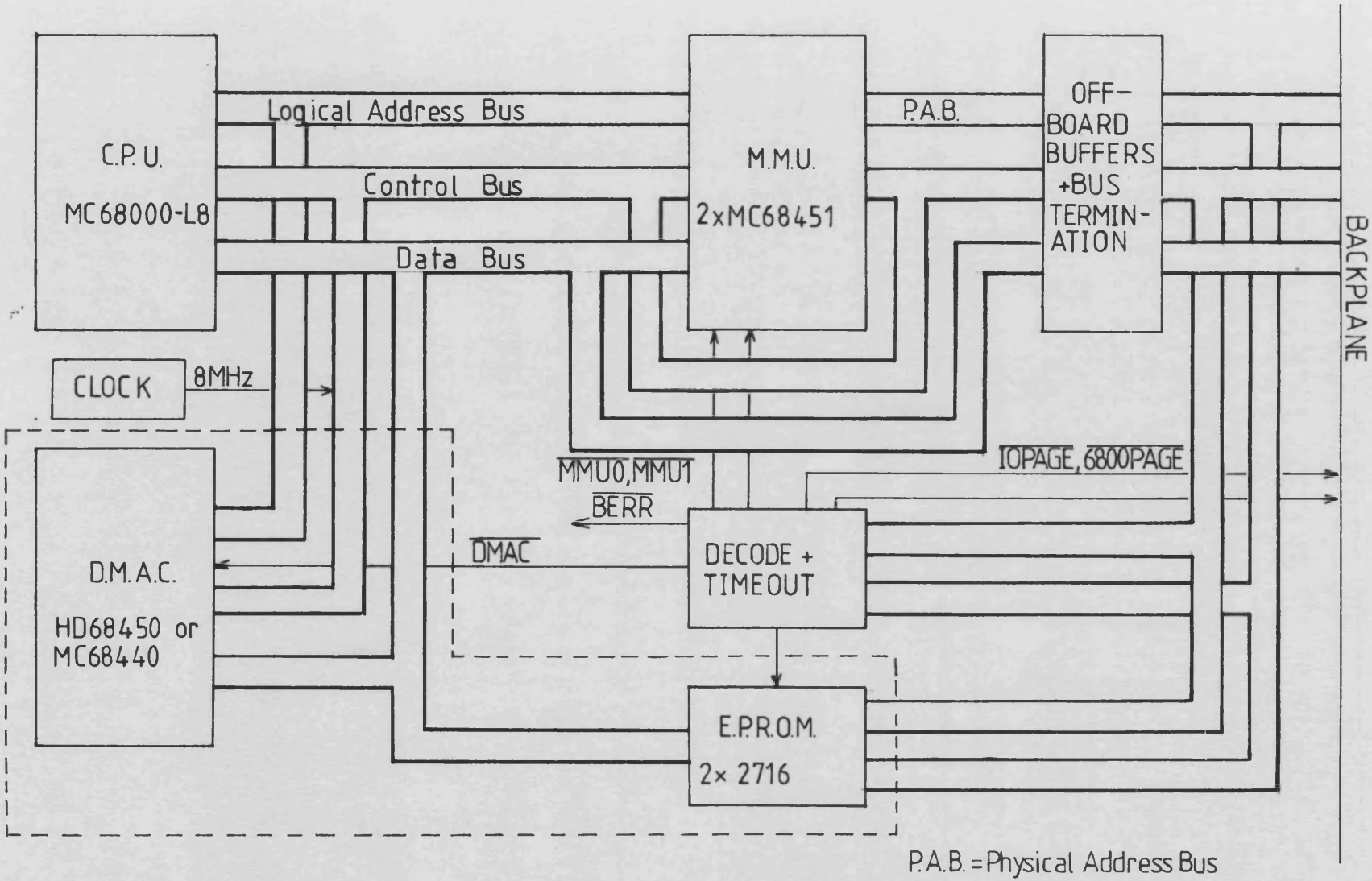


Fig 4.3 The Central Processing Unit

components can be identified as the processor itself, an 8MHz MC68000 microprocessor; clock generation; a direct memory access controller (DMAC); a memory management unit (MMU); bootstrap EPROMs (4Kbytes); and some decoding and time out logic. The MMU, DMAC and EPROMs are all optional and may be removed from the board if not required. The MMU is made up of two Motorola MC68451 integrated circuits which facilitate the translation of logical memory addresses into physical memory addresses and protect memory from illegal accesses. Part of the control bus passes through the MMU logic in order that illegal cycles (e.g. write cycles to write protected memory) may be inhibited. Memory management facilities offer the sort of memory protection required by multi-user, multi-tasking operating systems. Provision of these facilities on the CPU board greatly increases the range of applications a stand-alone subsystem may be put to. On the latest versions of the CPU board direct memory access is offered by the HD68450 or MC68440 DMAC integrated circuits. By providing DMA facilities on the CPU board it is possible for DMAC generated addresses to be translated by the MMU. This reduces the memory mapping problems that arise when disc blocks are loaded into a memory managed system by DMA. A small area of EPROM is also provided on the latest CPU boards which, in a disc based system, is intended to contain all the firmware necessary to load an operating system from the disc device. In non-disc based systems, a large amount of firmware may be required and this may be provided by the EPROM board described in a later section.

The block marked decode generates the necessary device selects for all the on-board devices (MMU, DMAC and EPROM) and it also generates two partial decode signals which identify two areas of

memory reserved for input/output devices. The signal IOPAGE identifies a 4Kbyte area ($840000 to $840FFF) intended for use by I/O devices interfaces using the MC68000 asynchronous bus interface and 256 bytes ($83FFΦΦ to $83FFFF), identified by the signal 6800PAGE, are intended for use by peripherals using the MC6800 type synchronous bus interface. These two partial decode signals reduce the size of the address decoding logic on peripheral boards since, instead of having to decode 23 address lines, use of IOPAGE requires the decode of 11 address lines, while use of 6800PAGE requires the decode of 7 address lines.

Finally, the CPU board generates the main 8MHz system clock and times out all accesses, so that accesses to unpopulated memory locations may be signalled via the Bus Error exception.

4.4.2 The 256Kbyte Memory Board

A schematic diagram of the memory board used for constructing processing subsystems is shown in Fig. 4.4. Many of the signals shown in this figure are unnamed and are intended to indicate a functional dependence rather than to represent a particular signal from the circuit diagram. The main memory array is made up of thirty-two 64K bit dynamic random access memory (DRAM) devices arranged in two banks of 64K x 16 bit words. In addition to the main memory, a further twelve DRAM devices are arranged as two banks of 64K x 6 bit words which are used to store check words generated by the Error Detection and Correction Unit (EDC). All the DRAM devices have access times of 200 nanoseconds and require a multiplexed address bus. The board timing is completely asynchronous, except for arbitration between

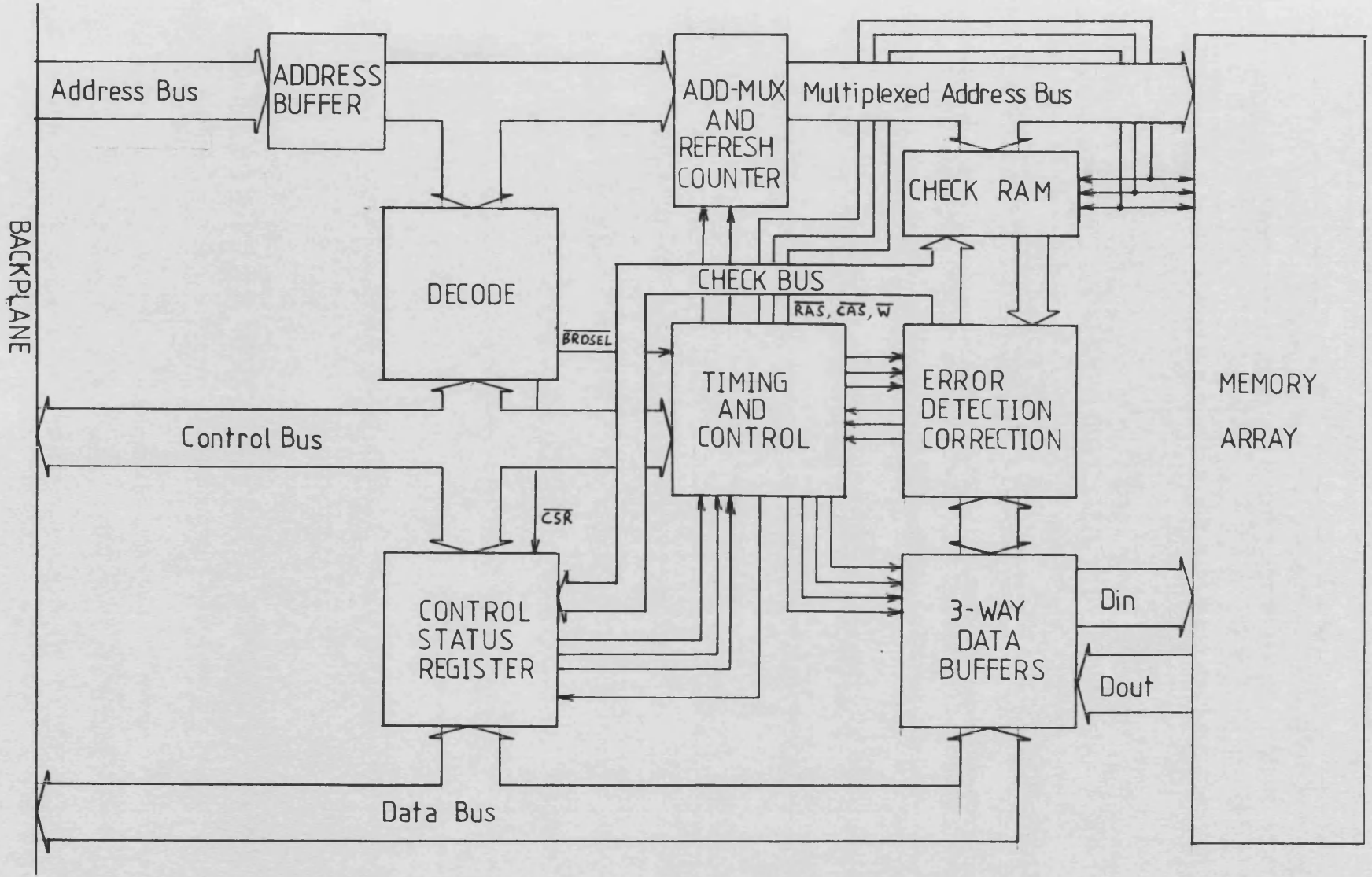


Fig 4.4 Memory Board

backplane and refresh cycles, which makes use of the system clock. Board refresh occurs transparently at regular intervals and may extend the access time to the board in the event of contention between backplane and refresh cycles. All the timing and control signals are generated from an analogue delay line which is triggered at the start of either a refresh cycle or a backplane access. Once a valid address is present on the address bus, and the appropriate data strobes for the particular cycle are active on the control bus, then, assuming that the board decodes and BRDSEL is asserted, a row address is supplied to the memory devices via the address multiplexer. This multiplexer partitions the address bus into row and column address and also generates refresh addresses for refresh cycles. The row address strobe (RAS) is then asserted so that the memory devices latch the row address. The multiplexer is then switched over to supply the column address and, after a short delay to allow the address to settle, the column address strobe (CAS) is asserted so that the complete memory address is latched in the memory devices. Up to this point in the cycle the write signal, W, has been inactive, so data is read from the 16 data bits and the 6 check bits. All memory cycles (except for refresh) internal to this board are read-modify-write cycles, although the backplane cycles may be read only, write only, or read-modify-write. Data is read from the memory array via the Dout bus and arrives at the EDC via the three way buffers. The EDC generates new check bits during the DRAM access time (200ns) and, at the end of this period, the EDC is put into correct mode in order that the freshly generated check bits may be used in conjunction with the previously stored value to, if necessary, correct any single bit error in the data. Up to this point, all backplane accesses follow the same timing. During the remainder of a

read access, the corrected data bits are presented to both the memory array (via the Din bus) and the backplane via the three way buffers. The six check bits are presented to the check RAM. The write signal, W, is then asserted (logic 0), to re-write data and check bits to the appropriate memory devices and the backplane is signalled, via the control bus (DTACK), that a valid data is available on the backplane data bus. During the remainder of a backplane write access to the memory board, following the error correction stage, partially new (byte write cycles) or wholly new (word write cycles) data is presented to the EDC for the generation of new check bits. The freshly generated check bits protect the whole of the new data word, be it wholly new, or comprised of one 'new' byte and one 'old' byte. W and DTACK are activated as in the read access to simultaneously write the data and check bits and to signal the backplane that the access is complete. In both read and write accesses, following the correction stage, the EDC is returned to the generate mode so as to supply valid check bits when necessary. Thus, all accesses to the memory board follow the same pattern, i.e. read the memory array, detect and correct errors, generate new check bits, then write back to the memory devices. The whole operation takes approximately 400 nanoseconds.

The EDC logic is comprised of an AM2960 integrated circuit and the associated data buffers are AM2961 devices (4-7). Each 16 bit data word is protected by a 6 bit check word which is generated according to a modified Hamming code. This code guarantees to detect and correct <u>all</u> single bit errors, detect <u>all</u> double bit errors, and detect many other multiple bit errors in the 22 bits comprising the data word and its check word. Not all multiple bit errors are

detectable, since many data words must share the same check word.

It was felt that EDC facilities should be provided on this board because, with the increasing density of memory devices in terms of memory storage for the same surface area, the charge storage for one bit of a data word diminishes. This increases the possibility that an alpha particle collision may corrupt data stored in memory and so increases the bit error rate and decreases the reliability of the memory board. The use of EDC can improve the reliability of the memory board by a factor of sixty (4-7). The control and status register (CSR) associated with each memory board offers a number of programmable options with regard to the handling of errors detected by the EDC. This register is located in the IOPAGE at an address governed by the base address of the memory board. Three bits are significant when this register is written to; bit 0 determines whether a detected error is reported, via the bus error exception, or not; bit 1 may be set to put the EDC into a pass through mode, so that no detection, correction, or generation of check words occur; bit 2 may be set so that only uncorrectable, multiple bit, errors are reported, otherwise all bit errors will be reported if bit 0 is set. Reading the CSR returns the value of the syndrome bits generated by the EDC during the correction step and output on the check bus. The syndrome bits are captured if any bit errors occur and either encode the identity of the single bit in error or report the occurrance of a double bit or detectable multiple bit error. If no errors have occurred since the last time the CSR was read, the CSR will read as zero.

Thus, this memory board offers full error detection and

correction facilities with programmable reporting options and transparent refresh. The cycle time of a typical access is approximately 500ns when used with an 8MHz CPU.

4.4.3 The EPROM Board

A schematic diagram of the EPROM board is shown in Fig. 4.5. The EPROM array consists of sixteen 24 pin sockets which may be configured, by the use of header plugs, to accept any of the JEDEC standard 24 pin EPROM/ROM packages. Each of these sixteen sockets supplies a byte of data to the data bus, so that the EPROM array is arranged in eight pairs of devices, each pair supplying a whole 16 bit data word. Each pair of devices is enabled by one of the chip selects, CS0-CS7, from the EPROM decode logic. This decode is switchable so that each socket may be decoded as either 1K x 8, 2K x 8, 4K x 8 or 8K x 8 devices, the latter only being available as a 24 pin read only memory (ROM) device. The access time to the devices may be varied by moving a wire link and there is a switch selectable option which reports attempted write accesses to the board via the bus error exception. A further switchable option is available which allows the first eight bytes of the EPROM board to decode at memory locations 0 to 7. This allows the reset vector to be stored in non-volatile EPROM whilst, by using a similar option on the DRAM board to disable access to RAM at addresses 0 to 7, the remaining exception vectors may be programmed by software.

Thus, the EPROM board supports a number of different EPROM/ROM devices and may be used to supply either 16K, 32K, 64K or 128K bytes of read only storage.

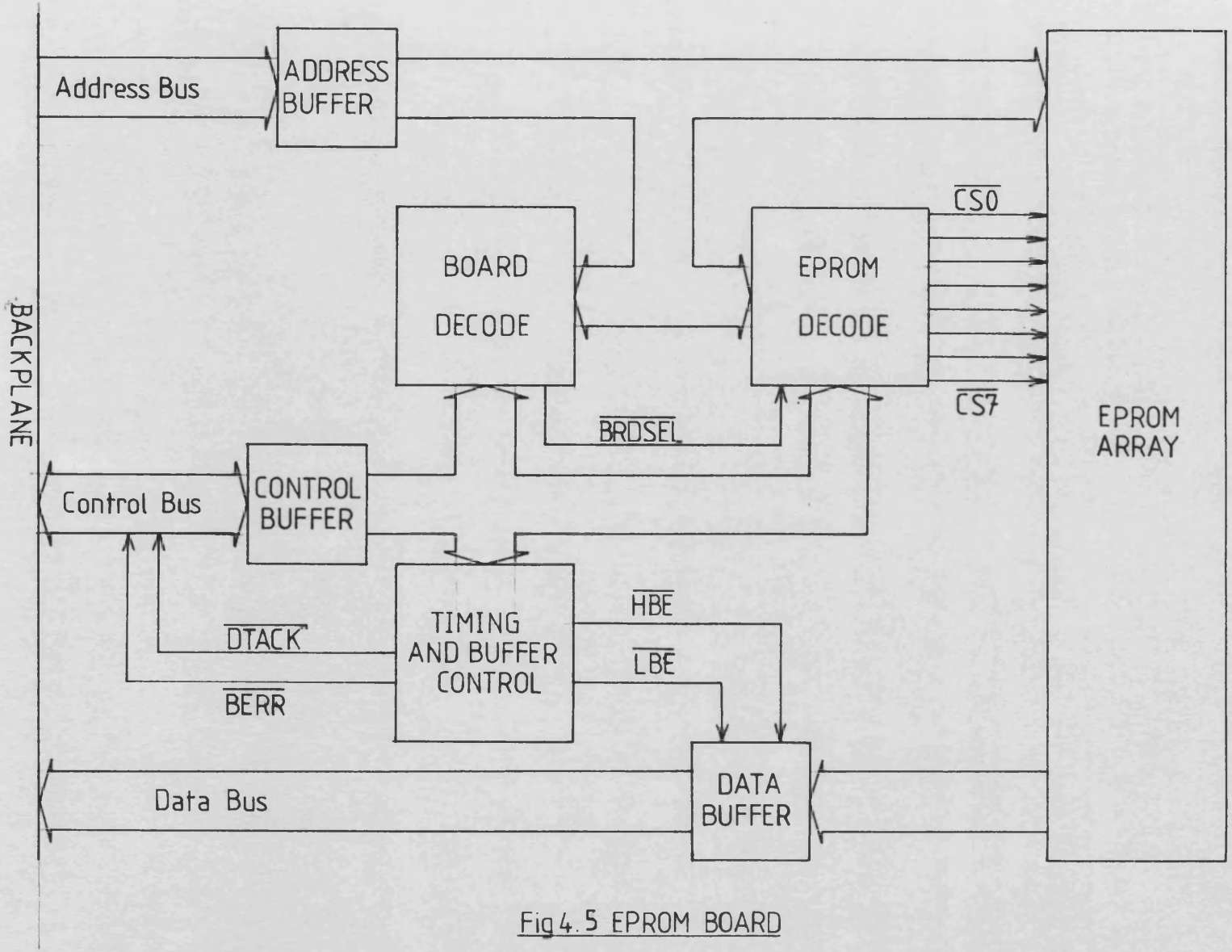


Fig 4.5 EPROM BOARD

4.4.4 The Display Driver and Peripheral Board

This board, a schematic of which is shown in Fig. 4.6, has three functions. First, it buffers all the address, control and data signals present on the backplane and uses these buffered signals to drive LEDs on the front panel of each subsystem, which indicate the state of the backplane. Secondly, this board terminates the backplane with termination networks identical to those on the CPU board. These two boards therefore belong at opposite ends of the backplane. Lastly, this board implements a number of simple peripheral and support devices which provide serial input/output and event timing.

Serial I/O is provided by two Motorola MC6850 Asynchronous Communication Interface Adaptors (ACIA), each of which (4-6) supports a full duplex, RS232-C, serial interface. Each of these devices is supplied with a switch selectable baud rate clock which enables communication at baud rates from 110 baud to 9600 baud.

Event timing is catered for by a Motorola MC6840 Programmable Timer Module (PTM). This device (4-6) contains three individual counter/timer circuits, each of which may be used for interval timing, period measurement, frequency measurement or event counting. Each counter/timer may be independently clocked by an external source or internally connected to the system E clock (0.8MHz) which is present on the control bus.

Both the ACIAs and the PTM are interfaced to the backplane using the simple MC6800 type bus interface. Both sets of devices are

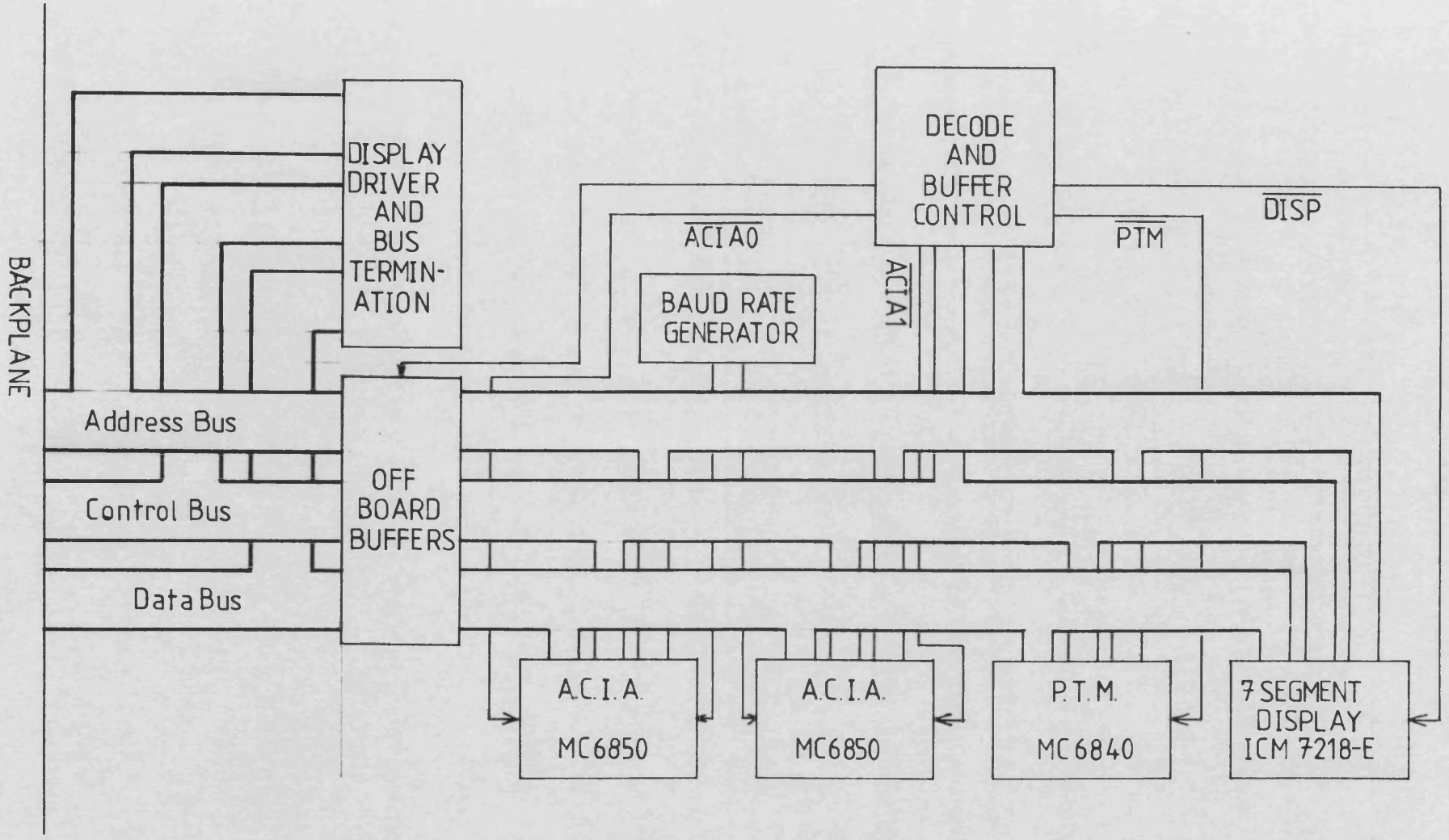


Fig 4.6 Display Driver and Peripheral Board

capable of generating interrupts to the CPU. The ACIAs use autovectored interrupt level 2 and the PTM uses autovectored interrupt level 6.

A seven segment display driver device (ICM7218-E) has also been implemented on this board. This has been configured as two four-digit, seven-segment, LED display ports on the front panel. These ports may be used to display program status information. For example, the real time simulation software uses one of these ports to display the passage of time and the other is used to indicate the occurrance of transmission errors on the communications link. The display driver provides two output formats. In hexadecimal mode, each digit displays, in hex, the value of a 4 bit binary number written to the corresponding address of the digit within the device. In the alternative mode, the display driver uses a B-code representation which gives the ten decimal digits, 0 to 9 and a selection of displayable letters.

On each subsystem front panel, in addition to the display LEDs, there are also reset and non-maskable interrupt switches. These switches are debounced and attached to the backplane by the display driver board.

4.4.5 The Floating Point Board

The floating point board was designed to improve the speed of floating point calculations. A schematic of this board is shown in Fig. 4.7. It can be seen that this board accommodates four separate AM9511 or AM9512 Floating Point Processors (FPPs) described in

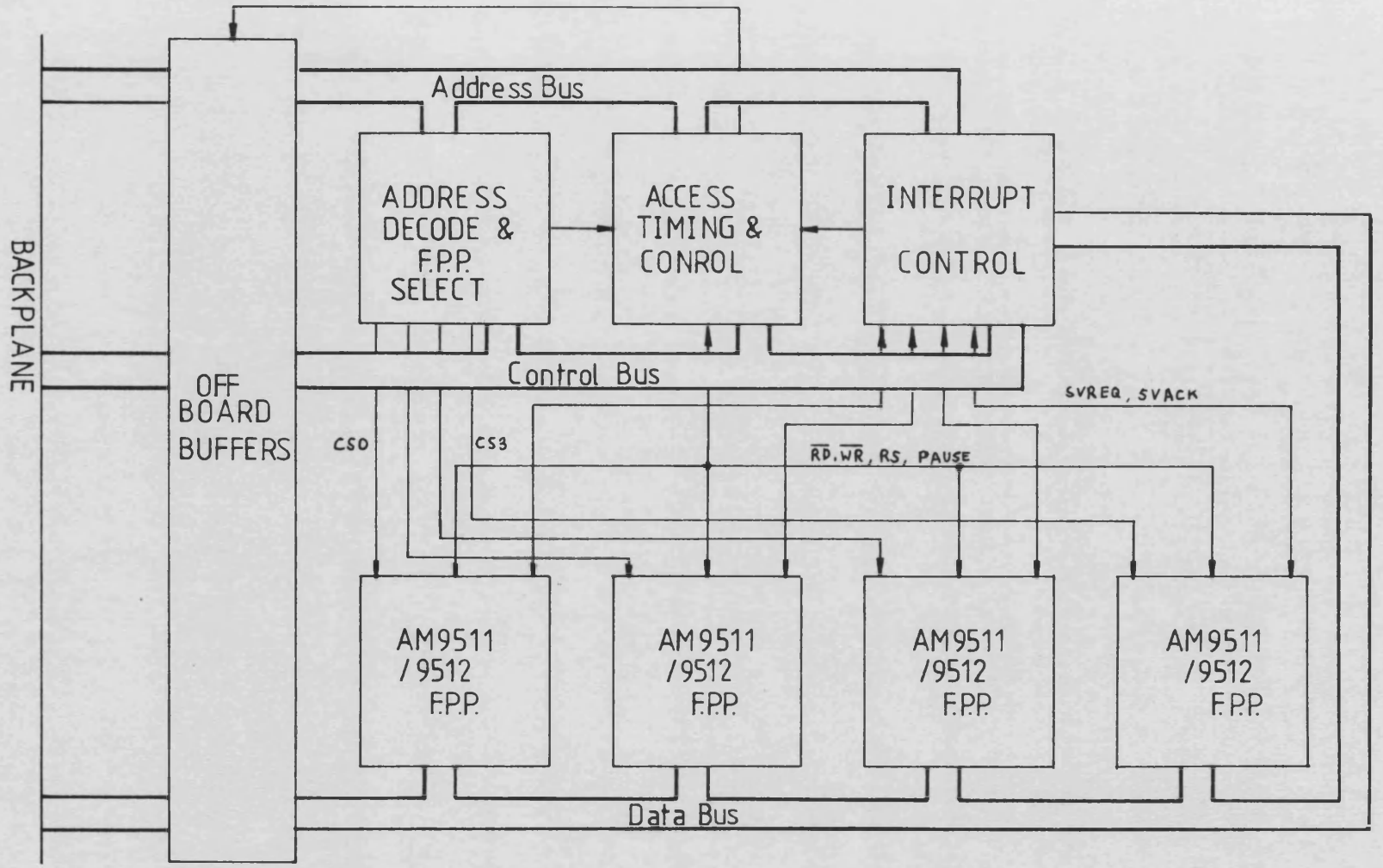


Fig 4.7 The Floating Point Processor Board

reference (4-8). When the board is accessed, the address decode determines which of the four FPPs to access and applies the appropriate chip select. The access timing and control logic generate the read, write and register select ($\overline{RD}$, $\overline{WR}$ and RS) signals, to control the direction of the exchange of data between the backplane and the FPP data register or command and status register (CSR). If the FPP is busy when it is accessed, then its PAUSE signal will become active during the access, and the access may not complete until the PAUSE signal becomes inactive. Each FPP handshakes with the interrupt controlling logic using the service request (SVREQ) and service acknowledge (SVACK) signals. The interrupt logic prioritises the individual FPP interrupts and passes pending interrupts through to the CPU. Associated with each FPP is a separate switch selectable interrupt vector which will be published on the data bus during an interrupt acknowledge cycle. All four FPPs may be active simultaneously and, once the CPU has initialised an FPP operation, it may also operate in parallel with the FPPs until the result of a floating point operating is required. Floating point results may be recovered from the FPP when it interrrupts or, since interrupting and polling versions of each FPP command are available, they may be recovered from the FPP data register when required, having ensured completion by reading the CSR.

The AM9511 device uses the floating point representation of Fig. 3.12 and also supports 16 and 32 bit twos complement integer arithmetic. In addition to floating point add, subtract, multiply and divide operations, the AM9511 supports the trigonometrical functions sine, cosine, tangent, arc-sine, arc-cosine and arc-tangent, together with the commonly used functions, square root,

common logarithm, natural logarithm, exponential and power functions. The AM9512 is much more humble in its capabilities. However, it does support the IEEE 32 bit single precision and 64 bit double precision floating point representations (4-9). This device supports floating point add, subtract, multiply and divide operations and both devices support the conversion of values between integer and floating point representations.

The provision of four FPP units offer a simple extension to the arithmetic capabilities of each subsystem and also increases the level of parallel operation capable when floating point calculations are performed.

4.4.6 The Communication Link

The communication link hardware was itself the subject of other research (4-10) and only a description of its operation as seen by the programmer will be given here.

A specification for the communication link data rate may be arrived at by estimating the amount of data that must be exchanged during each integration step and by noting that this exchange must occur within the time taken to perform an integration step. If the data exchange takes longer than one integration step period to complete, new data will be generated before the previous data transmission has completed. If a simulation is considered where 1000 data items, each represented by a 32 bit word, must be exchanged during each integration step and the integration step length is 5 milliseconds, then the minimum acceptable transmission rate for

real-time operation is one byte every 1.25 microseconds, or 0.8 bytes per microsecond. At this minimum data rate, data generated during one integration step is in transit during a second integration step and is not available for use by the receiving subsystem until the beginning of the third integration step. Clearly, a much higher data rate is necessary if subsystems are to complete their data exchange within the integration step in which it is calculated. Ignoring any parity and start/stop bits or framing characters associated with the transmission protocol, 0.8 bytes per microsecond requires a bit rate of 6400000 bits per second. This order of bit rate is too high for serial protocols such as HDLC (4-11), X25 (4-12) or 1553 (4-13). Possible contenders using standard specifications were Ethernet (4-11) and the Cambridge ring (4-15), each of which has a raw data rate of 10 million bits per second. Unfortunately, the transmission efficiency of both these systems is low for short messages and the effective data rate falls below the minimum rate given above. In addition to the purely hardware considerations regarding bit rates, higher levels of protocol are necessary in the system software to check the validity of each transmission in order that the possibility of transmission failure is minimised. It was concluded that none of the standard communication specifications could meet the requirements for communication between the processing subsystems of the real-time simulator, so a dedicated communications system was designed and built (4-10).

The communication link uses a byte wide parallel communication bus (Fig. 4.1(c)). A byte is transferred every 125 nanoseconds, giving a raw data rate of 8 Mbytes per second, which corresponds to a bit rate far in excess of the minimum requirement. The system

operates on a time slot mechanism, where each subsystem in turn is given the opportunity to transmit one of the outstanding messages. Should the communication controller detect no activity on the communication bus, it passes the transmission opportunity on to another subsystem. Up to 15 processing subsystems may be attached to this system and the transmission overhead amounts to two bytes per message, one leading byte to identify the message and one trailing byte which contains a checksum. The checksum is generated and checked by the link hardware. Should an error be detected, no message is passed to the receiving subsystem, and the sending subsystem is informed of the transmission failure via an interrupt. When none of the subsystems have any outstanding messages for transmission, the communication controller continuously cycles through each subsystem offering the opportunity to transmit. The destination of individual messages is independent of the time slot allocated to a processing subsystem. Each message is preceded by an identifying byte which is inserted and removed by the hardware. This identifier may be in the range 0 to 254, a value of 255 indicates to the controller that no message is to be transmitted and that it should pass on the transmission opportunity to the next subsystem. Thus, there are 255 possible message numbers and it is entirely up to the programmer to specify the content and size of each message.

To the programmer, the communication link has three functional areas. These three areas are 2Kbytes of high speed shared memory ($200000 to $2007FF) used for assembling messages, 2K bytes of 'slow' memory ($201000 to $2017FF) partitioned as two tables of message descriptors, and 4 bytes of registers ($201800 to $201803). The fast memory is divided into arbitrary length contiguous areas of memory by

the message descriptors held in the slow memory. For transmission, a message is copied into the appropriate area of the shared memory and, on reception, the received message may be copied out of the appropriate area of the shared memory. The slow memory is partitioned into two tables of message descriptors, one table for message reception ($201000 to $2013FF) and one table for message transmission ($201400 to $2017FF). Each entry in the descriptor tables is four bytes long, as shown in Fig. 4.8, and each table is indexed according to the message number, so that, for example, $201408 is the address of the transmission descriptor for message 2.

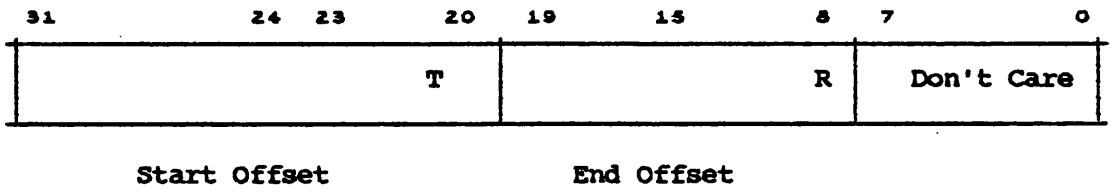


Fig. 4.8 Message Descriptor

The bottom byte of a descriptor is ignored and reads back as $FF, bits 31 to 20 give the starting byte offset of the message area in the fast memory and bits 19 to 8 give the end offset of the message. Note that both these offsets are even, since the least significant bit of each descriptor is either a transmit permit (bit 20) or a receive permit (bit 8) flag which indicates the validity of the descriptor. Thus, a descriptor value of $48150100 loaded into address $201004 enables reception of message number 1 with the message area reserved in fast memory, from $200480 to $2004FF inclusive. Both the receive and transmit permit bits have been set in this descriptor. However, the receive permit bit is ignored by the transmission table and the transmit permit bit is ignored by the reception table. When loading

the descriptor tables, no checks are made by the hardware and it is the programmer's responsibility to manage allocation of fast memory to messages.

The link control and status register (CSR) is a byte wide register located at address $201801. The significance of the CSR bits is illustrated in Fig. 4.9.

Read								Write							
E1	E0	RF	RE	TF	TE	I	X	X	X	X	X	CQ	SL	CL	IE
7							0	7							0

X Don't Care

E1, E0	Error Code	CQ	Clear Queues
RF	Receive Queue Full	SL	Set Lockout
RE	Receive Queue Empty	CL	Clear Lockout
TF	Transmit Queue Full	IE	Interrupt Enable
TE	Transmit Queue Empty		
I	Lockout Indicator		

Error Codes	E1	E0	Comment
0	0	0	Successful Transmission
1	0	1	Reception Complete
2	1	0	Transmission Error
3	1	1	Illegal Transmission

Fig. 4.9 Control Status Register

The communication link is initialised by setting the lockout bit in the CSR (SL) or by applying the link reset on the link controller. The descriptor table may then be flushed by writing zeros to all

locations and the appropriate descriptors then loaded. The transmit and receive queues (byte registers at $201800 and $201802 respectively) are then cleared by setting the CQ bit in the CSR. Finally, the interrupt may be enabled and the lockout bit cleared to enable the communication link to access the communication bus.

A message is transmitted by assembling the message data to the appropriate area in the shared memory. The corresponding message number is then written to the transmit queue (write only register at $201800) and the communication link will transmit the message in a future time slot. When the transmission complete, the link hardware enters the message number into the receive queue with the appropriate status bits set to indicate the success or failure of the transmission. If the interrupt enable bit has been set, then the communication link will interrupt the CPU for as long as entries remain in the receive queue (read only register at $201802).

When the CPU services an interrupt from the communication link, it may simultaneously read the message status from the CSR and the message number responsible for the interrupt by performing a word read from address $201802, since the CSR is reflected at $201803. Reading the receive queue will unload the entry causing the interrupt from the queue and will update the error code bits in the CSR to correspond to the next entry, if any, on the receive queue. Each interrupt from the communication link may then be handled on the basis of its message number and the error code returned in the status register.

Following the reception of a permitted message, the corresponding

receive permit bit will be cleared and an appropriate entry placed in the interrupt queue. In order that a further message of the same number may be received the corresponding permit bit must be set to indicate that the previous message, using that number has been handled by the CPU.

This communications link offers a fast and flexible mechanism for transporting information around the multiprocessor system. Since messages are identified by message number and not by reference to a particular subsystem, application software may be written so that it receives the expected messages no matter which particular subsystem it resides in. The messages require simple handling, i.e. copying, and protocol software may be kept to a minimum when using the simple type of messages employed in the real-time model.

4.4.7 The Floppy Disc Controller Board

A floppy disc controller (FDC) board has been designed (4-17) so that a 'mass' storage medium is available to subsystems equipped with this board. Fig. 4.10 shows a schematic diagram of the FDC board which is designed around the Western Digital FD1793-02 Floppy Disc Formatter/Controller (4-16) integrated circuit and its associated support devices. This 'chip set' is capable of supporting both double and single sided disc drives using both double and single density recording formats. The FD1793 supports a wide range of controller functions for establishing and verifying the position of the read/write head, reading and writing of single and multiple sectors for data storage and retrieval and reading and writing of entire tracks for diagnostic and formatting purposes. Command

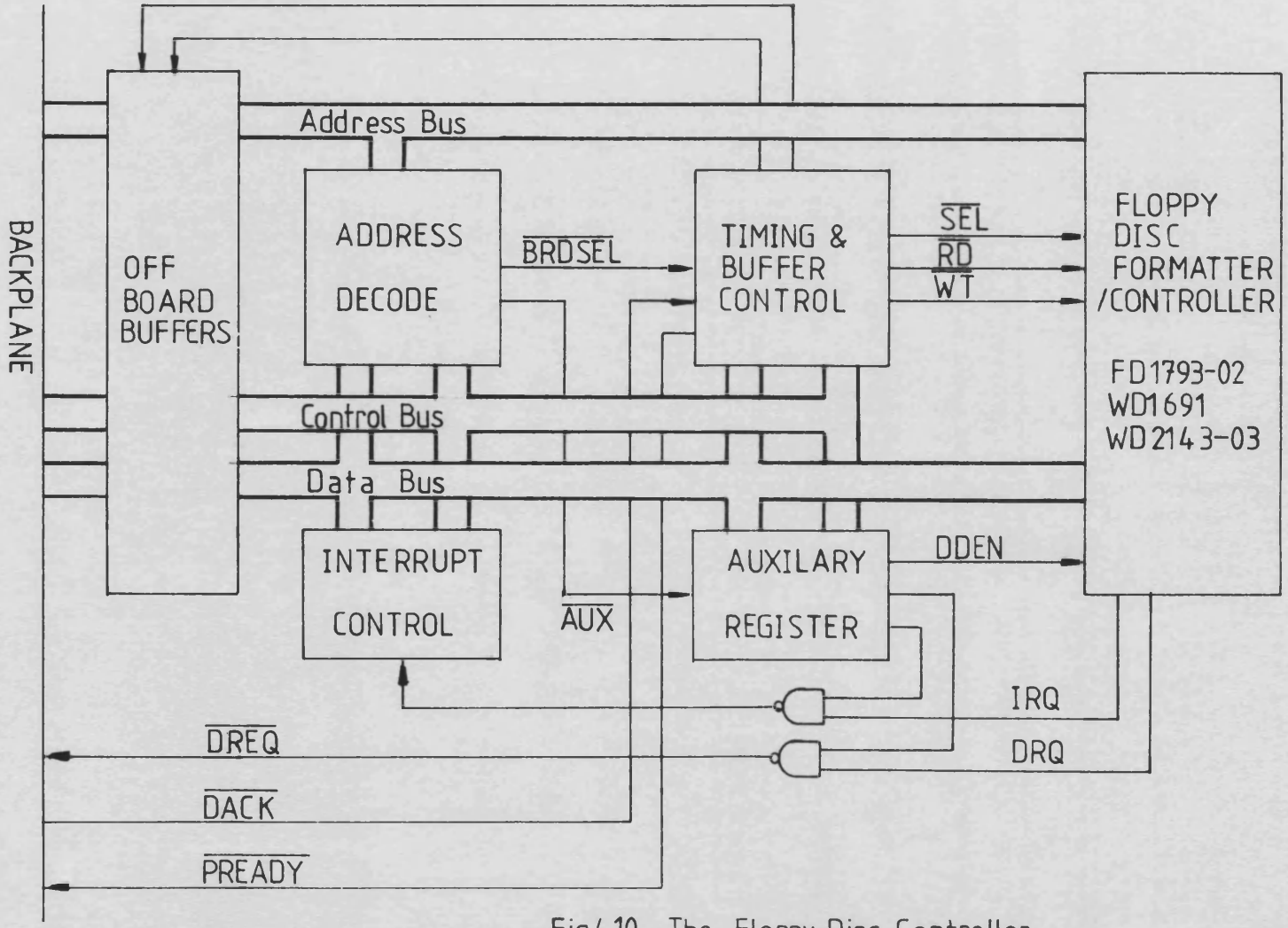


Fig 4.10 The Floppy Disc Controller

completion may be detected by polling the command/status register (CSR) of the FD1793 (a byte wide register at $840101) or by enabling the device interrupt request, IRQ, to reach the interrupt control logic by setting bit 1 of the auxiliary register (a byte wide register at $84010D). Enabling the device interrupt allows the FDC board to inform the CPU that it has completed a command by propagating an interrupt request to the interrupt control logic which, in turn, supplies the CPU with an 8 bit switch selectable interrupt vector.

Data transfer may also be handled by polling the CSR or, alternatively, by setting bit 2 of the auxiliary register, the device data request signal, DRQ, may be propagated to a DMAC via the DREQ signal. During a DMA access on behalf of the FDC, the FDC timing and control generation is altered by the DACK signal from the DMAC which identifies the DMA cycle. During a DMA cycle, the address bus references a memory device, so the direction of transfer from the point of view of the FDC, must be opposite to that normally associated with the particular sense of the read/write signal of the control bus. Also, during a DMA access, the FDC may not signal the end of a cycle by using the DTACK signal since this will be used by the memory device for the same purpose. Thus, an extra control signal, PREADY, is necessary so that the FDC may inform the DMAC that it has completed its part of the DMA transfer. This DMA interface allows data transfer between the FDC and memory, without processor intervention, leaving the CPU free to perform other tasks.

When used in conjunction with the operating system described in a subsequent section, a double density format is used on both sides of the disc. A total of 1232 sectors, each storing 1Kbytes of data are

formatted onto each disc, giving each disc drive a formatted storage capacity of 1.232 Mbytes. Up to four disc drives may be driven by the board offering a total storage capacity approaching 5Mbytes.

4.4.8 The Direct Memory Access Controller

Direct memory access was considered a desirable property for peripherals such as disc controllers to possess. Two alternative methods exist for implementing such facilities exist. Either each peripheral requiring direct memory access is fitted with address generation, bus cycle generation and bus acquisition logic, or a board is constructed which services the DMA requirements of a number of peripherals. The latter of these two options allows much of the logic required to furnish DMA facilities to be shared between a number of peripheral boards and so make their design less complicated. Fig. 4.11 shows a simplified schematic diagram of the Direct Memory Access Controller Board. In order to avoid cluttering, many of the connections between functional blocks have been omitted. For example, the address decode logic supplies device selects to the two AM9517 devices (4-8) and the 8x8 register files. The most significant omission from Fig. 4.11 is the mechanism by which the register files are addressed which will be described later in this section. At the time this board was designed, a specification existed for the HD68450/MC68440 type DMAC incorporated on the later CPU boards. However, production time scales made it necessary to consider designing this board as an interim measure. None of the available DMAC integrated circuits directly supported a 24 bit address bus and a 16 bit data bus and most were designed to interface to the synchronous buses associated with most of the popular 8 bit

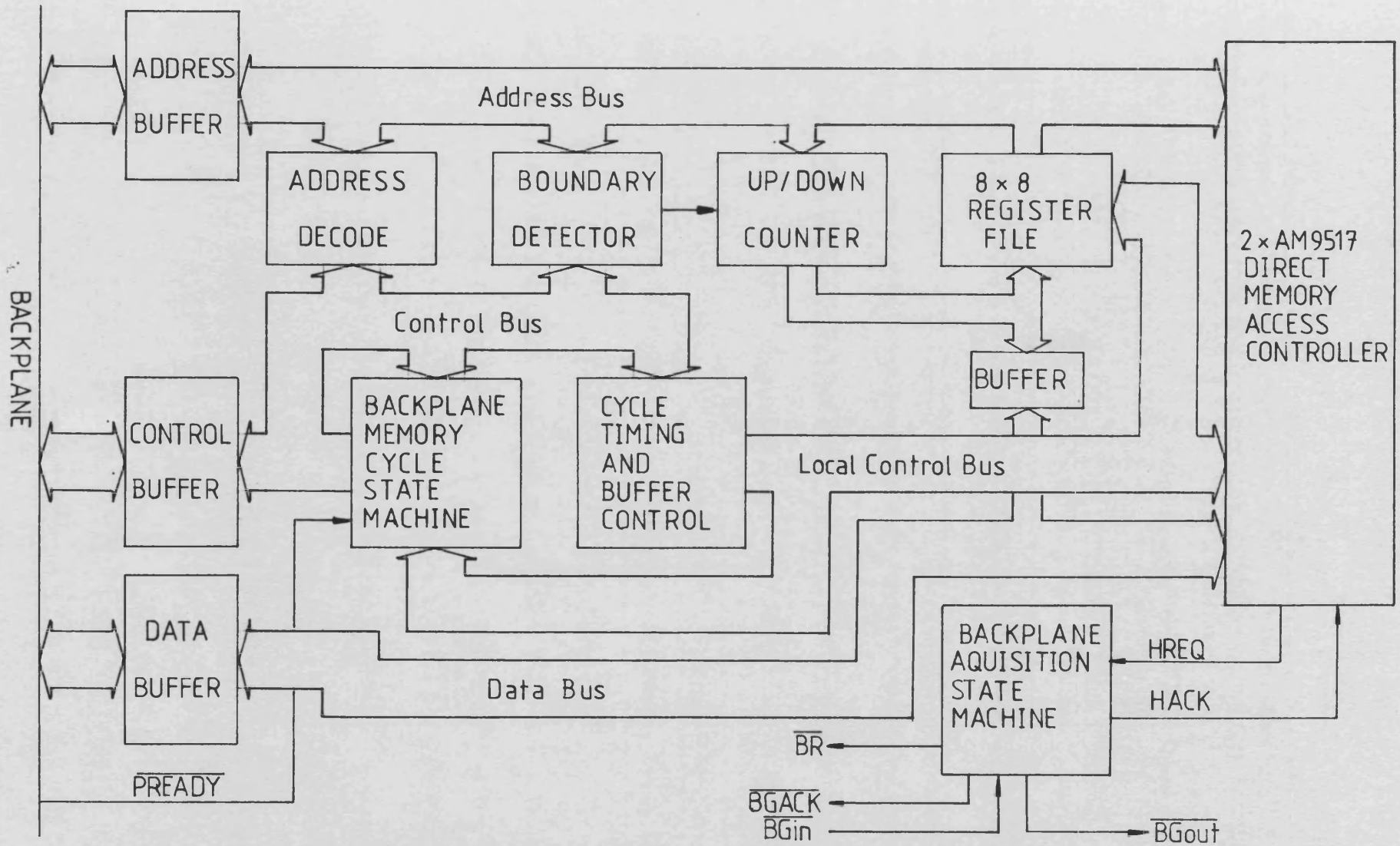


Fig 4.11 The Direct Memory Access Controller

microprocessors, the Z80 or I8080 family devices. Of these devices the AM9517 offered the most powerful facilities (4-8). The AM9517 supports 4 independent channels of 8 bit DMA in a 64Kbyte address space, with data transferred either in bursts, on demand, or by cycle stealing.

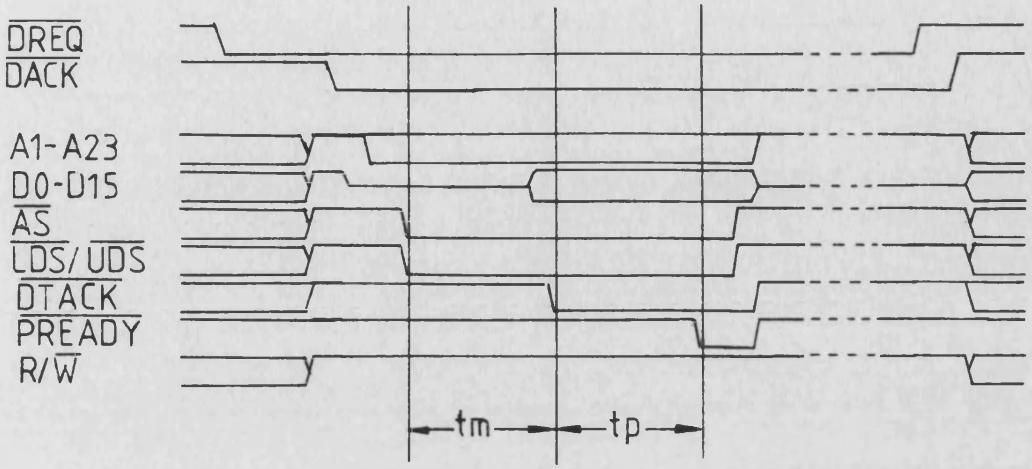
This board only supports byte wide DMA and this was considered acceptable since most peripherals requiring DMA facilities are byte orientated. Thus, the major function of this board is to increase the width of the address bus associated with the AM9517 to 24 bits in order that the full 68000 address space may be accessed. During DMA cycles, the top 8 address bits are supplied by the register files, while the lower 16 address bits are supplied by the AM9517. The address used to index these files during DMA is encoded from the device acknowledge (DACK) signals output by the AM9517 to indicate to the requesting devices that the backplane has been acquired on their behalf. Each AM9517 supports four DREQ/DACK handshake pairs which interface directly to the peripheral associated with each DMA channel. One of these handshake pairs is reserved for internal use so, with two AM9517 devices on the board, it can support DMA facilities for 7 DMA peripherals. During backplane accesses which initialise the register files, the register files are indexed from the address bus in the usual way.

Two state machines have been incorporated into the design of this board. One converts the simple request/acknowledge (HREQ/HACK) bus acquisition protocol of I8080 type devices, to the three line request/grant/grant acknowlege (BR/BG/BGACK) protocol of the MC68000 and supports daisy chaining of the bus grant if multiple DMAC boards

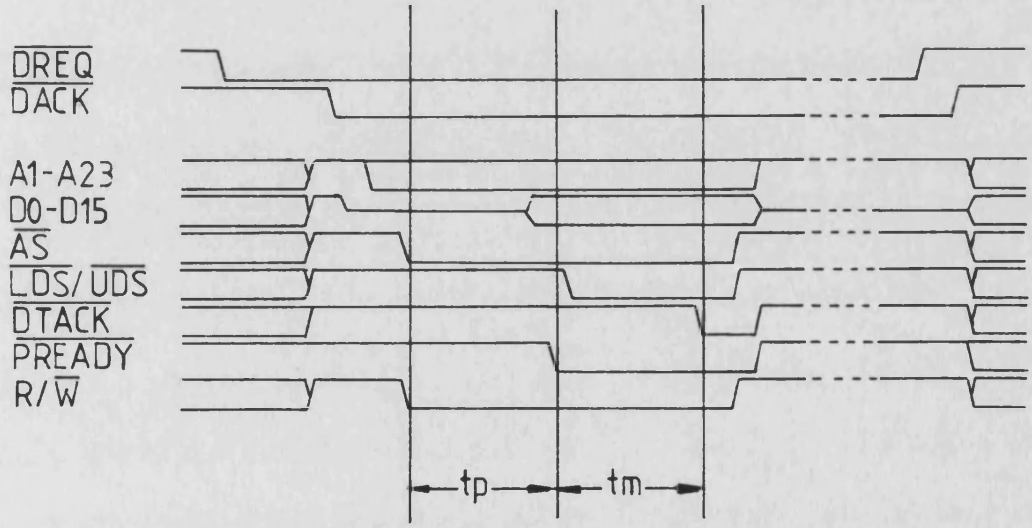
are required. The other state machine generates MC68000 backplane control signals from the control signals output by the AM9517. This state machine is responsible for applying the 68000 type control signals in the correct order and waiting for the DTACK and PREADY responses from the memory and the peripheral respectively.

In order that the 24 bit address generated by the combination of 8 bits from a register file and 16 bits from the AM9517 is capable of crossing a 64Kbyte boundary as a sequence of DMA cycles proceed, it is necessary to modify the value held in the register file when boundary crossing occurs. This is accomplished as follows. At the start of every DMA cycle, the top 8 bits of the address, supplied by the register files, are loaded into an 8 bit up/down counter and, shortly after the end of every DMA cycle, the value at the output of the counter is rewritten into the register file. Should the boundary detection logic detect that the address of the current cycle is the last address of the current 64K page, i.e. $XXFFFF during address increment or $XX0000 during address decrement, the up/down counter is clocked in the appropriate direction prior to the rewrite to the register file.

The handshaking protocol between peripherals and the DMAC was introduced in the discussion of the floppy disc controller board. Fig. 4.12 shows clearly the sequence of events during DMA read and write cycles to memory. Note that, during the DMA read cycle, a peripheral must observe the assertion of DTACK by the memory board at the end of the memory access time, t_m, before commencing the access to the peripheral device with access time, t_p. This ensures that valid data, read from the memory is written to the peripheral. Similarly,



DMA Read Cycle



DMA Write Cycle

Fig 4.12 DMAC Generated Backplane Cycles

during a DMA write cycle, the peripheral device must present data to the data bus (both halves since neither of the data strobes is active to identify whether a high byte or low byte data transfer is taking place) during the device access period, t_p. Once the peripheral has signalled the validity of the data by asserting PREADY, the DMAC asserts the appropriate data strobe to activate the memory board timing. The cycle completes with the assertion of DTACK. The broken lines in the timing diagram indicate that further DMA cycles may occur before the DMAC relinquishes the backplane, depending on the transfer mode.

The DMAC board provides byte orientated direct memory access for up to seven peripheral devices. It is primarily intended to permit the performance improvement available when used with disc devices, or other devices which require large volumes of data movement at hardware determined intervals.

4.4.9 The High Resolution Graphics Board

A high resolution graphics board was designed around the Thomson-Efcis EF9365/EF9366 Graphic Display Processor (GDP) (4-18). This board supports screen resolutions of 512x512 pixcels using an interlaced scan or 512x256 pixels using a non-interlaced scan. The pixel memory consists of two pages of three planes each, which permit each pixel to take on one of 8 simultaneously displayable colours. Two pages of memory assist animated graphics since erase and draw operations may be carried out on the background page whilst the foreground page is displayed and then the two pages may be swapped over. The eight displayable colours may be chosen from a palette of

4096 colours and shades made available via a colour lookup table attached to the pixel outputs. Light pen facilities are also available on this board to ease user interaction with the system.

The GDP itself is capable of plotting lines and pixels directly into the pixel planes at a maximum rate of 1,500,000 pixels per second, or an average rate of 900,000 pixels per second. The device also supports the display of characters using an on-chip ASCII character generator and generates all the synchronising and blanking pulses necessary to interface to a CRT display.

GDC commands include vector drawing, character drawing, screen clear, light pen position requests and direct access to the pixcel memory. After completion of each command, the GDC may be programmed to interrupt the CPU so that once a command has been initiated the CPU is free to perform other tasks until the GDC interrupts.

The graphics board has been used for displaying data collected from the real-time simulation in graphical form. This permits a very fast assessment of the transient behaviour of the power system to be made and is very useful for observing the progression of an optimisation study such as those described in Chapter 6.

The hardware configuration used for the studies presented in this thesis is comprised of two basic subsystems and one subsystem supporting a disc based operating system and high resolution graphics.

4.5 Simulator Software

The software associated with the simulator may be considered in three distinct groups: the system software, the application software and the support software. In the sections which follow, consideration will be given to the ROM based operating system, or monitor, used in the basic subsystem racks, the disc based operating system used with the expanded subsystem rack, the simulation and monitoring software which combine to form the real-time simulation and its user interface, and the debugging, loading and support software which ease the development and use of the real-time simulator.

4.5.1 System Software

The function of system software is to provide an environment for the development and execution of application software. The objective of this environment is to provide standard mechanisms for accessing peripheral devices, allocating memory, passing messages between tasks and scheduling tasks in a multi-tasking environment.

The system software used on the basic subsystems is an enhanced version of the Macsbug monitor by Motorola (4-19). This monitor provides the usual facilities expected from a monitor, such as memory and CPU register inspection and modification, program loading, execution and single stepping for debugging purposes. In its original form, this monitor supported only two serial ports, which were polled rather than interrupt driven. The facilities offered by this monitor were enhanced by the development of an interrupt driven

I/O subsystem (4-4). This subsystem gives the application programmer a standard interface for using the two serial ports, each of the three timers associated with the PTM, the floating point processor board and the graphics board. Provision of this standard interface removes the necessity for the application programmer to have explicit knowledge of how to program each of the various peripheral devices. In order that the I/O subsystem may be used in a protected or memory managed environment, it is entered by the use of a TRAP instruction rather than a subroutine call. Although very similar in function in a non-protected system, the use of a trap will force the MC68000 into supervisor mode and resume program execution (exception processing) at an address indicated by the supervisor controlled exception vector table. Since, in a protected system, programs executing in user mode may be denied direct access to the exception vector table and the I/O devices themselves, the use of the TRAP instruction provides a secure mechanism of controlling the use of I/O devices.

TRAP 12 is used to gain access to the I/O subsystem. A single parameter is passed to the I/O subsystem in address register A6. This parameter is a pointer to a 'packet' which describes the particular I/O subsystem function requested and supplies any further parameters necessary to fulfil the user's request. The structure of all I/O packets is shown in Fig. 4.13.

To request the execution of a particular I/O function, the address of the first entry in the packet is loaded into A6 and a TRAP instruction executed. The I/O subsystem comprises primitives for queueing, dequeueing, device initialisation, interrupt handling and I/O completion. On entry to the I/O subsystem, the packet function

code is checked to determine whether a valid function has been requested.

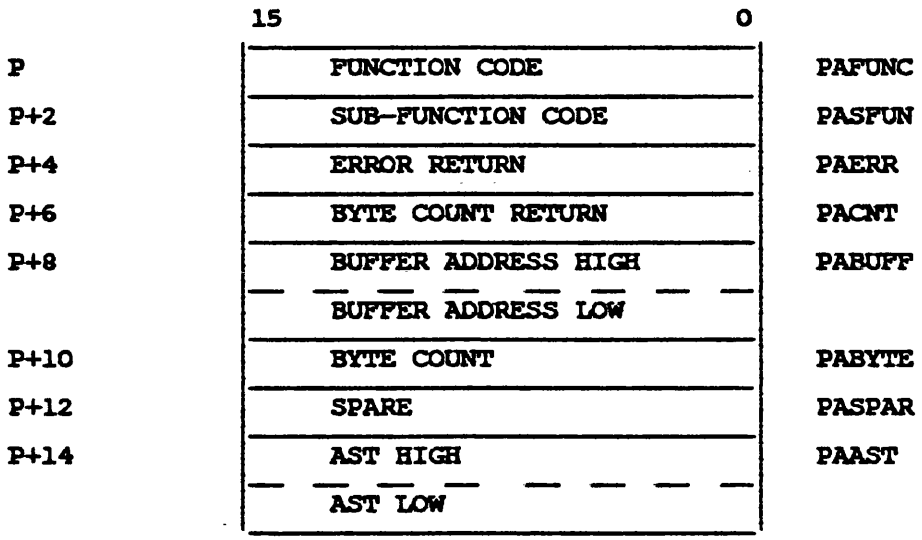


Fig. 4.13 I/O Packet Structure

The function code primarily determines which device the I/O packet activates and is used to select the appropriate queue on which to place the packet. This queueing action is the only action that will always occur synchronously with the task which made the I/O request. If the function code given is invalid, or the appropriate queue is already full, the I/O subsystem will return to the user task, following the trap instruction, with the carry bit set in the MC68000 status register. In all cases where errors are detected, be it here during the queueing operation or at some intermediate point in the I/O operation, an appropriate error code is returned in the packet. If the packet is successfully queued, then the device initialisation routine will be called. This routine attempts to dequeue the packet at the front of the queue and place it on the device's active list. If the device is already active, servicing another I/O request packet, this action will fail and be retried later when another packet is queued to the same queue or when the currently active I/O

request is completed. After the device initialisation routine has returned, the I/O subsystem will pass control back to the calling task by returning with the carry bit cleared. The device initialisation routine is also responsible for checking the sub-function code which determines the particular action required of the device identified by the function code. If the sub-function code is not supported by the device, then when the packet reaches the front of the queue, the I/O goes immediately to completion and an appropriate error code is placed in the packet. Once an I/O operation has been successfully initialised, the I/O sub-system may only be re-entered on behalf of that particular I/O operation by interrupts from the device concerned with the operation. The interrupt routine is responsible for all the intermediate handling of the I/O such as filling or emptying buffers and adjusting byte count values in the packet. When the end of the I/O operation is detected within the interrupt routine, either due to an exhausted buffer or some device specific circumstance, the I/O completion routine is called. The completion routine is responsible for placing the return code in the packet and removing the packet from the active list. At this point, if the address of an asynchronous service trap (AST) routine has been given in the last two words of the packet, this routine will be called as a subroutine of the I/O completion routine. This AST routine is optional (the AST field of the packet is set to zero if it is not required) and may be used to inform the task of return of its packet, or to perform some action that may not be performed prior to the return of the packet. In general, it is not desirable for a task to wait for the completion of an I/O request, which it may do by monitoring the packet error code field, and the AST is a task specific routine which handles the result of the I/O operation. Once the AST

returns, then the I/O completion routine will recall the device initialisation routine to commence processing of the next packet, if any, on the device queue.

All registers are preserved on entry and restored on exit from the I/O subsystem. The only critical areas of operation occur during queue manipulation. Since a queueing operation can be interrupted by a device interrupt associated with the same queue and this interrupt may cause a dequeueing action, then the consistency of the pointers handling the queue must be guaranteed by masking interrupts during queueing operations. Thus, the I/O subsystem is completely re-entrant, whether entered via a TRAP instruction or an interrupt request, or a further TRAP instruction called as part of an AST routine.

Devices supported by the I/O subsystem are the serial ports, the timers, the floating point processors and the high resolution graphics board.

The terminal driver supports full and half duplex modes of operation and each serial port is configured as a single bidirectional device. Each serial port has a separate function code, although the rest of the packet is interpreted in the same manner for both devices. The sub-function code is used to determine which direction of transfer is required. The buffer address (Fig. 4.13) points to the input or output buffer while the byte count (PABYTE) indicates the buffer length in bytes, i.e. the number of characters to be output or the maximum size of the input buffer. The byte count return (PACNT) contains the actual number of characters read during

input requests since input operations may be terminated by either a carriage return, an escape or a control Z character. Input lines may be edited prior to termination by use of the delete character to erace single characters, or control U to delete the entire line. Control R may be used to redisplay the current input line as held by the I/O subsystem prior to line termination. The standard XON/XOFF protocol is also supported by the driver.

The floating point processors are handled as one device capable of four simultaneous activities. I/O packets are routed to the first available processor and the sub-function code is the value of the command byte required by the FPP to perform the particular operation. The buffer address of the packet should point at a block of three fields reserved for the lefthand and righthand operands and the result of the operation. Where operations require only one operand, the lefthand operand is ignored by the FPP, although a dummy value must be present to maintain the structure of the three field block. The size of the three fields is given in bytes in the packet byte count, i.e. 1 for byte wide fields, 2 for word wide fields and 4 for longword fields. It is the responsibility of the programmer to ensure that the field size and the operation requested by the sub-function code are consistent. The error codes returned in the packet reflect the status of the AM9511/AM9512 floating point devices when the arithmetic operation completes.

The three timers are considered to be one device capable of three simultaneous operations. Timing operations are therefore identified by a single function code and the individual timers are identified by the sub-function code. Timing may either be continuous or single

shot, the distinction being made by the sign of the sub-function code. In continuous mode, an AST routine is mandatory since the I/O never completes and the packet remains busy (Error code = 0). This AST is called each time the timer interrupts and so must perform whatever tasks are necessary to inform the application software of the interrrupt. The timer delay is passed in the packet byte count (PABYTE), the remaining fields being redundant.

A comprehensive set of I/O requests are provided for handling the high resolution graphics board. This board supports a variety of functions and it was felt necessary to provide a user interface to this device via the I/O subsystem. Sub-functions are defined which manage a textual scrolling region so that the graphic display may be used as a terminal device; erase the screen to specified background colours; plot polylines, i.e. lists of continuous X-Y vectors; plot defined characters, i.e. lists of disjoint X-Y vectors; write ASCII characters directly to the display (rather than the scrolling region) in any position, orientation size and colour; and which obtain the light pen position. The bulk of the graphics driver is table driven and many of the sub-functions share sub-routines, such as those for placing characters on the screen and those for drawing straight lines.

A series of number conversion functions have also been implemented to assist in the parsing of lines input through the I/O subsystem, or in the formatting of lines for output. The functions are called using a TRAP 13 instruction with A6 pointing to a packet which describes the conversion required. Since no interrupting devices are involved in the execution of these functions, all

conversions occur synchronously with the user task which calls them and are completed when exception processing completes and normal processing resumes at the instruction following the trap instruction. These functions support the conversion of decimal, octal and hexadecimal ASCII strings into twos complement integer formats as bytes, words or long words and the conversion of decimal floating point ASCII strings into a binary floating point representation. Functions are also available to perform the converse operations and to display hexadecimal values on the seven segment port LEDs on the front panel.

Thus, the monitor environment furnished by the Macsbug monitor has been extended to provide efficient, interrupt driven support of the serial ports, the timers, the FPPs and the high resolution graphics board.

The environment used with disc based hardware subsystems is the Tripos portable operating system (4-20 to 4-23). This operating system was originally developed at the University of Cambridge Computing Science Laboratory. Most of the operating system is written in BCPL (4-24) an untyped systems programming language which Tripos was designed to support most readily. Many of the operating system primitives, such as those for storage management, the creation and deletion of tasks and devices and the task scheduler, are written in assembly language. Tripos is a multitasking operating system which is intended to support a single user and no protection is offered to prevent one task interfering with another task since no memory management facilities are implemented.

A wide range of utilities and programming tools are available. Compilers are available for a number of languages other than BCPL, for example, FORTRAN, ALGOL 68C and PASCAL. A selection of cross assemblers exist to support other microprocessors than the MC68000.

The main utilities used for this project were text editors, the BCPL compiler and the MC68000 Macro Assembler, the use of which are detailed in the Tripos User Guide (4–20) and the Tripos Programmers Guide (4–21).

Tripos is a message passing operating system and all scheduling occurs as a result of the transfer of a packet between two tasks or a task and a device driver. All tasks have a unique priority and the highest priority runnable task is always scheduled. Exchange of messages between tasks and devices is accomplished using the kernel primitive 'qpkt', which queues a packet of the form shown in Fig. 4.14.

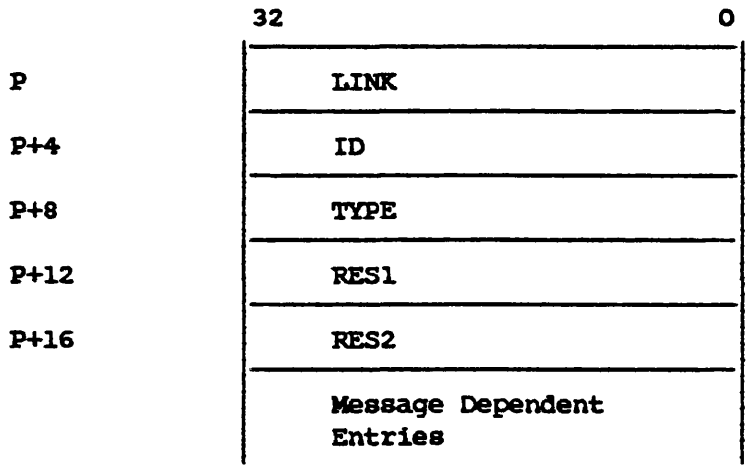


Fig. 4.14 Tripos Packet Structure

The ID field of the packet is used to identify the destination of

a packet. Tasks always have a positive ID value which is the task number. The task number is used to index the task table in order to locate the task control block (TCB). Likewise device driver ID values are all negative and their absolute value, the device number, is used to index the device table to locate the appropriate device control block (DCB). Associated with each control block is a work queue which is a linked list of packets which require the attention of the particular device or task. The packets are linked together by pointing the link field of one packet to the link field of the next. The link field of the last packet on a queue is zero and the link field of a packet which has been removed from a queue is marked 'not in use' (-1). Once 'qpkt' has located the queue identified by the ID field of the packet, it zeros the link field and points the link field of the packet on the end of the queue at the link field of the new packet. The 'qpkt' primitive also changes the ID field so that, once it arrives on the receiving queue, it identifies the queue of the sending device or task. In this way, a packet may be returned to its originator by a further call to 'qpkt'. The type field of the packet is analogous to the sub-function code used with the previously described I/O sub-system and indicates the specific action the receiving device or task is to take and how the message dependent parts of the packet are to be interpreted. The two result fields RES1 and RES2 return information to the packet originator regarding the success or failure of any actions requested by the transfer. The result RES1 is generally zero if the requested action was successful. Thus, communication between all tasks and device drivers is accomplished in a coherent fashion by the use of the 'qpkt' primitive.

Device drivers are also written in assembly language. Each driver contains five components. An INIT routine is required which is used to initialise the device when the device driver is loaded and entered into the device table. The INIT routine must ensure that the physical device controlled by the driver is initialised and ready to accept work arriving in subsequent packets. The INIT routine also initialises many of the entries in the device control block. INIT is either called by the kernel primitive 'create device' which allocates the device an entry in the device table, or, in the case of resident device drivers loaded with the operating system, during the initialisation of the operating system. An UNINIT routine is require which is used to make the physical device 'safe' prior to the removal of the device driver from the operating system. This routine is called by the kernel primitive 'deletedevice', which deallocates the device table entry and should ensure that the physical device will generate no further interrupts. A START routine is required which is called to initiate the handling of a new packet. START is called by 'qpkt' only if the freshly queued packet is the only packet on the device work queue, otherwise START is called from inside the driver once the work required by a previously queued packet has been completed and that packet returned. Once the processing of the START routine is complete, the only way to re-enter the driver code is by a device interrupt. A STOP routine is required which will terminate processing of the packet at the front of the work queue when this packet is recalled by the kernel primitive 'deqpkt' which removes a packet from a work queue. The final component of the driver is the interrupt routine, INT. This routine is responsible for all the intermediate action required to service a packet and ultimately for the return of the packet to its originator. INT is also responsible

for starting the processing of any further packets which are on the work queue, following the return of the head packet, by recalling START if the work queue is not empty.

It can be seen that the structure of a Tripos device driver is very similar to that of the device dependent routines used in the I/O subsystem. In fact, the driver for the graphics board described earlier in this section was converted to operate under Tripos with minimal changes. Detailed information regarding Tripos data structures such as the task table, the device table, device control blocks, stream control blocks and packet structures for various device driver types such as disc drivers and terminal drivers may be found in the Tripos Technical Guide (4–22).

The Tripos operating system image is built by a utility named syslink (4–20, 4–21). This utility creates a loadable file which contains the various system tables, the kernel, the resident libraries, the resident device drivers, and the initial tasks loaded with the system. Five tasks are generally loaded with the operating system, although it is quite possible to load a system image which contains any assortment of tasks. The five tasks generally loaded are a Command Line Interpreter (CLI), a Debug task, a Console Handler, a Filing System task and a Restart task. The CLI interprets command lines obtained from the console handler via the terminal device driver. The CLI assumes that the text contained in the command line is the name of a loadable file which it attempts to load and execute as a coroutine. The CLI also supports the redirection of command input to a file so that sequences of regularly used commands may be submitted to the CLI by a single command invocation.

The console handler may direct input lines to the debug task by means of an escape sequence. The debug task permits the inspection of CPU registers, memory, task stacks and subroutine local variables. It also supports the setting of breakpoints and single stepping through program code with line by line disassembly.

The console handler task interfaces the terminal device to other tasks. Although direct access to the terminal device driver is available to all tasks via the 'qpkt' mechanism, the console handler assembles character input via the terminal device driver into complete lines and then passes these complete lines on to the currently selected task. The console handler also supports the XON/XOFF protocol, the automatic insertion of line feed and carriage return characters when output lines exceed a specified length and optional pausing of the output to enable the user to inspect output before it is scrolled off the top of the terminal screen if the output is particularly rapid.

The file structure supported by the filing system task is described in the Tripos User Guide (4-20). Each Tripos 'volume' is structured as a tree of directories and files. Files may be referenced by their relative position in the tree structure from some currently set directory, or by an absolute reference from the root of the tree structure. As new Tripos volumes are mounted on new disc devices, new filing system tasks are created to manage access to the files on that volume. These multiple filing system tasks share the task software, although they maintain separate data areas.

The restart task is a transient task which checks the validity of

the tree structure of a disc volume. Until the restart task has ascertained the validity of this tree structure and constructed a map of all the allocated blocks on the disc volume, the volume is write protected. When the restart task completes, it sends a packet to the filing system task which contains a bit map of free disc blocks. This bit map is then maintained by the filing system as files are created and deleted. As new disc volumes are mounted, new restart tasks are created to validate the new volumes.

Multiple CLI tasks may also be created in order that a user may invoke more than one Tripos command line at a time. Tripos supports a simple interface for calling assembly language subroutines from within high level languages. This interface has been heavily exploited in the construction of the monitor task for the real-time simulator.

Thus, two operating environments are available, the extended Macsbug environment which offers execution and debugging facilities suitable for use in subsystems not fitted with disc storage and the Tripos operating system which supports an environment in which software may be created as well as excecuted and debugged. A subsystem fitted with disc drive hardware, running the Tripos operating system is ideally suited to the development of software to run both on itself as a Tripos task and on remotely connected subsystems which operate using the extended Macsbug facilities.

4.5.2 The Application Software

The application software is made up of three major components.

Two of these components perform the task of integrating the equations described in Chapter 3 and the third component monitors the simulation, collecting data for storage or display and applying disturbances to the simulated plant in a programmed manner. The first two components are written entirely in assembly language, whilst the third is written in a mixture of BCPL and assembly language. The first two components are executed on two separate subsystems and operate in the Macsbug environment, while the third component runs under a Tripos CLI task on a disc based subsystem. Each of these components may be considered to be a separate task, each task communicating with the other two via the communication link. These tasks will be known as the Generator task, the Volt task and the Monitor task.

The Generator and Volt tasks are each structured as a series of subroutines and each subroutine is responsible for the evaluation of algebraic equations or the integration of dynamic equations. By structuring the software in this manner, the various component subroutines perform a calculation that may be transported to another processing subsystem in the event that one subsystem does not have adequate time to complete all its calculations within an integration step time. The basic structure of the Generator and Volt tasks is shown in Fig. 4.15. Figs. 4.15(a) and (b) show the main task structure, while (c) and (d) show the sequence of subroutine calls which perform the calculation. On entry to both the Generator and Volt tasks, the communication links and the interval timers are initialised. A variety of synchronisation methods have been attempted, from allowing each task to perform its own, independent, interval timing with the simulation data exchanged by the tasks

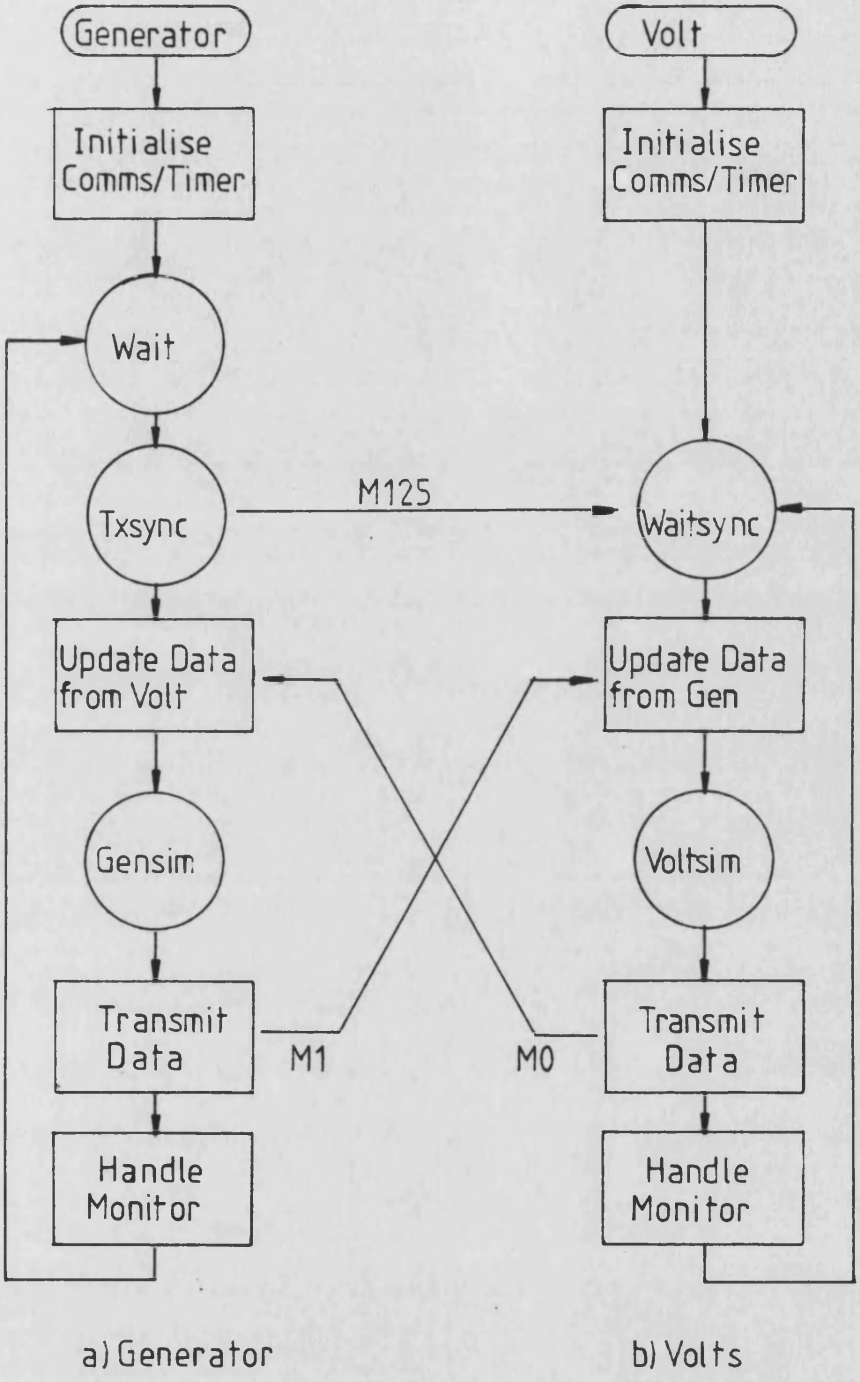
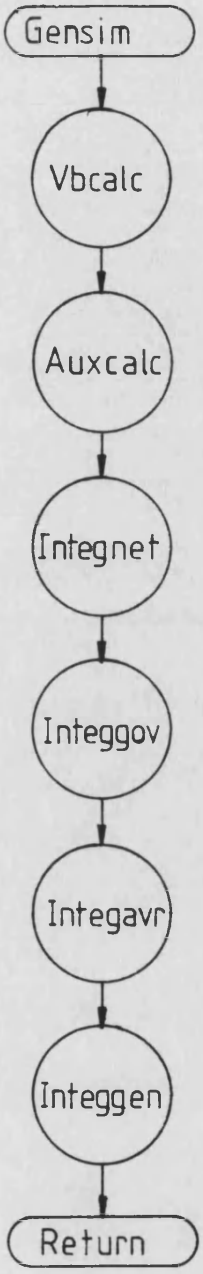
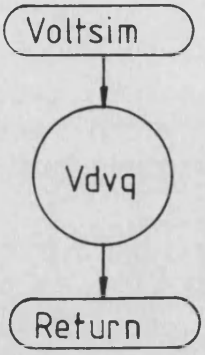


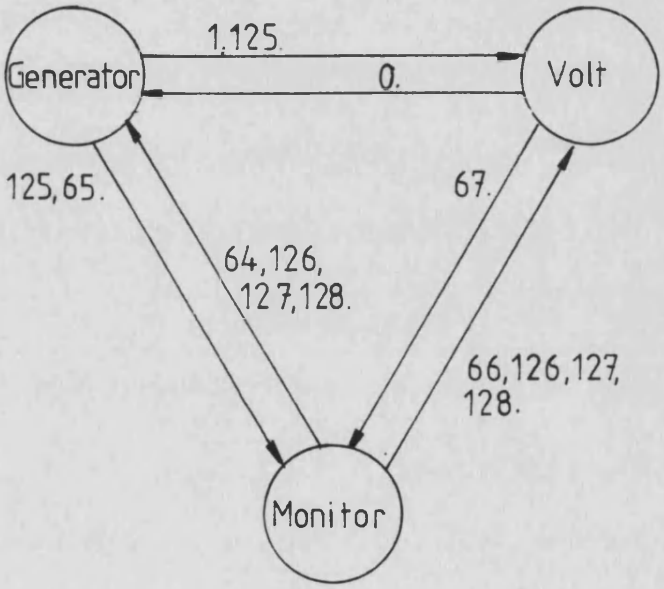
Fig 4.15 Task Structure (1 of 2)



c) Gensim



d) Voltsim



e) Intertask Communication

Fig 4.15 Task Structure (2 of 2)

updated asynchronously during the communication interrupt routine associated with message reception, to the tight synchronisation shown in Fig. 4.15, where message 125 is transmitted by the Generator task to initiate calculation of an integration step. The loosely synchronised strategy proved to be adequate for generating and displaying transient responses. However, identical fault conditions resulted in widely varying performance index values during optimisation studies. Tight synchronisation removes this problem since data is updated at consistent points during the execution of each task. Interval timing is accomplished by submitting a continuous timing request to the I/O subsystem. The delay specified in the I/O packet is equivalent to the integration step time, and the AST routine merely increments a counter maintained in the subsystem memory. The Wait subroutine monitors the value of this counter. If the counter is non-zero then the subroutine decrements the counter and then exits. Otherwise, it loops until the timer AST makes the counter non-zero. In this fashion the simulation will continue to operate if the calculation time is larger than the integration step length and this may be detected if the counter value ever exceeds unity.

Once the timer interrupt has caused the Wait subroutine to exit, the Generator task transmits the synchronisation message and then updates its internal copies of data transmitted by the Volt task (message 0) during the previous integration step. The Volt task behaves in a similar fashion following the reception of the synchronisation message. Both tasks then perform their respective calculations 'Gensim' and Voltsim', and then transmit data for reception by each other and the monitor task. Finally, prior to

returning to the top of their respective loops to wait for the next clock interrupt and the synchronisation message, both tasks perform any changes to their data bases requested by the Monitor task. These changes will be discussed later. Subroutine "Gensim' is merely a list of subroutine calls which perform parts of the simulation. Subroutine 'Vbcalc' calculates the direct and quadrature axis components of the infinite busbar voltage, $V_b \sin \delta$ and $V_b \cos \delta$ respectively, which are used later in both the calculation of the direct and quadrature axis armature currents, I_d and I_q, by the subroutine 'Integnet', and the calculation of the axis voltages, V_d and V_q, by the subroutine 'Vdvq'. Subroutine 'Auxcalc' calculates the auxiliary signals supplied to the governor and excitation systems by calculating the rotor acceleration and the electrical power and then combining these using the appropriate gains. Subroutine 'Integnet' evaluates the direct and quadrature axis currents using equation 3.125, while subroutine 'Integgov' first calculates the electrical torque using equation 3.27 and then integrates the equations associated with the appropriate prime mover representation, equations 3.112 to 3.113 for the turbine prime mover or equations 3.122 to 3.123 for the diesel engine prime mover. Subroutine 'Integavr' applies equations 3.104 and 3.105 in order to step the excitation system forward by one time step and, finally, subroutine 'Integgen' integrates the generator, equations 3.93 to 3.95. Subroutine 'Voltsim' consists of a single subroutine call to subroutine Vdvq which calculates the direct and quadrature axis terminal voltage components, V_d and V_q, using equations 3.48 and 3.49, and then calculates the terminal voltage magnitude using equation 3.12.

Fig. 4.15(e) illustrates the message traffic between the three tasks. Message 0 and 1 exchange simulation data between the Generator and Volt tasks. These two tasks, once initiated, do not require the presence of the Monitor task and will continue to operate without intervention from it. Messages 64 and 66 communicate parameter changes from the Monitor task to each of the simulation tasks and messages 65 and 67 deliver simulation data to the Monitor task every time step. Messages 125, 126, 127 and 128 do not carry any data, the reception of the particular message being all that is necessary to signal its significance. In general, message numbers below 64 have been used for transporting data between tasks directly involved in the simulation calculations, while message numbers 64 and above have been used for communication between the Monitor task and the two simulating tasks. Fig. 4.16 details the contents of each message. All the message entries, except for the 'Txlate', 'Rxlate' and 'Status' fields, are four byte entries. These three entries pass status information about the communication link and interval timing associated with the simulating tasks. 'Txlate' is a count of the number of messages transmitted which have not generated interrupts within an integration step time and 'Rxlate' is a count of the number of times there has been no reception of data with which to update the internal plant variables. The 'Status' entry is the value of the counter incremented by the timer AST and decremented once each time around the calculation loop. If this entry is other than one or zero, the simulation is failing to perform the necessary calculations within the allocated time.

In order that initial conditions may be accurately restored, messages 126 and 127 signal the Generator and Volt tasks to remember

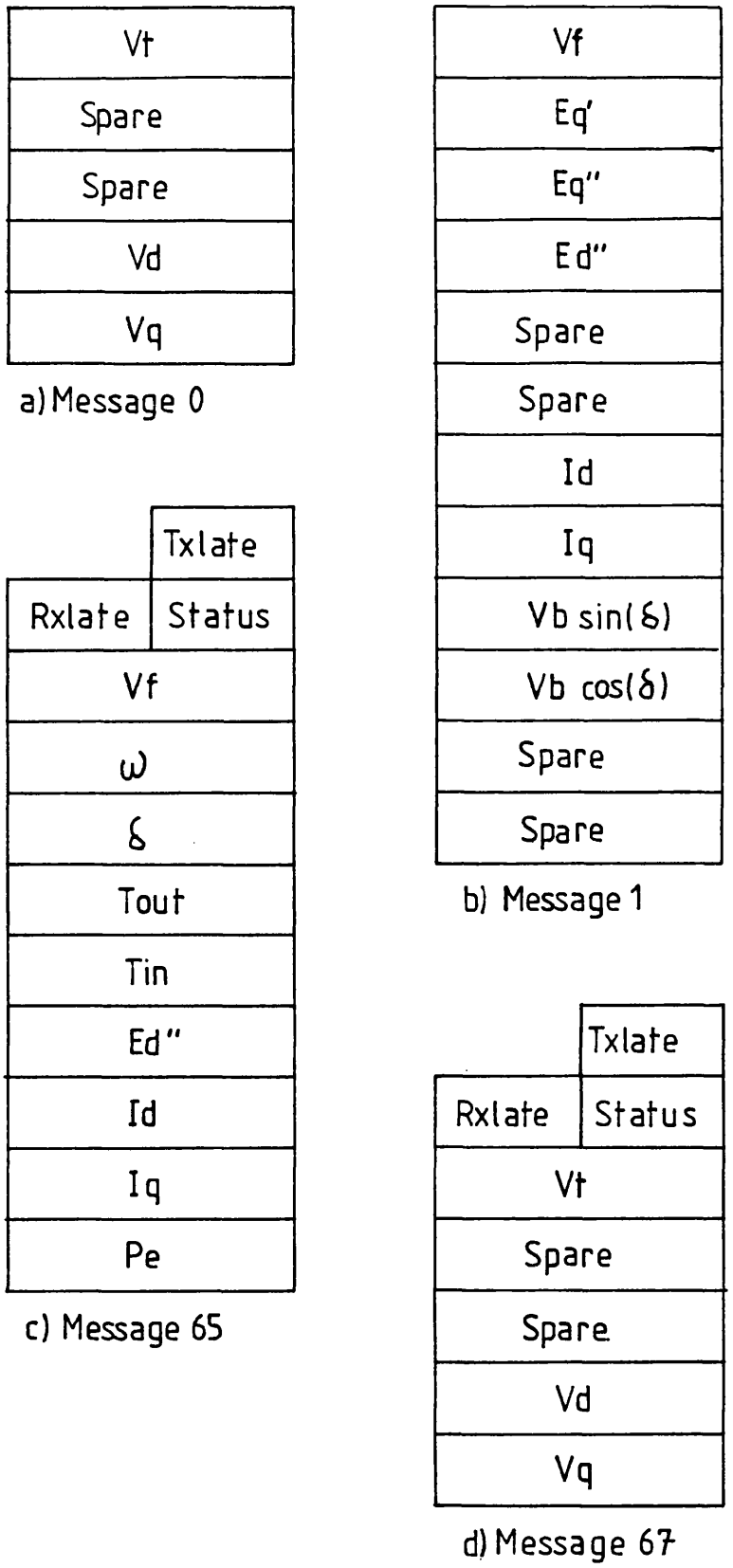


Fig 4.16 Message Formats (1 of 2)

(Rt+Ra)/D
(Xt+Xq″)/D
(Xt+Xd″)/D
Kavracc
Kgovacc
Vb
Avrref
Govref
Vfmax
Vfmin
Kavrpe
Kgovpe
Peinit
Kavrid
Kgovid
Idinit
Tinratemx
Tinratemn
Tinmax
Tinmin

e) Message 64

$$D=(Rt+Ra)^2+[(Xt+Xq'')\times(Xt+Xd'')]$$

Xt
Rt
Vb

f) Message 66

Fig 4.16 Message Formats (2 of 2)

and restore, respectively, all dynamic and algebraic quantities at the end of the current integration step. These two operations are performed by block copy operations in opposite directions. Message 128 attempts to reset the simulation in the event that the simulation becomes unstable. On reception of message 128, both the Generator and Volt tasks initialise all their dynamic and algebraic values to a set of stable initial conditions, identical to those with which the simulation was loaded, prior to execution.

Having described the simulation software and defined the format and purpose of all the messages passed between the participating tasks, attention may now be turned to the Monitor task.

The Monitor task has a number of functions. First, in order for the simulation to be useful, it must be capable of capturing the data sent to it in messages 65 and 67 and must be able to store that data in buffers. It must be capable of presenting this data in a form that is useful to the user and it must be capable of keeping a permanent record of the captured data in a disc file. The Monitor task must also be able to make changes to the plant being simulated in order to simulate transient behaviour as, assuming the simulation is stable, it will eventually settle into steady state operation and the data will become uninteresting. In order to perform higher level functions, other than the capture and display of transient data, such as system optimisation by repeatedly disturbing the simulation and using a performance index to guide the choice of gain parameters, it is necessary to provide a library of routines from which these higher level functions may be created and a simple means of adding to the commands available to the monitor. It is of prime importance that

nothing should be capable of interfering with the data collection function. Data collection must, therefore, be an autonomous function of the Monitor task. This is accomplished by making direct use of the communication link interrupt vector. During initialisation of the Monitor task, the communication link is initialised and programmed to receive messages 65, 67 and 125. On receipt of an interrupt from each message, the interrrupt routine sets a flag to indicate the reception of that particular message and, provided that the appropriate buffers are not full, it will transfer the received data to the data collection buffer, up-date the buffer pointers and increment a count of the messages received. Thus, the interrupt driven collection of data takes precedence over all other subsystem activity, except for higher priority interrupts.

As mentioned earlier, the Monitor task is written in a mixture of BCPL and assembly language. The assembly language routines form the bulk of the routines used for building higher level functions, whilst those functions themselves are written in BCPL in order that they may be easily modified and maintained. As far as possible, each separate monitor command has been kept as a separately compilable module. All the assembly language software is held in one disc file. Partitioning the functions in this manner increases the modularity of the software and reduces the time taken to rebuild the Monitor task following a modification.

The assembly language module contains eleven BCPL callable subroutines and interrupt routines for the clock/timer device and the communication link, as follows:

1. 'Startup' is called from BCPL with three parameters. The first parameter points at the main data collection buffer, the second parameter points at a buffer containing parameters to be sent in messages 64 and 66 and the third parameter defines the length of the data collection buffer. Each of these parameters is stored in a static area associated with the assembly language module, for reference by other BCPL callable subroutines and by the communication link interrupt routine. Once these parameters have been copied, 'Startup' initialises the communication link descriptor tables, points the appropriate interrupt vector at the interrupt routine and enables the communication link interrupt. From this point on, data collection becomes autonomous. The result of the 'Startup' routine, returned to the calling BCPL routine when 'Startup' exits, is a pointer to a status table maintained by the communication link interrupt routine, which keeps count of the number of messages received, transmission and reception failures and the state of the data collection buffers.

2. 'Kill' is called from BCPL to disable the communication link interrupt and leave the link in a state where the Monitor task may safely return its buffers to the operating system freestore and exit to the CLI environment.

3. 'Remember' is called from BCPL to send message 126 which requests the simulation tasks to save their current state.

4. 'Restore' is called from BCPL to send message 127 which

requests the simulation tasks to restore a previously remembered state.

5. 'Reset' is called from BCPL to send message 128 in an attempt to recover control of the simulation should it become unstable.

6. 'Replenish' is called from BCPL to zero the data collection buffer pointers held in the communication link status area. This results in the data collection buffers being rewritten as fresh data arrives from the communication link.

7. 'Prod' is called from BCPL to reinitialise the communication link by rewriting the descriptor tables if it is thought that the communication link has ceased to receive messages.

8. 'Fix 16' is called to convert a 32 bit IEEE single precision floating point value into a 32 bit twos complement fixed point number, with the binary point to the left of bit 15.

9. 'Float 16' performs the inverse operation of 'Fix 16'.

10. 'Fault' is called from BCPL to apply a severe disturbance to the simulation. This routine requires a single parameter, which is the fault duration in 1.25 microsecond ticks, i.e. 800 ticks per millisecond.

11. 'Send' is called from BCPL to send only the post-fault data associated with a fault, to the simulation tasks. This

allows control system reference levels, gains, limits and transmission line parameters to be updated without the necessity of disturbing the simulation with a call to 'Fault'. Careful use of this routine enables various effects to be simulated. The excitation and governing system references may be slowly varied, or it is possible to alter the transmission line parameters or infinite busbar voltage to represent load variations.

The principle disturbance supported by the real-time simulator is a three-phase symmetrical short circuit fault, applied at the high voltage terminals of the generator transformer. The 'Fault' routine, (10) above, applies this disturbance. This is accomplished by changing the impedance present between the generator terminals and the infinite busbar, whilst simultaneously reducing the infinite busbar voltage to zero. The parameter buffer pointed to by the second parameter supplied to the 'Startup' routine, (1) above, contains faulted and post-faulted data for transmission in messages 64 and 66. The 'Fault' routine replenishes the data collection buffers, waits 0.5 seconds and then transmits the faulted parameter values in messages 64 and 66. It then times out the fault duration before transmitting the post-fault data and exiting. Timing is accomplished by use of one of the spare counter/timer circuits of the PTM, since the regular operating system clock interrupt occurs at 20 millisecond intervals and so does not offer a fine enough resolution for timing fault durations. To make use of a different counter/timer requires that the clock interrupt vector is redirected to a piece of assembly code which determines whether the Tripos kernel services the timer interrupt or whether it is intended for use in timing the fault

duration. When the fault timer expires, the interrupt routine sets a flag to indicate this condition. This flag is polled by the 'Fault' routine prior to the transmission of messages 64 and 66. All timer initialisation and message transmission operations associated with the 'Fault' and 'Send' routines are synchronised to occur immediately following the reception of message 67 from the Volt task. This ensures that data transmission and, hence, reception of the data by the receiving tasks is synchronised to the simulation and that the updating of parameters in the Volt task occurs one integration period later than those in the Generator task. This reduces the interfacing error associated with the voltage calculation since it is one step behind the network calculation which generates the corresponding currents.

The main BCPL section of the task is entered when the task is loaded. It first claims the necessary buffer space from the operating system, to construct the parameter and data collection buffers. If this is successful, the assembly language 'Startup' routine is called to initiate the data collection. The task then drops through to its main loop which involves prompting the terminal for input, reading the first item from the input stream, i.e. the terminal, and searching for a matching item in a lookup table. If a match exists, the corresponding action routine is called. If no match exists, then the whole line is passed to a dynamically-created CLI task for execution. This feature permits operating system commands, such as those for mounting and dismounting disc volumes, to be executed from within the Monitor task, without the loss of data that would occur by exiting the Monitor task, issuing the operating system command and then reinvoking the Monitor task. Once the action

routine completes, the process is repeated. The currently available monitor commands are:

1. Fault; this command indirectly calls the assembly language 'Fault' routine, having first obtained a fault duration, in milliseconds, from the user.

2. Send; directly calls the assembly language 'Send' routine.

3. Reset; directly calls the assembly language 'Reset' routine.

4. Remember; directly calls the assembly language 'Remember' routine.

5. Restore; directly calls the assembly language 'Restore' routine.

6. Replenish; directly calls the assembly language 'Replenish' routine.

7. Plot; this command plots data, held in the data collection buffer, on the High Resolution Graphic display. The command prompts for new axis, the time period and the variable to be plotted. Consecutive plots may either be superimposed or given fresh axes and titles.

8. Change; this command allows the user to change the various values held in the parameter buffer. The user may modify transmission line parameters, control loop references,

excitation and governing system limits, the infinite busbar voltage and the gains supplying extra feedback signals to the excitation and governing systems.

9. Show; this command tabulates the contents of the Monitor task data base so that the user may ensure that a change has been made correctly.

10. Keep; this command stores the Monitor task data base to a specified file.

11. Get; this command recovers a data base file previously stored by the Keep command. Use of the Get and Keep commands eases the burden on the user when regular and repetitive changes to the data base are required.

12. Save; this command saves the data from the data collection buffer. The command requires two parameters, the time period over which data is to be saved and the stem of a file name. The stem of the file name is extended by a two or three letter extension, which identifies the particular plant variable stored in the file. If the keyword 'select' is included in the command line, the routine will prompt for a yes/no response with the two or three letter extension code of the file about to be written. Only a 'yes' response will cause the file to be written and the corresponding data saved. It was felt that selective saving of data buffers is important since, with 16 seconds of buffer storage at 5 millisecond intervals, each file may store up to 3200 data

values. Each file would then be a maximum size of 17 Tripos disc blocks (16 data blocks and one file header block). With 12 such buffers, one for each of the variables sent in messages 65 and 67, only 6 complete 16 second, 12 file records may be stored on one floppy disc. The structure of files generated by this command is shown in Appendix B.

13. Look; this command generates a tabular display of the data collection buffer on the user's terminal. If the command is followed by a numerical value, this is taken to be the starting time, in milliseconds, for the table. Otherwise, the tabulation will commence from the start of the data collection buffer.

14. Wait; this command allows the user to enter a delay in seconds, so that commands may be typed ahead. This command also checks that the data collection buffers have been filled beyond the specified delay time. If the buffer pointers have not passed the appropriate point when the time delay expires and have still not passed this point after a further short delay, the communication link is assumed to have failed and the assembly language routine 'Prod' is called.

15. Per; this calculates the performance index of the transient response stored in the data collection buffer, over a specified control interval, with the specified weighting coefficients.

16. Impedance; this initiates a sequence of faults which generate a pre-fault/post-fault impedance chart. This command will be discussed in more detail in Chapter 5.

17. PQ; this initiates a series of control loop reference changes which generate the dynamic stability limit in the leading region of the operating chart. This command will be discussed in more detail in Chapter 5.

18. Simplex; this initiates a parameter optimisation procedure which establishes optimum gain values for the feedback of extra signals into the excitation and governing systems. Parameter optimisation will be considered in more detail in Chapter 6.

19. Status; this produces a tabular display of the 'Txlate', 'Rxlate', and 'Status' values passed across with messages 65 and 67, along with a count of the total number of messages received from each simulation task and a percentage indication of how full the data collection buffers are.

20. Quit; this command calls the assembly language 'Kill' routine to stop data collection, releases the data collection and parameter buffers to the operating system freestore and causes the Monitor task to exit back to the operating system CLI task.

As can be seen from this list, many of these commands directly call assembly language subroutines or, conversely, many of the

commands may be called as subroutines from within higher level functions. Apart from the assembly language routines, the following routines, written in BCPL, have been found useful in writing higher level functions.

1. Waitbuffer (time); this is similar to the wait command, where 'time' is a floating point number. This subroutine returns when the data collection buffer fills beyond 'time' seconds.

2. Poleslip (time); this routine returns 'true' if the stored load angle response exceed π radians in the interval $0 \leq t \leq$time, where 'time' is a floating point value, in seconds. Otherwise this routine returns 'false'.

3. Pi (stime, period); this routine returns a floating point value for the performance index of the stored transient over the control interval 'period' seconds, starting at 'stime' seconds. Both parameters are floating point values.

4. Graph (false); this routine superimposes a new copy of the previously plotted transient on the graphics display. The screen is cleared every seven times this routine is called to avoid a cluttered display.

The Monitor task therefore performs all the data collection necessary to capture data from the real-time simulation. It provides a number of terminal commands which exercise direct control over the simulation and it provides a library of building blocks for

constructing higher level functions. This library is by no means exhaustive and, as new higher level functions are implemented, this library will grow.

4.5.3 Support Software

Apart from the standard utilities supplied with the Tripos operating system, such as the BCPL compiler, the MC68000 Macro Assembler and text editors, a number of small software packages have been written specifically for the support of the real-time simulator.

The Program Loader is a short assembly language program which is run on the basic subsystems to load software from the communication link into the subsystem memory and to finally execute the loaded program. This program is stored as firmware on the EPROM board and may be executed by issuing the appropriate command line to the monitor prompt. On entry, this program initialises the communication link and permits the reception of a single message. The message number of this message is different for each subsystem so that subsystems may be identified by their message number for program loading purposes. The loader maintains a workspace of four longwords in an unused area of the communication link fast memory. Programs loaded by the loader are unlikely to reside in the communication link memory. Having initialised the communication link, the loader enters an infinite loop and all further action occurs within the communication link interrupt routine. The loader expects to receive data in the format of Tripos absolute hunks (abshunks), shown in Appendix B, which are generated by various of the Tripos utilities. The entries kept in the workspace detail the progress of record

interpretation, the current load address, the number of longwords remaining in the current record (abshunk) and the program entry point. When a communication link interrupt occurs, the interrupt routine will only act on it if it signals the reception of the correct message number. The interrupt routine determines from the workspace whether the first word of the message is the first, second or third word of the abshunk header or a data word and commences or recommences record interpretation at the corresponding point. When processing reaches the end of the message, the workspace is updated so that record interpretation may continue in the correct manner when the next message arrives. If the incoming data contains a reference to the reset vector, the corresponding data is stored for later use as the entry point, otherwise the entry point will default to $2000. A sequence of abshunk records is terminated by an end of file record and, when this arrives, the loader will lower the CPU interrupt mask and perform a subroutine call to the program entry point. Should the program loaded return control to the loader, the loader will attempt to load and execute further programs.

A companion program has been written to run on the disc based subsystem. This utility reads a specified Tripos file into a buffer in the subsystem memory. It then calls an assembly language subroutine to break this buffer into message sized segments, which it then copies, in turn, into the communication link fast memory and transmits them using a specified message number. If the hardware detects a transmission failure, the failed message is retransmitted. A short pause is inserted between transmissions, to give the receiving subsystem time to handle the incoming message. No interpretation of the file is made by this program, the binary image

is merely divided into messages and sent for interpretation by the receiving subsystem.

These two programmes offer a simple mechanism for sending software, created and maintained on the disc based subsystem, to the basic subsystems for execution. Error handling is very rudimentary, retransmissions only occurring when a checksum error is signalled by the receiving subsystem. No flow control is implemented, so that, if messages are sent too fast, they may be missed entirely and the loader will fail. The delay inserted between transmissions gives adequate time for the receiving subsystem to become ready to receive the next message. Higher level communication software may be loaded by this mechanism in order to establish more sophisticated protocols, if necessary, in the future.

The calculation of the direct and quadrature axis components of the infinite busbar voltage require the calculation of the sine and cosine of the load angle. In order to save time, this calculation is performed using a lookup table. This lookup table comprises some 25736 entries giving regularly spaced sine values for angles between 0 and $\pi/2$ radians. This table is established by a short assembly language program which may be loaded into the simulating subsystems using the loading mechanism, described above, prior to loading the main simulation software. This program takes approximately 1 minute and 40 seconds to establish the lookup table.

Many of the coefficient values in the equations 3.93 to 3.125 are assembled into the simulation software. In order to make changes to parameters not accessible to the Monitor task, a small program has

been written which generates an assembly language source file defining the various coefficient values. Input to this program may either be direct from the terminal or via a Tripos command file. Use of the later method enables parameter changes to be made by editing the command file and regenerating the assembly language source file. The resulting file is then included as a header file when the simulation software is assembled.

A debug package has been rewritten (4-4) for the basic subsystems which allows the user to inspect and modify memory locations in a variety of ways. Memory locations may be examined as byte, word and longword sized operands, and inspected or updated using decimal and hexadecimal representations, or as ASCII character values, or as floating point numbers. This utility makes use of the conversion facilities mentioned in section 4.5.1. The user may step forward or backward through memory locations using single keystrokes, amending data values and changing the representation format as required.

Finally, a plotting utility has been written. This utility takes input data from files in the standard data file formats given in Appendix B. The utility prompts the user for file names, titles, plotting colour and whether or not to rescale the axes when separate plot files are superimposed. Two versions of this utility are available, one produces output on the high resolution graphics display, the other produces output on a digital plotter. This later version of the utility was used to generate the transient responses given in Chapter 6.

4.6 Simulator Operation

In order to run the simulator from a powered down situation, a number of steps must be followed. Following power up, each of the subsystems and the communication controller is reset. The disc based subsystem is then 'bootstrapped' to load the Tripos operating system from the floppy disc. The remaining subsystems are given the command to execute the program loader described in the previous section. From this point onwards, all the commands necessary to run the simulation may be issued from the disc based subsystem. Once the various operating system startup procedures, such as setting the date and time, and mounting further floppy discs containing the simulation software and sufficient space for recording simulation data have been performed, the user may then initialise the simulation. First, the sine lookup table must be generated by loading the sine program into both the simulating subsystems. Once the sine program exits, the Generator and Volt tasks may be loaded. The necessary commands to perform all these actions may be stored in a command file and the whole process automated. The Monitor task may now be entered and exited at will. The simulation will continue to run in the background while the disc based subsystem is performing other tasks, such as rebuilding the Monitor task during the development of a new command or facility.

References

4-1 Windsor, K.J.: "Transient Analysis of synchronous machines in power systems using analogue computer techniques",

University of Bath, School of Electrical Engineering, Project Report No.62-72, 1972.

4-2 Aldred, A.S. and Doyle, P.A.: Electronic analogue computer study of synchronous machine transient stability", Proc.IEE, 1956, Vol.104, A, p152.

4-3 Aldred, A.S.: "Electronic analogue computer simulation of multimachine power system networks", Proc.IEE, 1961, Vol.109, A, p195.

4-4 Tanner, D.G.: "Real-time simulation of power systems", PhD Thesis, 1982, University of Bath.

4-5 "MC68000 16 Bit Microprocessor Users Manual", 2nd Edition, Motorola Inc., 1980.

4-6 "Micro computer components", Motorola Inc., 1979.

4-7 Advanced Micro Devices: "Am2960 boosts memory reliability", Technical Report, Advanced Micro Devices, 1980.

4-8 Kane, J. and Osborne, A.: "An introduction to microcomputers, Vol.3, Some Real Support Devices", September 1978, Osborne & Associates Ltd.

4-9 "A proposed standard for binary floating-point arithmetic", (Draft of IEEE task P754), Computer Vol.14, No.3, May 1981, pp51-62.

4–10 Thorne, J.G.: "A high speed communication link used with parallel processing nodes", Private communication, 1982.

4–11 International Standard Organisation: "Data communication – high level data link control procedures – elements of procedures", ISO4335–1979(E).

4–12 CCITT: "Interface between DTE and DCE for terminal operating in the packet mode on public networks", CCITT recommendation X25, 1980.

4–13 "Military standard 1553 B", American Triforces Standards.

4–14 Xerox, DEC, Intel Corp.: "Ethernet specification", Joint publication, 1980.

4–15 Philips, S.: "A Cambridge base", International Systems, Vol.10, No.5, May 1982.

4–16 "Western Digital 1983 Components Handbook", Western Digital Corp., 1983.

4–17 Habersinski, C.: "Development of an intelligent floppy disc controller for the Bath University MC68000 computing system", Project Report for BSc, University of Bath, May 1982.

4–18 "Graphics Display Processor", Data Sheet, Thomson-Efcis, MOS Integrated Circuits.

4–19 "Macsbug Monitor Reference Manual", M68K MACSBG (D1), 1st Edition, Motorola Inc., 1982.

4–20 King, T.J.: "Tripos User Guide", Issue 2, March 1983, School of Mathematics, University of Bath.

4–21 King, T.J.: "Tripos Programming Guide", Issue 2, April 1983, School of Mathematics, University of Bath.

4–22 King, T.J.: "Tripos Technical Manual", Issue 2, May 1983, School of Mathematics, University of Bath.

4–23 Richards, M, Alyward, A.R., Bond, P., Evans, R.D. and Knight, B.J.: "Tripos – a portable operating system for mini-computers", Software Practice and Experience, Vol.9, pp513–526, 1979.

4–24 Richards, M. and Whitby-Strevens, C.: "BCPL – The language and its compiler", Cambridge University Press, 1980.

CHAPTER 5

DYNAMIC AND TRANSIENT STABILITY STUDIES

5.1 Introduction

When a scheme is proposed with the aim of improving the transient performance of a power system, it is important to consider the effect of such a scheme on both the dynamic and the transient stability of the generating plant. It is not sufficient merely to show that an improvement in transient performance has been obtained, and it is necessary to show that there has been no significant degradation of the dynamic and transient stability limits. It is also useful to be able to make quantative comparisons of the stability limit changes obtained from different schemes so that, for example, rival schemes may be assessed on the basis of the cost of the scheme versus the relative stability limit change obtained from that scheme.

In the previous Chapter, a Real Time Power System Simulator was described and its primary function is to simulate the transient behaviour of a power system following a disturbance. One of the main objectives of this thesis is to show the use of this simulator in the optimisation of the transient performance of a power system and, in order that changes in the stability limits may be compared, a secondary objective has been to implement some means of making this comparison.

Dynamic and transient stability changes may be assessed using two

high level functions supported by the Monitor task of the Real Time Simulator. This Chapter describes the mechanisms by which the Monitor task generates Dynamic Stability Limit curves and Prefault/Postfault impedance plots for the system of Fig 6.8.

5.2 Dynamic Stability

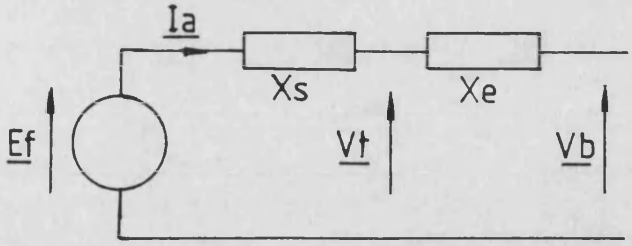
As defined in Chapter 1, the dynamic stability limit of a generating plant is the maximum power that it may output, without losing synchronism, when subjected to a small, slow disturbance.

The dynamic stability limit forms one of the boundaries of the machine operating chart. The other boundaries of the operating chart are imposed by the maximum power output of the prime mover and by heating of the rotor and stator windings. Fig. 5.1(a) shows an equivalent circuit diagram for a round rotor synchronous generator, with synchronous reactance, x_s, and connected to the infinite busbar by the external reactance, x_e. It has been assumed that the armature resistance is negligible and that the external impedance is entirely reactive. An expression for the real and reactive power at the busbar may be written as:

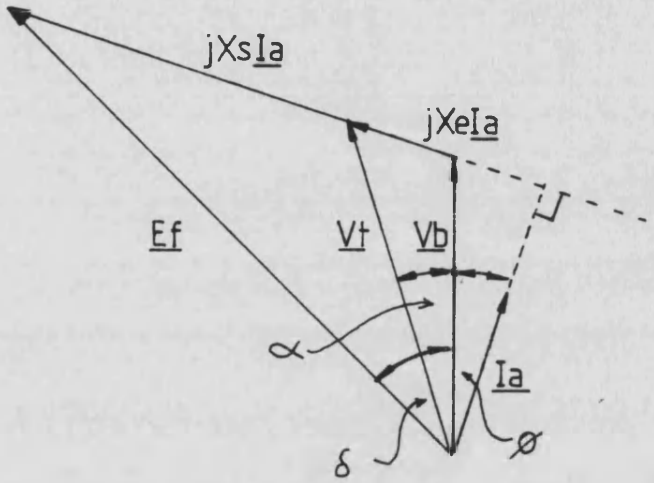
$$P + jQ = \underline{V_b} \cdot \underline{I_a}^* \qquad (5.1)$$

where P is the real power, Q is the reactive power and $\underline{I_a}^*$ is the complex conjugate of the armature current phasor. Equation (5.1) may be expanded as:

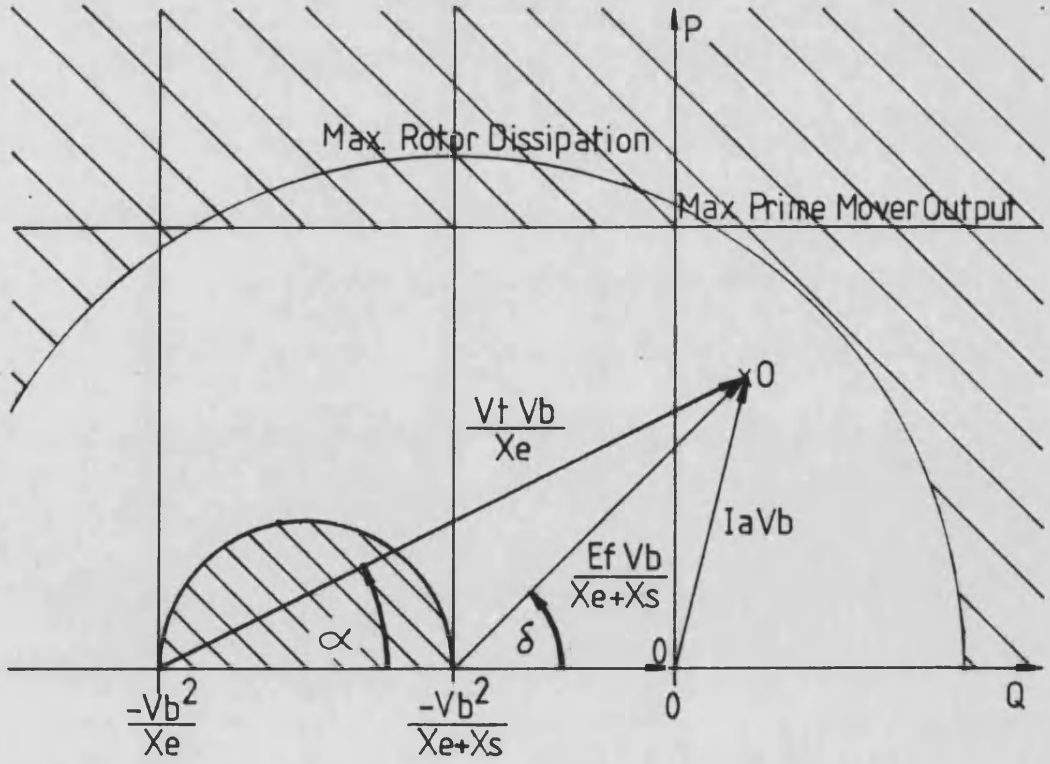
$$P + jQ = \underline{V_b} \cdot (\underline{E_f} - \underline{V_b})^*/j(x_e + x_s)^* = \underline{V_b}.(\underline{V_t} - \underline{V_b})^*/jx_e^* \qquad (5.2)$$



a) Equivalent Circuit



b) Phasor Diagram



c) Operating Chart

Fig 5.1 Round Rotor Synchronous Machine

Assuming the real axis to be aligned along the direction of the busbar voltage phasor, $\underline{V_b}$, in Fig. 5.1(b), then equation (5.2) may be written in terms of magnitudes and phase, as:

$$P + jQ = \frac{V_b.(E_f(\cos\delta - j\sin\delta) - V_b).j}{(x_s + x_e)} = \frac{V_b(V_t(\cos\alpha - j\sin\alpha) - V_b).j}{x_e} \qquad (5.3)$$

By collecting real and imaginary terms:

$$P = \frac{E_f V_b \sin\delta}{x_e + x_s} = \frac{V_t V_b \sin\alpha}{x_e} \qquad (5.4)$$

and

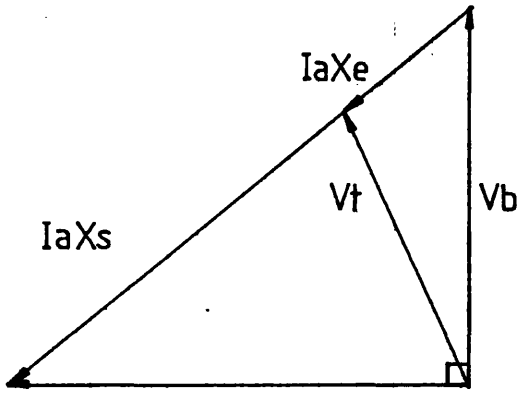
$$Q = \frac{V_b E_f \cos\delta - V_b^2}{x_e + x_s} = \frac{V_t V_b \cos\alpha - V_b^2}{x_e} \qquad (5.5)$$

The angles and magnitudes given in equations (5.3) to (5.5) are shown in Fig. 5.1(b). Positive angles are measured in an anticlockwise direction. Fig. 5.1(c) shows the machine operating chart, with the boundary imposed by the prime mover output power shown as a horizontal line and the boundary imposed by rotor heating shown as a circular arc. The power lost in heat due to the resistance of the field winding is proportional to square of the field current, thus a limit on the rotor power dissipation places a limit on the field current. The e.m.f. induced in the armature circuit by the motion of the machine rotor, E_f, is also proportional to the field current, so the rotor power dissipation restriction limits the

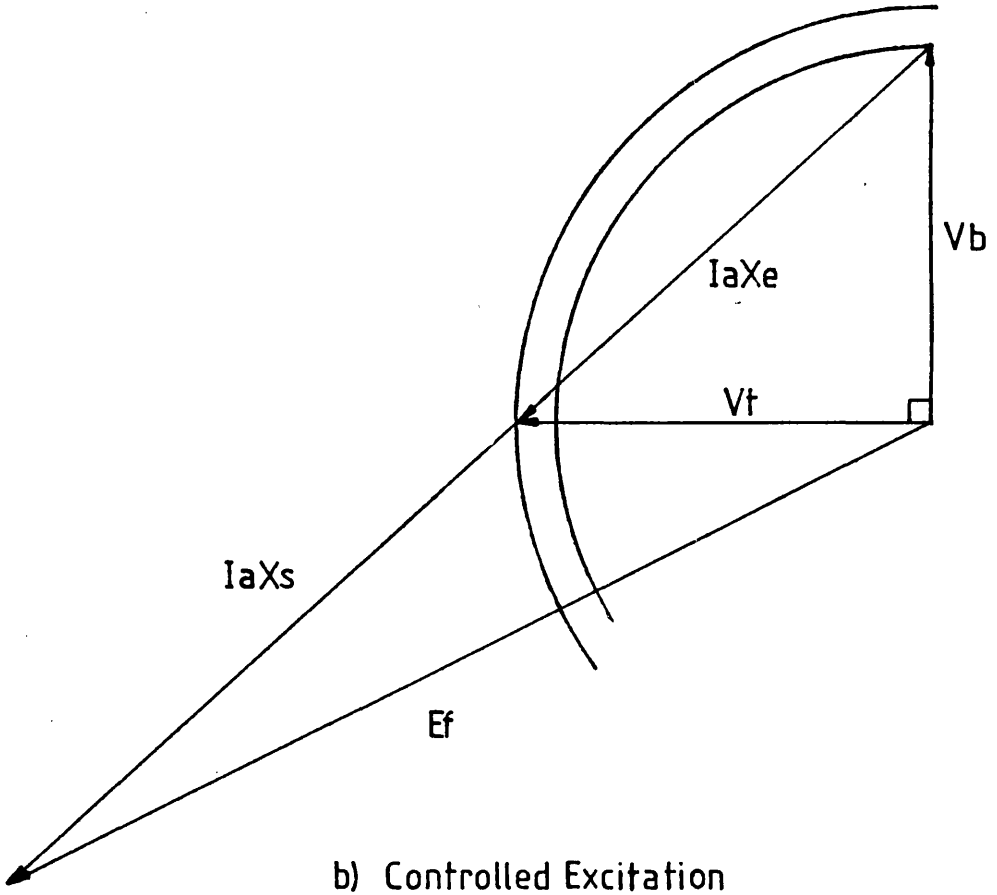
maximum value of E_f. It can be seen that, by scaling the voltage phasor diagram of Fig. 5.1(b) with a scaling factor of $V_b/(x_e + x_s)$, the voltage phasor diagram may be constructed on the operating chart. The restriction imposed by maximum rotor power dissipation is therefore a circular arc, centred at the point $-V_b^2/(x_s + x_e)$ on the reactive power, Q, axis.

When the generator is operated with constant excitation, i.e. E_f is constant, then, by equation (5.4), the maximum electrical power output to the busbar occurs when the load angle, δ, reaches 90 degrees. This limit, sometimes termed the steady state stability limit, is drawn onto the operating chart of Fig. 5.1(c) as a vertical line on the reactive power axis. If the machine is operating at a load angle of 90 degrees with constant excitation, and the prime mover output power is increased, the load angle will increase beyond 90 degrees. In accordance with equation (5.4) the power delivered to the infinite busbar will be reduced by this action. This will aggravate the problem, since it increases the power available to accelerate the rotor shaft, further increasing the load angle and reducing the electrical power output. Synchronism can be lost and the machine then pole slips. Fig. 5.2(a) shows a voltage phasor diagram for this limiting condition.

If an automatic voltage regulator is used to maintain the terminal voltage, the operating load angle may be increased beyond 90 degrees. Assuming that the excitation control system is able to maintain the terminal voltage at a constant magnitude, then, by equation (5.4), the maximum electrical power output to the busbar now occurs when the angle, α, between the terminal voltage vector and the

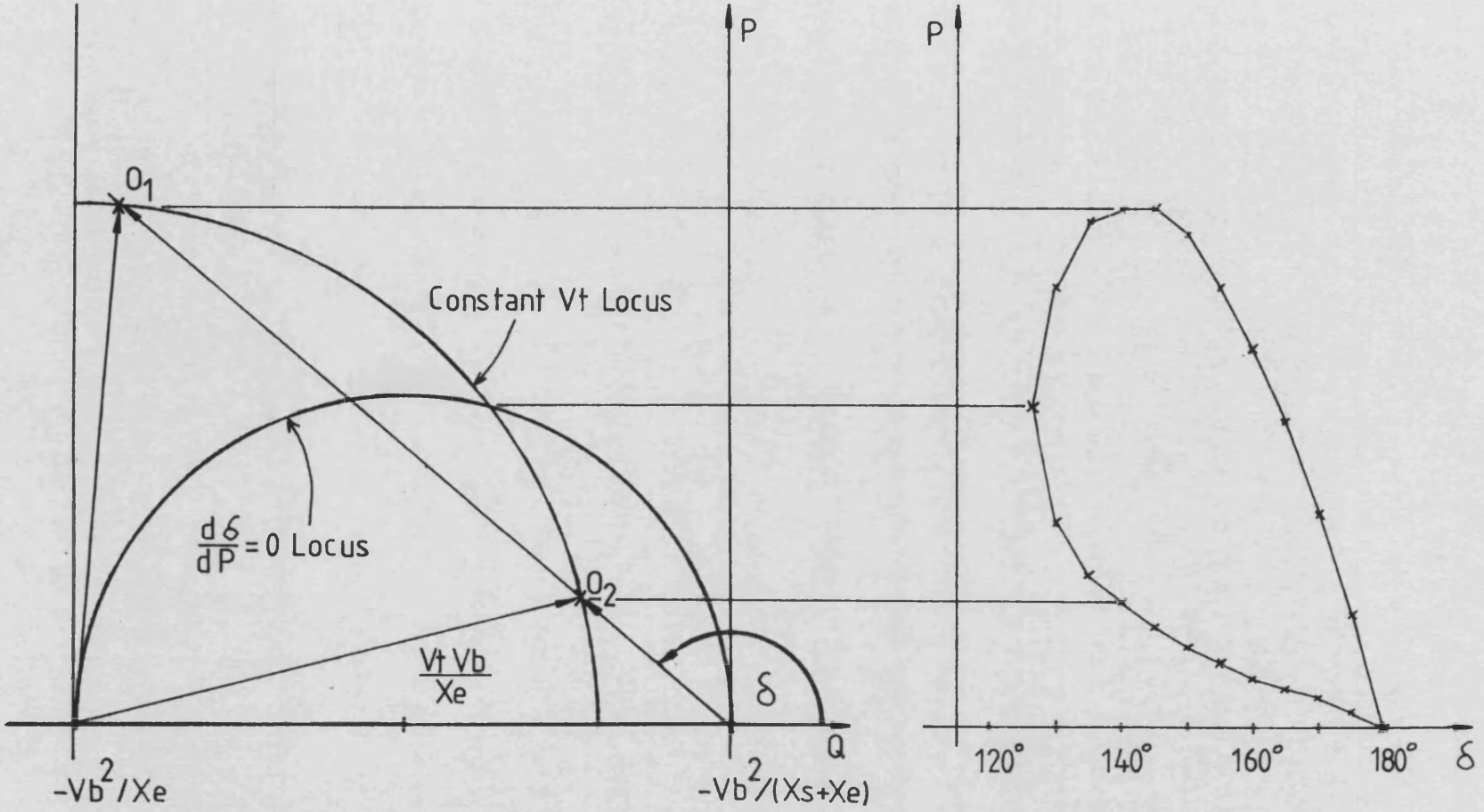


a) Constant Excitation



b) Controlled Excitation

Fig 5.2 Dynamic Stability Limits (1 of 2)



c) Operating Chart

d) Power v. Load Angle

Fig 5.2 Dynamic Stability Limits (2 of 2)

busbar voltage, reaches 90 degrees. This limiting circumstance is shown in Fig. 5.2(b). The terminal voltage vector, $\underline{V_t}$, may be written in terms of the magnitude of the induced e.m.f., E_f, the load angle, δ, and the busbar voltage, V_b, as:

$$\underline{V_t} = \frac{V_b x_s + E_f x_e \cos\delta}{(x_s + x_e)} - j\frac{E_f x_e \sin\delta}{(x_s + x_e)} \qquad (5.6)$$

This may be written in terms of the terminal voltage magnitude so that:

$$E_f^2 x_e^2 + 2V_b x_s x_e . \cos\delta . E_f + x_s^2 . V_b^2 - V_t^2 . (x_s + x_e)^2 = 0 \qquad (5.7)$$

and

$$\frac{dE_f}{d\delta} = \frac{E_f V_b x_s x_e \sin\delta}{E_f x_e^2 + V_b x_s x_e \cos\delta} \qquad (5.8)$$

Synchronism is lost when the slope of the power output versus load angle characteristic is negative. By considering the derivative of equation (5.4) with respect to load angle:

$$\frac{dP}{d\delta} = \frac{E_f V_b \cos\delta}{(x_s + x_e)} + \frac{dE_f}{d\delta}\frac{V_b \sin\delta}{(x_s + x_e)} \qquad (5.9)$$

Taking this slope to be zero and substituting equation (5.8) into equation (5.9) gives: $Q = V_b^2/x_e$ at the stability limit. This is shown in Fig. 5(c) by the vertical line through $Q = -V_b^2/x_e$. On the operating chart, under constant terminal voltage conditions, the loci

of the operating point, O, are circular arcs, centred at $Q = -V_b^2/x_e$ with a radius of $V_t V_b/x_e$. If this radius is such that the corresponding locus cuts the reactive power axis between $Q = -V_b^2/(x_s + x_e)$, and $Q = -V_b^2/x_e$, then the load angle δ is always greater than 90^o, and at a constant load angle the operating locus is intercepted at two points by a line extended along the direction of the excitation vector. This is illustrated in Fig. 5.2(c). Thus for a given terminal voltage and load angle, there are two operating points, O_1 and O_2. It can be seen from Fig. 5.2(d) that the slope of the Power versus load angle characteristic is positive only for a very small range of load angles. By inverting equation (5.9) and substituting equation (5.8) for $dE_f'/d\delta$ it is possible to show that the operating locus for $d\delta/dP = 0$ is a semi-circle centred at $Q = \frac{-V_b^2(x_s + 2x_e)}{2x_e(x_s + x_e)}$ with a radius of $\frac{V_b^2 x_s}{2x_e(x_s + x_e)}$. This locus is shown in Figs. 5.2(c) and 5.1(c). Operating inside this locus is unstable, hence the hatching lines in Fig. 5.1(c).

These two sets of stability limits were both obtained for ideal conditions; in particular, the controlled excitation system assumed perfect terminal voltage regulation. The dynamic stability limits were obtained from the steady state power versus load angle characteristic. Synchronism is lost in these cases due to the loss of synchronising torques and no consideration is given to the presence of damping torques. In order to account for both synchronising and damping torques, a dynamic representation of the synchronous machine must be considered.

The third order machine representation given in Chapter 3 may be

simplified by neglecting both the armature and the transmission line resistances. This representation gives rise to the following dynamic and algebraic equations:

$$pe'_q = (V_f - (x_d - x'_d).\, i_d - e'_q)/T'_{do} \qquad (3.28)$$

$$V_d = x_q.\, i_q = V_b \sin\delta - x_T.\, i_q \qquad (5.10)$$

$$V_q = e'_q - x'_d.i_d = V_b \cos\delta + x_T.i_d \qquad (5.11)$$

$$T_{elec} = e'_q.i_q - (x'_d - x_q)i_d.i_q \qquad (3.31)$$

$$MpY = \left[T_{out} - T_{elec} - \frac{K_d Y}{\omega_o}\right] \qquad (3.13)$$

$$p\delta = Y \qquad (5.12)$$

$$i_d = \frac{e'_q - V_b \cos\delta}{x'_q + x_T} \qquad (5.13)$$

$$i_q = \frac{V_b \sin\delta}{x_q + x_T} \qquad (5.14)$$

$$V_t = (V_d^2 + V_q^2)^{1/2} \qquad (3.12)$$

These equations may then be linearised about some operating point, e'_{qo}, V_{to}, V_{do}, V_{qo}, T_{eleco}, T_{outo}, δ_o, i_{do} and i_{qo} (the steady state slip speed, Y, is zero). The machine dynamics may then be expressed

in terms of the small disturbance variables, $\Delta e'_q$, ΔV_f, ΔV_d, ΔV_q, ΔT_{elec}, ΔT_{out}, $\Delta\delta$, Δi_d, and Δi_q. The dynamic equations may be expressed in state space form as:

$$\begin{bmatrix} \dot{\Delta e'_q} \\ \dot{Y} \\ \dot{\Delta\delta} \end{bmatrix} = \begin{bmatrix} a_{11} & 0 & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta e'_q \\ Y \\ \Delta\delta \end{bmatrix} + \begin{bmatrix} 1/T'_{do} & 0 \\ 0 & 1/M \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta V_f \\ \Delta T_{out} \end{bmatrix} \qquad (5.15)$$

where $$a_{11} = \frac{-(x_d + x_T)}{(x'_d + x_T)\, T'_{do}}$$

$$a_{13} = \frac{-V_b \sin\delta_o .(x_d - x'_d)}{T'_{do}.(x'_d + x_T)}$$

$$a_{21} = \frac{-i_{qo}(x_q + X_T)}{M\,(x'_d + X_T)}$$

$$a_{22} = \frac{-K_d}{\omega_o M}$$

$$a_{23} = \frac{-1}{M}\left[\frac{V_b \sin\delta_o .(x_q - x'_d).i_{qo}}{(x'_d + x_T)} + \frac{[e'_{qo} + (x_q - x'_d) i_{do}] V_b \cos\delta_o}{(x_q + x_T)} \right]$$

and the small change algebraic equations are:

$$\Delta V_d = x_q \, \Delta i_q \qquad (5.16)$$

$$\Delta V_q = \Delta e'_q - x'_d \Delta i_d \qquad (5.17)$$

$$\Delta T_{elec} = [\Delta e'_q - (x'_d - x_q)\Delta i_d]i_{qo} + [e'_{qo} - (x_d - x_q)i_{do}]\Delta i_q \qquad (5.18)$$

$$\Delta i_d = \frac{\Delta e'_q + V_b \sin\delta_o . \Delta\delta}{(x'_d + x_T)} \qquad (5.19)$$

$$\Delta i_q = \frac{V_b . \cos\delta_o . \Delta\delta}{(x_q + x_T)} \qquad (5.20)$$

$$\Delta V_t = \frac{\Delta Vq . Vq_o}{Vt_o} + \frac{\Delta Vd . Vd_o}{Vt_o} \qquad (5.21)$$

Fig. 5.3 shows a block diagram of this linearised representation.

Synchronous machine stability is concerned with the stability of the load angle and slip speed loop. In the manner of Demello and Concordia (5-1), positive damping torques may be identified as those in phase with the slip speed, and synchronising torques may be identified as those in phase with the change in load angle, $\Delta\delta$. Positive damping is therefore provided by the K_d/ω_o block in Fig. 5.3, while synchronising torques are positive provided the constant a_{23} is negative. Given an operating power output, P, and an operating load angle, δ_o:

$$P = V_b \sin\delta_o . i_{do} + V_b \cos\delta_o i_{qo} \qquad (5.22)$$

so that:

$$i_{qo} = \frac{V_b \sin\delta_o}{(x_q + x_T)} \qquad (5.23)$$

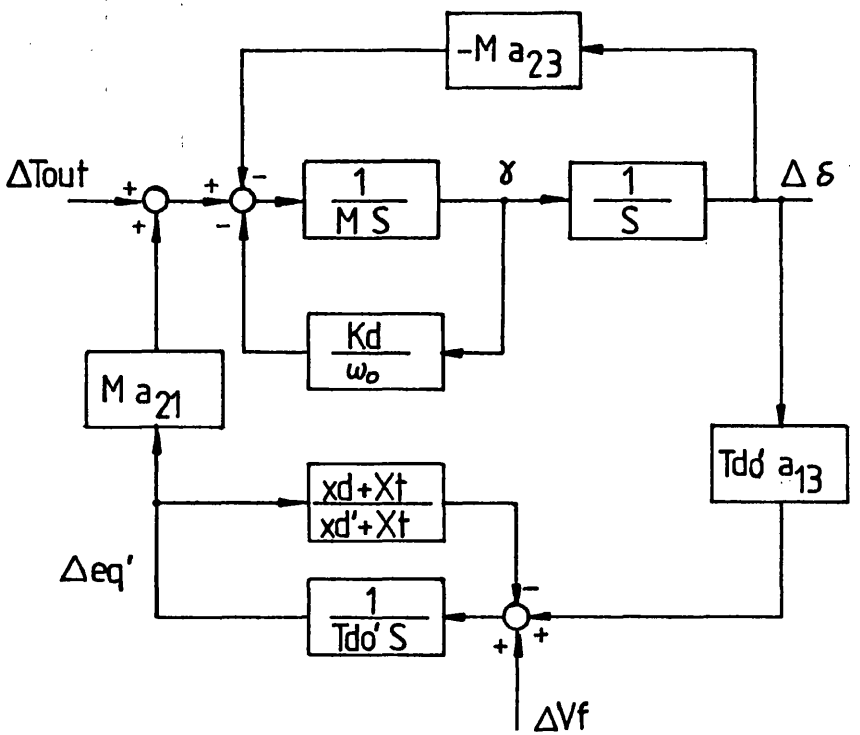


Fig 5.3 Linearised Synchronous Machine Representation

$$i_{do} = \frac{P}{V_b \sin\delta_o} - \frac{V_b \cos\delta_o}{(x_q + x_T)} \qquad (5.24)$$

and

$$e'_{qo} = V_b \cos\delta_o + (x_T + x'_d).i_{do} \qquad (5.25)$$

Equations (5.23) to (5.25) may be substituted into the expressions for the coefficient values a_{21} and a_{23}, in equation (5.15), so that:

$$a_{21} = \frac{-V_b \sin\delta_o}{M(x'_d + x_T)}$$

and

$$a_{23} = \frac{-1}{M}\left[\frac{V_b^2 \sin^2\delta_o (x_q - x'_d)}{(x'_d + x_T)(x_T + x_q)} + \frac{P}{\mathrm{Tan}\delta_o}\right]$$

It can be seen that the coefficients a_{13} and a_{21} both have the same sign, so that, in Fig. 5.3, the path from $\Delta\delta$ through the block $T'_{do} a_{13}$, the first order lag which generates $\Delta e'_q$, and the block $M\, a_{21}$ contributes to the positive damping torque and reduces the synchronising torque component.

Thus, the linearised system may be completely specified in terms of the machine and line reactances, the electrical power output and the operating load angle. The stability of the system may be assessed by considering the roots of the characteristic equation of the matrix equation (5.15). If the real part of any of these roots is positive then the system is unstable. This characteristic equation:

$$\lambda(\lambda-a_{11})(\lambda-a_{22}) - (\lambda-a_{11})a_{23}-a_{21}a_{13} = 0 \qquad (5.26)$$

may be written as:

$$\lambda^3 + b_2\lambda^2 + b_1\lambda^2 + b_0 = 0 \qquad (5.27)$$

where $b_2 = -(a_{11} + a_{22})$

$b_1 = (a_{11}a_{22} - a_{23})$.

and $b_0 = a_{11}a_{23} - a_{21}a_{13}$

In general, the roots of this equation will be complex so, putting $\lambda = A + jB$, expanding equation (5.27) and collecting real and imaginary components, gives:

$$(A^3-3AB^2) + b_2(A^2-B^2) + b_1(A) + b_0 = 0 \qquad (5.28)$$

and

$$(3A^2B-B^3) + b_2(2AB) + b_1(B) = 0 \qquad (5.29)$$

A stability boundary is reached when the real part of λ is zero. Thus, solving equation (5.29) for $A = 0$ gives:

$$B(b_1 - B^2) = 0 \qquad \text{i.e. } B = 0 \text{ or } B^2 = b_1 \qquad (5.30)$$

Equation (5.28) may now be solved for $A = 0$, $B = 0$ and $A = 0$, $B^2 = b$. Solving eqn. (5.28) for $A = 0$, $B = 0$, gives:

$$b_o = 0 \qquad (5.31)$$

and solving for $A = 0$ $B^2 = b_1$ gives:

$$b_o - b_2 b_1 = 0 \qquad (5.32)$$

Expanding b_o according to equations (5.27) and (5.15) gives:

$$b_o = \frac{(x_d + X_T)}{T_{do}'.\ M.(x_d'+x_T)} \left[\frac{V_b^2 \sin^2\delta_o (x_q - x_d)}{(x_d+x_T)(x_q+x_T)} + \frac{P}{\mathrm{Tan}\delta_o} \right] \qquad (5.33)$$

For a round rotor machine, $x_q = x_d$, so equation (5.31) requires that:

$$P/\mathrm{Tan}\delta_o = 0 \qquad (5.34)$$

which implies that either the power output at the stability boundary is zero, or that the load angle is 90 degrees. For a salient pole machine, $x_d > x_q$ in general, so that equation (5.31) requires that:

$$P = \frac{V_b^2 (x_d - x_q)}{(x_d+x_T)(x_q+x_T)} \quad [\mathrm{Tan}\delta_o - 1/2 \sin 2\delta_o] \qquad (5.35)$$

which gives the maximum possible power output for a given load angle. This equation may be rearranged to form a cubic equation in terms of $\mathrm{Tan}\delta_o$ as follows:

$$\frac{V_b^2 (x_d - x_q)}{(x_d+x_T)(x_q+x_T)} \mathrm{Tan}^3\delta_o - P.\mathrm{Tan}^2\delta_o - P = 0 \qquad (5.36)$$

which may be solved by iterative means. Solution of this equation,

for a given power, P, gives the maximum load angle that may be achieved without losing stability.

When excitation control of the form described in Chapter 3 is added, the linearised power system dynamic equations become fifth order. The linearised excitation system equations are:

$$p\Delta V_f = (K_1(\Delta Vref - \Delta V_s - \Delta V_t) - \Delta V_f)/T_e \qquad (5.37)$$

$$p\Delta V_s = (\frac{K_1K_2}{T_e}[\Delta V_{ref} - \Delta V_s - \Delta Vt] - \frac{K_2\Delta V_f}{T_e} - \Delta V_s)/T_s \qquad (5.38)$$

which may be used to express the power system equation in state space form as:

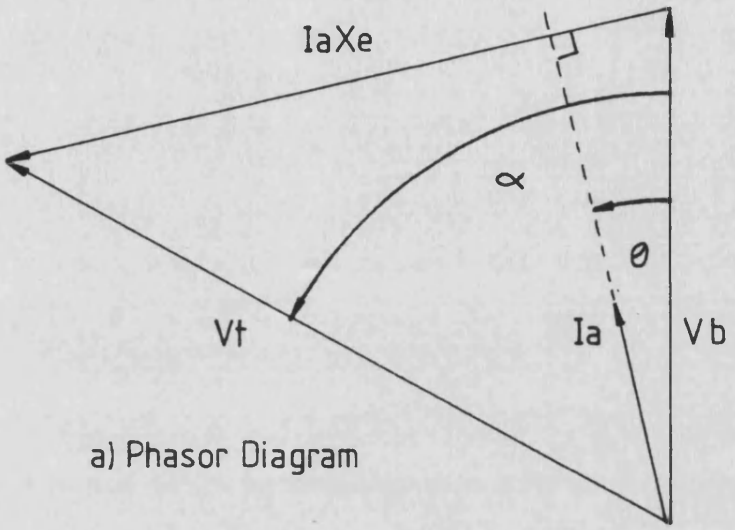
$$\begin{bmatrix} \dot{\Delta e}'_q \\ \dot{\gamma} \\ \dot{\Delta\delta} \\ \dot{\Delta V}_f \\ \dot{\Delta V}_s \end{bmatrix} = \left[\begin{array}{ccc|cc} a_{11} & 0 & a_{13} & 1/T'_{do} & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline a_{41} & 0 & a_{43} & a_{44} & a_{45} \\ a_{51} & 0 & a_{53} & a_{54} & a_{55} \end{array}\right] \begin{bmatrix} \Delta e'_q \\ \gamma \\ \Delta\delta \\ \Delta V_f \\ \Delta V_s \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1/M \\ 0 & 0 \\ K_1/T_e & 0 \\ \frac{K_1K_2}{T_eT_s} & 0 \end{bmatrix} \begin{bmatrix} \Delta V_{ref} \\ \Delta T_{out} \end{bmatrix} \qquad (5.39)$$

Consideration can again be given to the conditions necessary for the roots of the characteristic equation to have zero real parts, and the dyanmic stability limit obtained, taking into account the dynamics of the excitation system. The effect of the governor system and prime mover may be included in a similar manner, with a seventh order representation of the system.

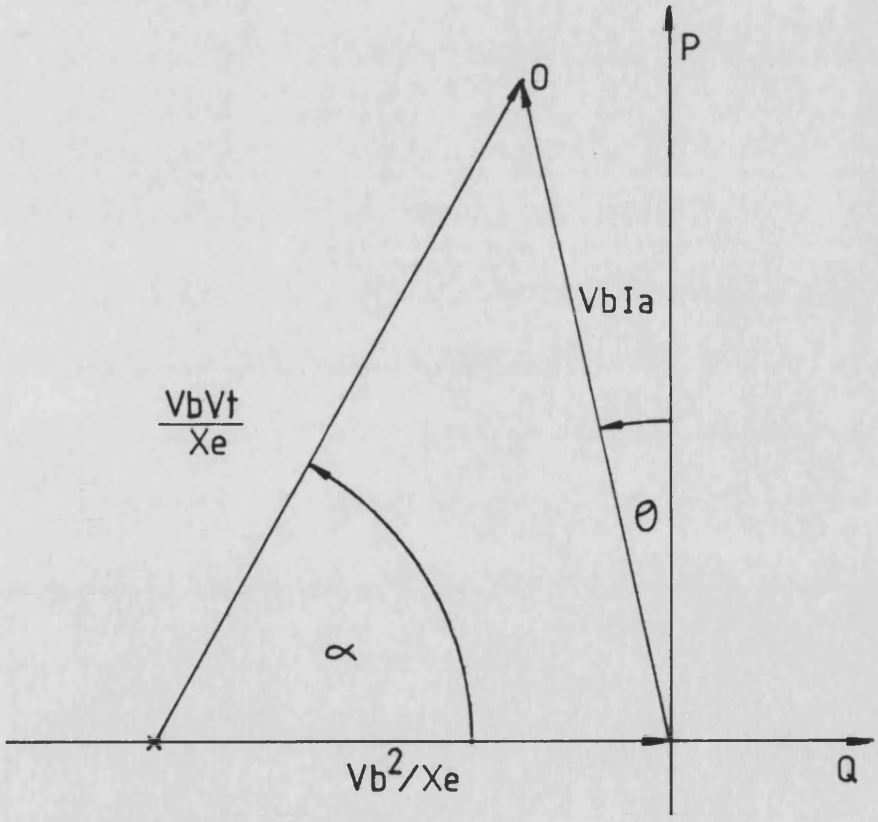
The real-time simulation described in the previous Chapter is not based on a linearised representation of the power system. The dynamic stability limit cannot be obtained from the real-time simulation by using the state space method described above. When operating under steady state conditions, the electrical power output to the infinite busbar, assuming negligible losses in the machine and transmission line resistances and in mechanical friction, is approximately equal to the prime mover output power. The prime mover output power is controlled by the governor system and, at steady state, is proportional to the governor reference. Thus, on an operating chart, such as that of Fig. 5.1(c), the governor reference directly controls the vertical displacement of the operating point, O. Note that only in the case of a round rotor machine with zero resistance, can the phasor diagram be drawn onto the operating chart, as shown in Fig. 5.1. In other cases, with a salient rotor, or armature and transmission line resistance, the operating point is no longer coincident with the scaled excitation voltage vector, as shown. However, in all cases, round rotor or salient rotor, assuming small losses, control of the prime mover output power, by the governor reference, controls the operating power. With the output power fixed by the governor reference, the reactive power output may be controlled by controlling the excitation voltage. A reduction of the excitation voltage magnitude requires an increase in load angle to maintain the power balance. In the case of a round rotor machine, it is clear from Fig. 5.1(c) that this will shift the operating point further towards, or into, the leading power factor region, i.e. further to the left of Fig. 5.1(c). However, in the other cases, a change in excitation will result in a change of operating point and, since the power output is approximately constant, this change of

operating point must be in the direction of the reactive power, Q, axis. With an automatic excitation control system, the machine terminal voltage is kept nominally constant. Fig. 5.4 shows a voltage phasor diagram and operating chart, drawn by consideration of the conditions at the machine terminals. Thus, assuming that steady state operation is achieved, control of the terminal voltage by the use of automatic voltage regulators may be used to control the reactive power output. The operating point, O, can therefore be moved vertically by changing the governor reference and horizontally by changing the excitation system reference. The approach adopted by the PQ command of the monitor task is to establish the dynamic stability limit by shifting the operating point of the system being simulated until the system loses stability.

On entry to the PQ action routine, a prompt for a filename is given. The specified file is opened for output and the limiting real and reactive power values are written to this file prior to the completion of the action routine. The routine then calls the 'remember' subroutine so that the simulation makes a record of its current state. The excitation system reference is then decremented by an arbitrary amount and sent by a call to the 'send' subroutine. After a call to 'waitbuffer', which allows the data collection buffer to fill, a call to 'graph' plots a transient response (load angle is perhaps the most useful variable to observe). A call to a BCPL subroutine, 'left search' is made to determine whether the system achieves steady state operation or becomes unstable. This subroutine returns a 'true' result if steady state is achieved and the dynamic stability limit is therefore to the left of the current operating point on the operating chart. If a new, stable, operating point is



a) Phasor Diagram



b) Operating Chart

Fig 5.4 Controlled Excitation Operation

achieved, the size of the excitation reference decrement is doubled and this whole process repeated until a change in operating point brings about instability. The last stable operating point is then restored by a call to the 'restore' subroutine and the excitation reference decrement is halved. The dynamic stability limit is then approached by successively smaller changes in the excitation reference until the decrement necessary to bring about instability is very small (less than 0.4%). The last stable operating point is then restored and the corresponding values of real and reactive power at the infinite busbar are recorded in a temporary data buffer. The governor reference is then decremented by one-tenth of its initial value, to shift the operating point down on the operating chart. The 'left search' subroutine is then called to determine whether the system is still stable. If the system is still stable, the search for the reactive power limit at this new real power level is made in the same manner as described above. Otherwise, the last stable operating point is restored, the excitation system reference is incremented by successively larger increments if necessary, and the change in governor reference repeated (since the last stable operating point used the previous governor reference setting). This progressively shifts the operating point to the right, further away from the reactive power limit. Once stable operation at the new power level is obtained, the search for the reactive power limit is made in the same manner as before. The governor reference is decremented a further eight times to obtain further points on the dynamic stability limit curve. Finally, the temporary data buffer is written to the open file, in a plottable format and the action routine then closes the file and exits. The 'left search' subroutine determines whether the system is stable by observing the load angle response. First, it

calls the subroutine 'poleslip' which looks for a load angle response which exceeds a load angle of 180°. If 'poleslip' returns 'true' then 'left search' restores the last stable operating conditions and returns a 'false' value to the calling routine. If 'poleslip' returns 'false', 'left search' then assumes that the load angle response is oscillatory and searches for a maximum in the recorded load angle transient. If it fails to find a maximum it ensures that the range of the load angle response is small enough for the resonse to be considered to have settled. If a maximum has been found, a second maximum is sought, so that it is possible to determine whether the response is damping out. If a second maximum is not found, the range of the load angle response is again checked to see whether the response may be considered to have settled. If the range of the response is too large, then the buffers are refilled by a call to 'replenish' and this inspection repeated on the new data. The system is deemed to be unstable in the event that successive maximums in the load angle response occur at larger load angle values or the system 'poleslips'. The system is deemed to be stable if the range of load angle values is small, or successive maximums in the load angle response occur at smaller load angle values. If the system is deemed to be stable, 'left search' makes a call to the 'remember' subroutine prior to returning a 'true' value. Prior to returning a 'false' value, 'left search' makes a call to 'restore', thus all the remembering and restoring of the system state occurs transparently to the main action routine, within 'left search'.

Figs. 5.5 and 5.6 show dynamic stability limit curves obtained by the use of the PQ command of the Monitor task. Fig. 5.5 shows the curve obtained for the turbogenerator/infinite busbar system shown

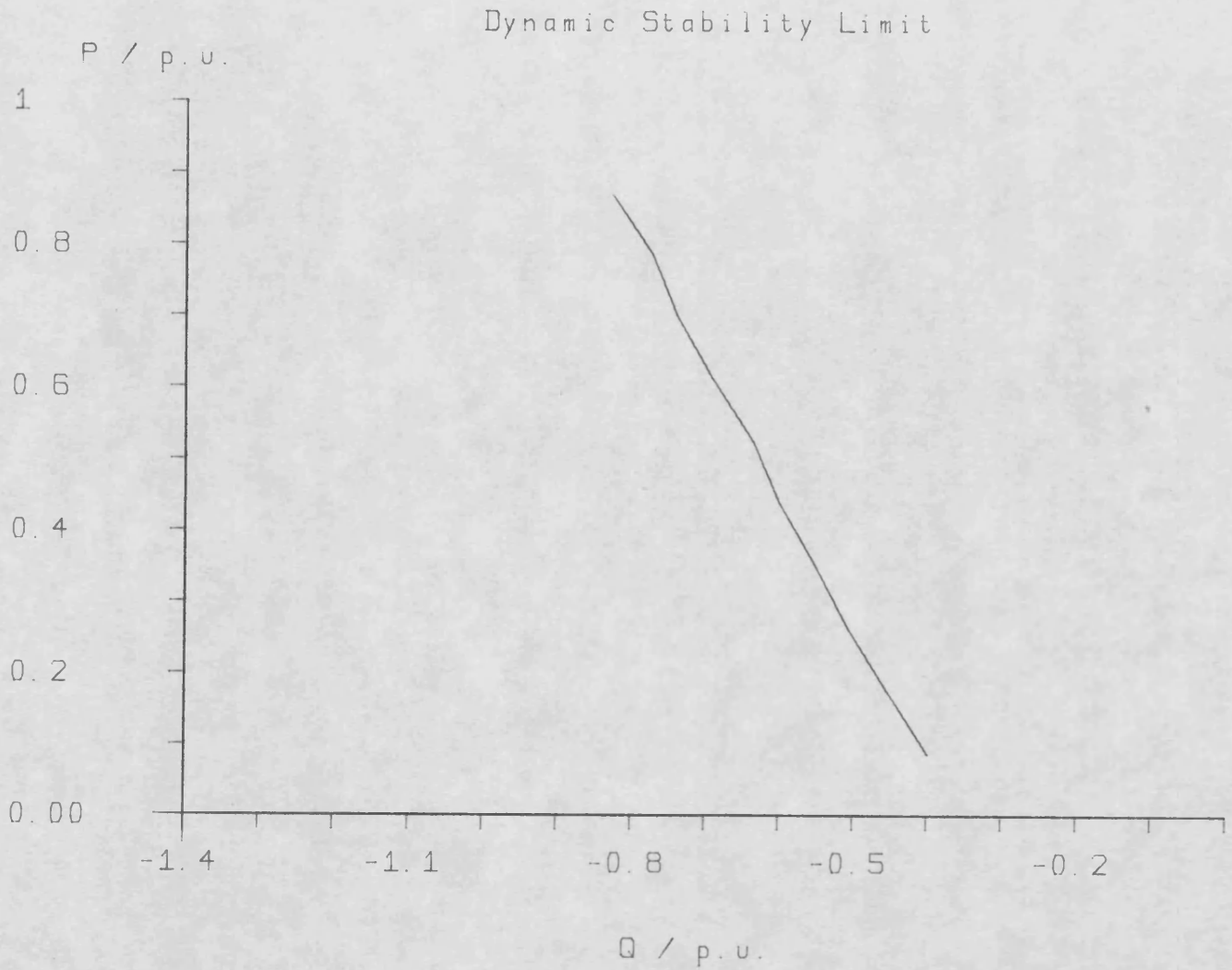


Fig 5.5

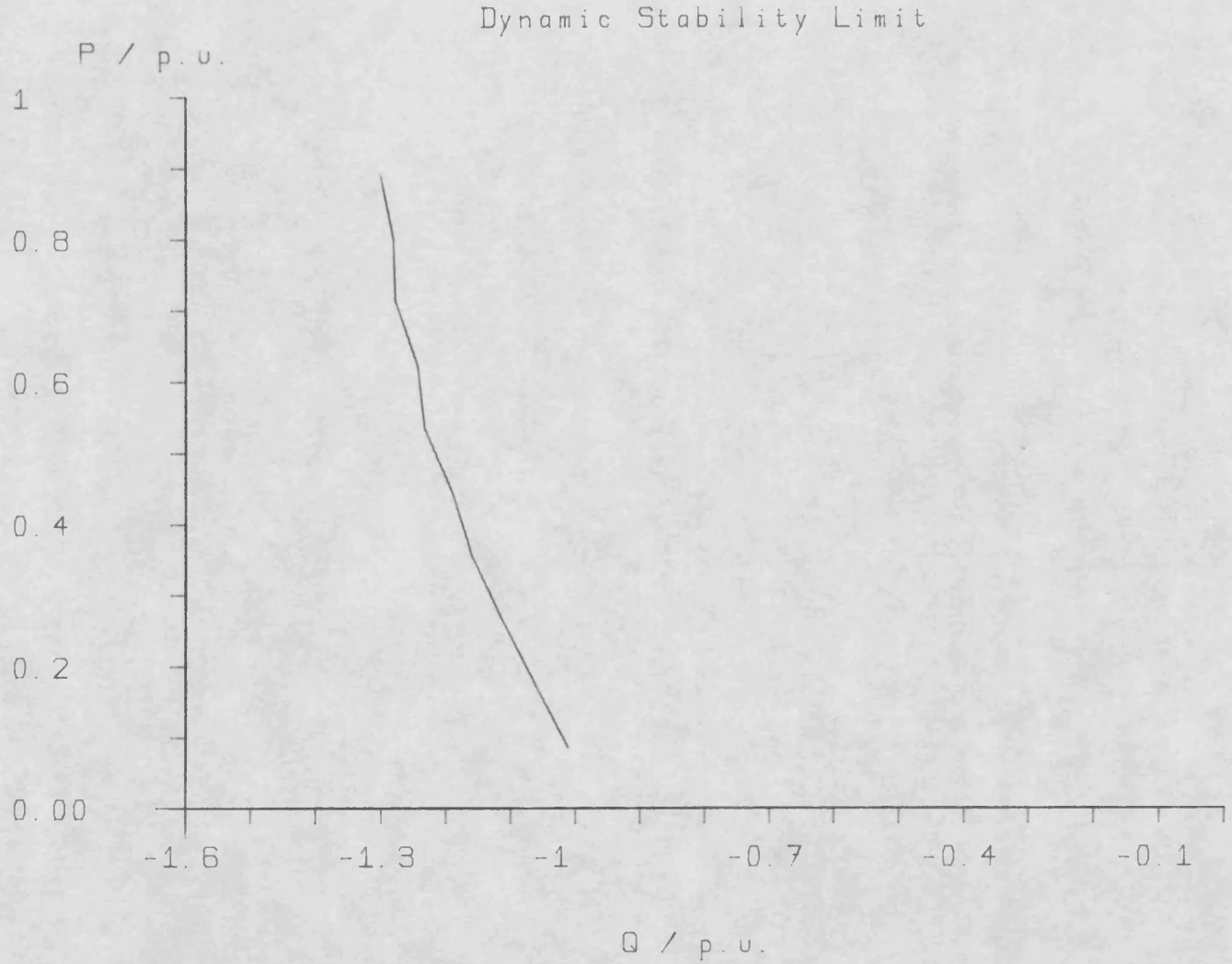

Dynamic Stability Limit
P / p.u.
1
0.8
0.6
0.4
0.2
0.00
-1.6
-1.3
-1
-0.7
-0.4
-0.1
Q / p.u.

Fig 5.6

in Fig. 6.8, while Fig. 5.6 shows the curve obtained for the hydrogenerator/infinite busbar system, also described by Fig. 6.8.

In the case of a salient pole machine with:

$$P = \frac{V_b V_f \sin\delta}{(x_d+x_e)} + \frac{V_b^2(x_d-x_q)}{2(x_d+x_e)(x_q+x_e)} \sin 2\delta \qquad (5.40)$$

$$Q = \frac{V_b V_f \cos\delta - V_b^2}{(x_d+x_e)} + \frac{V_b^2(x_d-x_q)}{2(x_d+x_e)(x_q+x_e)} [\cos 2\delta - 1] \qquad (5.41)$$

it can be shown that the dynamic stability limit intercepts the reactive power axis at $Q = -V_b^2/(x_q+x_T)$. Using the values for the machine, transformer and transmission line reactances given in Appendix and a busbar voltage of unity, this gives intercepts of $Q = -0.334$ p.u. and $Q = -0.978$ p.u. for the turbogenerator and hydrogenerator respectively. These values assume that machine and transmission line resistances are negligible. Extrapolation of the curves in Figs. 5.5 and 5.6 across the reactive power axis would pass close to the values given above.

Further dynamic stability limit curves are given in Chapter 6 to illustrate the change in dynamic stability obtained when the transient performance has been optimised.

5.3 Transient Stability

The transient stability limit of a generating plant is defined in Chapter 1 as the maximum electrical power that it may export, without

losing synchronism, when subjected to a large, sudden, disturbance. This stability limit is not only dependent on the machine operating conditions, but also on the type of disturbance used. The steady state torque versus load angle characteristic for a synchronous machine with negligible armature resistance and a purely reactive transmission system, is identical to the power versus load angle characteristic, scaled by the synchronous speed since $P_e = T_{elec} \cdot \omega$. Figs. 5.7(a) and (b) show the steady state torque versus load angle characteristics for round rotor and salient pole generators. The amplitudes of these two characteristics are dependent on the magnitude of the busbar and field voltages. The shape of the characteristic in Fig. 5.7(b) is also dependent on the relative magnitudes of the field and busbar voltages. Fig. 5.5(c) demonstrates the use of an equal area criterion for determining whether stability will be lost by a change in the mechanical input torque. When the mechanical input torque is increased from $T = T_o$ to $T = T_1$ the load angle increases from δ_o to δ_1. In order to bring about this change in load angle, the rotor is subjected to an accelerating torque of $T_{acc} = T_1 - T_m \sin\delta$. At the point in the transient when the load angle reaches $\delta = \delta_1$, the extra energy stored in the increased speed of the rotor, Δw_1, is given by:

$$\Delta w_1 = \int_{\delta_o}^{\delta_1} (T_1 - T_m \sin\delta) d\delta \qquad (5.42a)$$

This increase in stored energy is represented by the shaded area A_1 in Fig. 5.7(c). In order to reach steady state this excess stored energy must be lost, so that the machine can return to synchronous

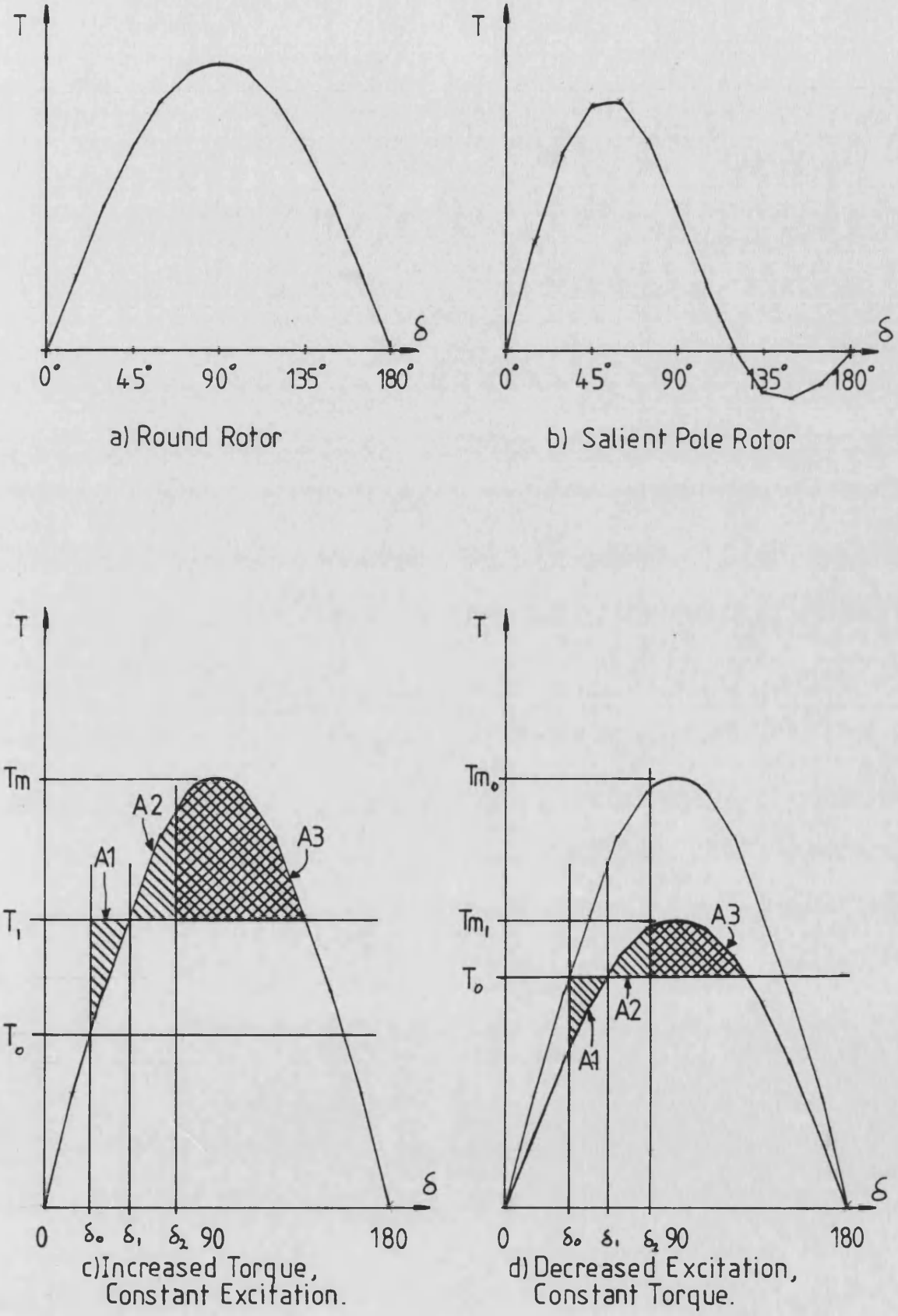


Fig 5.7 Torque, Load Angle Characteristics

operation. Once the load angle has increased above $\delta = \delta_1$ the sign of the accerlating torque is reversed, so braking the rotor. The load angle increases to $\delta = \delta_2$ where the increase in stored energy, Δw_2, is zero, i.e.:

$$\Delta w_2 = \int_{\delta_0}^{\delta_2} (T_1 - T_m \sin\delta) d\delta = 0 \qquad (5.42b)$$

It can be seen from Fig. 5.5(c) that, for Δw_2 to equal zero, the two areas, A_1 and A_2, must be equal. The total area, $A_2 + A_3$, indicates the amount of energy that can be removed from the rotor during this braking phase. If the area A_1 is greater than the area $A_2 + A_3$, then the machine has insufficient capacity to brake the rotor and synchronism is lost. Provided that there is some damping in the system, after reaching a load angle of $\delta = \delta_2$, the load angle will undergo a damped oscillation about $\delta = \delta_1$. If no damping is present, the load angle will continue to oscillate between $\delta = \delta_0$ and $\delta = \delta_2$. Similar considerations may be applied to the case where the machine is being driven by a constant mechanical torque, T_0, and the excitation is reduced. Fig. 5.7(d) shows this situation, where the reduction in excitation reduces the peak in the load angle characteristic from $T = T_{m0}$ to $T = T_{m1}$. When the excitation is reduced, the load angle, $\delta = \delta_0$, cannot advance instantaneously to $\delta = \delta_1$. On the torque versus load angle characteristic, the electrical torque at $\delta = \delta_0$ is less than the electrical torque at $\delta = \delta_0$ on the original characteristic. The machine is therefore subjected to an accelerating torque, $T_{acc} = T_0 - T_{m1}\sin\delta$. The same constraints on the areas A1, A2 and A3 apply, as in the case of an

increase in the mechanical input torque. This equal area approach may be applied to simple transient changes, where either the field voltage or the prime mover output torque is stepped from one steady value to another steady value, and the system allowed to settle into a new steady state in those cases where stability is maintained.

In the case where a three-phase short circuit fault, applied for a standard duration, at the high voltage terminals of the generator transformer, is used as the disturbance, and the generator is fitted with an automatic excitation control system, the equal area criterion cannot be used. In such cases the only approach left open to the system planner is to apply the disturbance used to define the transient stability limit to a full nonlinear simulation of the system and then determine whether the resulting transient is stable. The Real-Time simulator is ideal for this type of approach, since the disturbance may be applied, the data collected and the load angle transient examined very rapidly.

A wide range of system parameters affects the transient stability limit of a generator. Stability can be lost even in the dynamically stable, lagging power factor region of the operating chart. It is not very convenient to plot the transient stability limit as a locus on the operating chart, primarily because a point on such a chart assumes steady state operation. It would be possible to plot a locus of predisturbance operating points which are 'just' stable when the disturbance is applied. However, it is unlikely that useful comparisons of changes in the transient stability limit could be made from such a chart. Critical fault clearance time charts have been popular for making comparisons in the transient stability of

generating plants subjected to severe disturbances. The critical fault clearance time, i.e. the maximum possible fault clearance time which maintains stability, is plotted against the transmission line impedance, or the faulted infeed current (the current supplied to the fault by the rest of the system). The transmission line impedance is varied and a locus of critical fault clearance times plotted. Loci may be plotted for different control strategies and those which result in higher values of critical fault clearance time are deemed to give the best transient stability improvements. Quantative comparisons can then be made based on percentage improvements in the critical fault clearance times.

Pre-fault/post-fault impedance plots (5-2,5-3) convey more information than the critical fault clearance time plots. As the impedance between the generator terminals and the rest of the power system (the infinite busbar in this case) rises, the coupling between the generator and the power system becomes poorer. In order to export the same amount of power to the power system through higher transmission impedances either the generator excitation must be increased or the generator must operate at larger load angles. So, as the transmission impedance rises, it becomes increasingly difficult to transfer power between the generator and the busbar. Transient stability is impaired if the power flow becomes too restricted since, in order to regain synchronism following a fault, the energy stored in the accelerated rotor shaft, must be returned to the electrical system, i.e. transmitted to the busbar, before the machine poleslips. Thus, following a fault disturbance, there is a maximum or critical post-fault impedance at which synchronism is only just maintained. In general, the impedance between the generator and the rest of the

system increases following a fault. For example, if the fault occurs on one line out of a double circuit transmission system, the fault clearance equipment disconnects that line from the transmission system, and the post-fault impedance is double the pre-fault impedance. The magnitude of the critical post-fault impedance is therefore a measure of the transient stability of the generating plant. Generating plant with increased transient stability can withstand higher post-fault impedance values. One practical interpretation that may be placed on the pre-fault, post-fault impedance plot is that it gives an indication of the maximum length of the transmission system, since the line impedance is proportional to the line length. Thus control strategies which enable high post-fault impedances to be used allow generating plant to be sited more remotely from the rest of the power system. This type of consideration is of increasing importance as environmental pressures increasingly influence the siting of a power station.

A Monitor task command which generates pre-fault/post-fault impedance plot has been implemented. When the 'impedance' command is issued the Monitor task enters the 'impedance' action routine. The action routine first prompts for a filename in which to save the impedance plot and then prompts for the fault duration in milliseconds. It is assumed that, when the command is involked, the parameters in the data base for the faulted and post fault impedances give resistance and reactance values for the generator transformer alone and the generator transformer in series with the transmission line, respectively. The transmission line resistance and reactance are then calculated from the difference between the post fault and the faulted impedances. The action routine then normalises the

transmission line resistance and reactance so that the normalised transmission line impedance is unity. Subsequent changes in the pre-fault and post-fault reactance and resistance are made in proportion to those of the normalised transmission line impedance so that the changes correspond to lengthening or shortening of the transmission line. A loop counter is then zeroed and a repetitive loop is entered. Inside the loop the pre-fault transmission line resistance and reactance are calculated by multiplying the normalised transmission line resistance and reactance by the loop counter divided by twenty. Thus, each time around the loop the pre-fault impedance is increased by 0.05 p.u. The subroutine 'post-fault impedance' is then called. This subroutine required five parameters, the pre-fault transmission line resistance, the pre-fault transmission line reactance, the fault duration, the normalised transmission line resistance and reactance. This subroutine returns the critical values of the post-fault resistance and reactance, where stability is only just maintained. These post-fault values are stored as an impedance magnitude, along with the pre-fault impedance magnitude, in a data buffer. The loop counter is then incremented and the program returns to the beginning of the loop unless the post-fault impedance is less than the pre-fault impedance or the data buffer is full. Thus, the repetitive loop generates the trajectory of the post-fault/prefault impedance curve up to the point where the trajectory crosses into the region where the critical post-fault impedance is less than the pre-fault impedance. It is unrealistic for the post-fault impedance to be less than the pre-fault impedance as this would require a stronger transmission system following trhe fault than that prior to the fault. On exiting the repetitive loop, the data buffer is written to the file opened when the action routine

was entered, in a format which may be used by the plotting utilities described in Chapter 4. The file is then closed and the action routine exits to the main loop of the Monitor task.

The subroutine 'post-fault impedance' maintains three impedance magnitude values; 'minzt' is used to store a post-fault impedance value which is known to be stable; 'maxzt' is used to store a post-fault impedance value which is known to make the system unstable and 'midzt', which is an impedance midway between 'minzt' and 'maxzt'. The subroutine first clears 'minzt' to zero and sets 'maxzt' to an improbably high value (100 p.u.). It then makes a local copy of the Monitor task data base so that it may update the transmission line impedances and restore the original values before the subroutine exits. The data base post-fault impedance values are then modified to the sum of the pre-fault transmission line impedance, passed in the subroutine call and the transformer impedance, i.e. the faulted impedance values in the data base. These values are then sent by a call to the 'send' subroutine, and a long delay is introduced so that the pre-fault steady-state operating conditions may be established. The subroutine 'remember' is then called, so that the simulation tasks make an internal copy of their states and the pre-fault conditions may be restored by subsequent calls to the 'restore' subroutine. The data collection buffers are filled by a call to 'replenish' and a call to the 'poleslip' tests the stability of the pre-fault operating point. If the pre-fault operating point is unstable, then subroutine 'post-fault impedance' sets an error flag which causes the action routine to exit from its repetitive loop and returns to the action routine with a post-fault impedance of zero, having first restored the Monitor task data base.

If the pre-fault operating point is stable, the search for the critical value of the post-fault impedance is continued by entering a repetitive loop which establishes better values for 'minzt' and 'maxzt' than those assigned to them when the subroutine was entered. First, a fault of the specified duration is applied to the system by a call to the subroutine 'fault' and a call to 'waitbuffer' pauses execution of the subroutine for six seconds while the data collection buffer fills. A call to 'poleslip' then determins whether stability has been lost during the first six seconds of the transient. If stability has been lost, 'maxzt' is updated to the value of the post-fault transmission line impedance, i.e. the post-fault impedance minus the transformer impedance and the loop is broken out of. Otherwise, 'minzt' is updated with the value of the post-fault transmission line impedance and the post-fault transmission line impedance is then doubled. The pre-fault operating conditions are restored by a call to 'restore' and execution returns to the start of the repetitive loop. Thus, this loop starts with the post-fault and pre-fault impedances equal, and repeatedly doubles the post-fault impedance until stability is lost following the application of the disturbance. On exit from this loop, 'maxzt' contains a post-fault impedance value which brings about instability and 'minzt' contains a post-fault impedance value which is stable when the fault is applied. The subroutine then enters another repetitive loop which performs a binary search for the critical post-fault impedance value. The value of 'midzt' is set halfway between 'maxzt' and 'minzt'. The post-fault transmission line impedance is set to the value of 'minzt' by multiplying the normalised transmission line resistance and reactance by the value of 'minzt' and it is then added to the transformer resistance and reactance to form the post-fault impedance

stored in the data base. The prefault operating conditions are then restored and the fault applied for the specified duration. Once six seconds' worth of data have been accumulated in the data collection buffer, 'poleslip' is called to determine whether the system is stable with this particular value of post-fault impedance. If the system is stable, 'minzt' is updated with the value of 'midzt', otherwise 'maxzt' is updated with the value of 'midzt'. If the difference between 'maxzt' and 'minzt' is less than a small value (0.001 p.u.) the loop exits. Otherwise, the loop is repeated and the difference between 'maxzt' and 'minzt' becomes smaller at each iteration. Once the binary search loop exits, the Monitor task data base is restored from the local copy of the data base and the limiting values of the transmission line resistance and reactance are returned to the calling routine as the subroutine exits.

Figs. 5.8 to 5.12 show pre-fault/post-fault impedance plots made using the 'impedance' command of the Monitor task. Figs. 5.8 to 5.10 were made using the turbogenerator data given in Appendix A, while Figs. 5.11 and 5.12 were made using the hydrogenerator data, also given in Appendix A. Fig. 5.8 shows curves obtained for three different fault clearance times, 220ms in black, 140ms in red and 80ms in green. Consider the critical post-fault impedances, Z_o, for a pre-fault impedance, Z_i, of zero in these three cases. A decrease in fault clearance time from 220ms to 140ms results in an increase in the critical post-fault impedance from 0.352 p.u. to 0.667 p.u., i.e. an increase in the transient stability limit of 89.5%. Likewise, by considering the critical post-fault impedance values for the situation where the pre-fault and post-fault impedances are equal, an increase from 0.214 p.u. to 0.542 p.u. is obtained, i.e. an increase

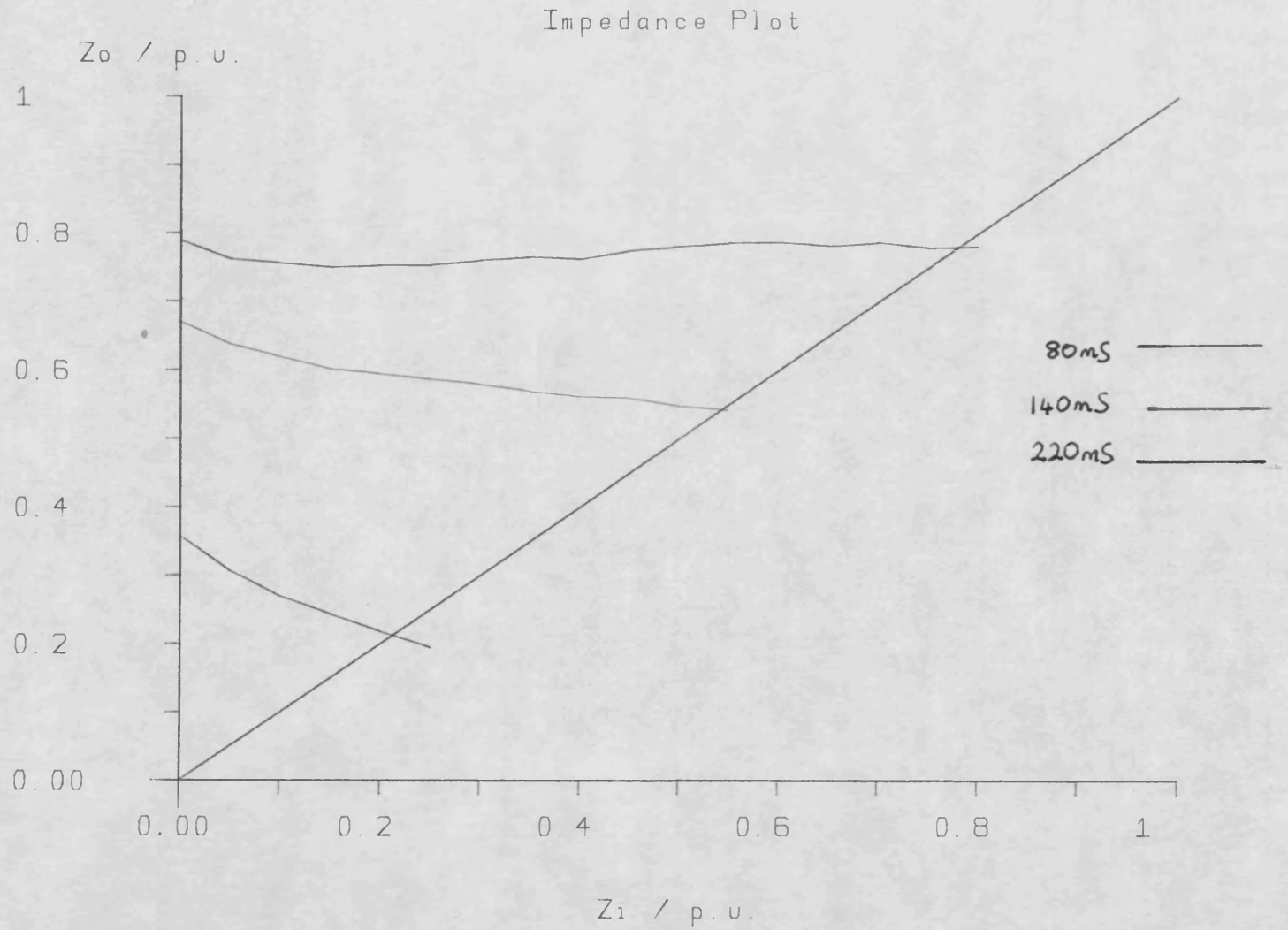

Impedance Plot
Zo / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
0.2
0.4
0.6
0.8
1
Zi / p.u.
80mS
140mS
220mS

Fig 5.8

in the transient stability limit of 153%. These percentage increases in transient stability values can be interpreted as meaning that between 89.5% and 153% extra plant of the same type can be installed at the same site, using the same transmission system or, alternatively, an increase of between 89.5% and 153% in the transmission line impedance can be made by reducing the fault clearance time from 220ms to 80ms. Likewise, a further decrease in the fault clearance time from 140ms to 80ms gives an increase in transient stability of between 18.4%, for a pre-fault impedance of zero, to 43.9% when the pre-fault and post-fault impedances are equal.

Fig. 5.9 shows pre-fault/post-fault impedance curves obtained by reducing the field voltage ceiling from 6.87 p.u. to 4 p.u. Four curves are drawn: two with a fault clearance time of 220ms, in black and green - the black curve having a field voltage ceiling of 6.87 p.u. and the green curve having a field voltage ceiling of 4 p.u. The red and blue curves have a fault clearance time of 140ms, the red curve has a field voltage ceiling of 6.87 p.u. and the blue curve has a field voltage ceiling of 4 p.u. By considering the impedances as above, with a fault clearance time of 220ms, an increase in field voltage ceiling from 4 p.u. to 6.87 p.u. gives an increase in transient stability of between 23.5% and 26.6%. At a fault clearance time of 140ms, the percentage improvement in transient stability limit for the same increase in field voltage ceiling is between 17.6% and 15.3%. Also, with a field voltage ceiling of 4 p.u., a decrease in fault clearance time from 220ms to 140ms gives a transient stability improvement of between 99.0% and 178%.

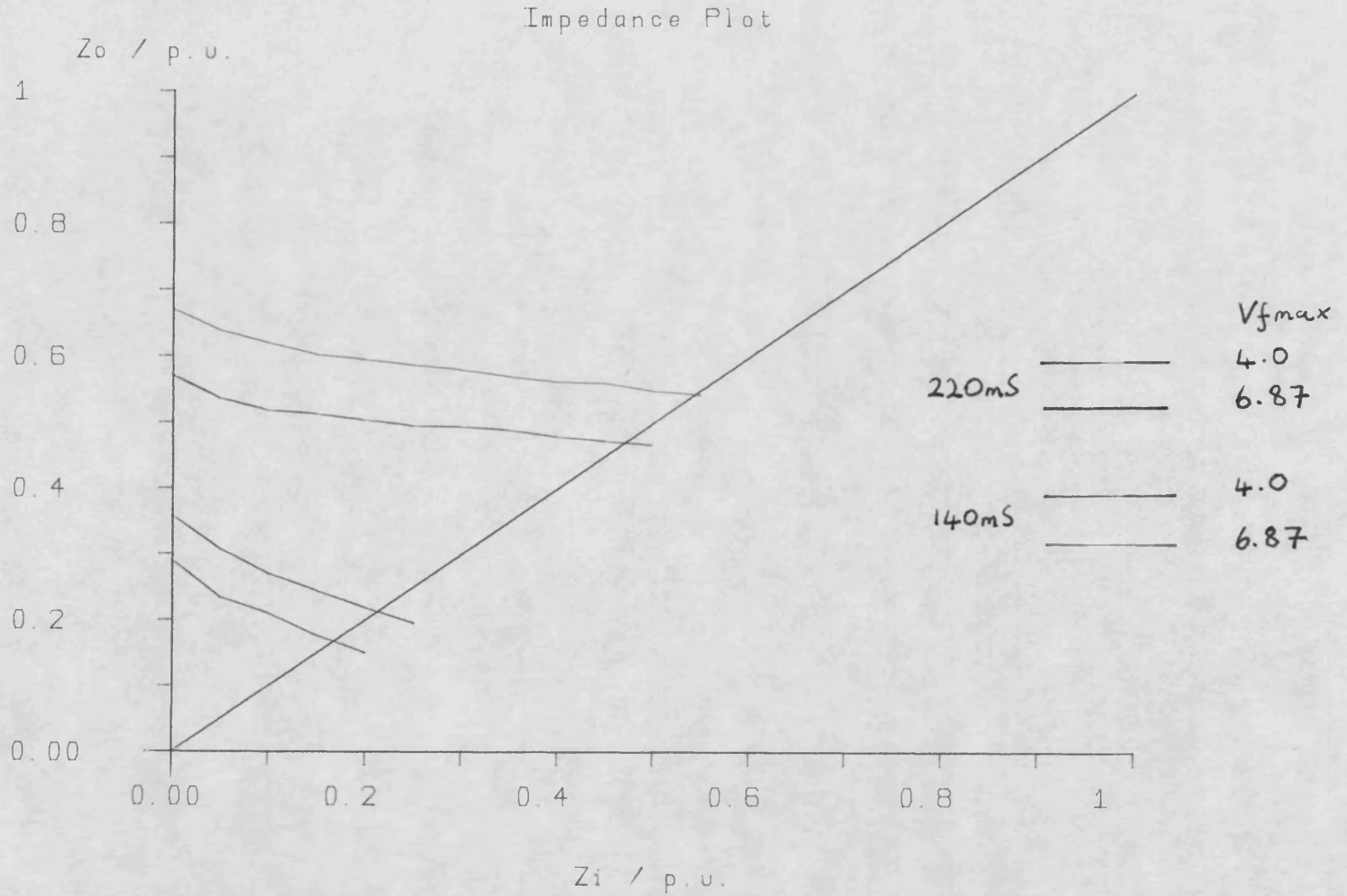

Impedance Plot
Zo / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
0.2
0.4
0.6
0.8
1
Zi / p.u.
Vfmax
220mS
4.0
6.87
140mS
4.0
6.87

Fig 5.9

Fig. 5.10 shows the effect on the transient stability limit of reducing the governor valve opening and closing rates. The black and red curves are the same as those in Fig. 5.9, with a valve opening rate of 4 p.u. per second and a valve closing rate of 6.6 p.u. per second and fault clearance times of 220ms in black and 140ms in red. The green and blue curves have valve opening and closing rates of 0.8 p.u. per second and 1.32 p.u. per second respectively, i.e. one-fifth of the rates used for the black and red curves. The green curve uses a fault clearance time of 220ms while the blue curve uses a fault clearance time of 140ms. With a fault clearance time of 220ms, an increase in valve rates from 0.8 and 1.32 p.u. per second to 4.0 and 6.6 p.u. per second gives an increase in transient stability limit of between 5.70% and 6.47%. Likewise, with a fault clearance time of 140ms the increase in transient stability limit for the same change in governor valve rates is between 2.93% and 1.88%. Also, at the reduced governor valve rates, a reduction in the fault clearance time from 220ms to 140ms gives a transient stability improvement of between 94.6% and 165%.

Fig. 5.11 shows the effect on the transient stability of the hydrogenerator system when the fault clearance time is reduced from 220ms to 140ms and then from 140ms to 80ms. The curves in Fig. 5.11 are colour coded as those in Fig. 5.8, i.,e. black for a 220ms fault clearance time, red for a 140ms fault clearance time and green for an 80ms fault clearance time. A decrease in fault clearance time from 220ms to 140ms gives in increase in the transient stability limit of between 40.1% and 66.3%. Likewise, a further decrease in the fault clearance time gives a transient stability improvement of between 10.6% and 26.3%.

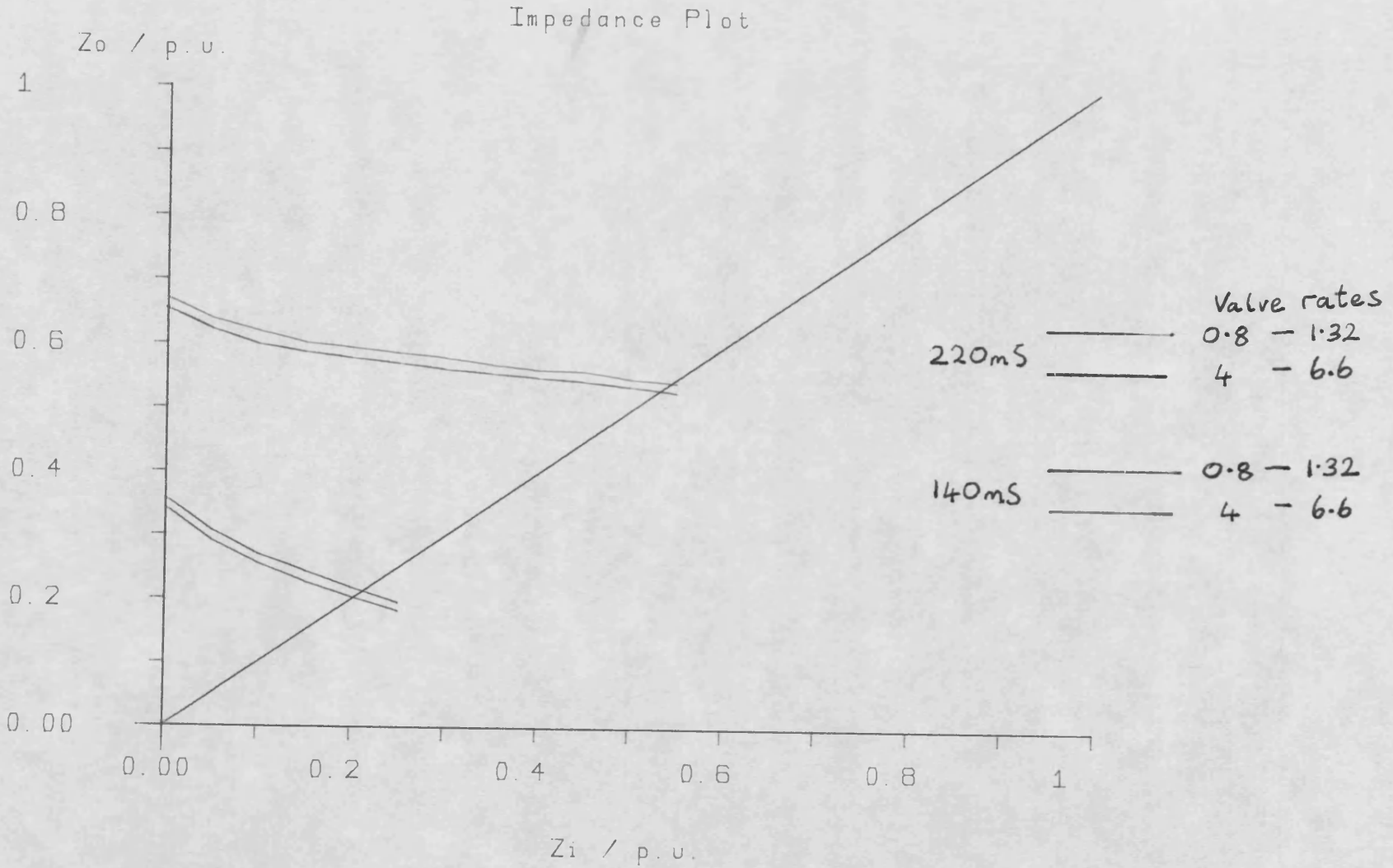

Impedance Plot
Zo / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
0.2
0.4
0.6
0.8
1
Zi / p.u.
Valve rates
220mS
0·8 – 1·32
4 – 6.6
140mS
0·8 – 1·32
4 – 6.6

Fig 5.10

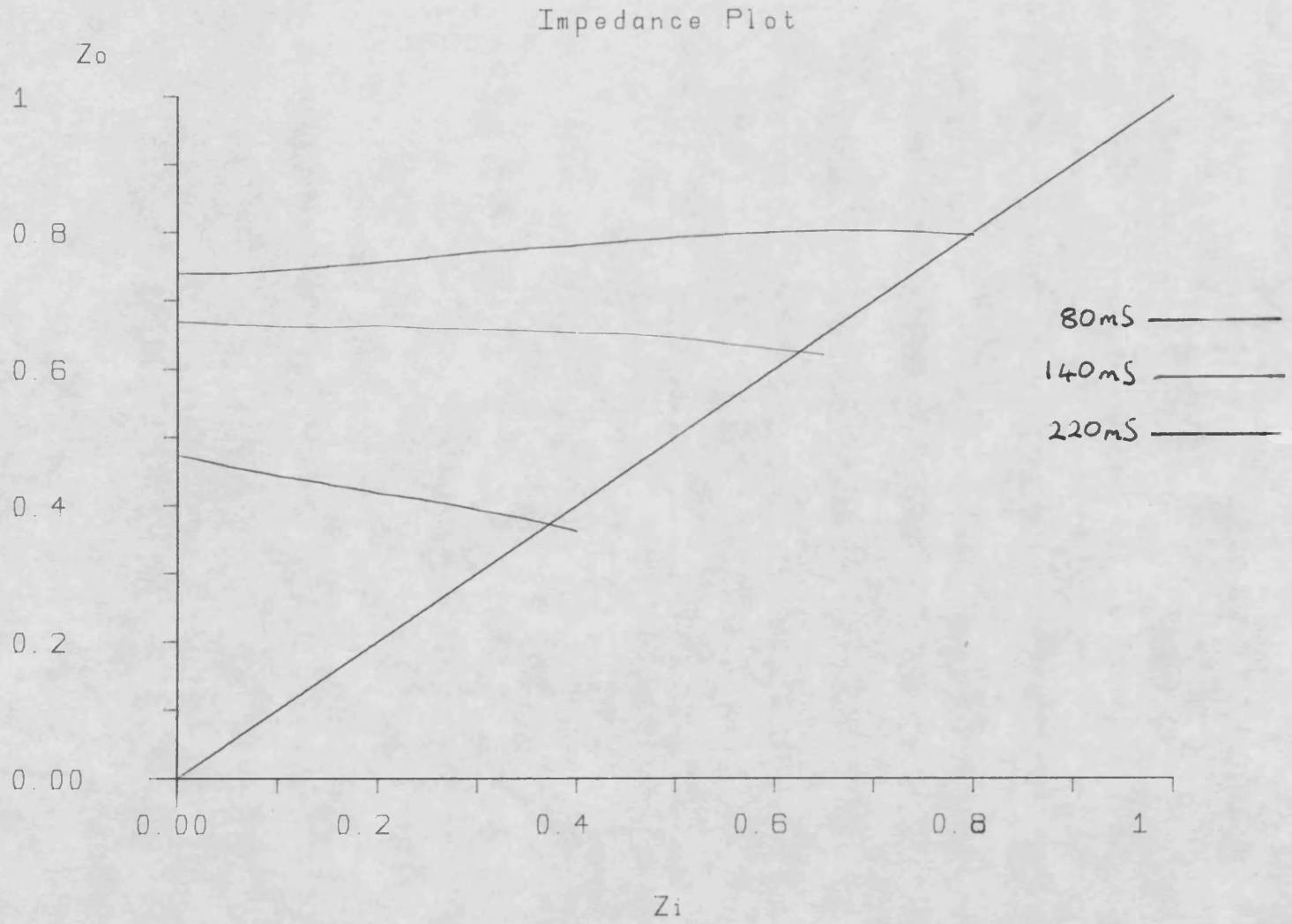

Impedance Plot
Zo
1
0.8
0.6
0.4
0.2
0.00
0.00
0.2
0.4
0.6
0.8
1
Zi
80mS
140mS
220mS

Fig 5.11

Fig. 5.12 is colour coded in the same manner as Fig. 5.9 and shows the effect on the transient stability limit of the hydrogenerator of a reduction in the field voltage ceiling from 6.87 p.u. to 4 p.u. The green and black curves show the effect of this reduction for a fault clearance time of 220ms, while the blue and red curves show this effect for a fault clearance time of 140ms. An increase in field voltage ceiling from 4 p.u. to 6.87 p.u., with a fault clearance time of 220ms gives an increase of between 7.69% and 6.50% in the transient stability limit, while the same change in field voltage ceiling with a fault clearance time of 140ms gives in increase in transient stability limit of between 4.55% and 3.81%. Also, with a field voltage ceiling of 4 p.u., a decrease in fault clearance time from 220ms to 140ms gives an increase in transient stability of between 44.3% and 70.6%.

It is clear from Figs. 5.8 to 5.12 that the most dramatic improvements in transient stability are achieved by a reduction in the fault clearance time. Quite significant improvements in transient stability can be obtained by increasing the field voltage ceiling. This action permits more power to be transferred to the electrical system during the fault and immediately following the fault while the field voltage remains in saturation due to the terminal voltage error. Variation of the governor valve opening and closing rate limits appears to have very little effect on the transient stability of the plant. Fig. 5.10 shows the effect of a 400% increase in the governor valve rate limits and the corresponding increase in the transient stability limit is very small (less than 7%). The change in the machine slip speed is very small during and after the fault, so it is likely that extra feedback signals are

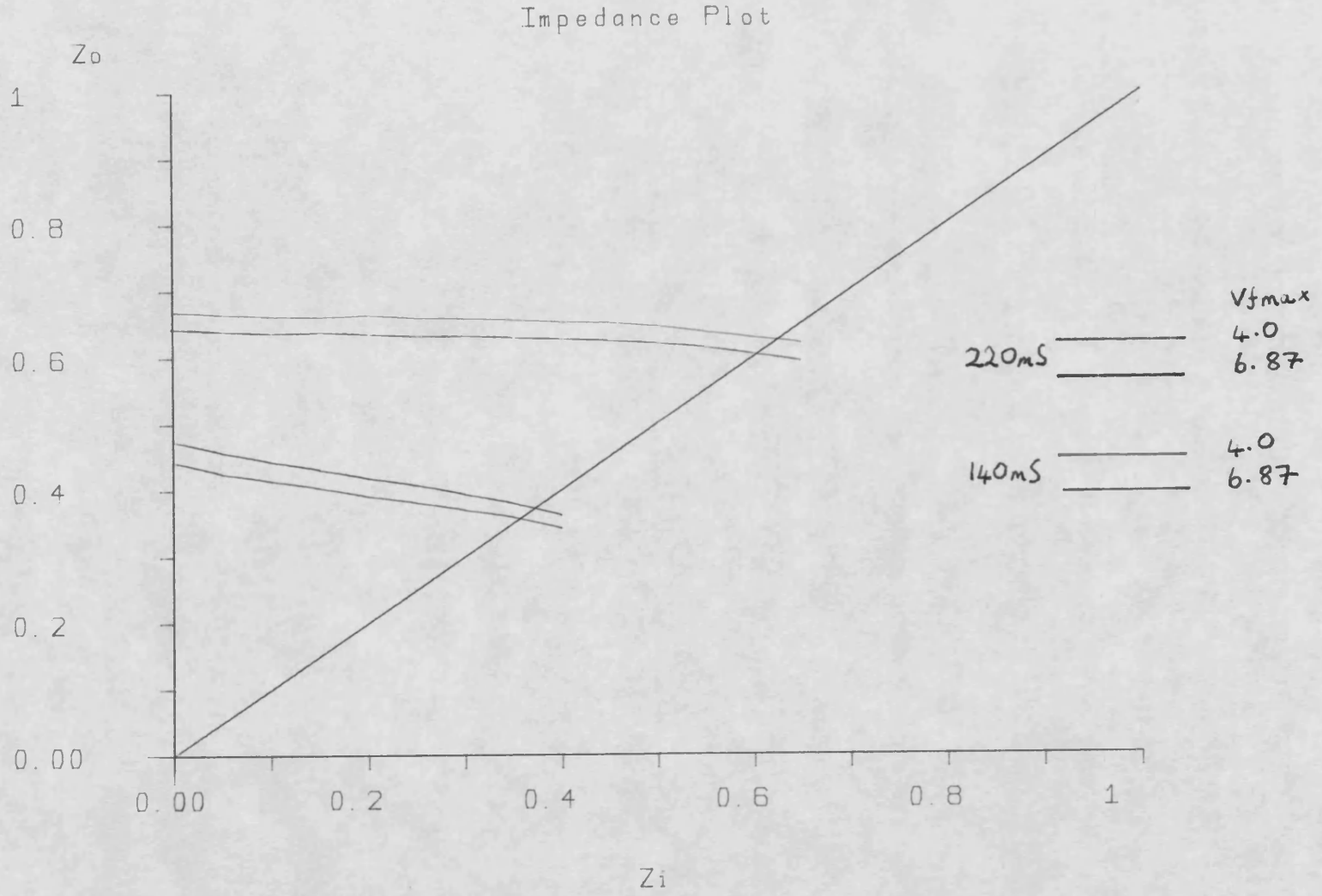

Impedance Plot
Zo
1
0.8
0.6
0.4
0.2
0.00
0.00
0.2
0.4
0.6
0.8
1
Zi
Vfmax
220mS
4.0
6.87
140mS
4.0
6.87

Fig 5.12

required to control the governor valves in order to make effectrive use of the faster valve gear.

Further pre-fault/post-fault impedance curves are presented in Chapter 6 to demonstrate the effect of strategies which optimise transient performance on the transient stability of the generating plant.

5.4 Summary

Both the dynamic and the transient stability of a generating system have been considered in this Chapter. A means of using the Real Time Simulator to generate a plot of the dynamic stability limit is described and used to generate the dynamic stability limit of the turbogenerator and the hydrogenerator power systems when conventionally controlled. Transient stability is far more difficult to consider since the transient stability of a generator is dependent not only on the operating point of the power system prior to the disturbance, but also on the type and severity of the disturbance itself. Transient stability has therefore been investigated by an indirect means, using pre-fault/post-fault impedance curves. A three-phase short circuit fault applied close to the terminals of the generator transformer is amongst the most severe disturbance encountered in a power system and transient stability studies based on this type of fault are of a worst case nature. Disturbances of a less severe nature are unlikely to result in transient instability if stability is ensured for a three-phase short circuit fault of this type. Not only does the pre-fault/post-fault impedance plot indicate an improvement or otherwise in the transient stability of

the plant, it also permits quantative assessments to be made of the percentage improvements offered by rival control strategies.

References

5-1 Demello, F.P. and Concordia, C.: "Concepts of synchronous machine stability as affected by excitation control", Trans.IEEE, PAS, Vol.88, pp318-329, April 1969.

5-2 Private communication with G. Shackshaft, included as Appendix C.

5-3 Brown, P.G., Demello, F.P., Lenfest, E.H. and Mills, R.J.: "Effects of excitation, turbine energy control and transmission on transient stability", Trans.IEEE, PAS, Vol.89, pp1247-1252, July/August 1970.

CHAPTER 6

OPTIMISATION STUDIES

This Chapter presents optimisation studies performed using the real-time simulator described in Chapter 4. Three different plant arrangements are considered in these studies, corresponding to:

(i) a steam-driven turbo-alternator connected via a transmission line to an infinite bus bar;

(ii) a water-powered hydro-generator connected via a transmission line to an infinite bus bar; and

(iii) a diesel-driven generator in a finite bus bar system, supplying an inductive load.

The objective of these studies has been to improve the system transient response and to establish appropriate gains through which additional feedback signals added into the excitation and governing control systems lead to this objective. As such, these are constrained optimisation studies, in that the structure of the controller is assumed and only a limited set of extra feedback signals is available. Three extra feedback signals are considered in these studies, namely, rotor acceleration, transient electrical power (at the generator terminals) and transient direct axis current. The usefulness of these additional feedback signals has been identified by a number of authors (6-1 to 6-5).

It was noted in Chapter 2 that, in general, an optimal controller for a linear system will require access to all the system states, and it would seem reasonable that an optimal controller for a non-linear system would also require access to all the state information. From the point of view of controller cost, it is desirable to keep the number of additional feedback signals to a minimum, since measurement of these signals may involve costly and complex transducers.

Optimal studies were performed by Lee (6-4) using the third order machine representation, taking as system states the change from their nominal values of transient quadrature axis voltage ($\Delta e'_q$), load angle ($\Delta\delta$) and slip speed (ΔY). Initial studies considered extra feedback into the excitation system only and significant improvements in transient performance after the first peak in rotor angle were achieved by this strategy. Expansion to the fifth order representation introduces two more states, the changes in direct and quadrature axis subtransient voltages ($\Delta e''_d$ and $\Delta e''_q$ respectively). Comparative studies by Lee (6-4) showed that little improvement was obtained by making these two further states available to the excitation system.

As anticipated, improvements in excitation control made no contribution to reducing the first peak in the load angle excursion following a fault, but the presence of extra feedback signals did improve the subsequent damping of rotor oscillations. In order to reduce the first peak in the load angle swing following a large system disturbance, it is necessary to reduce the power output by the prime mover in response to the fall in demand for power caused by the presence of the fault. Similar considerations to the above yield

change in load angle ($\Delta\delta$), slip speed (Y) and rotor acceleration ($P^2\delta$) as states to be fed into the governor loop. Sensitivity studies indicated that feedback of rotor acceleration into the governing system made the largest contribution to the improvement of system performance.

Lu (6-5) investigated the possibility of selecting a single extra feedback signal to be fed into both the excitation and governing loops. Consideration was given to change in load angle ($\Delta\delta$), rotor acceleration ($P^2\delta$), slip speed (Y) and transient electrical power (ΔPe). The justification for the latter is that, in the absence of a measurement for rotor acceleration, the change in electrical power gives a good indication of the power that the rotor will have to absorb by accelerating. Indeed, were there no change in the mechanical power output by the prime mover, identical improvements would be expected from the use of rotor acceleration and transient electrical power as extra feedback signals. Lu (6-5) also indicated the usefulness of change in direct axis current (Δi_d) as a substitute for the combination of quadrature axis transient voltage and change in load angle, two of the three signals used by Lee (6-4), in the excitation system.

6.1 Numerical Optimisation

It was noted in Chapter 2 that there is no general solution to the problem of minimising the value of a performance index subject to the necessary conditions of equations 2.28-2.31, with the important exception of a linear system and a performance index of quadratic form. This exception may be solved by application of either the

Matrix Ricatti or the Hamilton-Jacobi equation. In order to formulate such a solution for the systems under consideration, the system equations must be linearised about their operating point. This yields a small change representation of the system and ignores the effects of any limits that are present. The transients that are experienced by the machine following a fault are not small, and the limits encountered can significantly affect the transient performance of the plant. Such solutions tend to require access to all the states used to describe the system which conflicts with the need to reduce the number of extra signals involved.

An alternative approach is to assume a structure for a sub-optimal controller, to repeatedly apply faults to a dynamic model of the system, and to evaluate a performance index from the data collected over the control interval with the model being returned to identical initial conditions prior to each fault application. Adoption of this approach, which is highly suitable for use with a real-time digital model of the plant, reduces the problem of synthesising an optimal controller to that of function minimisation.

6.1.1 Function Minimisation

The performance index is considered to be a function of the gain parameters for which optimal values are required. This is illustrated in Fig. 6.1 for a two-dimensional parameter space, where the value of the performance index is indicated by the height of the surface.

The general procedure for locating a minimum point on the

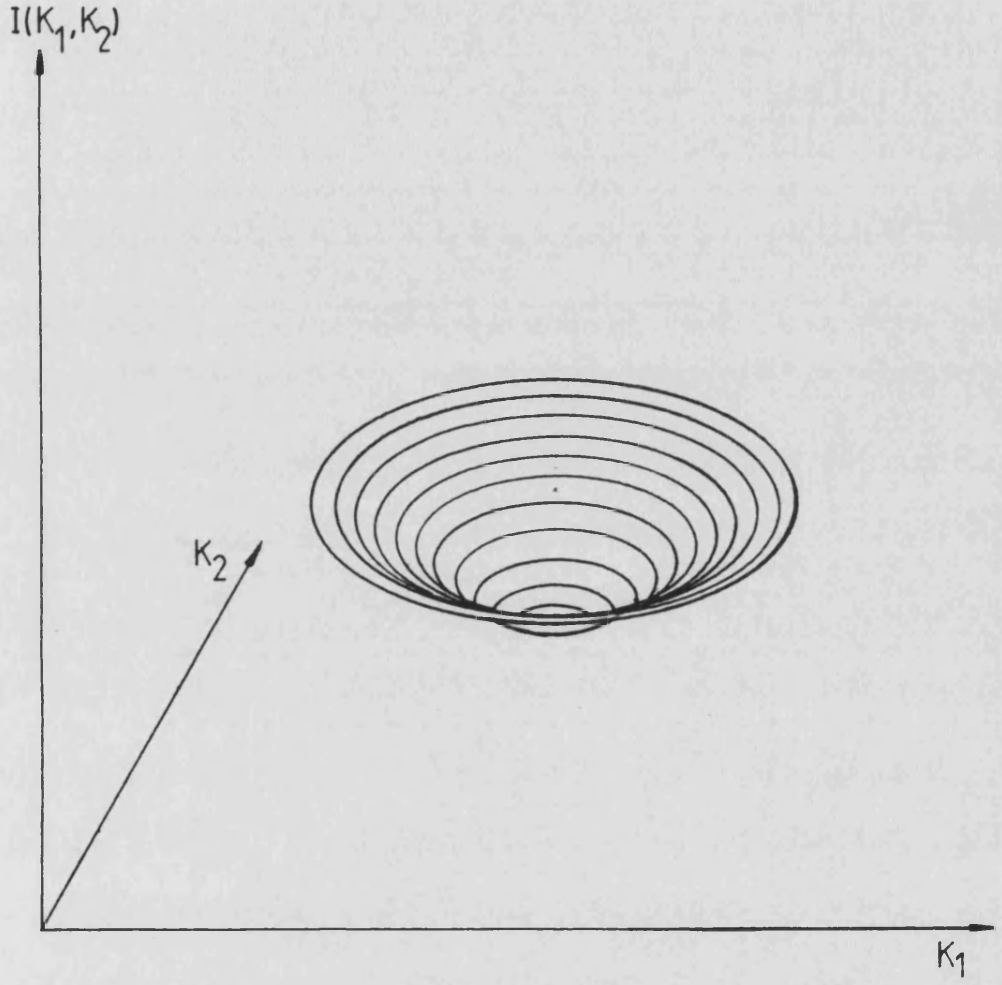


Fig 6.1 Performance Index Surface

performance index surface is as follows. Define a vector of parameters

$$\underline{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$$

where $\underline{a}$ is a position vector for an abitrary point in the parameter space P, which is a real n dimensional space. Define a scalar function $I(\underline{a})$ which is to be minimised with respect to the parameter vector $\underline{a}$. Searches are then made from an initial point $\underline{a}^o$ such that:

$$\underline{a}^{k+1} = \underline{a}^k + \lambda^k \underline{p}^k \qquad (6.1)$$

where $\underline{a}^{k+1}$ represents the starting point for the next iteration of the search algorithm, $\underline{a}^k$ represents the current starting point in parameter space, λ^k is a scalar and $\underline{p}^k$ is a vector which represents the direction along which the search for a minimum is conducted on the kth iteration.

At each step, the performance index $I(\underline{a}^{k+1})$ is evaluated and compared with the previous value of $I(\underline{a})$. This information is used to establish the next direction of search, $\underline{p}^{k+1}$. A variety of methods exist for choosing the direction in which to conduct the search for a minimum in $I(\underline{a})$. The most obvious method is to perform single dimensional searches along directions parallel to the axis of the parameter space, where $\underline{p}^k$ takes on one of n fixed values, each being a vector with n-1 zeros and a single element set to unity. For example, in two-dimensional parameter space, $\underline{p}^k$ may take on the two

values. $\underline{e}_1 = \begin{bmatrix}1\\0\end{bmatrix}$ and $\underline{e}_2 = \begin{bmatrix}0\\1\end{bmatrix}$. The objective of each iteration in the process is to establish a value for λ^k such that:

$$I(\underline{a}^{k+1}) = \min_{\lambda^k} I(\underline{a}^k + \lambda^k \underline{p}^k) \qquad (6.2)$$

Finding λ^k is now a single dimensional search problem and methods for estimating λ^k will be given in the next section. The search is then repeated in turn for each of the n directions $\underline{e}_n$ until there is 'little' reduction in the value of $I(\underline{a}^{k+1})$, at which point the search stops.

This method works well when $I(\underline{a})$ has circular contours. However, progress is slow in the presence of narrow valleys, as shown in Fig. 6.2, for the two-dimensional case.

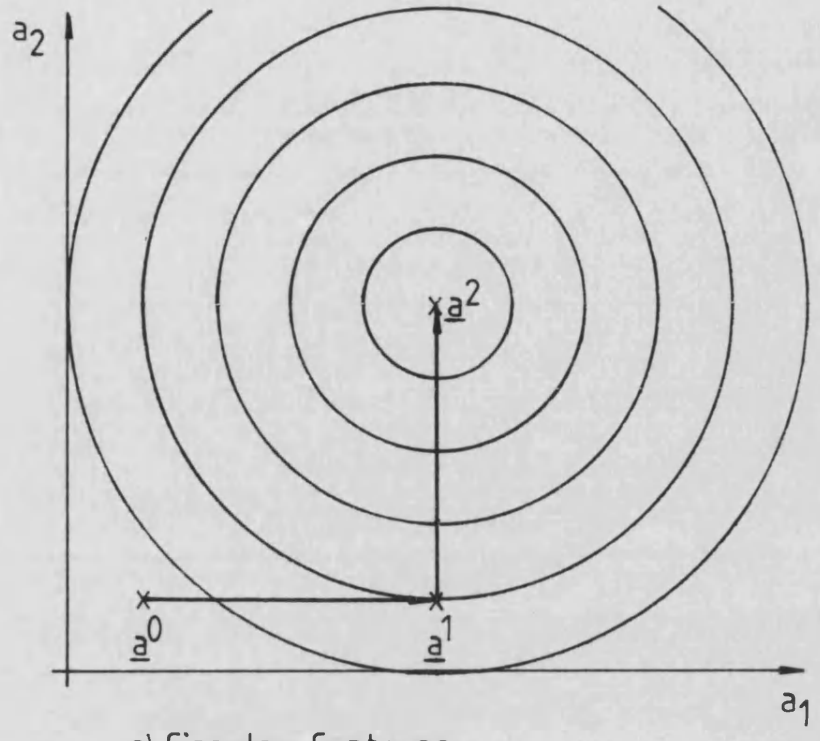
6.1.2 Gradient Methods

Another way in which to choose $\underline{p}^k$ is to consider the direction of steepest descent. This direction is given by the gradient function for the surface as follows:

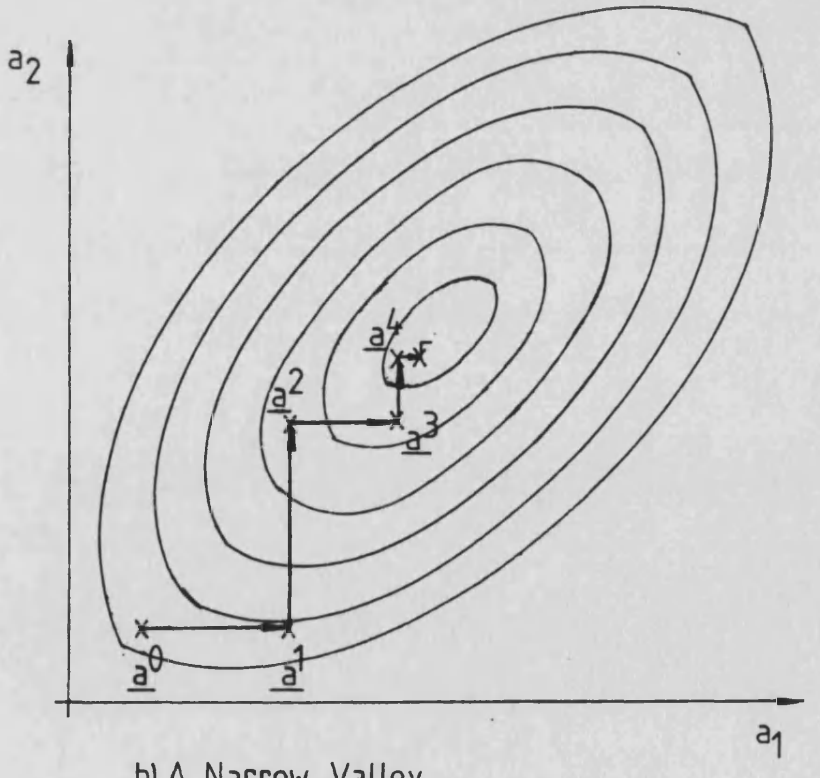
$$\underline{p}^k = -\nabla_a I(\underline{a})\big|_{\underline{a} = \underline{a}^k} \qquad (6.3)$$

where

$$\nabla_a I(\underline{a}) = \begin{bmatrix} \partial I/\partial a_1 \\ \vdots \\ \partial I/\partial a_n \end{bmatrix} \qquad (6.4)$$



a) Circular Contours



b) A Narrow Valley

Fig 6.2 A Simple Multiparameter Search

Once $\underline{p}^k$ has been chosen, the problem reduces to a single dimensional search to find a suitable value for λ^k as before. By considering the partial derivative of $I(\underline{a})$ with respect to λ, it can be established that $\underline{p}^k$ will be tangential to the contours of $I(\underline{a})$ at $\underline{a} = \underline{a}^{k+1}$. It can also be seen that $\underline{p}^{k+1}$, chosen as the direction of steepest descent, will be orthogonal to $\underline{p}^k$.

Consider equation 6.1:

$$\underline{a}^{k+1} = \underline{a}^k + \lambda^k \underline{p}^k$$

then
$$\left.\frac{\partial I}{\partial \lambda^k}\right|_{\underline{a} = \underline{a}^{k+1}} = \sum_{i=1}^{n} \frac{\partial I}{\partial a_i} \cdot \left.\frac{\partial a_i}{\partial \lambda^k}\right|_{\underline{a} = \underline{a}^{k+1}}$$

$$= \nabla_a I(\underline{a})\Big|_{\underline{a} = \underline{a}^{k+1}}^{T} \cdot \underline{p}^k$$

At $\underline{a} = \underline{a}^{k+1}$, $\partial I/\partial \lambda^k = 0$, which gives the next direction of search, $\underline{p}^{k+1} = -\nabla_a I(\underline{a})\big|_{\underline{a} = \underline{a}^{k+1}}$ as orthogonal to $\underline{p}^k$. Also, since the partial derivative $\partial I/\partial \lambda^k$ is zero at $\underline{a} = \underline{a}^{k+1}$, the search direction $\underline{p}^k$ must be tangential to the contours at this point. Given this, the steepest descent method is prone to zigzag in narrow valleys in the same manner that occurred with the previous method. This is due to the fact that, once $\underline{a}^0$ has been chosen, then $\underline{p}^k$ may only be one of 'n' different directions made up of p^0 and the $n-1$ directions orthogonal to $\underline{p}^0$. This is illustrated in Fig. 6.3.

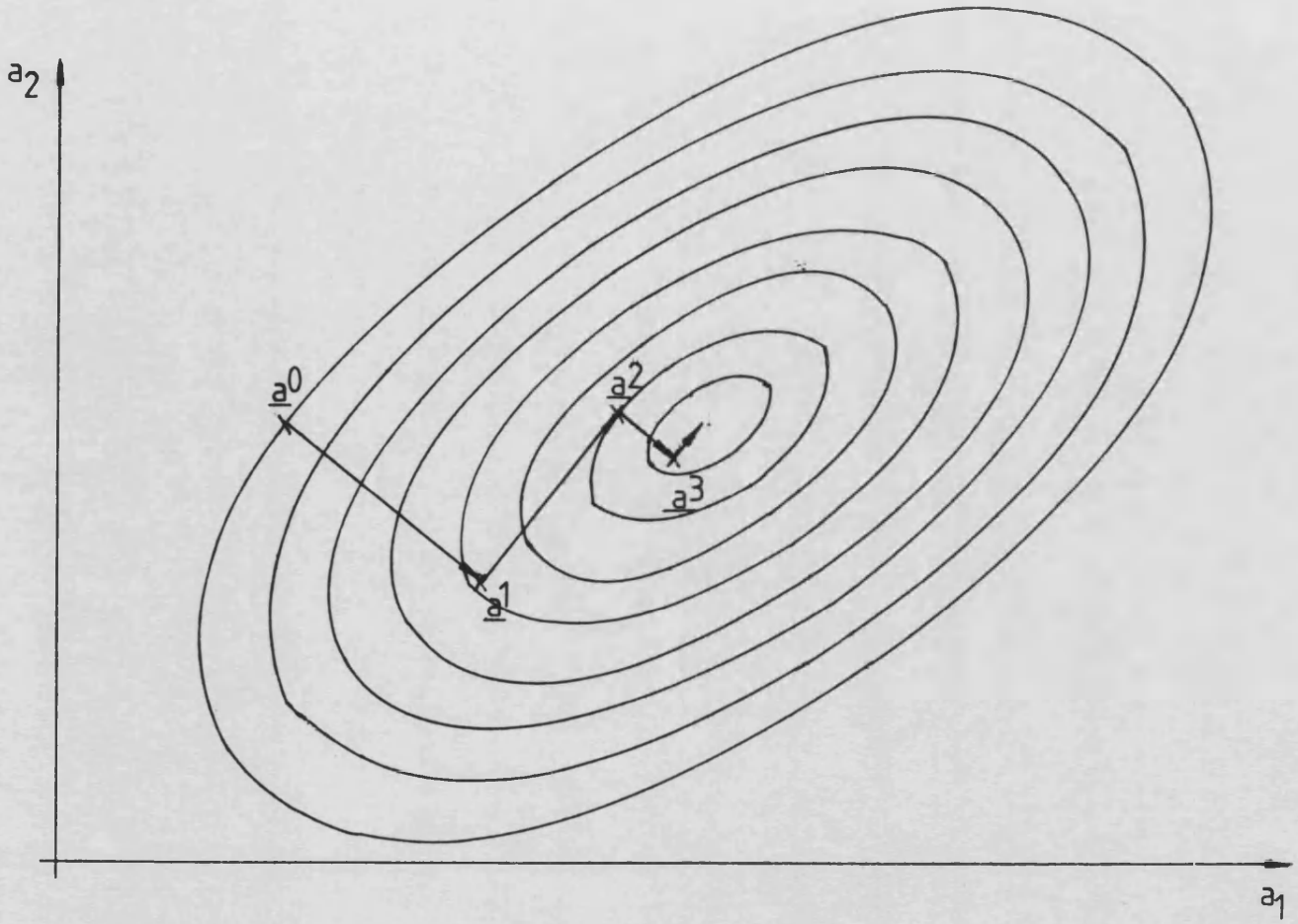


Fig 6.3 A Steepest Descent Trajectory

The gradient information required by the steepest descent method is not explicitly available from the real-time model, although estimates may be made by use of finite difference calculations which require small, local perturbations in parameter space.

The conjugate gradient method of Hestenes and Stiefel (6-6) which improves on the convergence properties of the steepest descent method, also requires gradient information, as do a whole host of Quasi Newton and Gauss Newton methods (6-8,6-9). Some of these methods require estimates of the so-called Hessian matrix of partial second derivatives of the performance index. These schemes are derived from a truncated second order Taylor series expansion which yields the result that:

$$\underline{p}^k = [G]^{-1} \nabla_a I(\underline{a})\Big|_{\underline{a} = \underline{a}^k} \qquad (6.5)$$

where $$G = \begin{bmatrix} \partial^2 I/\partial a_1 \partial a_1 & \cdots & \partial^2 I/\partial a_1 \partial a_n \\ \vdots & & \vdots \\ \partial^2 I/\partial a_n \partial a_1 & \cdots & \partial^2 I/\partial a_n \partial a_n \end{bmatrix}$$

and $\nabla_a I(\underline{a})$ is defined in equation 6.4.

6.1.3 Non-Gradient Methods

The minimisation technique proposed by Rosenbrock (6-10) does not require estimation of the gradient of the performance index. This method searches until it finds a valley and then aligns one of the search directions along the valley, adjusting the remaining search directions to form a set of orthogonal directions which span the

space.

The minimisation technique used to perform the optimisation studies presented in this Chapter is known as Simplex (6-11). A simplex may be thought of as a wire frame shape with one more vertex than the dimension of the parameter space, i.e. a triangle in two space or a tetrahedron in three space.

The position of the simplex vertices represents values for the parameter vector $\underline{a}$ and each vertex has a value corresponding to the value of the performance index at that point in parameter space. The procedure is started by choosing an arbitrary simplex and evaluating the performance index at each vertex. The vertex with the worst (highest) value is then reflected through the centroid of the remaining vertices. If this new vertex gives a better value for the performance index than the vertex that has just been reflected, this new vertex replaces the worst vertex in the simplex. Further, if the new vertex has a performance index value that is better than the best one currently held in the simplex a further trial is made at a point twice the distance from the centroid in the same direction. Should this 'expanded' vertex yield a better value for the performance index than the 'reflected' vertex, then the 'expanded' vertex is used to update the simplex instead of the reflected; vertex. This action permits the size of the simplex to grow in directions that give significant improvements in the performance index. Should the first trial, the reflection, result in a performance index value that is worse (higher) than the worst vertex of the simplex, a further trial is made at the mid-point between the centroid of the remaining vertices and the current worst vertex. If this trial should succeed,

i.e. yield a performance index value which is better than that of the current worst vertex, this new, 'contracted' vertex is used to replace the worst vertex in the simplex. However, if this last trial fails to yield a better performance index value, the entire simplex is halved in size towards the best vertex currently held in the simplex. These last two operations result in the contraction of one or more vertices, permitting the simplex to shrink in size as a minimum is approached, or to shrink in specific directions in order to enter a valley or to turn a corner when following a valley. Having established a new simplex from manipulation of the old simplex, the procedure is repeated until both the fractional size of the simplex and the fractional improvement in the performance index across the simplex are both below suitable values.

This ensures that the simplex vertices are close together in parameter space. The procedure is summarised in the flow chart of Fig. 6.4, and the various manipulations of the simplex are illustrated in Fig. 6.5 for the case of a two-dimensional parameter space.

Referring to Fig. 6.5, the following equations may be written relating the various vertices. In their vector form, these equations are general for the n-dimensional parameter space. Note that '$\underline{B}$' and '$\underline{W}$' refer to the best and worst vertices, respectively, which may also be referenced as specific instances of the vector $\underline{a}_i$. Let all the simplex vertices be represented by the vectors $\underline{a}_i \cdots \underline{a}_{n+1}$. The centroid required for the reflection is given by:

$$\underline{C} = \frac{1}{n} \sum_{\substack{i=1 \\ i \neq j}}^{n+1} \underline{a}_i \qquad \text{where } \underline{a}_j = \underline{W} \tag{6.7}$$

The reflected vertex $\underline{R}$ is given by:

$$\underline{R} = (1 + \alpha)\underline{C} - \alpha\underline{W} \tag{6.8}$$

where α is a reflection coefficient with, typically, $\alpha = 1$.

The expanded vertex, $\underline{E}$, is given by:

$$\underline{E} = \gamma\underline{R} + (1-\gamma)\underline{C} \tag{6.9}$$

where γ is an expansion coefficient with, typically, $\gamma = 2$.

The contracted vertex, $\underline{M}$, is given by:

$$\underline{M} = (1-\beta)\underline{C} + \beta\underline{W} \tag{6.10}$$

where β is a contraction coefficient with, typically, $\beta = 0.5$.

When shrinking, the shrunk vertices, $\underline{a}_i'$, are given by:

$$\underline{a}_i' = (1-\beta)\underline{\beta} + \beta\underline{a}_i \tag{6.11}$$

Nelder and Mead (6-11) investigated the effect of using different values for the coefficients α, β and γ. They concluded that, for a number of deterministic functions, the simplex converges most rapidly with α, β and γ set at 1, 0.5 and 2, respectively. The functions

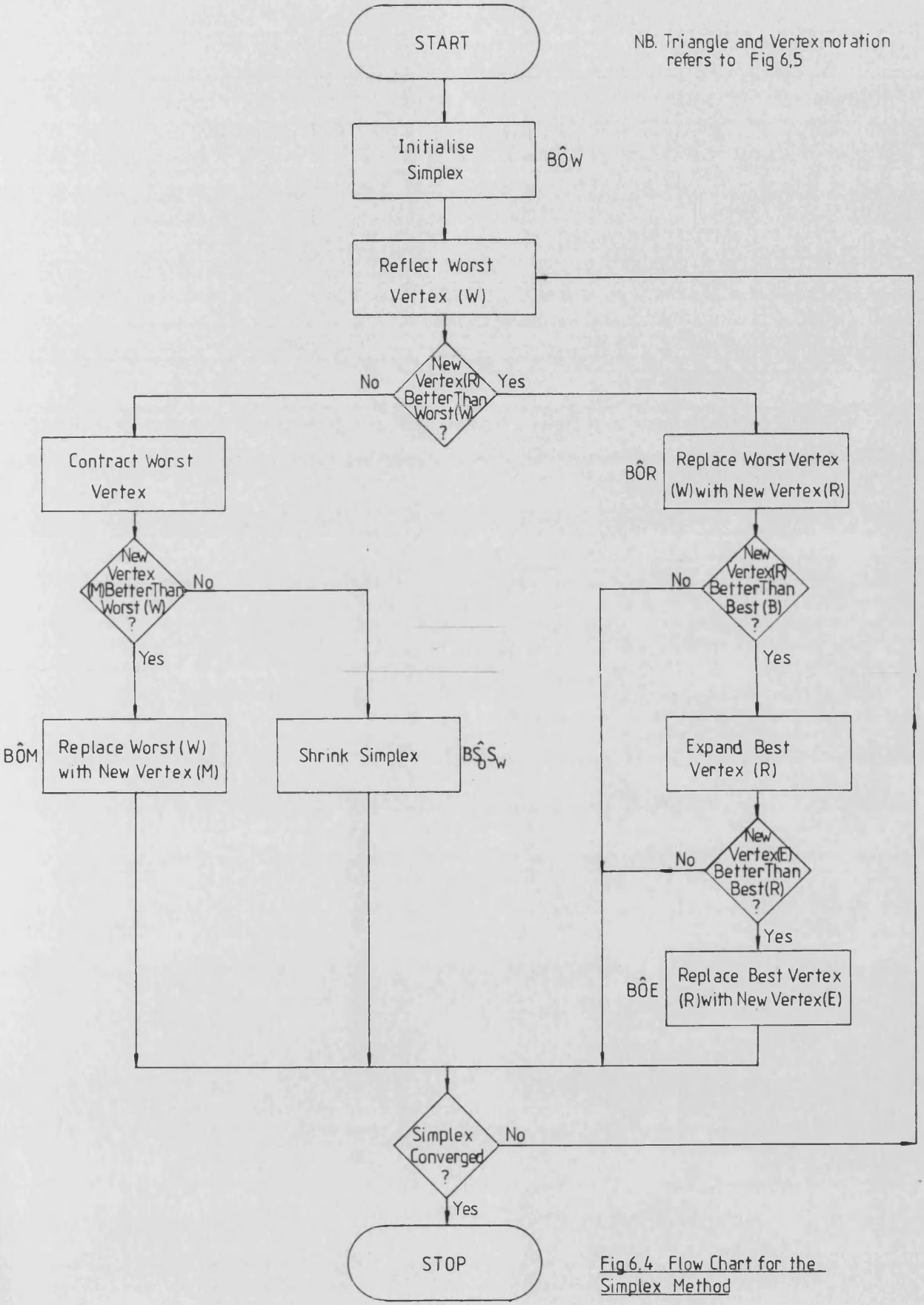


Fig 6.4 Flow Chart for the Simplex Method

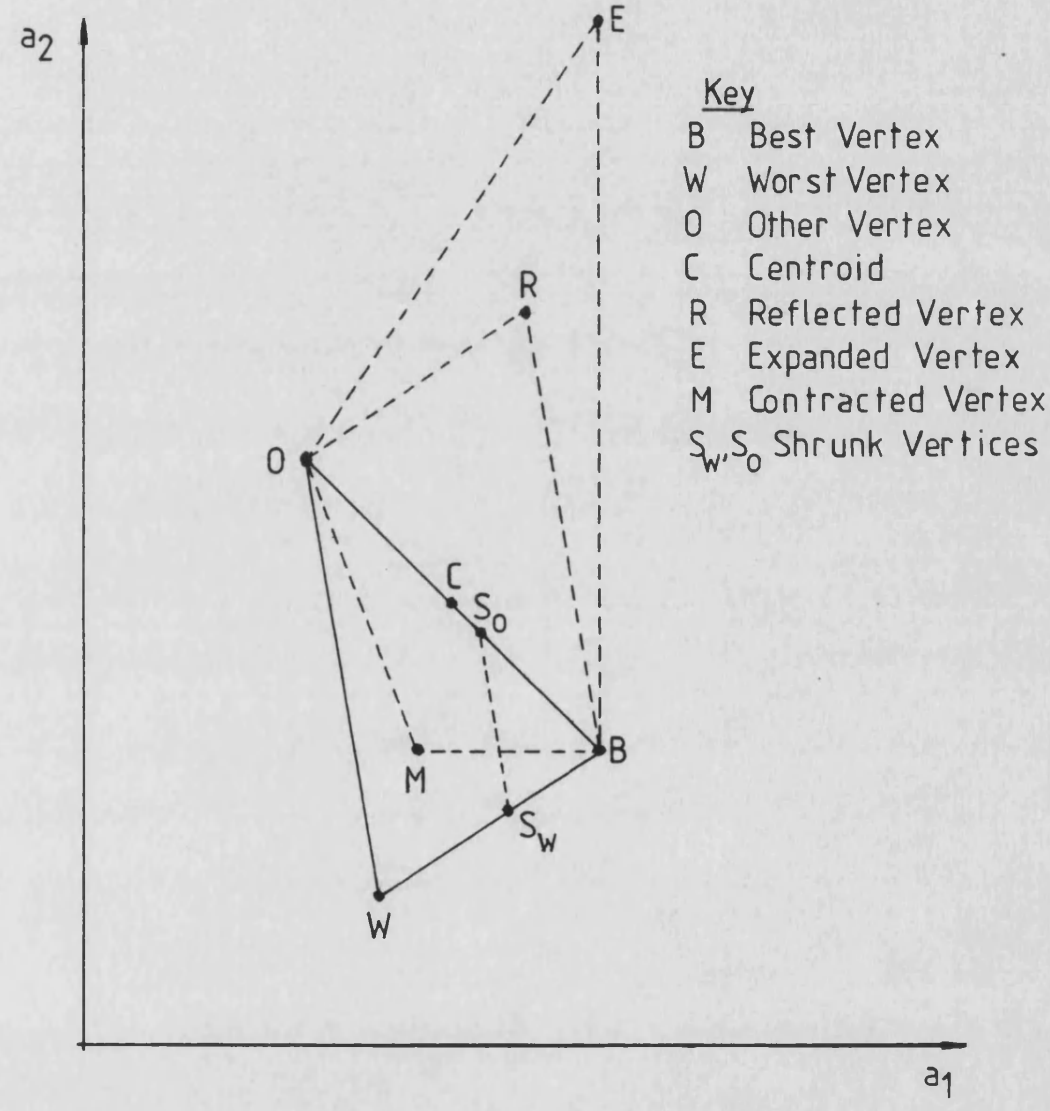


Fig 6.5 Two Dimensional Simplex Manipulations

used to make these comparisons were those used by other authors (6-10, 6-12,6-13) for evaluating the performance of their own algorithms. It was concluded that the convergence properties of the simplex method are at least as good as those of the best of Powell's methods (6-14).

The convergence of a simplex minimisation is illustrated in Fig. 6.6. The performance index function, I, used to generate this figure is similar to the parabolic valley used by Rosenbrock (6-10) and is of the form:

$$I(k_1, k_2) = A.(k_1 - k_2^2)^2 + (1 - k_1)^2 + 1 \qquad (6.12)$$

where A is an arbitrary constant and k_1 and k_2 are the parameters to be optimised. Note that the contours have been logarithmically spaced to avoid bunching. It can be seen that the initial simplex commences on the side of the curved valley. Initially, the simplex expands until it passes through the valley bottom. It then contracts to locate the valley bottom more accurately. On discovering that the valley runs downhill, the simplex expands along the downhill direction. Several expansion and contraction steps steer the simplex around the curvature in the valley before it finally homes in on the minimum point of the surface at $k_1 = k_2 = 1$.

6.2 Single Parameter Optimisation

It was shown in the previous section that, in general, the problem of searching for a minimum in a multi-dimensional parameter space, may be solved by repeatedly performing single dimensional

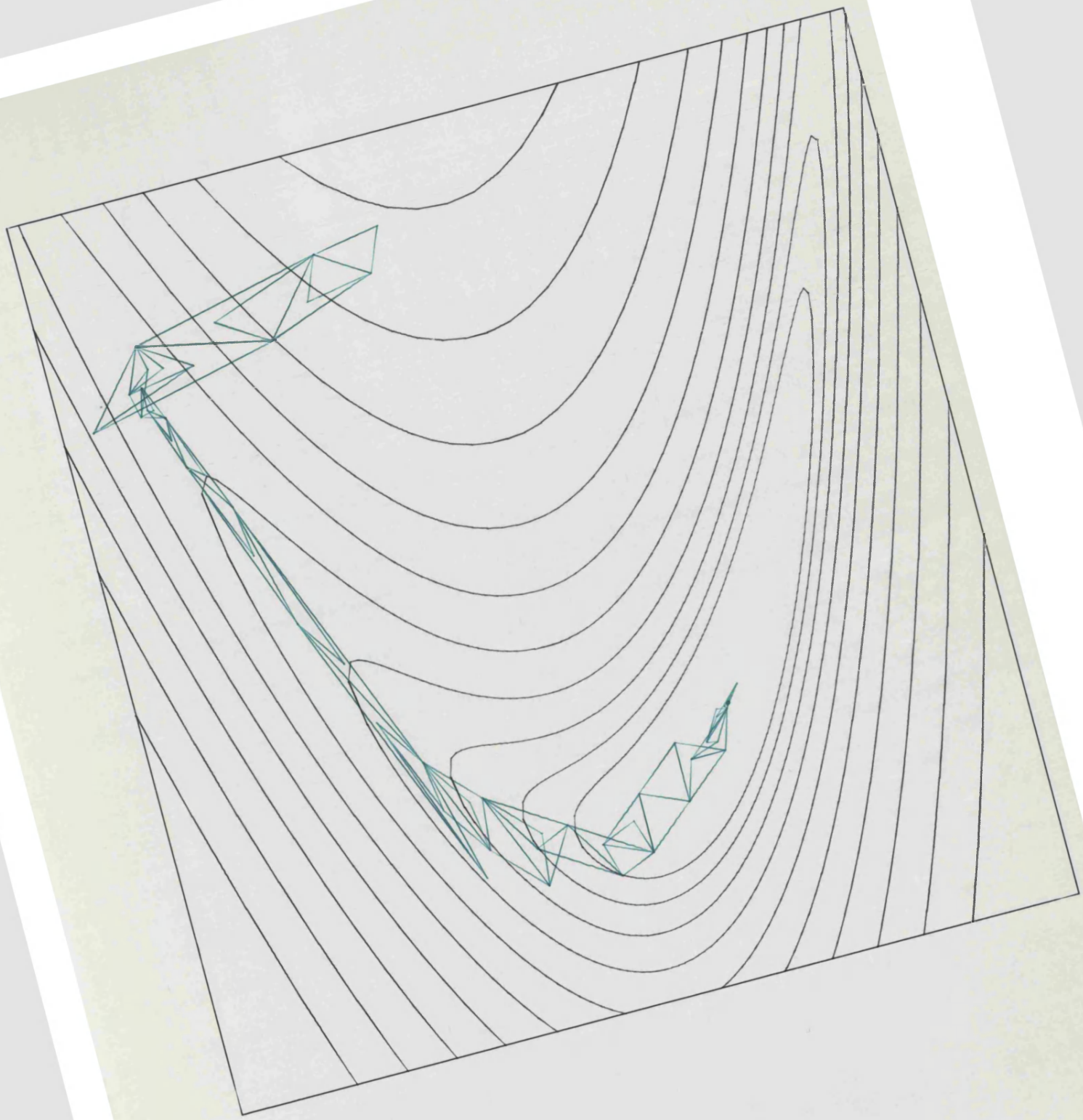


Fig 6.6 A Simplex Trajectory

minimisations along fixed directions in the multi-dimensional parameter space.

A variety of mechanisms are available for minimising a function with respect to a single parameter and these are known as linear searches. The problem was formulated in the previous section, equation 6.2, and, given $\underline{a}^k$ and $\underline{p}^k$, $I(\underline{a})$ may be expressed solely as a function of the scalar λ.

A direct search may be made by evaluating $I(\lambda)$ at some initial point $\lambda = \lambda_0$. Then $I(\lambda_1)$ is evaluated, where $\lambda_1 = \lambda_0 + h$ and h is a positive increment. If the value of λ that yields the desired minimum value in $I(\lambda)$ is denoted λ^*, then, if $I(\lambda_1) < I(\lambda_0)$, it follows that $\lambda^* > \lambda_0$. By making such comparisons, the direction of search may be controlled towards λ^*. Clearly, the rate of convergence is heavily dependent on h, whose magnitude may be large initially in order to speed convergence, and reduced once oscillations about λ^* are detected. This process continues until there is no significant change in $I(\lambda)$ and λ.

Curve fitting algorithms are also popular. A low order polynomial, typically quadratic, is considered to be a suitable approximation to $I(\lambda)$ in the region of the minimum. With three suitable values for λ and $I(\lambda)$, a quadratic function may be defined such that:

$$I(\lambda) = q_2\lambda^2 + q_1\lambda + q_0$$

and λ^* may be approximated as $\lambda^* = -q_1/2q_2$. Suitable values of λ and

$I(\lambda)$ would be such that a curve through them is concave, thereby ensuring that a minimum exists, i.e. $q_2 > 0$. This is illustrated in Fig. 6.7.

6.3 Optimisation Studies

6.3.1 Choice of Performance Index

Since an optimal control is designed to minimise a cost function, the performance index, it is vital that the performance index embodies the objectives of the control scheme. In the case of a generator connected to an infinite bus bar, there are three basic objectives, namely, rapid terminal voltage recovery following the clearance of a severe fault, high first swing stability margin, and effective damping of subsequent load angle oscillations.

The performance index adopted for the infinite bus bar studies takes the form:

$$I = \int_{t_o}^{t_1} (A1.\Delta V_t^2 + \Delta\delta^2)\,(1 + A2.t)\,dt \qquad (6.13)$$

When a fault is present, no action can be taken to restrict the terminal voltage deviation, so the term $A_1.\Delta V_t^2 + A1.A2.t.\Delta V_t^2$ under the integral in equation 6.13, contributes to the performance index, a constant term and a term dependent on the terminal voltage recovery. The term $\Delta\delta^2$ ensures that the value of the first peak in the load angle swing is limited by penalising large load angle excursions. The term $A2.t.\Delta\delta^2$ ensures that damping of the load angle oscillations is achieved by application of a time penalty on the load

For concave surface with $\lambda_3 > \lambda_2 > \lambda_1$:-

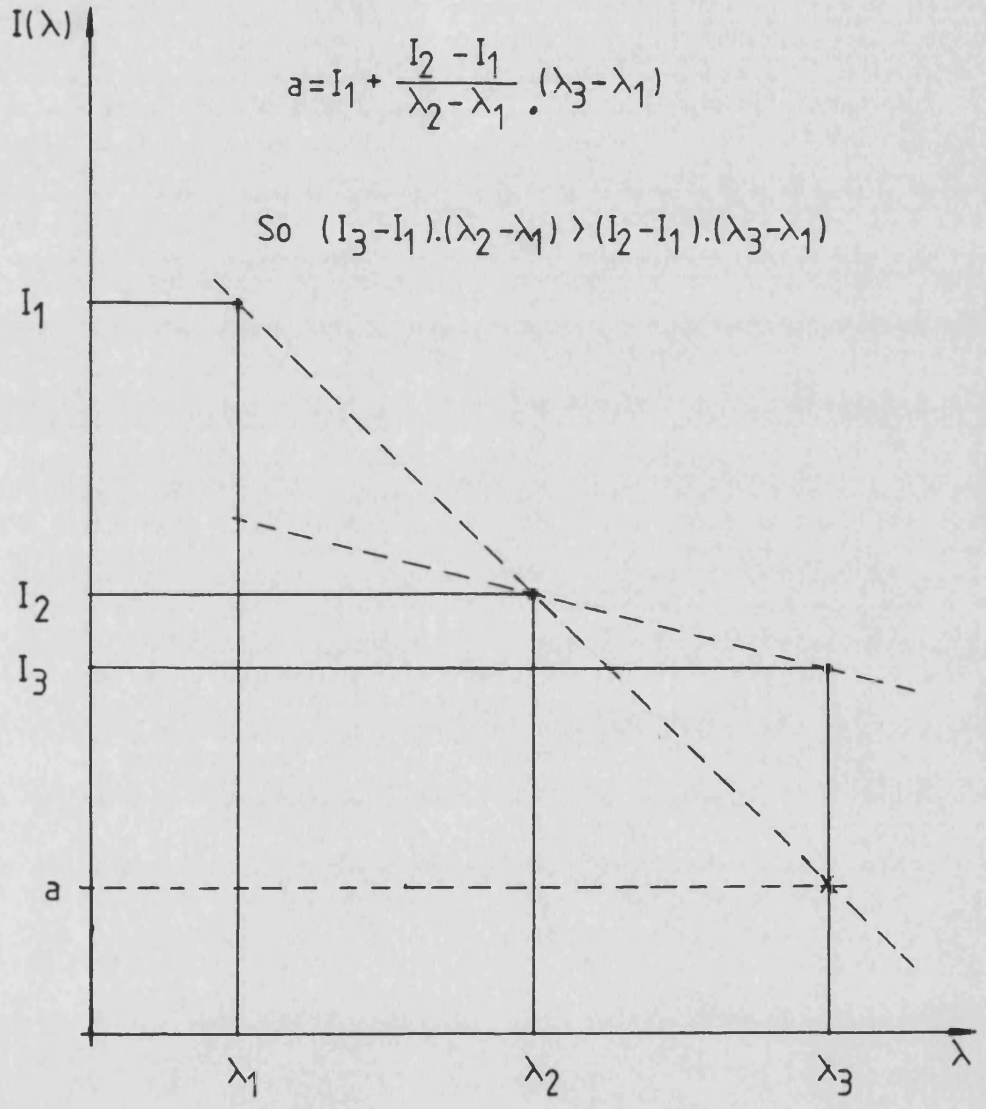


Fig 6.7 Inequality for a Concave Surface

angle deviations. The choice of weighting factors A1 and A2 is somewhat arbitrary. However, a low weighting on the term ΔV_t^2, a higher weighting on the term $t.\Delta V_t^2$, and a much higher weighting on the term $t.\Delta\delta^2$ would be consistent with the control objectives. Such relative weightings can be achieved by choosing A1 and A2 as 0.1 and 70, respectively.

In the case of the finite bus bar system, with a single generator, the stability criteria are quite different. Indeed, it is impossible for the generator to lose synchronism as there is nothing for it to be in synchronism with. The objectives of the optimal control then become twofold, namely, rapid terminal voltage recovery following a severe fault, and the minimisation of the rotor speed deviation. The performance index used for the finite busbar studies takes the form:

$$I = \int_{t_o}^{t_1} (\Delta V_t^2 + A1.\Delta\omega^2 + A2\Delta t_{out}^2)\,(0.5 + A3.t)\,dt \qquad (6.14)$$

The inclusion of the ΔV_t^2 and $\Delta\omega^2$ terms is clearly consistent with the control objectives, as is the inclusion of the time weighted term associated with A3, which will encourage rapid recovery from the transient. The motivation for the inclusion of the $\Delta tout^2$ term will be dealt with in a later section, along with a discussion on the choice of the weighting factors, A1, A2 and A3.

6.3.2 Optimisation of a Turbo-alternator, Infinite Busbar System

The system optimised in this section uses the data for Pembroke

power station given in Appendix A. The system is assumed to be fitted with an electrohydraulic governing system with fast valve gear. The excitation system is assumed to be capable of field voltage reversal, although optimisations have been performed without the ability to reverse field voltage. The time constants associated with the excitation system are therefore typical of those found in a thyristor type excitation system.

As mentioned at the beginning of this chapter, the optimisations have been performed to establish suitable gains through which the extra feedback signals, rotor acceleration ($p^2\delta$), transient electrical power (ΔPe) and transient direct axis current (Δid), may be added into the excitation and governing system to improve the systems transient performance. The system under study is represented in Fig. 6.8 with the two extra inputs denoted as Ti and Vi.

Symmetrical three-phase short-circuit faults are applied to the high voltage terminals of the generator transformer for a predetermined duration. The base fault clearance time used in this study is 220ms, and optimisations have been performed for a number of different operating conditions. The performance index is integrated over a 5 second control interval.

Table 6.1 gives results for two different optimisation runs aimed at finding suitable gains for rotor acceleration feedback.

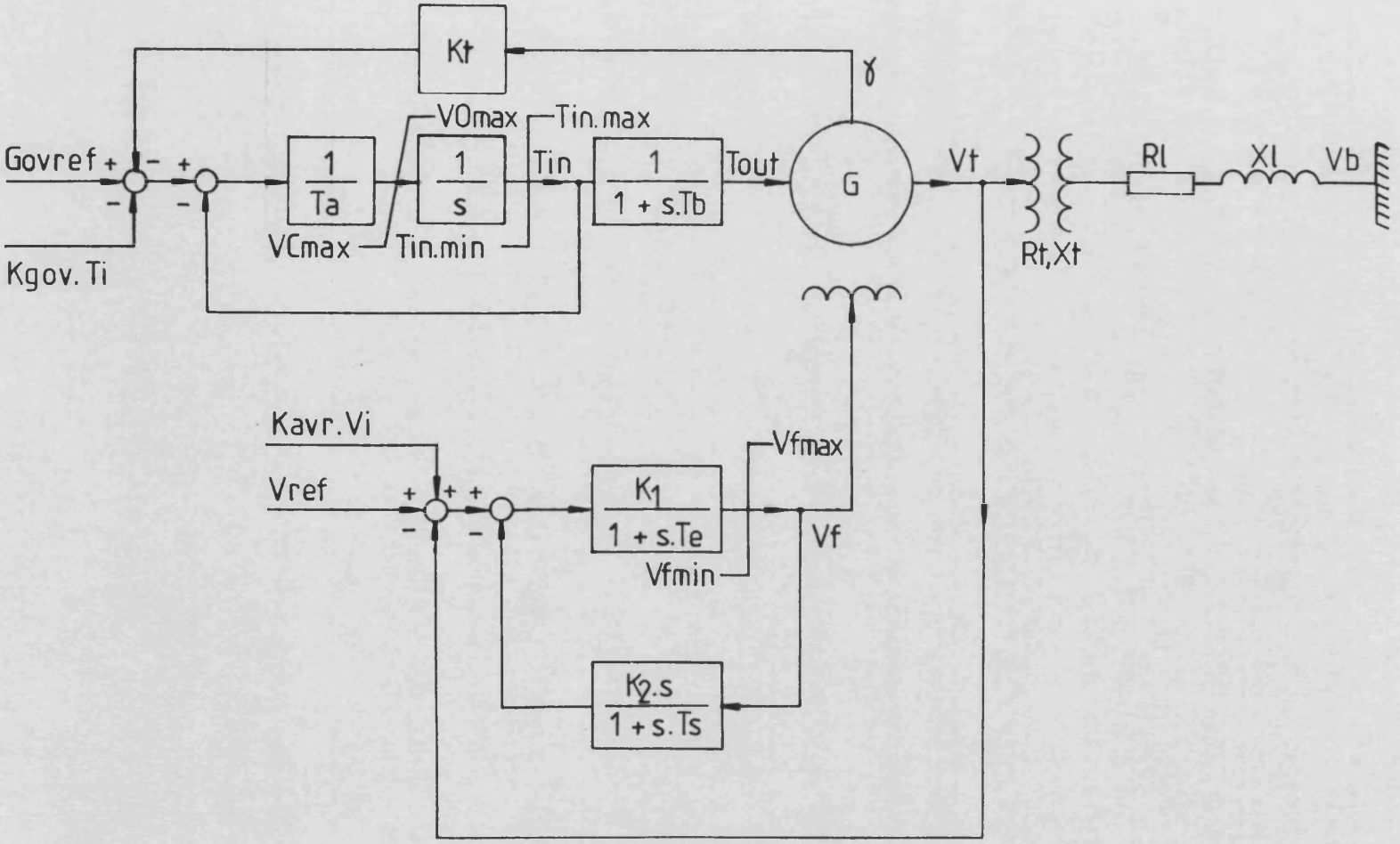


Fig 6,8 Block Diagram of the Turboalternator, Infinite Bus System

Initial Conditions

δ = 1.25 rad Vt = 1.03 p.u. P = 0.884 p.u. Q = −0.0488 p.u.

V_t = 2.85 p.u. Power Factor = 0.998 leading.

Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A1	None	0	0	118	220ms	6.87	−6.87
A2	$P^2\delta$	0.145	0.934	26.2	220ms	6.87	−6.87
A3	$P^2\delta$	0.0400	0.0301	17.4	220ms	6.87	−6.87

Table 6.1

Note that, in Table 6.1 and all the tables that follow, the values quoted for power, P, the voltamperes reactive, Q, and the power factor are with respect to the infinite bus bar.

Transient responses corresponding to controls A1-A3 are given in Fig. 6.9(a) to (f) being terminal voltage, field voltage, load angle, mechanical torque, direct axis current, and electrical power at the machine terminals. Responses for A1, A2 and A3 are black, red and green in colour, respectively. The two sets of gain values for A2 and A3 are widely separated in parameter space, simply because the search for control A2 was initialised with a larger starting simplex than that for A1, thereby converging to a different local minimum.

At fault inception, the terminal voltage drops almost instantly to a level determined by the ratio of the internal impedances of the

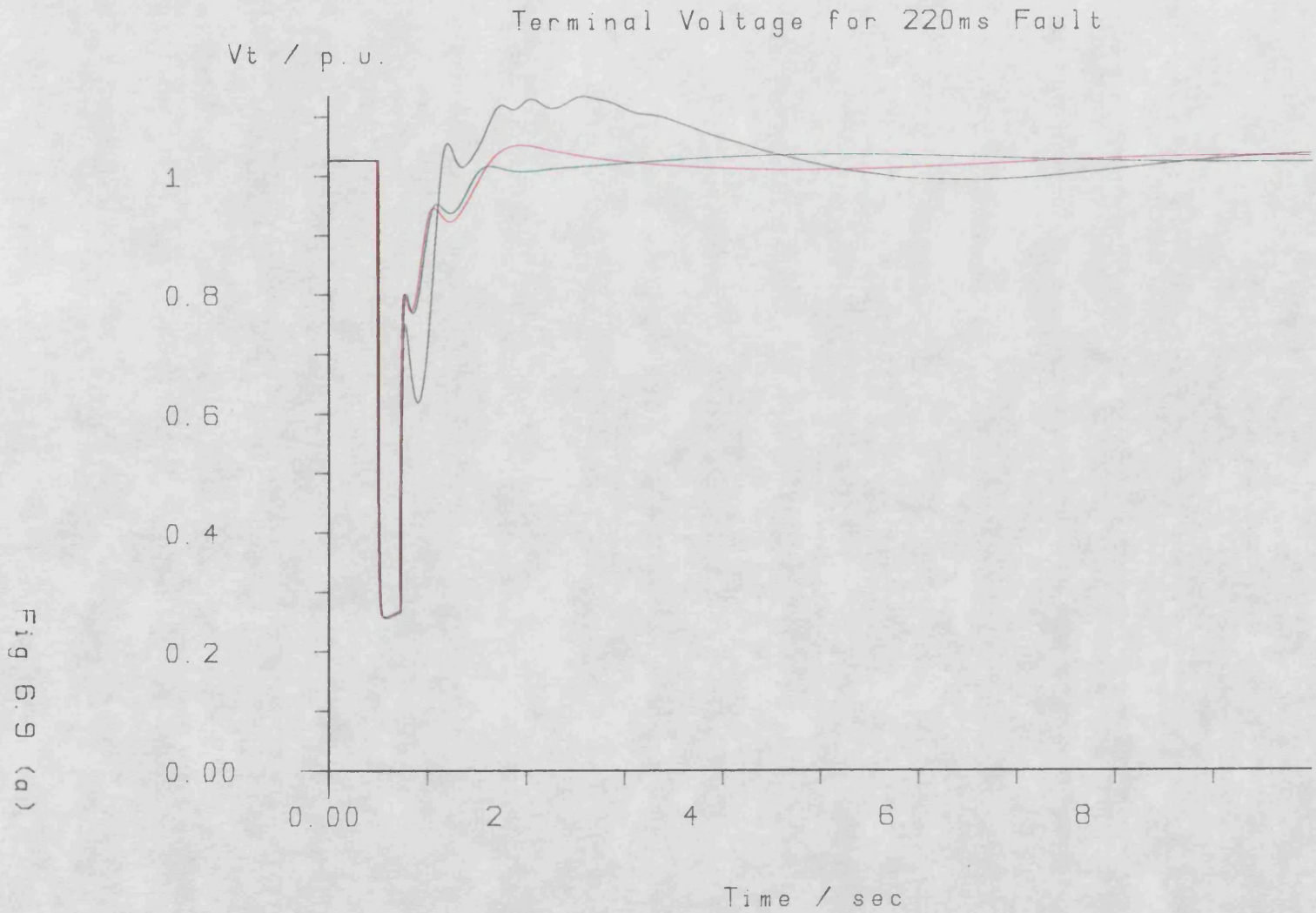

Terminal Voltage for 220ms Fault
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.9 (a)

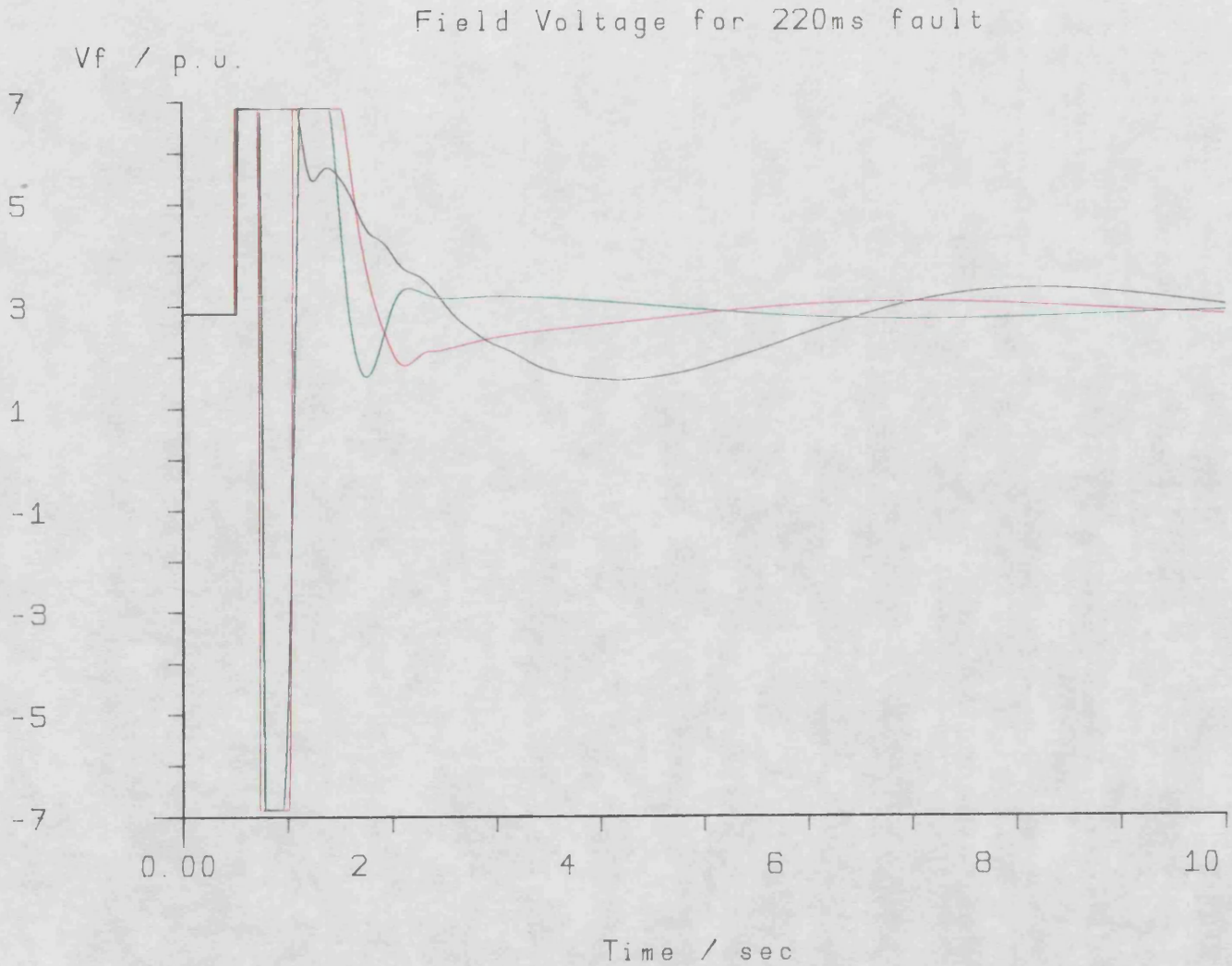

Field Voltage for 220ms fault
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
10
Time / sec

Fig 6.9 (b)

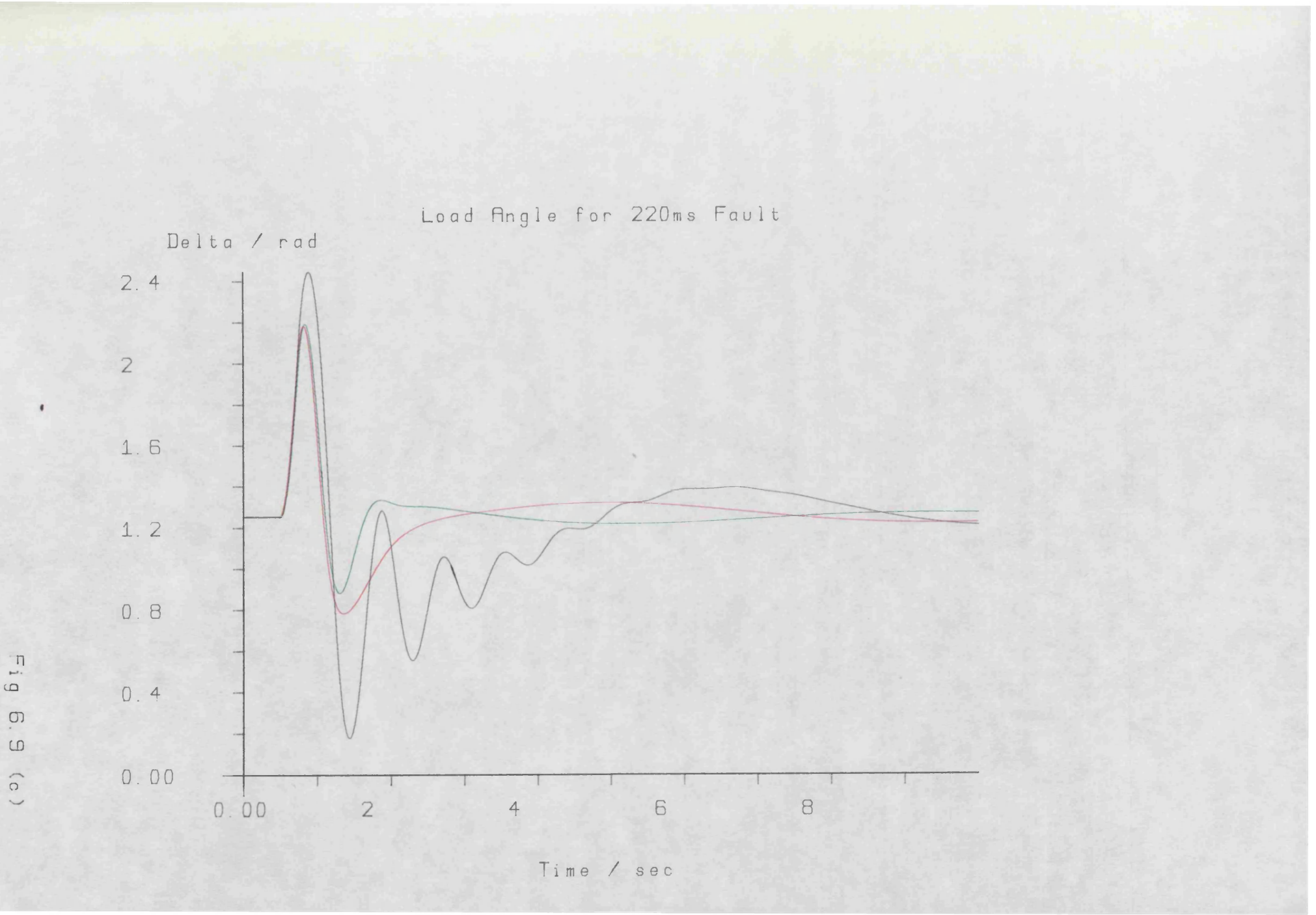

Load Angle for 220ms Fault
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
2
4
6
8
Time / sec

Fig 6.9 (c)

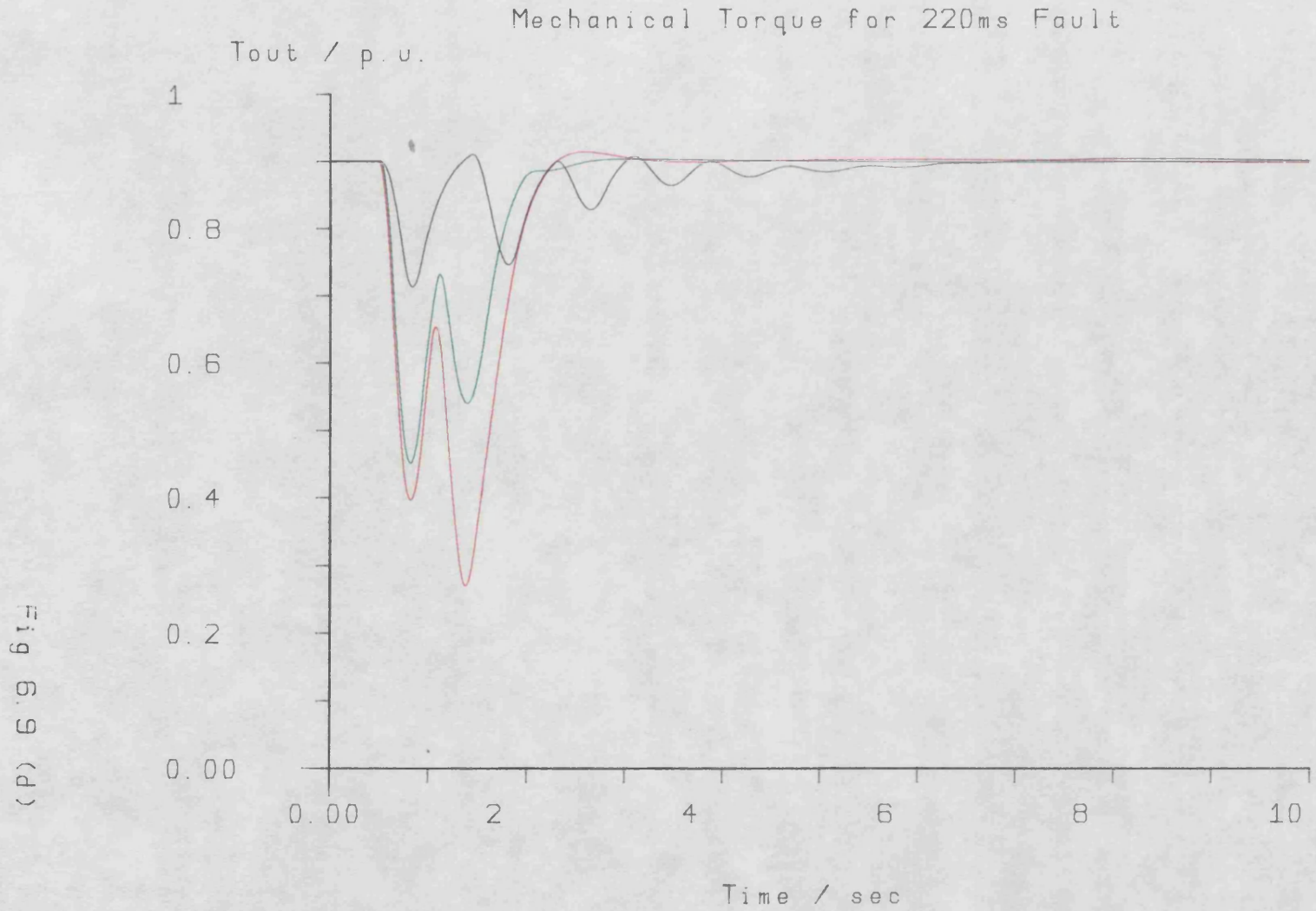

Mechanical Torque for 220ms Fault
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.9 (d)

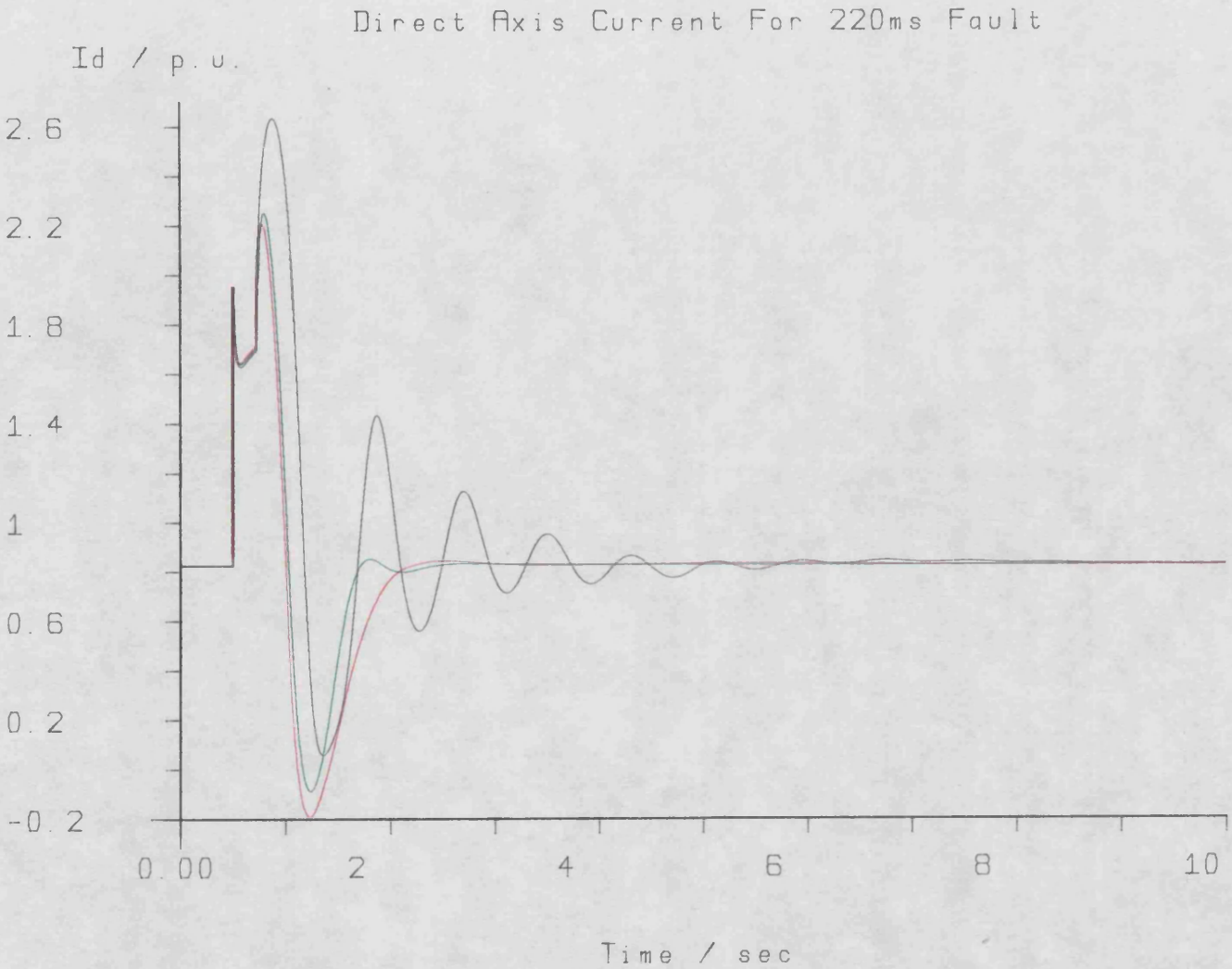

Direct Axis Current For 220ms Fault
Id / p.u.
2.6
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
10
Time / sec

Fig 6.9 (e)

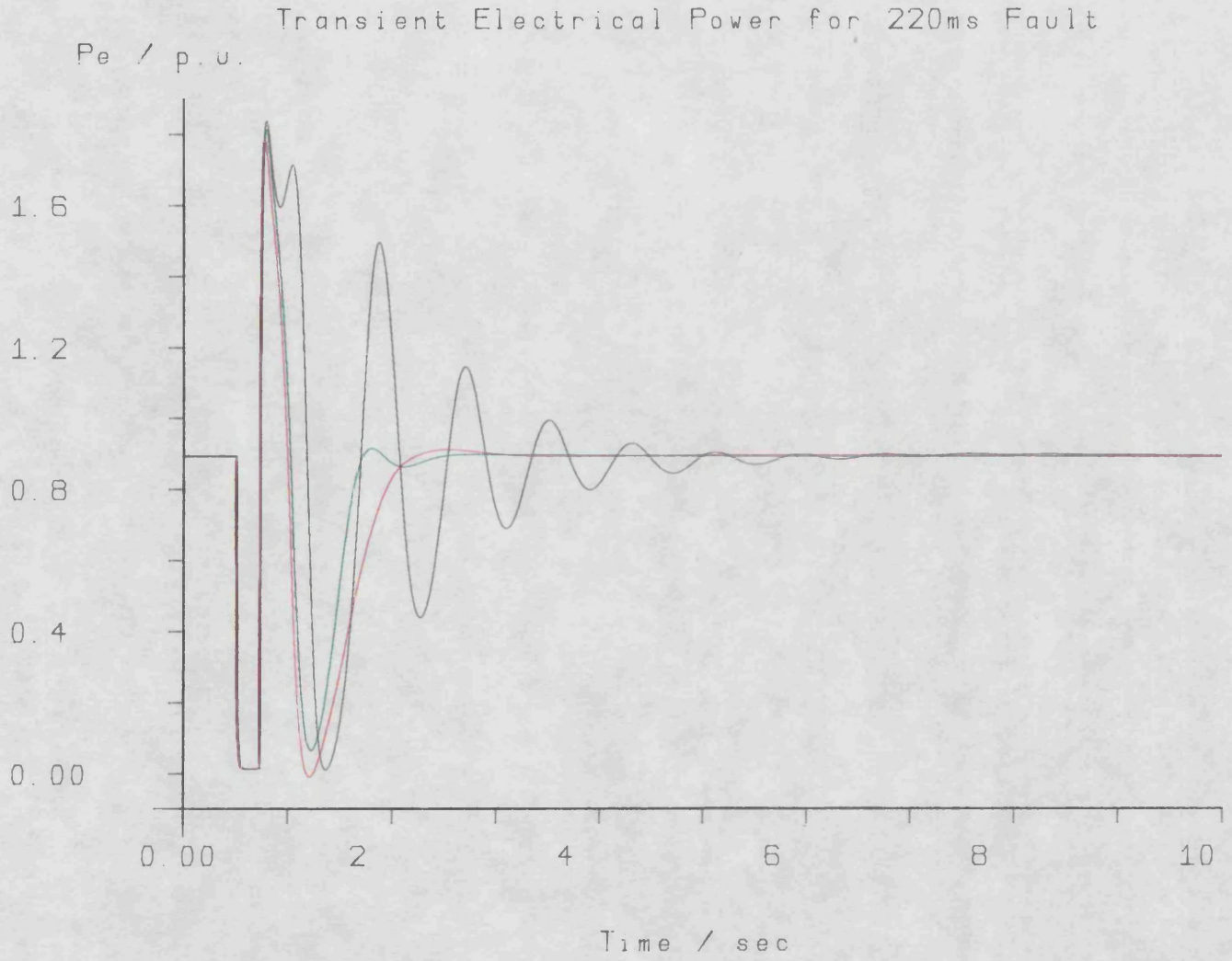

Transient Electrical Power for 220ms Fault
Pe / p.u.
1.6
1.2
0.8
0.4
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.9 (f)

machine to the impedance of the rest of the system as seen from the machine terminals. In practice, due to the high rate of change of stator currents, energy is stored in the magnetic circuits of the machine. This storage of energy causes an initial deceleration of the rotor shaft. However, this effect is neglected by the fifth order machine representation used in this study. Due to the loss in demand for power caused by the fault, the rotor begins to accelerate in order to take up the excess power being supplied by the turbine. The field voltage is driven into its positive limit due to the terminal voltage error, and the presence of acceleration feedback into the excitation system drives the field voltage for controls A2 and A3 into limit slightly quicker. In the base case (control A1), the mechanical torque from the turbine falls due to the feedback of slip speed into the governing system through the droop mode gain K_t. A much larger and quicker drop in the mechanical torque is obtained by the feedback of rotor acceleration into the governor loop. This drop in mechanical torque is responsible for the large reduction in the size of the first peak of the load angle swing. When the fault is cleared, there is an immediate demand for electrical power, which results in a decelerating electrical torque. The magnitude of the resulting deceleration is sufficient to bring the field voltage out of positive limit and drive it into the negative limit. Field voltage reversal following fault clearance enables the energy stored in the increased rotor speed to be transferred to the electrical system, which improves the subsequent damping of load angle oscillations. As the machine passes its initial load angle, following the first peak in its swing, the rotor is again subject to an acceleration, due to the import of synchronising power from the infinite bus. This drives the field voltage back into the positive ceiling and causes the second

reduction in the mechanical torque seen in Fig. 6.9(d) for the A2 and A3 curves as the load angle underswings its original position. This action better co-ordinates the flow of power from the machine and rapidly damps the oscillations in mechanical torque, load angle and electrical power.

It can be seen that the post fault terminal voltage error keeps the conventionally controlled field voltage (A1) in positive limit. This means that it can make no contribution to the control until the terminal voltage is close to the desired steady state value. The excess energy stored in the rotor, due to its increased speed, must be lost in order to return to steady state, and this results in oscillations in the load angle of the conventionally controlled system.

Both sets of acceleration gains show a significant improvement in the terminal voltage recovery and the load angle response. The bang-bang nature of the field voltage response following the fault makes the two responses, A2 and A3, initially similar. However, as the field voltage comes out of limit, the two responses diverge. The lower gain response, A3, tends towards steady state quicker than the higher gain response, A2, and the peak overshoot in terminal voltage occurs much later in the response, at a significantly lower amplitude. The performance index value for A3 is also lower than that for A2 which would indicate that the chosen performance index and weightings are suitable for this system.

Table 6.2 contains the results of optimisation studies made with each of the three signals being used, separately, as extra control

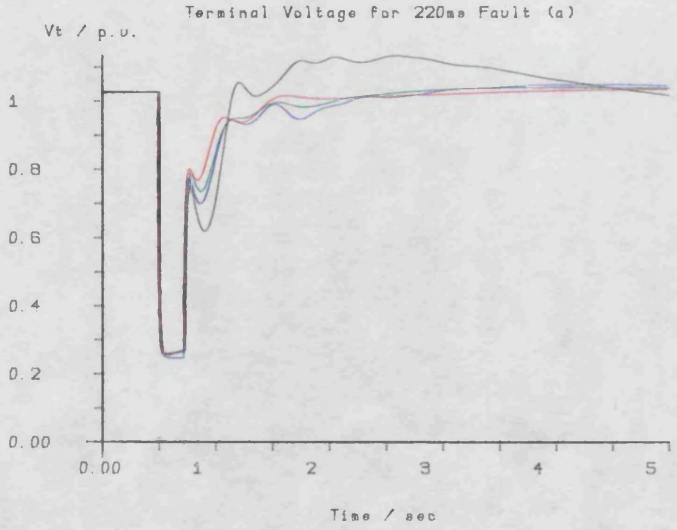
signals. Fig. 6.10 shows the transient responses for terminal voltage, field voltage, load angle and mechanical torque obtained using these controls. In this and all the following graphs, the responses are colour coded according to the feedback signal used. Unless otherwise stated, the colour coding is: Black for no extra feedback, the base case; Red for acceleration feedback; Green for transient electrical power feedback; and Blue for transient direct axis current feedback.

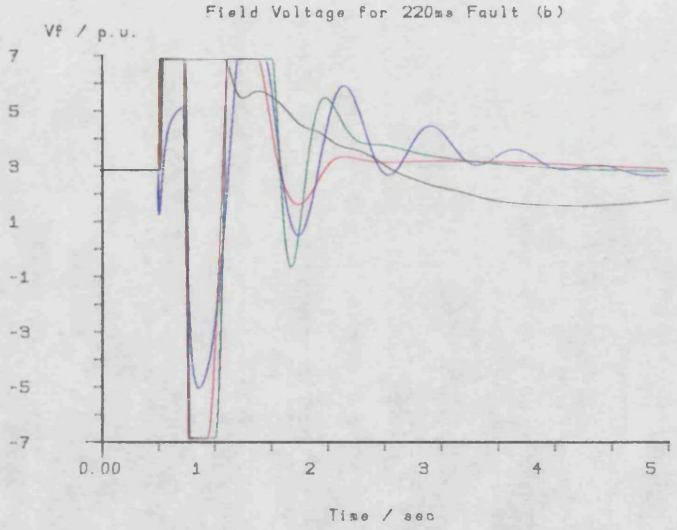
Initial Conditions as Table 6.1

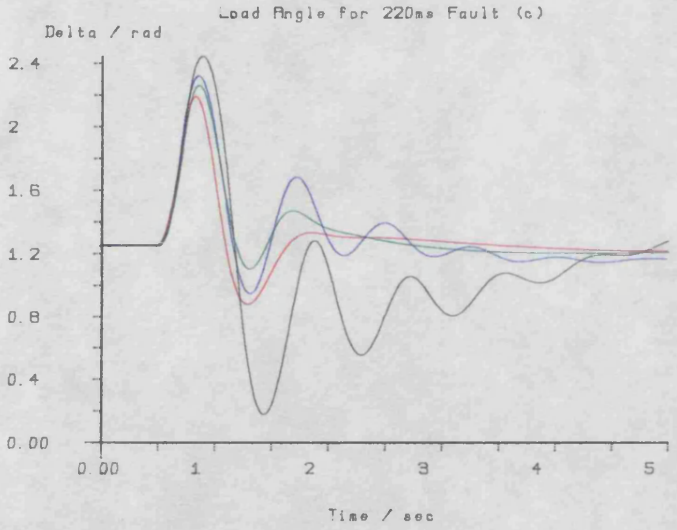
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A1	None	0	0	118	220ms	6.87	-6.87
A3	$P^2\delta$	0.0400	0.301	17.4	220ms	6.87	-6.87
A7	ΔPe	-3.54	-0.385	20.8	220ms	6.87	-6.87
A20	Δid	-0.774	0.325	32.1	220ms	6.87	-6.87

Table 6.2

It is interesting to note that the sign of the gains when using transient power feedback are opposite to those used for acceleration feedback. This is because, as mentioned earlier, under constant mechanical torque conditions, a drop in electrical power will result in a positive acceleration of the rotor. It can be seen that there is almost negligible difference between the acceleration controlled and power controlled terminal voltage responses of Fig. 6.10, the


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
1
2
3
4
5
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
1
2
3
4
5
Time / sec

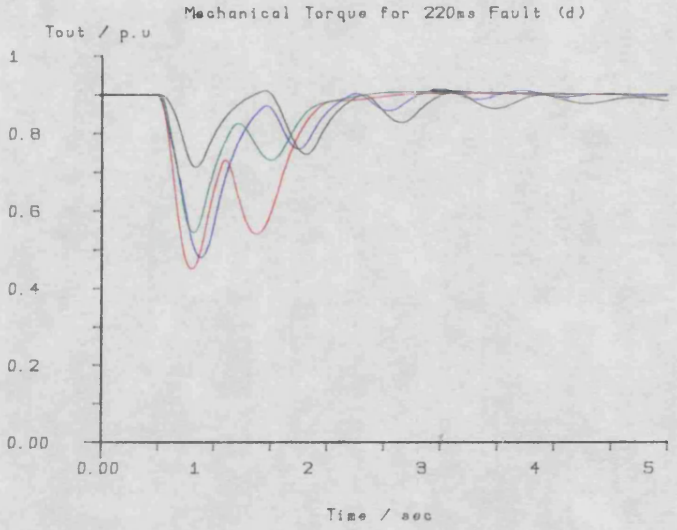

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 5.10

acceleration control giving a slightly quicker return towards steady state. The current controlled terminal voltage response reaches a lower faulted level. This is due to the failure of the field voltage to reach positive saturation.

All three extra feedback signals show a significant improvement in the first swing stability and, using this criteria, acceleration control gives the best improvement, followed by feedback of transient electrical power, and finally, transient direct axis current feedback. The use of either transient electrical power or rotor acceleration feedback results in a well-damped rotor angle response. However, the use of current feedback does not appear to significantly improve the damping of the load angle response following the first swing. It can be seen that the initial fall in mechanical torque using current feedback into the governor is comparable with that obtained by the use of the other two extra signals. When the fault is cleared, the direct axis current seems to have an almost linear dependence on the load angle, which may be seen by comparing Fig. 6.9(c) and Fig. 6.9(e). This may be confirmed by linearising the third order machine equations given in Chapter 3, to give:

$$\Delta i_d = \frac{\Delta e'_q + \Delta\delta V_t \sin\delta_o}{x'_d + x_d}$$

It has already been noted that Lee (6-4) found that the use of transient load angle ($\Delta\delta$) into the governing system made little contribution to improving the damping of rotor oscillations.

The effect of reducing the excitation ceiling voltage and

preventing field voltage reversal on the feedback gains was examined by re-optimising the three sets of gains. The results of these optimisations are summarised in Table 6.3.

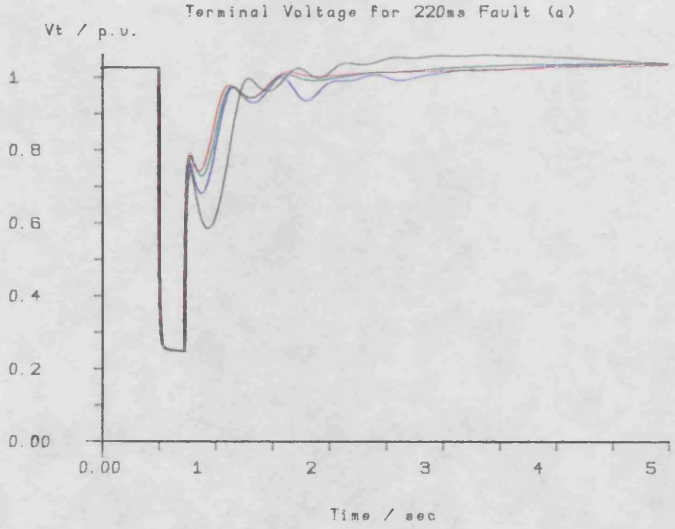
Initial Conditions as Table 6.1

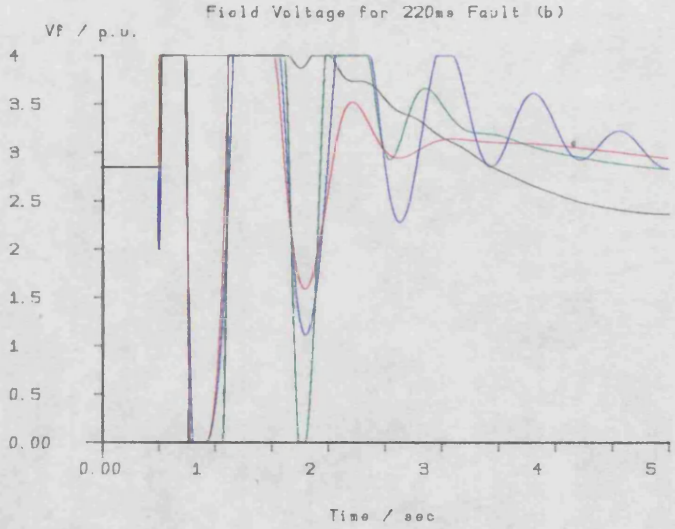
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A5	None	0	0	61.9	220ms	4	0
A6	$p^2\delta$	0.0175	0.0226	23.70	220ms	4	0
A8	ΔPe	-2.14	-0.447	23.73	220ms	4	0
A22	Δi_d	-0.455	0.250	39.4	220ms	4	0

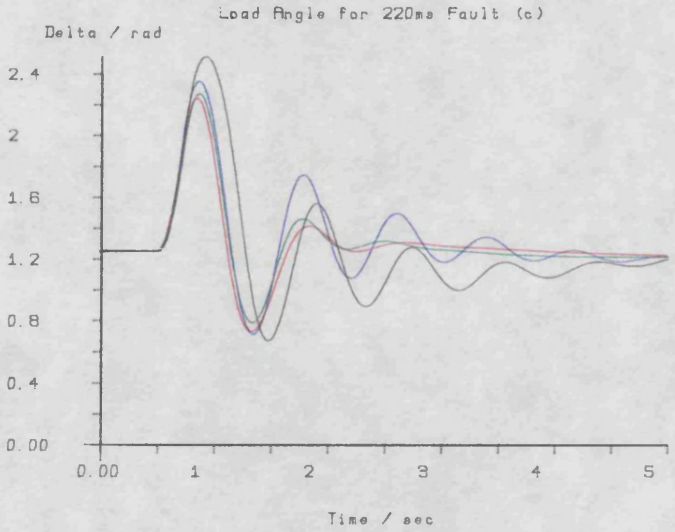
Table 6.3

Transient responses corresponding to these controls are plotted in Fig. 6.11. It can be seen that, despite restricting the excursions in the field voltage, the general trend in improvements, indicated by the previous results, still holds. Note that the base case performance index, control A5, is better than the performance index for the base case with the larger field voltage limits, control A1, but the higher limits permit a larger reduction in the performance index when extra feedback signals are used.

Again, it can be seen that there is little difference between the terminal voltage responses during the early part of the recovery when using acceleration or transient power feedback. With the upper field voltage ceiling reduced, all the field voltages reach saturation and, as a result, the terminal voltage responses all reach the same


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
1
2
3
4
5
Time / sec

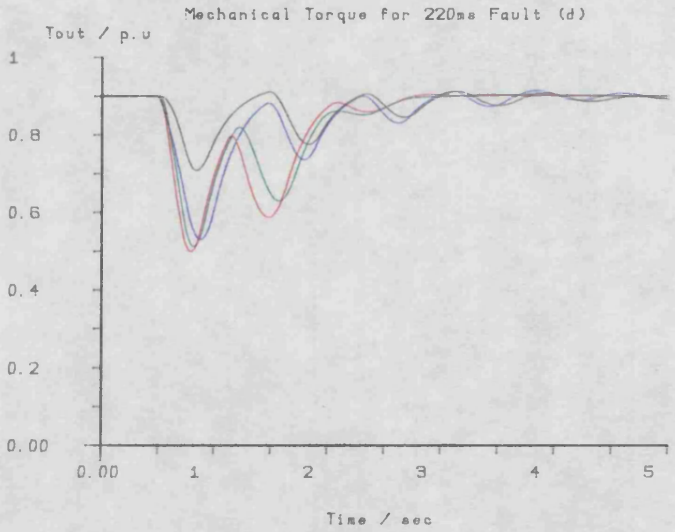

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.11

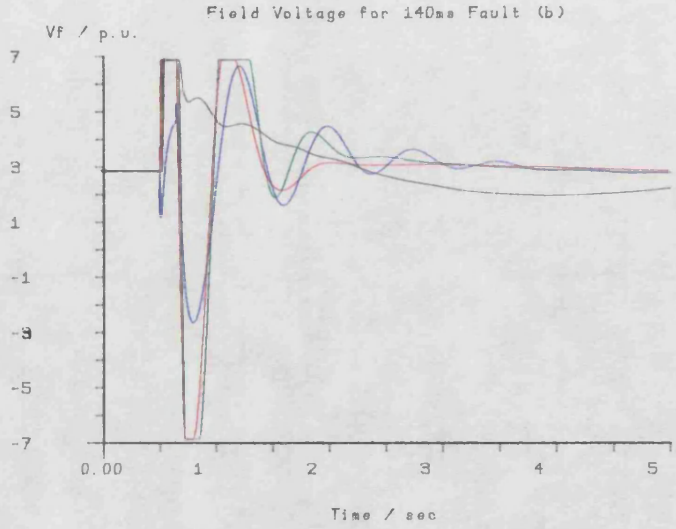
faulted level. The distance between the field voltage limits is somewhat less than half that in the previous case, Table 6.2. This is likely to account for the general reduction in the magnitude of the gains which supply the extra signals to the excitation system, since the bang-bang control may be achieved by small excursions in these signals.

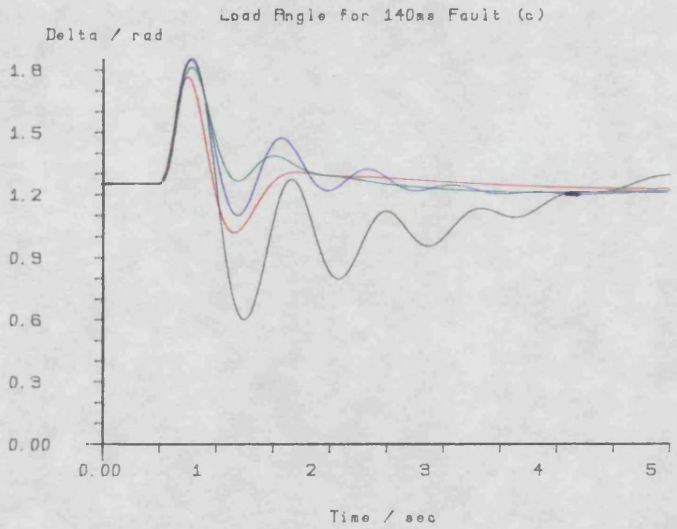
The peak overshoots in the load angle response are slightly larger than those obtained when using higher limits on the field voltage, although the order of improvement in terms of first swing stability is unchanged. The mechanical torque responses using acceleration and transient electrical power are virtually identical. Comparing the torque curves of Fig. 6.11 and Fig. 6.10, it can be seen that the reduction in acceleration gain and the increase in magnitude of the power gain into the governor would account for this similarity.

The effect on the feedback gains of the use of faster circuit breakers and fault clearance equipment is shown in Table 6.4.

The transient responses corresponding to these controls are plotted in Fig. 6.12 (A9, A10, A11 and A21) and Fig. 6.13 (A26, A12, A13 and A25). These show that the improvements in transient performance follow the general trends already discussed, both from the point of view of improvement in first swing stability and the effect of reducing the field voltage limits. One notable result is that the gains for control A3 and for control A10 are very close to each other in parameter space, as are those for controls A6 and A12. This tends to indicate that, with field voltage limits fixed, the


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
1
2
3
4
5
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.8
1.5
1.2
0.9
0.6
0.3
0.00
0.00
1
2
3
4
5
Time / sec

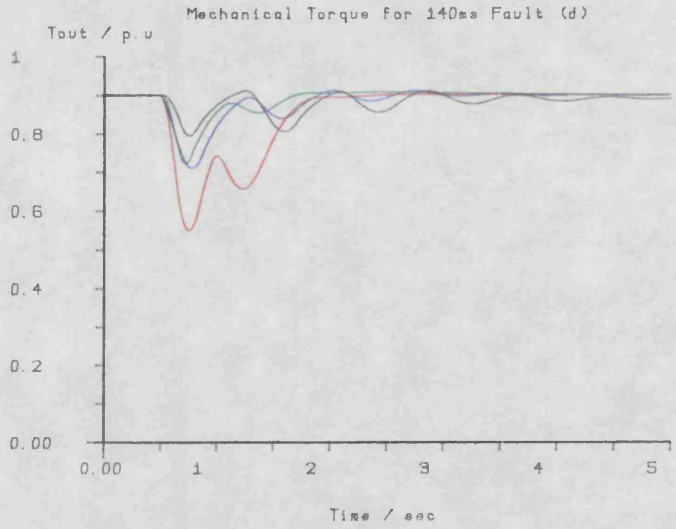
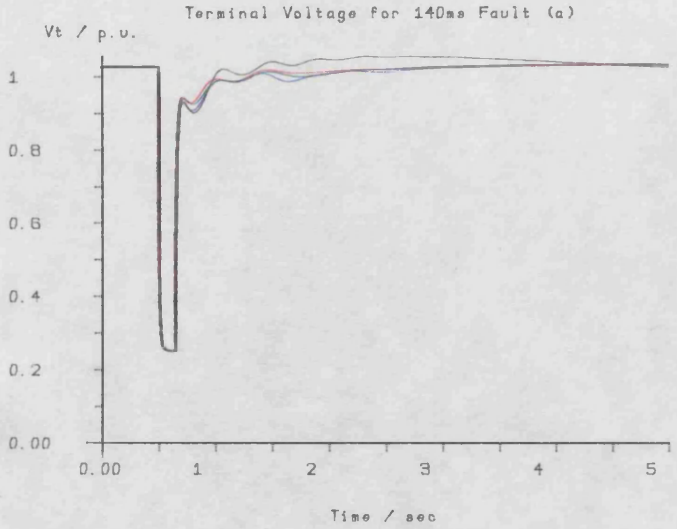
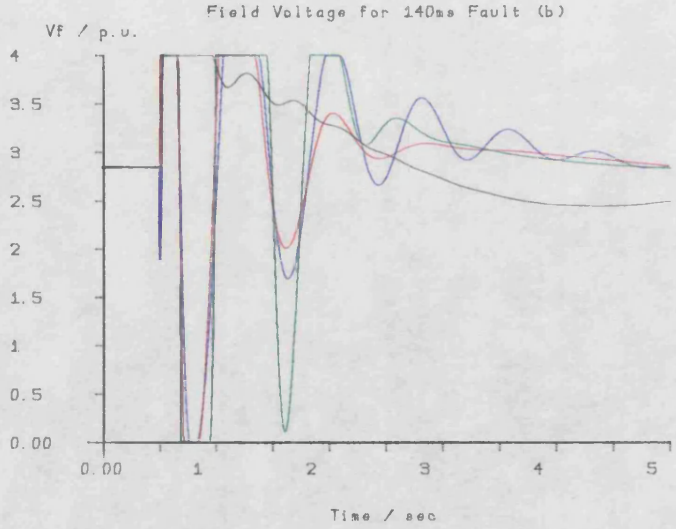
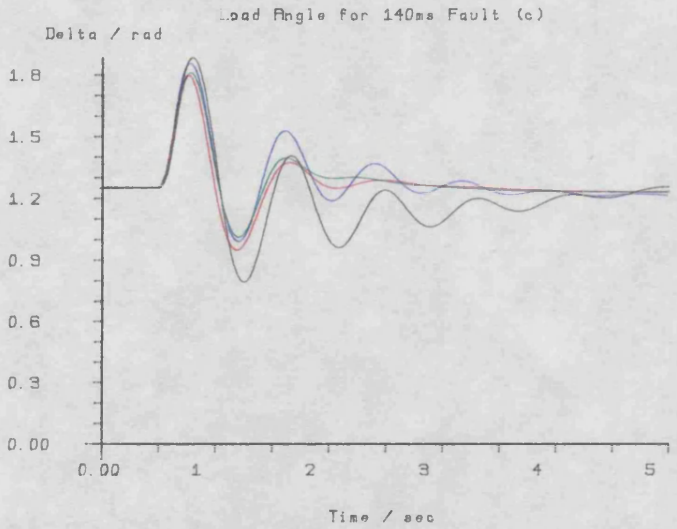

Mechanical Torque for 140ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.12


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.8
1.5
1.2
0.9
0.6
0.3
0.00
0.00
1
2
3
4
5
Time / sec

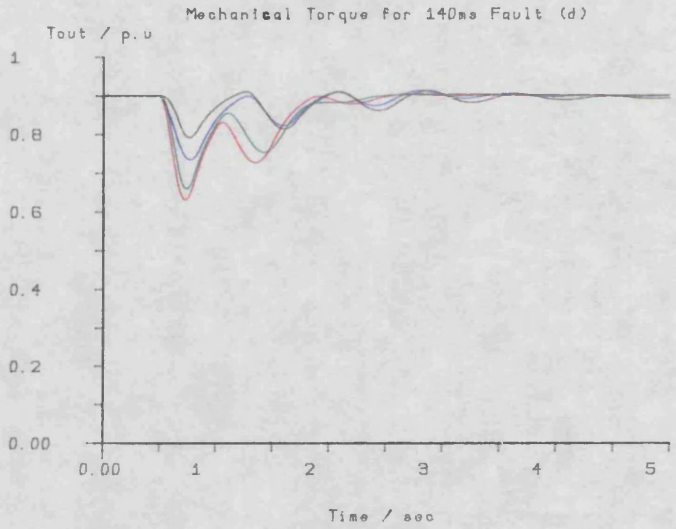

Mechanical Torque for 140ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.13

optimum acceleration gains are reasonably insensitive to changes in fault duration. This observation cannot be made in the case of the other two signals studied.

Initial Conditions as Table 6.1

Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A9	None	0	0	37.5	140ms	6.87	-6.87
A10	$P^2\delta$	0.0362	0.0273	4.73	140ms	6.87	-6.87
A11	ΔPe	-2.58	-0.253	6.27	140ms	6.87	-6.87
A21	Δi_d	-0.801	0.190	8.03	140ms	6.87	-6.87
A26	None	0	0	18.25	140ms	4	0
A12	$P^2\delta$	0.0171	0.0174	6.27	140ms	4	0
A13	ΔPe	-2.74	-0.399	6.18	140ms	4	0
A25	Δi_d	-0.512	0.132	9.41	140ms	4	0

Table 6.4

Initial Conditions

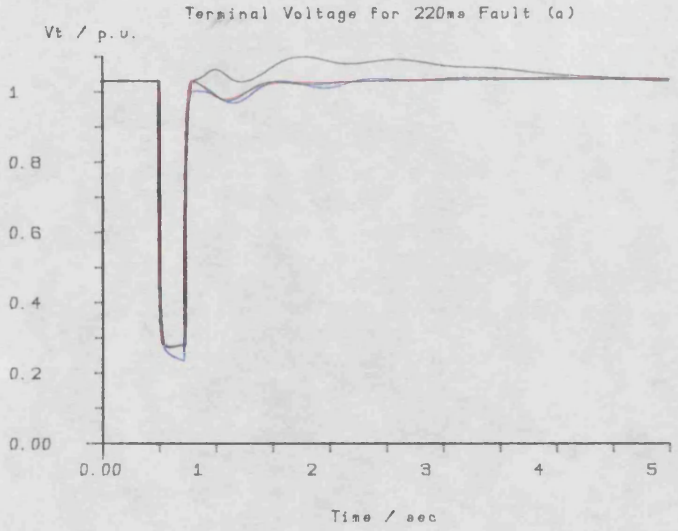
δ = 0.894 rad Vt = 1.03 p.u. P = 0.495 p.u. Q = 0.0591 p.u.

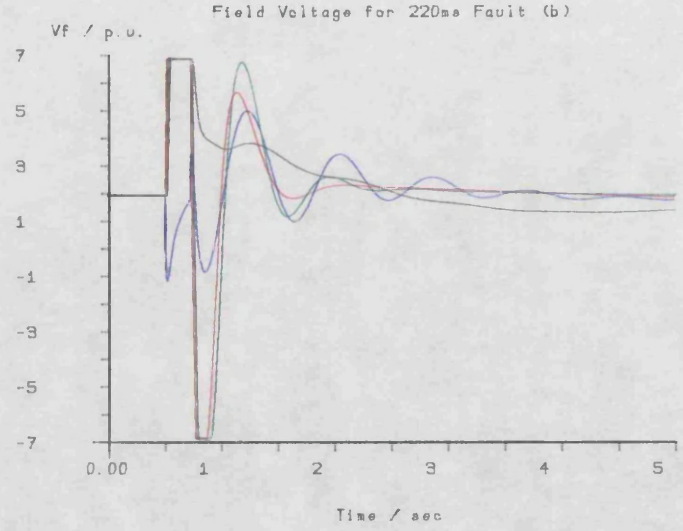
V_f = 1.93 p.u. Power factor = 0.993 lagging

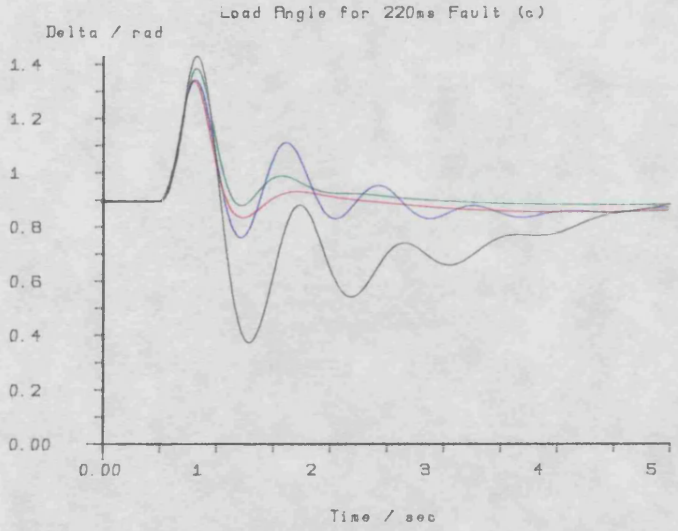
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A14	None	0	0	30.3	220ms	6.87	-6.87
A15	$p^2\delta$	0.0471	0.0201	3.97	220ms	6.87	-6.87
A16	ΔPe	-2.16	-0.281	4.18	220ms	6.87	-6.87
A24	Δid	-0.696	0.324	6.08	220ms	6.87	-6.87

Table 6.5

The effect on the feedback gains of reducing the generator output power is shown in Table 6.5. The corresponding transient responses are given in Fig. 6.14. It can be seen again that the terminal voltage recovery responses using acceleration feedback and transient power feedback are very similar. In this case, the feedback of transient direct axis current offers a better reduction in the first peak of the load angle swing than feedback of transient electrical power, although the subsequent response is much less damped. Also the torque reduction obtained with the use of transient direct axis current is by far the greatest. However, in this case, there is virtually no negative forcing of the field voltage subsequent to clearance of the fault and little power is absorbed by the electrical system of the machines. This failure to absorb power results in a larger first swing in load angle than might be expected from


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
1
2
3
4
5
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

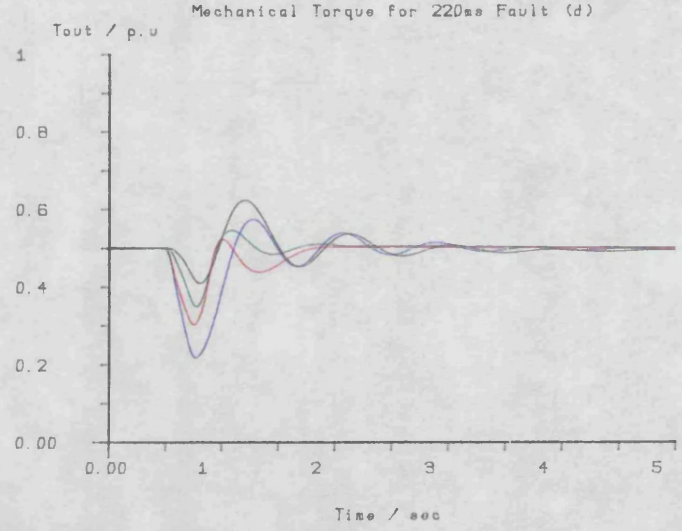
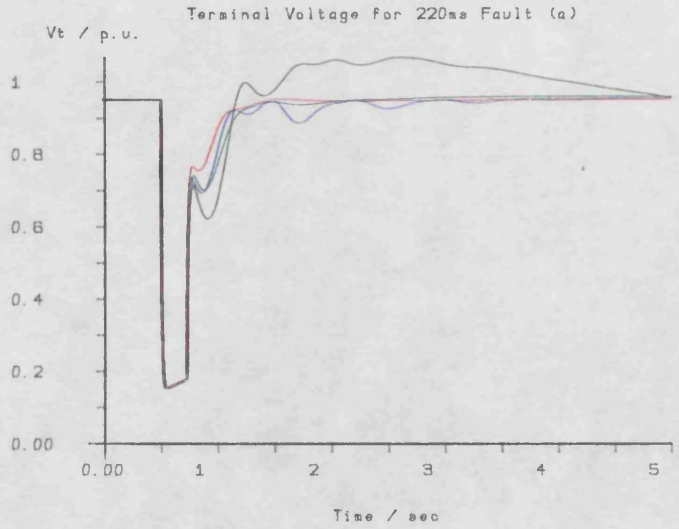

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

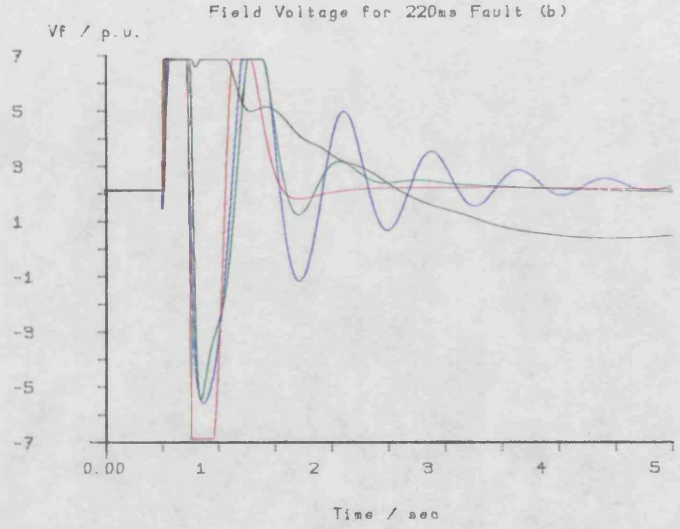
Fig 6.14

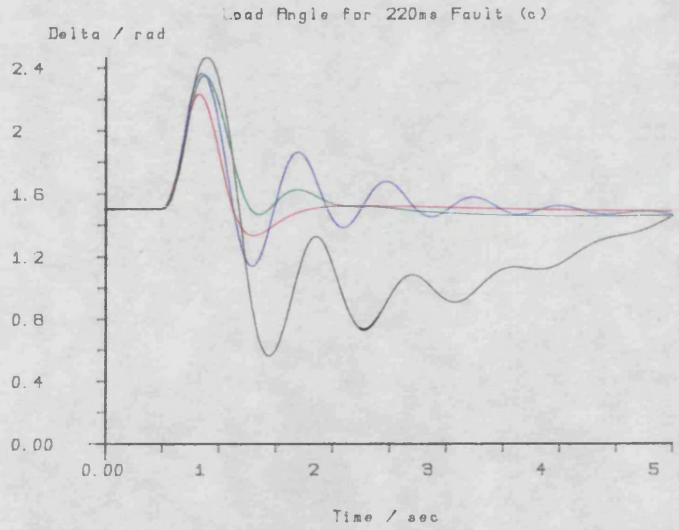
inspection of the torque response alone. It is also worth noting the presence of the larger overshoot in the torque response of Fig. 6.14 compared with that of Fig. 6.10. The torque response of Fig. 6.10 is 'clipped' by the position limits placed on the steam turbine valves, whereas, when working at about 0.5 p.u. power, the valves will be about half-open and so be much further from the fully-open position limit.

Feedback of rotor acceleration still provides the best improvement judged on a first swing basis, and it also furnishes a well-damped load angle response. In the case of transient direct axis current feedback, the field voltage again fails to reach positive saturation during the fault, and hence the faulted terminal voltage deviates from the faulted terminal voltage of the other cases shown in Fig. 6.14. A comparison of Tables 6.5 and 6.2 shows that, particularly in the case of acceleration feedback, there is very little change in the corresponding optimal gains and, although there is a fairly significant change in the gains associated with the feedback of transient electrical power, in both tables the gain into the excitation system is approximately an order of magnitude higher than that into the governing system.

In addition to reducing the output power of the machine, a set of optimisations was performed at a leading power factor. This involves under-exciting the generator by reducing the reference for the excitation system, and reducing the power output to offset the increase in steady state load angle which results from the under-excitation. The resulting gains are given in Table 6.6 and the transient responses are plotted in Fig. 6.15.


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
1
2
3
4
5
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
1
2
3
4
5
Time / sec

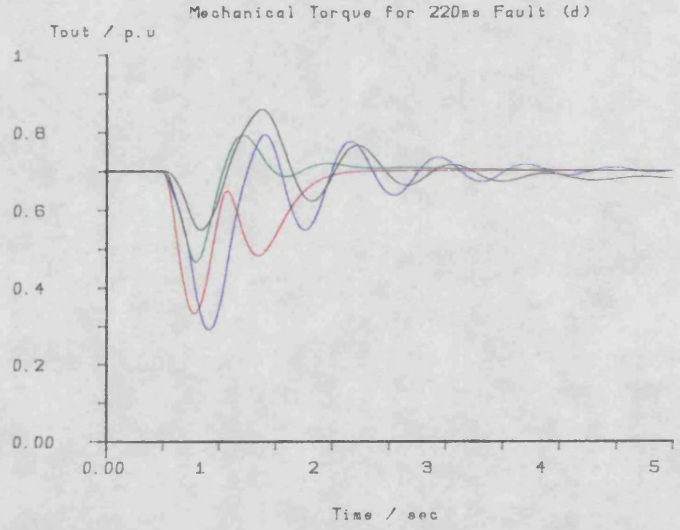

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.15

Initial Conditions

δ = 1.50 rad Vt = 0.949 p.u. P = 0.689 p.u. Q = −0.293 p.u.
V_f = 2.12 p.u. Power factor = 0.920 leading

Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
A17	None	0	0	167	220ms	6.87	−6.87
A18	$p^2\delta$	0.0586	0.0331	9.22	220ms	6.87	−6.87
A19	ΔPe	−1.85	−0.283	14.5	220ms	6.87	−6.87
A23	Δid	−0.870	0.444	20.5	220ms	6.87	−6.87

Table 6.6

It can be seen from Fig. 6.15 that the use of acceleration feedback gives the best improvement in the first swing stability and a slightly quicker terminal voltage recovery than use of the other two extra feedback signals. In terms of first swing stability, transient power feedback gives only a slight improvement over transient direct axis current feedback but, in this case, damping after the first swing is much better. Transient power feedback also gives a smoother terminal voltage recovery response than transient current feedback. It can also be noted that only in the case of acceleration feedback does the field voltage reach its negative limit after the fault has been cleared. In all three cases, the field voltage reaches its positive limit during the fault. Feedback of rotor acceleration into the governor, as in previous cases, gives a rapid reduction in mechanical torque, whereas the torque responses using power and current feedbacks are significantly slower in removing the mechanical torque.

The feedback gains of Table 6.6 have the same orders of magnitude as those in previous tables. However, none are particularly close in parameter space to any of the previous values.

As a result of the optimisations performed in this section, it is felt that the use of rotor acceleration as an extra feedback signal, available to both the excitation and governing systems, gives the best improvement in the transient performance of the system. It appears that the optimum acceleration gains are most sensitive to changes in the field voltage limits, but are reasonably tolerant of changes in fault duration and output power. This may be seen by comparing the lower half of Table 6.4 with Table 6.3, and comparing the upper half of Table 6.4 with Table 6.2. For this reason, a number of extra transient responses follow with rotor acceleration taken as the extra feedback signal. Each set of responses consists of a base case, in black, with both gains set to zero; an optimal case, in red, with both gains set to their optimised values; and a third case, in green, where the gains have been set to values optimised for a 220ms fault clearance time with a high ceiling excitation system and initial conditions as given in Table 6.1. The gains in this latter case are those used with control A3 in Table 6.1. Table 6.7 contains a summary of these transient responses, Fig. 6.16 to Fig. 6.20.

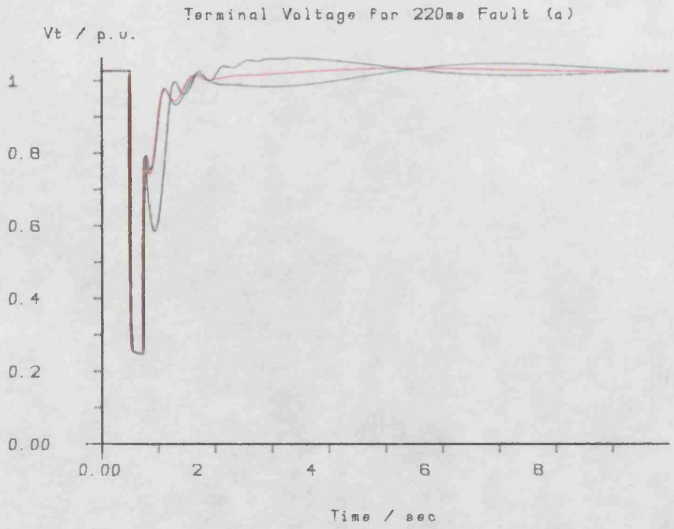
Kavr = 0.0400, Kgov = 0.0301 in all cases C1–C5.

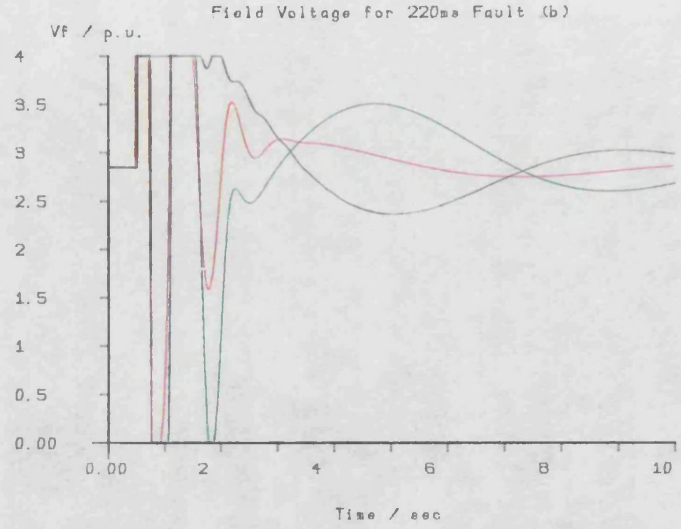
Control	Initial Conditions	P.I.	Fault Time	V_{fmax}	V_{fmin}	Fig.	Base Case	Optimal Case
C1	)	38.7	220ms	4	0	6.16	A5	A6
	) As							
C2	) Table	4.93	140ms	6.87	-6.87	6.17	A9	A10
	) 6.1							
C3	)	14.6	140ms	4	0	6.18	A26	A12
C4	As Table 6.5	9.93	220ms	6.87	-6.87	6.19	A14	A15
C5	As Table 6.6	25.0	220ms	6.87	-6.87	6.20	A17	A18

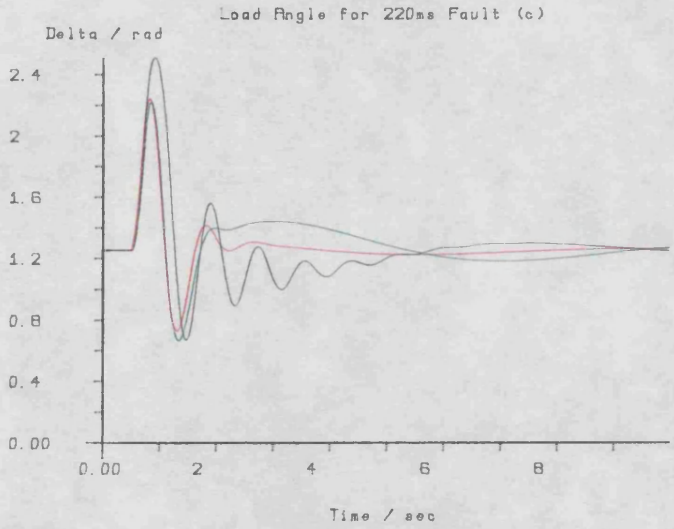
Table 6.7

Inspection of Figs. 6.17, 6.19 and 6.20, corresponding to the cases with the high ceiling excitation system with negative field forcing, shows that the transient response using the optimal gain values, in red, and that using gain values optimised for control A3, in green, are very similar. This is particularly noticeable for control C2 in Fig. 6.17, where the only change from the conditions of A3 is a reduction in the fault clearance time. This is not unexpected, as it has been noted earlier that the optimal gains obtained for controls A3, A10 and A15 are very similar. Also, despite the rather large difference in the optimal gains between controls A18 and A3, the responses in Fig. 6.20 show that control C5 has only a marginally different transient response from control A18, which uses optimised gains.

Figs. 6.16 and 6.18 show transient responses for the cases with reduced field voltage limits, C1 and C3, respectively. It can be seen that the initial terminal voltage recoveries when using the optimum


Terminal Voltage For 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
10
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
2
4
6
8
Time / sec

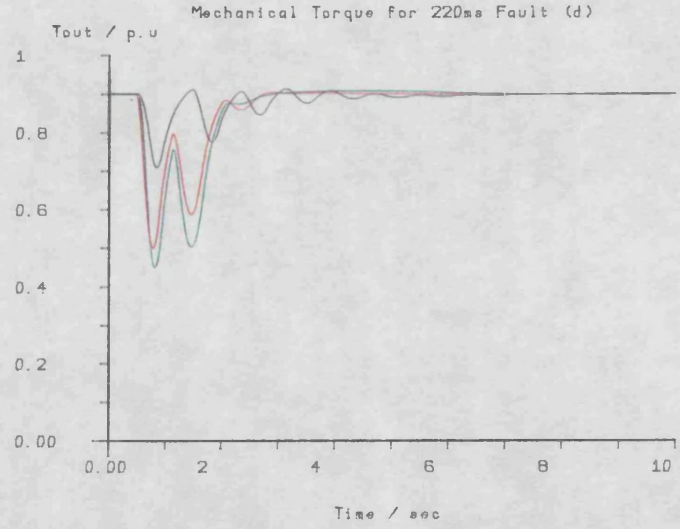
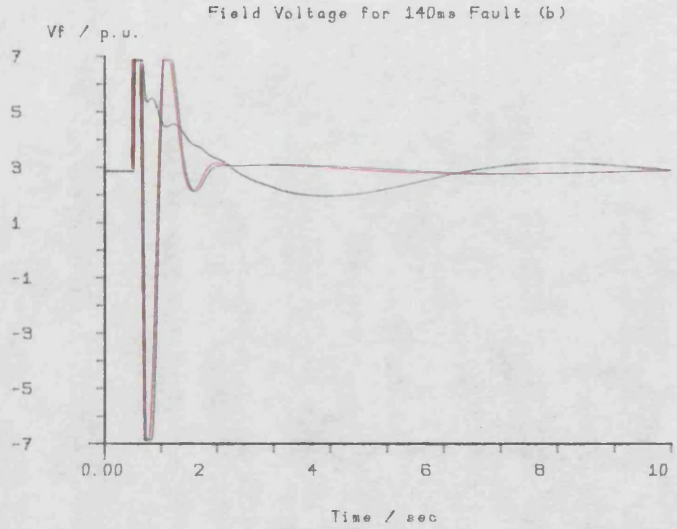
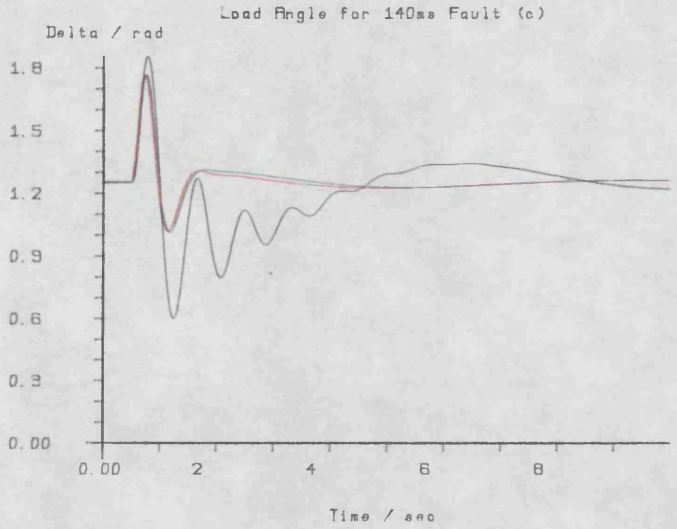

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.16


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
10
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.8
1.5
1.2
0.9
0.6
0.3
0.00
0.00
2
4
6
8
Time / sec

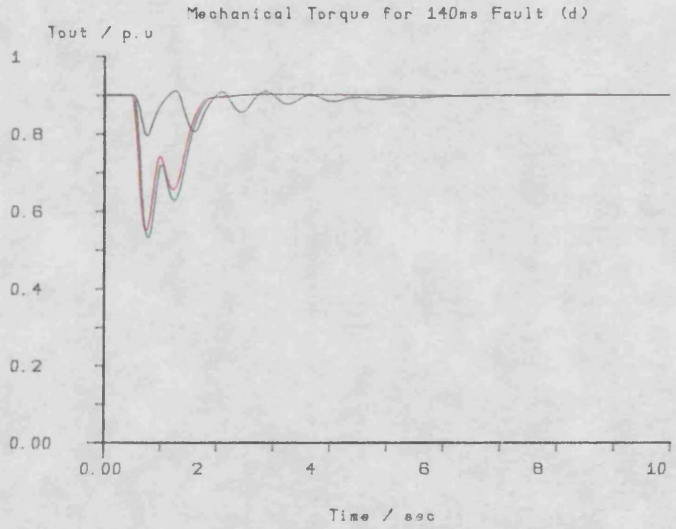
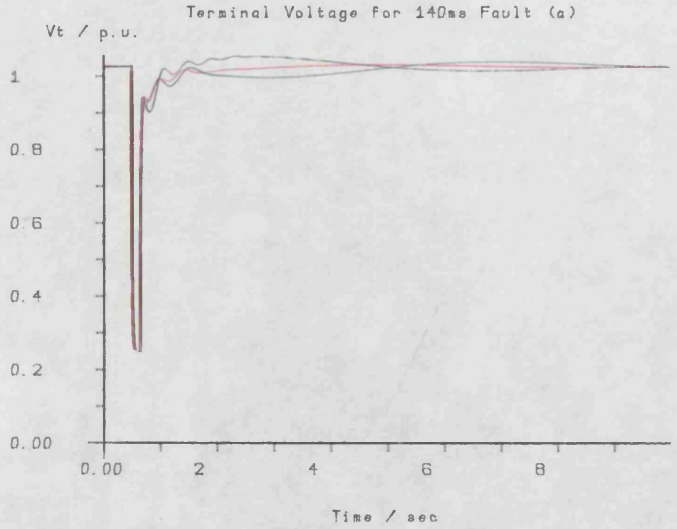
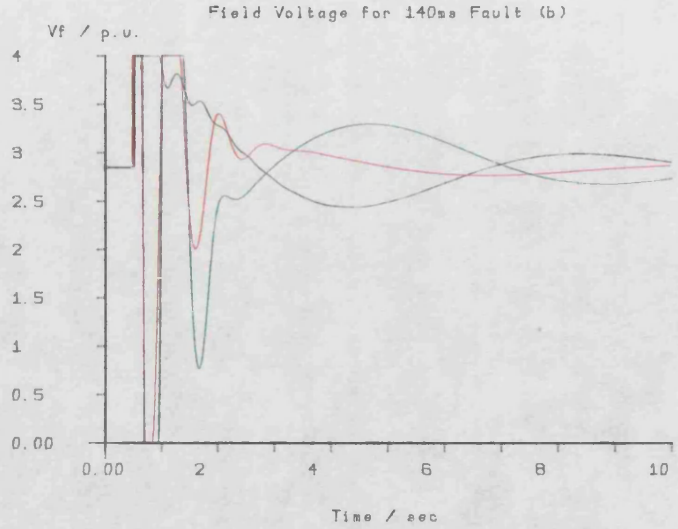
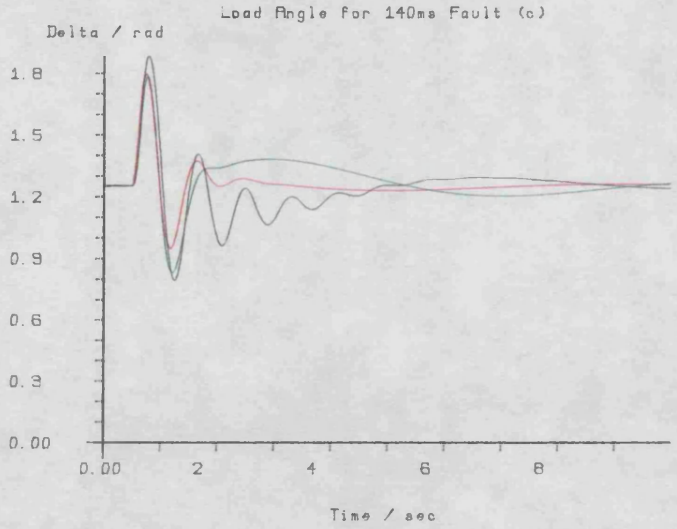

Mechanical Torque for 140ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.17


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
10
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.8
1.5
1.2
0.9
0.6
0.3
0.00
0.00
2
4
6
8
Time / sec

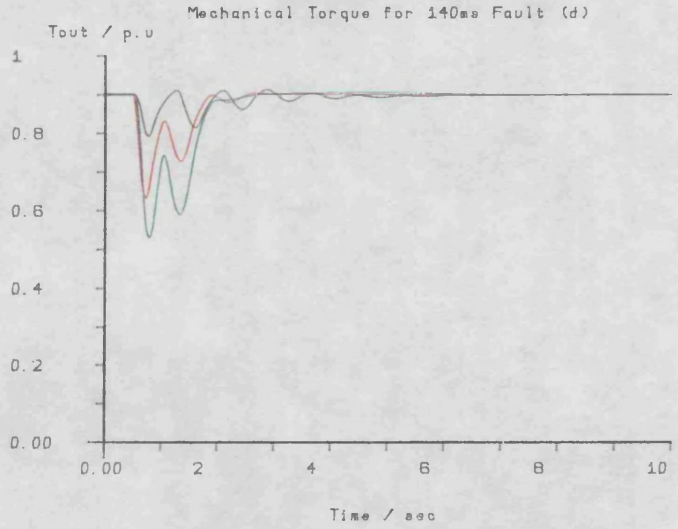
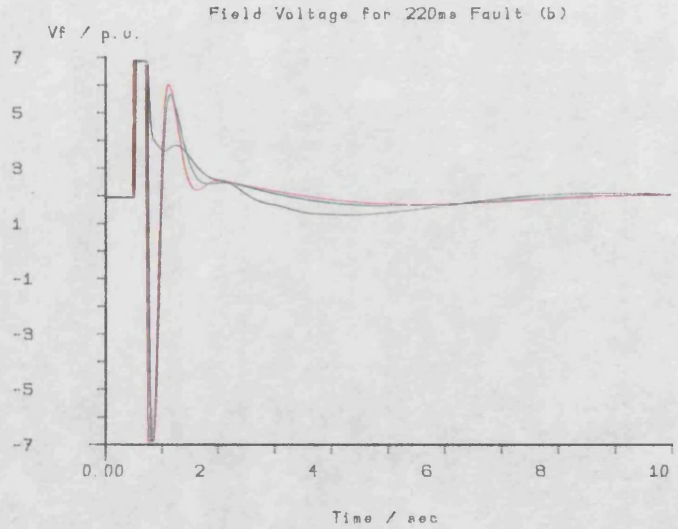
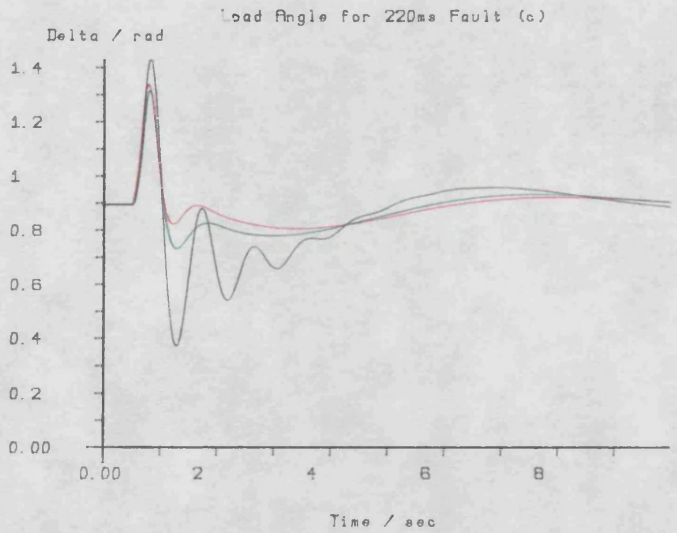

Mechanical Torque for 140ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.18


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
10
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

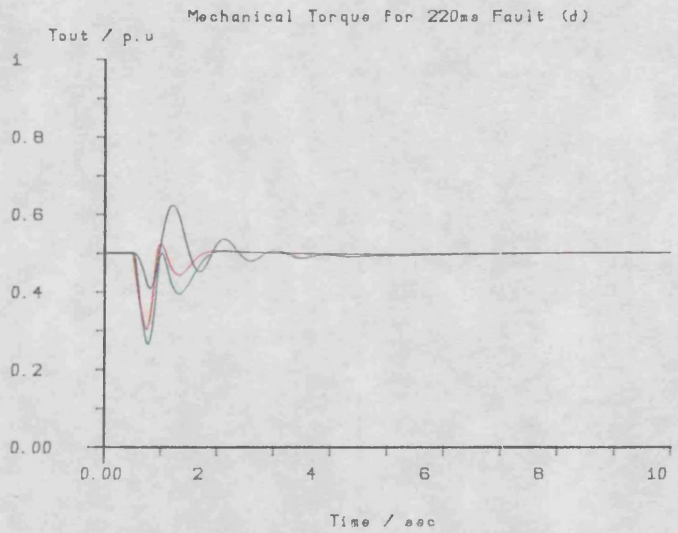
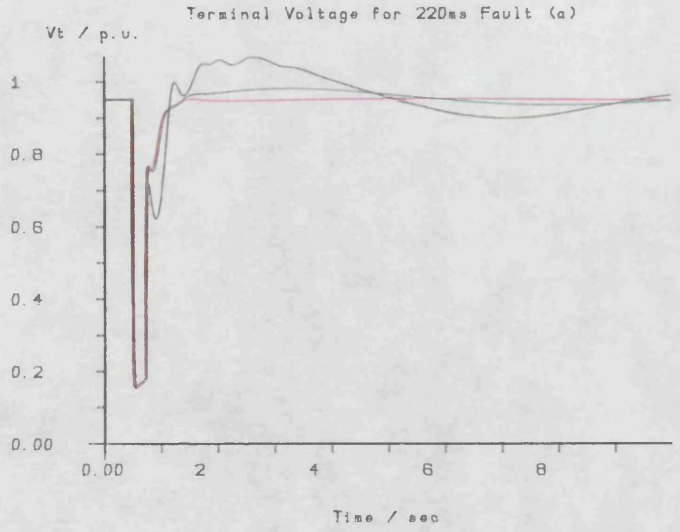
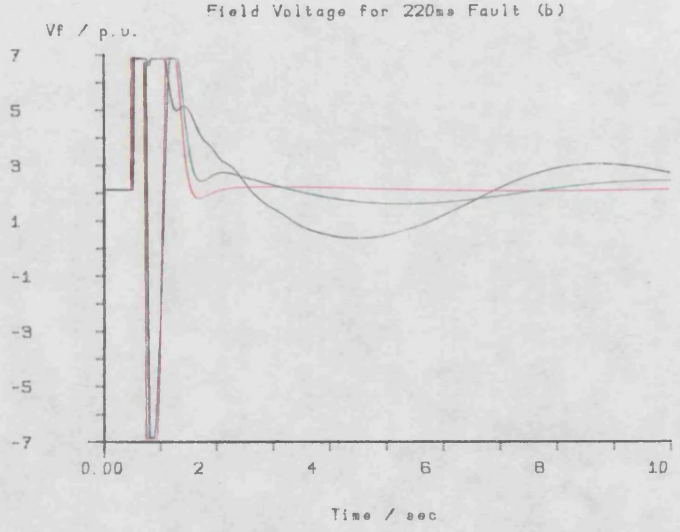
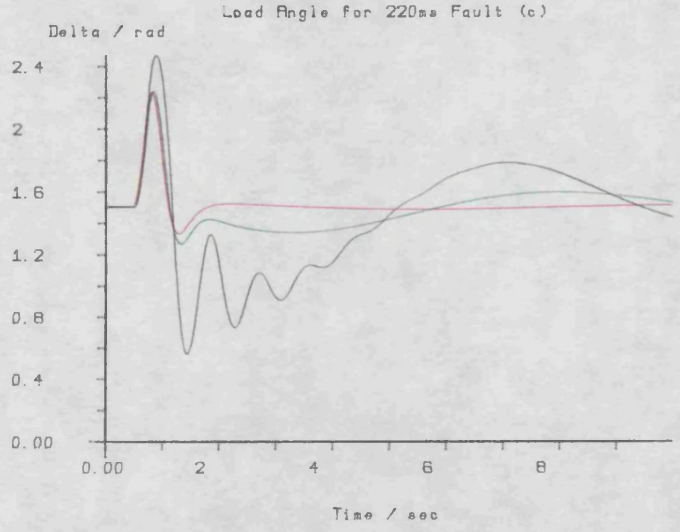

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.19


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
10
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.4
2
1.6
1.2
0.8
0.4
0.00
0.00
2
4
6
8
Time / sec

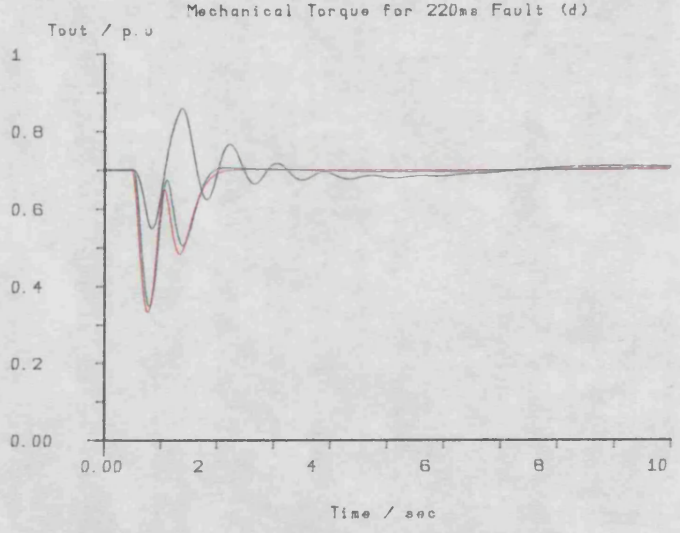

Mechanical Torque for 220ms Fault (d)
Tout / p.u
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
10
Time / sec

Fig 6.20

gains, (red), differ very little from those using the gains from A3. However, about 1.5 seconds after the fault has been applied, the terminal voltage responses tend to drift apart. A similar observation may be made for the load angle responses and, although there is little to determine which are the 'best' terminal voltage responses, the load angle responses for C1 and C3, in green, are much slower to return to steady state than those of A6 and A12. Despite these differences, all controls C1–C5 offer an improvement over the base cases in black, and all achieve reductions in the first peak in the load angle swing comparable with the use of gains optimised for the particular conditions.

A further series of optimisations have been carried out using acceleration feedback over a wider range of power variations. These results are given in Table 6.8.

Govref/P.U.	P/p.u.	Q/p.u.	Power factor	(Angle)	Kavr	Kgov
0.9	0.884	-0.0488	0.998	(-3°.16)	0.0400	0.0301
0.8	0.786	-0.000380	1.00	(0°)	0.0473	0.0273
0.7	0.690	0.0367	0.999	(3°.04)	0.0639	0.0219
0.6	0.593	0.0320	0.999	(3°.09)	0.0723	0.0231
0.5	0.495	0.0591	0.993	(6°.81)	0.0471	0.0201
0.4	0.397	0.0787	0.981	(11°.2)	0.0457	0.0151
0.3	0.297	0.0971	0.950	(18°.1)	0.0262	0.00348
0.2	0.199	0.108	0.879	(28°.5)	0.0168	0.00564
0.1	0.0990	0.131	0.603	(53°.0)	0.00305	0.00319

Table 6.8

It can be seen that these gains vary in magnitude from about 0.04 to 0.07 and 0.03 to 0.015 into the excitation and governing systems respectively, for governor reference settings of 0.4 p.u. and above. Below this value, the optimum gains reduce rapidly since the improvements that can be made in the transient response at these power levels is small. Most of the gains have values of about 0.045 into the excitation system and 0.025 into the governor. From the previous discussion, gains of this order would be suitable for a wider range of operating conditions.

A method of comparing the improvements in transient stability was introduced in Chapter 5. It was shown that pre-fault/post-fault impedance curves could be used to indicate the relative merits of various methods of improving transient stability. Pre-fault/post-fault impedance curves have been plotted for control strategies using controls A3, A7 and controls A10, A11. These correspond to use of rotor acceleration or transient electrical power as extra feedback signals, with fault clearance times of 220ms and 140ms. These curves are plotted in Fig. 6.21 and Fig. 6.22. Fig. 6.21 shows the curves for conventional control with no extra feedback and fault clearance times of 220ms and 140ms in black and red, respectively. The curve corresponding to the use of acceleration feedback and a fault clearance time of 220ms, control A3, is given in green and that corresponding to the use of transient electrical power feedback with the same fault clearance time, control A7, is plotted in blue. It can be seen from Fig. 6.21 that the use of rotor acceleration as an extra feedback signal gives an improvement in transient performance which is equivalent to about 75% of the improvement obtained using conventional control with the fault

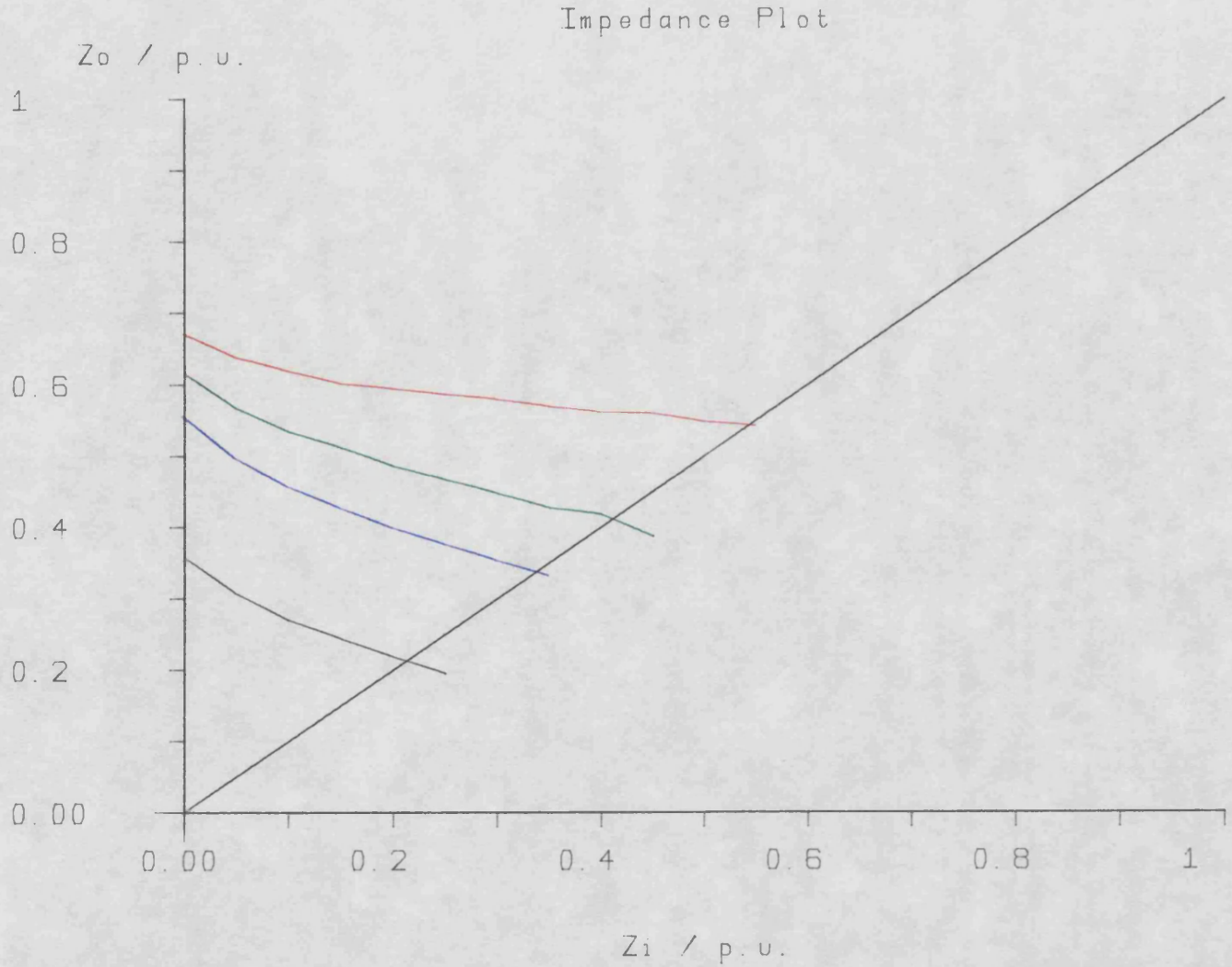


Fig 6.21

clearance time reduced from 220ms to 140ms. The use of transient electrical power as an additional feedback does not produce such a large performance improvement, and this improvement is equivalent to about 50% of that obtained by reducing the fault clearance time of the conventionally controlled system. Fig. 6.22 consists of a similar set of curves. The base cases, in black and red, are for a conventionally controlled system with fault clearance times of 140ms and 80ms respectively, and the green and blue curves were obtained by the use of rotor acceleration and transient electrical power as extra feedback signals, respectively, with a fault clearance time of 140ms. These latter two curves correspond to the use of controls A10 and A11.

It can be seen from Fig. 6.22 that, with zero pre-fault impedance, the use of either extra feedback signal gives an improvement in transient performance equivalent to that obtained by reducing the conventionally controlled fault clearance time from 140ms to 80ms. At the point where the post-fault and pre-fault impedances are identical, reduction of the fault clearance time gives a clear improvement over the use of rotor acceleration as an extra feedback, which, in turn, gives a better improvement than the use of transient electrical power feedback.

Finally, it is necessary to give some indication of the effect the use of these extra feedback signals may be expected to have on the dynamic or small change stability of the generating plant.

Fig. 6.23 consists of dynamic responses obtained when a 10% reduction is made in the excitation system reference. Such a change

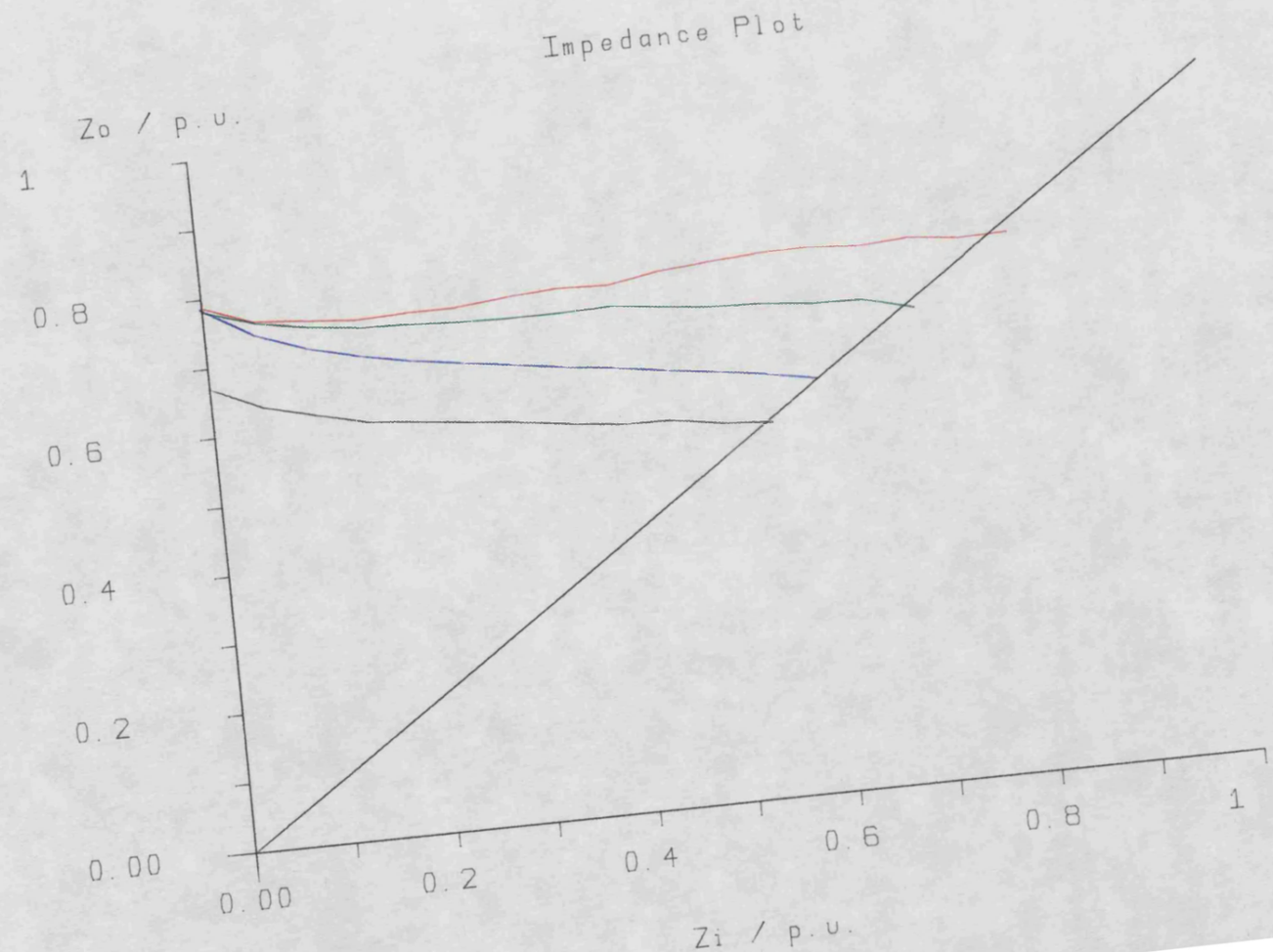
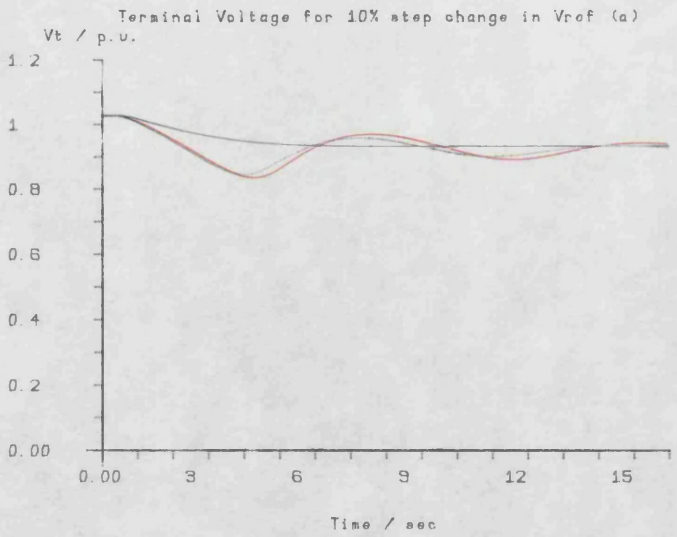


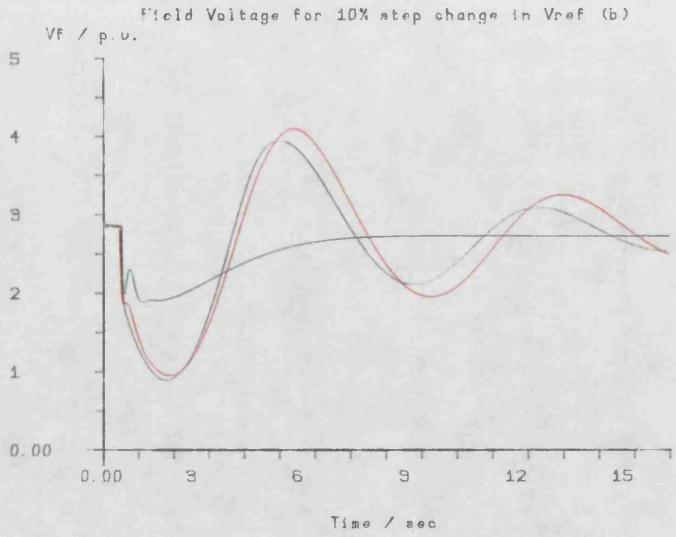
Fig 6.22

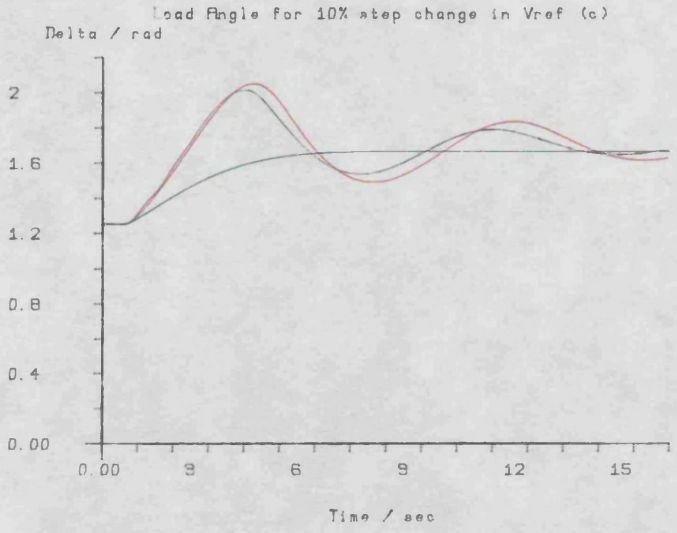
will tend to under-excite the generator and cause it to generate at a larger steady state load angle, closer to the point where stability may be lost. These responses correspond to a base case, A1 in black, where no extra feedback is applied, control A3 in red, where rotor acceleration is fed into both the excitation and governing systems, and control A7 in green, where transient electrical power is fed into both control loops. Fig. 6.24 and Fig. 6.25 are similar to Fig. 6.21, except that the extra feedback signal is made available to only one of the control loops at a time.

The extra feedback signal is made available to the excitation system only in Fig. 6.24, and to the governing system only in Fig. 6.25. It can be seen from Figs. 6.23 - 6.25 that feedback of transient electrical power into the excitation system, alone or in combination with the governing system, significantly damps both the terminal voltage response and the load angle response. Feedback of rotor acceleration in the same manner produces a slightly more oscillatory response than the base case. Feedback of either rotor acceleration or transient electrical power into the governing system alone, Fig. 6.25, produces no significant change in the response of terminal voltage and load angle to the change of reference. These responses suggest that the use of transient electrical power feedback into the excitation system will significantly improve the dynamic stability of the generator, whilst feedback of rotor acceleration will slightly degrade this stability. Fig. 6.26 confirms this by plotting the dynamic stability limit for the three controls, A1, A3 and A7, in black, red and green, respectively.

It may be concluded that, although use of rotor acceleration as a


Terminal Voltage for 10% step change in Vref (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec


Field Voltage for 10% step change in Vref (b)
Vf / p.u.
5
4
3
2
1
0.00
0.00
3
6
9
12
15
Time / sec


Load Angle for 10% step change in Vref (c)
Delta / rad
2
1.6
1.2
0.8
0.4
0.00
0.00
3
6
9
12
15
Time / sec

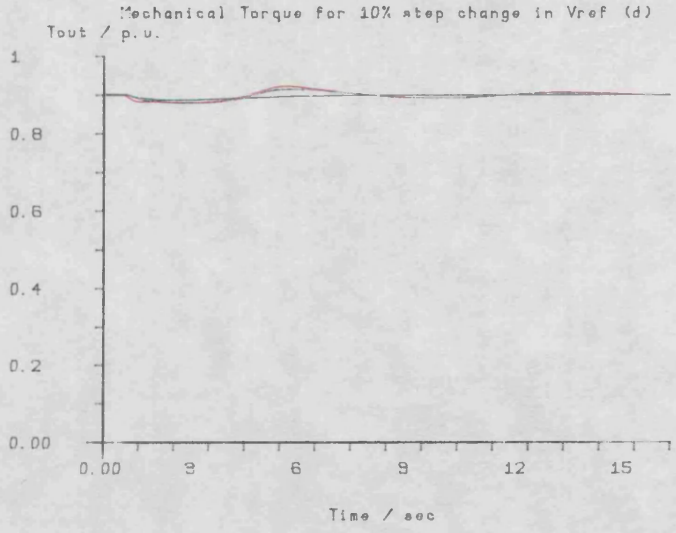
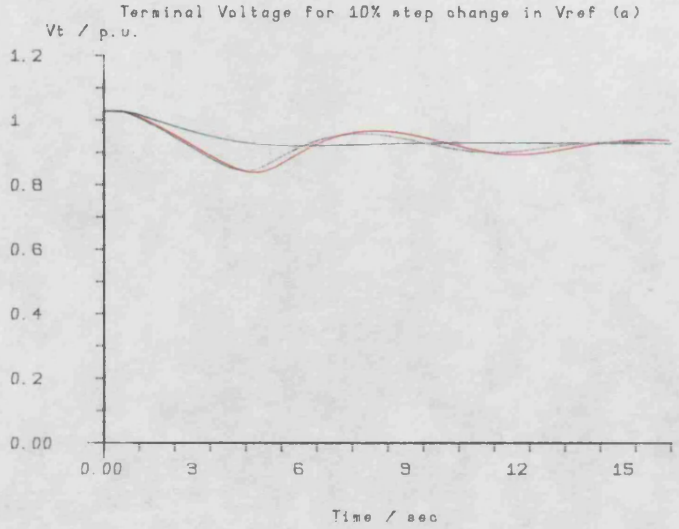
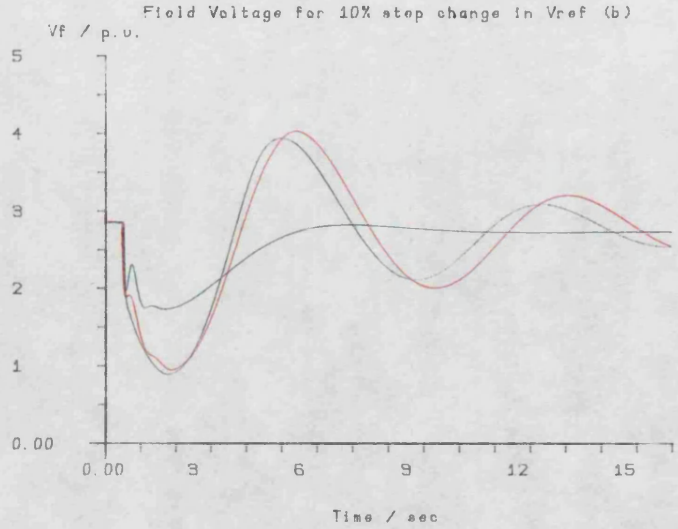
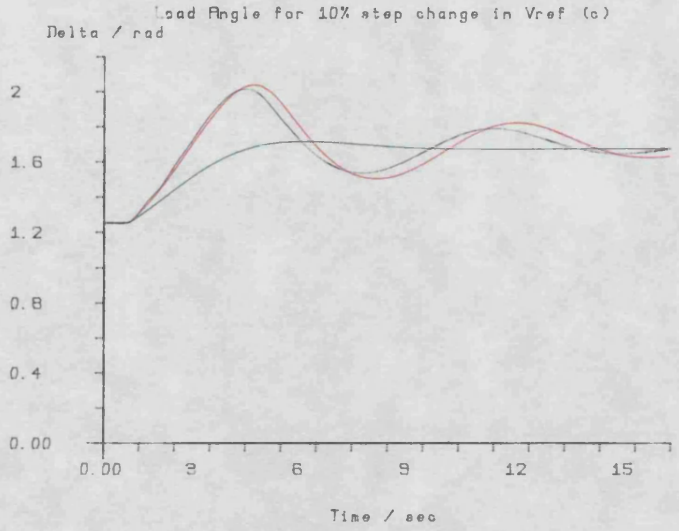

Mechanical Torque for 10% step change in Vref (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec

Fig 6.23


Terminal Voltage for 10% step change in Vref (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec


Field Voltage for 10% step change in Vref (b)
Vf / p.u.
5
4
3
2
1
0.00
0.00
3
6
9
12
15
Time / sec


Load Angle for 10% step change in Vref (c)
Delta / rad
2
1.6
1.2
0.8
0.4
0.00
0.00
3
6
9
12
15
Time / sec

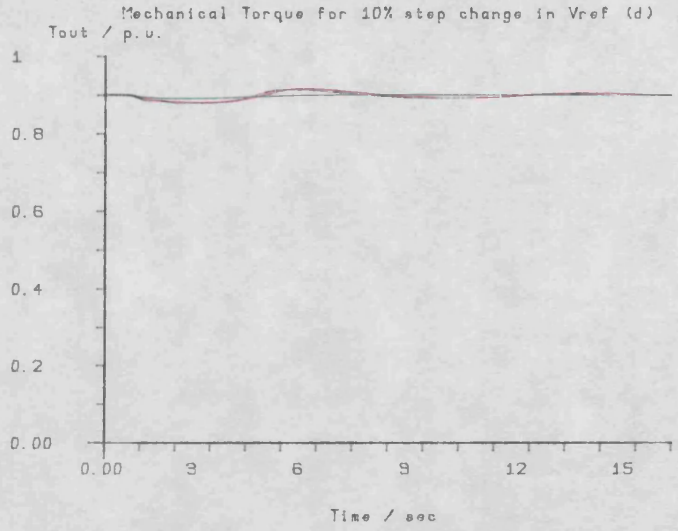
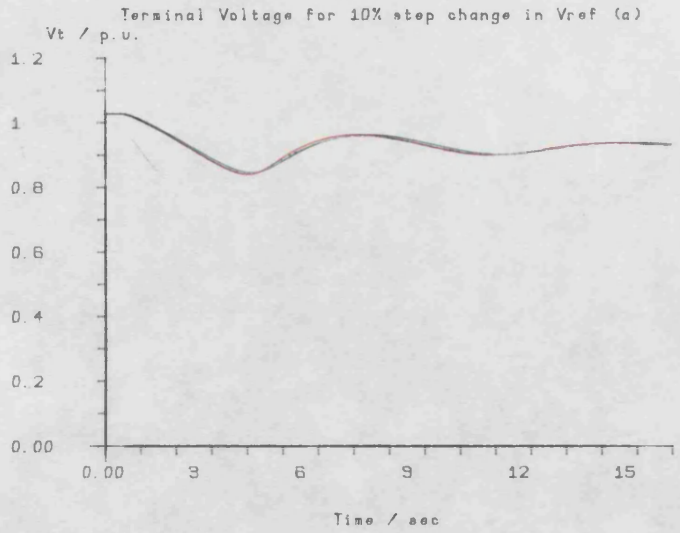
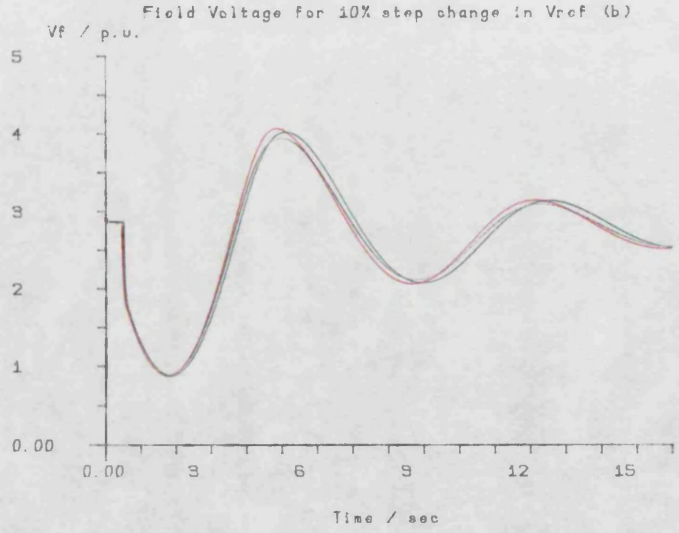
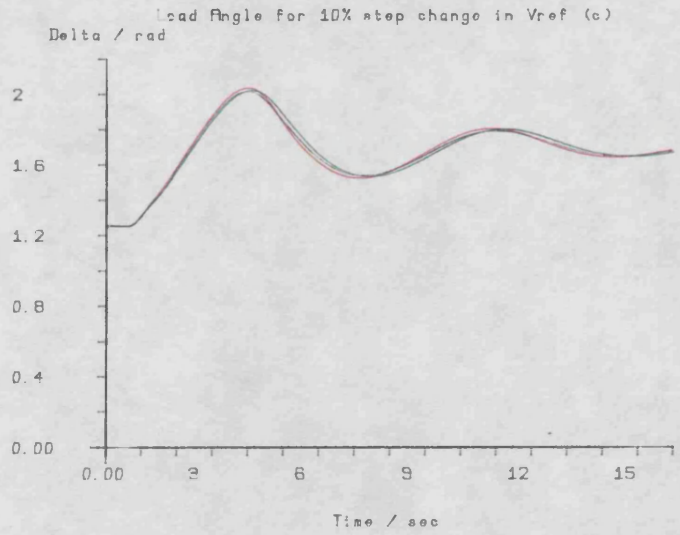

Mechanical Torque for 10% step change in Vref (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec

Fig 6.24


Terminal Voltage for 10% step change in Vref (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec


Field Voltage for 10% step change in Vref (b)
Vf / p.u.
5
4
3
2
1
0.00
0.00
3
6
9
12
15
Time / sec


Load Angle for 10% step change in Vref (c)
Delta / rad
2
1.6
1.2
0.8
0.4
0.00
0.00
3
6
9
12
15
Time / sec

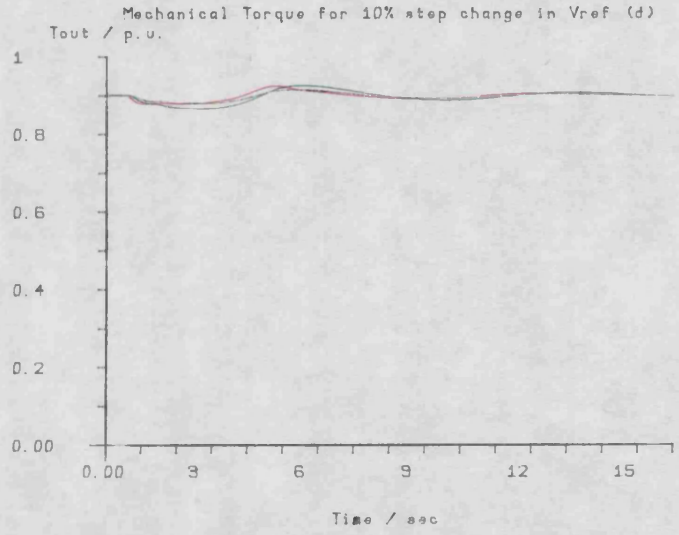

Mechanical Torque for 10% step change in Vref (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec

Fig 6.25

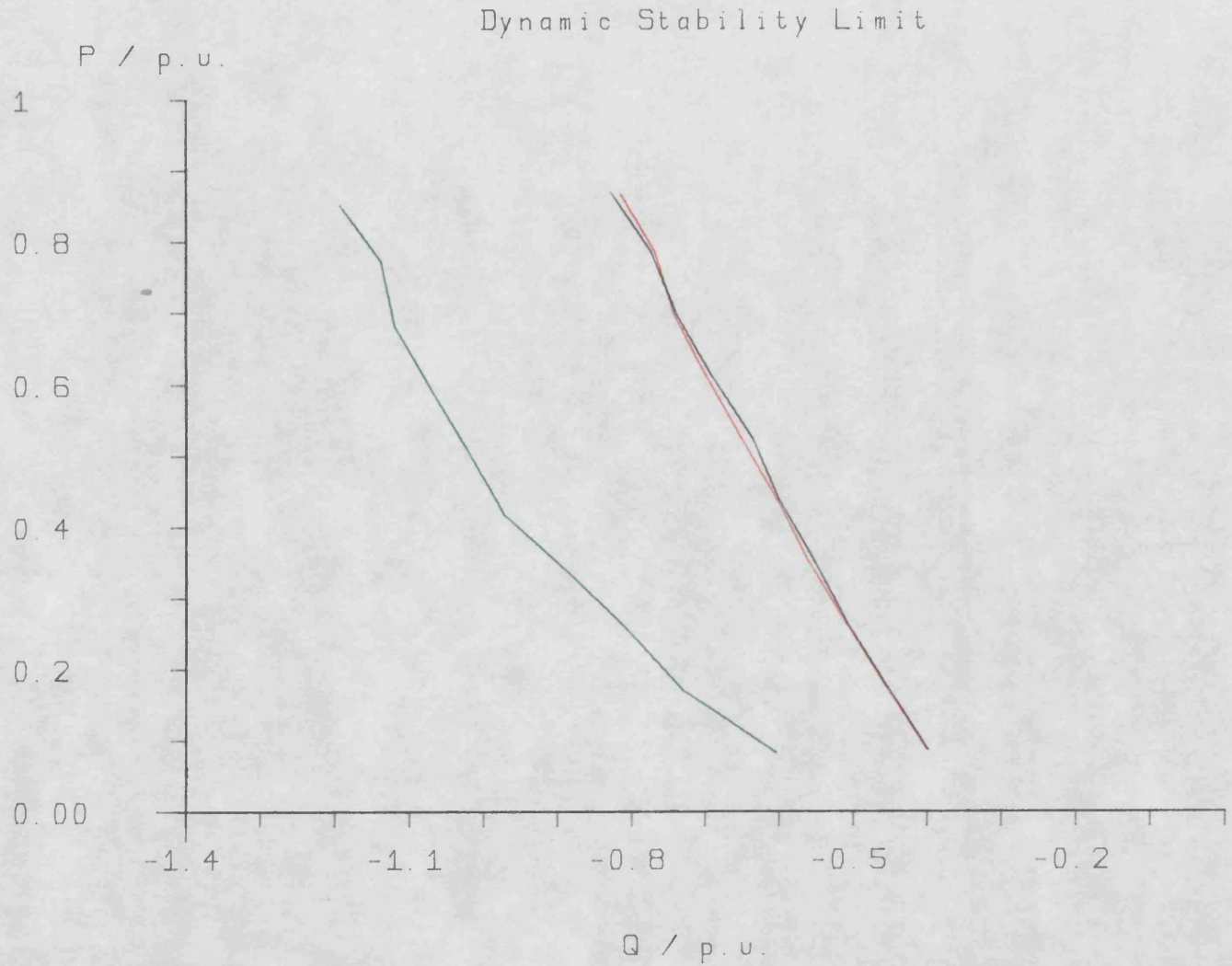


Fig 6.26

feedback signal into both the excitation and governing systems significantly improves the transient stability of the generating plant, such feedback fails to improve the dynamic stability of the plant. Use of transient electrical power as an extra feedback makes significant improvements in both the dynamic and the transient stability of this system.

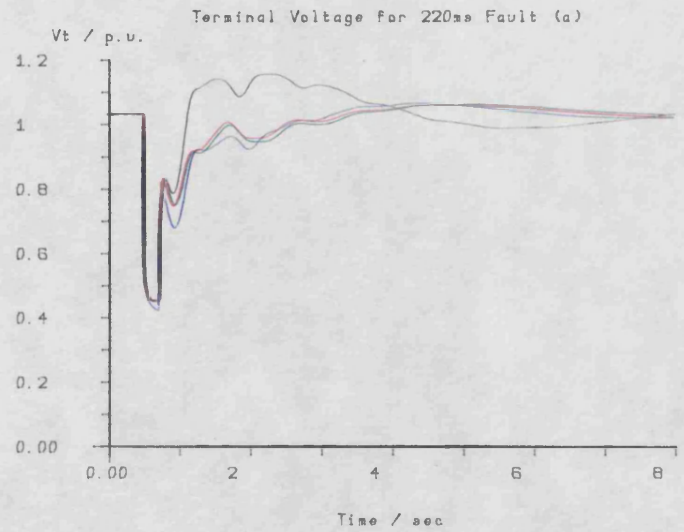
6.3.3 Optimisation of a Hydrogenerator, Infinite Busbar System

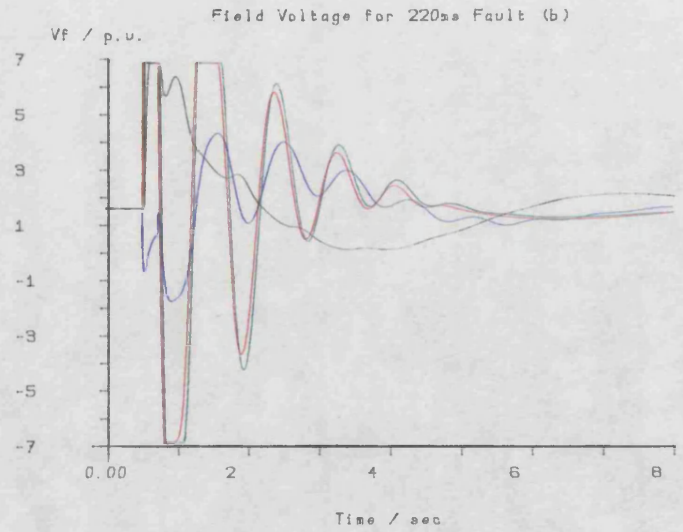
The optimisation studies presented in this section have been conducted using the plant data for the Dinorwig pump storage scheme given in Appendix A. The machine and generator transformer parameters have been supplied by the CEGB. The transmission line reactance is based on a CEGB supplied value of 8800 MVA for the summer minimum fault level at Dinorwig in 1983. This corresponds to a period when the generating network is most lightly loaded and most likely to suffer from stability problems.

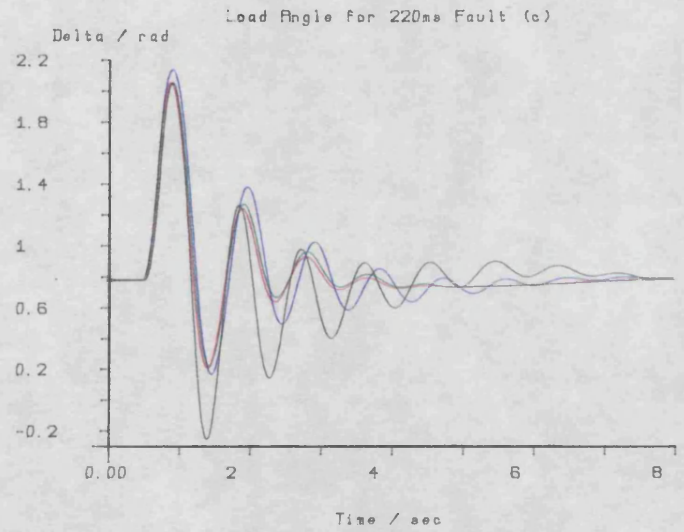
It has been assumed that the excitation system employed at Dinorwig is identical to that used in the previous section for the optimisation studies on the turboalternator. It has also been assumed that the dynamics of the water turbine are dominated by the opening and closing rate of the valve gear, which take 12 - 15 seconds to change from fully open to fully closed, or vice versa. Thus, identical models to those used in the previous section have been used in this section for the machine, the excitation system and the governing system, with appropriate changes in parameter values. The hydrogenerator system is therefore also represented by the block diagram of Fig. 6.8.

Inspection of the machine parameters in Appendix A shows that the per unit impedances of the hydrogenerator are about half those of the turbogenerator and also that the hydrogenerator has a far higher saliency. As a consequence of the lower machine impedance and the higher saliency, the hydrogenerator can be expected to deliver an equivalent per unit power output to that of the turbogenerator at a smaller steady state load angle and at a lower field voltage. The performance index used for the optimisations that follow is identical to that used in Section 6.3.2, equation 6.13, with the integration performed over 8 seconds and the weighting coefficients A1 and A2, set at 0.1 and 70, respectively. The base fault clearance time used for this study is 220ms. Optimisations have also been performed for a fault clearance time of 140ms, in addition to which both high and low ceiling excitation systems have been used.

Initially, optimisations were performed with each of the three extra feedback signals, rotor acceleration ($P^2\delta$), transient electrical power (ΔPe) and transient direct axis current (Δid), available to both the excitation and governing systems. Table 6.9 gives the initial conditions and resulting gains for this study and Fig. 6.27 shows the corresponding transient responses. These transient responses are colour coded as before, black for control E1, the base case; red for control E2, use of acceleration feedback; green for control E3, use of transient electrical power feedback; and blue for control E4, use of transient direct axis current feedback.


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

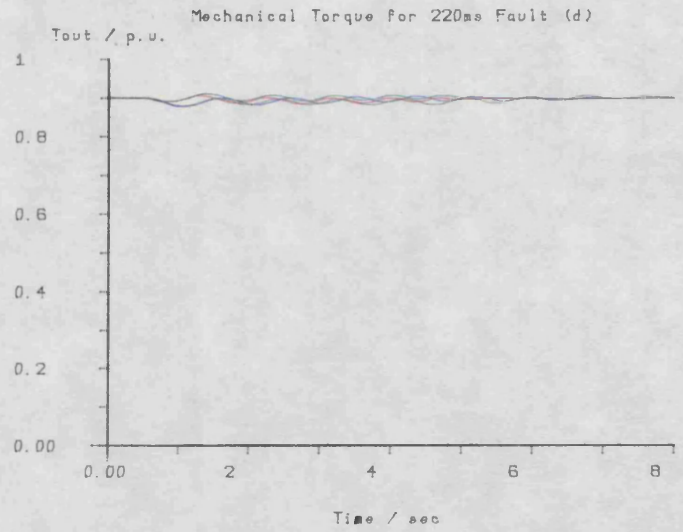

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.27

Initial Conditions

δ = 0.780 rad Vt = 1.03 p.u. P = 0.897 p.u. Q = -0.0738 p.u.

Vf = 1.60 p.u. Power factor = 0.997 leading

Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vfmax	Vfmin
E1	None	0	0	95.3	220ms	6.87	-6.87
E2	$P^2\delta$	0.0359	0.0480	50.0	220ms	6.87	-6.87
E3	ΔPe	-1.41	-0.621	51.0	220ms	6.87	-6.87
E4	Δid	-0.300	-0.00269	75.3	220ms	6.87	-6.87

Table 6.9

It can be seen from Fig. 6.27 that controls E2 and E3 result in almost identical transient responses. Control E4 results in a larger first peak in the load angle swing than the base case, control E1, while controls E2 and E3 fail to produce any observable change in the height of this peak. This is to be expected, since the rate limits applied to the governing system only allow very slow changes in the prime mover output torque and, as a result, the torque responses show virtually no change in output torque throughout the transient. The major improvement in the load angle response is the reduction in size of the first negative going peak after the fault has been cleared. Subsequent to this, controls E2 and E3 damp out much of the oscillation that is present in the load angle response for the base case, control E1. The field voltage excursions for controls E2 and E3 take on the bang-bang type characteristic that was noted in the previous section, whereas the field voltage response for control E4 fails to reach saturation in either direction. As a result, the terminal voltage response for control E4 reaches a lower faulted

level than the other three terminal voltage responses. All the terminal voltage responses, E2, E3 and E4, improve upon the base case terminal voltage response, in black.

A comparison of Tables 6.9 and 6.2 shows that the optimum gains feeding the extra control signals into the excitation system of the hydrogenerator are of a similar order of magnitude to those supplying the turbogenerator. The performance index values for controls E2 and E3 are very similar, and the similarity between the transient responses obtained using these controls has already been mentioned. If it is assumed that there are no speed dependent losses and that the mechanical torque from the prime mover remains constant, both approximately true, then from the swing equation (3.13) it follows that:

$$Mp^2\delta = -\Delta Pe \qquad (6.15)$$

It is further assumed that the extra signals input to the excitation system are identical, which is also not unreasonable on account of similarity of the transient responses, then:

$$Kacc.p^2\delta \simeq Kpe.\Delta Pe \qquad (6.16)$$

can be written, where Kacc is the optimum acceleration gain and Kpe is the optimum power gain, supplying the excitation system. Subject to these assumptions, the inertia constant M may be approximated from the ratio of the gains in equation 6.16. The estimate for M in this case is 0.0254, which compares very well with the value of 0.0286 used in the real time model.

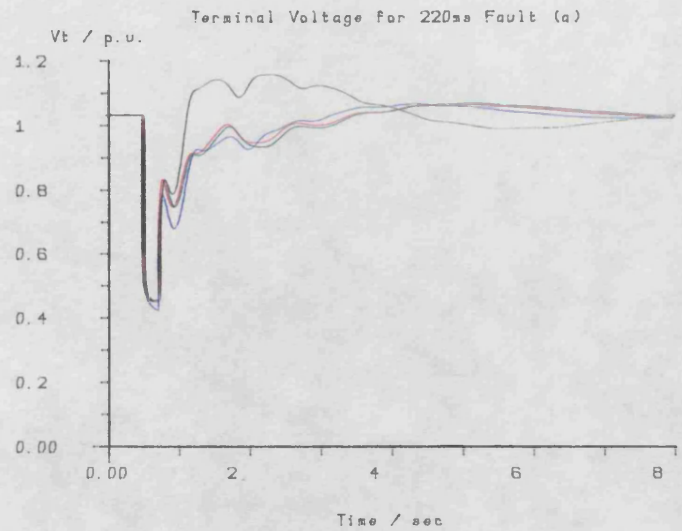
Since, in this generating system, the prime mover output remains effectively constant during the transient period, a further set of optimisations has been made for identical initial conditions, with no extra feedback into the governing system. Table 6.10 presents the optimum gains for this study and Fig. 6.28 shows the corresponding transient responses.

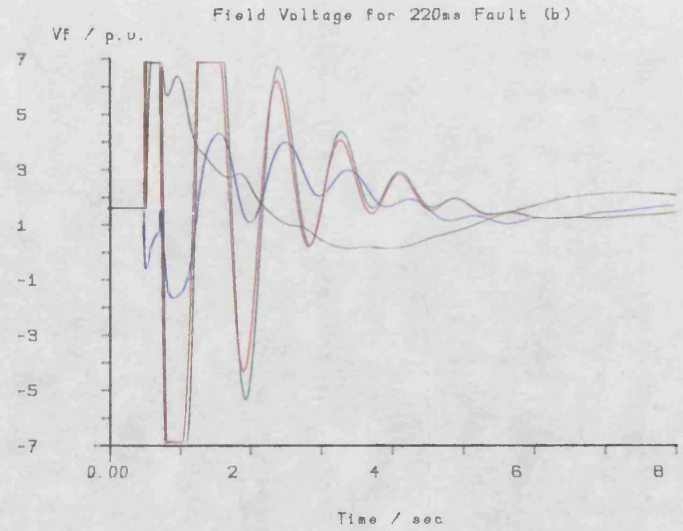
Initial Conditions as Table 6.9.

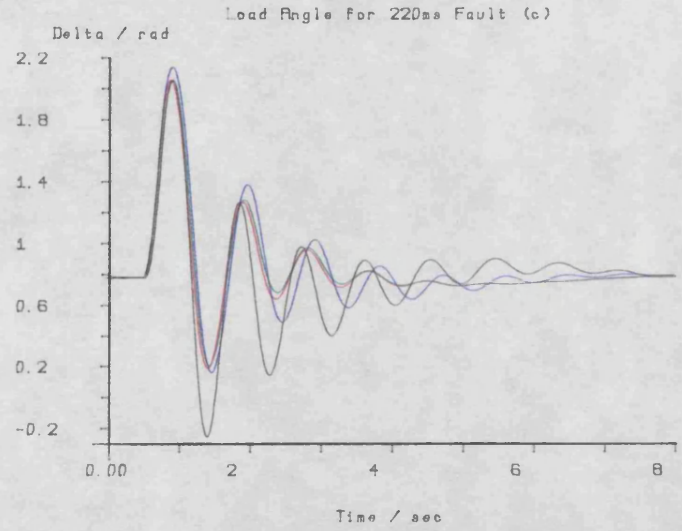
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf max	Vfmin
E5	$p^2\delta$	0.0381	0	52.2	220ms	6.87	-6.87
E6	ΔPe	-1.60	0	52.7	220ms	6.87	-6.87
E7	Δid	-0.297	0	75.3	220ms	6.87	-6.87

Table 6.10

The base case transient response for Fig. 6.28 was obtained by use of control E1 and so is identical to the base case plotted in Fig. 6.27. In all cases, the change in gain into the excitation system from Table 6.9 to Table 6.10 is less than 15%. The performance index values for controls E5 and E6 are only slightly worse than those for controls E2 and E3, and the corresponding transient responses of Figs. 6.27 and 6.28 are effectively identical. This confirms that little or no contribution is made to improving the transient response by making the extra feedback signals available to the governor loop of this system. The ratio of the gains for controls E5 and E6, 0.0238, also compares favourably with the value of M used.


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

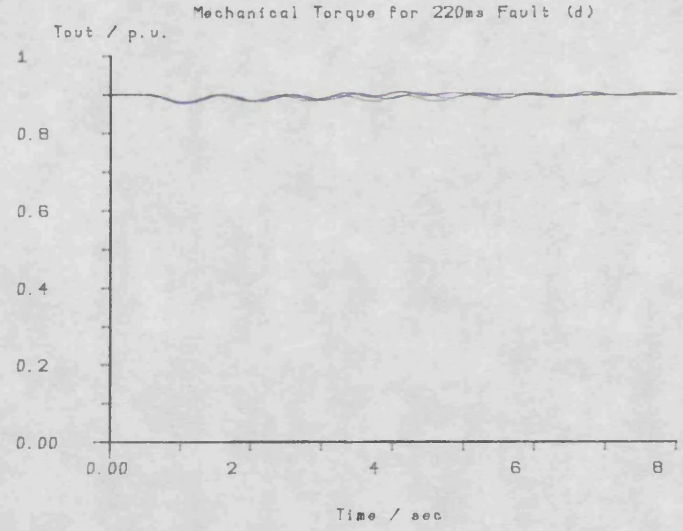

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.28

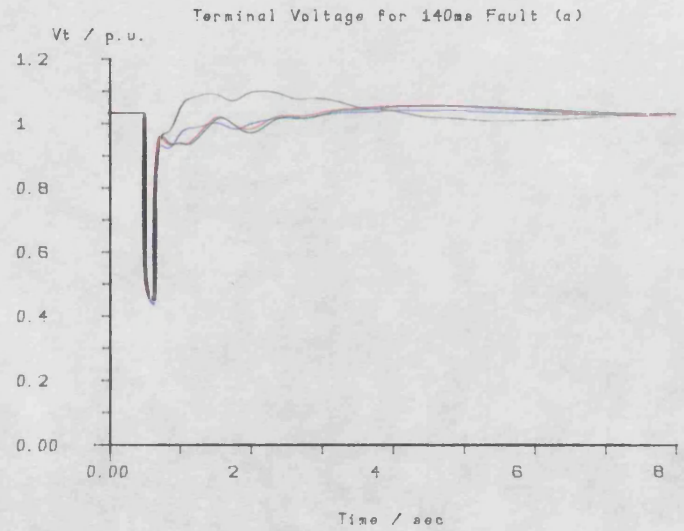
Initial Conditions as Table 6.9

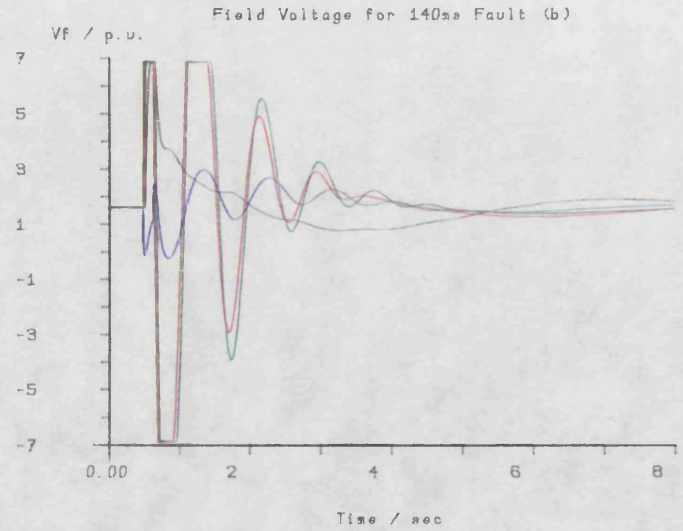
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
E8	None	0	0	26.7	140ms	6.87	-6.87
E9	$P^2\delta$	0.0641	0.0264	10.4	140ms	6.87	-6.87
E10	ΔPe	-2.53	-0.0145	10.7	140ms	6.87	-6.87
E11	Δid	-0.270	-0.769	17.4	140ms	6.87	-6.87
E12	$P^2\delta$	0.0463	0	11.2	140ms	6.87	-6.87
E13	ΔPe	-2.53	0	10.7	140ms	6.87	-6.87
E14	Δid	-0.333	0	18.9	140ms	6.87	-6.87

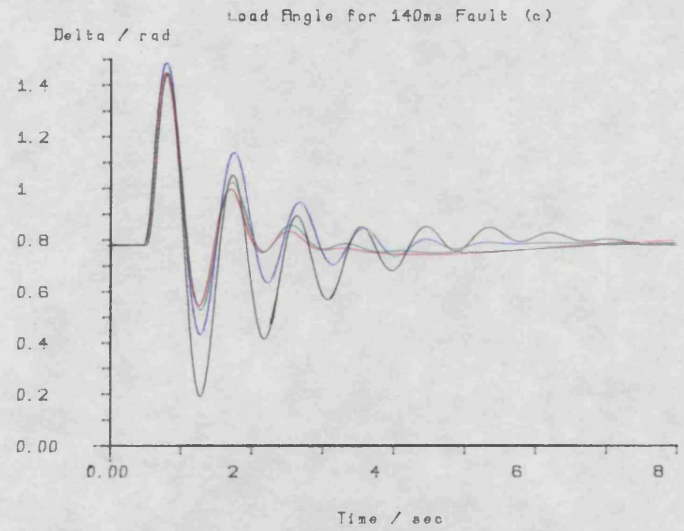
Table 6.11

Table 6.11 contains the results of optimisations performed for a fault clearance time of 140ms, and the transient responses are given in Fig. 6.29 (E8, E9, E10 and E11) and Fig. 6.30 (E8, E12, E13 and E14).

Controls E11 and E14 again show that use of transient direct axis current feedback causes an increase rather than a decrease in the first peak of the load angle excursion. The load angle response for these two controls is only slightly better damped than that of the base case, control E8. Neither of these controls, E11 or E14, succeeds in saturating the field voltage and, once again, the faulted terminal voltage reaches a lower level than that obtained either by use of the other two extra feedback signals, or in the base case. The major improvement in load angle response is again the reduction in


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

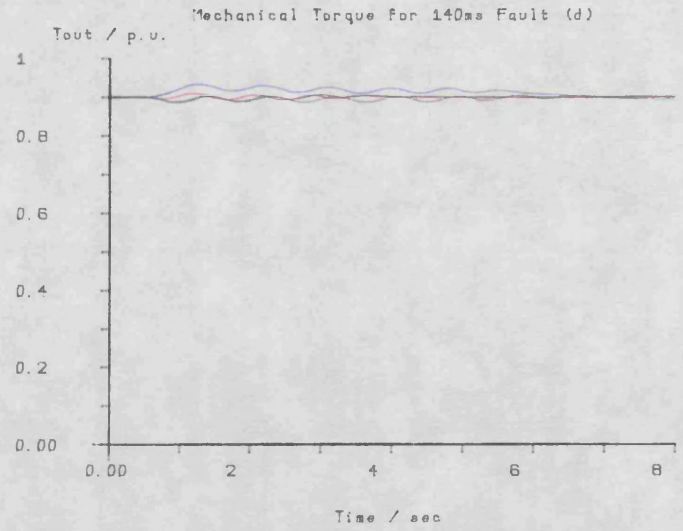
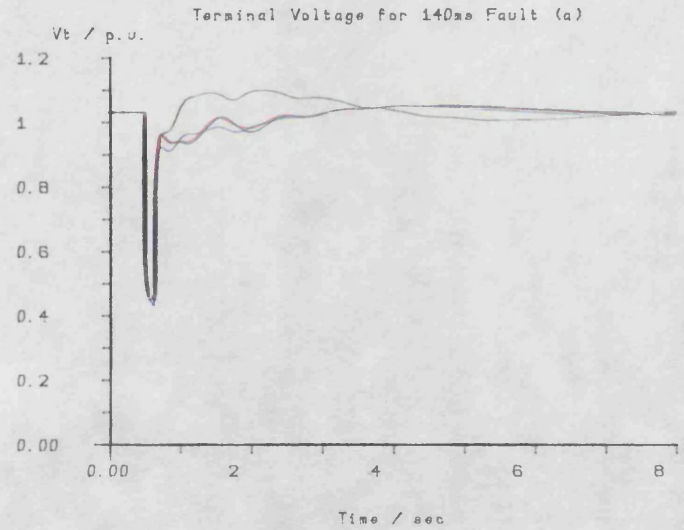
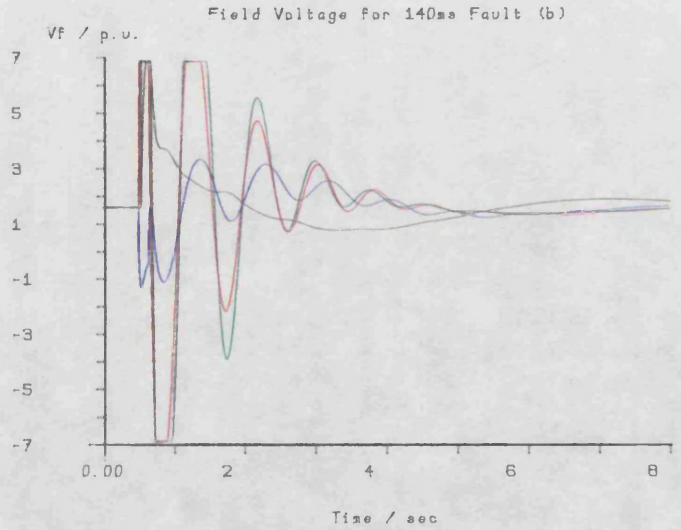
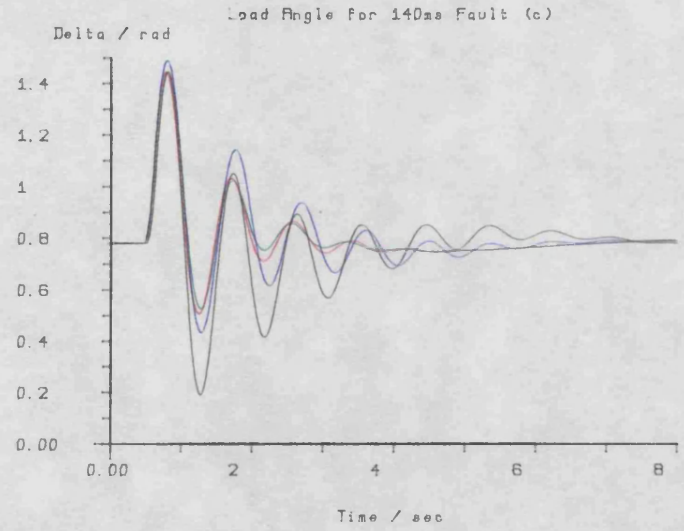

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.29


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

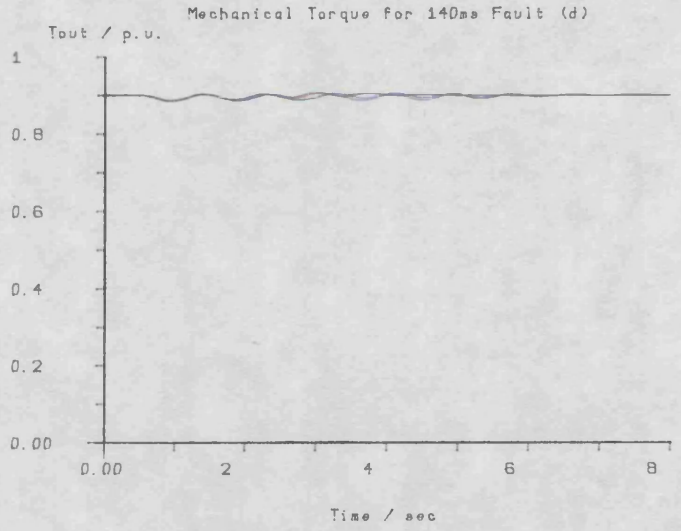

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.30

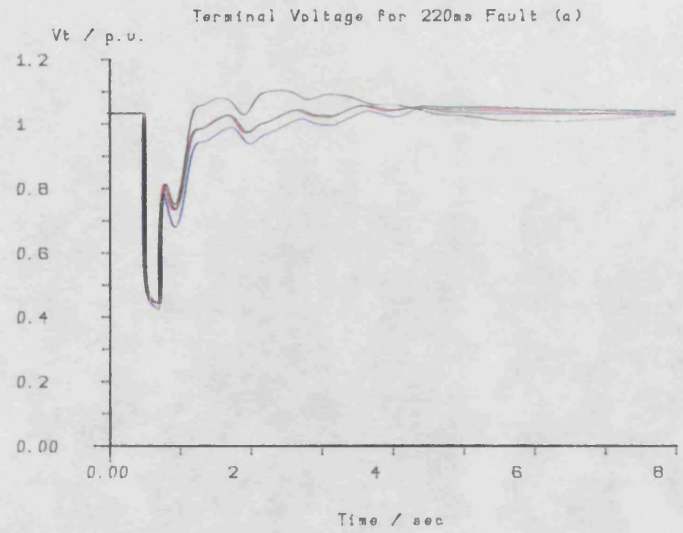
size of the first negative going peak obtained by use of rotor acceleration feedback, controls E9 and E12, or by use of transient electrical power feedback, controls E10 and E13. Damping of the subsequent load angle oscillations is also much improved by use of these controls. The use of each of the three additional feedback signals, applied individually, makes a considerable improvement in the terminal voltage transient response by reducing the peak overshoot. As noted in the previous study, controls E1-E7, the transient responses obtained with the use of rotor acceleration or transient electrical power feedback, E9 and E12, or E10 and E13, are very similar. Again, this can be accounted for by the constant nature of the torque response and the bang-bang nature of the field voltage response. The transient responses for the system, when re-optimised without the use of additional feedback into the governor loop, controls E12-E14, Fig. 6.30, are also very similar to those optimised with the additional signals fed into the governor loop, controls E9-E11, Fig. 6.29. The change in the gains feeding the excitation system between controls E9-E11 and controls E12-E14, is also small, particularly for the instances of transient electrical power and transient direct axis current feedback. The ratio of the optimum acceleration gains to the optimum power gains once more result in values in the order of the inertia constant M. Although the ratio using the gains from controls E12 and E13, 0.0183, is disappointingly small, the difference in transient responses is insignificant.

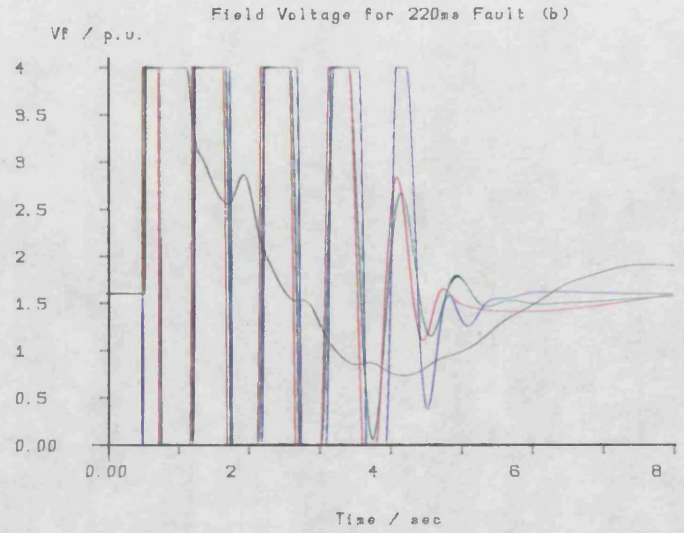
Initial Conditions as Table 6.9

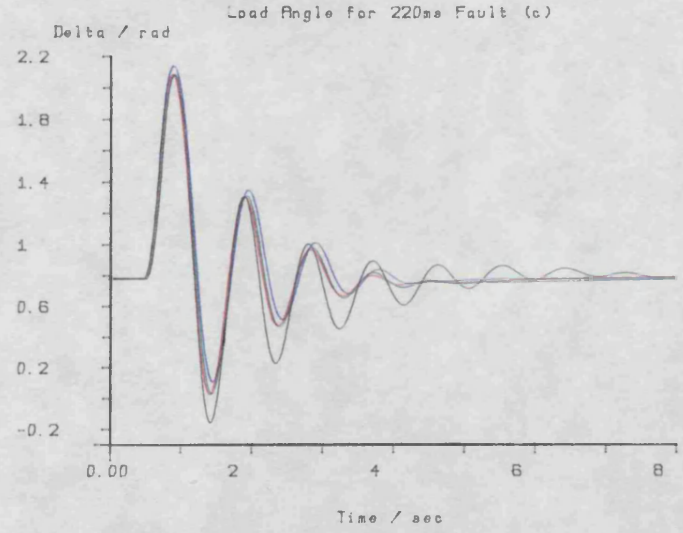
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
E15	None	0	0	88.7	220ms	4	0
E16	$P^2\delta$	0.0949	0.0557	63.4	220ms	4	0
E17	ΔPe	-2.00	-0.853	64.7	220ms	4	0
E18	Δid	-6.88	-0.00281	70.5	220ms	4	0
E19	$P^2\delta$	0.0563	0	67.7	220ms	4	0
E20	ΔPe	-2.63	0	68.1	220ms	4	0
E21	Δid	-18.7	0	70.4	220ms	4	0

Table 6.12

Table 6.12 presents the results of optimisations performed for the standard fault clearance time of 220ms, with a lower positive excitation ceiling, and no facility to reverse the field voltage. Transient responses for these controls are plotted in Fig. 6.31 (E15, E16, E17 and E18) and Fig. 6.32 (E15, E19, E20 and E21). The most notable feature of these responses, as compared with those in Figs. 6.27-6.30, is the field voltage response obtained by the use of transient direct axis current feedback, controls E18 and E21, plotted in blue. Figs. 6.31 and 6.32 show this response taking on the bang-bang nature that has so far been associated with the use of rotor acceleration or transient electrical power feedback. Subsequent to the initial excursions, the load angle responses for controls E18 and E21 are as well damped as those for controls E16, E17, E19 and E20. This damping must be attributed to the nature of the field voltage


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

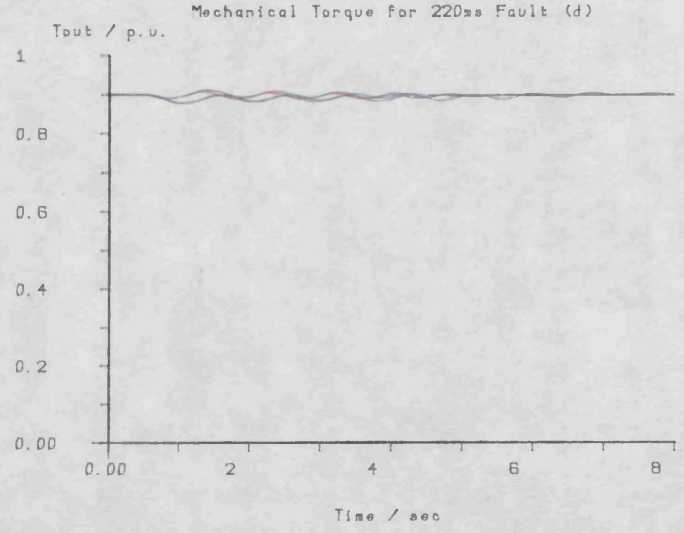
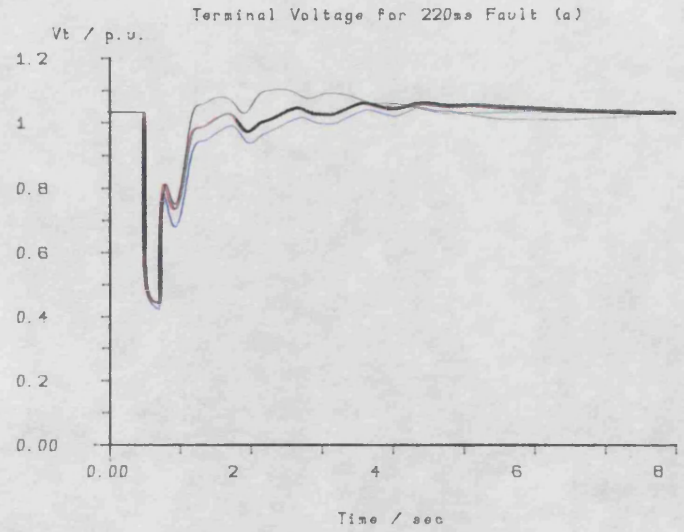
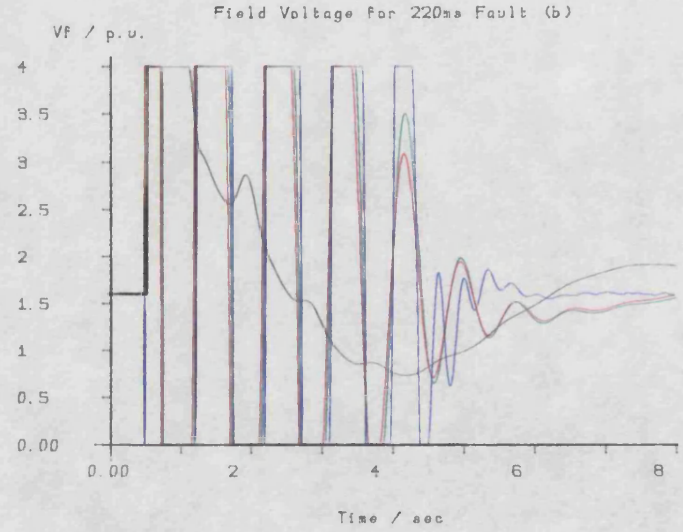
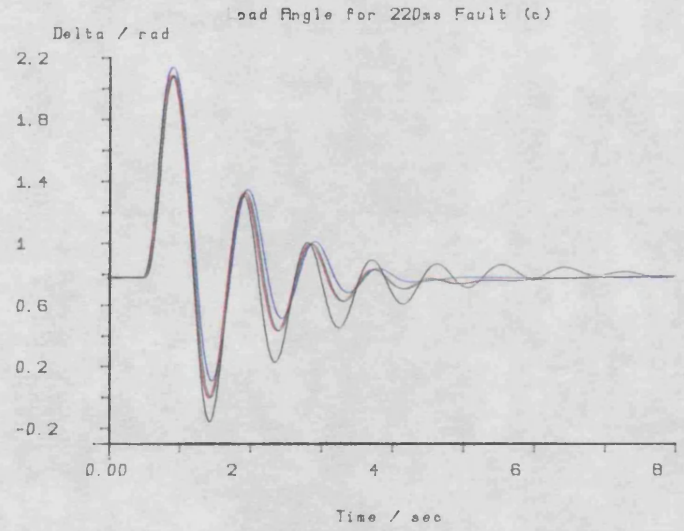

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.31


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


oad Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

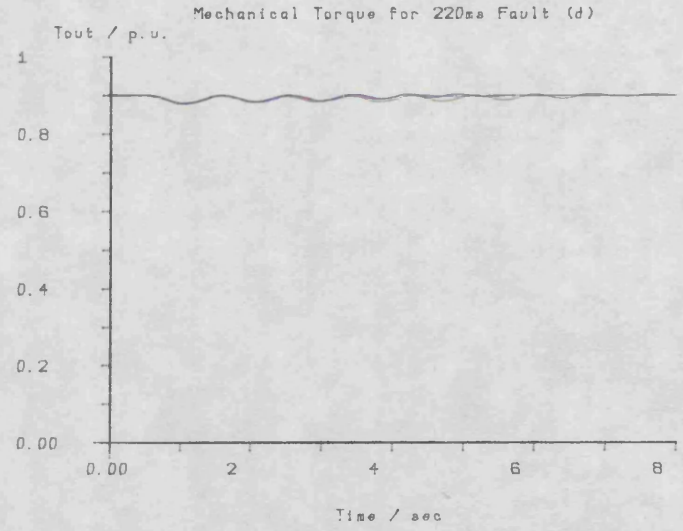

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.32

reponse. There is, however, one important difference in the field voltage response using controls E18 and E21, and that occurs at fault inception. Figs. 6.31 and 6.32 show the field voltage rising in response to the terminal voltage error at fault inception in all cases, except the two controls, E18 and E21, in which the field voltage is driven to zero. In the base case, control E15, the field voltage remains in positive saturation until the terminal voltage approaches the value set by the excitation system reference. In the case of feedback of rotor acceleration, controls E16 and E19, in red, or transient electrical power, controls E17 and E20 in green, the field voltage is brought out of positive saturation at fault clearance, either by the deceleration of the rotor, or by the increase in electrical power output. Once all the field voltage responses have reached zero, the field voltage responses for controls E16, E17 and E18, and for controls E19, E20 and E21 make transitions between the positive excitation ceiling and zero at approximately the same instants in time and in the same direction. This continues until, at about four seconds into the transient, the load angle oscillations have been damped out.

The difference in the early part of the field voltage response for controls E18 and E21 must be responsible for the much larger first peak in the load angle response. Since the field voltage is not at its positive ceiling during the fault, the terminal voltage, as noted before, falls to a lower level. This will result in less electrical power being transferred to what is left of the network during the fault, namely, the generator transformer. Since the input power from the prime mover is virtually unchanged, due to the slow valve rates maintaining an almost constant torque throughout the transient, more

excess power is available to accelerate the rotor than that available when the field voltage is in positive saturation. This accounts for the increase in the first peak in all the load angle responses of Figs.6.27 to 6.32 where transient direct axis current is used as an additional control signal.

It is clear that the high gain values through which the change in direct axis current is supplied to the excitation system, controls E18 and E21, is responsible for the resulting field voltage saturation. The difference between the optimum gains using these two controls makes a very noticeable difference to the latter part of the corresponding field voltage transients in Figs.6.31 and 6.32. Some of this difference must be attributed to the presence of extra feedback into the governor loop, control E18, Fig.6.31, or its absence, control E21, Fig.6.32. It can be seen that the latter part of the field voltage is better damped for controls E16-E18 than for controls E19-E21, but this results in no significant difference in the terminal voltage and load angle responses.

The terminal voltage and load angle responses obtained by the use of rotor acceleration and transient electrical power feedback are identical. The terminal voltage rersponses for all controls, E16-E21 are better than the response for the base case, control E1, and the responses for controls E16, E17, E19 and E20 recover slightly quicker than those for controls E18 and E21.

A comparison of Tables 6.9 and 6.10 with Table 6.12 shows that, although the performance index for control E15 (the base case with the lower excitation ceilings) is lower than that for control E1 (the

base case with the higher excitation ceilings) the higher excitation ceiling permits a better optimised performance index to be obtained, except in the case of transient direct axis current feedback. The effect of the very high magnitude gains for controls E18 and E21, as compared with controls E4 and E7, on the transient reponse has already been discussed. The optimum acceleration gains for controls E16 and E19, which feed the extra control signal into the excitation system are somewhat larger than those for controls E2 and E5. However, they remain within the same order of magnitude. This can also be said of the optimum transient power gains for controls E17 and E20 when compared with those for controls E3 and E6. The ratio of the optimum gains for controls E16 to E17 and E19 to E20, yield values of the same order of magnitude as the inertia constant M, although neither of the values is particularly close to the actual value used in the real time model.

Table 6.13 presents the results of optimisations performed with the generator operating at a leading power factor. The corresponding transient responses are plotted in Fig.6.33 (E22, E23, E24 and E25) and Fig.6.34 (E22, E26, E27 and E28).

Initial Conditions

δ = 1.08 rad Vt = 0.854 p.u. P = 0.817 p.u. Q = −0.544 p.u.

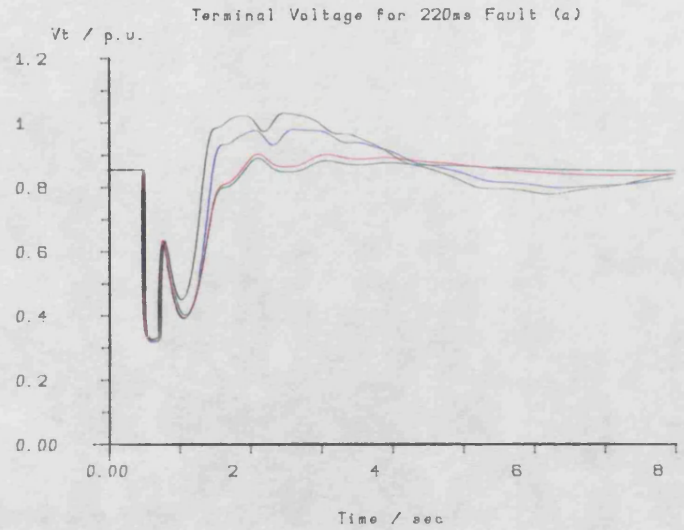
Vf = 1.19 p.u. Power factor = 0.832 (leading)

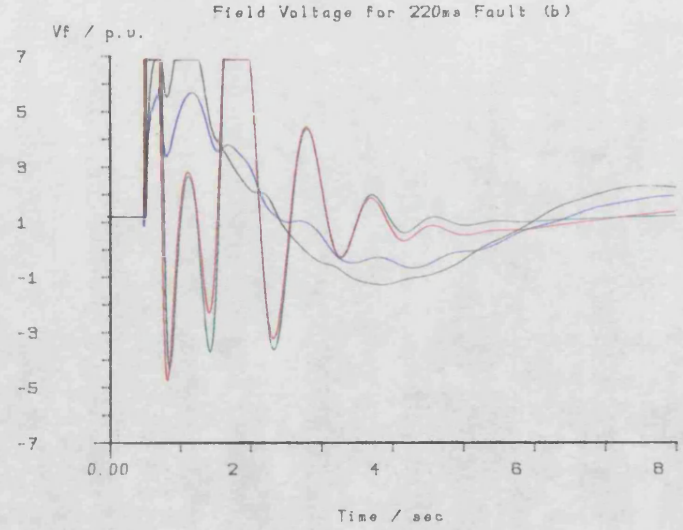
Control	Signal	Kavr	Kgov	P.I.	Fault Time	Vf_{max}	Vf_{min}
E22	None	0	0	229	220ms	6.87	−6.87
E23	$p^2\delta$	0.0448	0.00284	112	220ms	6.87	−6.87
E24	ΔPe	−1.72	−0.101	102	220ms	6.87	−6.87
E25	Δid	−0.101	−0.00137	218	220ms	6.87	−6.87
E26	$p^2\delta$	0.0453	0	113	220ms	6.87	−6.87
E27	ΔPe	−1.75	0	103	220ms	6.87	−6.87
E28	Δid	−0.104	0	217	220ms	6.87	−6.87

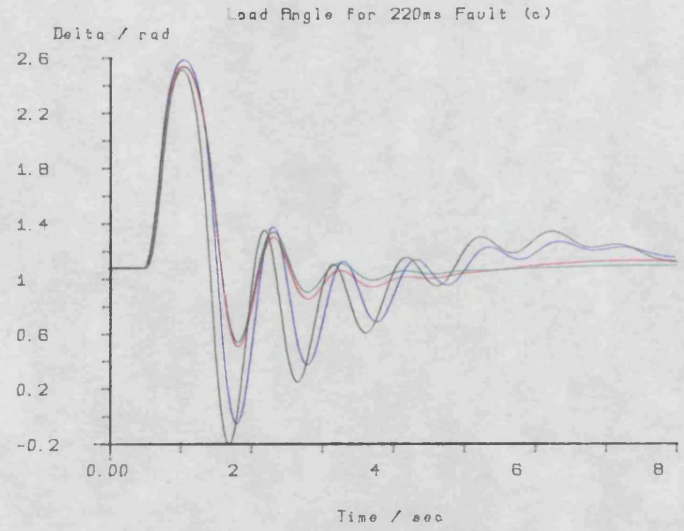
Table 6.13

It can be seen that the transient responses of Figs.6.33 and 6.34 are very similar. Although the performance index values differ slightly between corresponding controls, with and without additional feedback into the governing system, these two sets of transient responses may be considered identical.

The most noticeable feature of these responses is again the field voltage transient. It can be seen that none of these responses reaches negative saturation following fault clearance and all have a


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.6
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

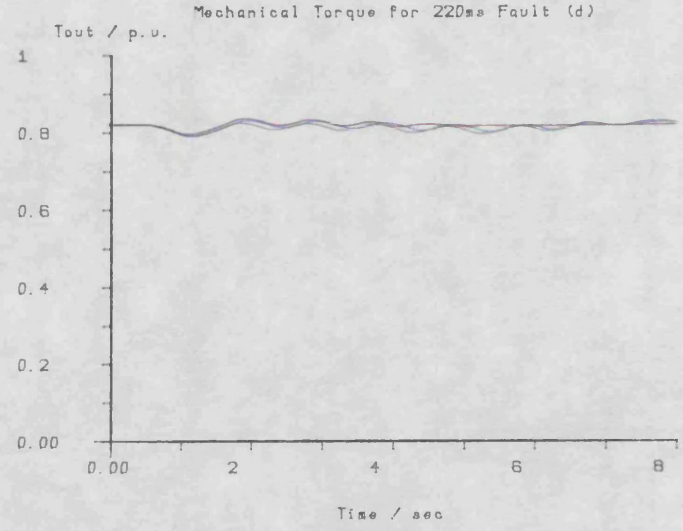
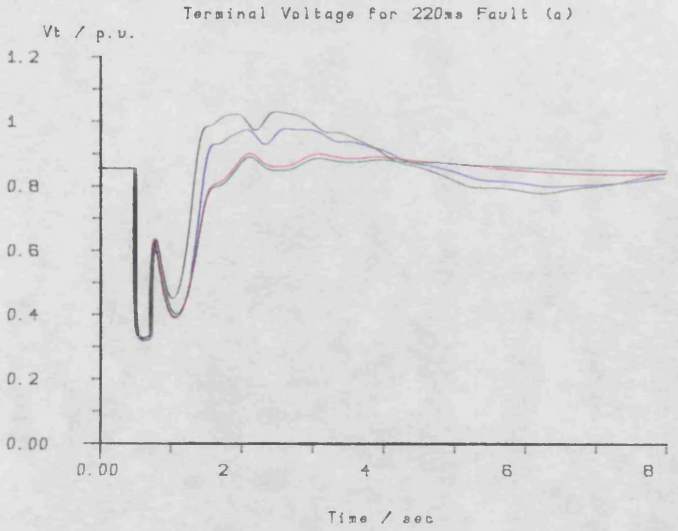
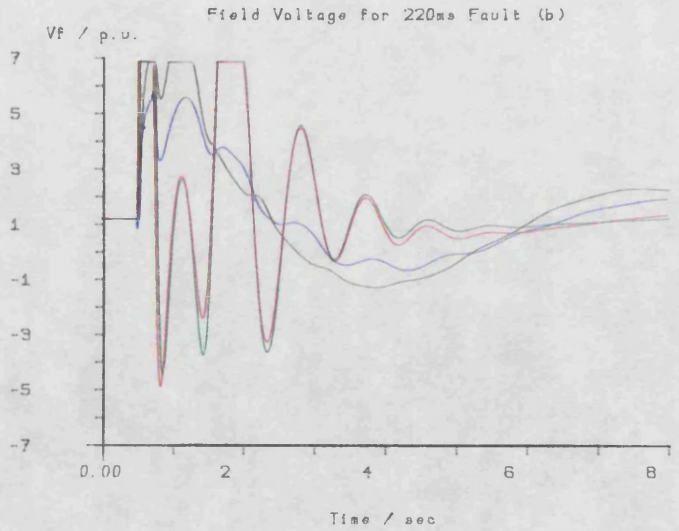
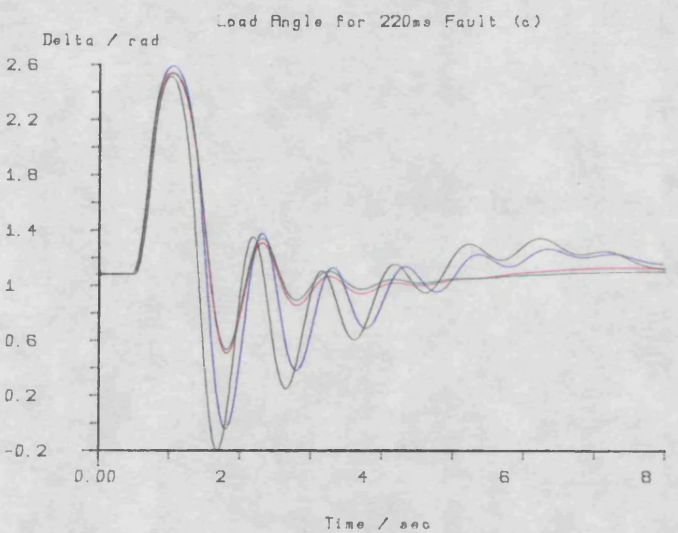

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.33


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.6
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

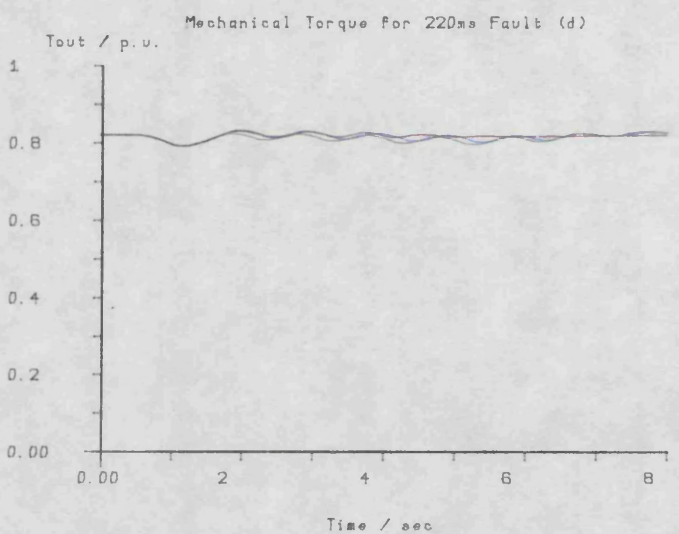

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.34

positive going peak which coincides with the comparatively large negative going peak in the terminal voltage response shortly after fault clearance. In the case of controls E23, E24, E26 and E27, it is this peak which prevents the excitation reaching its negative ceiling. In the case of controls E25 and E28, the field voltage increases towards the positive ceiling while the fault is present and, after fault clearance, the field voltage response is similar in shape to that of the base case, control E22. Both the terminal voltage and the load angle responses for controls E25 and E28 have a similar form to those for the base case, but slightly delayed in time and displaced with respect to the vertical axis. Due to the failure of the field voltage response to reach positive saturation for these two controls, the faulted terminal voltage reaches a lower value and there is an increase in the size of the first peak of the load angle response. No improvement in the damping of the subsequent response is observed.

Controls E23, E24, E26 and E27 result in a well-damped load angle response, with a significant reduction in the size of the first negative going peak, after fault clearance. There is also a small, but noticeable, increase in the size of the first positive going peak in the load angle response. Apart from the large negative peak shortly after fault clearance, the voltage recovery obtained by use of these controls is far better than that of the base case, or that obtained with controls E25 and E28.

The optimum gain values given in Table 6.13 are typical of those given in Tables 6.9 to 6.12, in that they have the same orders of magnitude and sign as those already given, and the ratio of the gains

for controls E23 to E24 and E26 to E27 give good estimates of the inertia constant M.

It can be concluded from the transient responses of Figs.6.27 to 6.34 that the use of transient direct axis current as an additional feedback signal fails to produce any useful improvement in the transient performance of this system. Any improvement in the terminal voltage response must be offset against the increased size of the first peak in the load angle response and the absence of a significant improvement in the subsequent damping of this response (except in the response of Figs.6.31 and 6.32). It can also be concluded that the use of transient electrical power or rotor acceleration as an extra control signal offer equivalent improvements in the transient performance of the plant. In all cases, their use greatly improves the terminal voltage recovery and provides damping of the load angle oscillations. These comments apply both to the cases where the extra control signal is made available to the governing system and to three cases where it is not. This implies that little or no benefit is obtained by supplying the extra feedback signal to the governing system.

Inspection of Tables 6.9 to 6.13 show that the optimum gains through which rotor acceleration is fed into the excitation system lie in the range 0.0359 to 0.0949, and those through which transient electrical power is fed lie in the range -1.41 to -2.63. The optimum gains are spread irregularly throughout these ranges, although they tend to be grouped towards the lower magnitude end of each range.

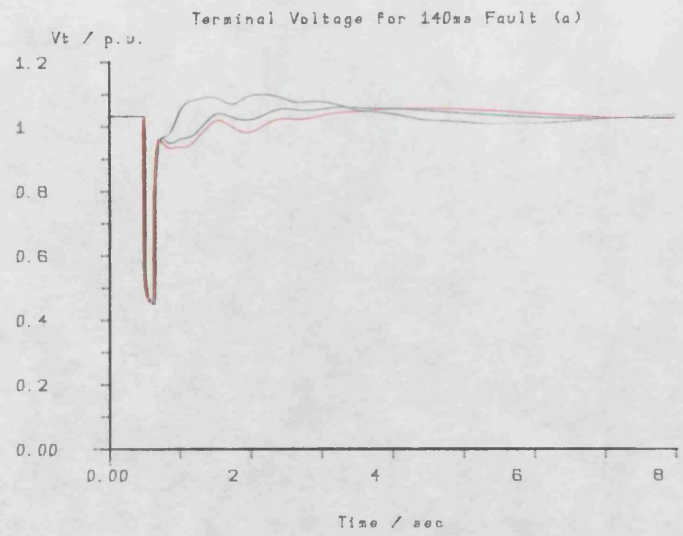
The transient responses plotted in Figs.6.35 to 6.44 each

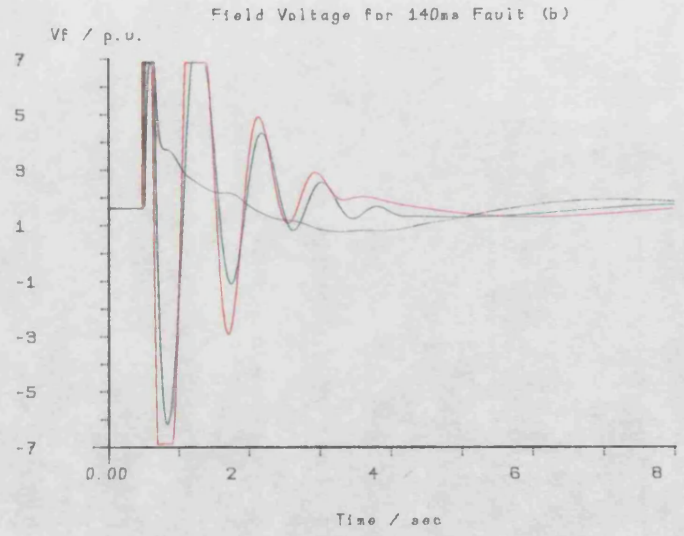
comprise of three responses; a base case, in black, where no additional feedback signals are used; an optimised control, in red; and a third case, in green, where the additional signals are fed into the control loops through gains optimised for a 220ms fault clearance time, using a high ceiling excitation system. Table 6.14 summarises these responses.

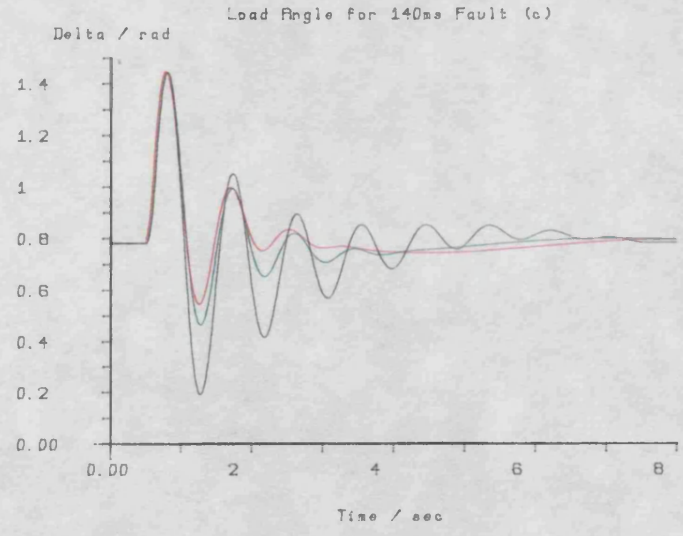
Cont.	Signal	Kavr	Kgov	P.I.	Fault Time	Vf max	Vf min	Base Cont.	Opt Cont.	Fig.
F1	$P^2\delta$	0.0359	0.0480	11.5	140ms	6.87	-6.87	E8	E9	6.35
F2	$P^2\delta$	0.0381	0	11.6	140ms	6.87	-6.87	E8	E12	6.36
F3	ΔPe	-1.41	-0.621	10.8	140ms	6.87	-6.87	E8	E10	6.37
F4	ΔPe	-1.60	0	11.4	140ms	6.87	-6.87	E8	E13	6.38
F5	$P^2\delta$	0.0359	0.0420	64.1	220ms	4	0	E15	E16	6.39
F6	$P^2\delta$	0.0381	0	64.0	220ms	4	0	E15	E19	6.40
F7	ΔPe	-1.41	-0.621	64.9	220ms	4	0	E15	E17	6.41
F8	ΔPe	-1.60	0	68.3	220ms	4	0	E15	E20	6.42
F9	$P^2\delta$	0.0381	0	117	220ms	6.87	-6.87	E22	E26	6.43
F10	ΔPe	-1.60	0	105	220ms	6.87	-6.87	E22	E27	6.44

Table 6.14

Figs.6.35 to 6.38 show the effect of using gains optimised for a fault clearance time of 220ms, when a reduced fault clearance time of 140ms is used. It can be seen that controls F1-F4 all offer an improvement in transient performance when compared with the base control, E8 in black. The improvement obtained is not as large as


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

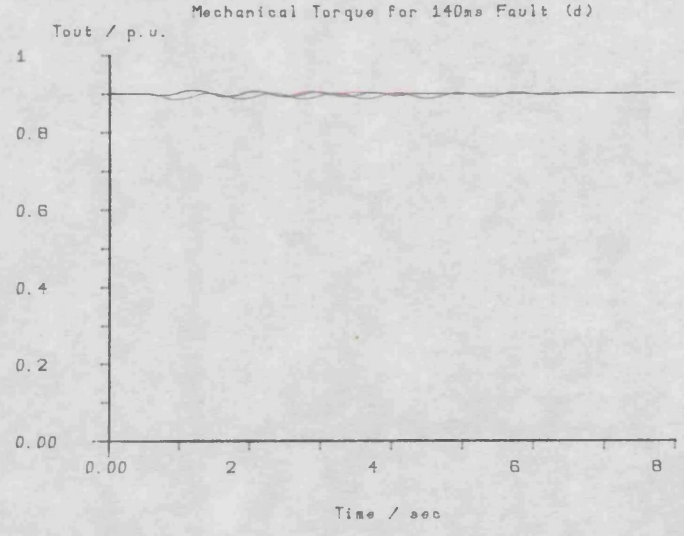
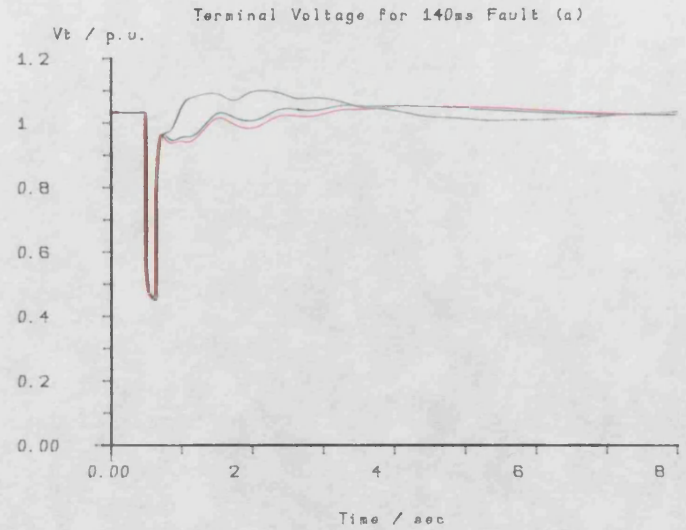
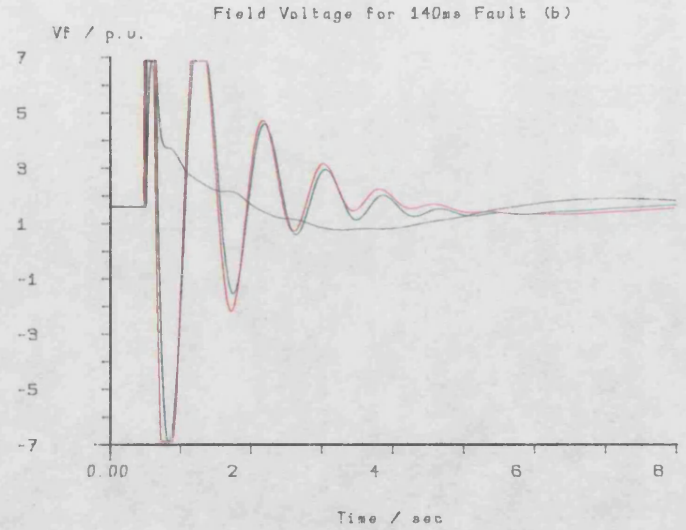
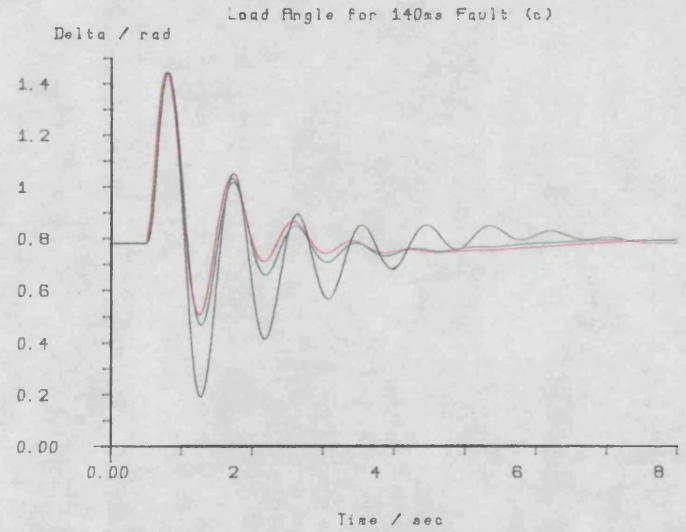

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.35


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

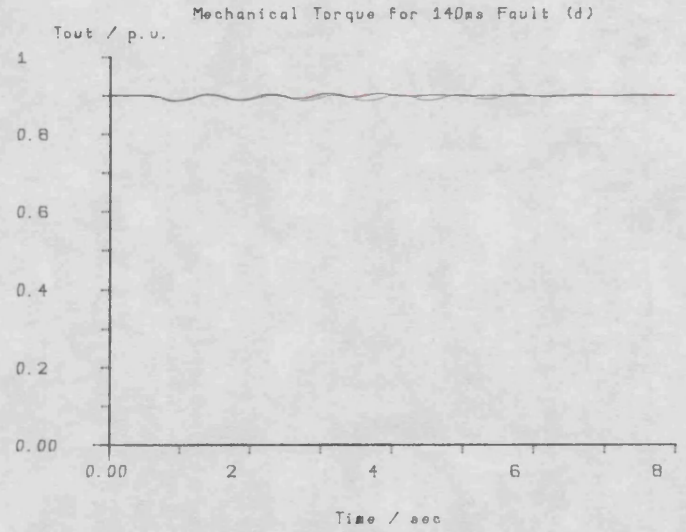
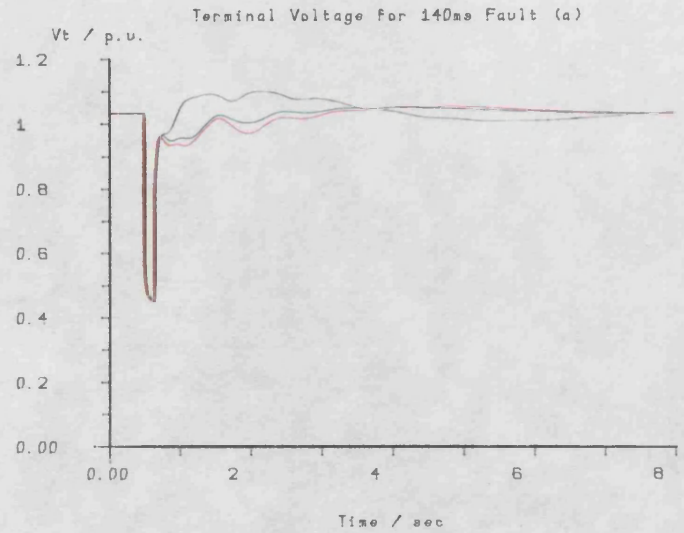
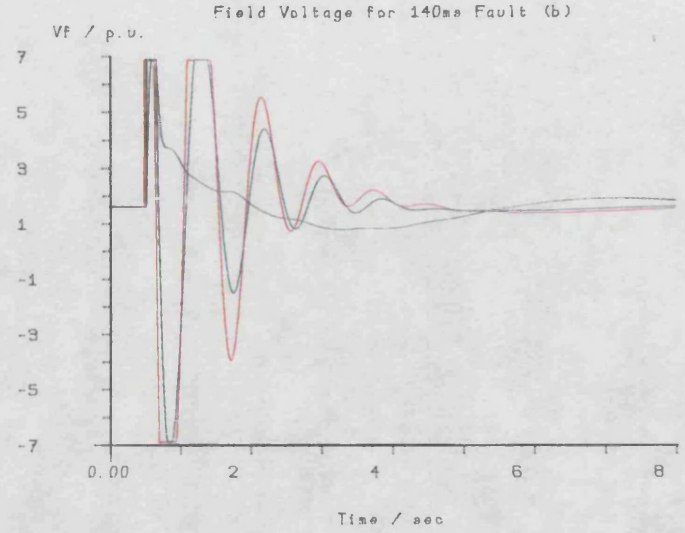
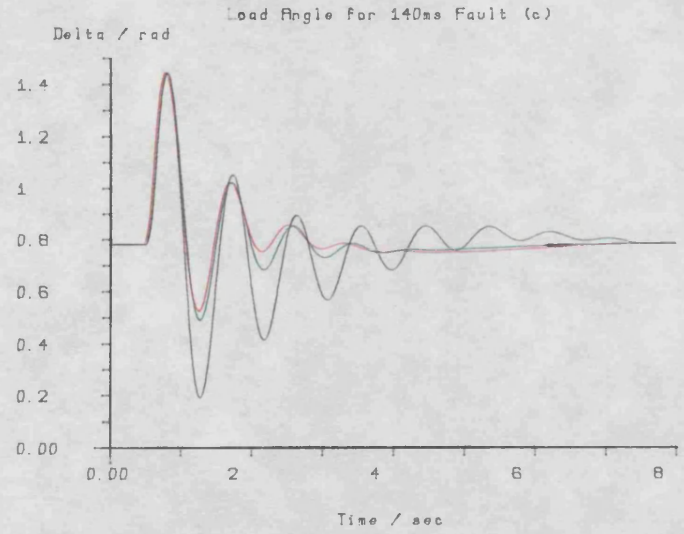

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.36


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

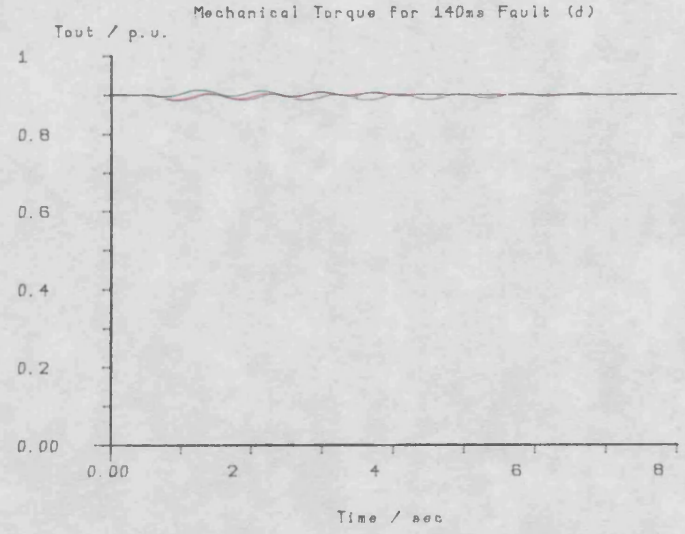
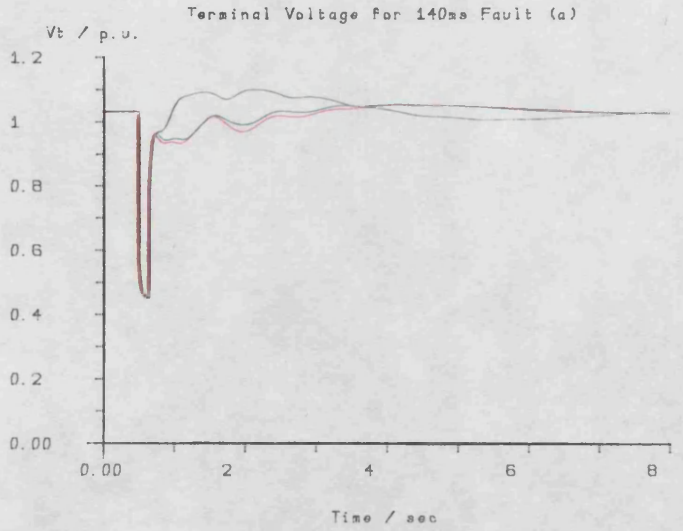
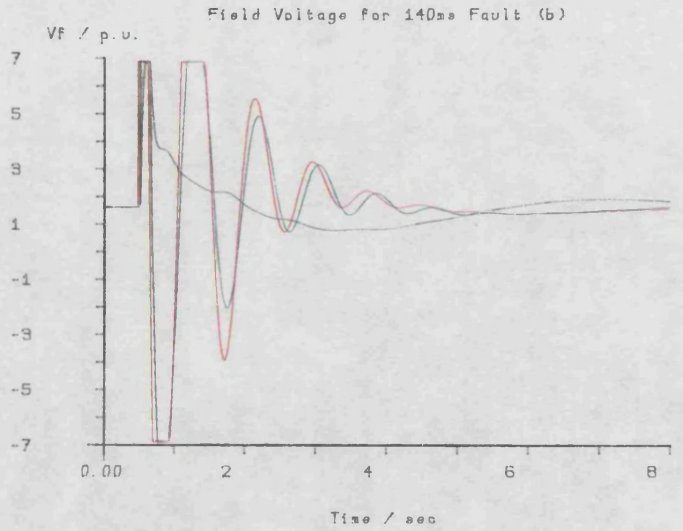
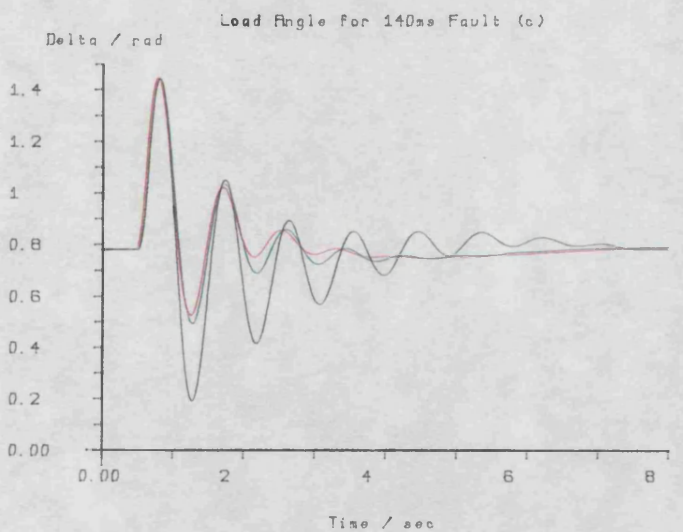

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.37


Terminal Voltage for 140ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 140ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 140ms Fault (c)
Delta / rad
1.4
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

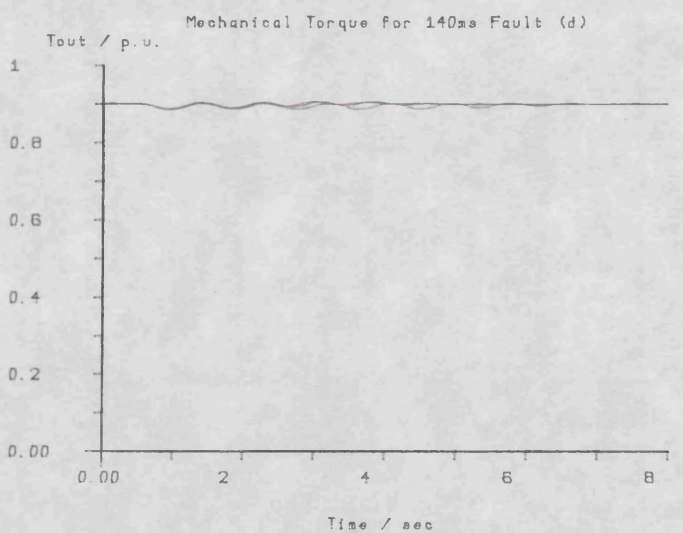

Mechanical Torque for 140ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.38

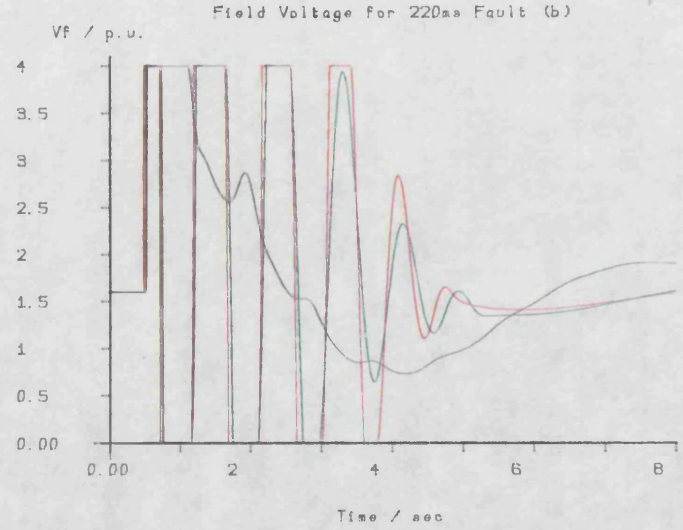
that obtained by the use of controls optimised for a 140ms fault clearance time, in red. However, in all cases, F1-F4, a well-damped load angle response, with a significant reduction in the size of the first negative going peak is achieved. The improved terminal voltage responses differ by a very small amount and it is not clear which is the better of the two responses, (red or green), in each of the figures, Figs.6.35 to 6.38.

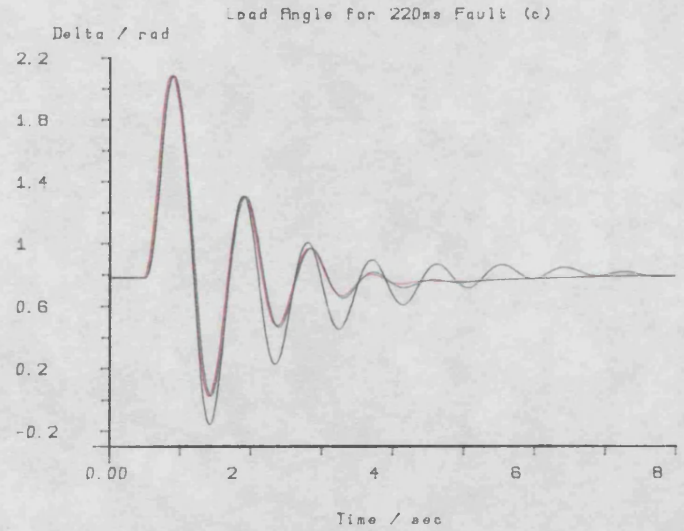
It is apparent from the field voltage responses that the transitions between the upper and the lower excitation ceilings are faster for the controls optimised with a 140ms fault clearance time. This indicates that a larger magnitude gain is required to feed the excitation system than is used in controls F1-F4, in order to achieve optimum performance. This may be verified by comparing the gains used by controls F1-F4 with those used by the corresponding optimal controls (E9, E12, E10 and E13) given in Table 6.11.

Figs. 6.39 to 6.42 show that use of controls F5-F8 produce an improvement in transient performance, comparable with that obtained by the use of controls optimised for a reduced ceiling excitation system. The magnitude of the gains supplying the excitation system is smaller for controls F5-F8 than those for the optimised controls, but it can be seen that they are sufficiently large to produce field voltage responses with the same bang-bang nature as the optimised controls. Consequently, the terminal voltage and load angle responses for controls F5-F8 are identical to the corresponding responses obtained with the optimal controls.

Similar comments may be made with regard to Figs.6.43 and 6.44,


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

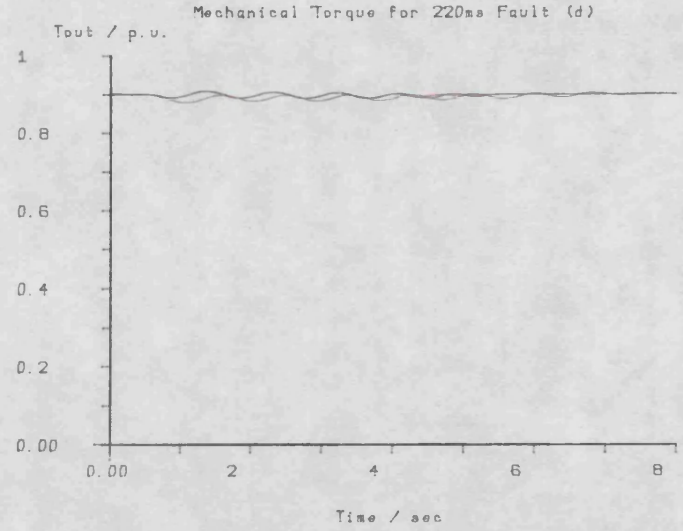
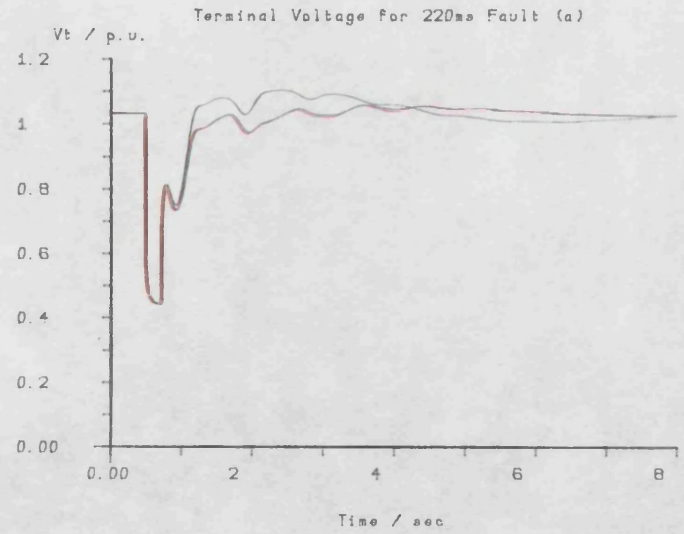
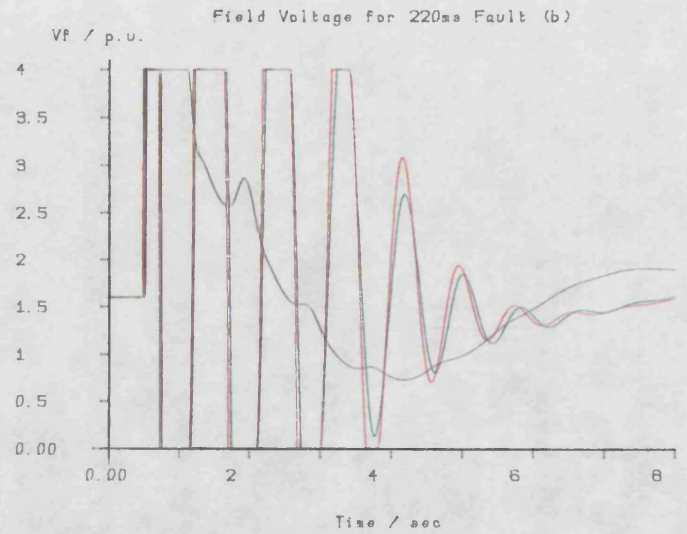
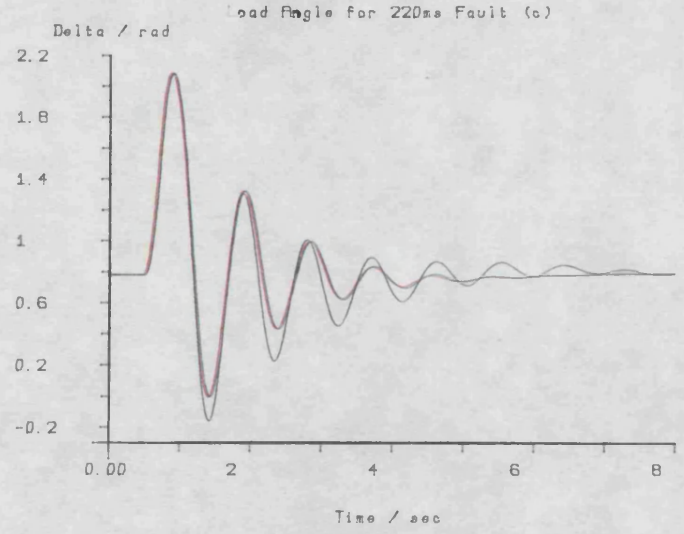

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.39


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


oad Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

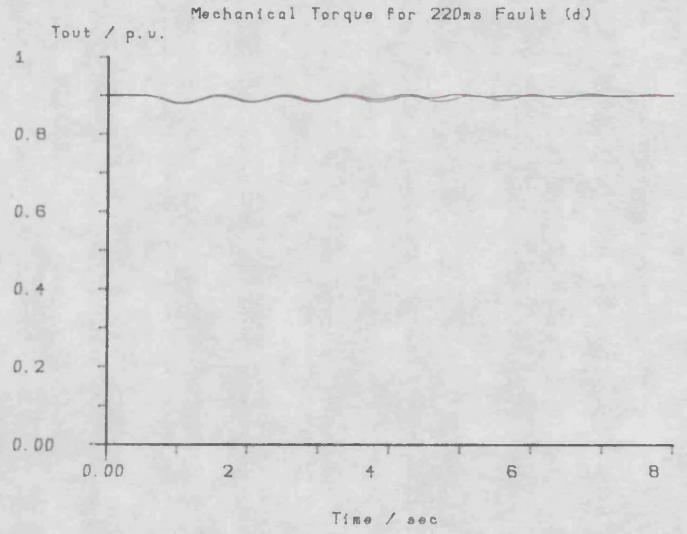
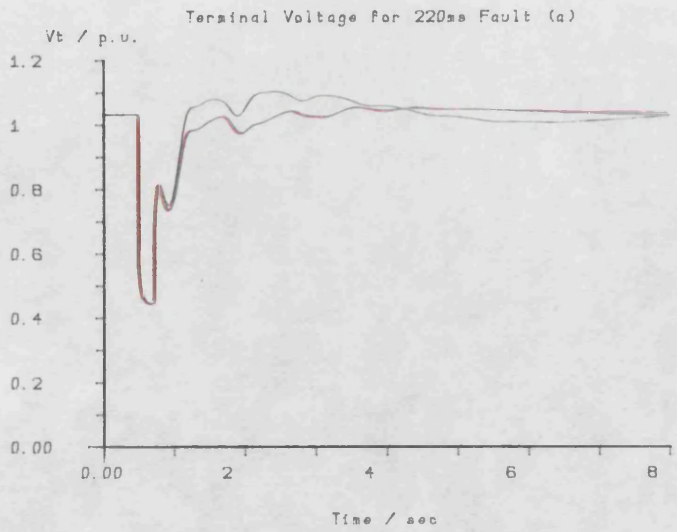
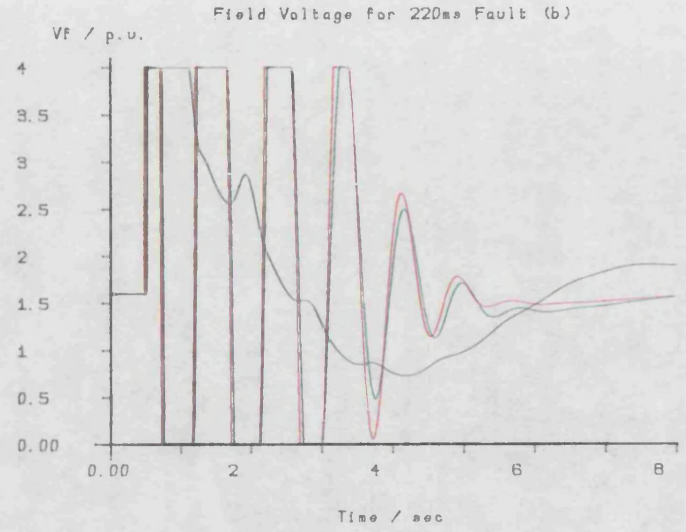
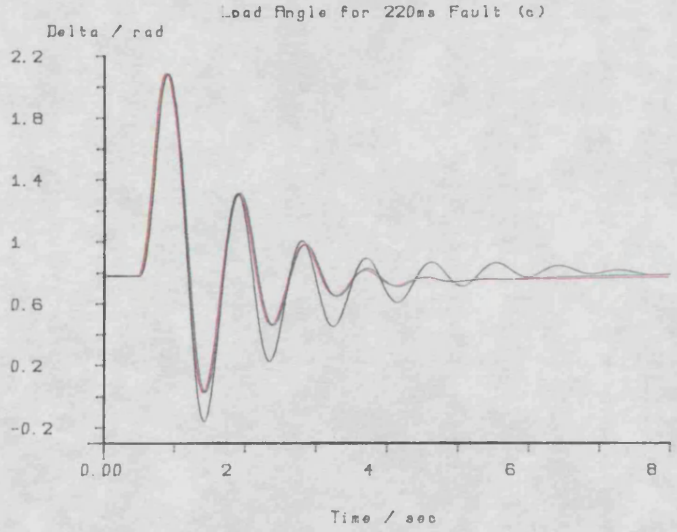

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.40


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

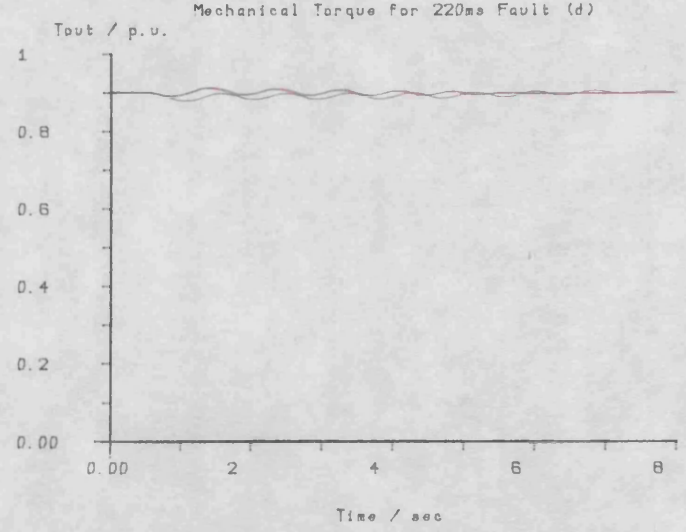
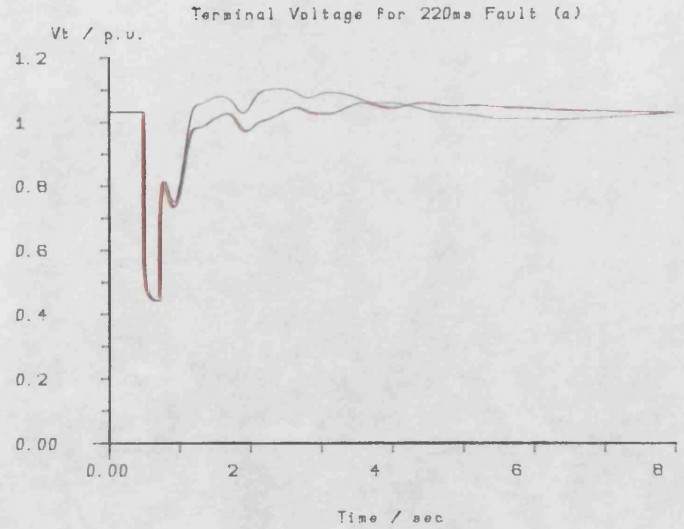
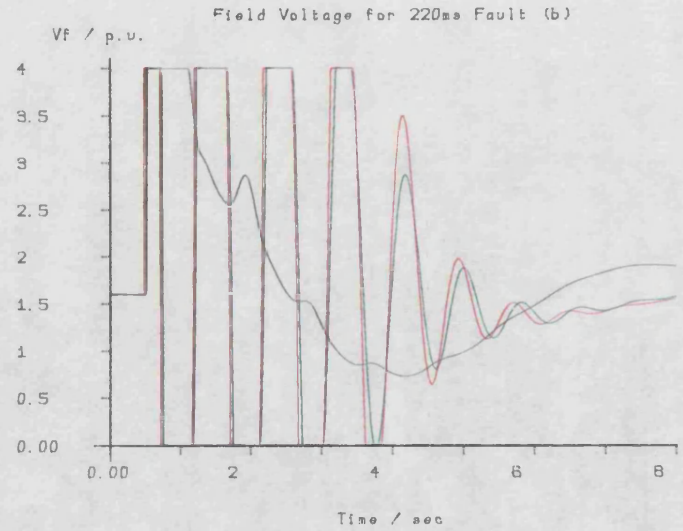
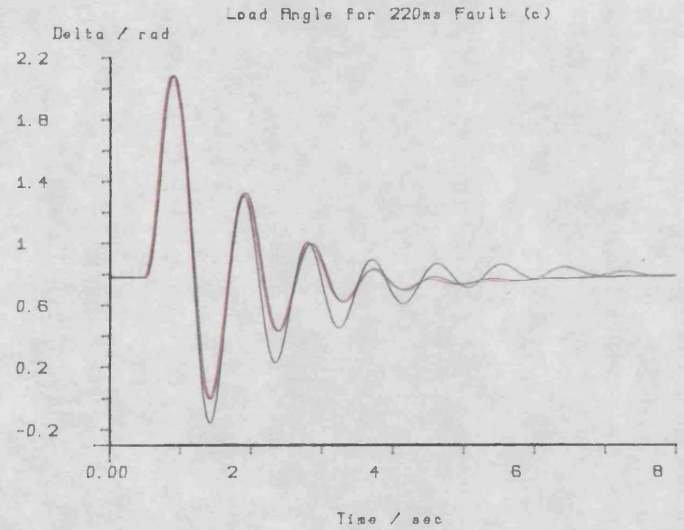

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.41


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

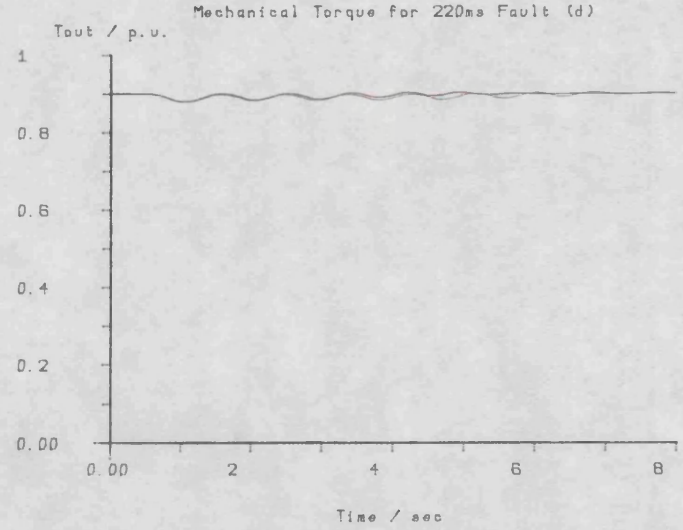
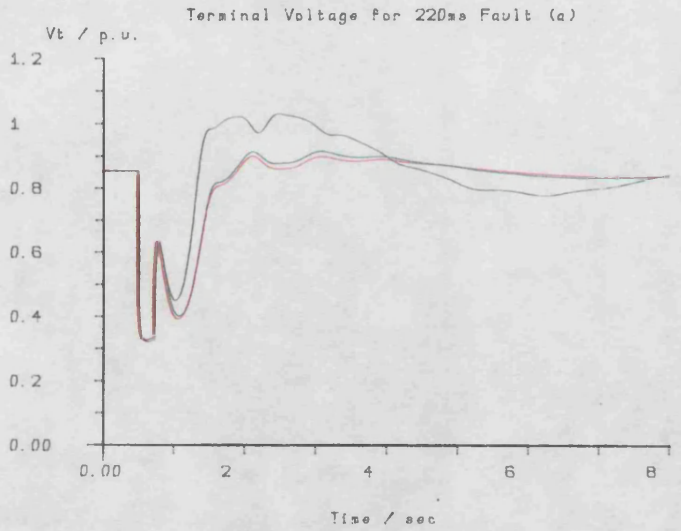

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

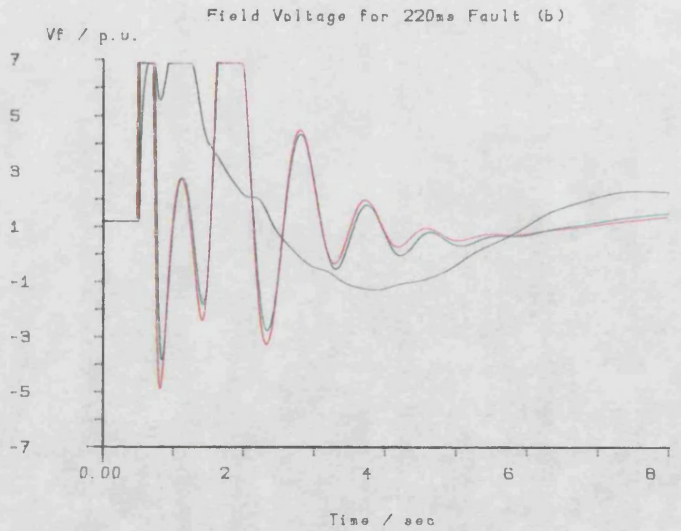
Fig 6.42

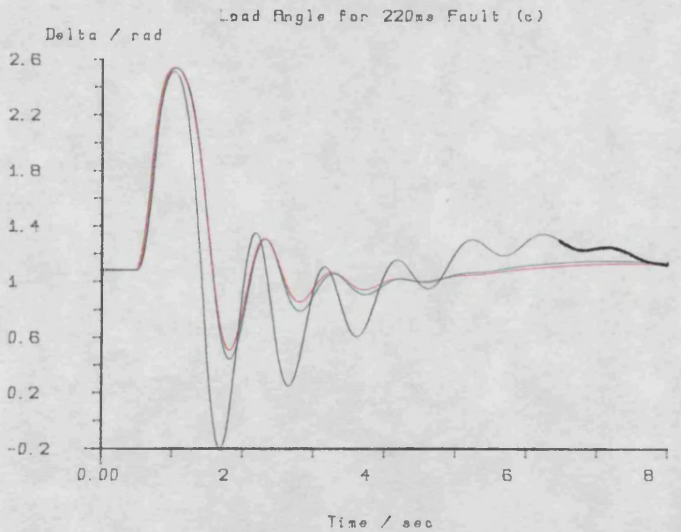
controls F9 and F10. It can be seen that the use of feedback gains optimised for operation at a power factor close to unity, perform very well in operation at a large leading power factor.

It can be concluded that the field voltage responses of all the optimal controls presented in this section are, to some extent, 'bang-bang' in their nature during the early part of the transient. Provided that the gains supplying the extra feedback signal to the excitation are sufficient in magnitude to achieve this bang-bang nature, with fast transitions between the excitation ceilings, the transient performance of the system will be insensitive to small variations in their values. Care must be taken to ensure that these gains are not given magnitudes so large that the excitation system remains in this switching mode longer than is necessary to damp out the load angle oscillations, as this would be detrimental to the steady state objective of excitation control, i.e. terminal voltage regulation.

The four pre-fault/post-fault impedance charts, Fig.6.45 to 6.48, each contain four curves; the black and red curves are obtained from a conventionally controlled generator with fault clearance times of 220ms and 140ms, respectively, while the green and blue curves were obtained with the use of extra feedback, also for fault clearance times of 220ms and 140ms, respectively. Rotor acceleration, Figs.6.45 and 6.46, and transient electrical power, Figs.6.47 and 6.48, are used as the extra feedback signals which are fed into both excitation and governing systems, Figs.6.45 and 6.47, and into the excitation system alone, Figs.6.46 and 6.48.


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.6
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

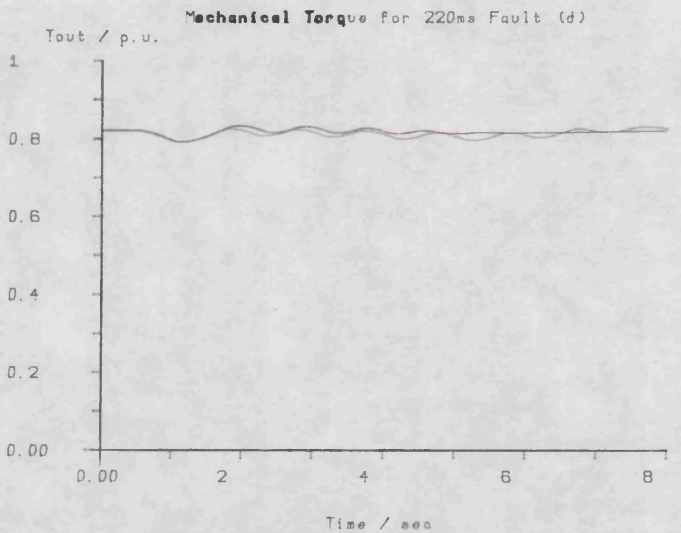
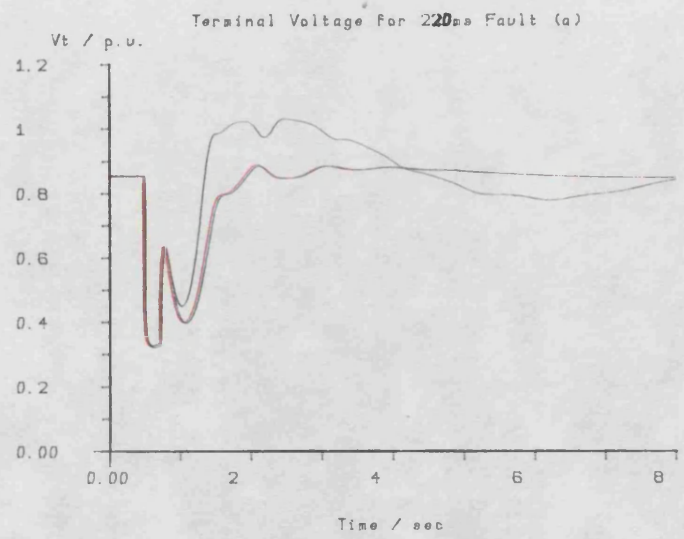
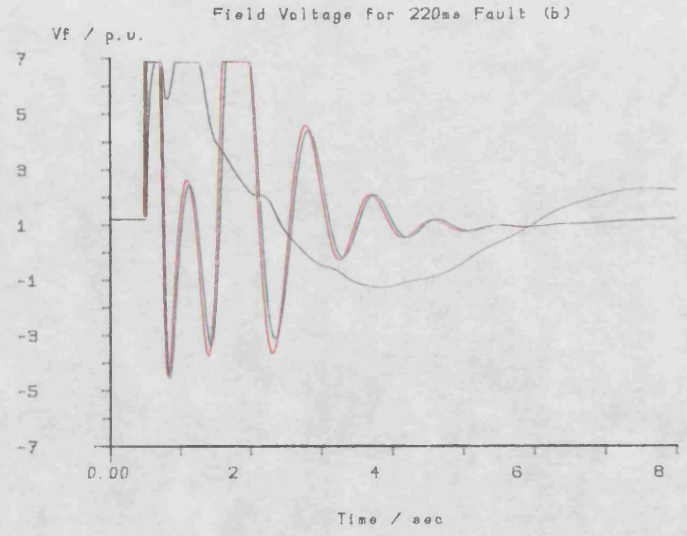
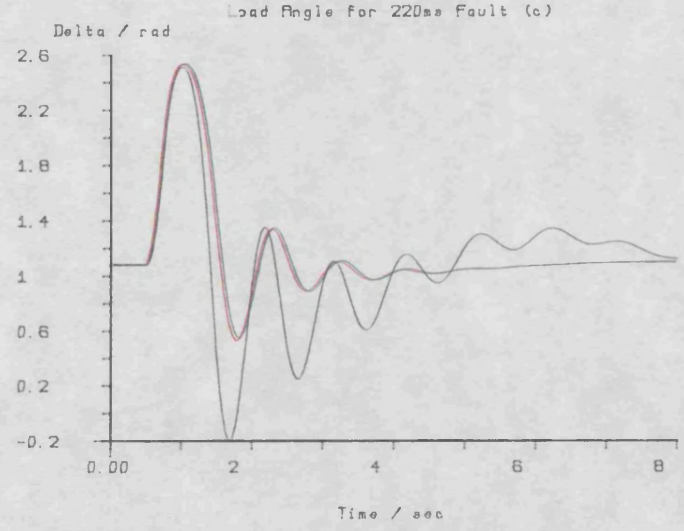

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.43


Terminal Voltage for 220ms Fault (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec


Field Voltage for 220ms Fault (b)
Vf / p.u.
7
5
3
1
-1
-3
-5
-7
0.00
2
4
6
8
Time / sec


Load Angle for 220ms Fault (c)
Delta / rad
2.6
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
2
4
6
8
Time / sec

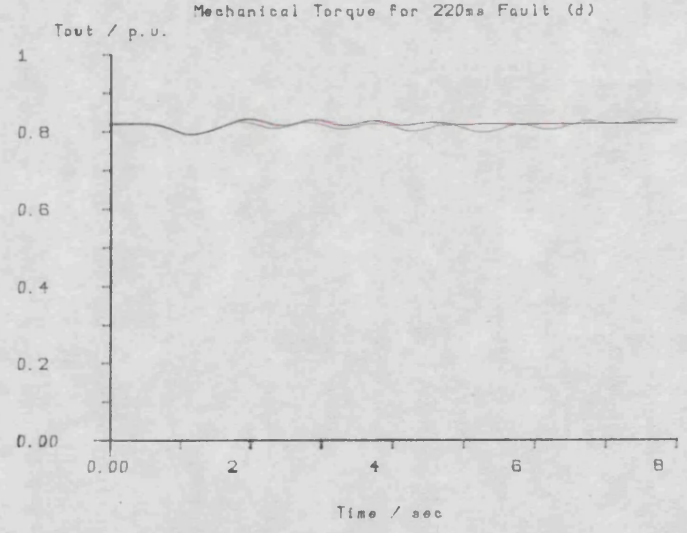

Mechanical Torque for 220ms Fault (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
2
4
6
8
Time / sec

Fig 6.44

It can be seen that none of the strategies employing the use of extra feedback make any improvement to the transient stability of the system as judged by the pre-fault/post-fault impedance chart. This is to be expected as this chart has its basis in the first swing stability criteria, and it has already been mentioned that no improvement in the size of the first peak of the load angle response has been obtained. In all cases, the curves for systems using extra feedback are below those for the corresponding conventionally controlled systems. This indicates a reduction in transient stability according to these criteria.

It can also be seen that strategies which use extra feedback into the excitation systems alone are slightly more stable than those with feedback into both excitation and governing systems. The use of transient electrical power feedback is marginally more stable than the use of rotor acceleration feedback.

In order to assess the effect that the use of rotor acceleration and transient electrical power are likely to have on the dynamic stability of the plant, a 10% reduction in the excitation system reference was made. The transients obtained by this action are plotted in Fig.6.49. These transients are colour coded as follows: black for the conventionally controlled system; red for the use of rotor acceleration fed into the excitation system alone; and green for the use of transient electrical power fed into the excitation system alone. The use of transient electrical power feedback improves the damping of the load angle response, while the use of rotor acceleration feedback produces a load angle response that is less damped than that for the conventionally controlled system. This

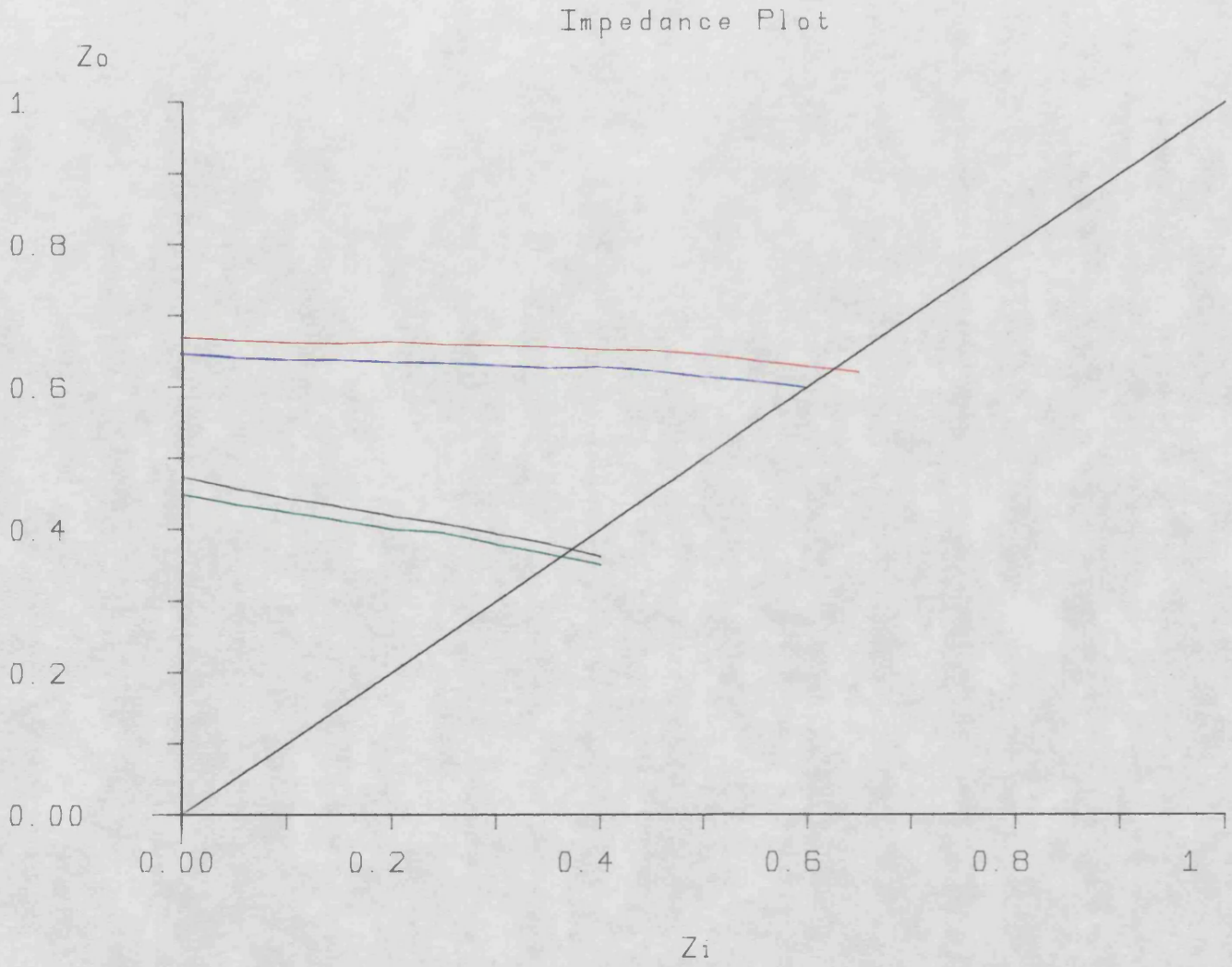


Fig 6.45

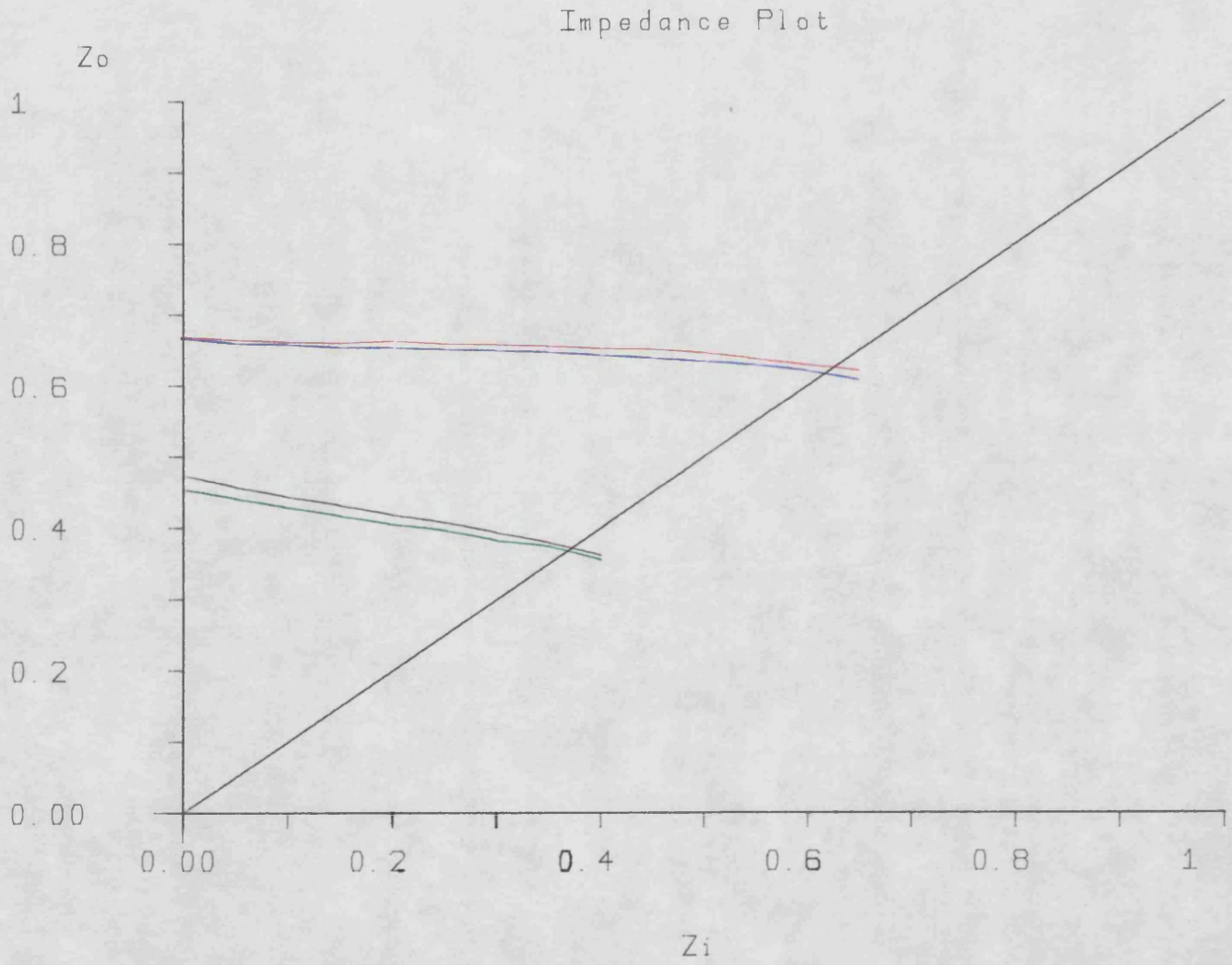


Fig 6.46

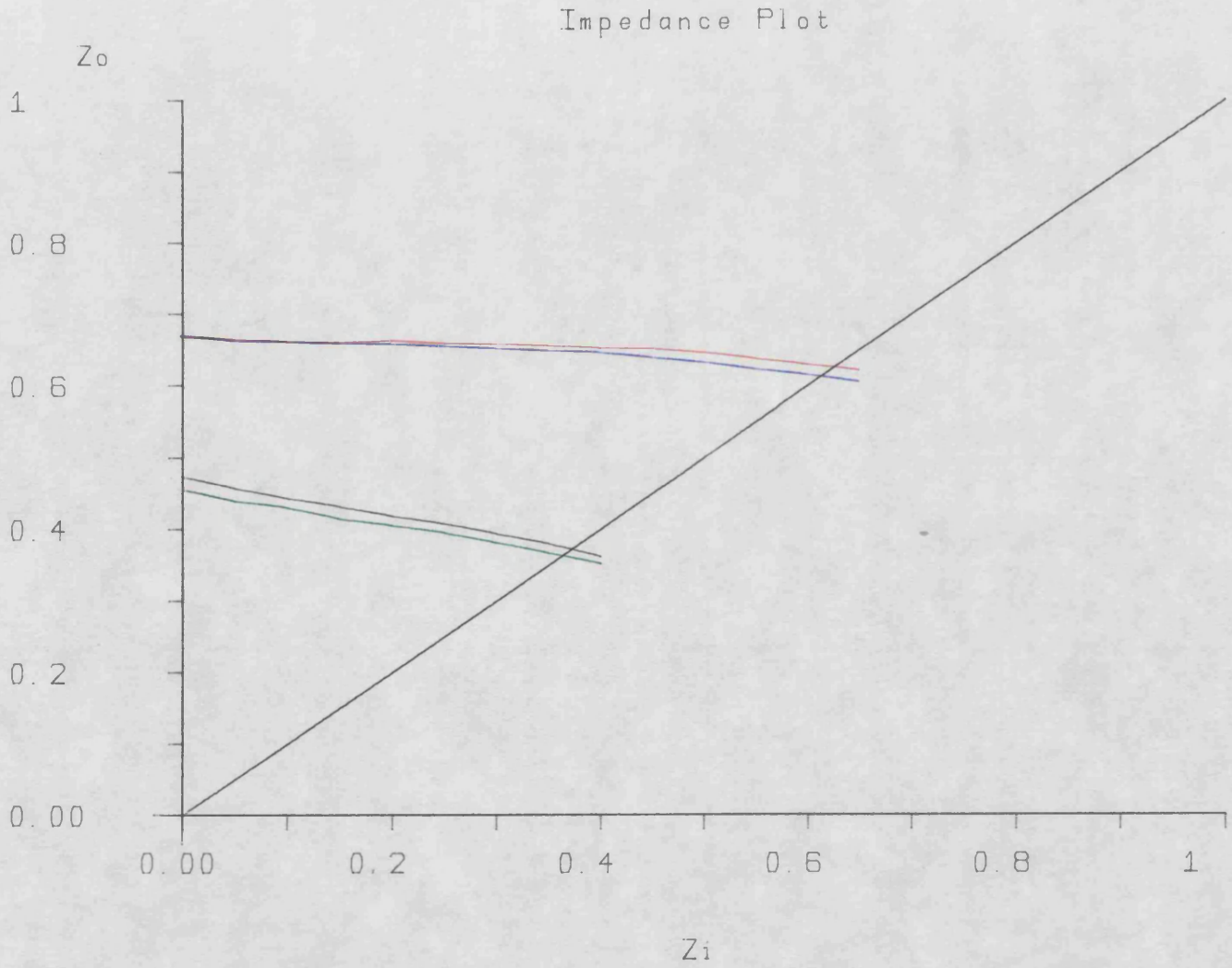


Fig 6.47

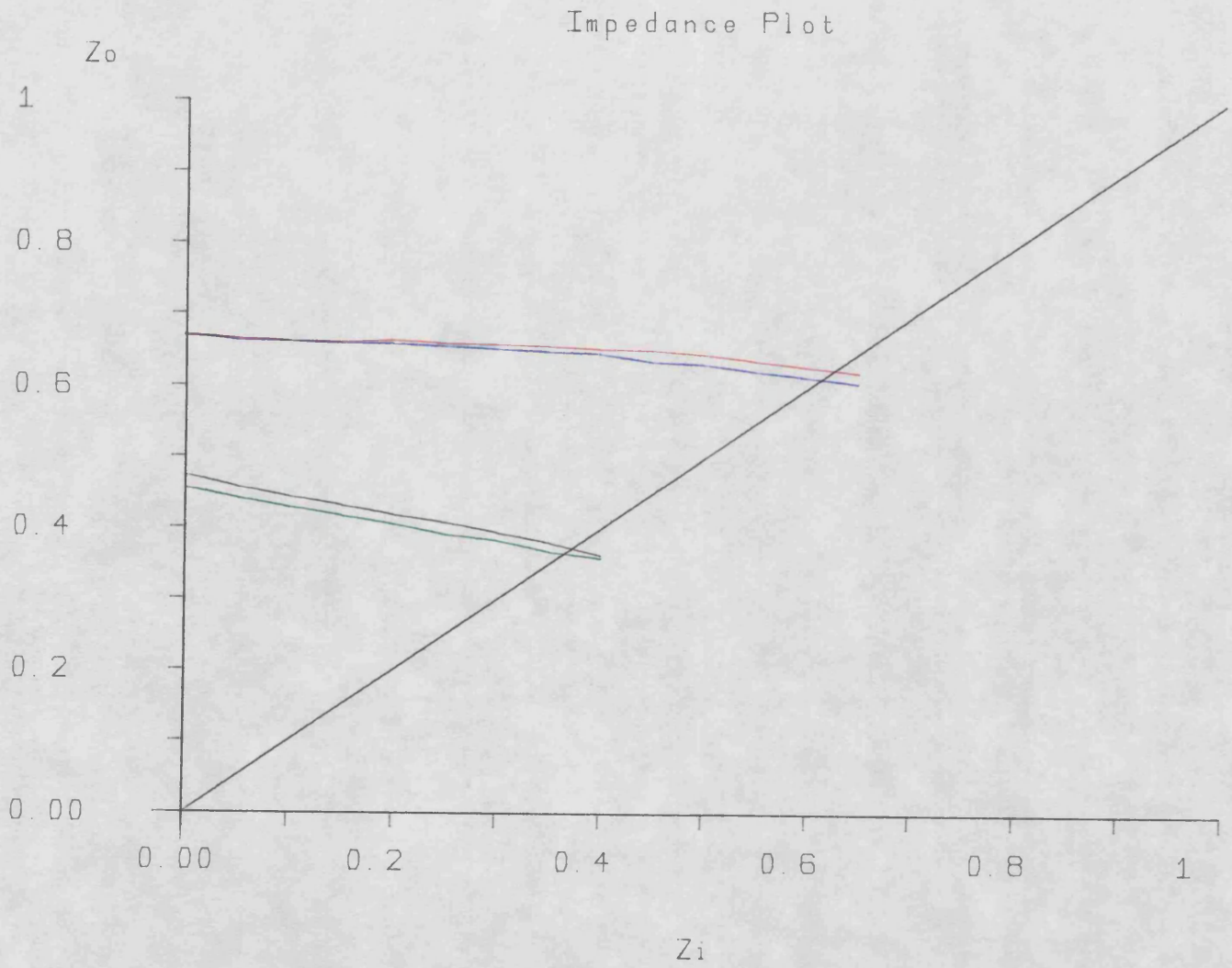
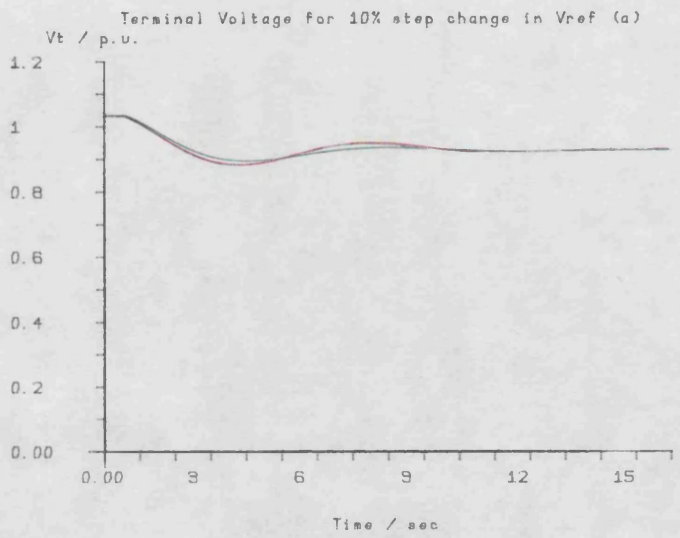
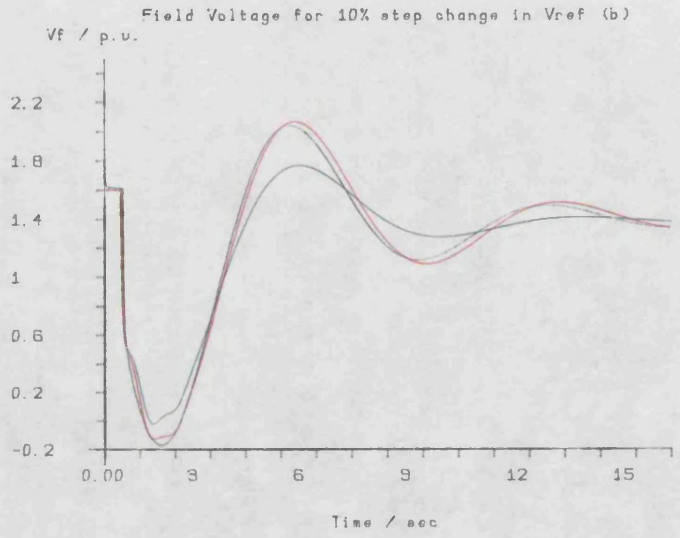


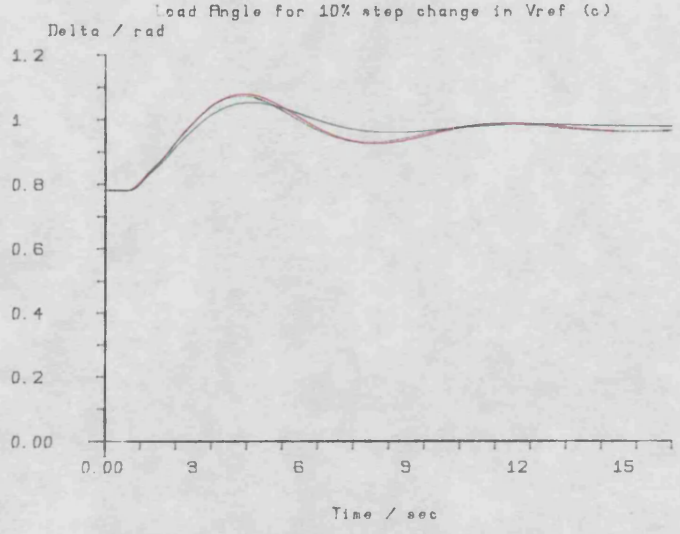
Fig 6.48

suggests that use of transient electrical power feedback will improve the dynamic stability of the generating plant, while use of rotor acceleration feedback will be detrimental to this stability. This is confirmed by the dynamic stability limit curves plotted in Fig.6.50. These are colour coded as Fig.6.49, and show the dynamic stability limit obtained by the use of transient electrical power feedback much further out to the left than the other two stability limits.

In concluding this section, it can be said that the transient performance of this Dinorwig-like system may be improved by the use of rotor acceleration or transient electrical power feedback into the excitation system alone. Additional feedback of these two signals into the governing system produces a negligible change in the transient performance due to the severe limits on the opening and closing rates of the main control valves. Despite improving the transient performance in terms of terminal voltage recovery, load angle damping, and a reduction in the size of the first negative going peak in the load angle excursion, no improvement can be made in the size of the critical first peak in the load angle swing. Judged on this basis no improvement in the transient stability of the plant is obtained. The dynamic stability of the plant is significantly improved by the use of transient electrical power feedback into the excitation system, but no corresponding improvement is obtained by the use of rotor acceleration feedback. The transient responses are insensitive to small changes in the feedback gain values which may safely be given nominal values of 0.0400 for rotor acceleration and -2.00 for transient electrical power, each fed into the excitation system alone.


Terminal Voltage for 10% step change in Vref (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec


Field Voltage for 10% step change in Vref (b)
Vf / p.u.
2.2
1.8
1.4
1
0.6
0.2
-0.2
0.00
3
6
9
12
15
Time / sec


Load Angle for 10% step change in Vref (c)
Delta / rad
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec

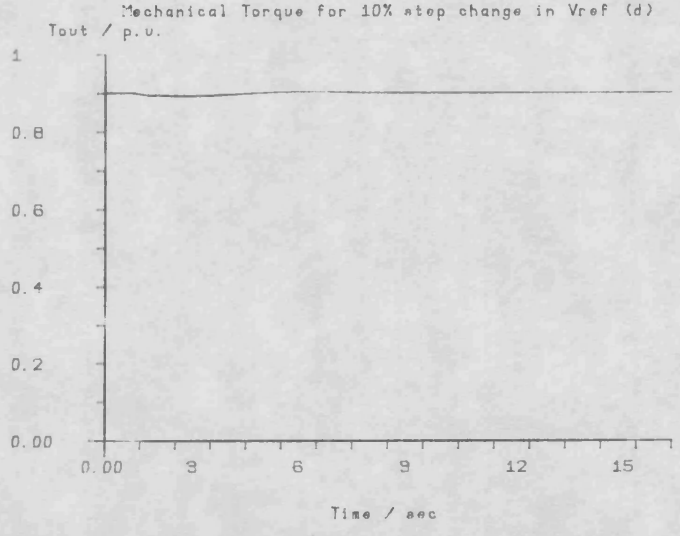

Mechanical Torque for 10% step change in Vref (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
3
6
9
12
15
Time / sec

Fig 6.49

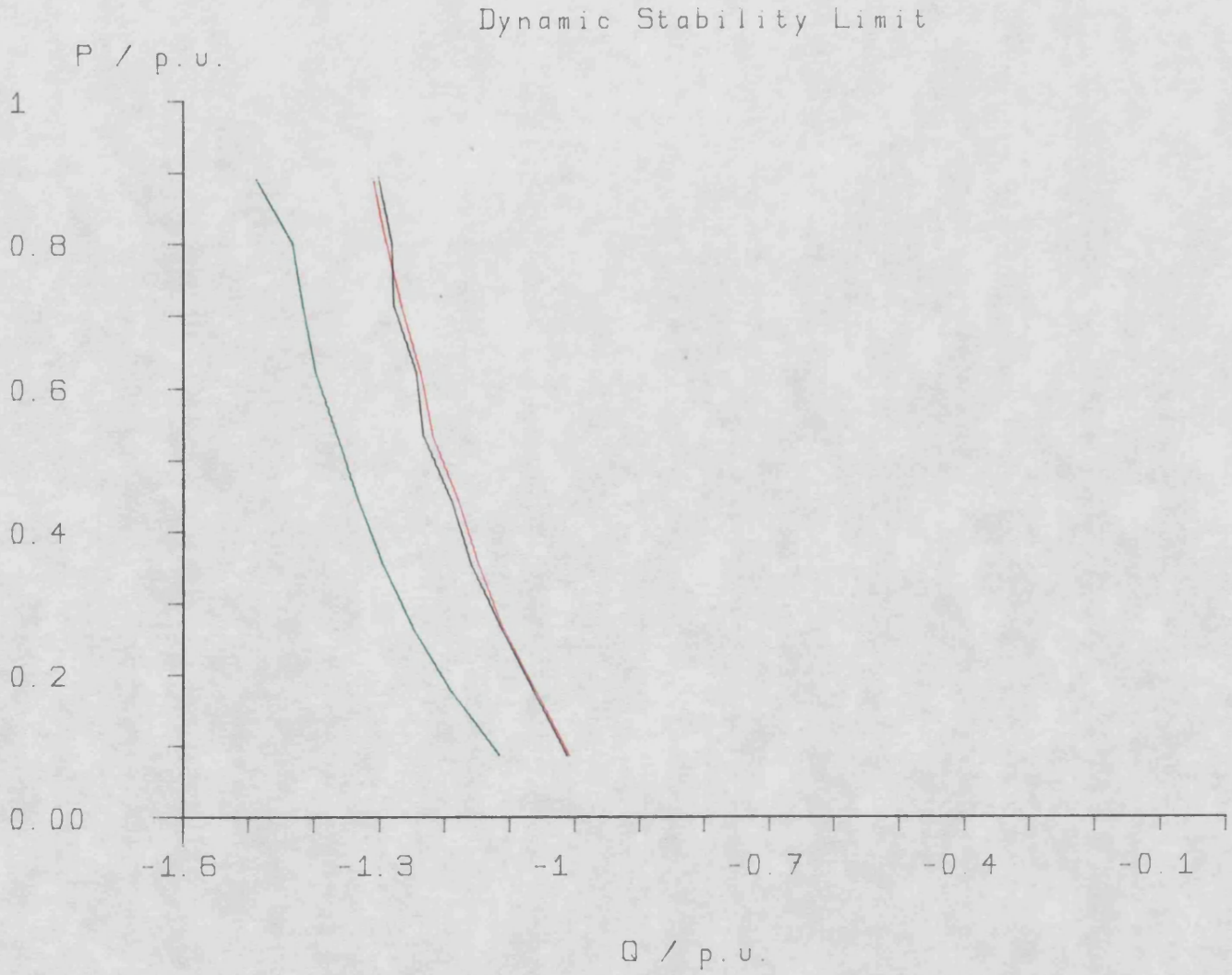

Dynamic Stability Limit
P / p.u.
1
0.8
0.6
0.4
0.2
0.00
-1.6
-1.3
-1
-0.7
-0.4
-0.1
Q / p.u.

Fig 6.50

6.3.4 Optimisation of a Diesel Generator, Finite Busbar System

This section presents the results of optimisation studies performed on a simple finite busbar system. The prime mover model considered in these studies is the diesel engine model described in Chapter 3, and the excitation system is of the form used in the previous two sections, also described in Chapter 3. A schematic diagram of this system is shown in Fig.6.51 which also shows the arrangement of the inductive load. The plant parameters for this system have been supplied by the Royal Naval Engineering College at Manadon and are given in Appendix A.

The generator load used in these studies is comprised of an inductive load, R_1 and X_1, which is permanently connected to the generator terminals, and a switched load, R_2 and X_2, which may be impacted across the first by closing the switch S1. Three different impacted loads have been used in the studies that follow, as detailed in Table 6.15.

Load	R_1	X_1	R_2	X_2	Z_1	Z_2	$Z_1//Z_2$
1	1.73	1	0.5	0.866	$2\angle 30^\circ$	$1\angle 60^\circ$	$0.687\angle 50^\circ.1$
2	1.73	1	1	1.73	$2\angle 30^\circ$	$2\angle 60^\circ$	$1.04\angle 45^\circ$
3	1.73	1	0	0	$2\angle 30^\circ$	0	0

Table 6.15

Load 1 has been chosen to overload the generator when the impacted load is applied, while Load 2 is an arbitrary load that will

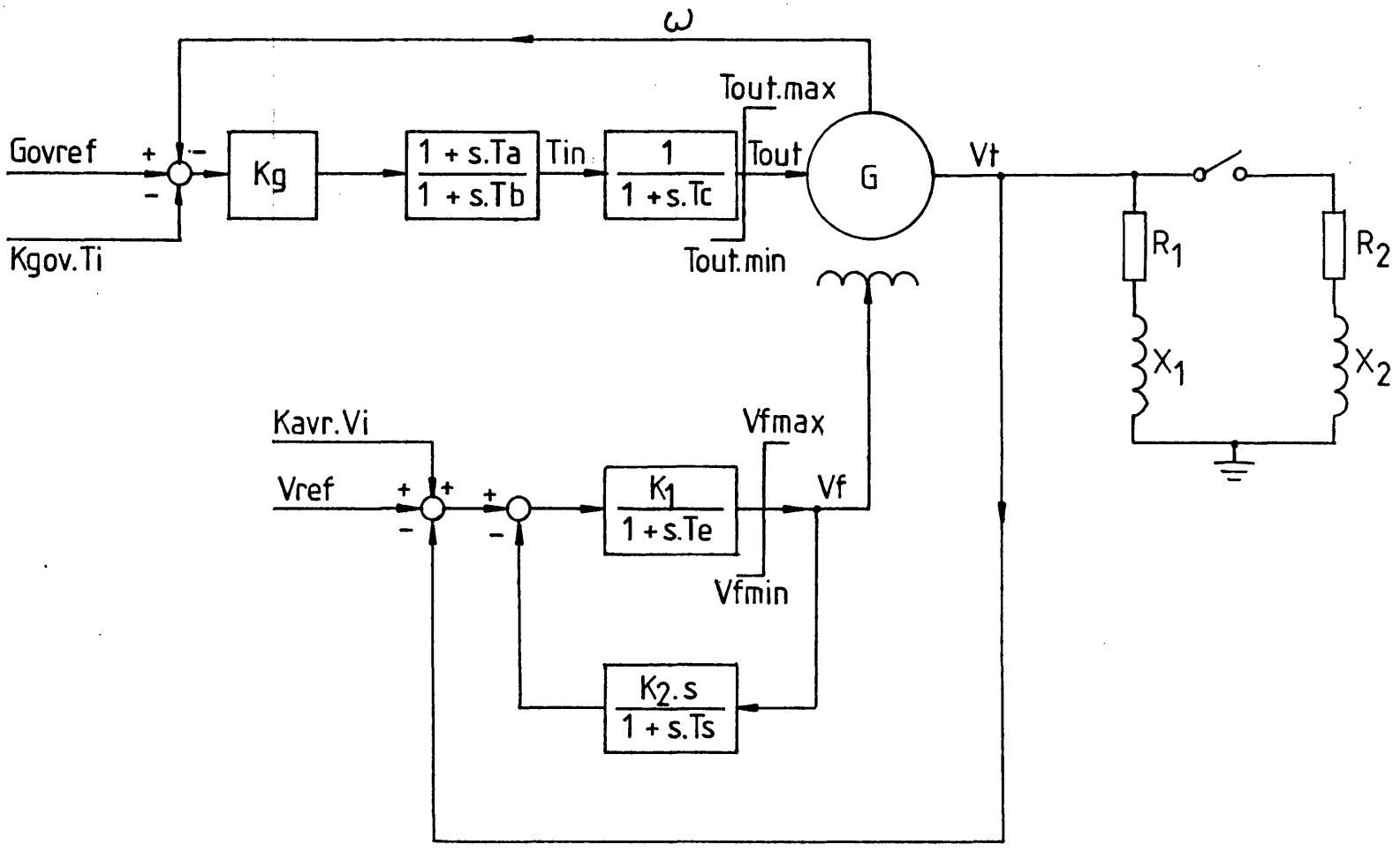


Fig 6.51 Block Diagram of the Diesel Generator, Finite Bus System

not overload the generator when the impact load is applied. Load 3 impacts a short circuit across the generator terminals.

The performance index used in the studies that follow is given in equation 6.14, which is repeated below:

$$I = \int_{t_o}^{t_1} (\Delta Vt^2 + A1.\Delta\omega^2 + A2.\Delta tout^2)\,(0.5 + t.A3)dt \quad (6.14)$$

Initially, the term $A2.\Delta tout^2$ had not been included in the performance index. The reason for its subsequent introduction will be given later but, in the meantime, the coefficient A2 is set to zero. The period of integration used throughout these studies is 8 seconds, starting from the time that the impact load is applied.

Optimisations were performed for each of the three loads, with each of the three feedback signals available in turn to both the excitation and governing systems. The results given in Table 6.16 were obtained with A1 set to 100, applying a heavy penalty to the speed deviation term, $\Delta\omega^2$, while those given in Table 6.17 were obtained with A1 set to the much lower value of 3. Both sets of results were obtained with A3 set to unity, applying a small time penalty to 'late' deviations.

Initial Conditions

Vt = 1.003 p.u. Werr= 1.68 x 10^{-4} P = 0.435 p.u. Q = 0.251 p.u.

Vf = 1.46 p.u.

Control	Signal	Load	Kavr	Kgov	P.I.	Fig.
B1	None	1	0	0	0.656	6.51
B2	$P^2\delta$	1	22.1	6.30	0.0314	
B3	ΔPe	1	-0.333	-0.0512	0.0444	
B4	Δid	1	-0.140	-0.0282	0.0392	
B8	None	2	0	0	0.0405	6.52
B9	$P^2\delta$	2	3.22	6.84	0.00931	
B10	ΔPe	2	-0.327	-0.0521	0.0102	
B11	Δid	2	-0.110	-0.0230	0.00855	
B21	None	3	0	0	1.14	6.53
B25	$P^2\delta$	3	3.18	6.68	1.04	
B26	ΔPe	3	-3.19	-0.0451	1.06	
B27	Δid	3	-0.126	0.00770	1.08	

Impact Time = 1s Vfmax = 4 p.u. Vfmin = -4 p.u. A1 = 100

A2 = 0 A3 = 1

Table 6.16

Initial Conditions as Table 6.16

Control	Signal	Load	Kavr	Kgov	P.I.	Fig.
B1	None	1	0	0	0.0372	
B5	$P^2\delta$	1	2.47	5.71	0.0196	6.54
B6	ΔPe	1	-0.116	-0.0668	0.0152	
B7	Δid	1	-0.0651	-0.0400	0.0148	
B8	None	2	0	0	0.00721	
B12	$P^2\delta$	2	3.60	7.15	0.00594	6.55
B13	ΔPe	2	-0.152	-0.0527	0.00498	
B14	Δid	2	-0.0610	-0.0258	0.00511	
B21	None	3	0	0	1.039	
B22	$P^2\delta$	3	3.01	0.433	1.028	6.56
B23	ΔPe	3	-2.64	-0.0445	1.030	
B24	Δid	3	-0.232	-0.00810	1.036	

Impact Time = 1s Vfmax = 4 p.u. Vfmin = -4 p.u. A1 = 3

A2 = 0 A3 = 1

Table 6.17

The transient response corresponding to the controls recorded in Tables 6.16 and 6.17 are shown in Figs.6.51 to 6.56, as indicated in the tables. The responses in each of these figures is colour coded: black for the base case, with no extra feedback; red for the use of rotor acceleration feedback; green for the use of transient electrical power feedback; and blue for the use of transient direct axis current feedback.

Before considering the effect of these extra feedback signals, some discussion of the transient responses obtained by impacting the various loads onto a conventionally controlled machine is necessary. These responses are the base case responses of Figs.6.51 to 6.53, plotted in black.

The base case transient response for Load 1 is shown in Fig.6.51. When the impacted load is applied, the terminal voltage falls and the terminal voltage error drives the excitation system into positive saturation in order to restore the terminal voltage to its steady state value. The terminal voltage rises in response to this action and, as it approaches the steady state value, the excitation system brings the field voltage out of saturation. During the same period of time, the generator detects the need to supply more prime mover power in order to meet the extra power demanded by the electrical load. This extra power is initially supplied by the conversion of energy stored in the motion of the rotor into electrical energy, with the result that the rotor decelerates. The increase in magnitude of the speed error (Werr) is used by the governor to open the fuel rack, supplying more fuel to the diesel engine and, consequently, more torque to the generator shaft. However, by design the torque necessary to supply sufficient current to the load at the required terminal voltage (which the excitation system is able to maintain) is more than the diesel engine can develop, and the mechanical torque output reaches its limit. Were the impacted load to have been applied for an indefinitely long period of time, the machine rotor would have continued to slow down either, until the voltage change due to the fall in speed had taken the excitation system into positive saturation and the machine had settled at a power level it could

support, or until the engine had stalled. The impacted load is, in fact, removed after one second and as a result the terminal voltage overshoots, causing the terminal voltage error to reduce the field voltage via the action of the excitation system. When the impacted load is removed, the excess power being supplied by the prime mover at that moment is used to accelerate the generator shaft, and the speed error is restored towards the initial, small, value. As the speed error goes positive, the prime mover torque comes out of saturation and the machine settles back into steady state operation at its original operating point.

Fig.6.52 shows the base case response for a one-second impact of Load 2 in black. This response takes much the same form as that for Load 1. Note that the terminal voltage error caused by the application of the impacted load is not sufficient to force the excitation system into limit. Likewise, the speed deviation, which reaches a little over 2% as compared with something approaching 9% in the case of Load 1, does not drive the prime mover torque into saturation. Had the load been impacted for a longer period of time, the generator would have successfully achieved steady state operation at a new operating point.

The base case transient response for the short circuit applied by Load 3, Fig.6.53, is quite different in form to the previous responses. In this case, no amount of excitation can restore the terminal voltage while the short circuit is applied. Thus the field voltage is driven quickly into positive saturation when the short circuit is made. Since the terminal voltage is zero, no power is transferred to the load and the excess supplied by the prime mover

causes the generator to accelerate in much the same manner as that with the generator connected to the infinite bus bar. This increase in rotor speed causes the governor loop to reduce the prime mover output torque. When the short circuit is removed, the terminal voltage is restored quickly to its original level by the action of the excitation system, and the prime mover torque is increased as the demand for electrical power to supply the load causes the machine to decelerate.

Before considering the remaining transient responses of Figs.6.51 to 6.52, one or two general observations may be made about the nature of these base case responses. First, in all cases when the impact load is applied the terminal voltage falls, resulting in positive forcing of the field voltage in an attempt to restore the terminal voltage. Also, in the first two cases, Load 1 and Load 2, more electrical power is demanded from the generator when the impacted load is applied, and consequently the rotor decelerates. When the short circuit of Load 3 is applied, the rotor experiences an acceleration due to a loss in demand for electrical power. Consider the sign of the gain through which rotor acceleration is fed into the excitation system. If it is negative then, in the case of Loads 1 and 2, the use of acceleration feedback will assist in forcing the field voltage into positive saturation when the impact load is applied, and so assist in the terminal voltage recovery. In the case of Load 3, a positive gain is required to give the same assistance towards saturating the field voltage. Similar considerations may be given to the sign of the corresponding gain for the use of transient electrical power feedback into the excitation system. These considerations are somewhat naive and ignore the fact that an optimal

controller may sacrifice the response at one point in time in order to improve the overall response throughout the period of interest.

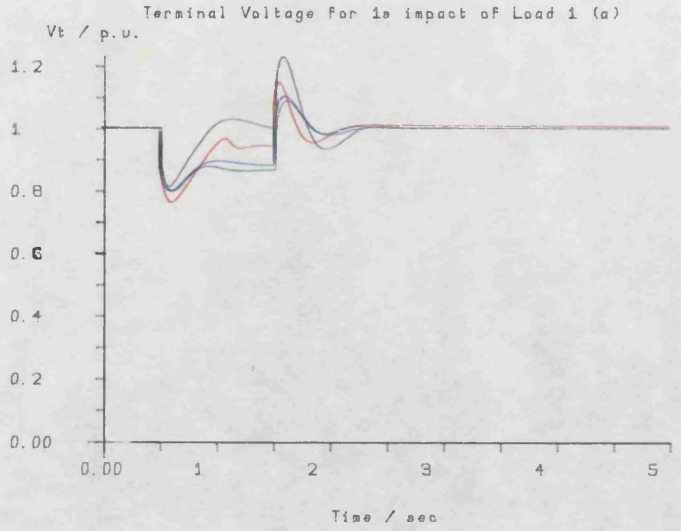
The responses of Fig.6.51 show the transients that occur with the use of controls B1-B4. The transients corresponding to control B1 have already been discussed at some length. The transient response obtained by use of control B2, shown in red, show that use of rotor acceleration feedback offers a large reduction in the speed error present when the impacted load of Load 1 is applied. This is consistent with the large weighting factor in the performance index which penalises the speed error. This reduction in speed error is obtained at the cost of the terminal voltage response. The sign of the gain feeding rotor acceleration into the excitation system is positive and this is responsible for the drop in field voltage at the instant the load is impacted, and the slowness of the subsequent climb towards the positive excitation ceiling brought about by the terminal voltage error. This results in the large undershoot observed in the terminal voltage response for control B2. While the impacted load remains applied, the terminal voltage settles towards a lower steady state value than that prior to its application. This reduces the amount of power supplied to the inductive load, and consequently the prime mover is able to supply sufficient torque without the field voltage remaining saturated while the impacted load remains connected. The 'noisy' form of the field voltage and prime mover torque responses will be discussed later.

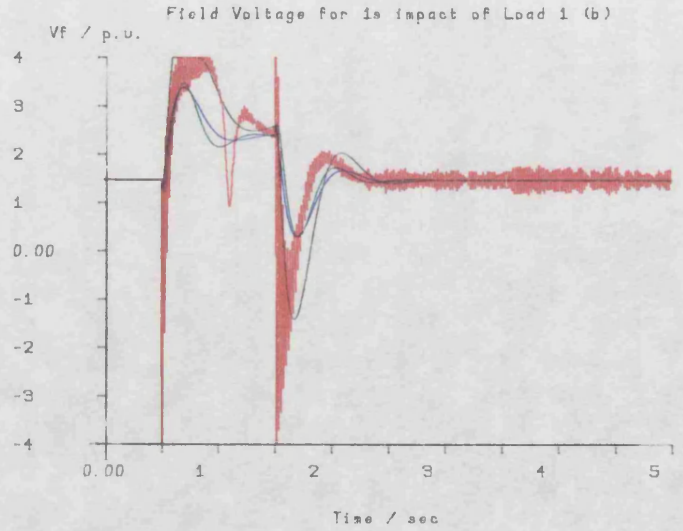
In a similar manner, for controls B3 and B4 the extra signals are fed back through gains which are beneficial to the rotor speed response at the cost of the terminal voltage response. In these two

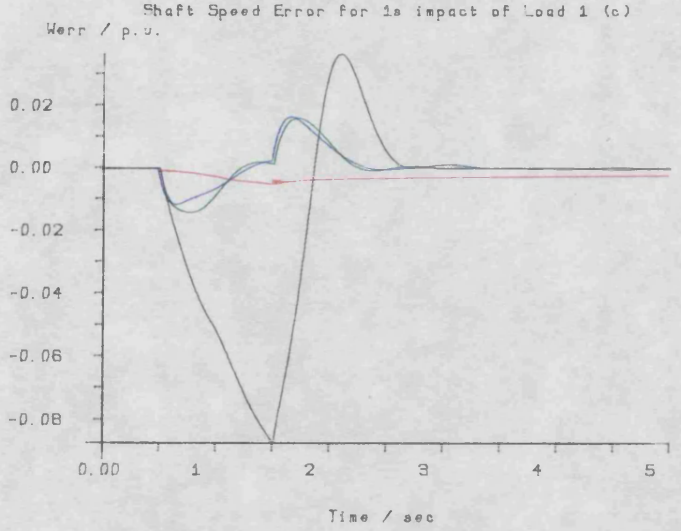
cases, the field voltage does not reach saturation despite the large terminal voltage error, and terminal voltage error reduces the electrical power output to a level than can be sustained by the prime mover without encountering the torque ceiling.

Fig.6.52 shows again that use of rotor acceleration feedback offers a very large reduction in the rotor speed deviation. In this case, control B9, Load 2 has been used and, since the prime mover output torque does not reach its limit in the base case response of Fig.6.52, control B8, it is not necessary to force a large terminal voltage error in order to reduce the electrical power output to a level that can be sustained. There is very little difference between the terminal voltage responses obtained for controls B8 and B9. Controls B10 and B11, feedback of transient electrical power and transient direct axis current respectively. Both offer an improvement in the speed response over the base case, control B8. The terminal voltage error is due to the change brought about by the load change such that, were the impacted load to remain applied, the two signals ΔPe and Δid would assume steady non-zero values. A 'washout' filter has not been used to derive these transient signals, but, with the time constants of such a filter set arbitrarily in the order of 5 to 10 seconds or more, there would be little difference in the responses obtained for a short, 1 second, application of a load change. Assuming a steady state can be reached, rotor acceleration will reach a zero steady state value. The field voltage and prime mover torque responses for control B9, using rotor acceleration feedback, appear once more to be very 'noisy'

Fig.6.53 shows the various transient responses obtained when the


Terminal Voltage for 1s impact of Load 1 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 1 (b)
Vf / p.u.
4
3
2
1
0.00
-1
-2
-3
-4
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 1 (c)
Werr / p.u.
0.02
0.00
-0.02
-0.04
-0.06
-0.08
0.00
1
2
3
4
5
Time / sec

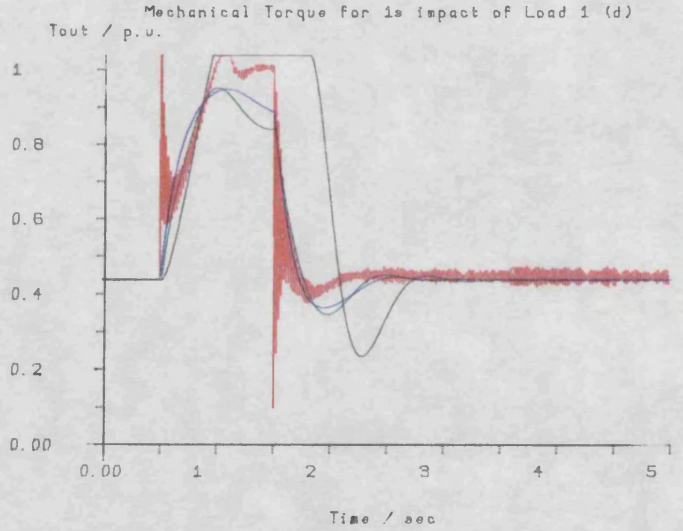
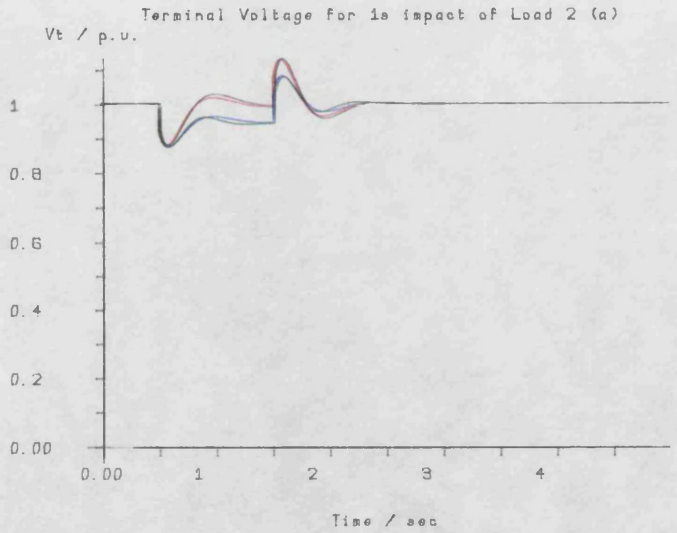
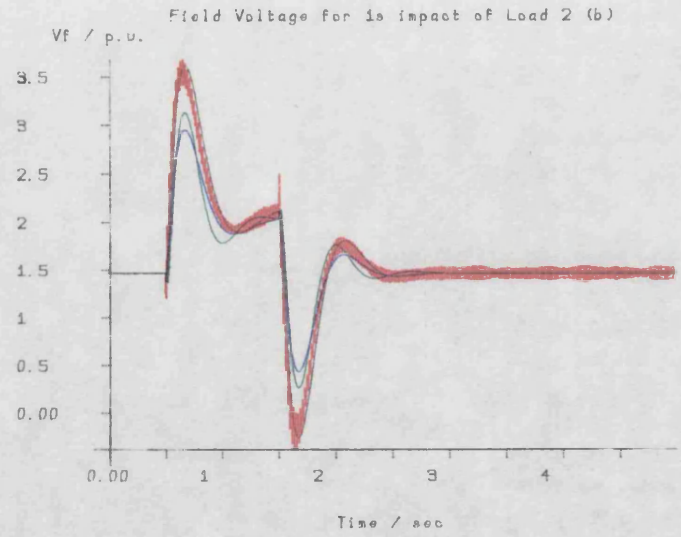
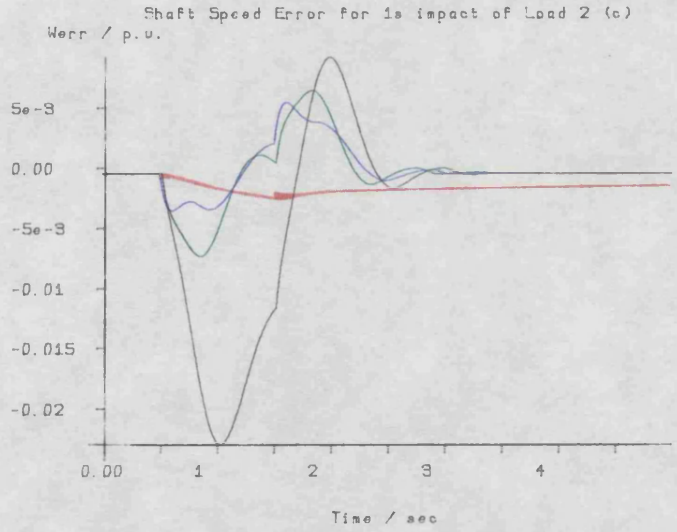

Mechanical Torque for 1s impact of Load 1 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.51


Terminal Voltage for 1s impact of Load 2 (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec


Field Voltage for 1s impact of Load 2 (b)
Vf / p.u.
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
Time / sec


Shaft Speed Error for 1s impact of Load 2 (c)
Werr / p.u.
5e-3
0.00
-5e-3
-0.01
-0.015
-0.02
0.00
1
2
3
4
Time / sec

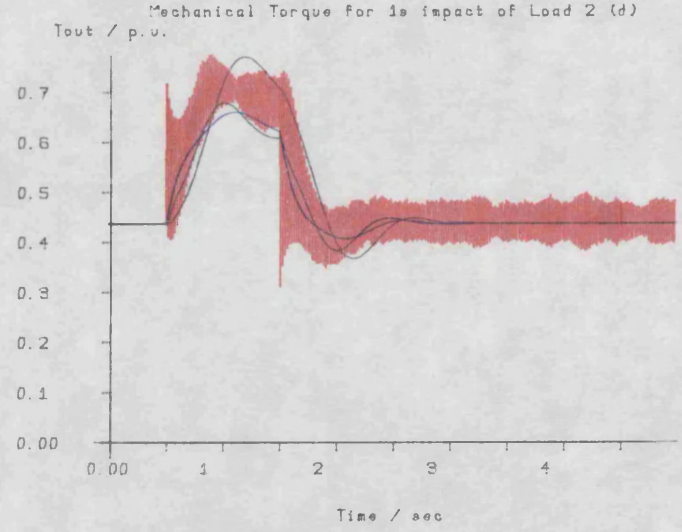
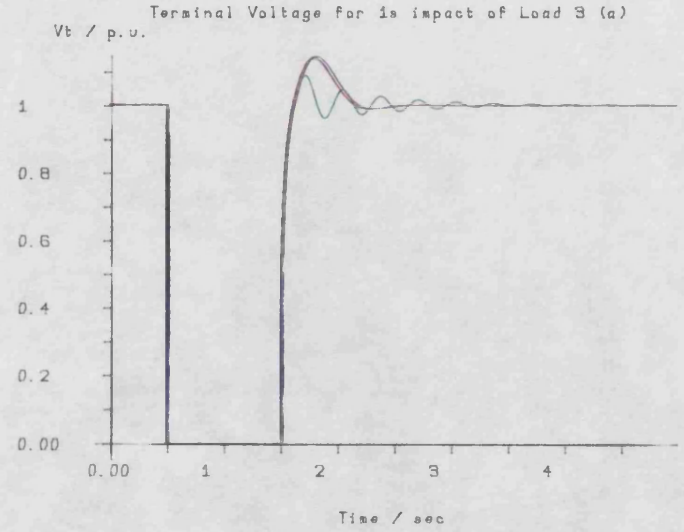
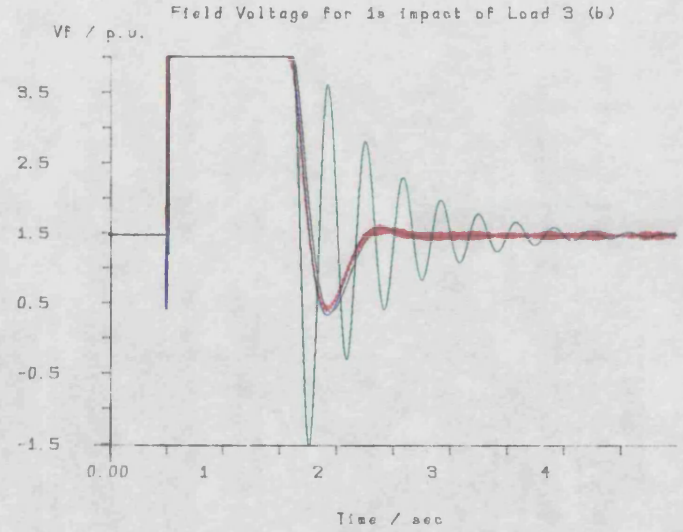
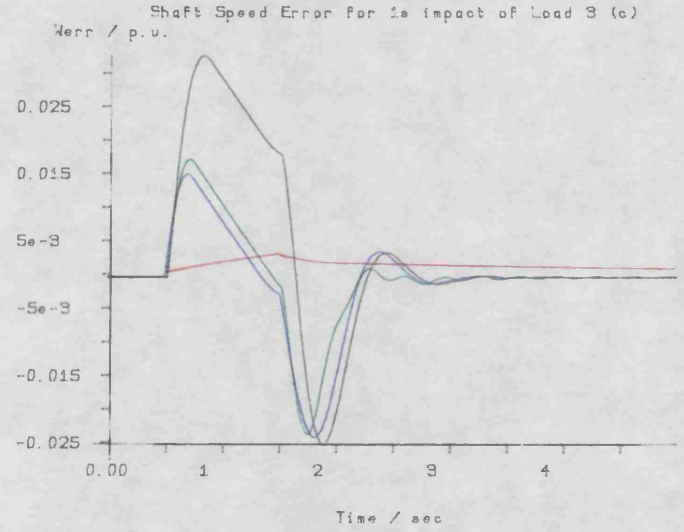

Mechanical Torque for 1s impact of Load 2 (d)
Tout / p.u.
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.00
0.00
1
2
3
4
Time / sec

Fig 6.52


Terminal Voltage for 1s impact of Load 3 (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec


Field Voltage for 1s impact of Load 3 (b)
Vf / p.u.
3.5
2.5
1.5
0.5
-0.5
-1.5
0.00
1
2
3
4
Time / sec


Shaft Speed Error for 1s impact of Load 3 (c)
Werr / p.u.
0.025
0.015
5e-3
-5e-3
-0.015
-0.025
0.00
1
2
3
4
Time / sec

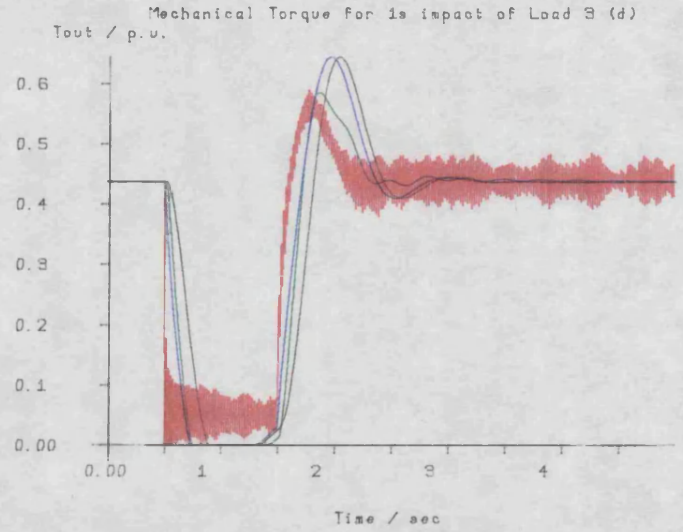

Mechanical Torque for 1s impact of Load 3 (d)
Tout / p.u.
0.6
0.5
0.4
0.3
0.2
0.1
0.00
0.00
1
2
3
4
Time / sec

Fig 6.53

short circuit of Load 3 is impacted across the generator terminals. The use of rotor acceleration feedback, control B25, once more offers a very large reduction in the rotor speed deviation when the impacted load is applied. Controls B26 and B27 both offer smaller improvements in the speed response over the base case, control B21. The terminal voltage responses obtained for controls B21, B25 and B27, the base case, rotor acceleration feedback, and transient direct axis current feedback, respectively, are indistinguishable, whilst that of control B26, transient electrical power feedback, is questionably an improvement on the base case response. There is a significant reduction in the size of the terminal voltage overshoot when control B26 is used, but the subsequent response is highly oscillatory. These oscillations are also present in the field voltage response for control B26, shown in green on Fig.6.53.

The large penalty on the speed deviation term of the performance index was reduced. The system was then re-optimised for each of the three extra feedback signals, with each of the three loads, to obtain the optimum gains shown in Table 6.17 and the transient responses of Figs.6.54 to 6.56.

Comparing the two sets of transient responses obtained when Load 1 is applied to the generator terminals, it can be seen that the speed error responses obtained with the reduced penalty on speed deviations, Fig.6.54, are slightly larger in magnitude than those obtained with the larger penalty, Fig.6.51. Relatively speaking, this reduction in the penalty associated with rotor speed deviation has increased the penalty associated with the terminal voltage error term, ΔVt^2. The terminal voltage responses of Fig.6.54 show this

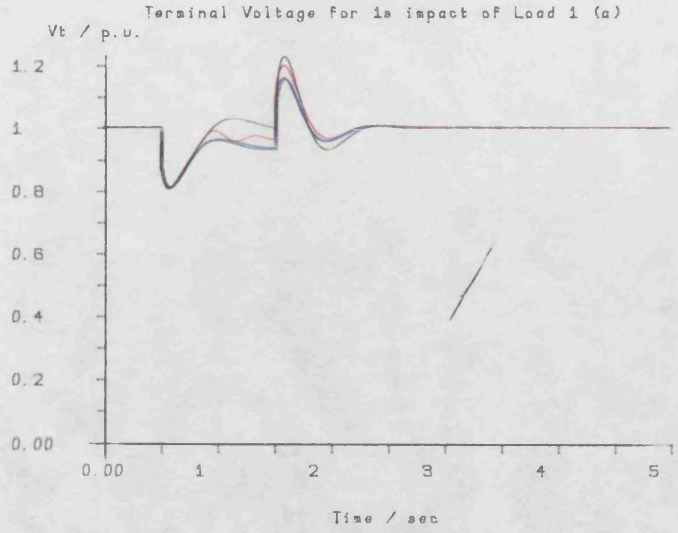
change of emphasis as the responses obtained for controls B5 to B7 achieve a smaller terminal voltage error than those of the corresponding controls, B2 to B4, Fig.6.51. Consequently, in order to supply sufficient electrical power to the load at these slightly higher terminal voltages, the prime mover torque reaches its limit. The field voltage responses for all three controls, B5 to B7, saturate quickly following the application of the impacted load. This action removes the large undershoot in terminal voltage that was observed with control B2 in Fig. 6.51, from the terminal voltage transient for control B5.

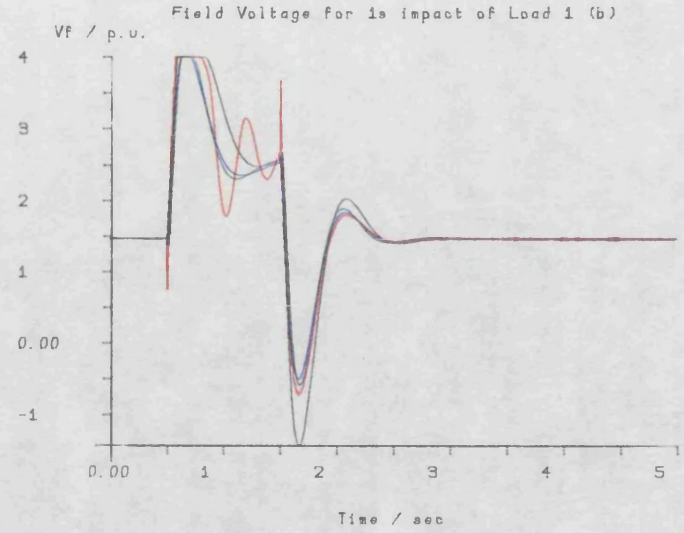
Load 2 was used to obtain the transient responses of Fig.6.55. These responses also show an improved set of terminal voltage responses at a small cost to the improvement in the speed error response when compared with the responses of Fig.6.52. Control B12, using rotor acceleration feedback, has very 'noisy' transient responses for field voltage, speed error and prime mover output torque. This latter response is particularly alarming and it is clear that a real diesel engine could not supply torque in the manner indicated by this response. It can be seen from Fig.6.55 that the speed error response has a large amplitude, high frequency component which will result in a large amplitude, high frequency acceleration signal. This large acceleration signal enters both the excitation and governing systems through gains that are both greater than unity. In the case of the governing system, the high frequency component of the signal output by the lead-lag network in the forward path is in the region of $\pm$15 p.u. Such a large amplitude signal is capable of moving the output of the fuel rack from 0 p.u. to 1 p.u. or vice versa in approximately 8ms with a fuel rack time constant of 0.11 sec as

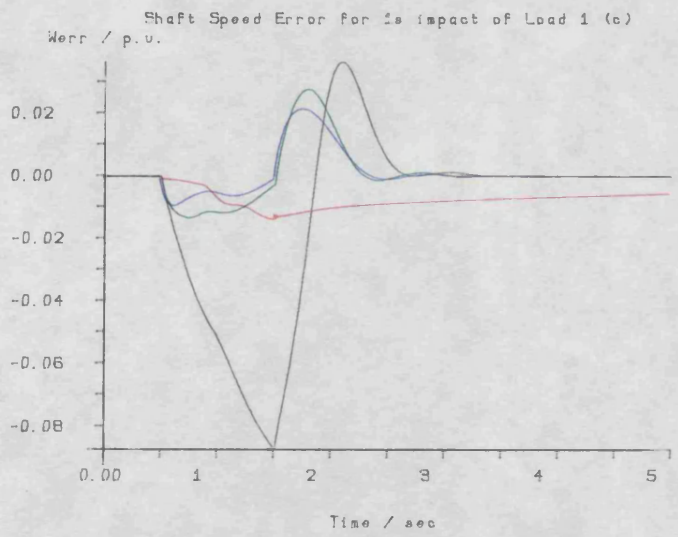
given in Appendix A. Observations of this sort cast doubts on the validity of the prime mover model in use and steps to alleviate this problem will be discussed shortly.

Fig.6.56 contains transient responses obtained by applying the short circuit of Load 3 to the generator terminals with and without the use of the extra feedback signals. The controls B21 to B24 which result from the optimisations, all offer a significant improvement in the speed error response over the base case speed error response. Use of acceleration feedback offers a dramatic improvement in the terminal voltage response, whilst use of transient electrical power feedback offers a very oscillatory response similar to that obtained by the use of control B26, in Fig.6.53. Transient direct axis current feedback appears to offer no improvement in the terminal voltage response, and an improvement in speed response very similar to that obtained by use of control B27.

Inspection of Tables 6.16 and 6.17 shows that, with the exception of control B2, all the optimum gains supplying rotor acceleration to the excitation system are of the same order of magnitude and range from 2.47 to 3.60. Similarly, with the exception of control B22, the optimum gains supplying rotor acceleration to the governing system are of the same order of magnitude and range from 5.71 to 7.15. In all cases, the use of rotor acceleration feedback has improved the rotor speed error response. However, no consistent improvement in the terminal voltage response can be observed. The most dramatic improvement in terminal voltage response by the use of acceleration feedback, is achieved for Load 3 by use of control B22, the red response of Fig.6.56. This improvement was obtained with A1 set to 3


Terminal Voltage for 1s impact of Load 1 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 1 (b)
Vf / p.u.
4
3
2
1
0.00
-1
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 1 (c)
Werr / p.u.
0.02
0.00
-0.02
-0.04
-0.06
-0.08
0.00
1
2
3
4
5
Time / sec

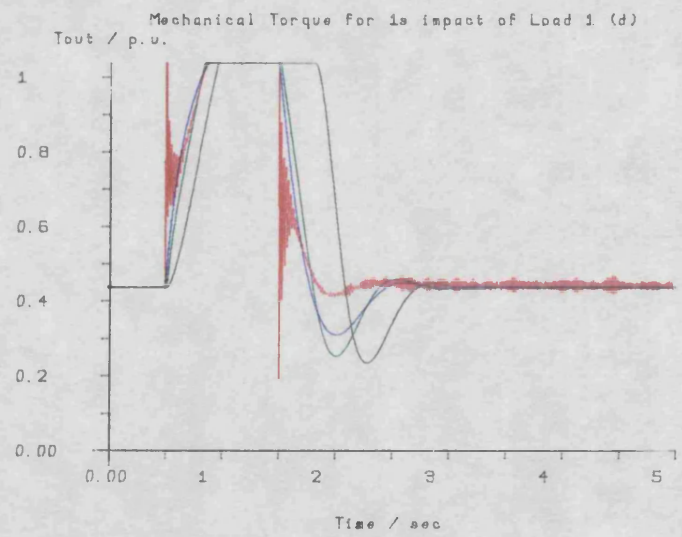
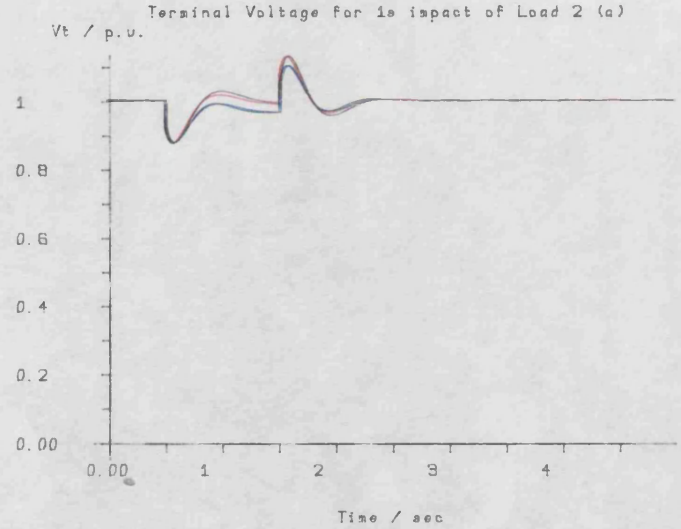
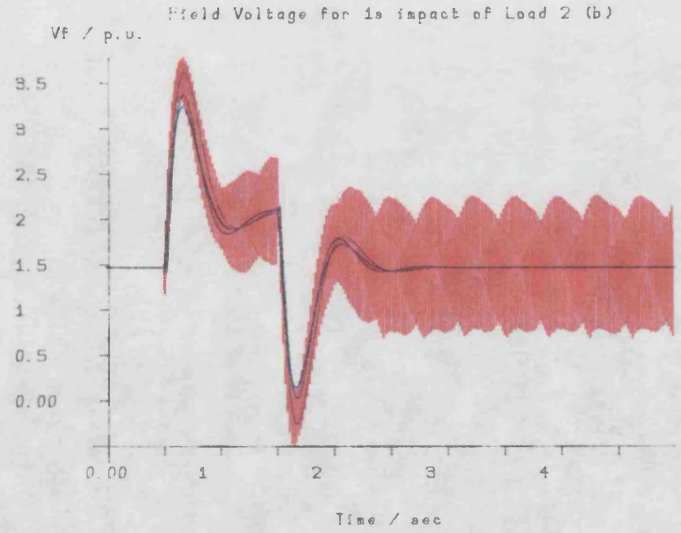
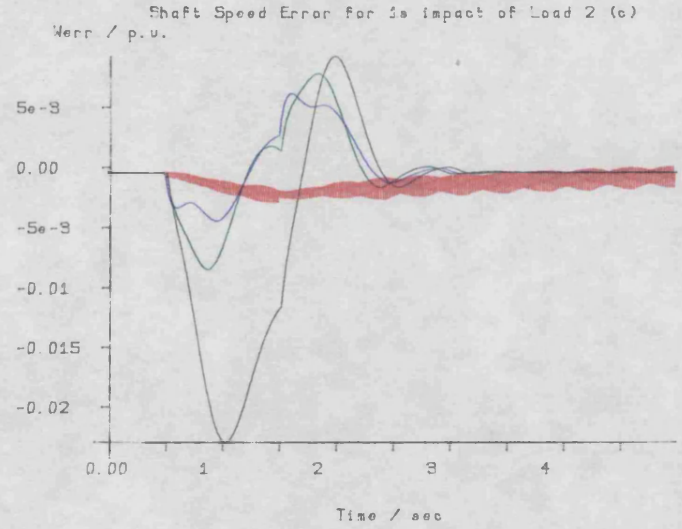

Mechanical Torque for 1s impact of Load 1 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.54


Terminal Voltage for 1s impact of Load 2 (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec


Field Voltage for 1s impact of Load 2 (b)
Vf / p.u.
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
Time / sec


Shaft Speed Error for 1s impact of Load 2 (c)
Werr / p.u.
5e-3
0.00
-5e-3
-0.01
-0.015
-0.02
0.00
1
2
3
4
Time / sec

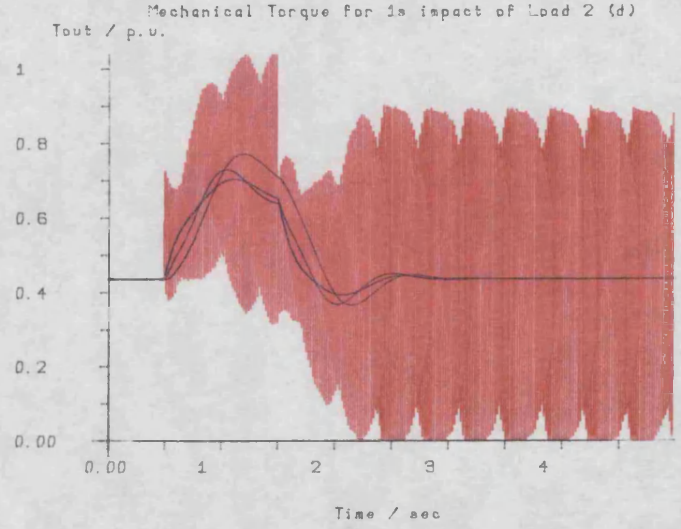
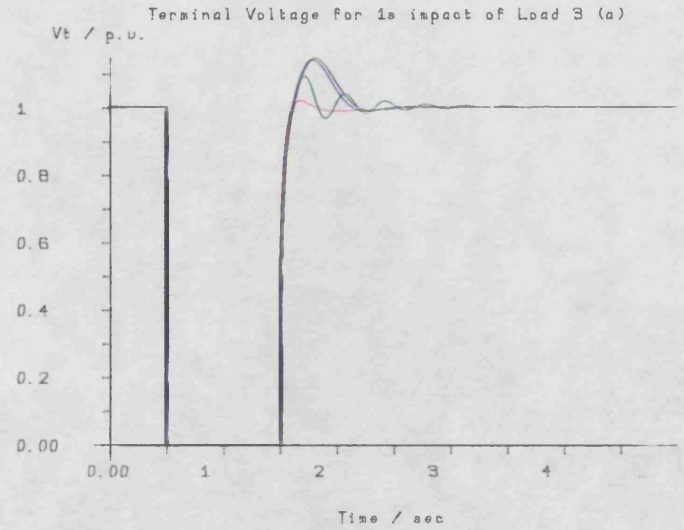
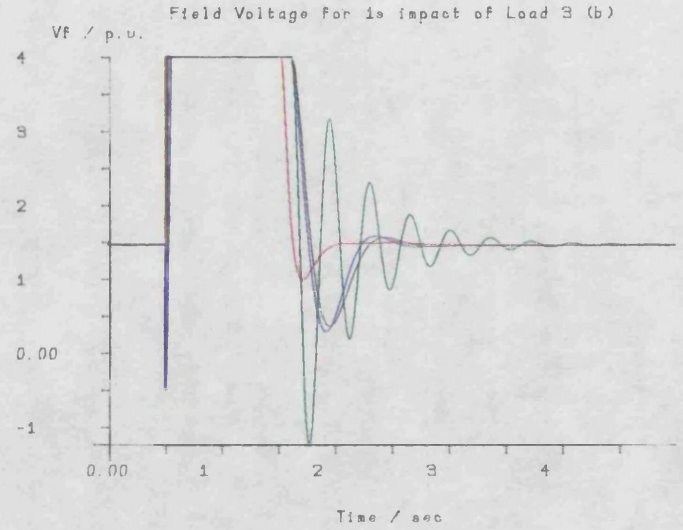
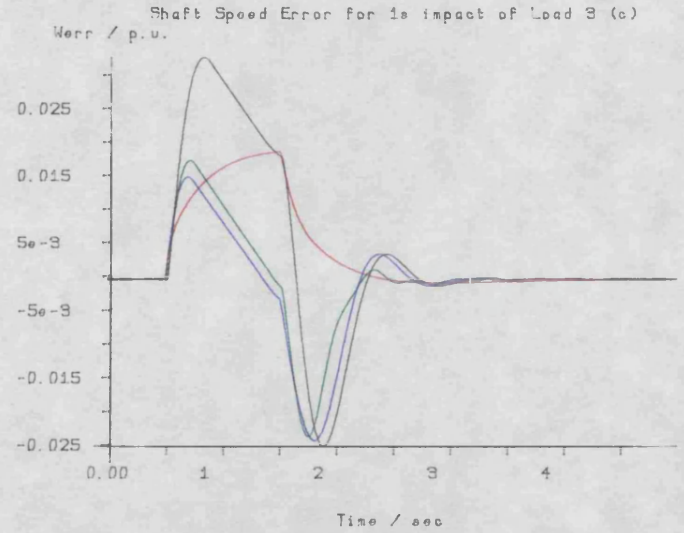

Mechanical Torque for 1s impact of Load 2 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec

Fig 6.55


Terminal Voltage for 1s impact of Load 3 (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec


Field Voltage for 1s impact of Load 3 (b)
Vf / p.u.
4
3
2
1
0.00
-1
0.00
1
2
3
4
Time / sec


Shaft Speed Error for 1s impact of Load 3 (c)
Werr / p.u.
0.025
0.015
5e-3
-5e-3
-0.015
-0.025
0.00
1
2
3
4
Time / sec

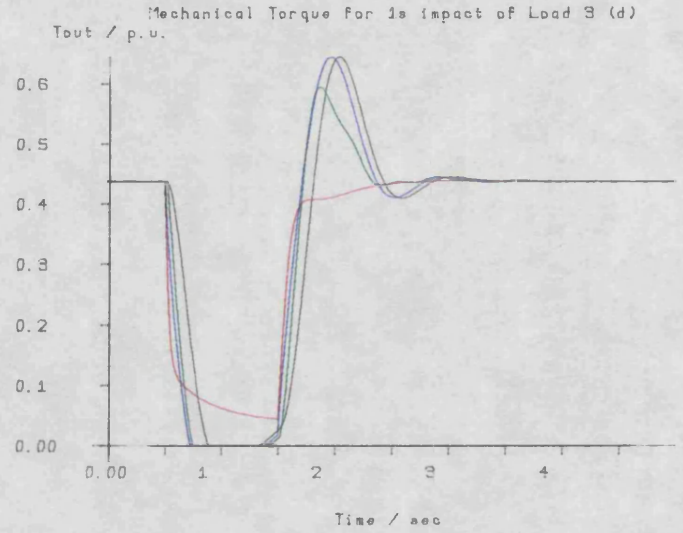

Mechanical Torque for 1s impact of Load 3 (d)
Tout / p.u.
0.6
0.5
0.4
0.3
0.2
0.1
0.00
0.00
1
2
3
4
Time / sec

Fig 6.56

and hence had a low penalty on the speed deviation. The trade off between terminal voltage response and speed error response is clearly shown by a comparison of the red responses of Figs.6.56 and 6.53.

The optimum gains obtained by the use of transient electrical power feedback for controls B3 and B10 are close together in parameter space as are those for controls B6 and B13. These controls correspond to optimisations conducted using Loads 1 and 2 with the different weightings on speed error in the performance index. It can also be seen that, when Load 3 is used, the magnitude of the optimum gain supplying transient electrical power to the excitation system increases by an order of magnitude, while the magnitude of the optimum gain supplying the governor loop decreases slightly. The gains for controls B26 and B23 are close together, particularly those feeding the governor loop. It would appear that for Loads 1 and 2 the optimum gains supplying transient electrical power to the excitation and governing systems are very similar and depend on the relative weighting of the terminal voltage error and speed deviation terms in the performance index. When Load 3 is used, these optimum gains are significantly different from those used with Loads 1 and 2, but they appear to be less sensitive to the change in performance index weighting.

The optimum gains supplying transient direct axis current to the excitation and governing sytems when using controls B4 and B7 show little variation with the change in load disturbance from Load 1 to Load 2. A similar observation may be made with regard to controls B7 and B14. However, a much larger change in the optimum gain for the governor system is observed. When the load disturbance is supplied

by Load 3, controls B27 and B24, the sign of the gains supplying the governor system change and they take on a small positive value. Although the gains supplying the excitation system in these two cases, B27 and B24, are very different, the transient responses, plotted in Figs.6.53 and 6.56 in blue, are very similar and show no improvement in the terminal voltage response over the base case.

The 'noisy' nature of the field voltage and prime mover torque responses obtained when rotor acceleration is used as an extra feedback signal has already been noted, particularly with regard to the responses of Fig.6.55. Two strategies have been investigated in an attempt to reduce the very large high frequency component of these responses. The first strategy involves a change in the performance index, and the second requires some alterations to the prime mover model.

If some penalty can be incorporated into the performance index which will penalise the high frequency components, then the action of the optimisation algorithm will tend towards solutions that reduce the amplitude of these components. The presence of the large amplitude high frequency components in the prime mover torque response has led to the introduction of the term $A2.\Delta tout^2$ into the performance index. Should the torque response 'burst' into a large amplitude oscillation, then the area under the square of the torque deviation will be very large. Thus a small value of A2 should be sufficient to prevent such oscillations occurring, without radically altering those optimum transient responses which do not generate this mode of oscillation. The optimisations for Loads 1 and 2 were repeated with the weighting coefficient A2 given the value 0.1.

Table 6.18 contains the resulting optimum gains, and Figs.6.57 to 6.58 show the corresponding transient responses as indicated in the table.

Initial Conditions as Table 6.16

Control	Signal	Load	Kavr	Kgov	P.I.	Fig.
B1	None	1	0	0	0.0947	6.57
B15	$P^2\delta$	1	2.60	3.44	0.0530	
B17	ΔPe	1	-0.315	-0.0421	0.0441	
B18	Δid	1	-0.154	-0.0244	0.0437	
B8	None	2	0	0	0.0165	6.58
B16	$P^2\delta$	2	11.77	4.61	0.0114	
B19	ΔPe	2	-0.318	-0.0420	0.0101	
B20	Δid	2	-0.113	-0.0262	0.00997	

Impact Time = 1s Vfmax = 4 p.u. Vfmin = -4 p.u. A1 = .3

A2 = 0.1 A3 = 1

Table 6.18

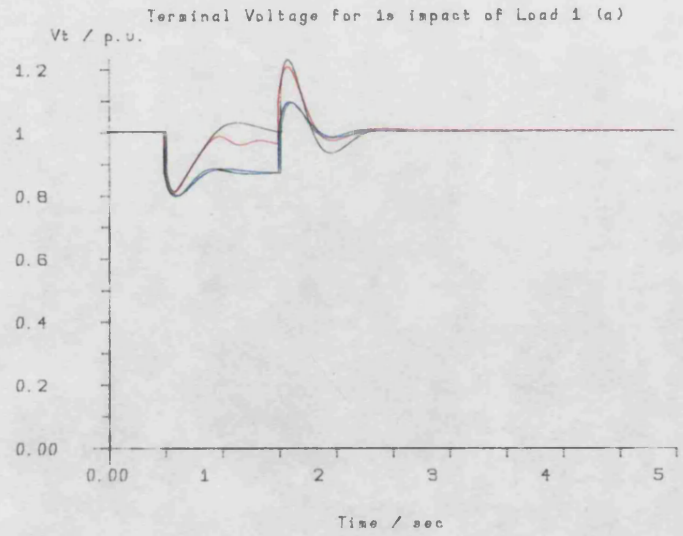
The optimum gains and transient responses for controls B17, B18, B19 and B20 are very similar to those for controls B3, B4, B10 and B11, shown in Table 6.16 and Figs.6.51 to 6.52. The inclusion of the $\Delta tout^2$ term has kept the prime mover torque out of limit and given rise to responses similar to those obtained with a high weighting in the speed deviation term. The responses of real interest are those for controls B15 and B16, shown in red in Figs.6.57 and 6.58

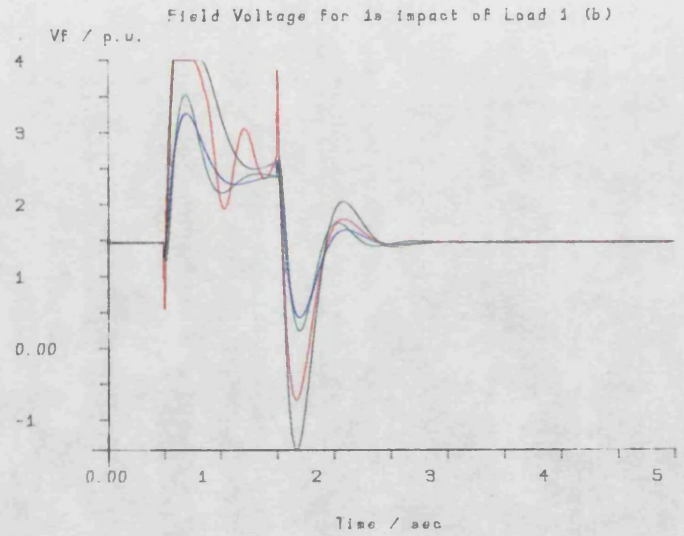
respectively. The responses for control B15 are almost identical to those for control B5, with a much lower amplitude high frequency content in the prime mover torque response. The gains supplying the excitation system with rotor acceleration are very similar for these two controls, whereas the gain supplying the governing system for control B15 is much lower than that for control B5. The high frequency content can therefore be associated with the larger gain of control B5, supplying rotor acceleration to the governing system.

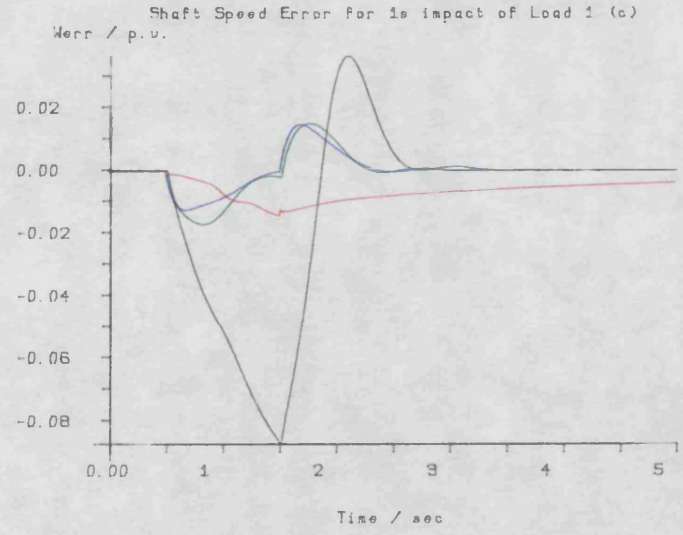
The speed error response for control B16, shown in red on Fig.6.58, follows a trajectory which lies within the envelope of the speed error response for control B12, also shown in red on Fig.6.55. The gain supplying the governing loop with rotor acceleration when using control B16 is significantly smaller in magnitude than the corresponding gain when using control B12. This difference is responsible for the large reduction in the high frequency content of the prime mover torque, speed error and field voltage responses. The terminal voltage response for control B16 is slightly flatter than that for control B12. This is due to the large gain used by control B16 to feed rotor acceleration into the excitation system, which causes the excitation system to bring about a reduction in the level that the field voltage reaches when the impact load is applied.

The second strategy for reducing the high frequency content in the prime mover torque, field voltage and speed error responses forms the basis of the remaining optimisation studies presented in this section.

The cause of the large amplitude, high frequency component


Terminal Voltage for 1s impact of Load 1 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 1 (b)
Vf / p.u.
4
3
2
1
0.00
-1
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 1 (c)
Werr / p.u.
0.02
0.00
-0.02
-0.04
-0.06
-0.08
0.00
1
2
3
4
5
Time / sec

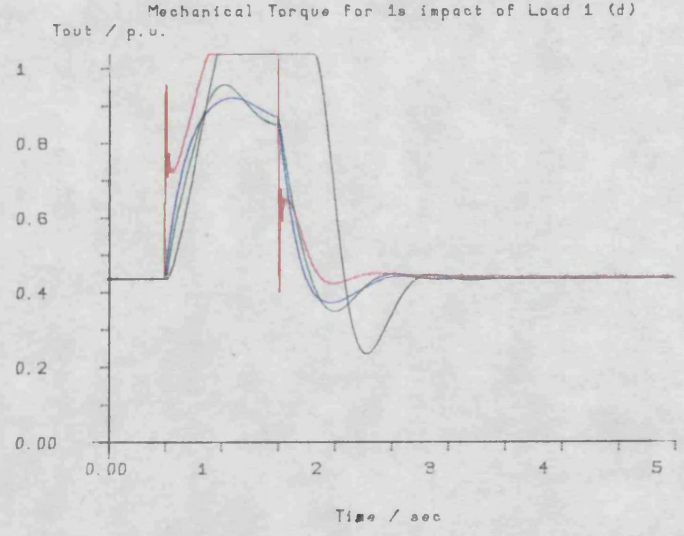
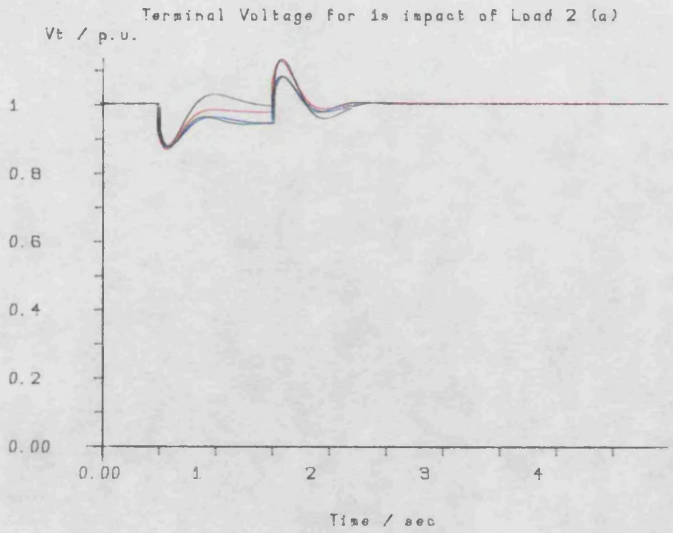
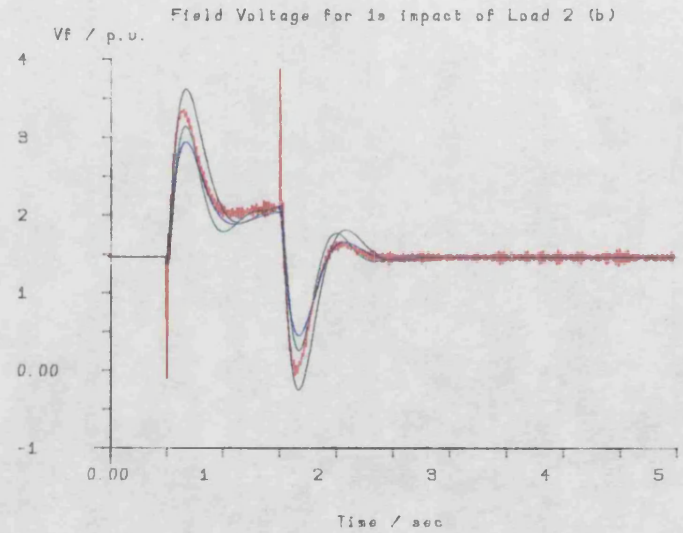
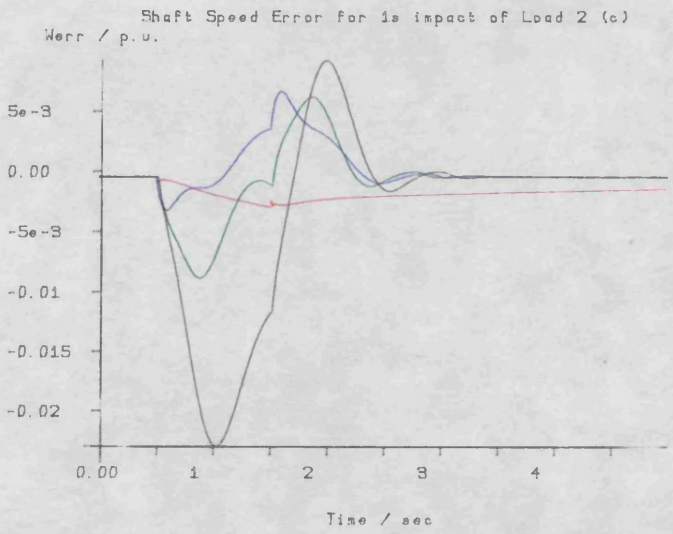

Mechanical Torque for 1s impact of Load 1 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.57


Terminal Voltage for 1s impact of Load 2 (a)
Vt / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
Time / sec


Field Voltage for 1s impact of Load 2 (b)
Vf / p.u.
4
3
2
1
0.00
-1
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 2 (c)
Werr / p.u.
5e-3
0.00
-5e-3
-0.01
-0.015
-0.02
0.00
1
2
3
4
Time / sec

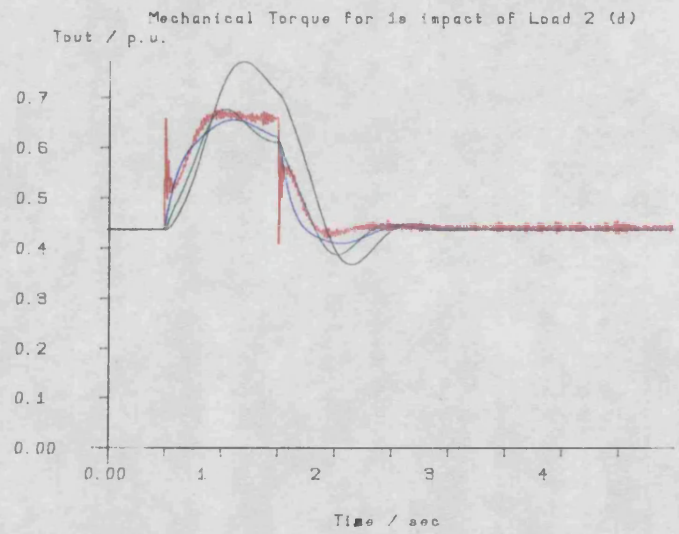

Mechanical Torque for 1s impact of Load 2 (d)
Tout / p.u.
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.00
0.00
1
2
3
4
Time / sec

Fig 6.58

present in the prime mover output torque has already been identified as the large amplitude signal forcing the input to the fuel rack. The question that arises is whether this signal would, in fact, reach such a high magnitude ($\pm$15 p.u. was observed), or would some real physical limit in the system prevent it?

To examine this the representation of the fuel rack and diesel engine must be reconsidered. In Chapter 3, it was considered that the output of the fuel rack is fuel rack position and that, assuming a nominally constant speed, the diesel engine output torque is directly proportional to the fuel rack position, delayed by the transport lag of the fuel system. This delay has not been modelled. The fuel rack has real position limits which restrict the extremes of the movement and these are translated into the maximum and minimum output torque limits of the diesel engine. The input to the fuel rack was considered to be some demand on the fuel rack position, and the first order lag with time constant T_C was assumed to adequately represent the dynamics of the movement of the fuel rack in response to this demand. After discussions with engineers from the Controls Division of Vosper Thornycroft Ltd. the significance attached to the input and output of the fuel rack was revised. It is considered that the output of the fuel rack is fuel flow and that the torque output by a diesel engine running at a nominally constant speed is proportional to this fuel flow, delayed by the transport lag of the fuel system. The input to the fuel rack becomes fuel rack position and it is at this point that the fuel rack position limits should be applied. It was also suggested that it would be appropriate to incorporate a rate limit on the fuel rack position such that the transition from fully open to fully closed, or vice versa, takes in the order of 0.2 seconds. With

these changes incorporated, the difference in the governing system in the area of the fuel rack is illustrated in Fig.6.59.

Optimisation studies were then performed on the system with the new fuel rack arrangement, for each of the three extra feedback signals with each of the three loads. The opening and closing rates for the fuel rack were set at ±5 p.u. per second. The resulting optimum gains are given in Table 6.19 and the corresponding transient responses are shown in Figs.6.60 to 6.62.

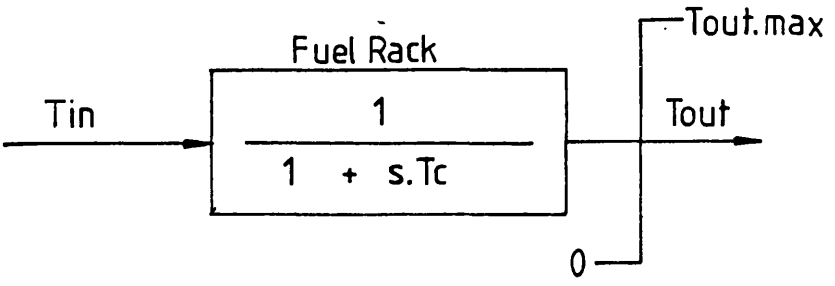
Initial Conditions as Table 6.16

Control	Signal	Load	Kavr	Kgov	P.I.	Fig.
D24	None	1	0	0	6.60	6.60
D25	$P^2\delta$	1	0.849	0.128	3.90	
D26	ΔPe	1	-0.250	-0.0116	1.41	
D27	Δid	1	-0.0740	-0.303	1.16	
D28	None	2	0	0	0.573	6.61
D29	$P^2\delta$	2	0.252	0.0420	0.536	
D30	ΔPe	2	-0.225	-0.0235	0.337	
D31	Δid	2	-0.0895	-0.0123	0.341	
D32	None	3	0	0	53.8	6.62
D33	$P^2\delta$	3	1.31	0.0976	53.0	
D34	ΔPe	3	-2.45	-1.06	52.8	
D35	Δid	3	-0.0821	0.000547	53.7	

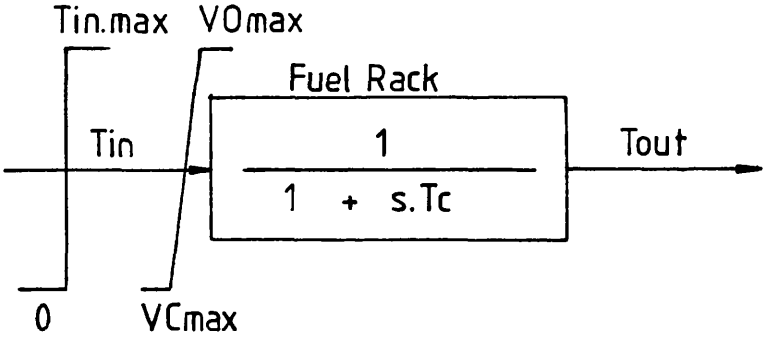
Impact Time = 1s Vfmax = 4 Vfmin = 0 V0 = 5 p.u.s.$^{-1}$

VC = -5 p.u.s.$^{-1}$ A1 = 10 A2 = 0 A3 = 100

Table 6.19



a)Before Modification



b) After Modification

Fig 6.59 Fuel Rack Arrangement

The weighting coefficients used in the performance index to obtain the results of Table 6.19 are somewhat different to those used in the previous studies. In obtaining the results of Table 6.16, the value of A1 was set to 100 giving an arbitrarily large penalty to rotor speed deviations. The value of 3, assigned to A1 to obtain the results of Table 6.17, was chosen so that, with a conventionally controlled system, i.e. using no extra feedback signals, the contributions made to the performance index by the terminal voltage error term and the speed deviation term are approximately equal, when the disturbance of Load 1 is applied with A3 set to zero. The value of 10 assigned to A1 in this case was chosen on this basis. A3 has been set at 100 for two reasons. First, to increase the numerical value of the performance index and secondly, to reduce the proportion of the contribution made to the performance index by terms which remain uncontrollably constant while the impacted load is applied (in particular, the large contribution made to the performance index by the terminal voltage error when the short circuit of Load 3 is applied). The lower ceiling of the excitation system has been set to zero for the optimisations of Table 6.19, so field voltage reversal is no longer possible.

The base case responses of Figs.6.60 to 6.62, plotted in black, are almost identical to those of Figs.6.51 to 6.53. The only noticeable differences are in the field voltage responses, which are clipped when field voltage reversal is attempted, and in the speed error and prime mover torque responses for control D24, Fig.6.60, as compared with those obtained for control B1, Fig.6.51. The application of the impacted load of Load 1 drives the prime mover into its torque limit as before, except that, in this case, the torque

ceiling is approached 'gently' since the limit is applied on the input side of the fuel rack. Consequently, the torque response takes longer to saturate, and the speed deviation becomes larger (approaching 11%) prior to the removal of the impacted load. When the impacted load is removed, the prime mover torque comes out of limit 'gently', and there is no overshoot in the speed error response.

Fig.6.60 contains the transient responses for controls D24 to D27. Use of the three controls, D25 to D27, can be seen to improve the speed error response. These improvements are not as large as those obtained for the earlier controls, shown in Figs.6.51 and 6.54. The position of the limit in front of the fuel rack prevents large excursions of the signal driving the fuel rack from causing rapid changes in the prime mover torque. In particular, the improvement obtained by use of rotor acceleration feedback, control D25 plotted in red, is much less than that obtained by the use of controls B2 or B5, shown in Figs.6.51 and 6.54, respectively. Moving the position of the limits associated with the fuel rack has, as intended, removed the large amplitude high frequency component from the field voltage and torque responses, but there is now a significant low frequency ripple in the latter part of the speed error response which the acceleration signal carries over into the field voltage and prime mover torque responses. The terminal voltage response corresponding to control D25 is worse than that for the base case, control D24. The acceleration signal hinders the excitation system reaching its positive ceiling when the load is impacted, giving rise to the larger undershoot of the corresponding terminal voltage response. Likewise, the acceleration signal stops the excitation system from reaching its negative ceiling when the impact load is removed, giving rise to the

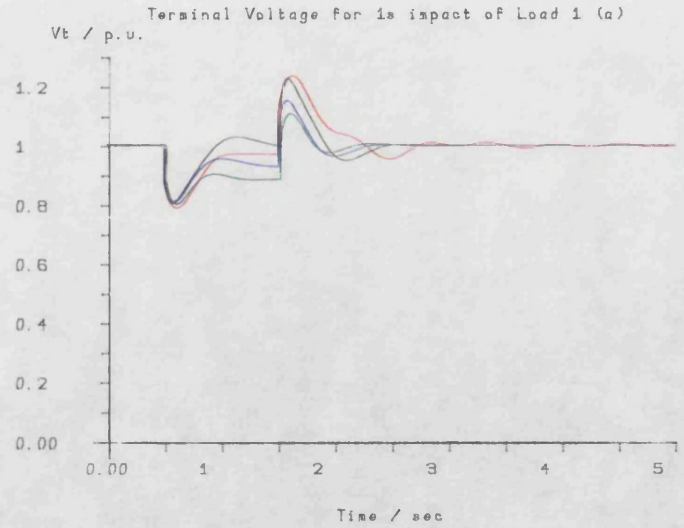
larger terminal voltage overshoot.

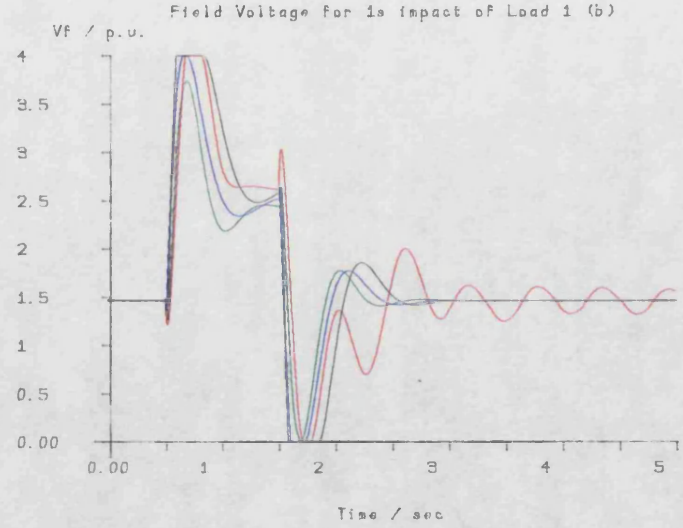
The responses for controls D26 and D27, plotted in Fig.6.60 in green and blue respectively, are similar in form to those obtained for controls B3 and B4, shown in Fig.6.51, or controls B6 and B7, shown in Fig.6.54. In each case, the terminal voltage response has a large error while the impacted load remains applied, and the main difference is in the way prime mover torque saturation is approached. The 'gentle' approaches, shown in Fig.6.60, for controls D26 and D27, result in larger amplitude speed error responses than those for the corresponding controls in Figs.6.51 and 6.54.

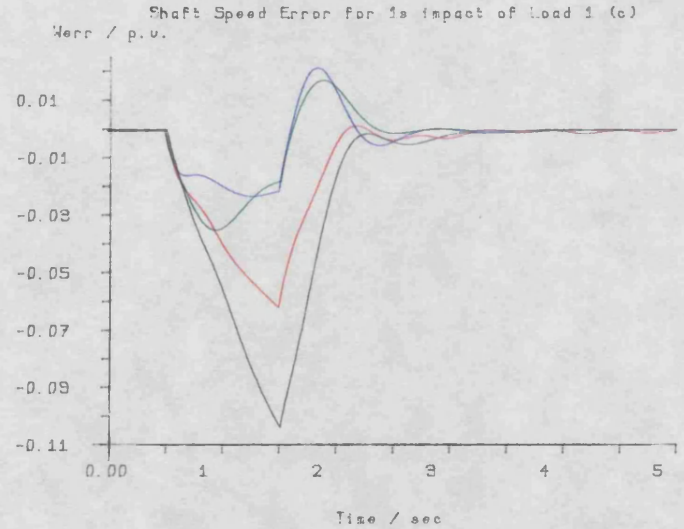
Fig.6.61 shows the transient responses obtained when controls D28 to D31 are used. These controls are optimised for the disturbance caused by a one-second application of the impact load of Load 2. The transient responses show virtually no difference between the response obtained by the use of rotor acceleration feedback, control D29 in red, and that obtained when no extra feedback is used, control D28 in black. There is a small reduction in the amplitude of the rotor speed error response, and a small reduction in the value of the terminal voltage overshoot following the removal of the impacted load. The terminal voltage response for control D29 is also slightly more oscillatory than that for control D28. Controls D30 and D31 offer a larger improvement in the rotor speed deviation arising when the impact load is applied, but very little change in the deviation following its removal. The corresponding terminal voltage responses settle with a significant terminal voltage error while the impact load is applied, and have smaller overshoots than the base case when the impact load is removed.

The transient responses for the controls D32-D34, optimised for the short circuit disturbance of Load 3, are shown in Fig.6.62. The use of rotor acceleration feedback, control D33, fails to produce any improvement in the speed error response, plotted in red, when the short circuit is applied. On the removal of the short circuit, control D33 greatly reduces the size of the speed error undershoot, and greatly improves the terminal voltage recovery. Use of transient electrical power feedback, control D34, produces the characteristically oscillatory terminal voltage and field voltage responses, similar to those obtained by use of controls B26 and B23, shown in Figs. 6.53 and 6.56, respectively. Prior to the removal of the short circuit, the speed error response for control D34 is similar in shape to those for controls B26 and B23. Following its removal, the speed error response is more oscillatory and the acceleration signal carries these oscillations through to the torque response. The low optimal gains obtained for control D35 produce transient responses which are indistinguishable from those of the base case, control D32.

The optimum gains supplying rotor acceleration to the excitation and governing systems when using controls D25, D29 and D33, are considerably lower than the corresponding gains used with controls B2, B9 and B25 (Table 6.16) and controls B5, B12 and B22 (Table 6.17). Apart from this very general observation, there is no trend in these gains which can be picked out. The optimum gains supplying transient electrical power to both control loops when using controls D26, D30 and D34, show a much better correlation with the corresponding gains used with controls B3, B10 and B26 (Table 6.16) and controls B6, B13 and B23 (Table 6.17). The only optimum gain that is orders of


Terminal Voltage for 1s impact of Load 1 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 1 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 1 (c)
Werr / p.u.
0.01
-0.01
-0.03
-0.05
-0.07
-0.09
-0.11
0.00
1
2
3
4
5
Time / sec

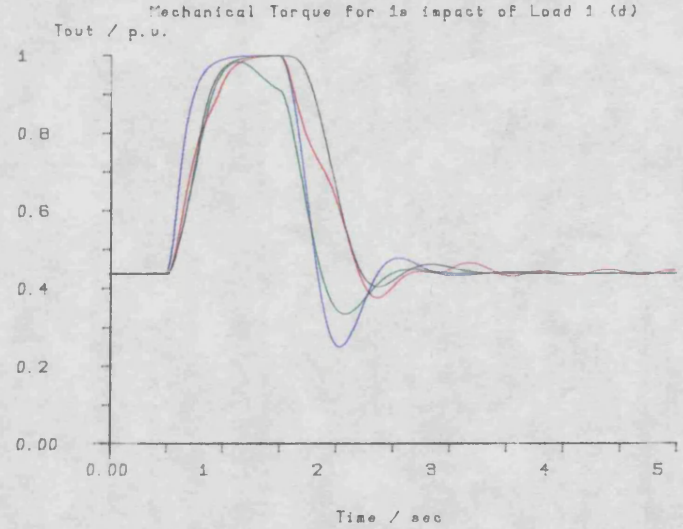
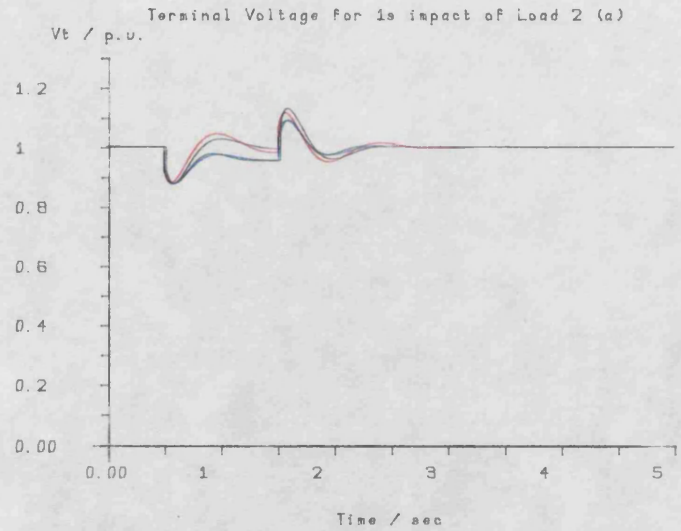
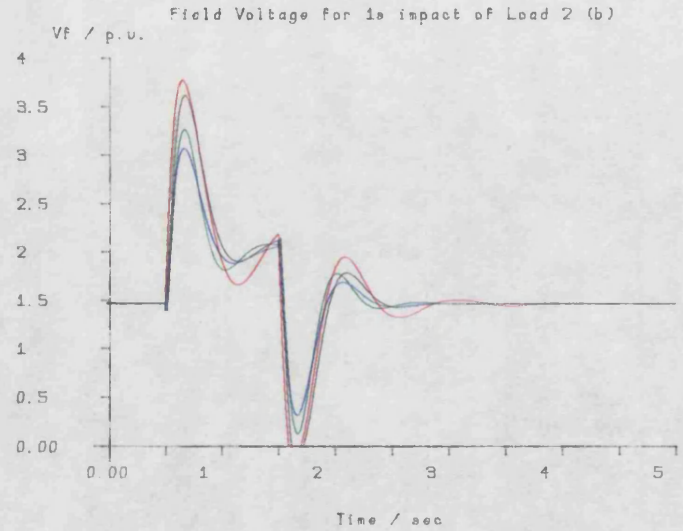
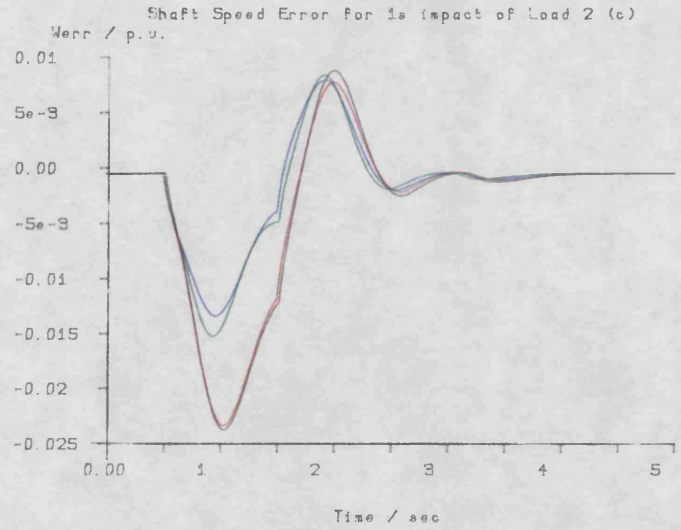

Mechanical Torque for 1s impact of Load 1 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.60


Terminal Voltage for 1s impact of Load 2 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 2 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 2 (c)
Werr / p.u.
0.01
5e-3
0.00
-5e-3
-0.01
-0.015
-0.02
-0.025
0.00
1
2
3
4
5
Time / sec

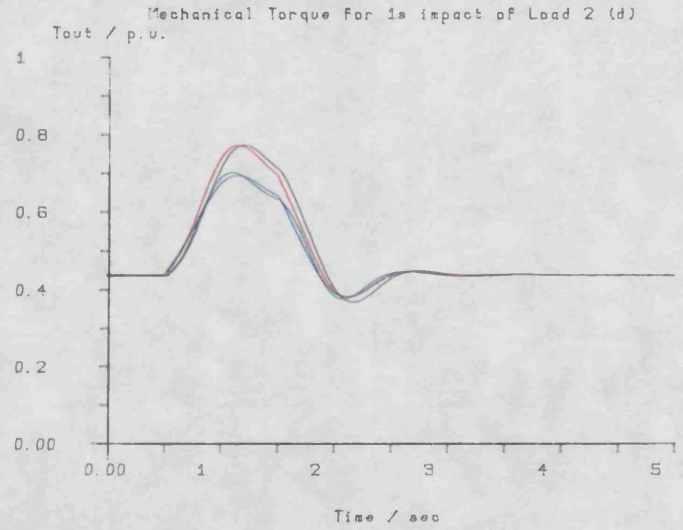
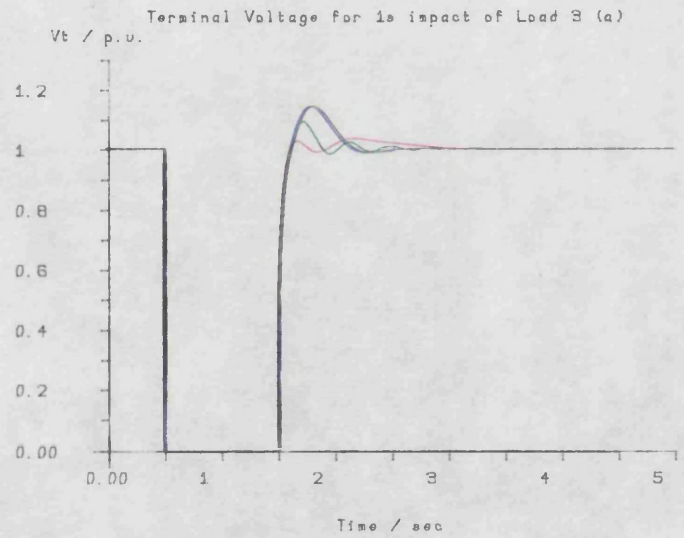
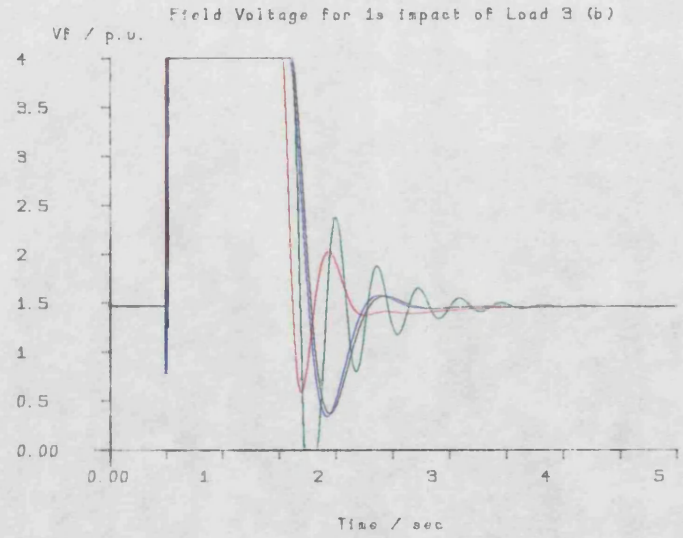
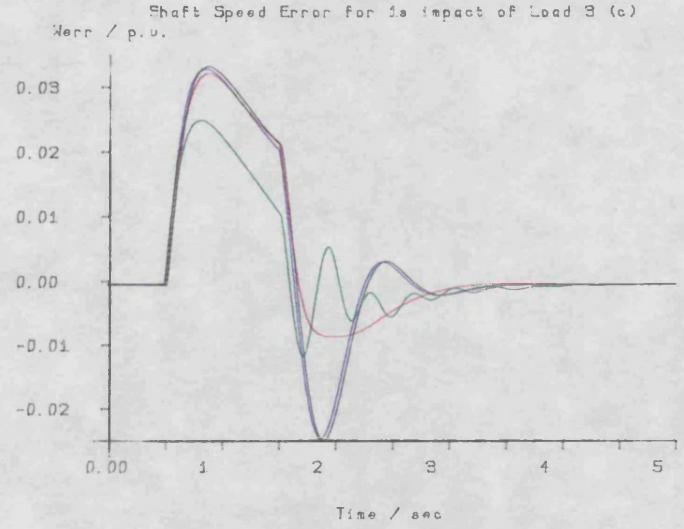

Mechanical Torque for 1s impact of Load 2 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.61


Terminal Voltage for 1s impact of Load 3 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 1s impact of Load 3 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 1s impact of Load 3 (c)
Werr / p.u.
0.03
0.02
0.01
0.00
-0.01
-0.02
0.00
1
2
3
4
5
Time / sec

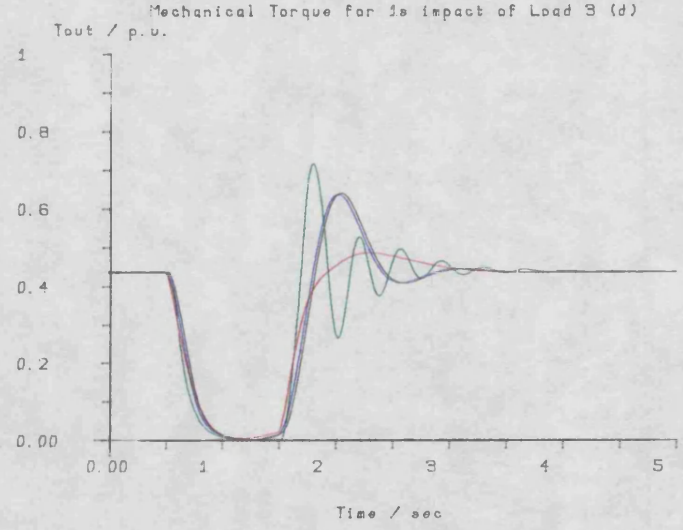

Mechanical Torque for 1s impact of Load 3 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.62

magnitude different is the gain used by control D34 to supply the governing system, -1.06, as compared with -0.0451 and -0.0445, used by controls B26 and B23, respectively. The optimum gains supplying the control loops with transient direct axis current when using controls D27, D31 and D35 can, to some extent, be compared favourably with the corresponding gains used by controls B4, B11 and B27, (Table 6.16) and controls B7, B14 and B24 (Table 6.17). The gains supplying the excitation system for all these controls lie in the range -0.0610 to -0.140, with the exception of control B24, for which the gain takes the value -0.232. The gains supplying the governor system with transient direct axis current used by controls D27, D31 and D35 are orders of magnitude different from the corresponding gains in Tables 6.16 and 6.17. The most consistent observation that can be made with respect to Tables 6.16 to 6.19 is that the signs of corresponding optimum gains are the same.

Table 6.20 contains the results of optimisations performed with the impact load applied for a shorter duration. The transient responses obtained by use of the controls in table 6.20 are shown in Figs.6.63 to 6.45, as indicated.

Initial Conditions as Table 6.16

Control	Signal	Load	Kavr	Kgov	P.I.	Fig.
D36	None	1	0	0	1.57	6.63
D37	$P^2\delta$	1	0.410	0.0103	1.28	
D38	ΔPe	1	-0.346	-0.00246	0.539	
D39	Δid	1	-0.130	-0.0159	0.601	
D40	None	2	0	0	0.430	6.64
D41	$P^2\delta$	2	0.456	0.0467	0.337	
D42	ΔPe	2	-0.477	-0.00991	0.143	
D43	Δid	2	-0.161	-0.00891	0.179	
D44	None	3	0	0	14.30	6.65
D45	$P^2\delta$	3	1.06	0.102	13.87	
D46	ΔPe	3	-3.44	-0.196	13.87	
D47	Δid	3	-0.0392	-0.00523	14.27	

Impact Time = 500ms Vfmax = 4 p.u. Vfmin = -4 p.u. VO = 5 p.u./s

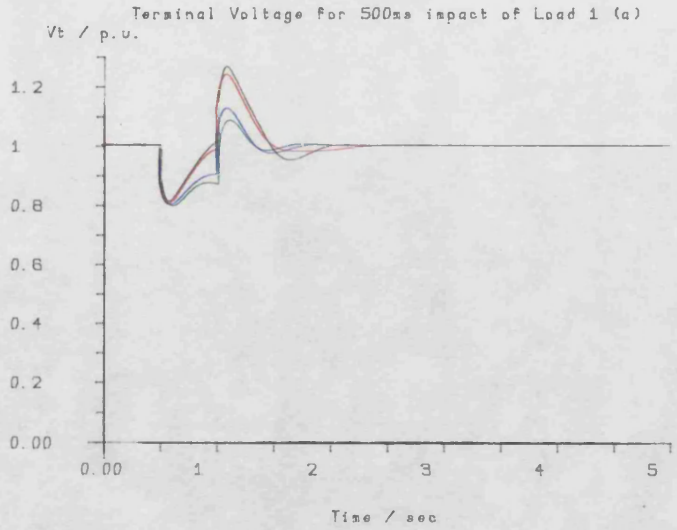
VC = -5 p.u./s A1 = 10 A2 = 0 A3 = 100

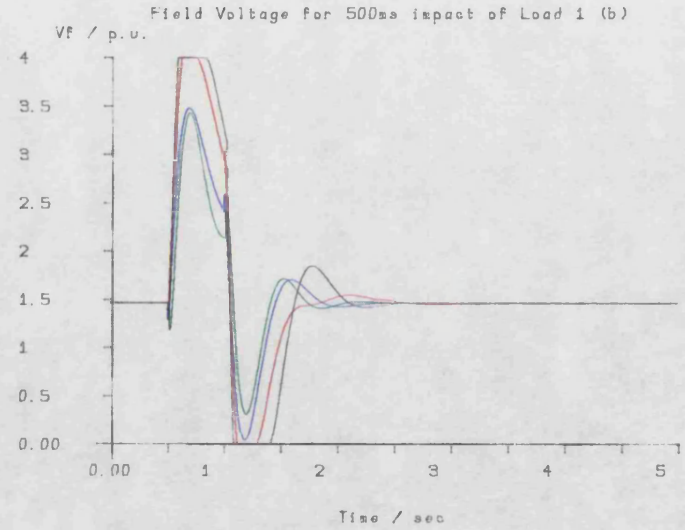
Table 6.20

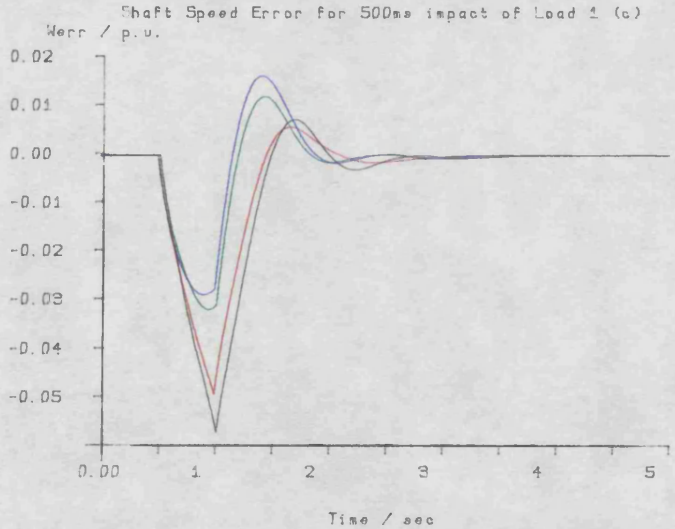
The transient responses of Figs.6.63 to 6.65 will not be discussed in detail. In general, all these figures show some improvement in the speed error response obtained by the use of extra feedback applied to the three different load disturbances. The amplitude of the speed error responses in Fig.6.63 is smaller than the amplitude of those in Fig.6.60, both figures showing transient responses with the Load 1 disturbance. Since the impact load is applied for a shorter period of time, the speed deviation caused by overloading the generator is smaller, and the torque responses do not reach their limits. The changes in the speed error responses of

Figs.6.64 and 6.65, as compared with the base case response, are similar to the changes seen in the speed error responses of Figs. 6.11 and 6.62. All the terminal voltage responses follow the already established trends, particularly those obtained by the use of rotor acceleration or transient electrical power feedback with disturbances created by application of the short circuit of Load 3.

A comparison of the optimum gain values given in Tables 6.20 and 6.19 shows significant differences between the values of corresponding gains. The optimum gains used by controls D45 and D46 compare favourably with those used by controls D33 and D34, although the difference in the gains supplying transient electrical power to the governor, between controls D46 and D34, is very large. Controls D47 and D35 both use very small magnitude gains to supply transient direct axis current to both the excitation and governing systems. Consequently, these controls have almost identical transient responses to the corresponding base cases, controls D44 and D32. It is notable that control D47 shows the only instance of a negative gain supplying transient direct axis current to the governing system when the disturbance of Load 3 is applied. No clear trend can be seen in the change in gains brought about by a change in the load disturbance from Load 1 to Load 2. The gains supplying the excitation system with transient electrical power or transient direct axis current appear to be fairly constant across the change in disturbance. The corresponding gains which supply the governing system with these two signals appear subject to large variations. In Table 6.20 there is little change in the optimum gains supplying the excitation system with rotor acceleration, between controls D37 and D41, whereas in Table 6.19 the corresponding change from control D25 to control D29


Terminal Voltage for 500ms impact of Load 1 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 500ms impact of Load 1 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 500ms impact of Load 1 (c)
Werr / p.u.
0.02
0.01
0.00
-0.01
-0.02
-0.03
-0.04
-0.05
0.00
1
2
3
4
5
Time / sec

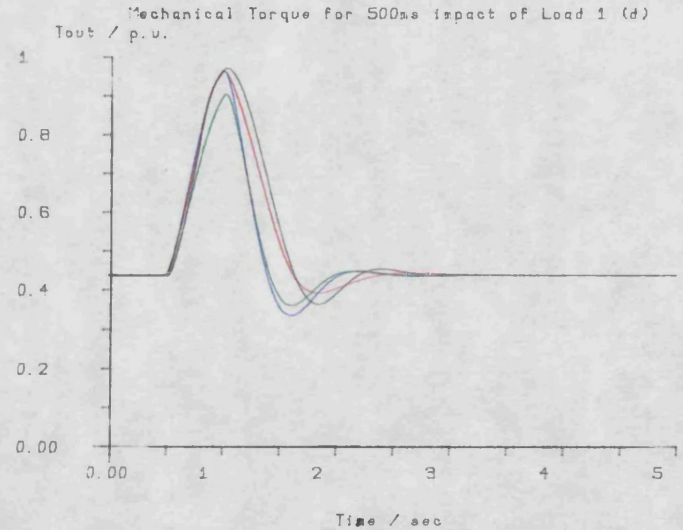
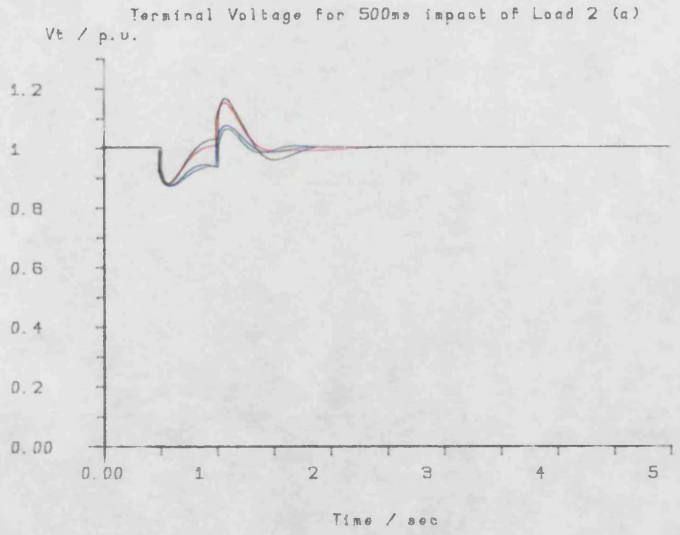
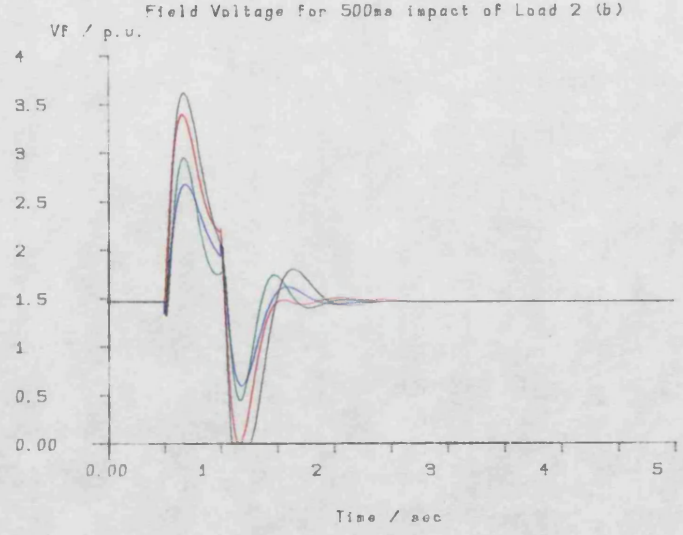
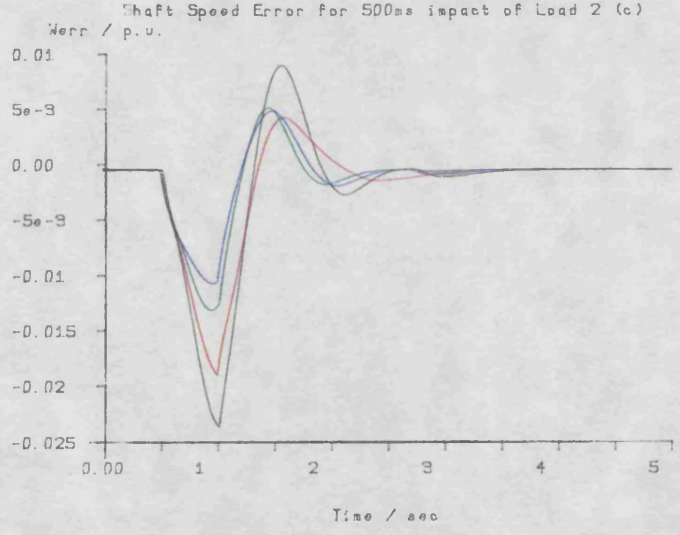

Mechanical Torque for 500ms impact of Load 1 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.63


Terminal Voltage for 500ms impact of Load 2 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 500ms impact of Load 2 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 500ms impact of Load 2 (c)
Werr / p.u.
0.01
5e-3
0.00
-5e-3
-0.01
-0.015
-0.02
-0.025
0.00
1
2
3
4
5
Time / sec

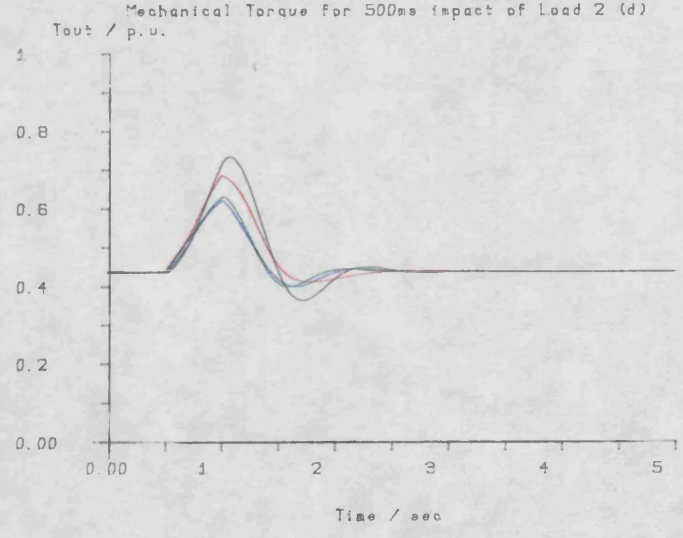
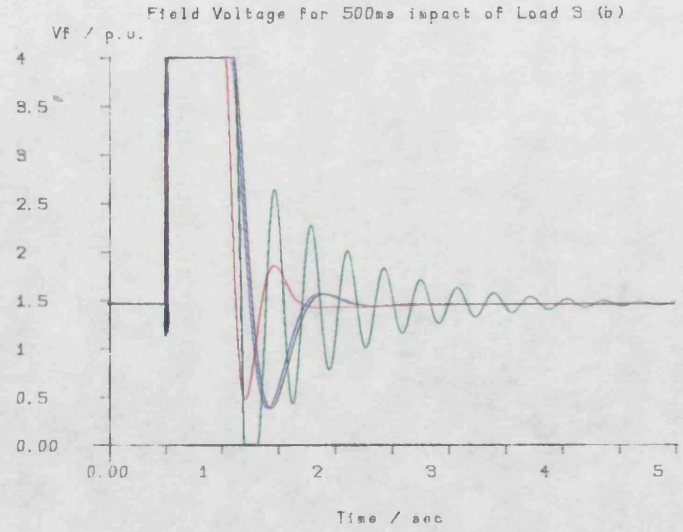
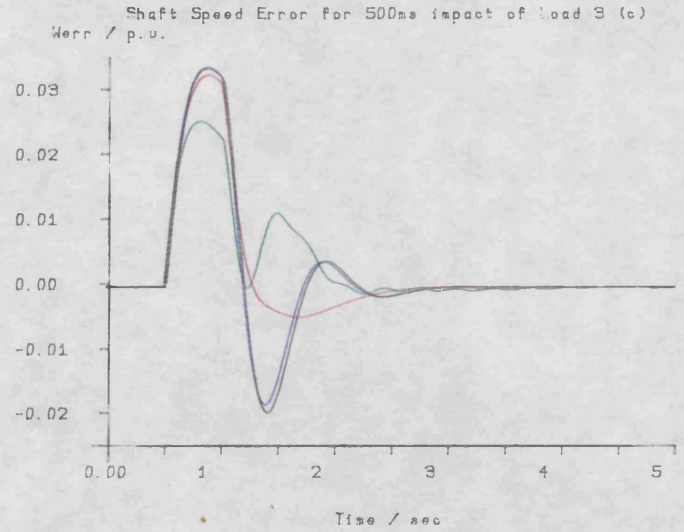

Mechanical Torque for 500ms impact of Load 2 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.64


Terminal Voltage for 500ms impact of Load 3 (a)
Vt / p.u.
1.2
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec


Field Voltage for 500ms impact of Load 3 (b)
Vf / p.u.
4
3.5
3
2.5
2
1.5
1
0.5
0.00
0.00
1
2
3
4
5
Time / sec


Shaft Speed Error for 500ms impact of Load 3 (c)
Werr / p.u.
0.03
0.02
0.01
0.00
-0.01
-0.02
0.00
1
2
3
4
5
Time / sec

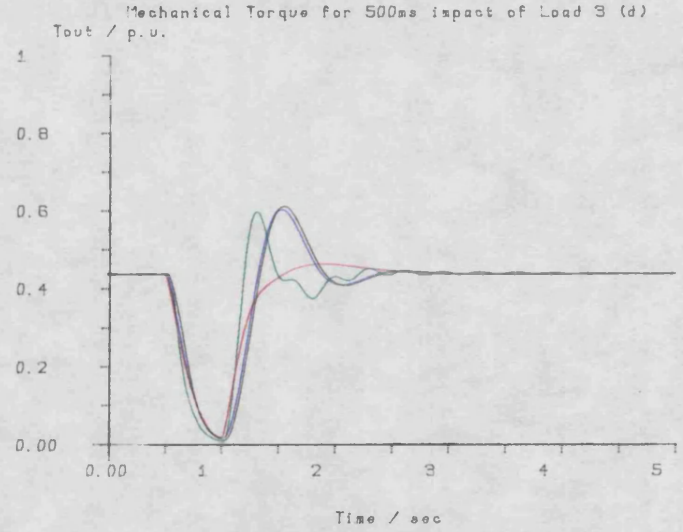

Mechanical Torque for 500ms impact of Load 3 (d)
Tout / p.u.
1
0.8
0.6
0.4
0.2
0.00
0.00
1
2
3
4
5
Time / sec

Fig 6.65

results in a change in optimum gain of about 3 to 1. Similarly, although the optimum gains supplying rotor acceleration to the governing system when using controls D41 and D29 are very close together, the differences between these values and those used by controls D37 and D25 indicate no identifiable trend.

In concluding this section, it must be remarked that no universally useful set of optimum gains was obtained for any of the extra feedback signals. In general, the use of either rotor acceleration or transient electrical power feedback is capable of improving the rotor speed response. Unfortunately, manufacturers are far more concerned with the maintenance of terminal voltage regulation than frequency control, and the only significant improvements in the terminal voltage response were obtained for the recovery from the short circuit applied by Load 3. This improvement requires the use of rotor acceleration feedback through gains which depend on which prime mover model is used. In the case of the original arrangement of the fuel rack, suitable gains for feeding rotor acceleration into the excitation and governing systems would be in the order of 3 and 0.4, respectively, while, in the case of the modified fuel rack arrangement, suitable gains would be in the order of 1 and 0.1, respectively.

6.4 Conclusions

The optimisation studies performed in sections 6.3.2 and 6.3.3 of this Chapter show that the use of simple extra feedback signals allows significant improvements to be made in the transient performance of a synchronous generator connected via a transmission

line to an infinite busbar when it is subjected to a symmetrical three phase short circuit fault close to the generator terminals. Further, it was shown that where it is possible to significantly reduce the prime mover output torque before fault clearance occurs, as is the case in the studies conducted in section 6.3.2, it is possible to improve the transient stability of a generator subjected to this type of fault, as judged on a first swing basis. These improvements can be made by feeding back either rotor acceleration or transient electrical power into both excitation and governing systems. In addition to improving the transient performance of the plant, it has been shown that use of transient electrical power feedback can significantly improve the dynamic or small change stability of the system.

The optimisation studies conducted on the finite busbar system in section 6.3.4 have proved inconclusive except for short circuit type disturbances. In general, improvements in rotor speed response are possible with the use of each of the extra feedback signals considered. For the short circuit load change, it is felt that significant improvements can be made in the terminal voltage recovery when the short circuit is removed, by the use of rotor acceleration feedback. For finite load changes, which cause a temporary dip in the terminal voltage, the action of a conventional excitation system is capable of restoring the terminal voltage and the use of extra feedback signals is of little benefit.

The single dimensional optimisations conducted in section 6.3.3. took, typically, 4.5 minutes to converge, requiring 18 performance index evaluations, and 8 movements of the simplex. The

two-dimensional optimisations performed throughout this Chapter take, typically, 10 minutes to converge, with 50 performance index evaluations and 25 movements of the simplex. These timings vary with the initial position and size of the simplex.

References

6-1 Humpage, W.D., Smith, J.R. and Rogers, G.J.: "Application of dynamic optimisation to synchronous generator excitation control", Proc.IEE, Vol.120, 1973, pp87-93.

6-2 Lu, H., Hazell, P.A. and Daniels, A.R.: "Co-ordinate single variable excitation control and governing of turboalternators", Proc.IEE, Vol.129, Pt.C, No.6, 1982, pp278-284.

6-3 Dineley, J.L. and Mikhail, S.E.: "Effect of feedback signals in excitation and governor/turbine systems on the transient and dynamic stability", IEEE PES Meeting Paper A76, 113-1, Jan. 1976.

6-4 Lee, Y.B.: "Sensitivity and optimal control studies of power systems", PhD Thesis, 1975, University of Bath.

6-5 Lu, H.: "Optimisation studies of a single machine power system", PhD Thesis, 1979, University of Bath.

6-6 Hestenes, M.R. and Stiefel, E.: "Methods of conjugate

gradients for solving linear systems", Journal of Research National Bureau of Standards, Vol.49, No.6, pp409-436, December 1952.

6-7 Gill, P.E. and Murray, W.: "Quasi-Newton methods for unconstrained optimisation", J.Inst.Maths.Applic, Vol.9, 1972, pp91-108.

6-8 Broyden, C.G.: "Quasi-Newton methods and their application to function minimisation", Maths.Comput., 1967, Vol.21, pp368-381.

6-9 Hartley, H.O.: "The modified Gauss Newton method for fitting non-linear regression functions by least squares", Techmetrics, Vol.3, No.2, pp269-280, May 1961.

6-10 Rosenbrock, H.H.: "An automatic method for finding the greatest and least value of a function", The Computer Journal, 1960, Vol.3, pp175-184.

6-11 Nelder, J.A. and Mead, R.: "A simple method for function minimisation", The Computer Journal, 1965, Vol.7, pp308-313.

6-12 Powell, M.J.D.: "An iterative method for finding stationary value of a function of several variables", The Computer Journal, Vol.5, p147, 1962.

6-13 Fletcher, R and Powell, M.J.D.: "A rapidly convergent descent method for minimisation", The Computer Journal, 1963, Vol.6, p163.

6-14 Powell, M.J.D.: "An efficient method for finding the minimum of a function of several variables without calculating derivatives", The Computer Journal, 1964, Vol.7, p155.

CHAPTER 7

CONCLUSIONS

The work presented in this thesis falls into four main areas: the real-time simulator; power system simulation; parameter optimisation; and the supporting stability studies.

7.1 The Real-Time Simulator

The hardware and software of the real-time simulator were described in Chapter 4. The simulator has clearly achieved its main objective, which is the simulation, in real-time, of the transient behaviour of power systems. The use of the Tripos operating system has proved to be a suitable environment for the development and testing of the simulator software. The high level language support offered by Tripos has greatly simplified the development of simulator applications such as the parameter optimisation and stability studies.

The simulator is configured so that two processing subsystems perform all the calculations involved in integrating the plant equations, and the remaining subsystem manages the collection and interpretation of the data generated by the simulator. In this configuration the Monitor task provides a core of functions which are applicable to real-time simulation in general and a number of functions which are specific to the particular simulation being undertaken (in this instance the simulation of power systems). The

functions which manage the data collection, data storage and data presentation (either in the form of tables or as graphical data) are of a general nature, while the functions which interpret the simulation data, such as the functions which perform the stability studies, are only applicable to power system simulation. However, due to the modular nature of the Monitor task, new functions may easily be added or redundant functions may easily be removed. Thus, the Monitor task presented in this thesis may conveniently form the basis for a similar task used to monitor and control the simulation of an entirely different plant. This flexibility has been brought about by the partitioning of the data generation and the data capturing functions into separate tasks.

Real-time simulation generates a considerable amount of data, approximately 10Kbytes every second in this case, so it is clearly impractical to store all the data generated by the simulator in disc files. Applications such as the optimisation of the transient response of a power system, or studies of its stability, make use of data generated by the simulator to influence the next disturbance and/or controls applied to the plant. When a conventional power system simulator, generally in the form of an absentee job run on a mainframe computer, is used to perform applications of this type, most of the simulation data is never presented to the user. The volume of data generated by the application makes its storage impractical, whilst the rate at which the data is generated means that an interactive session with the simulator is not sensible. The speed at which data is generated when a real-time simulator is used to implement these applications means that valuable information, in the form of continuous graphical output, may be given to the user. In an

optimisation study, for example, the user can very quickly assess whether a particular performance index is developing the required characteristic in the transient response. Using a conventional simulator, graphical data is often only made available for the base case and optimised transient responses. Only after the optimisation has completed would it be possible to judge whether the reduction in performance index, brought about by the optimisation process, results in a corresponding improvement in transient performance.

Together, the simulating tasks, Generator and Volt, comprise a little over 4Kbytes of assembly code. They integrate the ninth order set of differential equations and solve the set of algebraic equations that describe the power systems used in this thesis, every 5 milliseconds. A typical optimisation study, using a control interval of 5 seconds, requires of the order of 10 minutes to converge and makes in the order of 50 performance index evaluations. This time may be approximately halved if continuous graphical output is not required and varies with the size of the control interval, the initial control parameters, and the required accuracy. Few optimisation studies require more than 15 minutes to converge and several have required less than 10 minutes.

Typical transient stability limit studies locate the critical post-fault impedance to within 0.001 p.u. and generally require between 20 and 30 minutes to generate a complete pre-fault/post-fault impedance curve. Similarly, the dynamic stability limit studies converge when power system stability is lost following a disturbance to the excitation system reference of less than 0.004 p.u. and, typically, require four hours to generate a complete dynamic

stability limit curve. This type of study takes such a large duration of time to complete because transient response data is ill-suited to the location of the dynamic stability limit.

7.2 Power System Simulation

Implicit in the optimisation and stability studies presented in Chapters 5 and 6 has been the simulation of the transient behaviour of power systems. The mechanism by which the dynamic equations of the plant are transformed into the discrete time domain for the purposes of real-time simulation was described in Chapter 3.

The use of an implicit integration technique has transformed the dynamic plant equations into a set of algebraic equations. These equations, together with the algebraic plant equations, give the plant output at the end of the current time step, based on the plant input and output at the beginning of the current time step. As a result, the solution of the plant equations is not iterative, and so the time taken to step the simulation forward by one time step is fixed (except for slight variations due to the particular numerical values involved at each time step). Also, unlike Runge Kutta techniques, the equations need only be solved once per time step, so the Z transform method employed in this work readily lends itself to applications, such as real-time simulation, where it is desirable that the integration algorithm requires a fixed execution time.

The transient responses presented in Chapter 6, typically Fig. 6.9, were obtained using the real-time simulator and compare favourably with curves generated using Runge Kutta techniques. The Z

transform technique employed in the real-time simulator has therefore proved to be a numerically stable and efficient algorithm for integrating the dynamic plant equations using an integer based number representation.

Three different power systems have been simulated successfully using the Z transform method and typical transient responses for these systems may be found in the base case responses given in Chapter 6. The effect on the transient response of changes in controller parameters, such as excitation ceilings, and governor valve rate and position limits, as well as the effect of changing the steady-state operating point, may be readily demonstrated using the real-time simulator. Results from such studies may be presented rapidly on a high resolution colour graphic display.

7.3 The Stability Studies

Both transient and dynamic stability studies were undertaken and the results are presented in Chapters 5 and 6. The dynamic stability limit is represented by plotting the locus of operating points for which the power system is critically stable. In general, the stability limit locus lies in the leading power factor region of the operating chart, while operating in the lagging power factor region is limited, not by stability considerations, but by consideration of rotor losses and the maximum prime mover output power. It was shown in Chapter 5 that representative dynamic stability limit curves can be generated by repeatedly moving the generator operating point into the leading power factor region until stability is lost. The stability limit is then located by searching for the critical point

where only a small disturbance is necessary for stability to be lost.

Whereas the dynamic stability limit of a power system is defined in terms of small changes in operating point, or small disturbances, and is independent of the type of disturbance applied, the transient stability limit of a power system is defined in terms of the disturbance applied. Transient stability limits are not generally measured directly, but inferred by indirect measurements such as critical fault clearance times, or pre-fault/post-fault impedance plots. Chapter 5 demonstrated use of a real-time simulator in generating pre-fault/post-fault impedance curves, by repeatedly applying a particular three phase short circuit fault to the power system. A binary search algorithm is then used to locate the critical post-fault impedance for a fixed pre-fault impedance. It was shown that increased excitation ceilings, faster governor valves and faster fault clearance times, all improve the transient stability of a power system, and that quantative comparison between these different methods of improving transient stability can be made. The transient stability studies presented in Chapter 6 also show that transient stability can be significantly improved by the use of optimised controller gains.

7.4 Optimisation Studies

A simple parametric optimisation algorithm has been used successfully to optimise the gain values associated with the feedback of extra control signals into the excitation and governing systems of three different generating plant. The simplex algorithm used in these studies differs from the more generally used 'gradient methods'

in that it does not require the evaluation of either the first or second partial derivatives of the performance index, with respect to the parameters being optimised. This saves a great deal of computation since evaluation of these performance index 'sensitivity' values requires that the dynamic sensitivity equations of the plant are integrated in real-time alongside the dynamic plant equations. As there are far more dynamic sensitivity equations than there are dynamic plant equations, the use of a conventional gradient method would have required the solution of considerably more equations, which in turn would have required far more computational power to achieve real-time operation from the simulator. Since the simplex algorithm only requires the value of the performance index at particular points in the parameter space, it is easily implemented and changes in the form of the performance index do not require the significant recoding of the simulating tasks that would arise due to changes in the performance index sensitivity equations.

Like all 'hill climbing' algorithms, convergence to a unique minimum cannot be guaranteed, unless it is known that only one minimum point exists in the parameter space. The results presented in Chapter 6 show the performance improvements that may be obtained when optimised controller gains are used. It was concluded in Chapter 6 that the use of rotor acceleration as an extra feedback signal, applied to both the excitation and governing systems, offered the most significant performance improvements obtainable from a turbogenerator system, connected via a transmission line to an infinite busbar. In the case of a hydrogenerator, connected to an infinite busbar via a transmission line, similar transient performance improvements were obtainable by the use of rotor

acceleration or transient electrical power as the extra feedback signal, supplied to the excitation system alone. In both cases significant damping of the load angle oscillations was achieved but only in the case of the turbogenerator was it possible to significantly reduce the size of the first peak in the load angle response. This difference is due to the large difference in the speed of governor valve operation for the two systems. It is notable that in both these cases, the turbogenerator and the hydrogenerator, significant improvements in the dynamic stability limit were only obtained by the use of transient electrical power as the additional feedback signal.

In the case of a finite busbar system, consisting of a single diesel driven generator connected to an inductive load, consistent performance improvements were only obtained in the cases where a severe disturbance was applied to the plant. In these cases, the use of rotor acceleration as an extra feedback signal offered the most significant performance improvements.

7.5 Suggestions for Further Work

Since the simulator used for the work presented in this thesis was designed, the continuing improvement of integration technology has resulted in faster, more powerful, microprocessor and microprocessor support devices, being available. It is anticipated that a simulator based on one of the emerging 32 bit microprocessors and incorporating memory, implemented using the currently available 256K bit memory 'chips', on the same board, could execute simulation tasks of the type described in this thesis at rates in the order of

four times as fast as the simulator described in Chapter 4. This extra computational speed could be used for two purposes. First, more complex simulations could be undertaken by the same number of processing subsystems due the straightforward increase in instruction throughput. Secondly, some of the extra computation speed could be used to accommodate the overhead of using a high level language to code the simulating tasks. The benefit of using a high level language is that it enforces discipline in the way large and complicated software tasks are written which, in turn, makes that software more maintainable. The penalty of using a high level language is that, in general, compilers cannot extract the full performance from the underlying processor and it is generally possible to write a particular function in assembly language in such a way that it has a faster execution time.

Another advantage of the improvement in integration technology is that far more functionality can be designed into a single board. Thus it would now be possible to provide the equivalent of one of the processing subsystems described in Chapter 4, on a single printed circuit board. It would be appropriate in this case to consider a simulator which uses a shared memory architecture of the type shown in Fig. 4.1(b), so that the entire simulator could be housed in a single 19 inch rack. Interprocessor communication would then be conducted via the backplane in the rack, rather than by the use of a communications link, although a communication link could still be used in the event that a particular simulation requires more processing resources than can be accommodated in a single rack. The use of the communications link has complicated the maintenance of the simulator software. When certain modifications are required it is

necessary to make consistent updates to all the simulator source files. The use of a shared memory architecture would allow the processors participating in the simulation to access simulation data in a consistent manner, no matter which processor is responsible for updating the particular value accessed. Thus, the simulation software would not need to know which of the processors is expected to update a particular value and the fundamental processes involved in performing a simulation task, may be re-ordered without regard to which processor is destined to make use of the data that each process generates.

The extra computation power offered by the emerging microprocessor devices, together with the option to use a high level language to implement the simulation of a plant, makes the implementation of a real-time multimachine power system simulation of arbitrary size much more attractive than an implementation restricted to the use of assembly language for reasons of execution speed.

APPENDIX A

SYSTEM PARAMETERS

The following tables contain plant parameters for the turbogenerator, the hydrogenerator and the diesel generator, respectively.

Base MVA	2,352	
Direct Axis Synchronous Reactance, X_d	2.80	pu
Quadrature Axis Synchronous Reactance, X_q	2.72	pu
Direct Axis Transient Reactance, X'_d	0.362	pu
Direct Axis Subtransient Reactance, X''_d	0.230	pu
Quadrature Axis Subtransient Reactance, X''_q	0.220	pu
Direct Axis Transient Open Circuit Time Constant, T'_{do}	7.28	sec
Direct Axis Subtransient Open Circuit Time Constant, T''_{do}	0.028	sec
Quadrature Axis Subtransient Open Circuit Time Constant, T''_{qo}	0.116	sec
Inertia Constant, H	4.4	pu

Table A1 Normalised Turbogenerator Parameters

Base MVA	2,352	
Transformer Resistance, R_x	0.00490	pu
Transformer Reactance, X_x	0.157	pu

Table A2 Normalised Generator Transformer Parameters for the Turbogenerator

Base MVA	2,352	
Line Resistance, R_l	0.00794	pu
Line Reactance, X_l	0.109	pu

Table A3 Normalised Transmission Line Parameters for the Turbogenerator

Governor Valve Time Constant, T_a	0.02	sec
Prime Mover Time Constant, T_b	0.25	sec
Maximum Valve Opening Rate, VO_{max}	4.0	pu/sec
Maximum Valve Closing Rate, VC_{max}	6.6	pu/sec
Slip speed gain, K_t	0.0676	
Damper Constant, K_d	6.0	

Table A4 Prime Mover and Governor Parameters for the Turbogenerator

Forward Path Exciter Gain, K_1	200	
Stabiliser Gain, K_2	0.035	
Exciter Time Constant, T_e	0.50	sec
Stabiliser Time Constant, T_s	0.50	sec

Table A5 Excitation System Parameters for the Turbogenerator and the Hydrogenerator

Base MVA	1,980	
Direct Axis Synchronous Reactance, X_d	1.152	pu
Quadrature Axis Synchronous Reactance, X_q	0.643	pu
Direct Axis Transient Reactance, X'_d	0.216	pu
Direct Axis Subtransient Reactance, X''_d	0.143	pu
Quadrature Axis Subtransient Reactance X''_q	0.143	pu
Direct Axis Transient Open Circuit Time Constant T'_{do}	13.86	sec
Direct Axis Subtransient Open Circuit Time Constant T''_{do}	0.057	sec
Quadrature Axis Subtransient Time Constant T''_{qo}	0.057	sec
Inertia Constant H	4.5	pu

Table A6 Normalised Hydrogenerator Parameters

Base MVA	1,980	
Transformer Resistance R_x	0.0	pu
Transformer Reactance X_x	0.155	pu

Table A7 Normalised Generator Transformer Parameters for the Hydrogenerator

Base MVA	1,980	
Line Resistance R_l	0.0	pu
Line Reactance X_l	0.225	pu

Table A8 Normalised Line Transmission Line Parameters for the Hydrogenerator

Governor Valve Time Constant T_a	0.02	sec
Prime Mover Time Constant T_b	0.25	sec
Maximum Valve Opeingin Rate VO_{max}	12.0	pu/sec
Maximum Valve Closing Rate VC_{max}	12.0	pu/sec
Slip Speed Gain K_t	0.0676	
Damper Constant K_d	6.0	

Table A9 Prime Mover and Governor Parameters for the Hydrogenerator

Direct Axis Synchronous Reactance X_d	1.36	pu
Quadrature Axis Synchronous Reactance X_q	0.93	pu
Direct Axis Transient Reactance X'_d	0.34	pu
Direct Axis Subtransient Reactance X''_d	0.20	pu
Quadrature Axis Subtransient Reactance X''_q	0.19	pu
Direct Axis Transient Open Circuit Time Constant T'_{do}	2.05	sec
Direct Axis Subtransient Open Circuit Time Constant T''_{do}	0.03	sec
Quadrature Axis Subtransient Open Circuit Time Constant T''_{qo}	0.08	sec
Inertia Constant J	2.00	pu

Table A10 Normalised Diesel Generator Parameters

Governor Gain K_g	21.33	
Lead/Lag Time Constant T_a	0.44	sec
Lead/Lag Time Constant T_b	0.68	sec
Fuel Rack Time Constant T_c	0.11	sec
Maximum Output torque T_{outmax}	1.039	pu

Table A11 Prime Mover and Governor Parameters for the Diesel Generator

Exciter Gain K_1	294
Stabiliser Gain K_2	0.015
Exciter Time Constant T_e	0.760 sec
Stabiliser Time Constant T_s	0.278 sec

Table A12 Excitation System Parameters for the Diesel Generator

APPENDIX B

FILE FORMATS

All Tripos files may be considered as an array of longword or BCPL values which may be read into memory using the BCPL library routine, 'readwords'. Record formats may then be embedded into this 'array' so as to define either loadable binary files, ASCII text files, or a file with a user defined format.

The loader software described in Chapter 4 is capable of loading a task image into the two simulating subsystems via the communication link. The image so loaded must be stored on the subsystem running Tripos as a binary file containing only absolute records (or hunks). An absolute hunk contains data which must be loaded into the target system at a specific location in memory. This type of record is generated by the MC68000 assembler when a program is assembled at an absolute address, or by the SYSLINK utility which generates a bootable copy of the operating system. The format of a Tripos absolute hunk is shown in Fig. B1.

Thus, each absolute hunk consists of: a header word, 't.abshunk', which has the decimal value 1003 and identifies the record type as an absolute hunk; a load address, which is a BCPL pointer to the address where the data is to be loaded; a word count which specifies the size of the following data field; and the data field which contains the loadable data. The loadable image file may then be made up of several absolute hunks, concatenated one after the

other.

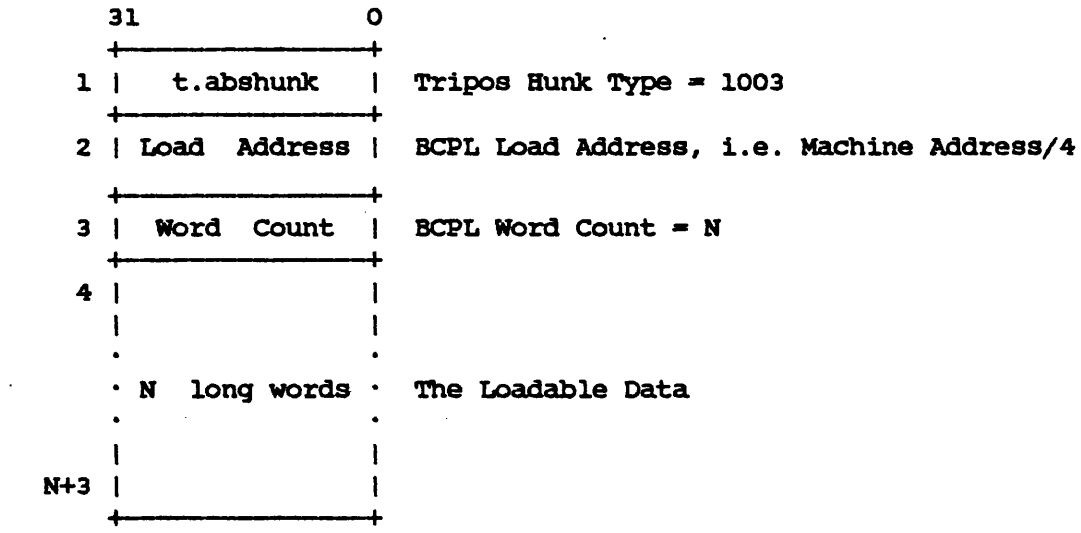


Fig. B1 Tripos Absolute Hunk Record Format

Two file formats have been defined for the storage of plottable data. The first type of plot file consists of a single record in the format shown in Fig. B2.

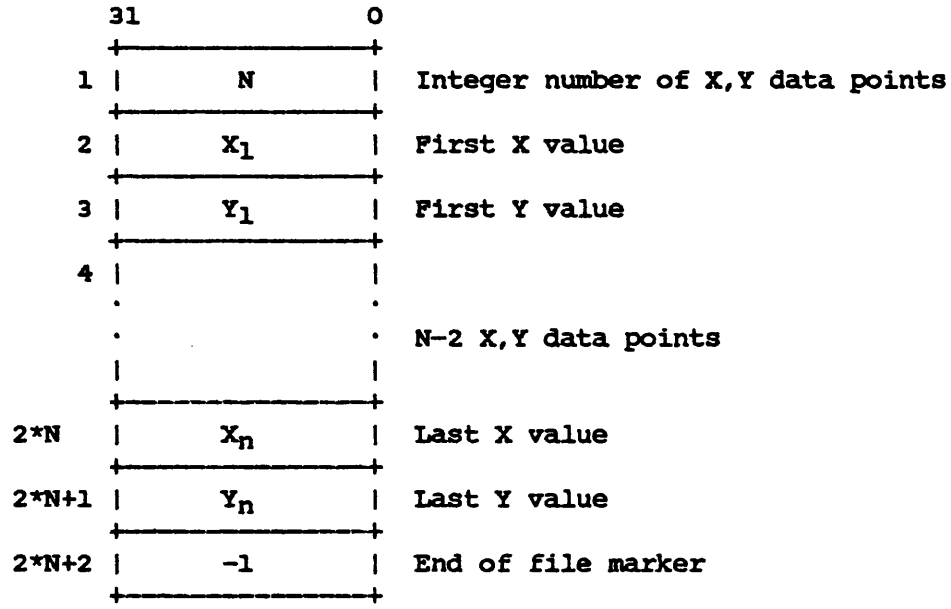


Fig. B2 Long format Plot File Record

This format is used to store arbitrary X,Y co-ordinates in

sequence, and consists of: an integer, N, which specifies the number of X,Y co-ordinate pairs stored in the file; then a data field consisting of that number of X,Y pairs stored as IEEE single precision floating point values; and the integer value, -1, to mark the end of the file.

When transient data generated by the simulator is to be stored, the X co-ordinate values represent the passage of time. Since the data samples are spaced at regular intervals in time, it is possible to almost halve the file space required to store a single transient response by adopting the second plot file record format which is shown in Fig. B3.

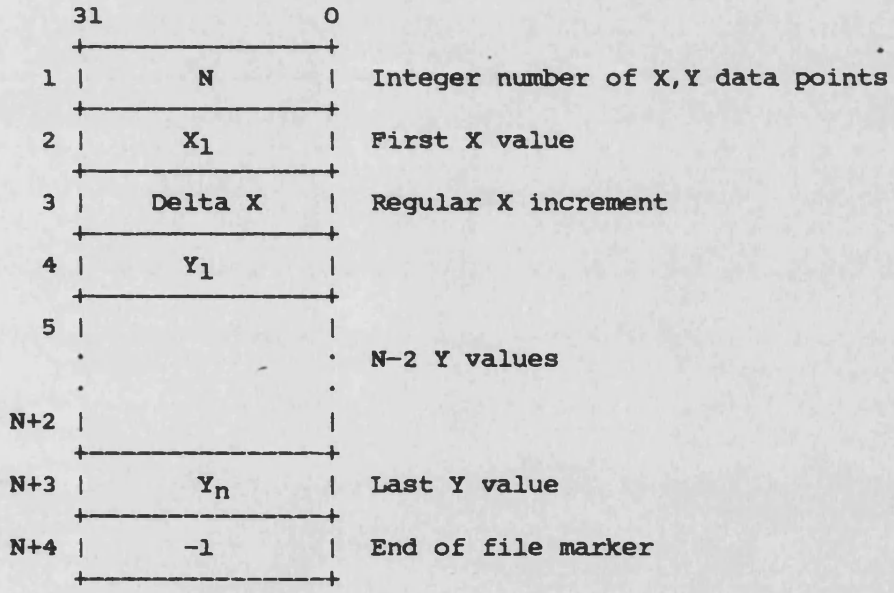


Fig. B3 Short Format Plot File Record

This second format is similar to the format shown in Fig. B2, the number of co-ordinates and the end of file marker both being integers, while the remaining fields contain single precision IEEE floating point values. In both cases, the end of file marker may be

used as a consistency check. If the value after the last Y value is not -1 or the length of the file is neither 2*N+2 or N+4 entries, then it is unlikely that the file contains plottable data.

The following Appendix is a copy of a CEGB memo, referenced in Chapter 5, which is reproduced with permission.

APPENDIX C

METHOD OF COMPARING ALTERNATIVE TECHNIQUES FOR IMPROVING TRANSIENT STABILITY OF TURBO-GENERATORS

Many papers have been published, and techniques proposed, showing how the transient stability of turbo-generators can be improved, usually for a generator operating through a switchable impedance to an infinite busbar. Comparison of these techniques is often difficult due to the limitations of the test or calculation techniques causing some parameters to be kept constant, e.g. fault clearance time, type of fault, generator power factor, system impedance or pre-fault/post-fault impedance ratio. Among the data variables are the parameters of the generator AVR and governor, and these components can be modelled with varying degrees of accuracy.

Suggested ways of judging the measure of improvement have included increased fault clearance time, increased generator power output or reactive absorption from a fixed system, increased net energy available to slow the generator and hold synchronism (Ref. 1), or the allowable increase in system impedance for a fixed generator (Ref. 2). One paper (Ref. 3) has presented results in a more general way and this has been extended to form the attached graph.

The system studied consists of generator, step up transformer to the H.V. busbar which is the fault location, and reactance to an infinite busbar, the reactance generally being larger after fault clearance. Whilst system reactance is taken as the measure of the tie between busbars, the system is represented by RXB values, the

relationship between these depending on the design voltage of the system and whether it is overhead line or cable. Voltages are per unit on design voltages, and MVA is p.u. on generator rating. A 'basic case' is taken for the generator and its control systems, which can include current practice, e.g. 660 MW generator with magnetic amplifier voltage regulator, Watt governor with reheat turbine without instantaneous interceptor governing, subject to a 3-phase fault on a 400kV overhead line system, the fault being cleared in 140ms. This basic case is studied for a range of values of pre-fault reactance. For each of the values taken there will be a critical value of post-fault reactance, this reactance being the largest value that will allow 'first swing' synchronism to be retained between the generator and the infinite busbar. The corresponding values of pre-fault and post-fault reactance are plotted as shown, to give the curve marked BASIC CASE. A horizontal line would indicate that stability depends solely on the post-fault conditions. The diagonal line through zero indicates no change in system reactance, generally the zone below this line is not feasible physically since it corresponds to a post-fault reactance that is less than the pre-fault reactance.

Measures taken to improve the stability of the basic case will allow higher post-fault reactance for the same pre-fault reactance, giving a curve above the original curve. The displacement of the new curve from the original curve is a measure of the improvement in stability. This is illustrated by fictitious curves due to improvement in parameters 'A' and 'B'. Parameter 'B' gives a larger improvement in stability, but could well involve more capital cost than implementing parameter 'A'. These could then be related to one

another in a plot such as the one shown, comparing performance improvement with increase in cost. The relationship between performance improvement and increased post-fault reactance could be a simple linear relationship, but need not necessarily be so. The intention of the technique is to enable widely different methods of stability improvement to be compared, whether they be new generator designs, fast control systems, fast switchgear or different security standards.

References

1. Shackshaft, G.: "Improving the Stability of Turbo-Generators, the Interconnected Winding Rotor", (Electrical Times, 29th January 1971).

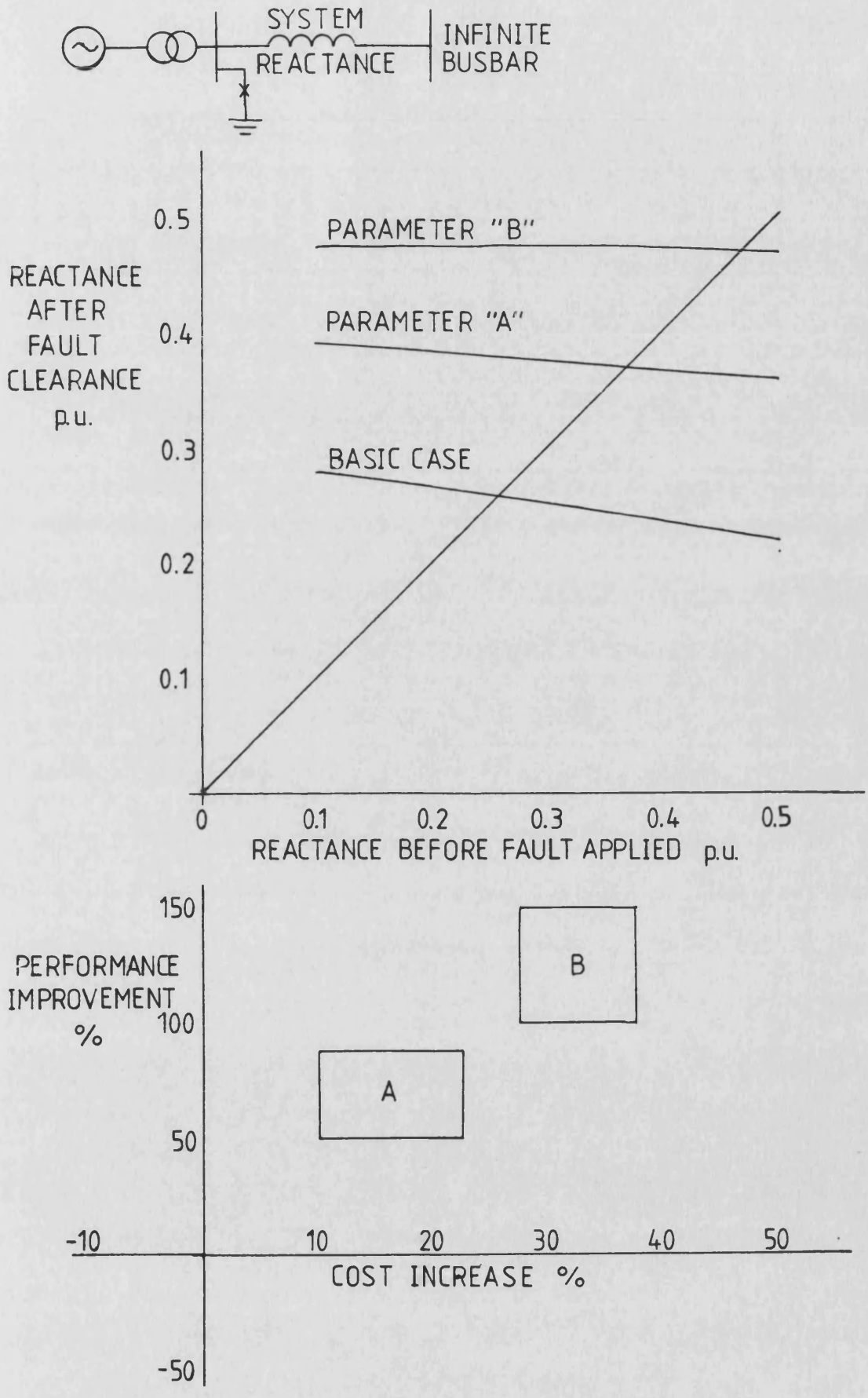
2. Exon, J., La Trobe, CERL, Colloquium on Control of Turbo-Generators, June 1971, Appendix 3.

3. Brown, P. G. et al: "Effects of Excitation, Turbine Energy Control and Transmission on Transient Stability", (IEEE, PAS-89, No. 6, July/August 1970).

PL-ST/PM

21 September 1971

DM


SYSTEM
REACTANCE
INFINITE
BUSBAR
REACTANCE
AFTER
FAULT
CLEARANCE
p.u.
0.5
0.4
0.3
0.2
0.1
PARAMETER "B"
PARAMETER "A"
BASIC CASE
0
0.1
0.2
0.3
0.4
0.5
REACTANCE BEFORE FAULT APPLIED p.u.
PERFORMANCE
IMPROVEMENT
%
150
100
50
-50
B
A
-10
10
20
30
40
50
COST INCREASE %