Practical Optimal Control of Infinite Interval Systems

Leonard H. Lees

University of Aston in Birmingham
Department of Electrical Engineering.

Ph. D. Thesis October, 1968.

SUMMARY

Pontryagin's Maximum Principle was investigated for the finite and infinite time interval cases to assess its usefulness in practical applications. Attention was focused on the analogue control of a second order position control system.

The findings demonstrated that the principle was a useful mathematical tool but not satisfactory for direct application for the finite time interval and virtually impossible for the infinite time interval.

To produce a method of optimisation for the infinite time interval compatible with that of Dynamic Programming, and yet, preserving the formulated advantages of Pontryagin's Maximum Principle, equations were evolved to replace the characteristic two point boundary value problem. A practical controller was then evolved which would enable optimal control to be obtained without the need for resort to a computer.

Optimal control of an actual second order position control system was effected and the results compared with those generally obtained from the application of Dynamic Programming. They were observed to be better.

The evolved method of optimisation was further extended to encompass second and third order systems possessing two time constants, optimal control of an actual third order plant being effected.

ACKNOWLEDGEMENTS

I wish to thank Mr. F. G. Willetts for his continuous guidance and encouragement throughout the project and Professor W. K. Roots for the helpful discussions.

CONTENTS

Page

Introduction

The classical approach to control systems design through such techniques originated by Nyquist and Bode are inadequate for the needs of present day technology. This is particularly true for process control, space vehicle design and guidance and flight controllers. Such problems have thus produced a flood of 'modern mathematical techniques' totally independent of established control theory. The problem of optimisation may be singled out as experiencing the most profound escalation. This escalation, however, has been provoked by mathematicians with almost total disregard for application. It is the purpose of the engineer, therefore, to consolidate and manipulate these new ideas into practical forms. This has successfully been achieved for complex systems, mainly connected with space vehicles for which the techniques were probably originally evolved, but has not been generally presented for application to more mundane problems. Therefore, there still exists a large gulf between 'modern theory' and practice.

Many authors (ref. 6, 33, 34) have given voice to the need for application orientation, but such papers have not been readily forthcoming. There exists a magnitude of papers dealing with optimisation of mathematical problems and computer models. A general formulation of the difficulties and errors entailed in the contruction of an actual analogue controller to realise the postulated goals was not obtainable unless a computer was used as the controlling element. The majority of work in this field is devoted to bang-bang control. An exception, perhaps, is Bellman's Dynamic Programming (ref. 1,20) which has gained popularity as one of the most readily applicable of the modern techniques. Pontryagin's Maximum Principle (ref. 24, 25) on the other hand has gained prominence solely amongst mathematicians or for computer manipulations, its use for actual analogue control not being exploited. This is mainly due to the excess computer time required for the solution of two point boundary value problems peculiar to the Maximum Principle, compared with that required for the solution of the Riccatian equations characteristic to Dynamic Programming. Furthermore, as the canonical equations of Pontryagin have been derived from the equations of Dynamic Programming (ref. 20, 26), the two methods are frequently quoted as producing similar results (ref. 26). This latter statement, due to the lack of literature on the practical application of the Maximum Principle, could not be completely assessed.

It was the object of this research to evolve a direct method of optimisation eliminating the need for a computer as the controlling element and to reduce its role in the solution of the control equations. The resulting control strategy, however, was required to be continuous in nature and maintain the benefits offered by the existing techniques, the principal benefit being the minimisation of a performance index. The techniques considered were Bellman's Dynamic Programming and Pontryagin's Maximum Principle.

The only reference to this topic was that of Freeman and Abbott (ref. 10) who produced an alternative method for the design of optimal linear systems based on Pontryagin's Maximum Principle. The performance index considered was a quadratic and similar to that used throughout the research. The resulting control was similar to that effected by Dynamic Programming, i.e. constant gains were calculated and feedback of all the state vectors required. The resulting trajectories, for the mathematical problem solved, possessed overshoots and no attempt, at any stage, was made to measure the performance index or compare the method with existing techniques.

The most difficult task in any 'modern' optimising technique is the correct choice of the weighting factors for the performance index. Generally, the only way of acquiring these values is by trial and error. This is comparable with the classical method of design where a controller is designed and the resulting plant trajectories evaluated. If these are not compatible with the specification, the controller is modified. The major difference between the two methods is that for the classical approach the physical construction of the controller may have to be modified (the initial form of control being virtually an intuitive guess), whereas the modern approach necessitates the change of parameters, such as gain coefficients. The control strategy and physical form of the controller remain identical for each trial run.

The majority of papers dealing with optimal design do not attempt to measure the resulting value of performance index. This is because it is known that the procedure used will produce the mimumum value. However, due to the trial and error technique usually involved, the value of the performance index will show if the resulting extra cost (i.e. need of greater power amplification, etc.) is compensated by the system improvement. An exception to this is presented in a paper by Ellert and Merriam (ref. 9) where an aircraft landing system is designed. Here, as the system specifications were so complex and the sole object of the design was the correct landing of an aircraft, the value of the performance index was insignificant.

Throughout this research the value of the performance index, and hence the cost of control, was taken as a predominant design feature. When comparing methods of optimisation this proved invaluable as the resulting plant trajectories, although being comparable, produced different values of performance index.

Nomenclature

N.M.P.	Normally monotonic plant, i.e. over-damped output for a plant with negative unity feed-back.
N.O.P.	Normally oscillatory plant, i.e. under-damped output for a plant with negative unity feed-back.
A,B,C	Initial co-state vector conditions
a,b	Reciprocal of plant time constant
P_i	Co-state vectors
x_i	Plant state vectors
λ	Weighting factor
m	Control effort
ω_n	Natural frequency
J.	Value of performance index
E	Plant input function (unless otherwise stated $E \equiv$ a unit step)
S	The Laplace operator
H(f)	Hamiltonian function
Y	Gain coefficient

1. Application of Pontryagin's Maximum Principle to a Second Order Plant

1.1. Pontryagin's Maximum Principle

Rozonoer (ref.28) has given a good explanation of the mathematical application of Pontryagin's Maximum Principle (ref.24,25) and the following is a brief statement of the method as given by Brennan (ref. 2).

Suppose that the response of the controlled object may be described by n first order differential equations. Then, at any time, the state of the object may be described by n state variables $x_1 \ldots\ldots x_n$. Suppose that there are r controlling elements with position $m_1 \ldots\ldots m_r$. The system may then be described by the n first order differential equations.

$$x_i = f_i(x_1, \ldots x_n; m_1, \ldots m_r; t), \; i = 1, \ldots n \qquad (1.1.1)$$

The control problem is to choose m(t), where m is the vector $m_1, \ldots, mr$, to minimise the performance index

$$J = \sum_{k=1}^{n} c_k x_k \;(T) \qquad \ldots \qquad (1.1.2)$$

where T is some time which may be fixed or free. This is a general criterion and it includes the problem of minimizing

$$J = \int_{o}^{T} F(x_1, \ldots, x_n; m_1, \ldots, m_r)dt \qquad (1.1.3)$$

In this case we introduce a new state variable.

$$x_{n+1}(t) = \int_{o}^{t} F(x_1, \ldots, x_n; m_1, \ldots, m_r)dt \qquad (1.1.4)$$

Thus a new equation

$$\dot{x}_{n+1} = F(x_1, \ldots, x_n, m_1, \ldots, m_r) \qquad (1.1.5)$$

may be added to the set of equations (1.1.1). Then the problem of minimizing the integral of equation (1.1.3) reduces to that of minimizing the (n + 1)st state variable at the final instant of time T.

To solve this problem, form the function

$$H(x,p,m,t) = \sum_{k=1}^{n} p_k f_k(x_1, \ldots, x_{n+1}, m_1, \ldots, m_r; t) \qquad (1.1.6)$$

where $\dot{x}_i = \frac{\partial H}{\partial pi}$ .. .. (1.1.7)

and $\dot{p}_i = \frac{-\partial H}{\partial x_i}$ $i = 1, \ldots, n + 1$ .. .. (1.1.8)

and x;p are the vectors $x_1, \ldots, x_{n+1}$; $p_1, \ldots, p_{n+1}$, respectively. The boundary conditions are given by

$$x_i(o) = x_i^{o}$$

the initial conditions of the state variables, and

$$p_i(T) = -C_i \qquad .. \qquad .. \qquad .. \qquad (1.1.9)$$

At time T, the vector p is equal in magnitude but opposite in direction to the vector $c = c_1, ..., c_{n+1}$. The maximum principle then states that

$$J = \sum_{k=1}^{n+1} c_k x_k(T)$$

is minimized when H(x,p,m,t) is maximized for all t,

$$0 \leqslant t \leqslant T$$

As demonstrated, optimal control is effected through Pontryagin's Principle by constructing, via mathematical means, an adjoint system to produce an optimal control effort (m). Although the proof is highly mathematical, the mathematical manipulations required to produce the adjoint system are relatively simple and may be performed by purely algebraic means. Great difficulty arises, however, in solving the inevitable two point boundary problem for the required initial conditions (c) of the co-state or 'p' vectors of the adjoint system.

1.2. Performance Index

The initial step in the optimisation of any system is that of choosing a performance index. The form of the index and its associated weighting factors are very critical to the resulting optimal design and lengthy analysis is often necessary to obtain its correct or required form.

An effect of the performance index is that, depending upon the optimising procedure, the index may dictate the whole optimal strategy. Pontryagin's Maximum Principle was observed to be such a procedure for the index determined whether control would be effected via analogue or bang-bang means and which vectors would be required for feedback purposes.

A performance index whose integrand consists of vectors raised to unity power will in general not be convex and lead to bang-bang control (section 2.1). Under such circumstances a unique optimal solution may not be obtained (ref. 18). A quadratic performance index is generally convex and results in continuous control. This produces a unique optimal solution.

A quadratic index was used throughout the research. This took the form:

$$J = \int_0^T \left[(\text{error})^2 + \lambda m^2\right] dt$$

where 'm' is the optimal control effort and λ a weighting or penalty factor. (Throughout the research the value of λ was taken as 0.1.) Optimisation was achieved by minimising J. J may be assumed to be a cost function for minimising a system error and cost of control.

The significance of the performance index and its integrand are discussed in section 2.1.

1.3. Mathematical application to a second order plant

As mentioned in the introduction, literature was not available which clearly stated the practical application of Pontryagin's Maximum Principle. The initial research was therefore devoted to this end.

Consider a second order plant whose transfer function in the S plane may be represented by

$$\frac{y}{s(s + a)}$$

The resulting state vectors may be represented by the first order differential equations

$$\dot{x}_1(t) = Yx_2(t) \quad .. \quad .. \quad .. \quad .. \quad (1.3.1)$$

$$\dot{x}_2(t) = -ax_2(t) + m \quad .. \quad .. \quad .. \quad (1.3.2)$$

The performance index

$$J = \int_0^{t_f} \left[(E - x_1(t))^2 + \lambda m^2(t)\right] dt \quad .. \quad (1.3.3)$$

where E is the system's desired output, produces a third state variable

$$\dot{x}_3(t) = (E - x_1(t))^2 + \lambda m^2(t) \quad .. \quad (1.3.4)$$

The Hamiltonian for the system may now be written as

$$H = p_1 {}^{y}x_2 + p_2.(-ax_2 + m) + p_3.((E - x_1)^2 + \lambda m^2) \quad .. \quad (1.3.5)$$

Pontryagin proved that maximising the Hamiltonian minimised the index thus, in maximising H w.r.t.m

$$\frac{\partial H}{\partial m} = p_2(t) + 2\lambda m(t) p_3(t) \quad .. \quad .. \quad .. \quad (1.3.6)$$

$$\dot{p}_i(t) = -\frac{\partial H}{\partial x_i} \quad .. \quad .. \quad .. \quad .. \quad (1.3.7)$$

and $$\dot{x}_i(t) = \frac{\partial H}{\partial p_i} \quad .. \quad .. \quad .. \quad .. \quad (1.3.8)$$

From equations 1.3.5 and 1.3.7

$$\dot{p}_3(t) = -\frac{\partial H}{\partial x_3} = 0 \quad .. \quad .. \quad .. \quad .. \quad (1.3.9)$$

Equation 1.3.9 demonstrates that p_3 is a constant and will be equal to -1.

$$\therefore \quad m^o(t) = \frac{P_2(t)}{2\lambda} \quad .. \quad .. \quad (1.3.10)$$

From equations 1.3.5 and 1.3.7

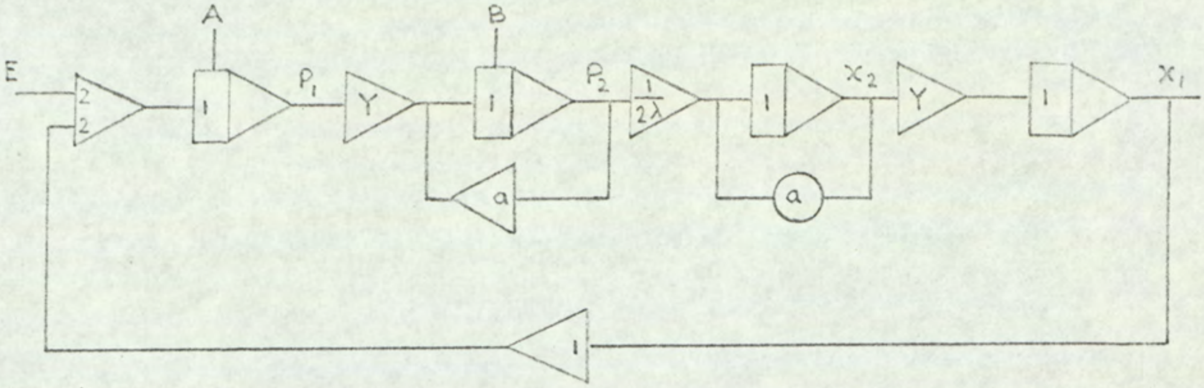
$$\dot{p}_1(t) = 2.(x_1(t) - E) \quad .. \quad .. \quad .. \quad .. \quad (1.3.11)$$

and

$$\dot{p}_2(t) = a.p_2(t) - Yp_1(t) \quad .. \quad .. \quad .. \quad (1.3.12)$$

The values of the p vectors at time $t = t_f$ must be zero, their value at time t = 0 are unknown.

The complete optimal system may be simulated on an analogue computer with equations 1.3.11 and 1.3.12 representing the adjoint system (fig. 1.3.1).



Computer Model of Optimal System

Fig. 1.3.1

The values of the initial conditions of the co-state vectors p_1 and p_2 (A and B respectively) are the solutions of a two point boundary problem.

1.4. Iterative Procedure for Optimal System Trajectories

Initially it was decided to solve the two point boundary problem on a digital computer. This necessitated an iterative procedure. Such procedures are well established and include 'Method of Finite Differences', various gradient and relaxation methods and 'Boundary Iterations'. The mentioned methods are summarised in ref.22 .

It was decided that a gradient method would be used and the 'Steepest Ascent of the Hamiltonian' (ref.3,14), was observed to have been formulated to solve the problem in the form presented by the mathematics of Pontryagin (Appendix 1). The method took the form of computing the controlling effort (m), trajectory, correcting it in such a direction as to increase the Hamiltonian according to

$$m_{i+1} = m_i + K \frac{\partial H}{\partial m} \quad \text{.. .. .. ..} \quad (1.4.1)$$

for each iteration (i), and re-calculating until a convergence was acquired, i.e. when the required optimum effort had been attained, the partial derivative was equal to zero producing $m_{i+1} = m_i$. (The resulting flow diagram is shown in Appendix 9.1.)

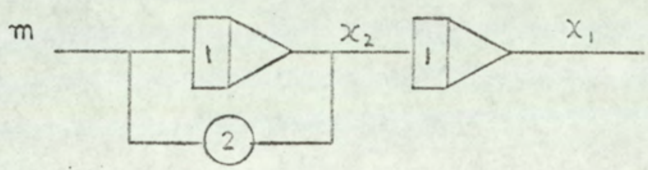
1.5. Optimisation by Digital Simulation

The initial plant considered possessed an open loop transfer function of

$$\frac{1}{s(s + 2)}$$

the desired output on closed loop being unity.

When the system is depicted as in fig. 1.5.1, the



Analogue Model for $\frac{s}{s(s + 2)}$

Fig. 1.5.1

state equations may be expressed as

$$\dot{x}_1(t) = x_2(t) \quad \text{.. .. ..} \quad (1.5.1)$$

$$\dot{x}_2(t) = m(t) - 2x_2(t) \quad \text{.. ..} \quad (1.5.2)$$

The third state variable to be minimised may, in accordance with equation 1.1.5, be expressed as

$$x_3(t) = (E - x_1(t))^2 + 0.1m^2(t) \quad .. \quad .. \quad (1.5.3)$$

where 0.1 is the weight (λ) attached to m(t).

The Hamiltonian for the system may be written as

$$H = p_1(t)x_2(t) + p_2(t).(m(t) - 2x_2(t)) + p_3(t).[(E - x_1(t))^2 + 0.1m^2(t)]$$

given $\dot{p}_i = -\dfrac{\partial H}{\partial x_i}$

$$\dot{p}_1 = 2(x_1 - E) \quad .. \quad .. \quad .. \quad (1.5.4)$$

$$\dot{p}_2 = 2p_2 - p_1 \quad .. \quad .. \quad .. \quad (1.5.5)$$

and $\dot{p}_3 = 0$ .. .. .. (1.5.6)

$$\frac{\partial H}{\partial m} = p_2(t) + 0.2p_3(t)m(t) = 0 \text{ (for a maximum)}$$

$$\therefore m^o(t) = \frac{-p_2(t)}{0.2p_3(t)} \quad .. \quad .. \quad .. \quad (1.5.7)$$

Equation 1.5.6 shows that p_3 is a constant and will equal -1. Equation 1.5.7 may therefore be written

$$m^o(t) = +5p_2(t) \quad .. \quad .. \quad .. \quad (1.5.8)$$

where $m^o(t)$ is the optimum system control effort.

The initial values on the state vectors (x_1 and x_2) were taken to be zero. The final value of x_1 was required to be unity while that of x_2 was unknown. The final values of the co-state vectors would be zero while their initial conditions are unknown.

A block diagram of the optimum system is shown in fig. 1.5.2. The digital programme which simulated the system for an optimising period of 2 seconds is shown in Appendix 5. The integrating routine was that of ref.29 with a time increment of 0.01 seconds.

The resulting computer convergence is depicted in table 1.5.1 and the optimum trajectories by fig. 1.5.3.

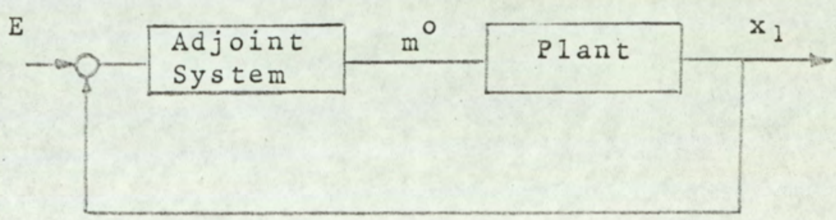
A great disadvantage of the method employed was in the selection of the up-rating constant constant K of equation 1.4.1. It was observed that this constant was very critical to the convergence of the programme. For an optimising period of two seconds, four trial runs were required to determine a rough value, and a fifth to obtain a value which produced an acceptable convergence. If the value of K was too large, the programme became unstable, if it was too small, convergence was very slow. The resulting value of K was 2.5 and the time taken for each iteration was 1½ minutes. (An Elliott 803 computer was used). It thus necessitated approximately two hours actual computing time to obtain the initial conditions for the adjoint system.

As the period over which optimisation was being considered was increased, so the constant K became more

TABLE TO SHOW CONVERGENCE OF CONTROL VARIABLE

Iteration / time	1	10	15	18
2.0	0.5000	0.0009	0.0000	0.0000
1.8	0.5214	0.04427	0.0420	0.04198
1.6	0.5923	0.1107	0.1105	0.1104
1.4	0.7124	0.2242	0.2244	0.2244
1.2	0.8812	0.3886	0.3892	0.3892
1.0	1.0978	0.6138	0.6147	0.6147
0.8	1.3611	0.9131	0.9142	0.9142
0.6	1.6694	1.3014	1.3027	1.3026
0.4	2.0202	1.7931	1.7946	1.7945
0.2	2.4097	2.3978	2.3997	2.3996
0.0	2.8324	3.1143	3.1166	3.1165

Table 1.5.1



Block Diagram of Optimal System

Fig. 1.5.2

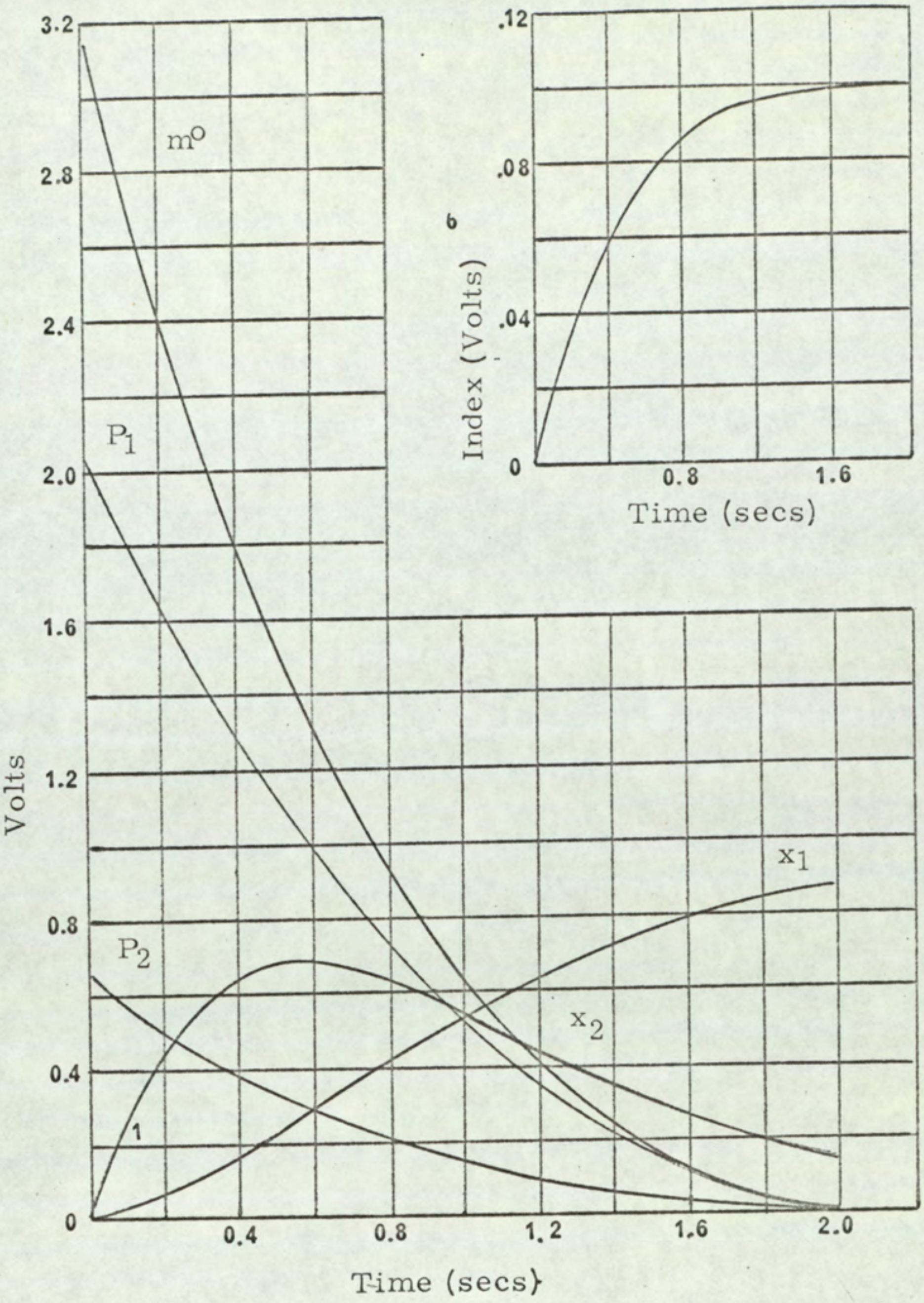


FIG. 1.5.3.
Optimum trajectories for plant with open loop-transfer

function $\frac{1}{S(S+2)}$

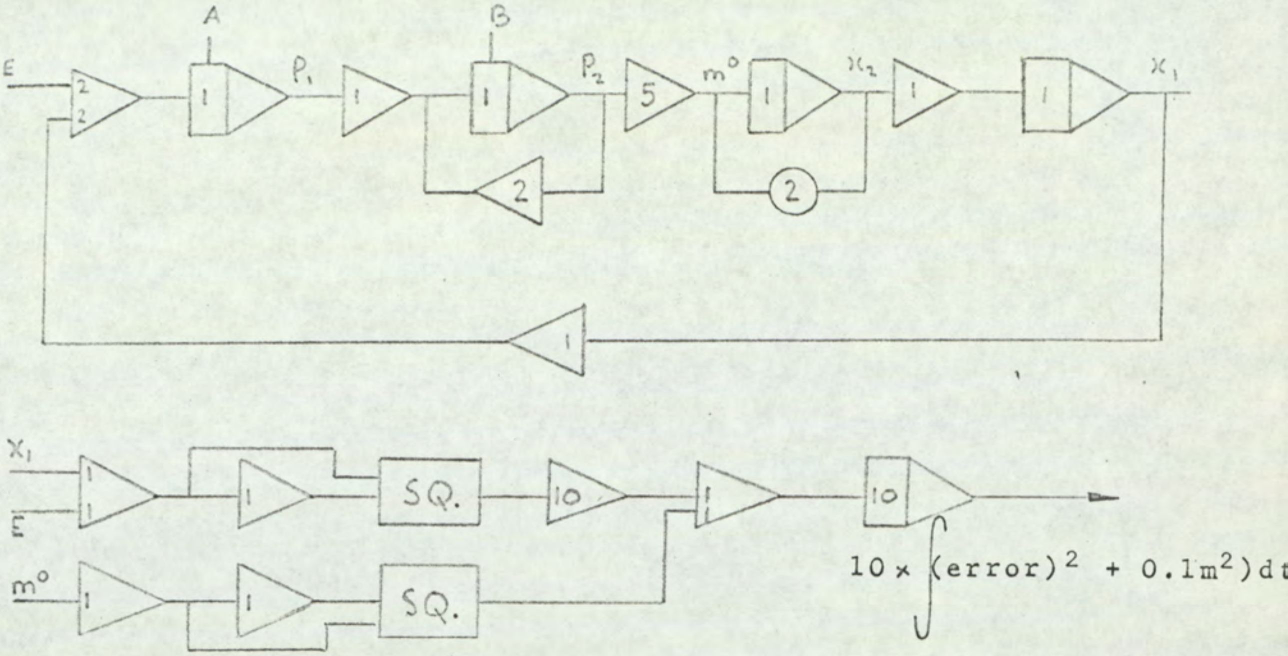
critical. With an optimising period of four seconds, a satisfactory K was observed to be 0.70. The time per iteration was three minutes. It was further observed that to obtain an acceptable convergence, many more iterations were required. The number of iterations for longer optimising periods were reduced by making K as large as possible, without invoking instability, for the initial iterations and reducing it for the latter.

Other digital techniques (as mentioned in section 1.4) were considered but all were observed to have some deficiency or undesirable characteristic similar to those discussed.

1.6 Solutions of two-point boundary value problems via an Analogue Computer

As the solution of the characteristic two-point boundary problem required lengthy computing time per solution on a digital computer, an analogue computer was employed with the intention of obtaining much faster solutions.

The analogue computer diagram for the system (equations 1.5.1, 1.5.2, 1.5.4, 1.5.5 and 1.5.8), as shown in fig. 1.6.1, was used with the initial values of the co-state vectors applied via potentiometers. The analogue circuit for measuring the index is also shown.



Analogue Computer Circuits for Plant $\frac{1}{s(s+2)}$ and the Evaluation of the Performance Index

Fig. 1.6.1

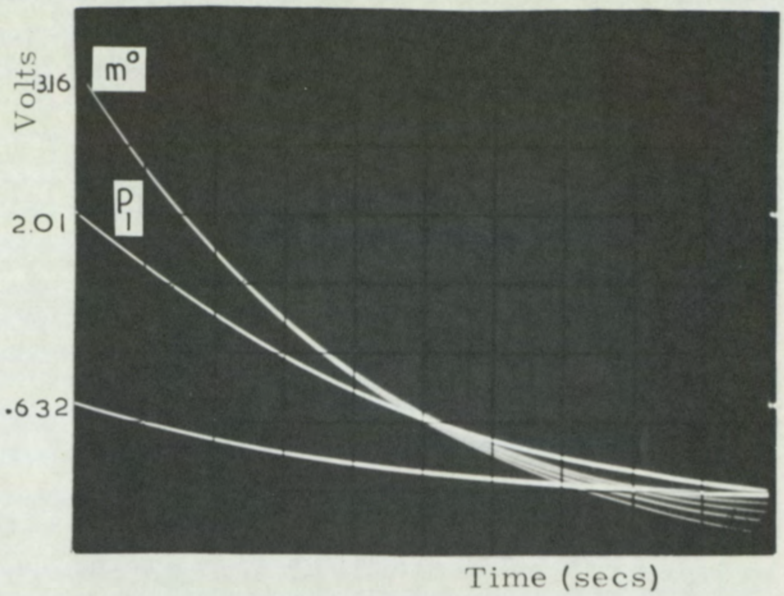
Optimisation was achieved by operating the computer in its cyclic mode and adjusting the co-state vector initial conditions until they attained zero magnitude at the termination of the optimising interval. The cyclic period was initially small and gradually increased to the required optimising interval; the initial values of co-state vectors being adjusted to produce zero co-state values at the termination of each cycling increment.

Initially an Electronics Associates TR-10 computer was used, but it was found unsatisfactory due to insufficient resolution of the potentiometers for the setting of the initial conditions. To obtain optimum conditions, an Electronics Associates TR48 computer was used, the results of which are shown in fig. 1.6.2. The values

Co-state vectors
and control effort

0. 5 Volts/cm

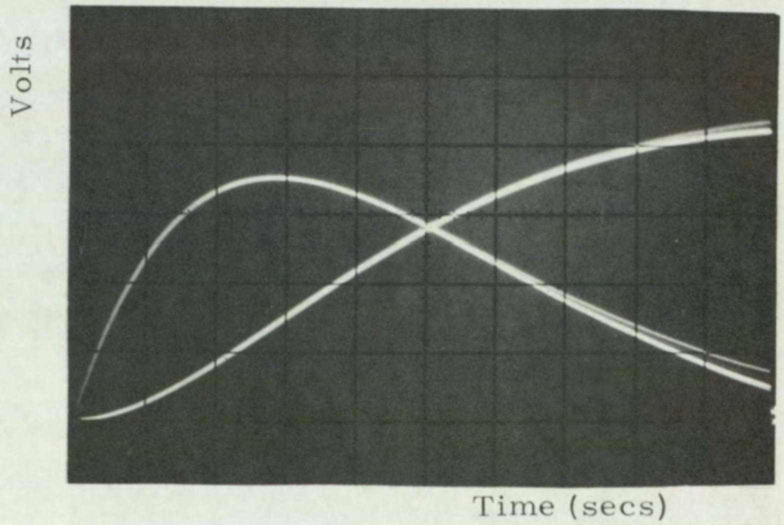
0. 2 secs/cm.



Plant state vectors

0. 2 Volts/cm

0. 2 secs/cm.



Index.

0. 2 volts/cm

0. 2 secs/cm.

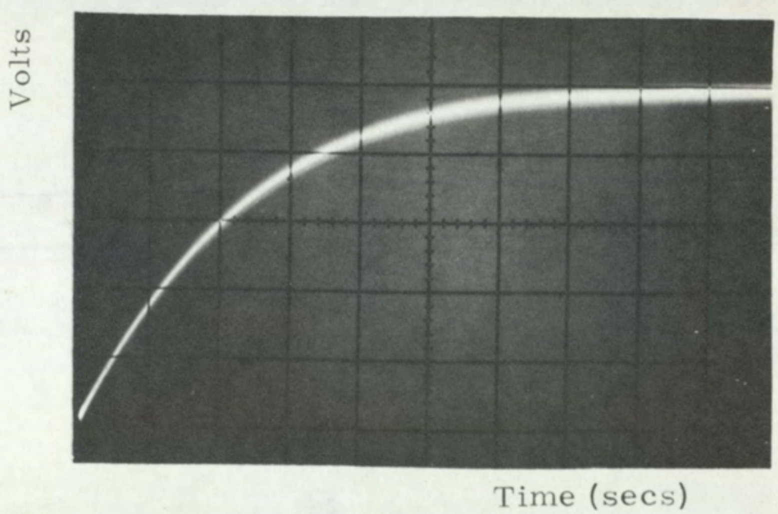
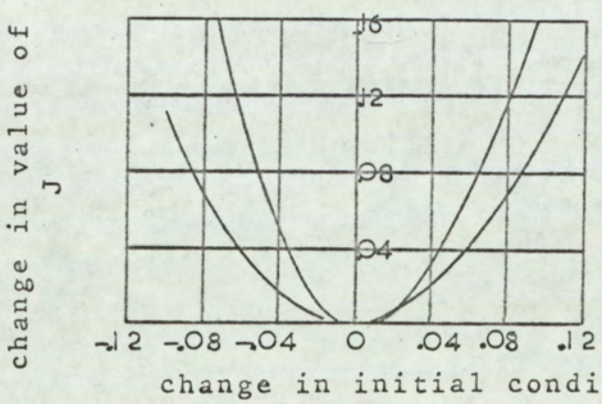


FIG. 1. 6. 2.

Optimum trajectories
for plant with open
loop transfer function $\frac{1}{S(S+2)}$

of the initial conditions of the co-state vectors and the value for the index compare very favourably with those obtained from the digital solution (fig. 1.5.3). (The instability of the trajectories of fig. 1.6.2 are discussed in section 2.4). To verify that an optimum system had been attained, the change in the value of the index was observed for changes in the initial conditions. The results are depicted in fig. 1.6.3 from which it may be observed that any change in the initial conditions only increased the value of the index.



Curves showing the increase in the value of performance index (J) for change in the value of the co-state vectors.

Fig. 1.6.3

Once the circuit had been patched on the computer, it took approximately ten minutes to obtain the required initial values of the co-state vectors for an optimising period of two seconds, which, due to the inherent optimising procedure, included the relevant initial conditions for many optimising periods less than two seconds.

2. Results and Observations from the Analogue Computer

The facility for quickly solving the two-point boundary value problem rendered the analogue computer a much more usable tool for the investigation of Pontryagin's Principle than that of the digital computer. It was also much more adaptable and versatile for model manipulation.

2.1. Performance Index

Many authors have stated that Pontryagin's Maximum Principle either produces bang-bang systems or open loop control (ref.19,37). This was conceived by considering an isolated problem and drawing general conclusions.

The optimum control effort for a plant, according to Pontryagin mathematics, is given by:

$$\frac{\partial H}{\partial m} = 0 \qquad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \qquad (2.1.1)$$

If m is only present in the Hamiltonian to unity power, then equation 2.1.1 will not contain m and thus a bang-bang system will result. If, however, m appears in the Hamiltonian in quadratic form, then m will be present in equation 2.1.1 and the resulting optimal system will consist of only analogue vectors. Similarly, as the co-state vectors are given by:

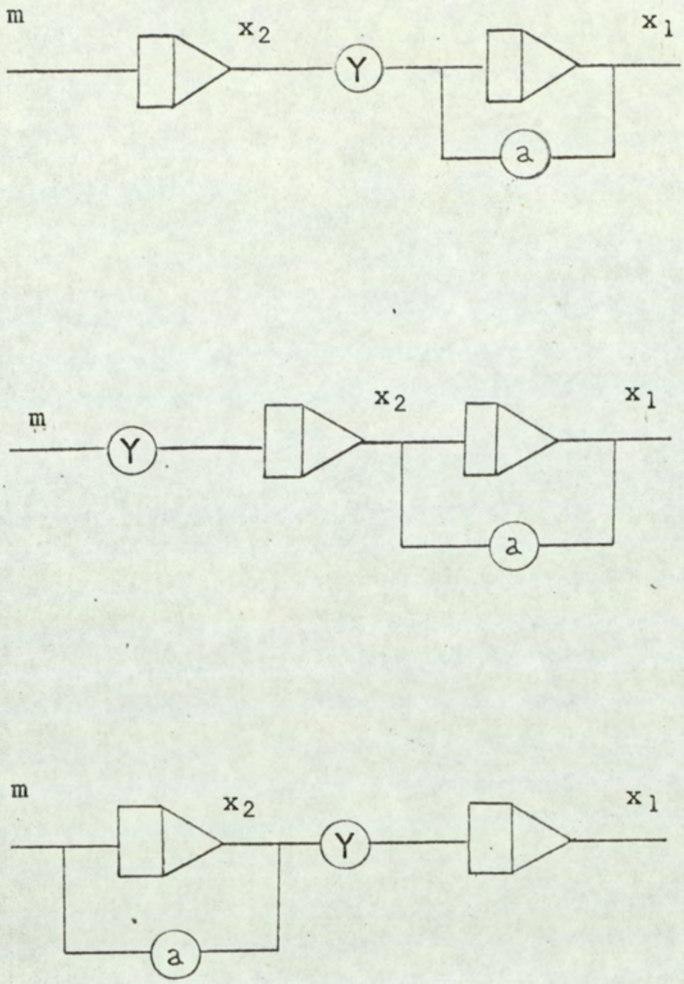
$$\dot{p}_i = -\frac{\partial H}{\partial x_i} \qquad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \qquad (2.1.2)$$

any state vector x_i which appears in quadratic form in the Hamiltonian will be an integral part of a co-state vector and will thus be required for feedback purposes.

These quadratic terms may only be introduced by way of the (i + 1)th state vector, i.e. via the performance index. The type of control, (i.e. bang-bang or analogue), and the state vectors required for feedback may therefore be completely determined at the onset by the choice of the performance index.

2.2. Mathematical Model of Plant

The plant to be optimised may be represented in many different forms, the transfer function of each form remaining identical. This is demonstrated in fig. 2.2.1 where a general second order plant with a transfer function of $\frac{y}{s(s + a)}$ is represented in three different modes. Due to differing state equations, each mode will give rise to a different adjoint system. The optimum control effort m^o must, however, remain identical in each case to produce identical outputs and values of performance index. This requires the co-state vectors p_2 (the co-state vector directly responsible for m^o) to possess identical initial



$$\dot{x}_1 = Yx_2 - ax_1$$
$$\dot{x}_2 = m$$

$$\dot{x}_1 = x_2 - ax_1$$
$$\dot{x}_2 = Ym$$

$$\dot{x}_1 = Yx_2$$
$$\dot{x}_2 = m - ax_2$$

Fig. 2.2.1

Three Models Representing the Transfer Function

$$\frac{y}{S(S + a)}$$

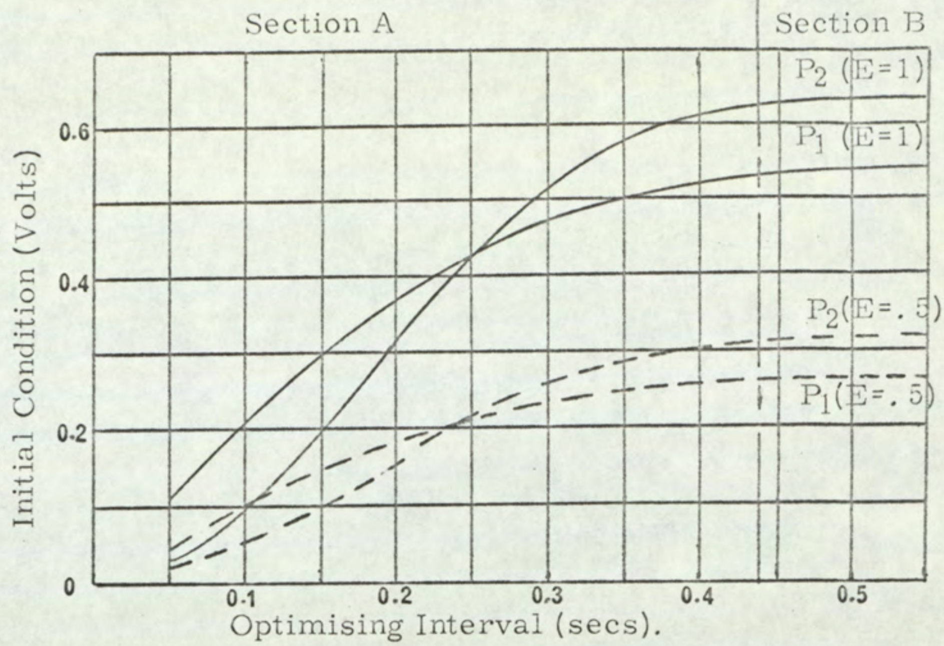
conditions. The value of the initial conditions on the other co-state vectors must therefore be different to compensate for the re-arrangement of the system components.

The variation of choice for the mathematical model was of assistance in avoiding saturation when optimisation was performed on an analogue computer.

2.3. Variation of System Vectors with Change in Optimising Interval

Curves of the initial conditions for the co-state vectors, which produced optimum systems for various optimising intervals (fig. 2.3.1), show that after a certain period of time the initial conditions tend to a constant value (i.e. region B of fig. 2.3.1). These constant initial conditions may be explained with reference to the co-state vector trajectories. When both co-state vectors tend to zero magnitude asymptotically at the termination of the optimising interval, increasing the interval would still produce zero terminal magnitudes and so satisfy the requirement for optimality without further change of the initial conditions (fig. 2.3.2). For optimising intervals in region A (fig. 2.3.1), at least one co-state vector would be expected not to approach zero asymptotically.

Trajectories for plant with transfer function $\frac{10}{S(S+2)}$ (N. O. P.)



Trajectories for plant with transfer function $\frac{1}{S(S+2)}$ (N. M. P.)

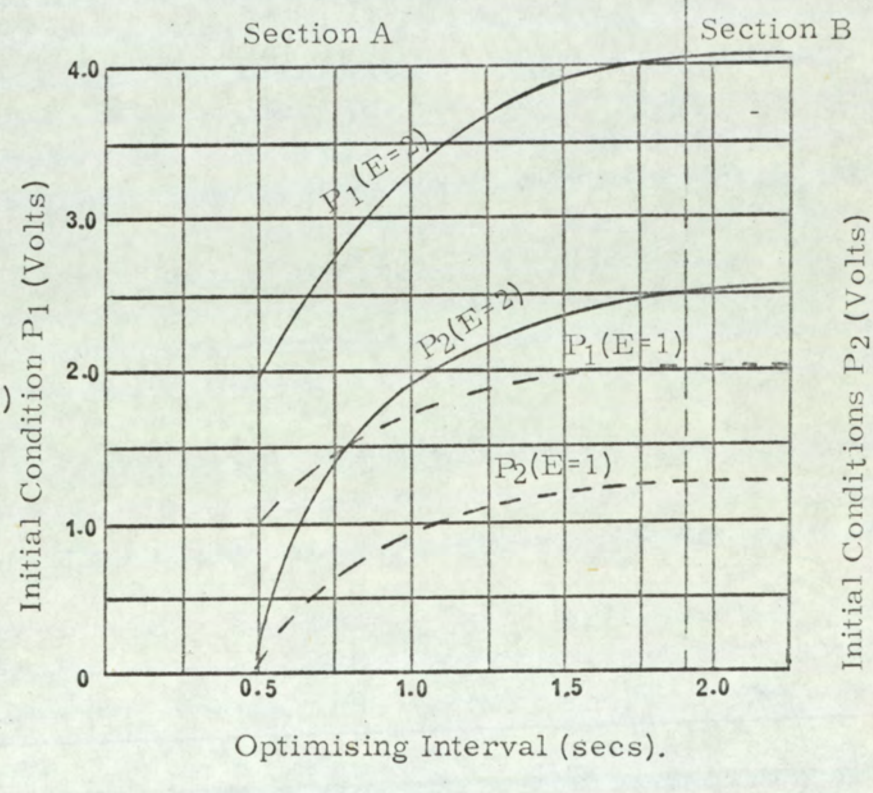


FIG. 2.3.1.

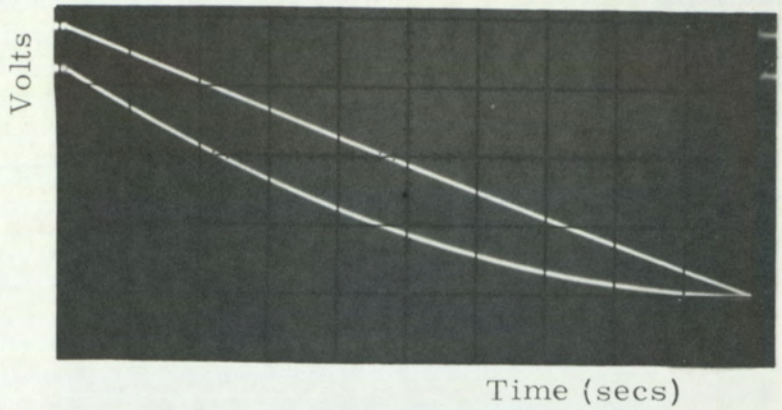
Curve of co-state vector initial conditions (to produce optimisation) against optimising interval.

Co-state vectors not asymptotic.

Plant N. O. P. $\frac{10}{S(S+2)}$

0. 1 Volt/cm.

0. 02 secs/cm.

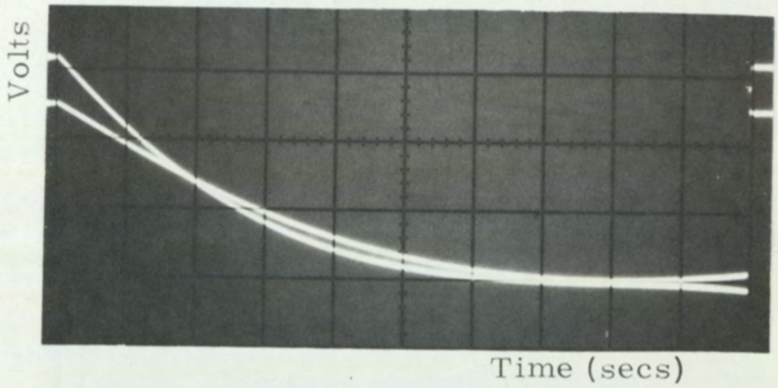


Co-state vectors asymptotic.

Plant N. O. P. $\frac{10}{S(S+2)}$

0. 06 secs/cm

0. 2 Volts/cm.

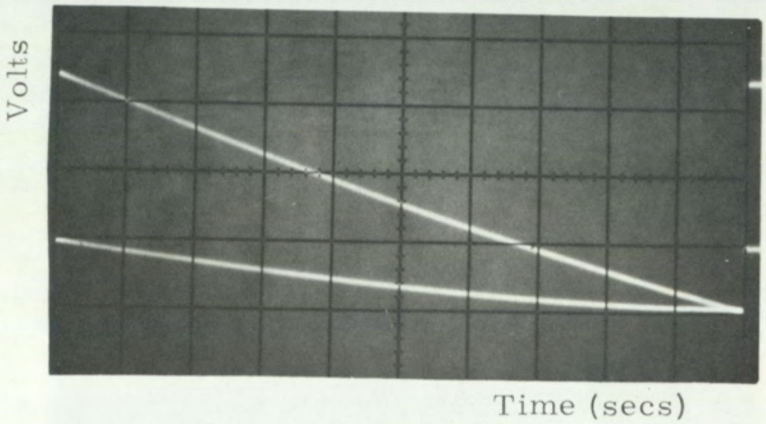


Co-state vectors not asymptotic.

Plant N. M. P. $\frac{1}{S(S+2)}$

0. 1 Secs/cm

0. 5 Volts/cm.



Co-state vectors asymptotic.

Plant N. M. P. $\frac{1}{S(S+2)}$

0. 25 secs/cm.

0. 5 Volts/cm.

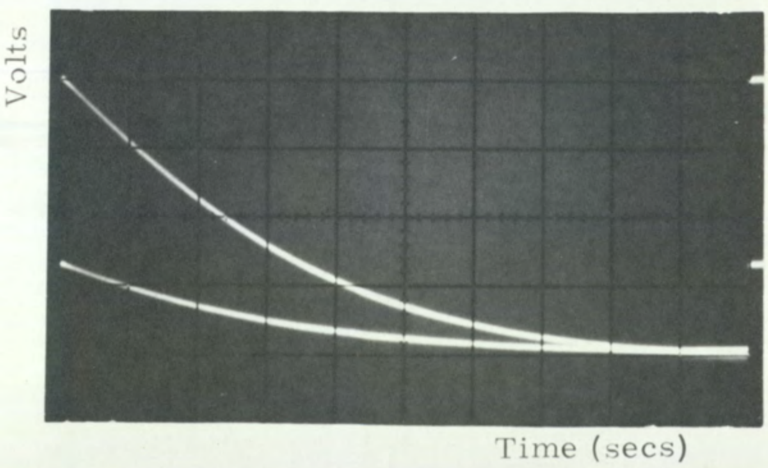


FIG. 2. 3. 2.

(fig. 2.3.2). Increasing the optimising interval would thus not maintain zero terminal conditions. The value of the initial conditions in region B was further observed to be directly proportional to the desired plant output.

The constant value of the initial conditions on the co-state vector p_2 (producing m^o) was observed to be identical for every plant considered of the form $\frac{y}{s(s + a)}$. This suggested that this initial condition was independent of the plant's parameters. (This was later verified (Section 4) for a large class of systems).

The maximum values of the optimum plant output for optimising periods less than that at which constant initial values prevailed, i.e. region A of fig. 2.3.1, was always less than that desired; the desired output, or slightly greater, being attained for optimising periods in region B as shown in fig. 2.3.3. Fig. 2.3.3 also depicts the observation that plants which were N.O.P produced optimum outputs which did not attain zero slope at the termination of the optimising interval, while systems which were N.M.P. did attain zero slope when their optimum output approached the desired value.

2.4. Optimising Period and System Noise

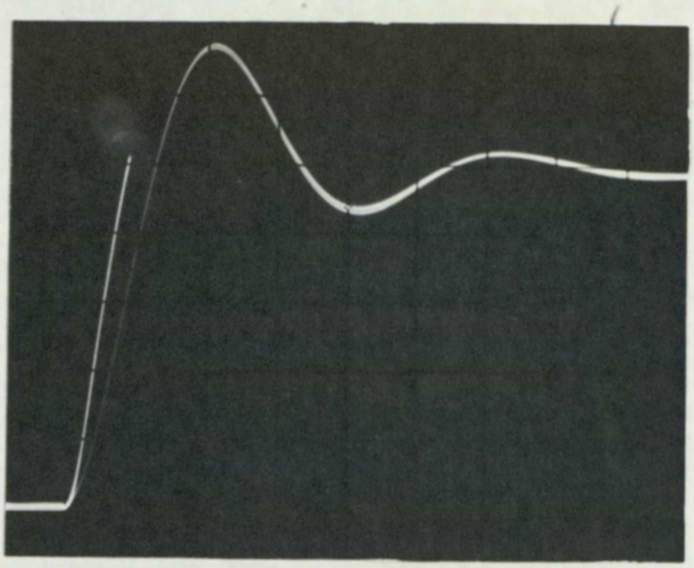
Negative exponentials in the plant being optimised

N.O.P $\frac{10}{S(S + 2)}$

0.5 secs/cm

0.2 volts/cm

Volts



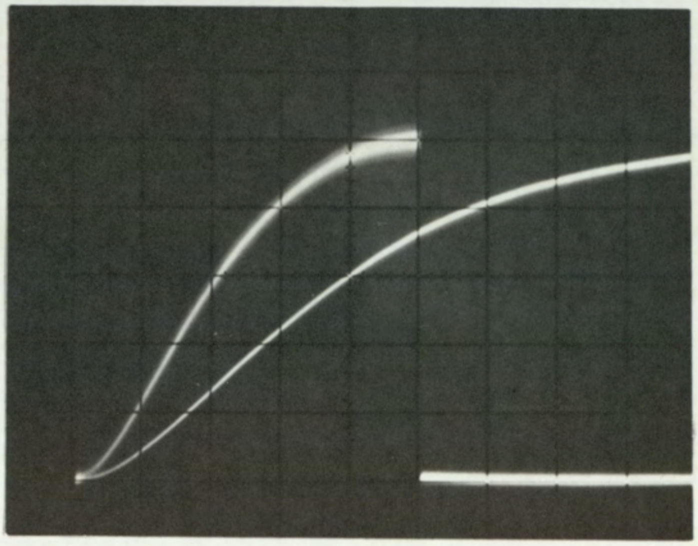
Time (secs)

N.M.P. $\frac{1}{S(S + 2)}$

0.5 secs/cm

0.2 volts/cm

Volts



Time (secs)

Fig. 2.3.3

Output of Second Order Plant with and without Optimising Control produce positive exponentials in the adjoint system by virtue of the negative sign in the canonical equation for the co-state vectors (equation 1.1.8). These positive exponentials render the adjoint system extremely sensitive to noise and magnitudes of the initial-conditions. This sensitivity to variation in initial conditions (portrayed in detail in section 5.2) produced curves, taken from the analogue computer, as families rather than single trajectories (fig. 1.6.2). This was due to error in the computer when re-setting the integrators at the commencement of each cycling period (an error of approximately 0.5%). Error in the re-setting of the initial conditions would produce error in the co-state vectors and, as the optimal control effort for the performance index used was five times greater than p_2, this error was accordingly amplified.

The sensitivity of the co-state vectors will be most pronounced in the region of zero magnitude as there is little or no control over the adjoint system in this region. It is for this reason that, when the optimising period has terminated, the system must be switched off as, once the co-state vectors attain zero magnitude, there is no corrective control over them and the slightest noise will produce instability. This phenomenon was experienced on the analogue computer and the trajectories of fig. 2.4.1. show them beginning to leave the zero axis soon after they attain zero magnitude, the optimising period being sufficient to produce zero error. (The co-state vectors will go unstable after the termination of the finite time interval due to noise and the non-zero error, i.e. the finite time interval may be assumed to be that interval of time at the termination of which the integrand of the performance index has not been reduced to zero magnitude.)

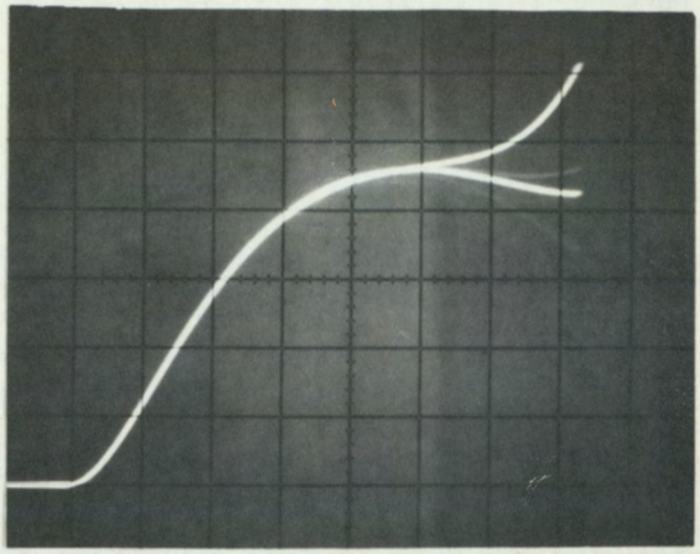
The foregoing underlines the fact that Pontryagin's principle cannot be used in the infinite interval case and that there is a limit to the period over which optimisation may be achieved. This limit was the time at which the co-state initial conditions tend towards a constant value and will depend upon the plant parameters, i.e. the maximum optimising period is that at which the plant output attains its desired value. For the majority of N.M.P. instability occured before a sufficient optimising period was attained to enable the optimum output to reach its desired value (fig. 2.3.3). This was due to the co-state vectors converging to zero magnitude very gradually and thus being exceptionally susceptible to noise. This gradual convergence contributed to the optimum output of normally monotonic systems tending towards zero slope.

Output

0.2 volts/cm

0.5 secs/cm

Volts



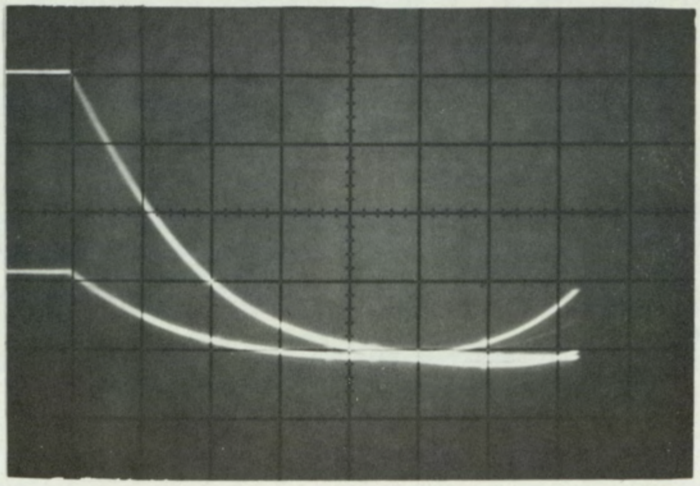
time (secs)

Co-state vectors

0.5 volts/cm

0.5 secs/cm

Volts



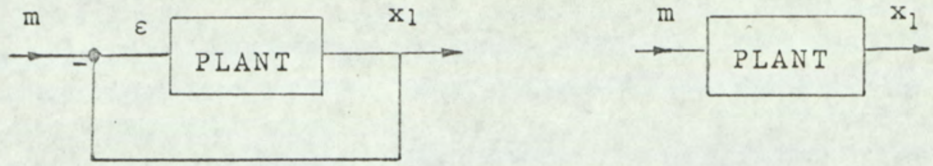
time (secs)

Fig. 2.4.1.

Unstable Vectors after the Termination of the Optimising Interval

2.5. Optimisation of an Open or Closed Loop Plant

Consider the closed and open loop plants of fig. 2.5.1.



Closed Loop   Open Loop

Fig. 2.5.1

The performance index for the open loop plant is:

$$\int_{o}^{t_f} \left[(E - x_1)^2 + \lambda m^2\right] dt \quad .. \quad .. \quad (2.5.1)$$

An equivalent index for the closed loop plant would be:

$$\int_{o}^{t_f} \left[(E - x_1)^2 + \lambda (m - x_1)^2\right] dt \quad .. \quad (2.5.2)$$

as the error ε is comparable with m of the open loop system.

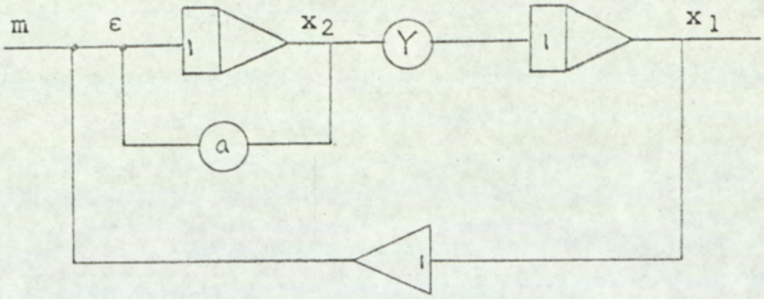
From the mathematical model of fig. 2.5.2, the state equations for the closed loop plant are:

$$\dot{x}_1 = Yx_2 \quad .. \quad .. \quad .. \quad .. \quad (2.5.3)$$

$$\dot{x}_2 = m - ax_2 - x_1 \quad .. \quad .. \quad .. \quad (2.5.4)$$

and

$$\dot{x}_3 = (E - x_1)^2 + \lambda(m - x_1)^2 \quad .. \quad (2.5.5)$$



Analogue Computer Model for a Unity Feedback Plant with Open Loop Transfer Function $\frac{y}{(S + a)}$

Fig. 2.5.2.

The Hamiltonian may be written:

$$H = p_1 Y x_2 + p_2(m - ax_2 - x_1) + p_3.((E - x_1)^2 + \lambda(m - x_1)^2)$$

$$\frac{\partial H}{\partial m} = p_2 - 2\lambda(m - x_1); \; p_3 \text{ being equal to } -1$$

$$\therefore \; m^o = \frac{p_2}{2\lambda} + x_1 \quad .. \quad .. \quad .. \quad .. \quad (2.5.6)$$

From the canonical equations:

$$\dot{p}_1 = p_2 + 2(x_1 - E) + 2\lambda(x_1 - m^o) \quad .. \quad .. \quad (2.5.7)$$

$$\dot{p}_2 = ap_2 - Yp_1 \quad .. \quad .. \quad .. \quad .. \quad (2.5.8)$$

From equation 2.5.6:

$$p_2 = 2\lambda(m^o - x_1)$$

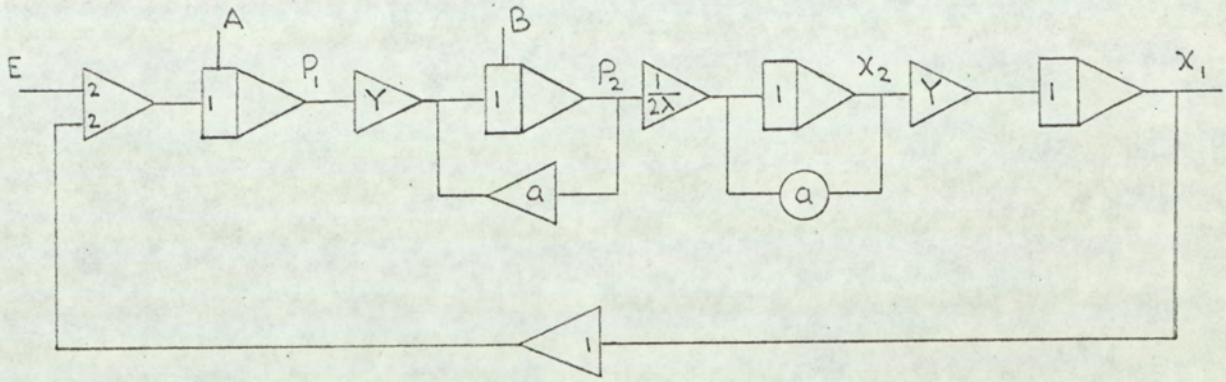
Equation 2.5.7 may now be written:

$$\dot{p}_1 = 2(x_1 - E) \quad .. \quad .. \quad .. \quad .. \quad (2.5.9)$$

Substituting equation 2.5.6 into equation 2.5.4 produces

$$\dot{x}_2 = \frac{p_2}{2\lambda} - ax_2 \quad .. \quad .. \quad .. \quad (2.5.10)$$

The optimum system may be realised from equations 2.5.3, 8, 9 and 10 (fig. 2.5.3).



Computer Model of Optimal System

Fig. 2.5.3

Fig. 2.5.3 is identical to that of fig. 1.3.1 (analogue diagram for equivalent open loop plant).

The net result is that optimising an open loop system to the index of equation 2.5.1 produces identical control, and hence values of co-state vector initial conditions, as optimising an initially closed loop system to the comparable index of equation 2.5.2.

Thus, an adjoint system designed to optimise an open loop plant could be used to optimise the same plant possessing unity feedback. This enables optimal control to be applied to a plant possessing unity feedback without the need for removing the feedback loop or even stopping the process. This may be advantageous for chemical plants, large furnaces and any process where shut down would prove an extremely costly or lengthy procedure.

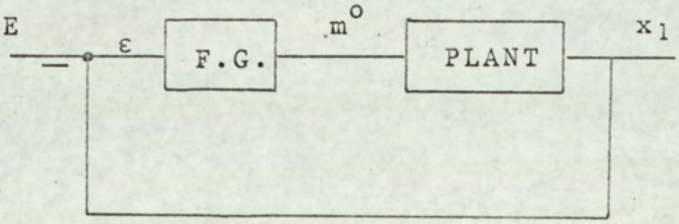
3. A Practical Controller to Implement Pontryagin's Principle

A literature survey did not reveal a paper in which the author had applied optimal analogue control to an actual plant and fully observed the improvements. In the main, actual application, not being the theme of the majority of papers, was glossed over by such statements as, "the control strategy when applied to a particular plant produced an improved output", or that "the evolved optimal strategy may be exerted via a computer". This latter statement may be perfectly satisfactory when controlling very large plants or processes, but may prove uneconomic for the control of small individual plants.

This section of the research formulates a controller which may be used for the finite time interval and forms the basic element for the control of an actual plant (section 5), in the infinite time interval.

Error in the computation of the actual plants transfer function would be reflected in the model. Although an adjoint system, according to Pontryagin's Maximum Principle, may be obtained to optimise the model, it is very unlikely that the same adjoint system would optimise the actual plant. This is because the co-state vectors and at least one of their initial conditions are totally dependent upon the parameters of the model. Furthermore, as it was observed that the adjoint system was extremely sensitive to variation or error in the initial conditions of the co-state vectors, it is feasible that instability would occur before the termination of the optimising period.

To obtain a controller which would optimise the model, the adjoint system was required to be replaced by a system which was inherently stable and exhibited a similar transfer function. Such a system was realised by replacing the adjoint system with a function generator (fig. 3.1), constructed to reproduce the optimum plant input (output of adjoint system, m^o) for an input of system error (input to adjoint system, ε).



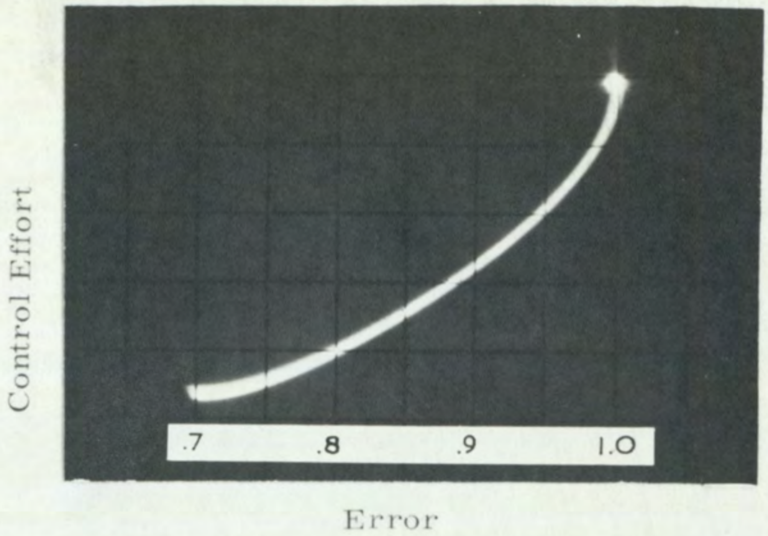
Optimal Control with Function Generator (F.G.)

Fig. 3.1.

For the original plant considered $\dfrac{1}{s(s + 2)}$, a curve of error v m^o was taken from an analogue simulation for an optimum period of one second (fig. 3.2). The resulting curve was reproduced on an analogue computer function generator (fig. 3.3) and used on closed loop

FIG. 3.2.

Curve of control effort (.5v/cm) against system error (.05v/cm) (from analogue computer for the model $\frac{1}{S(S+2)}$)



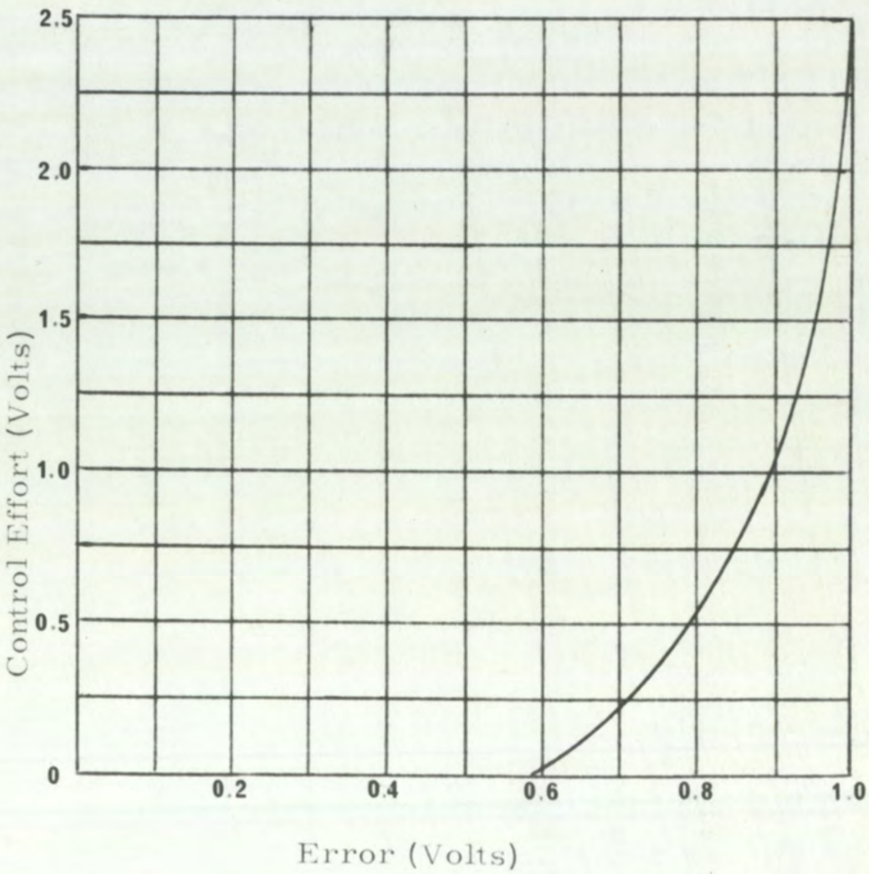


FIG. 3.3.

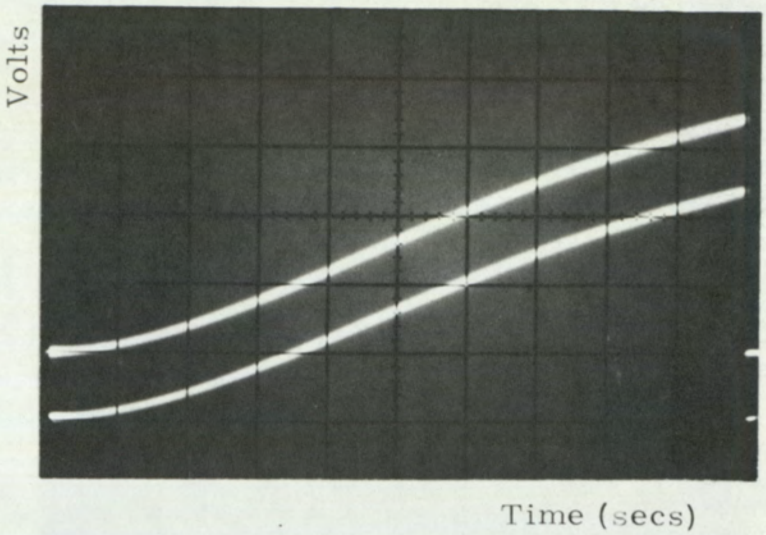
Actual function generator curve for optimal control of the model $\frac{1}{S(S+2)}$

Upper trace (origin displaced) : output with function generator control.

Lower trace: output with adjoint system control.

0. 1 volts/cm

0. 1 secs/cm.



Upper trace (origin displaced) : Index with function generator control.

Lower trace : Index with adjoint system control.

0. 2 volts/cm.

0. 1 secs/cm.

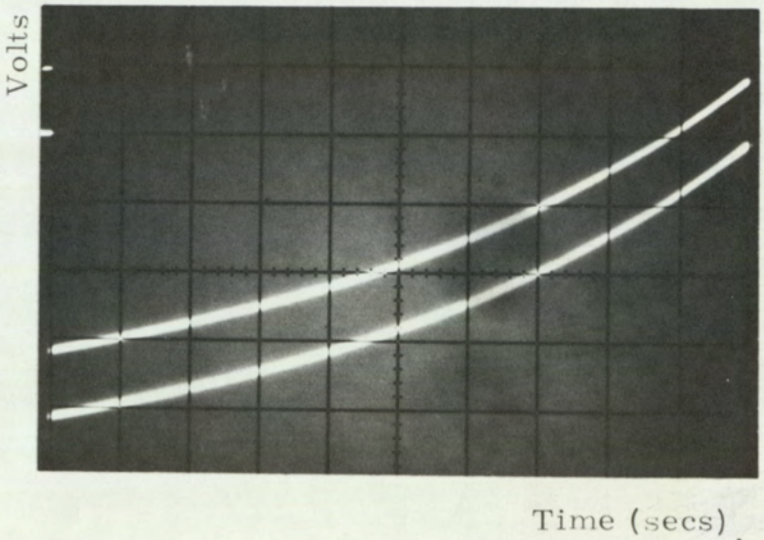


FIG. 3. 4.

Comparison of function generator and adjoint system control for the finite time interval.

Plant open loop transfer function $\frac{1}{S(S+2)}$

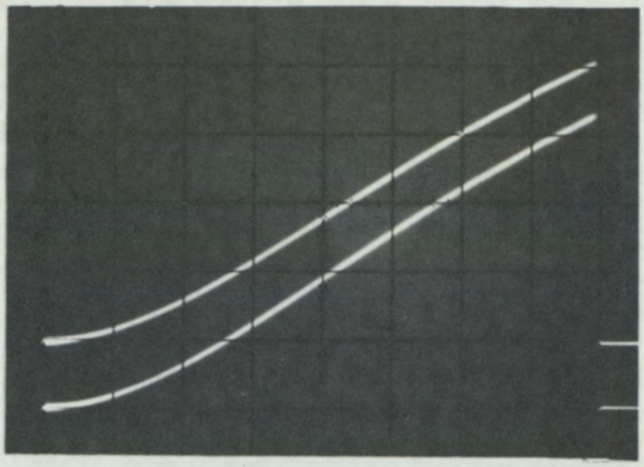
Upper trace (origin displaced): output with adjoint system control.

Lower trace : output with function generator control

0.05 secs/cm.

0.2 Volts/cm.

Volts



Time (secs).

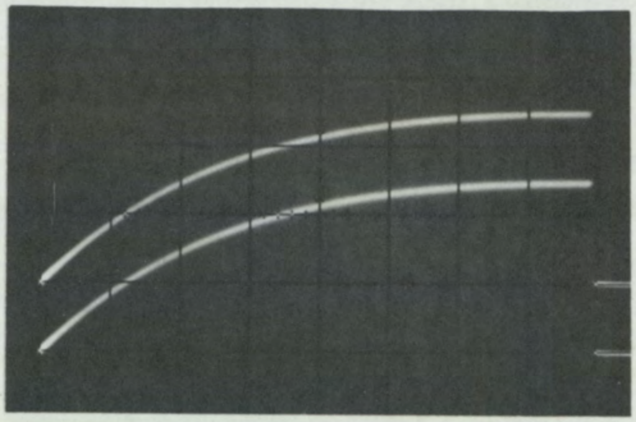
Upper trace (origin displaced): index with adjoint system control.

Lower trace : index with function generator control.

0.05 Secs/cm.

0.1 Volts /cm.

Volts



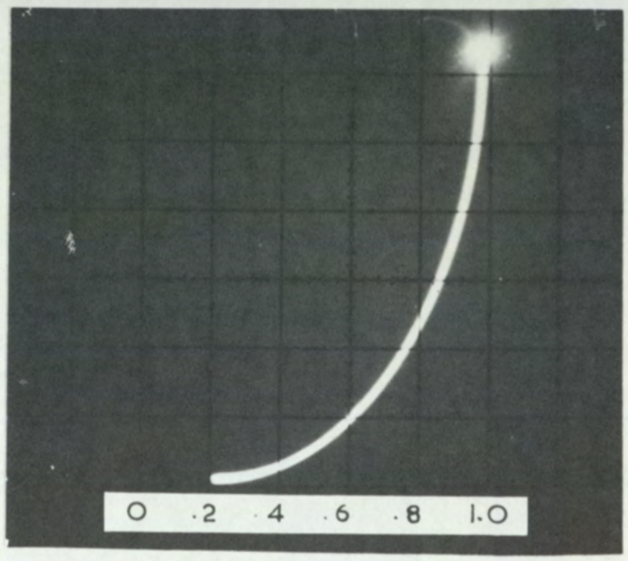
Time (secs)

Control effort 0.5 volt/cm

error 0.2 volt/cm.

(function generator curve).

Control Effect (Volts)



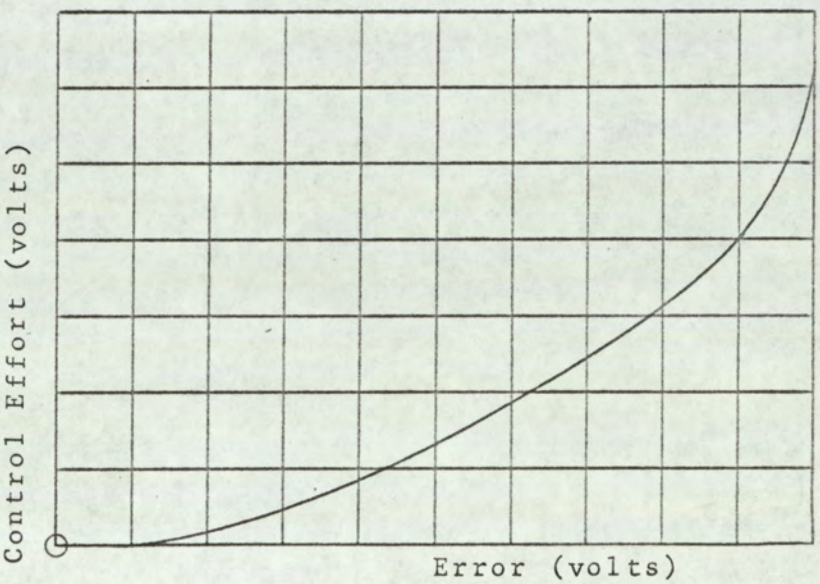
Error (Volts)

FIG. 3.5.

Comparison of function generator and adjoint system control for the finite time interval. Plant open loop transfer function $\frac{10}{S(S+2)}$

to control the model. The resulting model output and value of index showed negligible variation from that obtained when control was effected via the Pontryagin adjoint system (fig. 3.4). Similar results were obtained for plants which were N.O.P. (fig. 3.5)

The curve of error $\vee$ m^o for N.M.P. for optimising periods sufficient to produce plant outputs with zero error were, in the main, observed to approach zero m^o asymplotically with reference to the error axis (fig. 3.6). This virtually represented a dead zone so that once the



Typical Function Generator Curve for N.M.P. to attain Zero Error

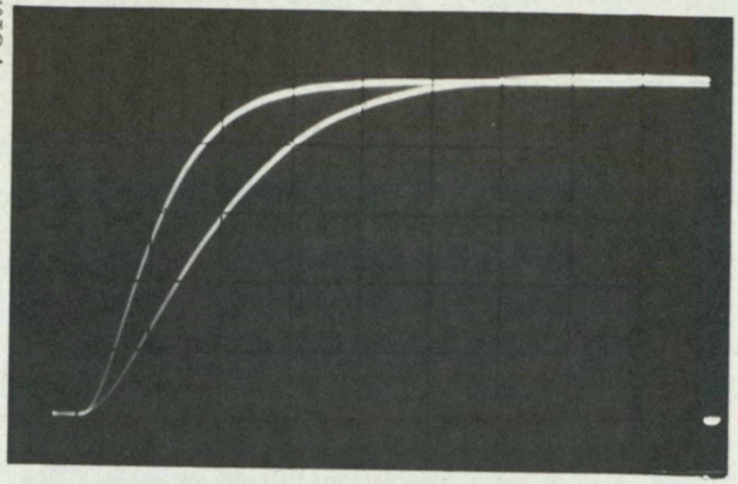
Fig. 3.6

vectors had attained zero magnitude, small variations in error would not produce change in system output. Furthermore, for the same class of systems, the output tended towards its desired value with zero slope at the termination of the optimising interval; the second state vector also tending to zero magnitude. This type of controller may therefore be used in the infinite interval case (fig. 3.7) as at the termination of the optimising interval the system transients are also terminated. In general, however, the output of the optimum plant did not tend towards its desired value with zero slope (section 2.3) and thus, using the function generator described, an optimum output would be obtained for the designed optimising interval, after which large overshoots and an excessive value of performance index would be obtained.

At this stage, the function generator approach was not used to control an actual plant as it could not be used on the general system in the infinite interval case. Furthermore, it is very unlikely, due to the apparent dead zone at the origin, that it could be used successfully, for the infinite time interval in the form described, even for an N.M.P.

Plant output with and without infinite interval control. Plant open loop transfer function;

$$\frac{1}{S(S+2)}$$



Upper trace (origin displaced) : Index with adjoint system control.

Lower trace : Index with infinite interval control.

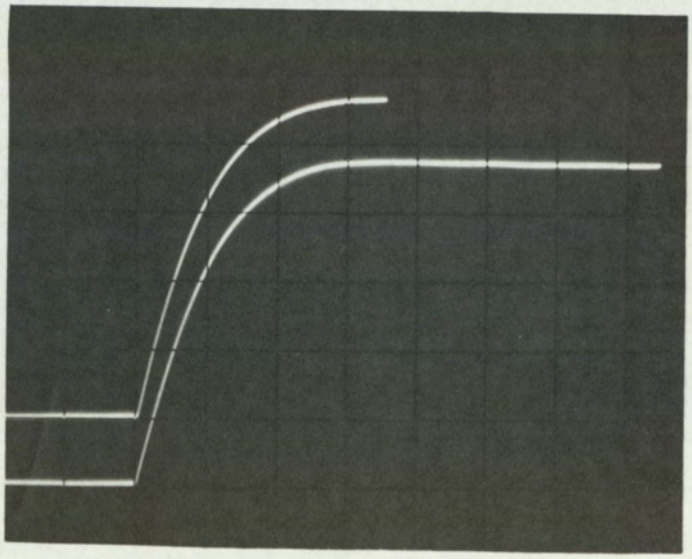


FIG. 3. 7.

4. Initial conditions for adjoint system to operate for the infinite time interval.

It has been demonstrated that Pontryagin's Maximum Principle cannot be directly applied to optimise on actual plant. With the replacement of the adjoint system by a function generator however, a practical controller was obtained for the finite time interval. Such a controller for N.M.P.'s was also demonstrated to be applicable to the infinite time interval. It was therefore considered relevant that further research, based on the mathematics concerning the adjoint system of Pontryagin's Maximum Principle, should be carried out to present a general and direct method of obtaining the initial conditions of the co-state vectors to present an adjoint system capable of infinite time interval control. Such a method would be required which did not necessitate the solution of a two point boundary value problem, render the resulting optimal strategy applicable via a function generator technique and produce trajectories and values of index comparable with those obtained when other optimising techniques, (i.e. Dynamic Programming), were implemented.

4.1 Determination of required plant output.

A block diagram for the optimal plant $\frac{Y}{S(S+a)}$ is shown in figure 4.1

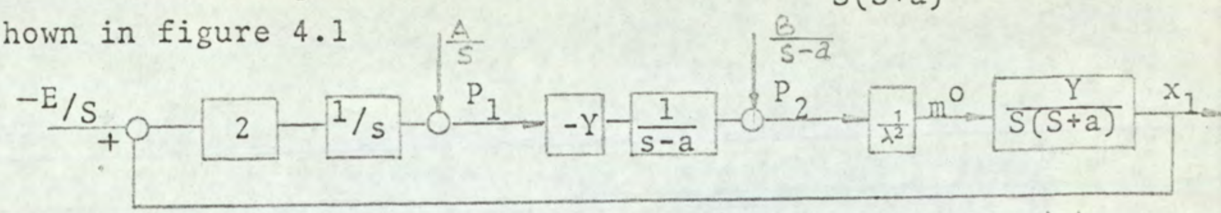


FIG 4.1

The plant output is given by

$$\left[\left[(x_1-{}^{E}/s)(2)({}^{1}/s) + {}^{A}/s\right](-Y)\left(\frac{1}{s-a}\right) + {}^{B}/s-a\right]\frac{1}{2\lambda}\cdot\left(\frac{Y}{S(S+a)}\right) = x_1$$

$$\therefore \quad x_1(S) = \frac{Y}{2\lambda S^2+2\lambda aS}\left[\frac{-2x_1Y}{S^2-Sa} + \frac{2EY}{S^3-S^2a} - \frac{AY}{S^2-aS} + \frac{B}{s-a}\right]$$

$$= \frac{2\lambda S^4-2\lambda a^2S^2}{2\lambda S^4-2\lambda a^2S^2+2Y^2}\left[\frac{2EY^2}{2\lambda S^5-2\lambda a^2S^3} - \frac{AY^2}{2\lambda S^4-2\lambda a^2S^2} + \frac{BY}{2\lambda S^3-2\lambda a^2S}\right]$$

$$\therefore \quad x_1(s) = \frac{2EY^2-AY^2S+BYS^2}{S(2\lambda S^4-2\lambda a^2S^2+2Y^2)} \quad . \; . \; . \; . \; . \; . \; . \; . \; 4.1.1.$$

The denominator of equation 4.1.1 may be written

$$2\lambda S\,(S^4-a^2S^2+ Y^2/\lambda) \quad . \; . \; . \; . \; . \; 4.1.2.$$

Equation 4.1.2. has five factors; one being the laplace transform S, (produced by the step input ($^E/S$) to the system), and two quadratic factors of the form $(S^2-\alpha)$, $(S^2-\beta)$ where α and β are given by

$$\frac{a^2}{2} \pm \sqrt{\frac{a^4}{4} - \frac{Y^2}{\lambda}} \quad . \; . \; . \; . \; . \; . \; . \; 4.1.3.$$

If $\frac{Y^2}{\lambda} > \frac{a^4}{4}$ then the factors of equation 4.1.2. (neglecting the constant multiplier2λ), may be written.

$$(S^2-c\pm jd)$$

The factors may be isolated as

$$(S-h+jf)(S+h-jf)(S-h-jf)(S+h+jf) \; . \; . \; . \; . \; . \; . \; 4.1.4.$$

Factors (S-h+jf) and (S-h-jf) will give rise to an output in the time domain of the form

$$Ke^{ht}\left[\,\mathrm{Sin}ft+(90-\alpha)\right] \quad . \; . \; . \; . \; . \; . \; 4.1.5.$$

If $\frac{a^4}{4} > \frac{Y^2}{\lambda}$ then the factors of equation 4.1.2. may be written (again neglecting the constant multiplier2λ),

$$(S^2-c')(S^2-d')$$

The resulting roots being isolated as

$(S-f')(S+f')(S-h')(S+h')$ . . . . . . 4.1.6.

The factors $(S-f')$ and $(S-h')$ will produce factors in the time domain of the form

$K_1e^{f't}$ and $K_2e^{h't}$ . . . . . . 4.1.7.

Consider the numerator of the plant output (equation 4.1.1.)

$$\text{Numerator} = BYS^2-AYS+2EY^2$$

$$= BY\left(S^2-\frac{AYS}{B}+\frac{2EY}{B}\right) \quad \ldots\ldots\ldots 4.1.8.$$

Roots of equation 4.1.8, neglecting the multiplier BY, will be given by:

$$\frac{AY}{2B} \pm \sqrt{\frac{A^2Y^2}{4B^2} - \frac{2EY}{B}} \quad \ldots\ldots 4.1.9$$

If $\frac{2EY}{B} > \frac{A^2Y^2}{4B^2}$ then the factors of 4.1.8. may be written

$$(S-h''+jf'')(S-h''-jf'') \quad \ldots\ldots 4.1.10.$$

If $\frac{A^2Y^2}{4B^2} > \frac{2EY}{B}$ the factors of 4.1.8. may be written

$$(S-h''')(S-f''') \quad \ldots\ldots 4.1.11.$$

As complex poles will give rise to complex zero's the optimum plant output may be written by combining equations 4.1.11 and 4.1.6, 4.1.10 and 4.1.4, not forgetting the respective constant multipliers.

$$x_1(S) = \frac{BY(S-h''+jf'')(S-h''-jf'')}{2\lambda S(S-h+jf)(S+h-jf)(S-h-jf)(S+h+jf)} \quad \ldots 4.1.12$$

or $$x_1(S) = \frac{BY(S-h''')(S-f''')}{2\lambda S(S-f')(S+f')(S-h')(S+h')} \quad \ldots 4.1.13$$

Equations 4.1.5. and 4.1.7 demonstrate that the optimum plant output will always possess two poles which will produce positive exponentials in the time domain. Therefore, for the plant to be infinitely stable the initial conditions on the co-state vectors (A and B), which only appear in the numerator (equation 4.1.1.), must be arranged so that zero's are generated to exactly cancel the positive poles. This implies the following identities:

$$h=h'' \; ; \; f=f'' \quad . \; . \; . \; . \; . \; . \; . \; . \; 4.1.14$$

$$h'=h''' \; ; \; f'=f''' \quad . \; . \; . \; . \; . \; . \; . \; . \; 4.1.15.$$

which results in the following outputs:

$$x_1(S) = \frac{BY}{2\lambda S(S+h-jf)(S+h+jf)} \quad . \; . \; . \; . \; . \; 4.1.16.$$

$$\text{or } x_1(S) = \frac{BY}{2\lambda S.(S+f')(S+h')} \quad . \; . \; . \; . \; . \; 4.1.17.$$

The stable output as postulated by equation 4.1.17 will be over-damped and hence would approach its desired value with zero slope and no overshoot. Equation 4.1.16 postulates an output which is under damped and thus overshoot may be exhibited. (This overshoot was observed to always be small,(less than 4.3% in the worst case, Section 4.4.), and was counteracted in the actual controller, section 5.).

4.2 Determination of adjoint system initial Conditions.

The values of the co-state vector initial conditions were obtained by a method of comparisons related to the identities of equations 4.1.14 and 4.1.15 (section 4.2(a)). This method however, was not general as initial conditions could not be obtained to satisfy both of the equations 4.1.16 and 4.1.17, and would have been very difficult to apply to higher order systems. A second method was therefore evolved to produce a completely general approach (section 4.2(b).)

4.2(a) Initial conditions by method of identities

The identities of 4.1.14. may be expressed:

$$(s-h+jf)(S+h-jf)=(S-h'' +jf'')(S+h''-jf'') \quad \ldots\ldots \; 4.2(a).1.$$

where (S-h+jf)(S+h-jf) are the factors of (S^2-c+jd)

Substituting equations 4.1.3 and 4.1.9 into equation 4.2(a).1. yields:

$$-\frac{a^2}{2}+\sqrt{\frac{a^4}{4}-\frac{Y^2}{\lambda}} = \left(\frac{-AY}{2B}+\sqrt{\frac{A^2Y^2}{4B^2}-\frac{2EY}{B}}\right)\left(\frac{AY}{2B}-\sqrt{\frac{A^2Y^2}{4B^2}-\frac{2EY}{B}}\right. \quad \ldots \; 4.2(a).2$$

where each square root term is an imaginary quantity.
Equating real parts

$$-\frac{a^2}{2} = -\frac{A^2Y^2}{4B^2} - \frac{A^2Y^2}{4B^2} + \frac{2EY}{B}$$

$$\therefore A = \sqrt{\frac{a^2B^2}{Y^2} + \frac{4EB}{Y}} \quad \ldots\ldots \; 4.2(a).3.$$

Equating imaginary parts

$$\sqrt{\frac{a^4}{4} - \frac{Y^2}{\lambda}} = \frac{AY}{B}\sqrt{\frac{A^2Y^2}{4B^2} - \frac{2EY}{B}}$$

$$\therefore \frac{a^4}{4} - \frac{Y^2}{\lambda} = \frac{A^4Y^4}{4B^4} - \frac{2EY^3A^2}{B^3}$$

Substituting for A^4 and A^2 from equation 4.2(a).3 produces

$$\frac{Y^2}{\lambda} = \frac{4E^2Y^2}{B^2}$$

$$\therefore B = 2E\sqrt{\lambda} \quad \ldots\ldots \; 4.2(a).4.$$

The identities of 4.1.15 may be interpreted as

$$(S-f')(S+f')(S-h')(S+h') = (S-h''')(S-f''')(S+h''')(S+f''') \quad \ldots 4.2(a).5$$

Substitution of equations 4.1.3 and 4.1.9. into 4.2(a).5. yields:

$$\left(\frac{a^2}{2}\right)^2 - \left(\sqrt{\frac{a^4}{4} + \frac{Y^2}{\lambda}}\right)^2 = \left[\left(\frac{AY}{2B}\right)^2 - \left(\sqrt{\frac{A^2Y^2}{4B^2} - \frac{2EY}{B}}\right)^2\right]^2$$

$$\therefore \frac{Y^2}{\lambda} = \frac{4E^2Y^2}{B^2}$$

$$\therefore B = 2E\sqrt{\lambda} \quad \ldots . 4.2(a).6$$

Identical equations for the initial condition B were obtained (equations 4.2(a).6 and 4), for both forms of the optimal system output as represented in equations 4.1.16 and 4.1.17. The method employed however did not facilitate the evaluation of the initial condition A for the system output where all roots were real.

4.2(b). General method for calculation of the co-state vector initial conditions.

As discussed in section 4.1. the numerator of the optimum plant output is a quadratic in S and for a stable system the roots must cancel with those of the denominator which would give rise to positive exponentials in the time domain.

When complex roots are present the optimum stable plant output is that of equation 4.1.12 namely:

$$x_1(S) = \frac{BY(S-h+jf)(S-h-jf)}{2\lambda S(S-h+jf)(S+h-jf)(S-h-jf)(S+h+jf)}$$

which may be expanded and written:

$$x_1(S) = \frac{BY(S^2-2hS+h^2+f^2)}{2\lambda S(S^4-(2h^2-2f^2)S^2+2h^2f^2+h^4+f^4)} \quad . . . \; 4.2(b).1$$

By substituting x = 2h

and $Z = h^2+f^2$

equation 4.2(b).1 may be written

$$x_1(S) = \frac{BY(S^2-XS+Z)}{S.2.\lambda(S^4-(X^2-2Z)S^2+Z^2)} \quad . . . . . . \; 4.2(b).2$$

When all roots are real the optimum stable plant output will be of the form given in equation 4.1.13 namely

$$x_1(S) = \frac{BY(S-h')(S-f')}{2\lambda S(S-f')(s+f')(S-h')(S+h')} \quad . . . \; 4.2(b).3$$

which may be expanded and written

$$x_1(S) = \frac{BY(S^2-S(h+f)+hf)}{S.2\lambda(S^4-S^2(h^2+f^2)+h^2f^2)} \quad . . . . \; 4.2(b).4$$

By substituting: $Z' = hf$

and $X' = h+f$

equation 2.2(b).4 may be written

$$x_1(S) = \frac{BY(S^2-X'S+Z')}{2\lambda.S(S^4-(X'^2-2Z')+Z'^2)} \quad . . . . \; 4.2(b).5$$

As the ratio between numerator and denominator are identical for equations 4.2(b).5 and 4.2(b).2 only one equation need be considered for both real and complex poles in the optimum plant output.

The general form of the optimum plant output (equation 4.1.1), may be written

$$\frac{BY(S^2-\frac{AY}{B}S+\frac{2EY}{B})}{S.2.\lambda(S^4-a^2S^2+Y^2/\lambda)} \quad . . . . . . 4.2(b).6.$$

Equating coefficients of equation 4.2(b).2 with those of equation 4.2(b).6

$$X = \frac{AY}{B} \quad . . . . 4.2(b).7$$

$$Z = \frac{2EY}{B} \quad . . . . . . 4.2(b).8$$

$$(X^2-2Z)=a^2 \quad . . . . 4.2(b).9$$

$$Z^2 = \frac{Y^2}{\lambda} \quad . . . . 4.2(b).10$$

From equations 4.2(b).8 and 10

$$\frac{4E^2Y^2}{B^2} = \frac{Y^2}{\lambda}$$

$$\therefore B = 2E\sqrt{\lambda} \quad . . . . . 4.2(b).11$$

From equations 4.2(b).7, 8 and 9.

$$a^2 = \frac{A^2Y^2}{B^2} - \frac{4EY}{B}$$

$$\therefore A = \sqrt{\frac{a^2B^2}{Y^2} + \frac{4EB}{Y}} \quad . . . . 4.2(b).12$$

Substituting equation 4.2(b).11 into 4.2(b).12.

$$A=E.\sqrt{\frac{4a^2\lambda}{Y^2} + \frac{4\sqrt{\lambda}}{Y}} \quad . . . . . . 4.2(b).13.$$

Equations 4.2(b).11 and 13 give the values of the initial conditions, B and A respectively, to produce the stable plant output of equations 4.1.16 and 17. If the plant had been conceived in any other form (elements of the mathematical model rearranged), then a different equation for A would have been evolved. The equation for B however, would be expected to remain the same for as observed in section 2 and verified by equation 4.2(b).11, B is independant of the plant parameters and therefore independant of the plant model. This may be demonstrated by considering the initial plant with the rearranged model of fig. 4.2(b).1. The plant state equations may now be written: $\dot{x}_1 = Yx_2 - ax_1$

$$\dot{x}_2 = m$$

$$\dot{x}_3 = (E-x_1)^2 + \lambda m^2$$

The Hamiltonian may be written

$$H = p_1(Yx_2 - ax_1) + p_2 m - \left[(E-x_1)^2 + \lambda m^2\right]$$

$$\frac{\partial H}{\delta m} = p_2 - 2\lambda m$$

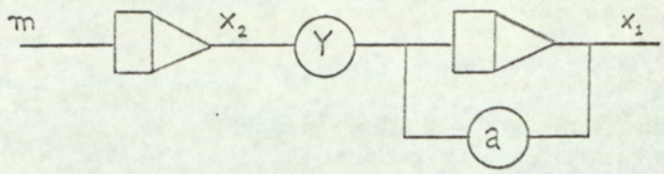
$\therefore$ for maximum H, $m^o = \frac{P_2}{2\lambda}$

$$\dot{p}_1 = ap_1 + 2(x_1 - E)$$

and $\dot{p}_2 = -Yp_1$

From the block diagram of the complete optimal system (fig. 4.2(b).2) the optimum output may be calculated as:

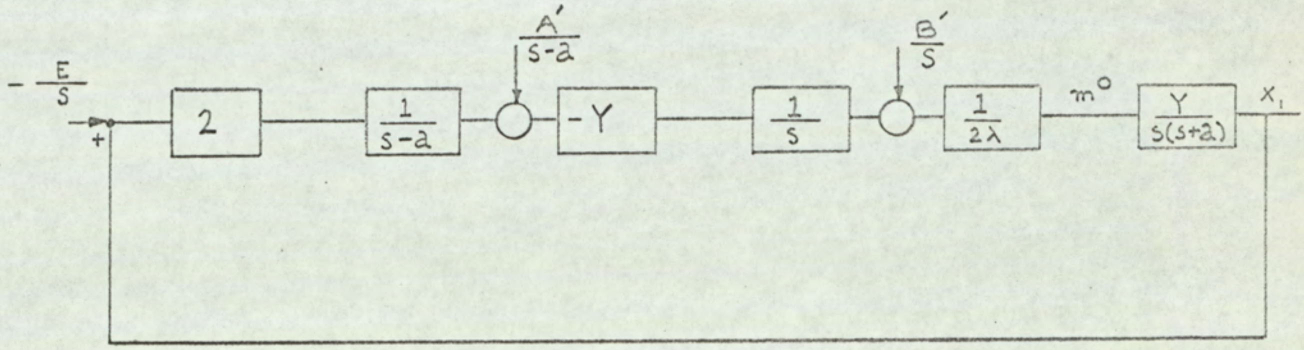
$$x_1(S) = \frac{B'Y(S^2 - S(a + \frac{A'Y}{B'}) + \frac{2EY}{B'})}{2.\lambda.S(S^4 - S^2a^2 + Y^2/\lambda)} \quad \ldots\ldots \quad 4.2(b).14.$$



Re arranged Model of

$$\frac{y}{S(S + a)}$$

Fig. 4.2(b).1



Block Diagram of Optimum System

Fig. 4. 2(b).2

This equation is identical to 4.2(b).6 except for the addition of ($-B'YaS$) in the numerator. The required ratio between numerator and denominator will therefore, to comply with the stable outputs of equations 4.1.16 and 4.1.17, be that of equation 4.2(b).2 with:

$$Z = \frac{2EY}{B'} \quad \ldots\ldots\ 4.2(b).15$$

$$X = a + \frac{A'Y}{B'} \quad \ldots\ 4.2(b).16$$

$$(X^2 - 2Z) = a^2 \quad \ldots\ 4.2(b).17$$

$$Z^2 = \frac{Y^2}{\lambda} \quad \ldots\ 4.2(b).18$$

From equations 4.2(b).15 and 18

$$B' = 2E\sqrt{\lambda} \quad \ldots\ 4.2(b).19$$

$$\therefore \quad B' = B$$

From equations 4.2(b).15, 16 and 17

$$a^2 = a^2 + \frac{A'^2Y^2}{B^2} - \frac{4EY}{B} + \frac{2A'Ya}{B}$$

$$\therefore \quad A'^2 + \frac{2aBA'}{Y} - \frac{4EB}{Y} = 0$$

$$\therefore \quad A' = -\frac{aB}{Y} \pm \sqrt{\frac{a^2B^2}{Y} + \frac{4EB}{Y}} \quad \ldots\ldots\ 4.2(b).20$$

Taking the required positive solution

$$A' = -\frac{Ba}{Y} + \sqrt{\frac{a^2B^2}{Y^2} + \frac{4EB}{Y}} \quad \ldots\ldots\ 4.2(b).21$$

or $\quad A' = A - \frac{Ba}{Y}$

This same result may also be obtained directly from equations 4.2(b).14 and 4.2(b).6 for, as the plant output must be the same for both models:

$$\frac{AY}{B} = a + \frac{A'Y}{B}$$

$$\therefore A' = A - \frac{aB}{Y}$$

4.3 Determination of initial co-state vector value from the performance index.

The initial condition B, (equation 4.2(b).11), is independant of the plant parameters and only dependant upon the elements of the performance index. It should therefore be possible to derive a similar equation from the performance index.

The performance index may be represented as:

$$\int_{t=0}^{\infty} ((\text{error})^2 + \lambda m^2)\, dt.$$

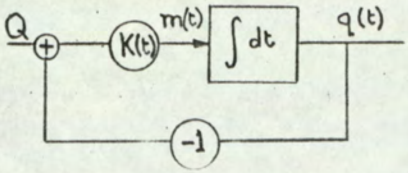
At time t=0, $x_1=0$, $m=\frac{B}{2\lambda}$ and error = -E (fig. 4.2(b).2), enabling the integrand of the index to be written

$$E^2 + \frac{B^2}{4\lambda} \quad . \ . \ . \ . \ . \ . \ 4.3.1$$

Substituting for $B = 2E\sqrt{\lambda}$ equation 4.3.1. may be written

$$E^2 + E^2 = 2E^2 \quad . \ . \ . \ . \ . \ . \ 4.3.2$$

Equation 4.3.2. demonstrates that for the index to be a minimum the value of the integrand at time t = 0, will always be equal to $2E^2$. A similar result may be inferred from an example worked by Merriam (ref.20), where the system of fig. 4.3.1was optimised according to the mathematics of the Calculus of Variations.



Optimal System
FIG. 4.3.1

The performance index optimised being:

$$\int_t^T \left[(Q - q(\sigma))^2 + \rho m^2(\sigma) \right] d\sigma$$

Merriam derived the optimum gain K(t) as:

$$K(t) = \frac{1}{\rho^{1/2}} \cdot \text{Coth} \frac{T-t}{\rho^{1/2}} \quad . . . . 4.3.3$$

and the optimum control signal as:

$$m^o(t) = K(t).(Q-q(t)) \quad . . . . 4.3.4.$$

For an optimizing interval ranging from zero time to infinity, K(t) at time t=0, may be written

$$K(0) = \frac{1}{\rho^{1/2}} \text{ Coth } \frac{\infty}{\rho^{1/2}}$$

$$= \frac{1}{\rho^{1/2}} \quad . . . . 4.3.5.$$

Also at time t=0, q(t)=0, and substitution of equation 4.3.5. into 4.3.4. produces

$$m^o(0) = \frac{Q}{\rho^{1/2}} \quad . . . . . . 4.3.6.$$

The integrand of the performance index may now be written

$$Q^2 + \rho \frac{Q^2}{\rho} = 2Q^2 \quad . . . 4.3.7.$$

Equation 4.3.7. again demonstrates that for the infinite interval, in order that the performance index can be minimized, the value of the integrand at time t=0, will be equal to twice the squared value of the desired output.

It is thus evident that the initial value of the co-state vector directly responsible for the control effort (i.e. $p_2(0)$ or B), may be derived from the integrand of the performance index at time t=0, by equating it to $2E^2$.

i.e. $\text{error}^2 + \lambda m^2 = 2E^2$ at time t=0

$$\therefore \quad E^2 + \frac{\lambda B^2}{4\lambda^2} = 2E^2$$

$$\therefore \quad B = 2E\sqrt{\lambda}$$

4.4 Maximum per-centage overshoot for oscillatory output.

The optimal plant output may be oscillatory as depicted by equation 4.1.16, and thus overshoots may result. The maximum value of these overshoots may be calculated as follows:
The required output is shown by equation 4.2(b).5 to be of the form

$$x_1(S) = \frac{BY(S^2-XS+Z)}{2.\lambda S.(S^4-(X^2-2Z)S^2+Z^2}$$

which may be written

$$x_1(S) = \frac{BY}{2.\lambda.S.(S^2+XS+Z)} \quad \ldots\ldots\ 4.4.1.$$

The closed loop transfer function ($\frac{x_1(S)}{E(S)}$), may be written

$$\frac{x_1(S)}{E(S)} = \frac{\frac{Y}{\sqrt{\lambda}}}{(S^2+XS+Z)} \quad \ldots\ldots\ 4.4.2$$

For a second order system with one time constants, $x = \frac{AY}{B}$, $Z = \frac{2EY}{B}$ and $B = 2E\sqrt{\lambda}$ Equation 4.4.2. may therefore be written,

$$\frac{x_1(S)}{E(S)} = \frac{\frac{Y}{\sqrt{\lambda}}}{S^2+\frac{AY}{B}.S+\frac{Y}{\lambda}} \quad \ldots\ldots\ 4.4.3$$

Therefore equation 4.4.3. may be written

$$\frac{x_1(S)}{E(S)} = \frac{W_n^2}{S^2+2\eta W_n S+W_n^2}$$

$$\therefore\ x_1(S) = \frac{E(Wn^2)}{S(S^2+2\eta WnS+Wn^2)} \quad \ldots\ldots\ 4.4.4.$$

It is shown in appendix 2, that the maximum overshoot, in the time domain, of equation 4.4.4. will be:

Maximum per cent overshoot = exp. $\left(-\frac{\eta\pi}{\sqrt{1-\eta^2}}\right)$ .....4.4.5.

Equation 2.1.16 gives form of equation 4.4.1 i.e.

$$\frac{BY}{2\lambda S(S+h-jf)(S+hjf)} = \frac{BY}{2.\lambda.S.(S^2+XS+Z)} = \frac{BY}{2.\lambda.S(S^2+\frac{AYS}{B}+\frac{2EY}{B})}$$

$$\therefore (S^2+2hS+h^2+f^2) = (S^2+\frac{AYS}{B} + \frac{2EY}{B})$$

equating coefficients

$$h = \frac{AY}{2B} \quad .... \; 4.4.6.$$

$$f = \sqrt{\frac{2EY}{B} - \frac{A^2Y^2}{4B^2}} \quad .... \; 4.4.7.$$

From equation 4.4.6. and 4.2(b).12.

$$h^2 = \frac{a^2}{4} + \frac{YE}{B} \quad .... \; 4.4.8.$$

From equation 4.4.7. and 4.2(b).12.

$$f^2 = \frac{2EY}{B} - \frac{a^2}{4} - \frac{YE}{B}$$

$$= \frac{EY}{B} - \frac{a^2}{4} \quad .... \; 4.4.9.$$

Therefore $h > f$

The conjugate complex poles may be represented in the complex plane as shown in fig. 4.4.1. where $\phi = \cos^{-1}\eta$ (η = damping factor).

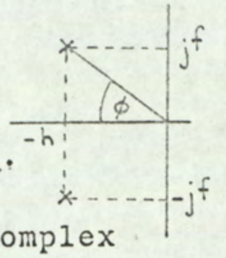


FIG. 4.4.1.

Location of complex poles

As $h > f$, ϕ will always be less than 45 degrees and therefore η will always be greater than 0.7071.

From equation 4.4.5. the maximum possible overshoot will be:

$$\exp\left(\frac{-0.7071.\pi}{\sqrt{1-(0.7071)^2}}\right)$$

$$= \exp(-\pi) = 0.043$$

or 4.3%

4.5 Initial co-state vector values for a second order plant with two time constants.

The previous theory has been developed solely for a second order plant possessing one time constant. The initial conditions of the adjoint vectors are now considered for a plant which possesses two time constants. (i.e. a simple Ward Leonard control system). The state equations for such a plant fig. 4.5.1. may be written.

$$\dot{x}_1 = Yx_2 - bx_1 \quad . \; . \; . \; . \; . \; . \; 4.5.1.$$

$$\dot{x}_2 = m - ax_2 \quad . \; . \; . \; . \; . \; . \; . \; 4.5.2.$$

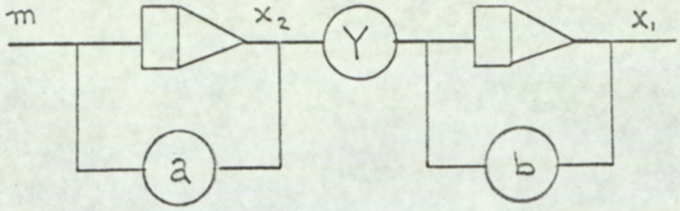
$$\dot{x}_3 = (E-x_1)^2 + \lambda m^2.$$

The Hamiltonian may be written

$$H = p_1(Yx_2 - bx_1) + p_2(m - ax_2) - ((E-x_1)^2 + \lambda m^2)$$

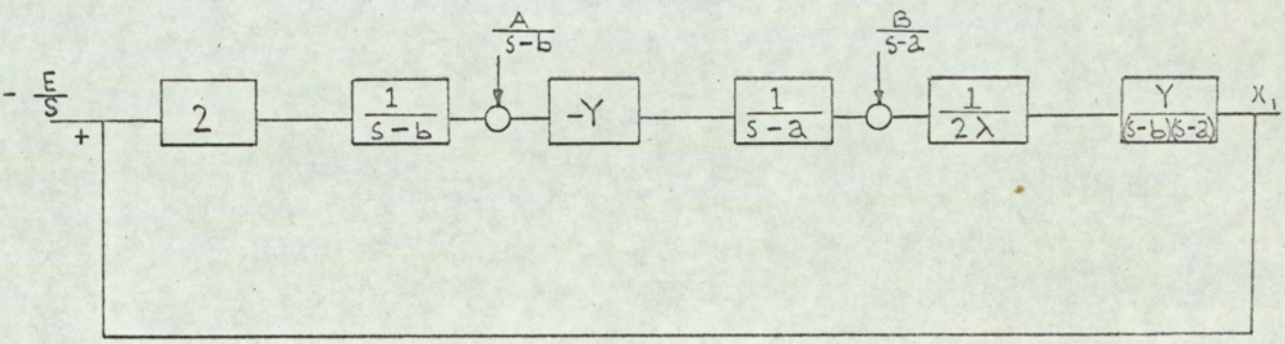
$$\frac{\partial H}{\partial m} = p_2 - 2\lambda m$$

Therefore for maximum H: $m^o = \dfrac{p_2}{2\lambda}$ . . . . . .4.5.3.



Model of Plant $\frac{y}{(S + a)(S + b)}$

Fig. 4.5.1



Block Diagram of Optimum Plant

Fig. 4.5.2

$\dot{p}_1 = 2(x_1-E)+bp_1$ . . . . 4.5.4.

and $\dot{p}_2=ap_2-Yp_1$ . . . . 4.5.5.

Equations 4.5.1 to 5 enables the complete block diagrams for the optimum system to be constructed (fig. 4.5.2.) from which the optimum plant output may be obtained:

$$x_1(S) = \frac{BY\left[S^2-S(b+\frac{AY}{B})+2\frac{EY}{B}\right]}{2.\lambda.S(S^4-S^2(b^2+a^2)+a^2b^2+\frac{Y^2}{\lambda})} \quad . . . . . . 4.5.6.$$

As the form of equation 4.5.6. is the same as that obtained for the second order plant with one time constant the required ratio between numerator and denominator for a stable system will be:

$$\frac{BY(S^2-XS+Z)}{2.\lambda.S(S^4-(X^2-2Z)S^2+Z^2)} \quad . . . . 4.5.7.$$

Comparing coefficients of equation 4.5.6. with those of equation 4.5.7.

$$X = b+\frac{AY}{B} \quad . . . . 4.5.8.$$

$$Z = \frac{2EY}{B} \quad . . . . 4.5.9.$$

$$(X^2-2Z) = b^2+a^2 \quad . . . . 4.5.10.$$

$$Z^2=a^2b^2+\frac{Y^2}{\lambda} \quad . . . . 4.5.11.$$

From equations 4.5.9. and 11

$$\frac{4E2Y^2}{B^2} = a^2b^2 + \frac{Y^2}{\lambda}$$

$$\therefore \quad B = 2.E\sqrt{\frac{\lambda Y^2}{\lambda a^2b^2 + Y^2}} \quad \ldots\ldots\ldots \quad 4.5.12$$

From equations 4.5.8, 9 and 10

$$b^2 + a^2 = b^2 + \frac{A^2Y^2}{B^2} + \frac{2bAY}{B} - \frac{4EY}{B}$$

$$\therefore \quad A^2 + \frac{B2bA}{Y} - \frac{4EB}{Y} - \frac{a^2B^2}{Y^2} = 0$$

$$\therefore \quad A = -\frac{Bb}{Y} \pm \sqrt{\frac{Bb^2}{Y^2} + \frac{4EB}{Y} + \frac{aB^2}{Y^2}}$$

Taking the required positive value for A

$$A = -\frac{Bb}{Y} + \sqrt{\frac{B^2}{Y}(b^2 + a^2) + \frac{4EB}{Y}} \quad \ldots\ldots\ldots \quad 4.5.13.$$

Equations 4.5.13. and 4.5.12 become similar to the equations representing A and B, for the second order plant with one time constant, when a or b is made equal to zero depending upon the required form of the model.

4.6 Optimisation by formulae of a second order system with one time constant.

As demonstrated in section 2 the optimum plant trajectories could be generalised into two categories; each being descriptive of the plant output when operated with normal unity feedback (N.O.P. or N.M.P.), This generalisation was observed to be applicable to the equations evolved for the co-state vector initial values when compared with those obtained from the solution of the two point boundary value problem i.e. the value attained from the analogue computer for an optimising interval as long as possible without invoking instability. For plants which were initially N.M.P. the values differed by magnitudes in the order of 1%. This error was attributed to the analogue computer. Plants which were N.O.P. exhibited appreciable difference (up to +15% or +20%), and could not be accounted for by computing error.

A difference for the N.O.P. was expected as 'Pontryagins' technique did not allow steady state conditions to prevail at the termination of optimizing interval, which is a requirement for the infinite interval plant. As steady state conditions were attainable for N.M.P., the requirement for the infinite interval were met and therefore, no change in the initial conditions of the co-state vectors were required.

Calculation of the initial co-state vector value B, (initial condition on P_2 producing m^o), yielded the same result, for both N.O.P. and N.M.P., as that obtained from the solution of the two point boundary value problem. This was expected as this condition was totally independant of the plant parameters.

As the two initial conditions for the N.M.P. were identical to those obtained from the solution of the two point boundary value problem, the resulting trajectories and value of index were identical to those obtained via Pontryagin's Maximum Principle for the 'longest' time interval. The N.O.P. however, having a larger value of 'A' produced different vectors and larger values of index. The calculated values of 'A' producing co-state vectors and hence, control efforts which over-shoot. This was necessary to reduce the over-shoot of the output and attain steady state conditions at the instant all the co-state vectors reached zero magnitude. Actual plant trajectories, obtained when optimation was accomplished with the aid of equations 4.2(b).11 and 13, are represented in later sections. A comparison between the new trajectories and those obtained from Pontryagin's Maximum Principle would not be valid as the latter did not produce infinitly stable systems, or from N.O.P., produce outputs which possessed zero slope at the desired plant output value. It was for these reasons that for a N.O.P. the value of index and trajectories were compared with those obtained from a similar plant optimized according to the principle of Dynamic Programming (section 6).

5. Realisation of a Practical Controller

As mentioned in section 4.6, optmisation of an N.O.P., when the initial co-state vector values are calculated according to the formulae of section 4 , produce vectors which over-shoot. Typical outputs and control efforts are shown in fig. 5.1. The resulting function generator curve (m^ov error)

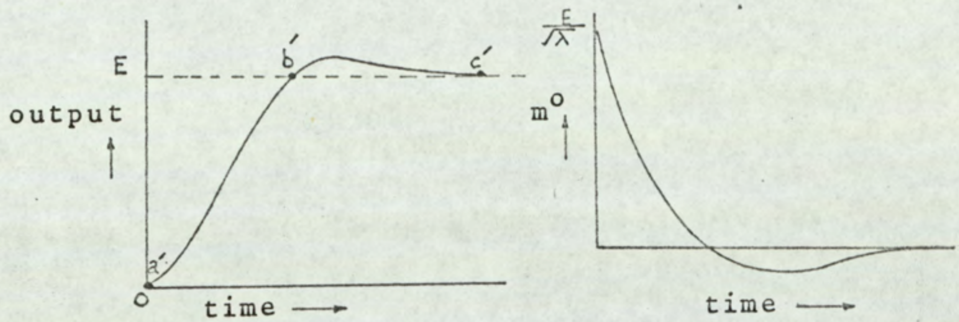


Fig. 5.1

Typical output and control effort for N.O.P.

would take the form shown in fig. 5.2. If this function

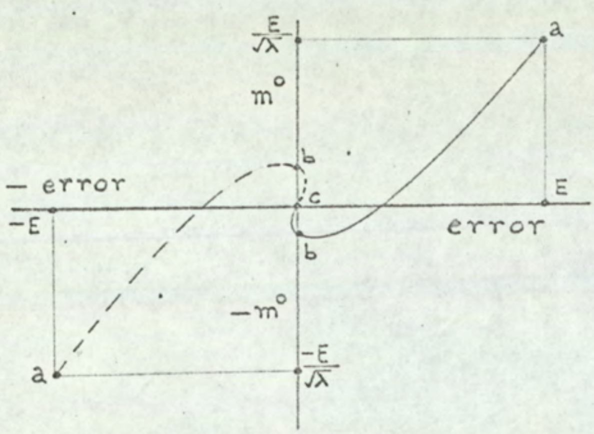


Fig. 5.2

Typical Function Generator Characteristic was used as the controlling element, the points a, b and c of fig. 5.2 would correspond to a', b' and c' on the output of fig. 5.1., i.e. at the instant of switching on, the plant output is assumed zero, and the control effort will be at its maximum value ($\frac{E}{\sqrt{\lambda}}$). The error of the system will then be decreased as the output increases producing a change in m^o according to the function of fig. 5.2. (The broken characteristic of fig. 5.2. is for switching off or applying a negative step input). When the error attains its first zero, point b (fig. 5.2), the output would have attained its desired value (point b', fig. 5.1). The portion of the characteristic b.c (fig. 5.2) produces the overshoot on the output b'c' (fig. 5.1).

The function generator curve of fig. 5.2 would be extremely difficult to simulate in its entirety. For error in the range 0 volts to E volts, however, the function is completely amenable to simulation. If a system was completely controlled by a function generator of this form (fig. 5.3), an identical output trajectory would be obtained, from

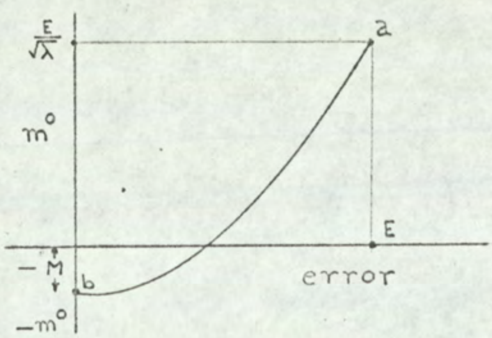


Fig. 5.3

Actual Function Generator Curve

zero to the desired value, as depicted in fig. 5.1. At the instant the system attained zero error, the plant vectors would be as shown in fig. 5.4., i.e. zero error, negative control effort (-M) and the desired output. The

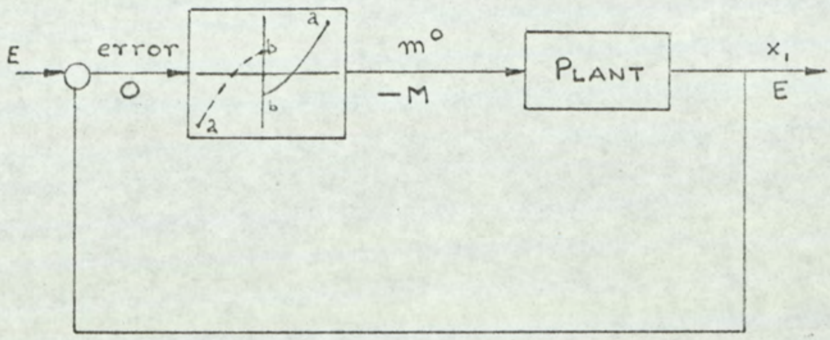


Fig. 5.4

Optimal System Showing Vector Values at the Instant of Attaining Zero Error

only forcing vector will be the negative control effort. This will cause the output to decrease thus producing a positive error. Small system error will also produce negative control efforts thus causing the output to decrease still further. When the error is sufficient to produce a positive control effort, the output will start to increase and thus reduce the error. Finally, possibly after many oscillations, the output will settle in a position where the error is such that the output of the function generator is zero. A typical output trajectory is shown in fig. 5.5

The maximum peak (position b) is not likely to be discontinuous as the slope at the instant before b will be almost zero, i.e. maximum overshoot is 4.3%. If the control was switched at point b, position of zero slope fig. 5.5, from the function generator to a linear gain function,

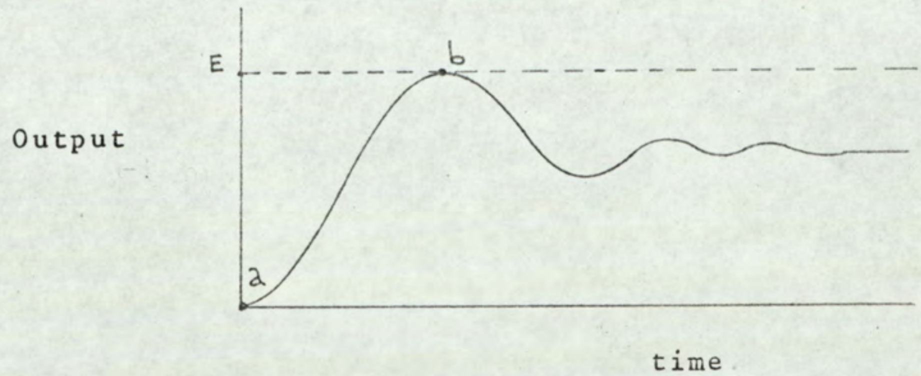


Fig. 5.5

Typical Output when Plant is Controlled via the Function Generator of Fig. 5.3

the steady state value may be maintained. At the instant before switching the values of the system vectors would be as represented in fig. 5.4. If the function generator was now replaced by a linear gain function, the control effort would be instantly reduced to zero thus maintaining the output of fig. 5.5 constant at its desired value E. The plant output would thus be improved over that of fig. 5.1 as the overshoot has been eliminated. This would result in a smaller value for the performance index.

The controller so described controls the transient response in an optimum manner and the steady state response as a normal system with unity feedback. A plant controlled in such a manner is represented pictorially in fig. 5.6 where switch 1 is closed and 2 open for transients initiated by changes of E, and 2 closed and 1 opened once steady state conditions have been attained.

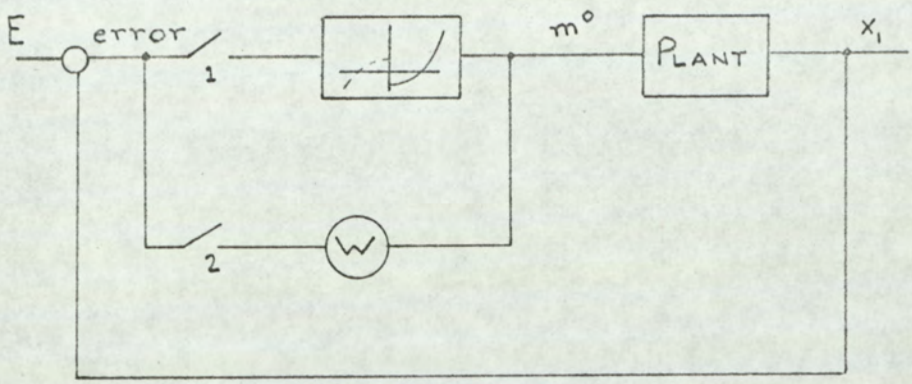


Fig. 5.6

Plant Controlled by Two State Controller

When the plant being optimised possesses a high gain it may be desirable for the gain coefficient W to be less than unity to reduce sensitivity of the system while operating in the steady state. Conversely, for a plant whose sensitivity is required to be increased, the steady state gain may be increased by making W greater than unity. It would also be possible to increase the plant gain for its transient response, and thereby produce a faster response, and reduce it under steady state conditions. Such a controller is shown in fig. 5.7 where the plant to be optimised is taken to have a transfer function K.Y, the gain factor

K being replaced by W for the steady state.

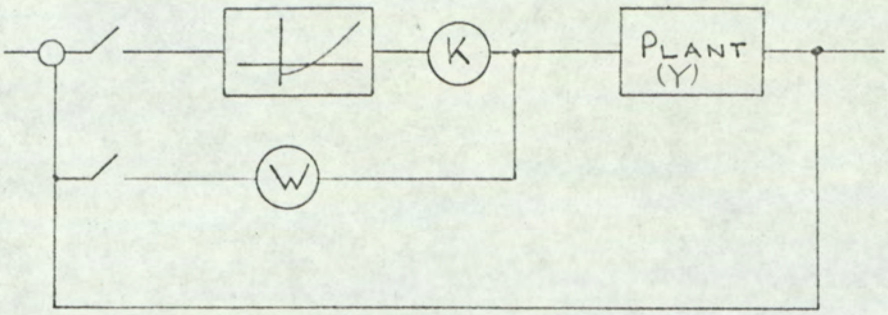


Fig. 5.7

Two State Controller to Enhance the Transient Response

Similar results obtained from the controller of fig. 5.7 may be obtained with the controller of fig. 5.6 if the weighting factor λ of the performance index is appropriately reduced.

The curve of control effort against error (the function generator characteristic) may be obtained without the use of a computer, i.e. mathematical formulae may be obtained for both the system error and control effort. Section 6.2 shows that for the plant whose open loop transfer function may be represented by $\frac{y}{S(S + a)}$, the optimal control effort may be written as:

$$m^{o}(t) = e^{-pt}\left(T \text{ Cos } \omega t + U \text{ Sin } \omega t\right) \quad .. \quad .. \qquad (5.1)$$

and the error as:

$$\mathcal{E}(t) = -e^{-pt}\left(E \text{ Cos } \omega t + R \text{ Sin } \omega t\right) \quad .. \qquad (5.2)$$

The constants of equations 5.1 and 5.2 are dependent upon the co-state initial conditions which may be calculated from the formulae of section 4.

5.1. Control of an actual Second Order Plant

The plant which was to be controlled is shown in fig. 5.1.1 where the amplifier supplied the field of a d.c. motor which, via a gear train, positioned the slider of a potentiometer. The actual plant is shown in fig. 5.1.1(a).

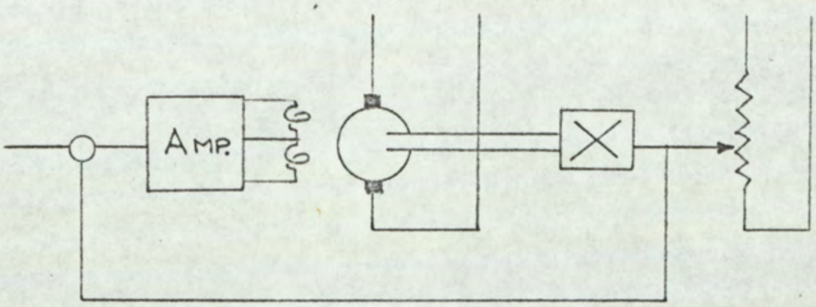
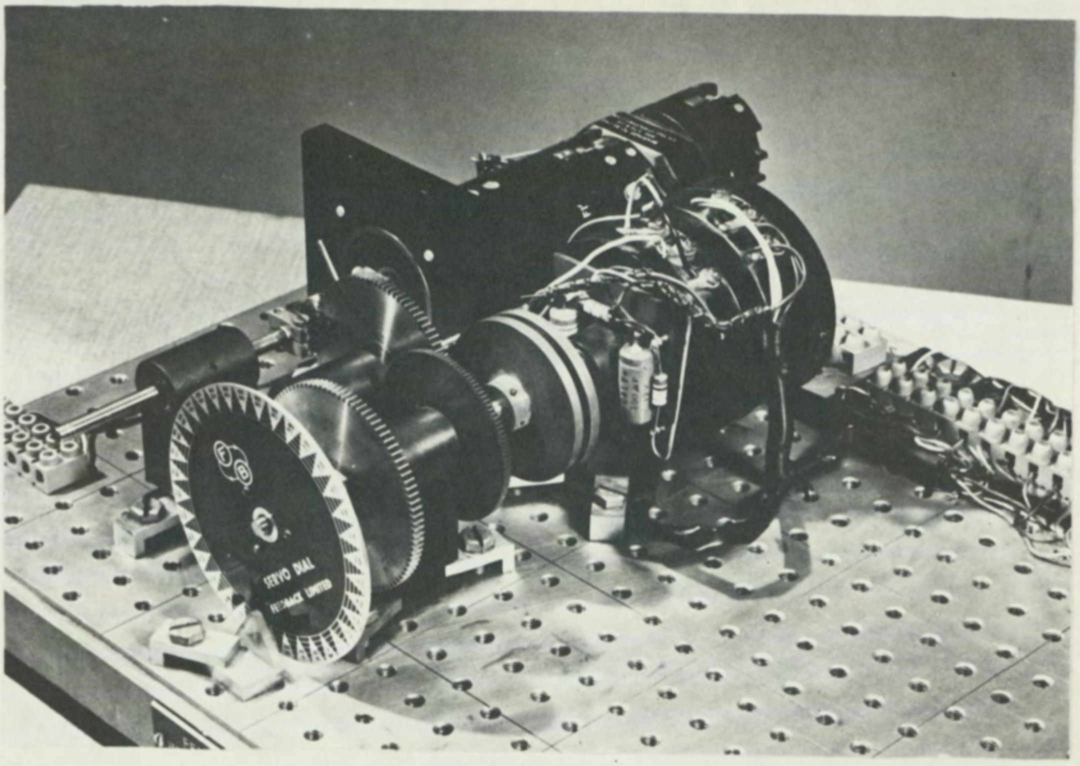


Fig. 5.1.1

Diagrammatic form of Plant

Initially, the transfer function of the plant was required. The most convenient and accurate method of obtaining this was to set up an analogue computer second order model with unity feedback and variable time constant and gain controls. (The plant being approximated as possessing one major time constant). The plant and model outputs were then matched, for the same step input, by adjusting the parameters of the model. An exact match was not possible due to the non-linearity of the plant and the possible influence of other time constants. A good approximation to the transfer function of the plant (fig. 5.1.2) was observed to be:



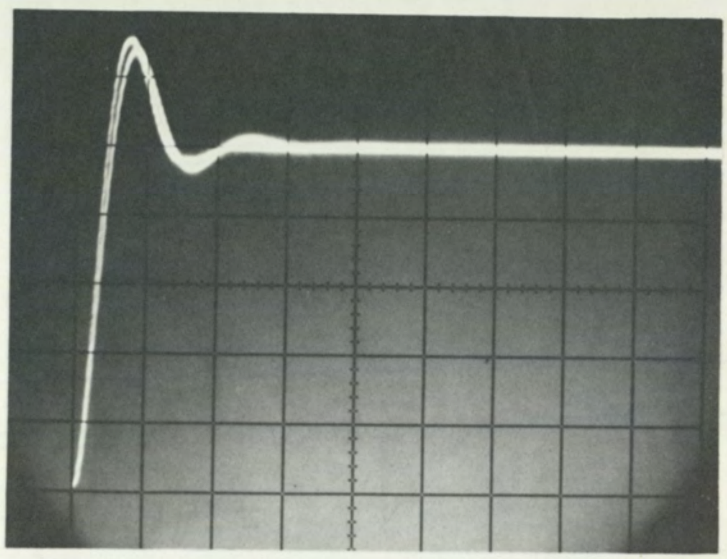
Actual second order plant

FIG. 5.1.1(a).

0.2 Volts/cm

0.5 Secs/cm

Output (Volts)



time (secs)

Model and Plant output.

FIG. 5.1.2.

$$Y(s) = \frac{71}{s(s + 6.6)} \quad \ldots \quad \ldots \quad \ldots \quad (5.1.1)$$

The state equations representing equation 5.1.1 may be written:

$$\dot{x}_1 = 71x_2 \quad \ldots \quad \ldots \quad \ldots \quad (5.1.2)$$

$$\dot{x}_2 = m - 6.6x_2 \quad \ldots \quad \ldots \quad \ldots \quad (5.1.3)$$

Applying Pontryagin's Maximum Principle, the third state variable may be written:

$$\dot{x}_3 = (E - x_1)^2 + \lambda m^2 \quad \ldots \quad \ldots \quad (5.1.4)$$

The Hamiltonian may be written:

$$H = p_1(71x_2) + p_2(m - 6.6x_2) - \left[(E - x_1)^2 + \lambda m^2\right] \quad (5.1.5)$$

$$\frac{\partial H}{\partial m} = p_2 - 2\lambda m \quad \ldots \quad \ldots \quad (5.1.6)$$

For H to be maximised w.r.t.m.

$$m^o = \frac{p_2}{2\lambda} = \frac{p_2}{0.2} \quad \ldots \quad \ldots \quad (5.1.7)$$

$$\dot{p}_1 = \frac{-\partial H}{\partial x_1} = 2(x_1 - E) \quad \ldots \quad \ldots \quad (5.1.8)$$

$$\dot{p}_2 = \frac{-\partial H}{\partial x_2} = 6.6p_2 - 71p_1 \quad \ldots \quad \ldots \quad (5.1.9)$$

Equations 5.1.2, 3, 7, 8 and 9 may be mechanised on an analogue computer as shown in fig. 5.1.3 where $p_1' = {}^{p1}/2$ and $p_2' = {}^{p2}/2$, $A' = \frac{A}{2}$ and $B' = \frac{B}{2}$.

The initial value of p_2 (B), may be calculated from:

$B = 2E\sqrt{0.1}$

For E = 1.0, B = 0.6324 .. .. .. (5.1.10)

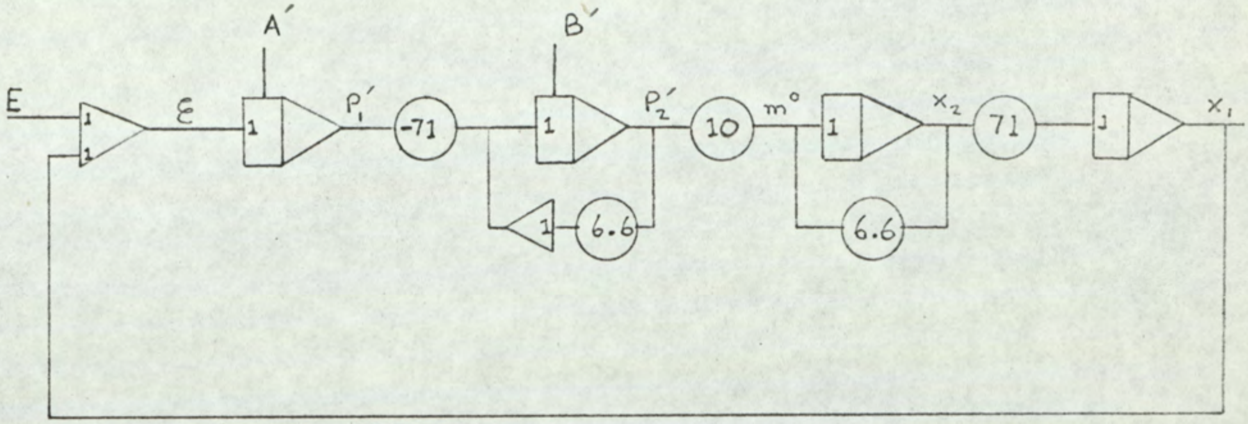


Fig. 5.1.3

Analogue Circuit for Optimum System: $\frac{71}{S(S + 6.6)}$

The initial values of p_1,(A), may be calculated from

$$A = 2E\sqrt{\frac{a^2\lambda}{y^2} + \frac{2\sqrt{\lambda}}{y}}$$

where a = 6.6

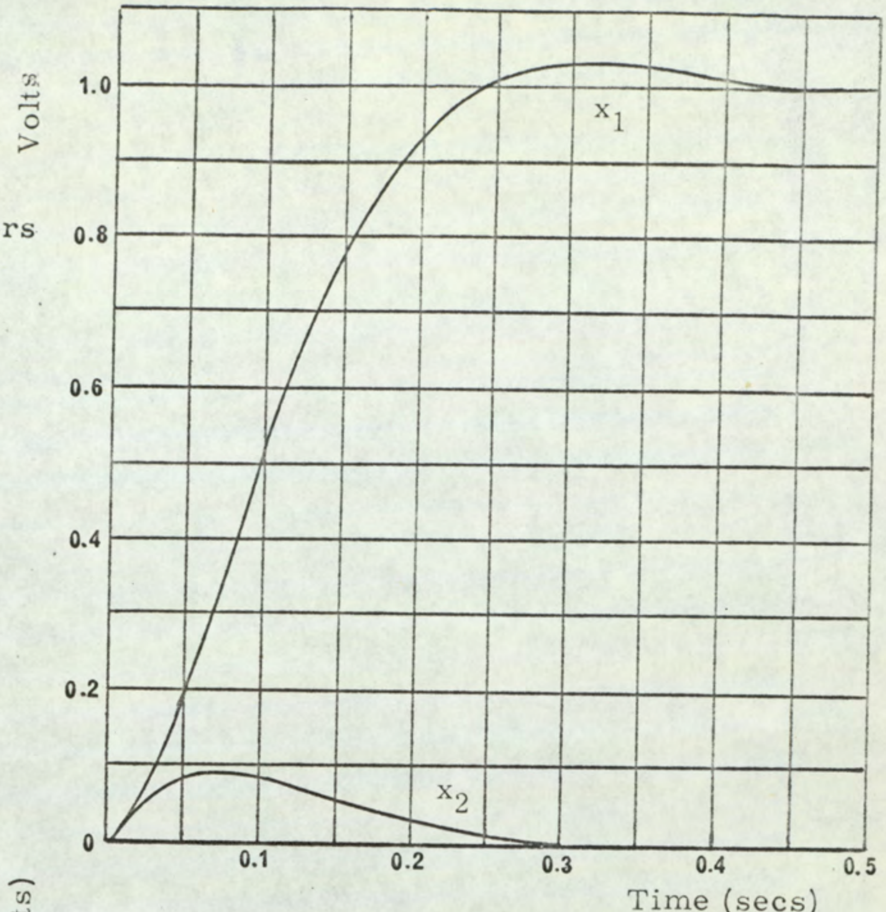
y = 71

λ = 0.1

For E = 1.0, A = 0.1977 .. .. .. (5.1.11)

The resulting model optimum trajectories were obtained by digital simulation and the resulting trajectories are depicted in fig. 5.1.4(a) and (b). The m°v error curve is shown in fig. 5.1.4(c).

Plant state vectors



Co-state vectors

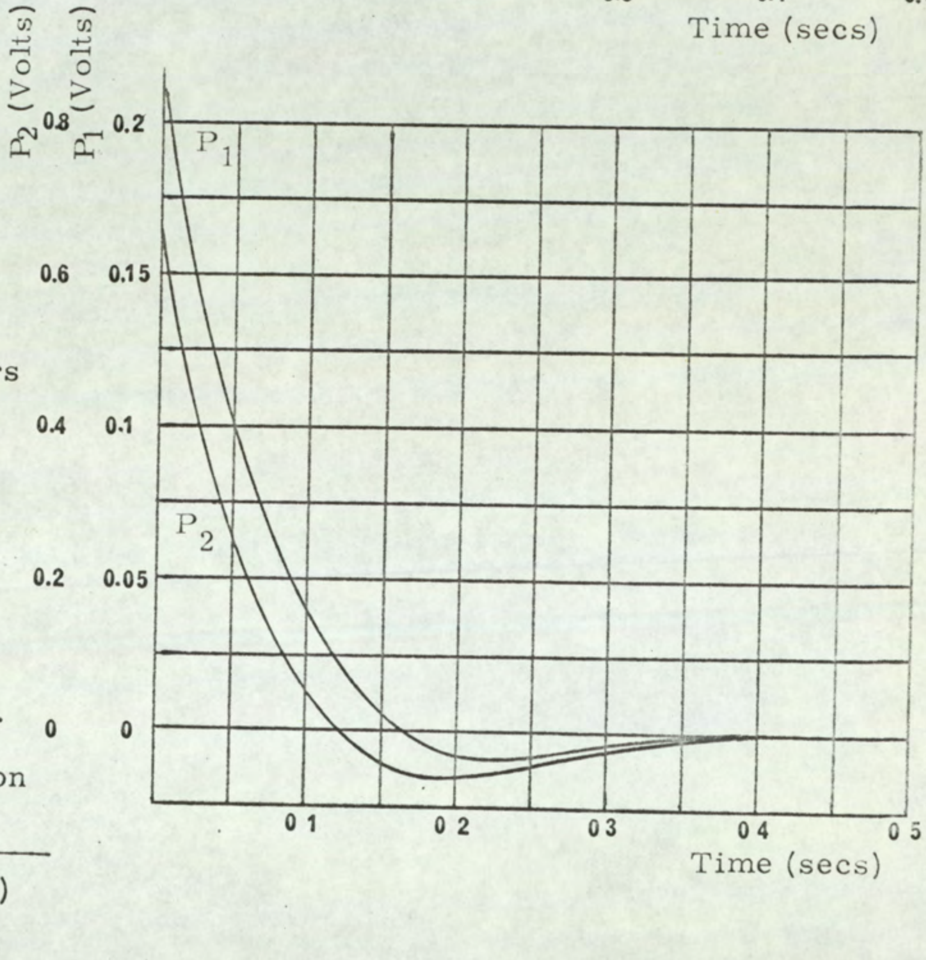


FIG. 5.1.4A. Optimum trajectories for digital simulation of plant $\frac{71}{S(S+6.6)}$

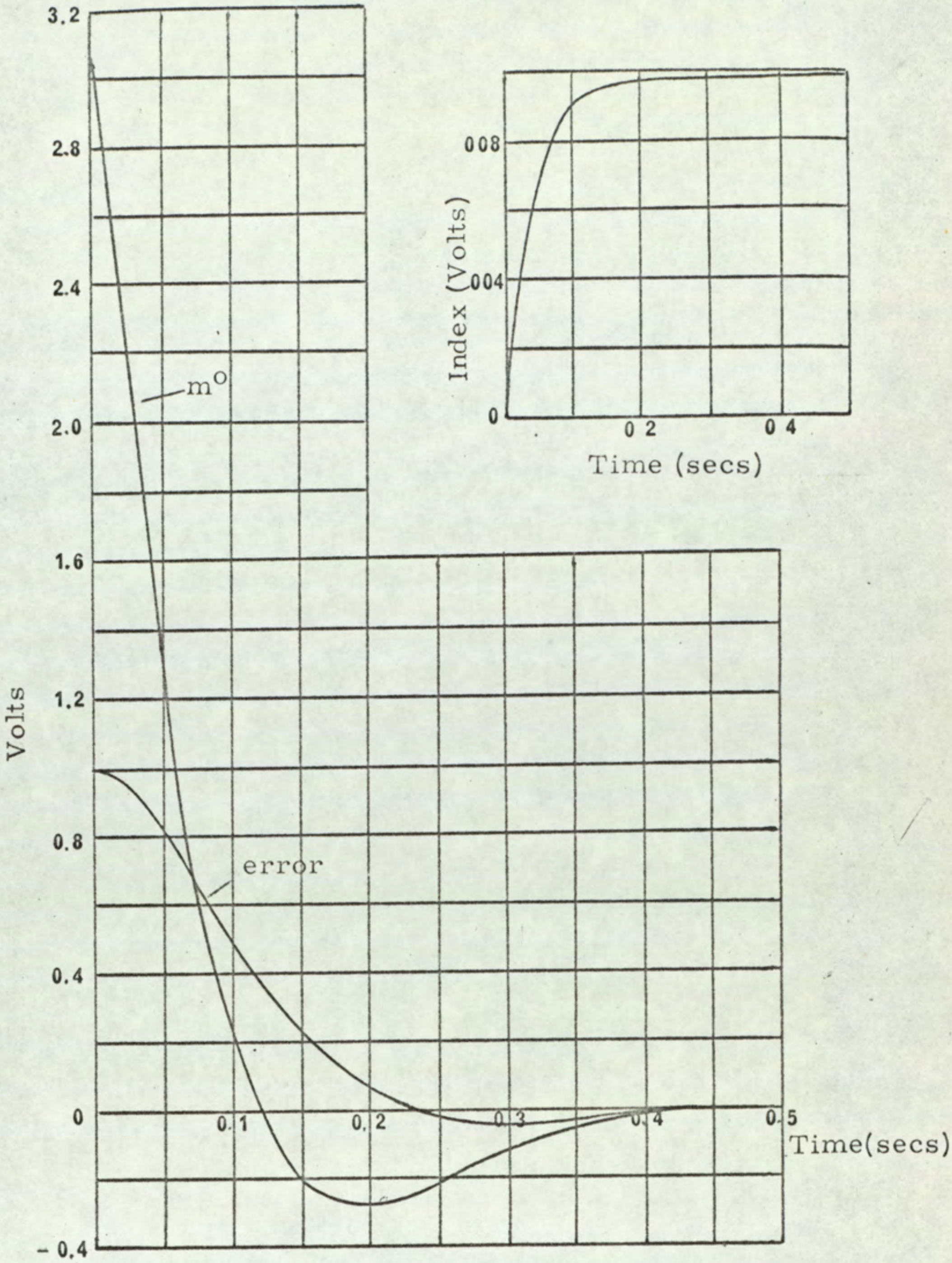


FIG. 5.1.4B.

Optimum trajectories from digital simulation for plant $\frac{71}{S(S+6.6)}$

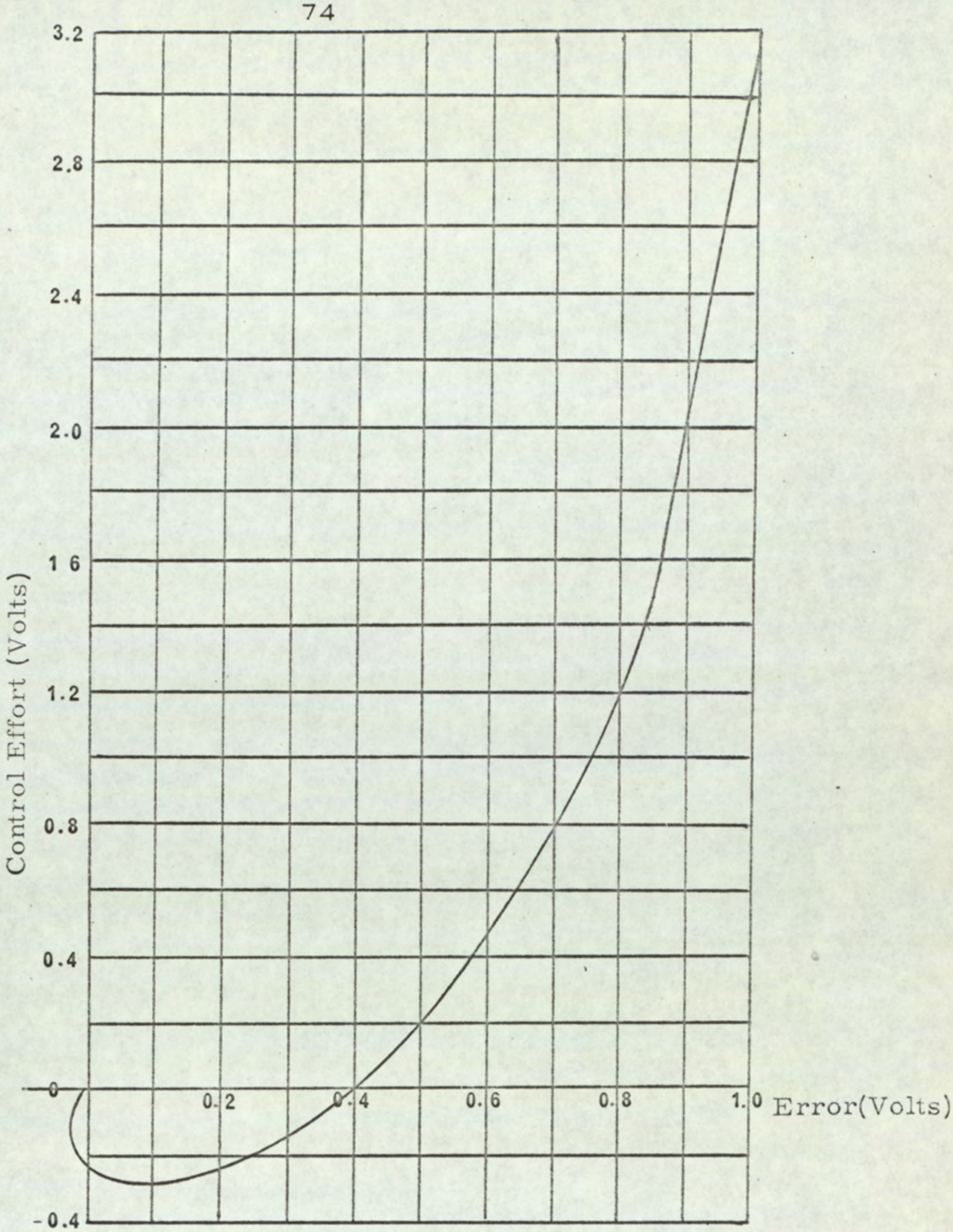
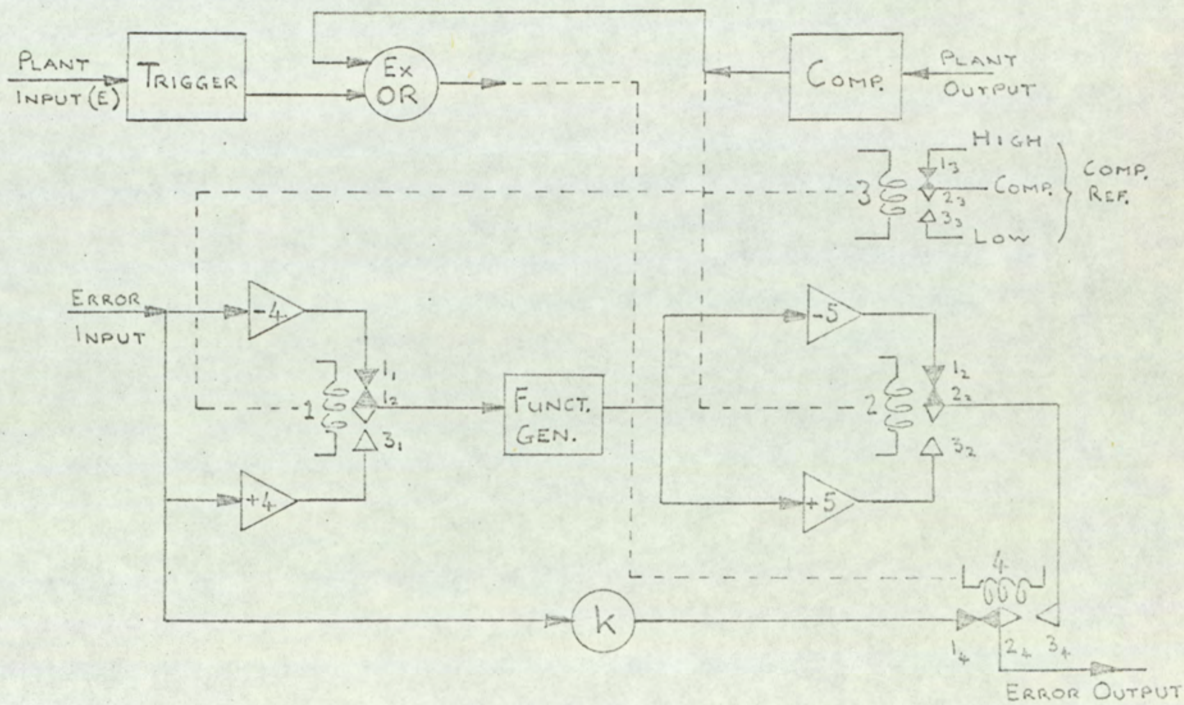


FIG. 5.1.4C.

Curve of control effort against error, for plant $\frac{71}{S(S+6.6)}$

A two stage controller, as shown in fig. 5.6, was designed to incorporate the function generator curve of fig. 5.1.4(c) for error in the range 1.0 volts to 0 volts. (The actual design of the function generator is given in appendix 3). A schematic diagram of the actual controller is shown in fig. 5.1.5.

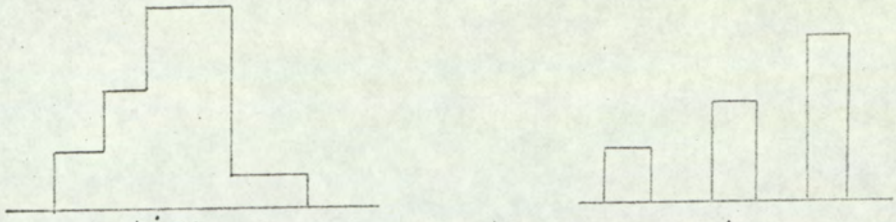


Two State Controller

Fig. 5.1.5

This controller was specifically designed to operate from negative step inputs for switching on and off. With the system input at zero, the relay contacts were in the position shown, i.e. dark contacts closed. Under such conditions, the error output = K × (error input). The value of K was made equal to unity. When the plant input or control effort changed in a negative direction, relay 4 was energised and contacts 2_4 and 3_4 closed. The negative plant error was now inverted by the amplifier with a gain of -4, shaped by the function generator, inverted by the amplifier with a gain of -5 and injected back into the plant. When the plant output attained its desired value, the comparitor excited all the relays. This set the comparator reference to the next transient desired value, switched control to the non-inverting amplifiers and produced the output error again equal to K × (input error). For a positive change in plant input, i.e. switching off, relay 4 was again energised and the positive error would now be channelled via the non-inverting amplifiers through the function generator to the error output. When the output again reached its desired value, the comparator activated all the relays returning them to their initial state.

With the aid of additional logic circuitry and required function generators, it would be possible to accommodate multi-step inputs as shown in fig. 5.1.6. Furthermore, by using a zero slope detector on the plant output to initiate switching, the need for a multi-reference comparator would be eliminated.



Examples of possible plant Inputs

Fig. 5.1.6

The controller of fig. 5.1.5 was initially used to control the analogue computer model. Fig. 5.1.7 depicts the output trajectories with and without switching and fig. 5.1.8 the output with and without control. The control effort is shown in fig. 5.1.9 which clearly portrays the switching instant. Fig. 5.1.10 portrays the index. The steady state value of 0.092 was less than that obtained from the digital simulation of fig. 5.1.4(b), i.e. 0.098. This difference may be attributed to error in the analogue computer, the absence of overshoot in output and the premature attainment of zero for m^o (fig. 5.1.9) when the model was controlled via the double function generator.

Having successfully obtained control of the model, the same controller was used to control the actual plant. The results are depicted in the trajectories of figs. 5.1.11-14 with the measured value of the performance

Model Output with

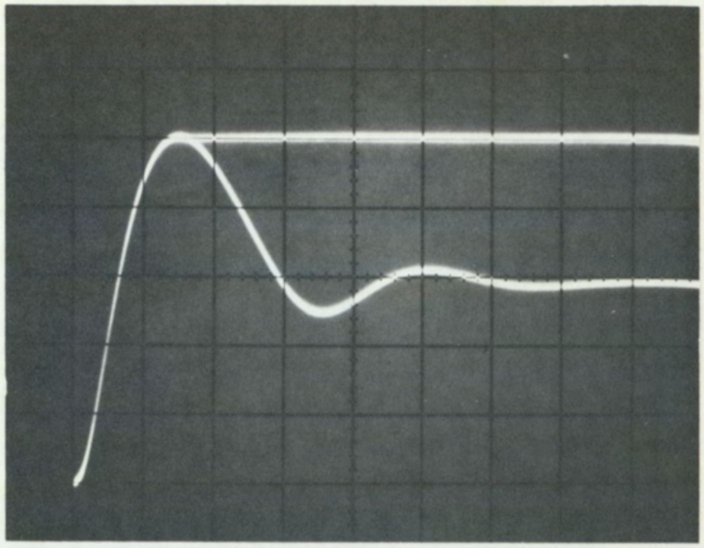
and without

Switching

0.2 volts/cm

0.2 secs/cm.

Output (volts)



time (secs)

Fig. 5.1.7

Model Output with

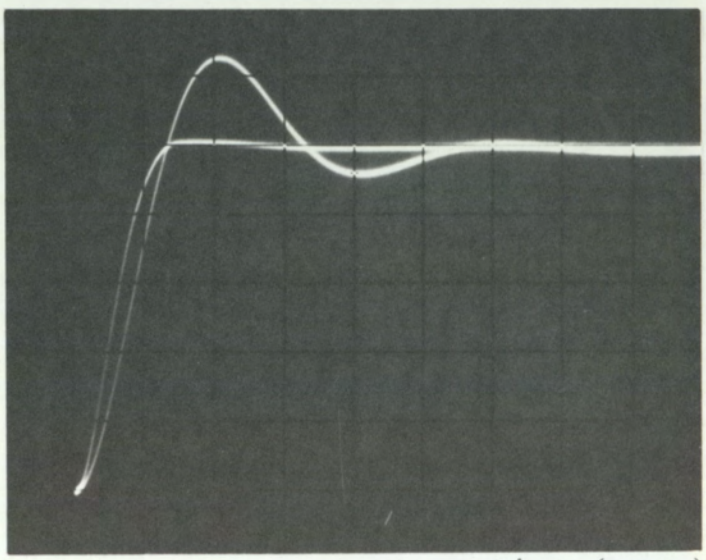
and without

Control

0.2 volts/cm

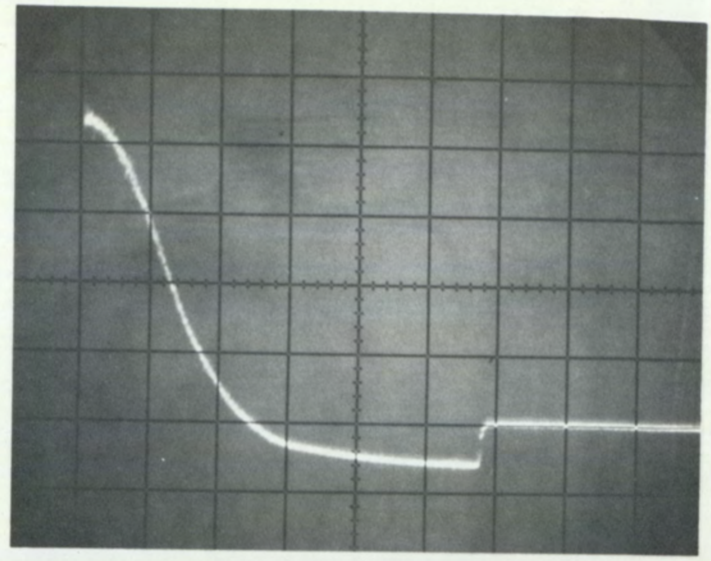
0.2 secs/cm

Output (volts)



time (secs)

Fig. 5.1.8

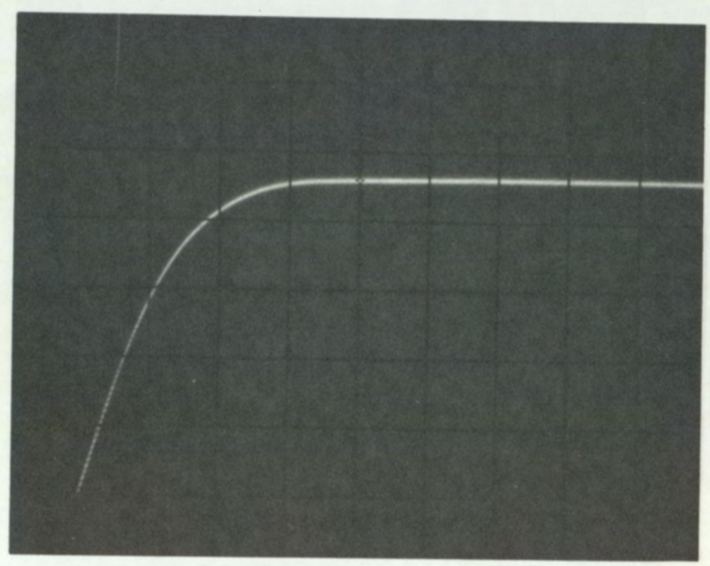


Model Control Effort for Control effected via the two state Controller

0.5 volts/cm

0.05 secs/cm

Fig. 5.1.9



Value of index for Control effected via the two state Controller

0.02 volts/cm

0.1 secs/cm

Fig. 5.1.10

Actual Plant Output

with and without

Switching

0.2 volts/cm

0.2 secs/cm

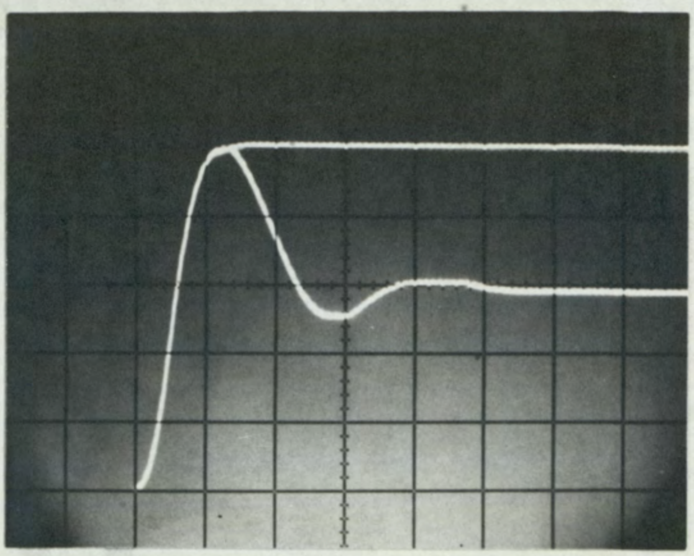


Fig. 5.1.11

Actual Plant Output

with and without

Control

0.2 volts/cm

0.2 secs/cm

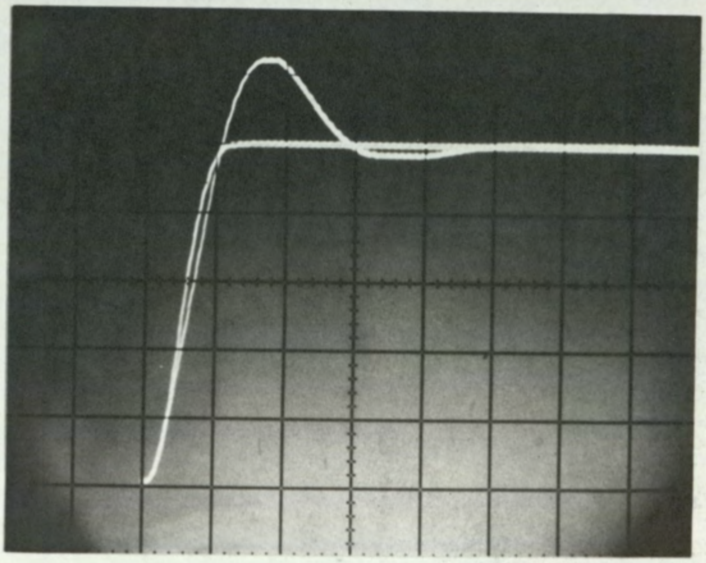


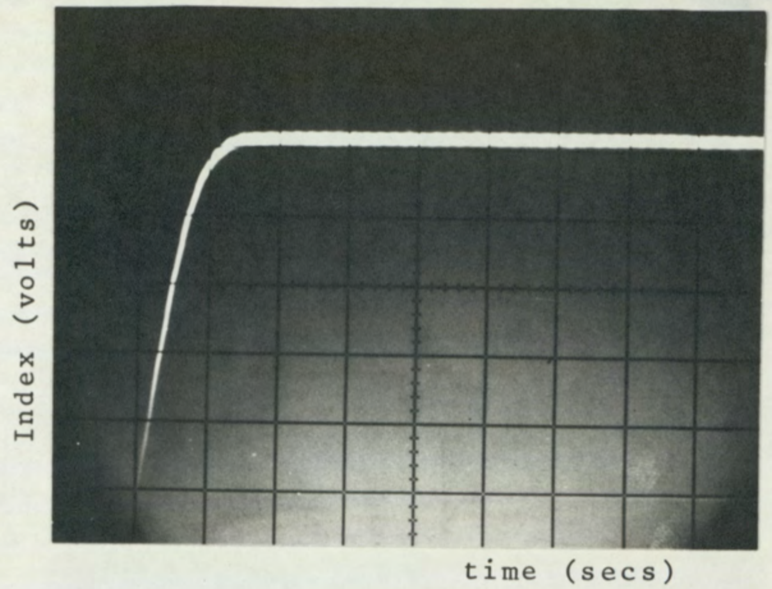
Fig. 5.1.12

Index of actual Plant controlled via the two state Controller

0.02 volts/cm

0.1 secs/cm

<u>Fig. 5.1.13</u>

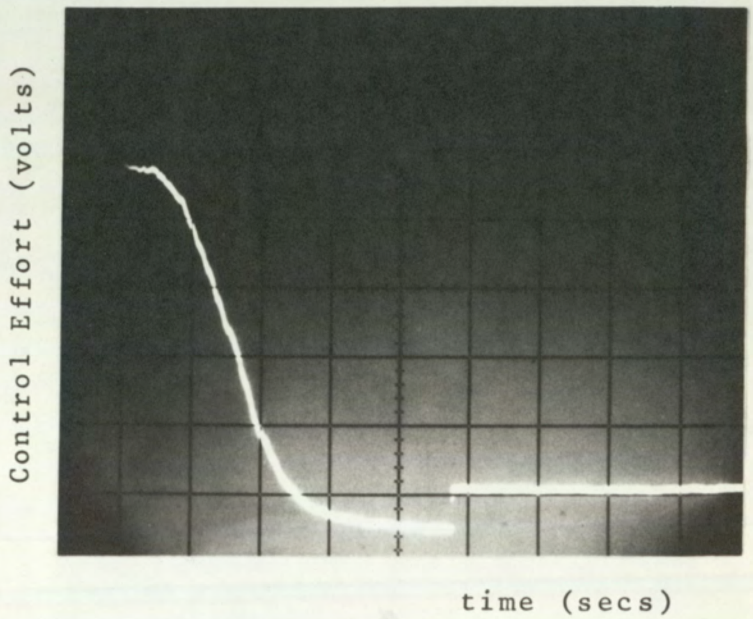


Control Effort of actual Plant for Control via the two state Controller

0.5 volts/cm

0.05 secs/cm

<u>Fig. 5.1.14</u>



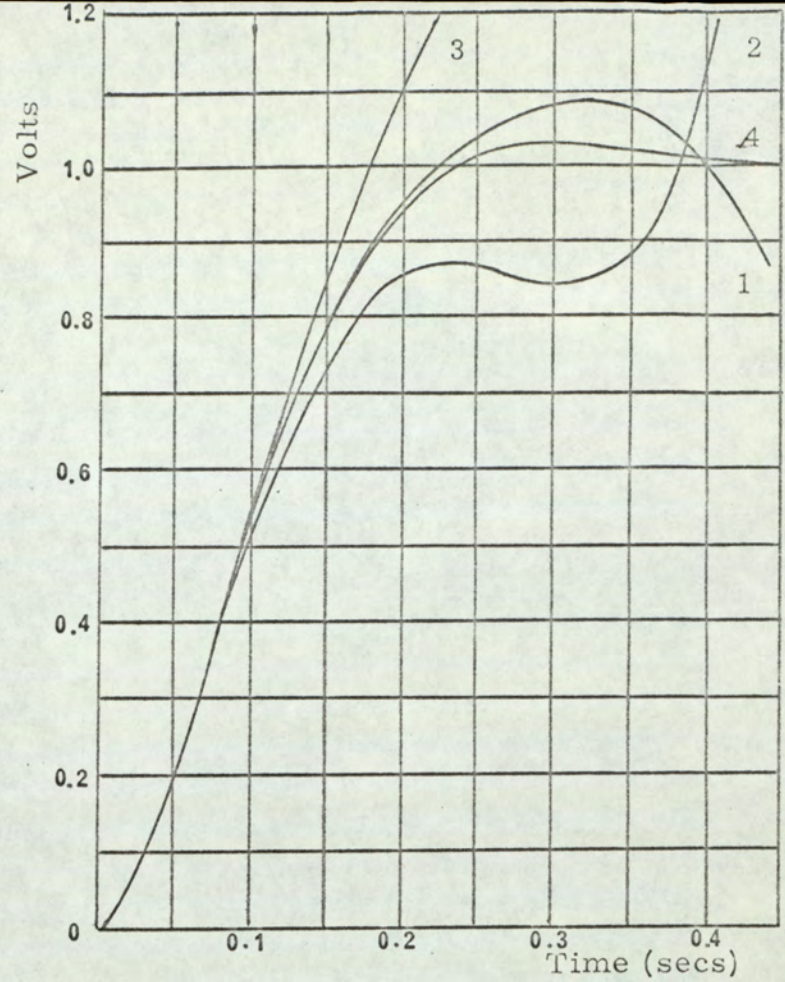
index (fig. 5.1.13) being 0.102, i.e. an approximate increase of 10% compared with that of the model.

5.2 Sensitivity and Correction for Model Error

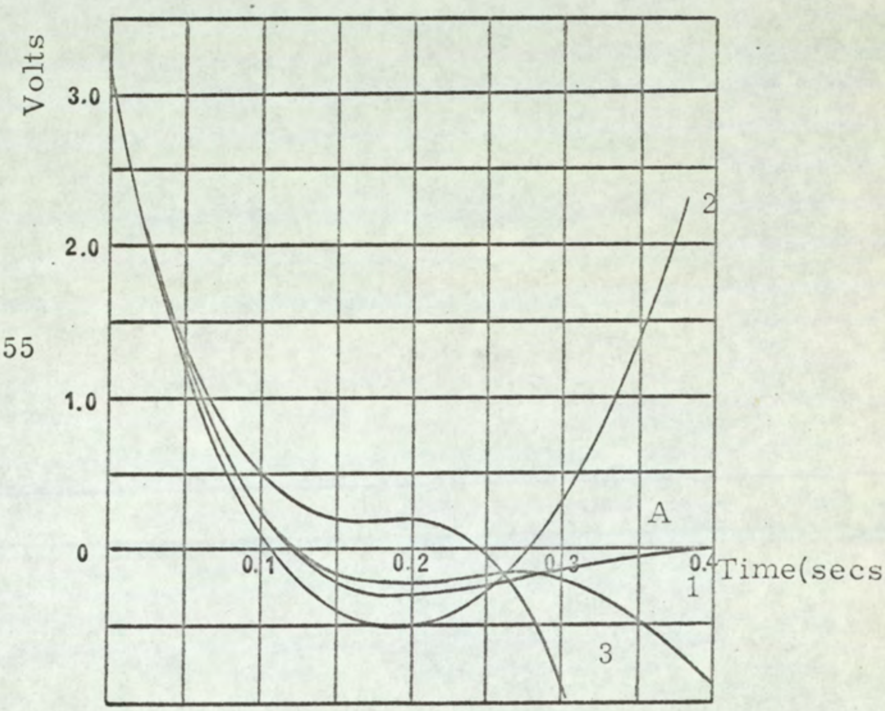
Figure 5.2.1 demonstrates the sensitivity of the plant to variation in the value of initial co-state vector magnitudes when optimised for the infinite time interval. It may be observed that a change in the order of 1% will produce an excessive change in the control effort and resulting output. An error in the model time constant (a) and gain coefficient (Y) of 2% compared with those existing in the actual plant would produce an error in the initial condition A, according to equation 4.2(b).13 of 5%. It is thus apparent that, neglecting the instability produced by noise in the steady state, it is very unlikely that a stable system, due to error in the model, would evolve if the model adjoint system was used to control the actual plant. Control effected via a switched function generator would not, however, invoke instability but, according to the degree of model error, may produce sub-optimal control.

The resulting plant output trajectory, if controlled via a function generator without switching, would be similar to that of fig. 5.5. except that the magnitude of the initial peak would not be equal to the desired output.

Output trajectories



Control effort trajectories



	Initial P_1	Initial P_2
A:	0.197706	0.632455
1:	0.197	0.632
2:	0.20	0.632
3:	0.192	0.630

FIG. 5.2.1.

Curves of plant output and control effort against time for different initial co-state vector values. Plant transfer function $\frac{71}{S(S+6.6)}$

Under such circumstances, the function generator may be modified so as to produce the required peak value. When the model is obtained by matching the plant output with that of the model, the resulting error will be small and will affect the function generator in the region of zero magnitude. Any necessary adjustment of the function generator may be obtained by variation of the governing potential chain.

6. Comparison of Optimisation by the Modified Pontryagin Principle and Dynamic Programming

6.1 Dynamic Programming

The state equations for the second order system $\frac{y}{S(S + a)}$ are:

$$\dot{x}_1 = Yx_2 \quad .. \quad .. \quad .. \quad (6.1.1)$$

$$\dot{x}_2 = m - Ax_2 \quad .. \quad .. \quad .. \quad (6.1.2)$$

The performance index to which the system is to be optimised is:

$$I = \int_0^\infty \left[\alpha_K (E - x_1)^2 + \lambda m^2\right] dt$$

where $\alpha_2 = 0$, $\alpha_1 = 1$

Equations 6.1.1 and 2 may be written in matrix notation:

$$\dot{x}(t) = Bx(t) + Dm(t) \quad .. \quad (6.1.3)$$

where $B = \begin{bmatrix} 0 & Y \\ 0 & -A \end{bmatrix}$ $D = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$.

Merriam (ref. 20) shows that the optimum control effort for a system expressed in matrix notation and optimised according to a quadratic performance index is of the form:

$$m_i^o(t) = \frac{d_{ii}(t)}{\lambda} \left[k_i(t) - \sum_{m=1}^{N} k_{im}(t)\, x_m(t) \right] \quad (6.1.4)$$

Where $k_i(t)$ and $k_{im}(t)$ are time varying gains, the values of which are obtained via the solution to the Riccatian equations:-

$$-\dot{k}_{mk}(t) = \sum_{n=1}^{N} (\alpha_n a_{nm} a_{nk} + b_{nm} k_{nk} + b_{nk} k_{nm} - d_n^2 k_{nm} k_{nk})$$

(6.1.5)

and

$$-\dot{k}_m(t) = \sum_{n=1}^{N} (\alpha_n \mathcal{E}\; a_{nm} + b_{nm} k_n - d_n^2 k_n k_{nm}) \qquad (6.1.6)$$

Where a_{ii} are the elements of a unit matrix of order N.

Equations 6.1.4, 5 and 6 were obtained with the assumption that the instantaneous error function possessed a solution of the form:

$$\mathcal{E}[x(\mu),\mu] = k(\mu) - 2 \sum_{m=1}^{N} k_m(\mu) x_m(\mu)$$

$$+ \sum_{m=1}^{N} \sum_{k=1}^{N} k_{mk}(\mu) x_m(\mu) x_k(\mu)$$

where $k_{mk}(\mu) = k_{km}(\mu)$

Equation 6.1.4 produces the optimum control effort for the system considered as:

$$m^o(t) = \frac{1}{\lambda} \left[k_2(t) - k_{12}(t) x_1(t) - k_{22}(t) x_2(t) \right] \qquad (6.1.7)$$

The resulting Riccatian equations are:

$$-\dot{k}_1(t) = E\ (t) - k_2(t)k_{21}(t) \quad .. \quad .. \quad .. \quad (6.1.8)$$

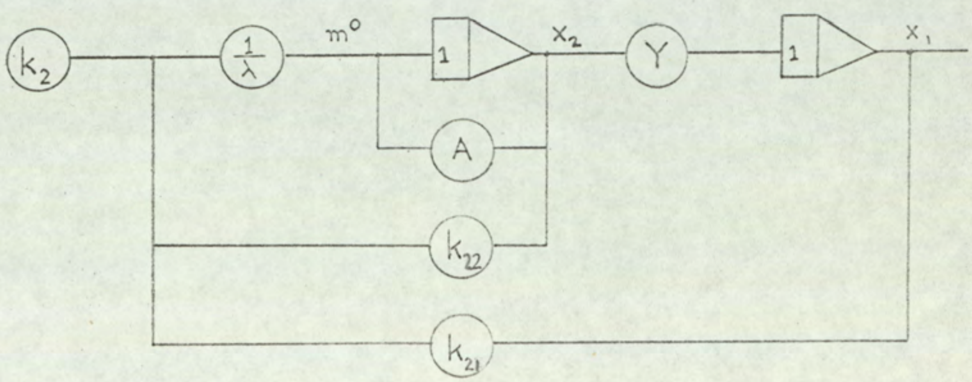
$$-\dot{k}_2(t) = Y.k_1(t) - A.k_2(t) - k_2(t)k_{22}(t) \quad .. \quad (6.1.9)$$

$$-\dot{k}_{11}(t) = 1 - k_{21}(t)^2 \quad .. \quad .. \quad .. \quad .. \quad (6.1.10)$$

$$-\dot{k}_{12}(t) = Y.k_{11}(t) - A.k_{21}(t) - k_{21}(t)k_{22}(t) \quad .. \quad (6.1.11)$$

$$-\dot{k}_{22}(t) = 2.Y.k_{12}(t) - 2.A.k_{22}(t) - k_{22}(t)^2 \quad .. \quad (6.1.12)$$

Equation 6.1.7 produces the system of fig. 6.1.1 from which it can be observed that the Riccatian equations need only be solved for the gains $k_2(t)$, $k_{12}(t)$ and $k_{22}(t)$.



Optimisation of $\frac{y}{S(S + a)}$ according to Dynamic Programming

Fig. 6.1.1

When solving for the infinite interval case, all

all the derivatives of the gains will be zero, i.e. each gain becomes constant. Hence to aid the calculation of the infinite interval gains, equations 6.1.8-12 may be equated to zero:

From equation 6.1.10:

$$k_{12}^2(\infty) = 1$$

$$\therefore \quad k_{12}(\infty) = \pm 1 \quad .. \quad .. \quad .. \quad (6.1.13)$$

For a stable system at $t = \infty$, the feedback gains depicted in equation 6.1.7 must all be positive. Therefore,

$$k_{12}(\infty) = 1$$

Substituting equation 6.1.13 into 6.1.12 produces:

$$2.Y. - 2.A.\ k_{22}(\infty) - k_{22}^2(\infty) = 0$$

$$\therefore \quad k_{22}(\infty) = -A \pm \sqrt{A^2 + 2Y}$$

Required $\quad k_{22}(\infty) = -A + \sqrt{A^2 + 2Y} \quad .. \quad (6.1.14)$

Substituting equation 6.1.13 into equation 6.1.8:

$$k_2(\infty) = E \quad .. \quad .. \quad .. \quad (6.1.15)$$

Equation 6.1.15 indicates that the optimum input gain will always be equal to the desired output and the overall feedback (equation 6.1.13) will always be unity. These are the required parameters for any infinite interval position control system linearly controlled and may be deduced without any prior mathematics.

The Riccatian equations were solved on a digital computer. Specimen results verifying equations 6.1.13, 14 and 15 are shown in fig. 6.1.2. Fig. 6.1.3. depicts the optimum trajectories and index obtained for the system $\frac{71}{S(S = 6.6)}$, the optimum value of $k_{12}(\infty)$ being 7.022, $k_2 = E = 1.0$ and $k_{12} = 1.0$.

	TIME (SECS)	k_2	k_{12}	k_{22}
E = 1.0				
	0.1	0.279	0.279	1.014
	0.2	0.755	0.755	4.300
	0.3	0.967	0.967	6.430
	0.6	0.999	0.999	7.017
	0.8	1.000	1.000	7.021
	1.0	1.000	1.000	7.022
E = 2.0				
	0.04	0.102	0.051	0.085
	0.1	0.558	0.279	1.014
	0.2	1.509	0.755	4.296
	0.3	1.933	0.967	6.430
	0.4	1.998	0.999	6.957
	0.5	1.999	0.999	7.016
	1.0	2.000	1.000	7.022

Time Varying Gains for Plant $\frac{71}{s(s + 6.6)}$

Fig. 6.1.2

Output

Control Effort

Index

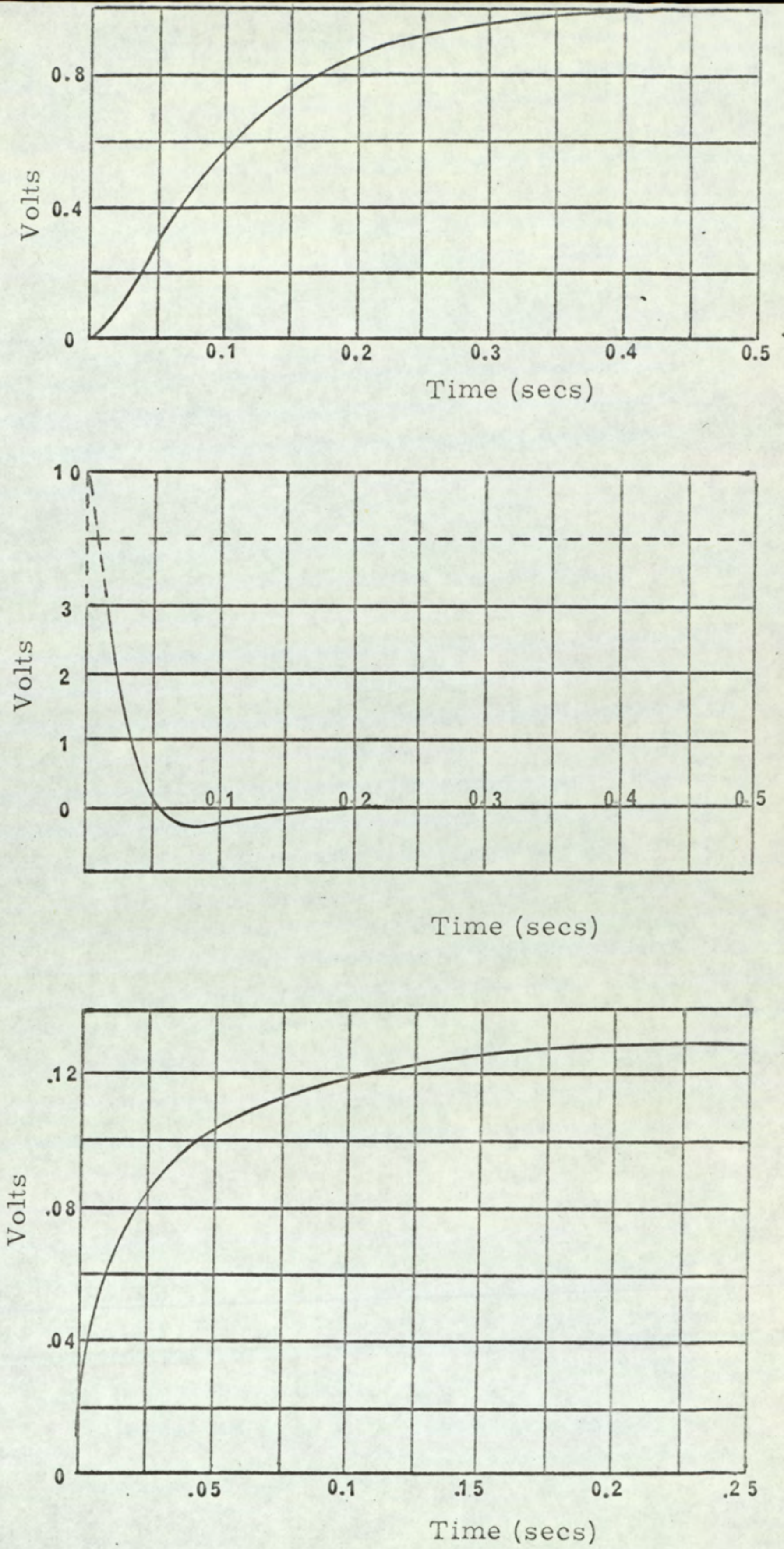


FIG. 6.1.3.

Trajectories of plant $\frac{71}{S(S+6.6)}$ when optimised according to Dynamic Programming.

6.2 Calculation of the Minimum Value of the Performance Index

Equation 4.2(b).2 gives the system output in the 's' plane as

$$x(s) = \frac{BY(s^2 - xs + z)}{2.\lambda.s.\left(s^4 - (x^2 - 2z)s^2 + z^2\right)}$$

$$= \frac{BY}{2.\lambda.s.(s^2 + xs + z)} \quad .. \quad .. \quad (6.2.1)$$

Thus $x_1(t)$ may be expressed as:

$$x_1(t) = E - e^{-pt}(E \cos \omega t + R \sin \omega t)$$

where $p = \frac{x}{2}$; $R = \frac{x}{2\omega}$; $\omega = \sqrt{z - \frac{x^2}{4}}$

The system error $(E - x_1)$ will therefore be:

$$\varepsilon = -e^{-pt}(E \cos \omega t + R \sin \omega t)$$

$$\varepsilon^2 = \frac{e^{-pt}}{2}\left[E^2 + R^2 + (E^2 - R^2)\cos 2\omega t + 2ER \sin 2\omega t\right] \quad (6.2.2)$$

From fig. 4.1 and equation 6.2.1:

$$m^o(s)\left[\frac{y}{s(s+a)}\right] = \frac{BY}{2.\lambda.s.(s^2 + xs + z)}$$

$$\therefore \quad m^o(s) = \frac{BS + Ba}{2.\lambda(s^2 + xs + z)} \quad ... \quad (6.2.3)$$

$$\therefore \quad m^o(t) = e^{-pt}\left[T \cos \omega t + U \sin \omega t\right]$$

where $T = \frac{B}{2\lambda}$; $U = \frac{aB}{2.\lambda.\omega} - \frac{Bx}{4.\lambda.\omega}$

$$\therefore \quad \lambda[m^o(t)]^2 = \frac{\lambda e^{-2Pt}}{2}\left[T^2 + U^2 + (T^2 - U^2)\,\text{Cos}\,2\omega t + 2TU\,\text{Sin}\,2\omega t)\right]$$

The performance index, $\int_0^\infty (E - x_1)^2 + \lambda m^2)dt$,

may be written:

$$J = \int_0^\infty \frac{e^{-2Pt}}{2}\Big[E^2 + R^2 + (E^2 - R^2)\,\text{Cos}\,2\omega t + 2ER\,\text{Sin}\,2\omega t$$

$$+ \lambda(T^2 + U^2) + \lambda(T^2 - U^2)\,\text{Cos}\,2\omega t + \lambda.2.T.U\,\text{Sin}\,2\omega t)\Big]\,dt$$

$$= \frac{e^{-2Pt}}{2}\Bigg[\frac{-1}{2P}(E^2 + R^2) + \frac{(E^2 - R^2)(2\omega\,\text{Sin}\,2\omega t - 2P\,\text{Cos}\,2\omega t)}{4P^2 + 4\omega^2}$$

$$+ \frac{2ER(-P2\text{Sin}\,2\omega t - 2\omega\,\text{Cos}\,2\omega t)}{4P^2 + 4\omega^2} - \frac{\lambda}{2P}(T^2 + U^2)$$

$$+ \frac{\lambda(T^2 - U^2)(2\omega\,\text{Sin}\,2\omega t - 2P\,\text{Cos}\,2\omega t)}{4P^2 + 4\omega^2}$$

$$+ \frac{\lambda.2.T.U.(-2P\,\text{Sin}\,2\omega t - 2\omega\,\text{Cos}\,2\omega t)}{4P^2 + 4\omega^2}\Bigg]_{t=0}^{\infty}$$

$$= \frac{E^2}{4P} + \frac{R^2}{4P} + \frac{(E^2 - R^2)(2P)}{2(4P^2 + 4\omega^2)} + \frac{4.ER\omega}{2(4P^2 + 4\omega^2)}$$

$$+ \frac{\lambda(T^2}{4P} + \frac{U^2)}{4P} + \frac{\lambda(T^2 - U^2)(2P)}{2(4P^2 + 4\omega^2)} + \frac{2.\lambda.T.U.\omega.}{(4P^2 + 4\omega^2)}$$

Or

$$J = \frac{1}{4P}(E^2 + R^2) + \frac{(E^2 - R)(P)}{(4P^2 + 4\omega^2)} + \frac{2E.R.\omega}{4P^2 + 4\omega^2} + \frac{\lambda}{4P}(T^2 + U^2)$$

$$+ \frac{\lambda(T^2 - U^2)(P)}{4P^2 + 4\omega^2} + \frac{2.T.U.\omega.\lambda.}{4P^2 + 4\omega^2} \quad .. \quad .. \quad (6.2.4)$$

From fig. 6.1.1 the output in the S plane of the system when optimised according to Dynamic Programming is:

$$x_1(s) = \frac{EY}{\lambda S(S^2 + S(^k/\lambda + a) + ^y/\lambda)} \quad .. \quad .. \quad (6.2.5)$$

where $k = k_{12}$.

and

$$m^o(s) = \frac{ES + Ea}{\lambda(S^2 + S(^k/\lambda + a) + ^y/\lambda)} \quad .. \quad .. \quad (6.2.6)$$

$\therefore \quad x_1(t) = E - e^{-Ct}(E \text{ Cos } \omega_1 t + V \text{ Sin } \omega_1 t)$

and $\quad m^o(t) = e^{-Ct}(E \text{ Cos } \omega_1 t + W \text{ Sin } \omega_1 t)$

where

$$\omega_1 = \sqrt{\frac{y}{\lambda} - \frac{(^k/\lambda + a)^2}{4}} \; ; \quad V = \frac{E}{2\omega_1}\left(\frac{k}{\lambda} + a\right)$$

$$C = \frac{k}{2\lambda} + \frac{a}{2} \; ; \quad W = \frac{Ea}{\omega_1} - E\left(\frac{k}{2\lambda} + \frac{a}{\omega_1}\right)$$

The resulting value of the performance index will be:

$$J_1 = \frac{1}{4C}(V^2 + E^2) + \frac{(E^2 - V^2)(C)}{4C^2 + 4\omega_1{}^2} + \frac{2.E.V.\omega_1}{4C^2 + 4\omega_1{}^2} + \frac{1}{4.C.\lambda}(E^2 + W^2)$$

$$+ \frac{(E^2 - \omega^2)(C)}{\lambda(4C^2 + 4\omega_1{}^2)} + \frac{2E.W.\omega_1}{(4C^2 + 4\omega_1{}^2)} \quad .. \quad .. \quad (6.2.7)$$

Consider the system of fig. 6.1.3 where a = 6.6, Y = 71, $\lambda = 0.1$, E = 1.0. The value of the performance index may be divided into two parts, J_e and J_m, where J_e is the contribution due to the error and J_m the contribution due to the control effort.

The Pontryagin approach yields from equation 6.2.4:

$$J_e = \frac{1}{4P}(E^2 + R^2) + \frac{(E^2 - R^2)(P)}{4P^2 + 4\omega^2} + \frac{2E.R.\omega}{4P + 4\omega^2}$$

$$J_m = \frac{\lambda}{4P}(T^2 + U^2) + \frac{\lambda(T^2 - U^2)(P)}{4P^2 + 4\omega^2} + \frac{\lambda.2.T.U.\omega}{4P^2 + 4\omega^2}$$

For initial co-state vector values A and B of 0.1977 and 0.6325 respectively, $J_e = 0.071$ and $J_m = 0.027$

$$\therefore J = 0.098 \quad .. \quad .. \quad .. \quad (6.2.8)$$

The Dynamic Programming approach yields from equation 6.1.7:

$$J_{1_e} = \frac{1}{4C}(V^2 + E^2) + \frac{(E^2 - V^2)(C)}{4C^2 + 4\omega_1{}^2} + \frac{2.E.V.\omega_1}{4C^2 + 4\omega_1{}^2}$$

$$J_{1_m} = \frac{1}{4.C.\lambda}(E^2 + W^2) + \frac{(E^2 - W^2)(C)}{\lambda(4C^2 + 4\omega_1{}^2)} + \frac{2E.W.\omega_1}{\lambda(4C^2 + 4\omega_1{}^2)}$$

For an optimum plant K = 7.02 and $J_{1_e} = 0.061$, $J_{1_m} = 0.068$

$\therefore \; J_1 = 0.129$

The values of J and J_1 are comparable with the values of performance index obtained experimentally, (i.e. 0.098 fig. 5.1.4B , and 0.297 fig. 6.1.3 , respectively.

6.3 Comparison of Pontryagin's Infinite Interval Approach with that of Dynamic Programming

The equations derived for the control effort, system output and hence value of index (section 6.2), appear algebraically to be similar for both systems. Substitution of actual numerical values reveal, however, that the two systems are different. This was demonstrated by the different values obtained when optimisation was effected according to the same index.

Equation 1.3.10 shows that the maximum value of the control effort for a system, with at least one pure integrator, optimised according to Pontryagin as:

$$\hat{m}p = \frac{E}{\sqrt{\lambda}} \quad \ldots \quad (P_2(o) = 2E\sqrt{\lambda}) \quad \ldots \quad .. \quad .. \quad (6.3.1)$$

Equation 6.1.7 gives the maximum value for the control effort when optimisation is achieved according to Dynamic Programming as

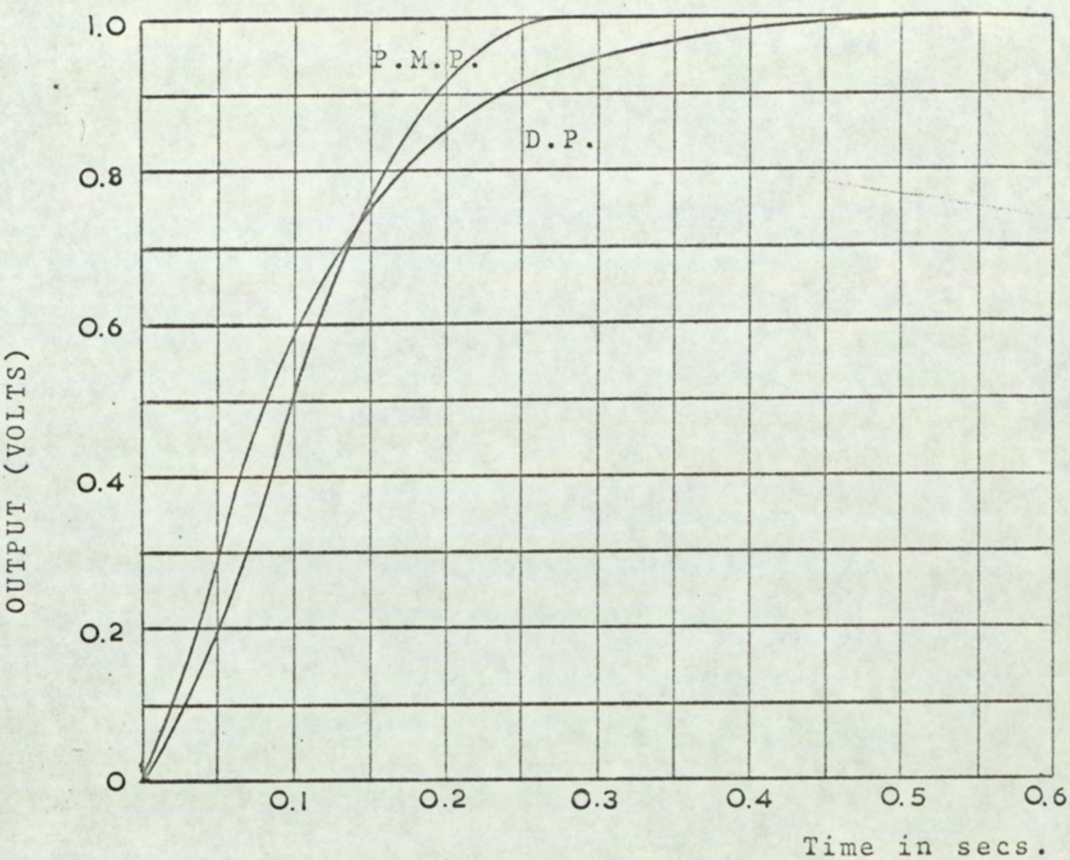
$$\hat{md} = \frac{K_2}{\lambda}$$

$$\therefore \quad \hat{md} = \frac{E}{\lambda} \quad .. \quad .. \quad .. \quad .. \quad .. \quad .. \qquad (6.3.2)$$

(i.e. maximum value occurs at time $t = 0$ when $x_1 = x_i = 0$)

When λ is less than unity, $\hat{md}$ will be greater than $\hat{mp}$. This may be advantageous when optimisation is to be achieved to produce the shortest settling time. To obtain such systems, the maximum value of the control effort would be made equal to the voltage at which the system would saturate. The value of λ to obtain such a system could be directly calculated from equation 6.3.1 or 2. Under such circumstances, λ would be less than unity otherwise the plant output would be required to be greater than the value at which it saturated. The value of the performance index for such plants would be smaller when optimisation was carried out according to the formulae of equations 4.2(b)11 and 13 while maintaining a smaller maximum controlling effort and shorter settling time. The shorter settling time (fig. 6.3.1) is brought about by the absence of overshoot inherent in the controller.

The difference in the value of performance index is emphasised by the evaluation in section 6.2, where it is shown that by optimising to the same index of performance,



Output of Plant $\frac{71}{S(S + 6.6)}$ when Optimised Via Dynamic Programming and the Modified Pontryagin Principle

FIG. 6.3.1

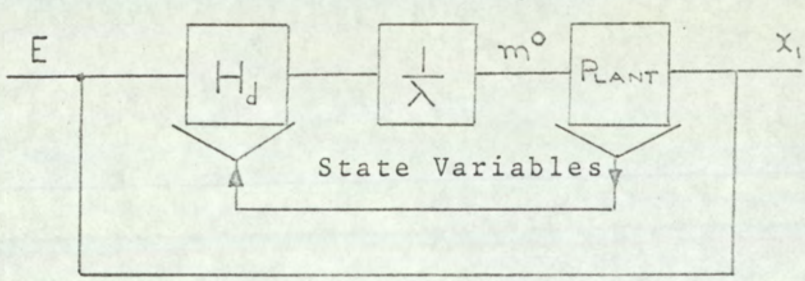
the value is less via the modified Pontryagin Principle. Furthermore, due to the elimination of overshoot, the value of the performance index for a controlled plant (according to the controller of section 5) will be reduced. This will produce a greater difference for the value of performance index than represented in section 6.2. The main reason for the difference may be observed to be in the contribution made by the respective controlling efforts (contribution due to mp = 0.027 and for md = 0.068; section 6.2).

If the system considered saturated at 9 volts (say), then optimisation according to Dynamic Programming would only be achieved by increasing λ above 0.1 (i.e. with at 0.1, md = 10v) and so decreasing md. This may have the effect of increasing the index and would certainly increase the settling time. Saturation may not have been provoked by optimising according to Pontryagin, however, (i.e. with λ at 0.1, mp = 3.16v) and thus λ may be reduced increasing mp, reducing the settling time and value of the index.

Optimisation with λ greater than unity would produce a reverse situation; mp would be greater than md and the value of the index may be greater when using Pontryagin's Principle. It is unlikely, however, that optimisation would be carried out with λ greater than unity as this would produce maximum control efforts (m^o), in both cases, less than the desired output. This would render the plant more sluggish with control than without it.

The most important and most difficult task of optimisation is determining the required index. Once the form of index and hence control strategy has been formulated, the relevant weighting factors have to be obtained. In general, these are obtained by trail and error runs (an exception is the case where the maximum control effort is required).

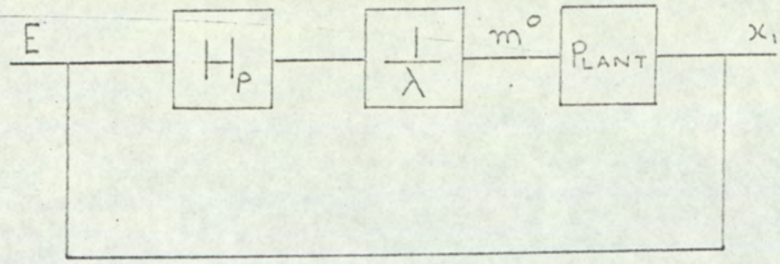
The feedback gains inherent in the procedure of Dynamic Programming are independent of the weighting factor λ (equations 6.1.13, 14 and 15) of the quadratic index considered. The optimum control effort however may be observed to be greatly dependent upon λ (fig. 6.3.2). This enables the required feedback gains to be calculated and the effect of verying λ observed for a particular system without the need to re-calculate the gains and thus the required λ may be readily determined.



Hd: Optimum State Variable Gains

Fig. 6.3.2

Block Diagram of Optimal System designed via Dynamic Programming



H_p : Function Generator

Fig. 6.3.3

Block Diagram of Optimal System designed via Pontryagin Principle

When the system is optimised by 'Pontryagin' and a practical controller constructed from a function generator is implemented, the complete system may be represented in the block diagram form of fig. 6.3.3 where H_p represents the function generator; $^1/\lambda$ inherent in the mathematics not being included in H_p. Unlike H_d of fig. 6.3.2, H_p is not independent of the performance index, i.e. the initial values of the co-state vectors are directly dependent upon λ (equations 4.2(b).11 and 13). It is therefore evident that for each value of λ a new function generator would have to be designed. Selecting the required value of λ may therefore necessitate the running of several different systems on a computer. As the initial conditions of the co-state vectors have been obtained in formulae form, this is not a lengthy process. As the most convenient method of acquiring the correct value of λ for the Dynamic Programming plant would also be via simulation on a computer, this is not a distinct disadvantage.

The figures of 6.3.2 and 6.3.3 demonstrate the sometimes misleading similarity between control effected via Dynamic Programming and Pontryagin's Maximum Principle. Besides the difference in the mathematics for evolving H_p and H_d, the main differences are:-

i) H_p is dependent upon the weighting factors of the index, H_d is not. This may be a contributing factor for the difference in performance index.

ii) Probably the most important advantage Pontryagin has over Dynamic Programming is that only the output of the plant is required for feedback to produce an optimum system and not each state variable.

It has further been shown that the actual feedback vectors that will be required may be determined at the onset by the form of the performance index. This would enable the optimising controller, or adjoint system, to be tailored to individual systems unlike Dynamic Programming which, to a quadratic index, generally demands every state variable to be fedback. It is thus evident that for such systems where it is impossible to monitor every state variable, such as for chemical plants, furnaces and electrical and mechanical systems where the vectors may be encased within the machinery etc., the 'Pontryagin' approach may be used where Dynamic Programming may not. (In such circumstances it would be possible to apply a modified version of Dynamic Programming ref.13 . This method, however, due to lack of system information would generally produce sub-optimal control).

Merriam (ref.20 page 123) states:

"The construction of a control equation possessing feedback around the dynamic process poses an entirely different problem from the numerical problem. In particular, the solution to the condition for minimum error must be structural in a fashion that suggests physical components needed in the construction of the control equation. This structuring of the solution is really a control-system-synthesis problem. On the one hand, the equations of Pontryagin give solutions along a single trajectory and hence the control signals are found as functions of time and not as functions of the state vector and time. In other words, structure and hence a control equation are not obtained. On the other hand, the dynamic programming equation expresses the condition for minimum error for all values of the state vector and hence provides hope for obtaining directly the structure of the control equation."

This statement, given by many authors as an argument against the application of the Maximum Principle, is a little harsh for the optimal control effort as calculated by Pontryagin's Principle may be expressed in terms of its dependent state variables, namely the system output for the index considered, and thus producing a control equation with reference to fig. 6.3.3 of

$$m^{o}(t) = (E - x_{1}(t)) . H_{\rho}/_{\lambda} \quad .. \quad .. \quad 6.3.3$$

where H_{ρ} is the transfer function of the adjoint system. When H_{ρ} is considered to be a function generator, equation 6.3.3 provides directly the structure of the complete optimal system. Furthermore, it has been shown (by virtue of not requiring each state variable for feedback purposes) that not producing the control effort as a function of all the state variables is a great advantage which the Maximum Principle offers over Dynamic Programming.

Roberts (ref.26) demonstrates that the application of Dynamic Programming and Pontryagin's Maximum Principle are identical. As, however, the control equations for Pontryagin's Maximum Principle cannot in general be solved, for the analogue case, in the infinite time interval, this statement cannot be absolutely verified. The comparison of the modified Pontryagin Principle with Dynamic Programming has been shown, both experimentally and mathematically, not to be comparable. Furthermore, Roberts' approach to the infinite time interval would prove extremely difficult to apply to a plant whose model could not be represented as a string of integrators without internal feedback of the state vectors.

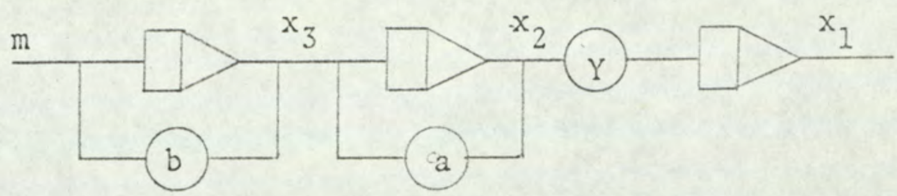
6.4. Comparison with Calculus of Variations

A comprehensive comparison with that of Calculus of Variation was not undertaken as Pontryagin's Principle may be regarded as an extension of the method to take into account vector saturation.

Optimisation by Calculus of Variations (ref.20,32) results in the solution of a Euler-Lagrange equation which entails the solution to a two-point boundary problem. The Euler-Lagrange equation is always unstable and therefor solutions for the infinite interval case may produce control efforts of infinite magnitude.

7. Third Order Plant.

Work on second order plants was extended to third order (fig. 7.1.), implementing an analogue computer. The procedure



Model of third order plant $\frac{Y}{s(s+a)(s+b)}$

FIG. 7.1

for optimisation being to vary the initial conditions of the co-state vectors until they were all zero at the termination of the optimising interval.

In general, this method of optimising could not be performed. This was mainly due to the inability of all three co-state vectors to reach zero magnitude at the same time. In general it was characteristic for one co-state vector to attain zero magnitude before the other two. This state vector, due to the presence of positive exponentials in the adjoint system, would go unstable before the others reached zero magnitude. Further, as shown in section 7.4 the third order plant was observed to be extremely sensitive to variation in the setting of the initial co-state vectors, much more so than that of a second order plant. Instability and saturation of the computer amplifiers was thus possible due to slight error, when the computer was run in its cyclic mode, in the resetting of the initial conditions. The latter problem could have been over-come by not running the computer in its cyclic mode. This however, would have rendered the setting of three initial conditions, by a trial and error technique, impractical.

Dead zones, for the co-state vectors, in the region of zero, were introduced so that once a vector entered the zone it would be clamped. This offered a slight advantage but, in general the dead zone was required to be so large, to overcome the variation in the vectors produced by the error in resetting, that an optimum system was not attained.

A second method of holding the co-state vectors at zero employed was open circuiting the inputs to each relevant integrator when its output attained zero magnitude. This appeared to be more sensitive to the resetting error than the previous method.

The only systems that could be optimised to any degree were those which possessed low gains and whose co-state vectors came to zero within 'short intervals' of one another. Where 'short interval' implies an interval of time where the vectors remained at zero magnitude without going unstable. Such systems produced families of curves when viewed on a scope. This was due to the computer resetting error.

Optimisation of a third order plant was therefore accomplished by digital simulation.

7.1 Formulation of optimising procedure for the infinite time interval

To obtain a method of applying Pontryagin's Maximum Principle to the infinte interval case and to calculate the initial conditions of the P vectors without the necessity of solving a two point boundary value problem, a procedure, similar to that evolved for the second order case (section 4), was used.

Determination of system output

From fig. 7.1.

$$\dot{x}_1 = Yx_2 \quad . \ . \ . \ . \ . \ . \ . \ 7.1.1.$$

$$\dot{x}_2 = x_3 - ax_2 \quad . \ . \ . \ . \ . \ 7.1.2.$$

$$\dot{x}_3 = m - bx_3 \quad . \ . \ . \ . \ . \ 7.1.3.$$

$$\dot{x}_4 = (E - x_1)^2 + \lambda m^2 \quad . \ . \ 7.1.4.$$

$\therefore$ The Hamiltonian will be:

$$H = P_1 . Y . x_2 + P_2(x_3 - ax_2) + P_3 .(m - bx_3) - ((E - x_1)^2 + \lambda m^2)$$

$$m^0 = \frac{P_3}{2\lambda}$$

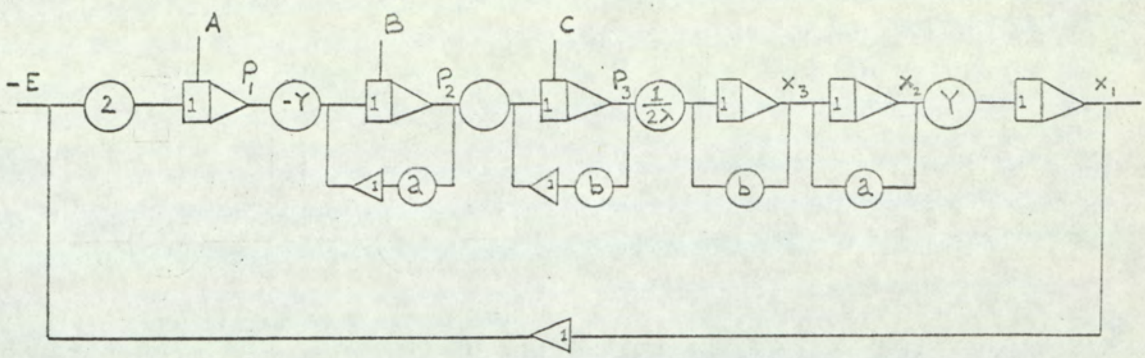
$$\dot{P}i = -\frac{\partial H}{\partial xi}$$

$$\therefore \dot{P}_1 = 2(x_1 - E) \quad . \ . \ . \ . \ . \ . \ 7.1.5.$$

$$\dot{P}_2 = aP_2 - YP_1 \quad . \ . \ . \ . \ . \ . \ 7.1.6.$$

$$\dot{P}_3 = bP_3 - P_2 \quad . \ . \ . \ . \ . \ . \ 7.1.7.$$

Equations 7.1.1.-7 may be mechanised to form an analogue computer diagram as shown in fig.7.1.1. where A,B and C are



Analogue computer diagram for equations 7.1.1.-7.

Fig. 7.1.1.

the initial conditions on vectors P_1 P_2 and P_3 respectively.
From fig. 7.1.1.

$$\left[(x_1(s) - \frac{E}{s}) \frac{2}{s} + \frac{A}{s} \right] \left[(\frac{-Y}{s-a}) + \frac{B}{s-a} \right] \left[(\frac{-1}{s-b}) + \frac{C}{s-b} \right] \frac{Y}{2\lambda s(s+a)(s+b)} = x_1(s)$$

$$\therefore x_1(s) = \frac{1}{s(2\lambda s^2(s^2-a^2)\ (s^2-b^2)-2Y^2)} \left[-2EY^2+AY^2s-BYs^2CYS^2(s-a) \right]$$

$$\therefore x_1(s) = \frac{CY(s^3-s^2(\frac{B}{C}+a) + (\frac{AY}{C})S - \frac{2EY}{C}}{s2\lambda(s^6-s^4(a^2+b^2)+S^2a^2b^2-\frac{Y^2}{\lambda})} \quad \ldots\ldots\ldots 7.1.8.$$

It is important to note that equation 7.1.8.,for the system output in Laplace transform, will vary according to the configuration or original conception of the third order system, i.e. the elements of figure 7.1 may be interconnected in many ways to produce different physical analogue simulations. The output, x_1, of each simulation however, when subjected to indentical inputs, m, would produce identical trajectories only the mode of variation of the other state trajectories would differ. Thus, as for the second order case equation 7.1.8. may take many forms. Each form would possess different values of A and B to produce identical outputs x_1. The simulation of figure 7.1 was chosen as it produces the simplest and most convenient form for $x_1(s)$.

Equation 7.1.8. is analogous to that of equation 4.1.1. The denominator will possess three roots which will give rise to negative exponentials, in the time domain, and three which will give rise to positive exponentials. The numerator will possess three roots the nature of which will be dependant upon the initial conditions A, B or C. If the system is to be stable in the infinite interval case then the zero's of the system, or the roots of the numerator, must cancel with the poles of the system which give rise to positive exponentials. i.e. for a stable system the numerator must be of the form:

$$(s-e)(s-f)(s-q)$$

and denominator of the form:

$$(s^2-e^2)(s^2-f^2)(s^2-q^2)$$

Where e, f and q may be real or complex.

$$(S-e)(S-f)(S-q) = S^3-S^2(e+f+q) + S(ef + eq + qf) - efq \quad . . . 7.1.9.$$

$$(S^2-e^2)(S^2-f^2)(S^2-q^2) = S^6-S^4(e^2+f^2+q^2) + S^2(e^2f^2+e^2q^2+q^2f^2) - e^2f^2q^2 \quad . . . . . . 7.1.10.$$

Equations 7.1.9. and 10 may be observed to be similar to the numerator and denominator respectively of equation 7.1.8. Hence $x_1(s)$ may be expressed as:

$$\frac{S^3-S^2(e+f+q)+S(ef+eq+qf)-efq}{S^6-S^4(e^2+f+q^2)+S^2(e^2f^2+e^2q^2+q^2f^2)-e^2f^2q^2}$$

or $$\frac{S^3-WS^2+XS-Z}{S^6-S^4(W^2-2X)+S^2(X^2-2ZW)-Z^2} \quad . . . . 7.1.11.$$

Where $W = e+f+q$

$X = ef+eq+qf$

$Z = efq$

Comparing like powers of S, in the numerator and denominator, of equation 7.1.8. with equation 7.1.11:

$$W^2 - 2X = a^2 + b^2 \quad \ldots\ldots \; 7.1.12.$$

$$X^2 - 2ZW = a^2b^2 \quad \ldots\ldots \; 7.1.13.$$

$$Z^2 = Y^2/\lambda \quad \ldots\ldots \; 7.1.14.$$

$$Z = \frac{2EY}{C} \quad \ldots\ldots \; 7.1.15.$$

$$W = \frac{B}{C} + a \quad \ldots\ldots \; 7.1.16.$$

$$X = \frac{AY}{C} \quad \ldots\ldots \; 7.1.17.$$

From 15 and 14 : $\frac{Y^2}{\lambda} = \frac{4E^2Y^2}{C^2}$

$$\therefore C = 2E\sqrt{\lambda} \quad \ldots\ldots \; 7.1.18.$$

From 7.1.12, 16, and 17.

$$\frac{B^2}{C^2} + \frac{2Ba}{c} - \frac{2AY}{c} = b^2$$

$$\therefore A = \frac{B^2}{2Yc} + \frac{Ba}{Y} - \frac{b^2c}{2Y} \quad \ldots\ldots \; 7.1.19$$

From 13,14,16 and 17

$$\frac{A^2Y^2}{C^2} - \frac{2Y}{\sqrt{\lambda}}\left[\frac{B}{c} + a\right] = a^2b^2$$

$$\therefore \frac{A^2Y^2}{C^2} - \frac{2YB}{\sqrt{\lambda}.C} - \frac{2Ya}{\sqrt{\lambda}} = a^2b^2$$

$$\therefore A^2 = \frac{2cB}{Y\sqrt{\lambda}} + \frac{2ac^2}{Y\sqrt{\lambda}} + \frac{a^2b^2c^2}{Y^2} \quad \ldots\ldots 7.1.20$$

$$\text{or } B = \frac{A^2Y\sqrt{\lambda}}{2c} - aC - \frac{\sqrt{\lambda}\, ca^2b^2}{2Y} \quad \ldots\ldots 7.1.21.$$

From 7.1.19. and 21.

$$A = \left[\frac{A^4Y\sqrt{\lambda}}{2c} - \frac{\sqrt{\lambda}ca^2b^2}{2Y} - ac\right]\frac{1}{2Yc} + \frac{a}{Y}\left[\frac{A^2Y}{2c} - \frac{ca^2b^2\sqrt{\lambda}}{2Y} - ac\right] - \frac{b^2C}{2Y}$$

$$\therefore A^4 - A^2\left(\frac{2C^2a^2b^2}{Y^2}\right) - A\left(\frac{8c^3}{Y\lambda}\right) + \frac{C^4a^4b^4}{Y^4} - \frac{4C^4}{Y^2\lambda}(a^2+b^2) \quad \ldots 7.1.22.$$

From equations 7.1.20 and 19.

$$\left(\frac{B^2}{2YC} + \frac{Ba}{Y} - \frac{b^2C}{2Y}\right)^2 = \frac{2CB}{Y\sqrt{\lambda}} + \frac{2aC^2}{Y\sqrt{\lambda}} + \frac{a^2b^2C^2}{Y^2}$$

$$\therefore B^4 + B^3(4aC) + B^2(4a^2C^2 - 2b^2C^2) - B\left(4ab^2C^3 + \frac{8YC^3}{\sqrt{\lambda}}\right) + \left(b^4C^4 - \frac{8YC^4a^4}{\sqrt{\lambda}} - 4C^4a^2b^2\right) \ldots\ldots 7.1.23$$

Equation 7.1.18 gives a direct solution for the initial value, C, of the vector P_3. This equation is identical to that of equation 4.2(b).11. which confirms that for a system which contains at least one pure integrator the initial condition of the co-state vector, directly responsible for the control effort m^0, is totally independant of the system parameters. and only dependant upon the elements of the Performance Index. Equation 7.1.18 may therefore be evolved directly from the performance index in exactly the same manner as for the second order system of section 4.3.

Equations 7.1.22 and 23 give quartic equations in the unknown initial conditions for vectors P_1 and P_2 respectively (fig. 7.1.1).

As portrayed for the second order plant a similar identity to that of equation 7.1.11 would have been obtained if the roots given in equations 9 and 10 where complex.

7.2 Determination of initial Conditions

A quartic equation may be solved by pure algebraic manipulations, (appendix 4).

Since equation 7.1.22 was that of a reduced quartic, (quartic with the unknown cubed term missing), its roots were more readily obtainable than those of equation 7.1.23. It was thus decided to obtain the initial value of P_1 by solving equation 7.1.22 and the initial value of P_2 with the use of equation 7.1.21. As depicted in appendix 4 the roots of equation7.1.22 could have been obtained with the aid of a slide rule or desk calculator. The latter producing a more accurate result. However, as a digital computer was available, and for the final analysis its facilities would have to have been sort, the quartic was solved by means of a digital programme. A flow diagram of the required programme is depicted in figure 7.2.1. This programme, appendix 9.5, appears at first sight, to be lengthy for its actual mathematical content. This was necessitated at the onset to prevent the computer from attempting to evaluate mathmatical roots of negative numbers. Further, since initially, the only relevant known requirement of the required root was that it must be real, facility had to be made for all possible real roots to be evaluated. A further requirement of the required root was that it must be positive.

The programme of Appendix 9.5. also calculated the relevant initial conditions for P_2 and P_3.

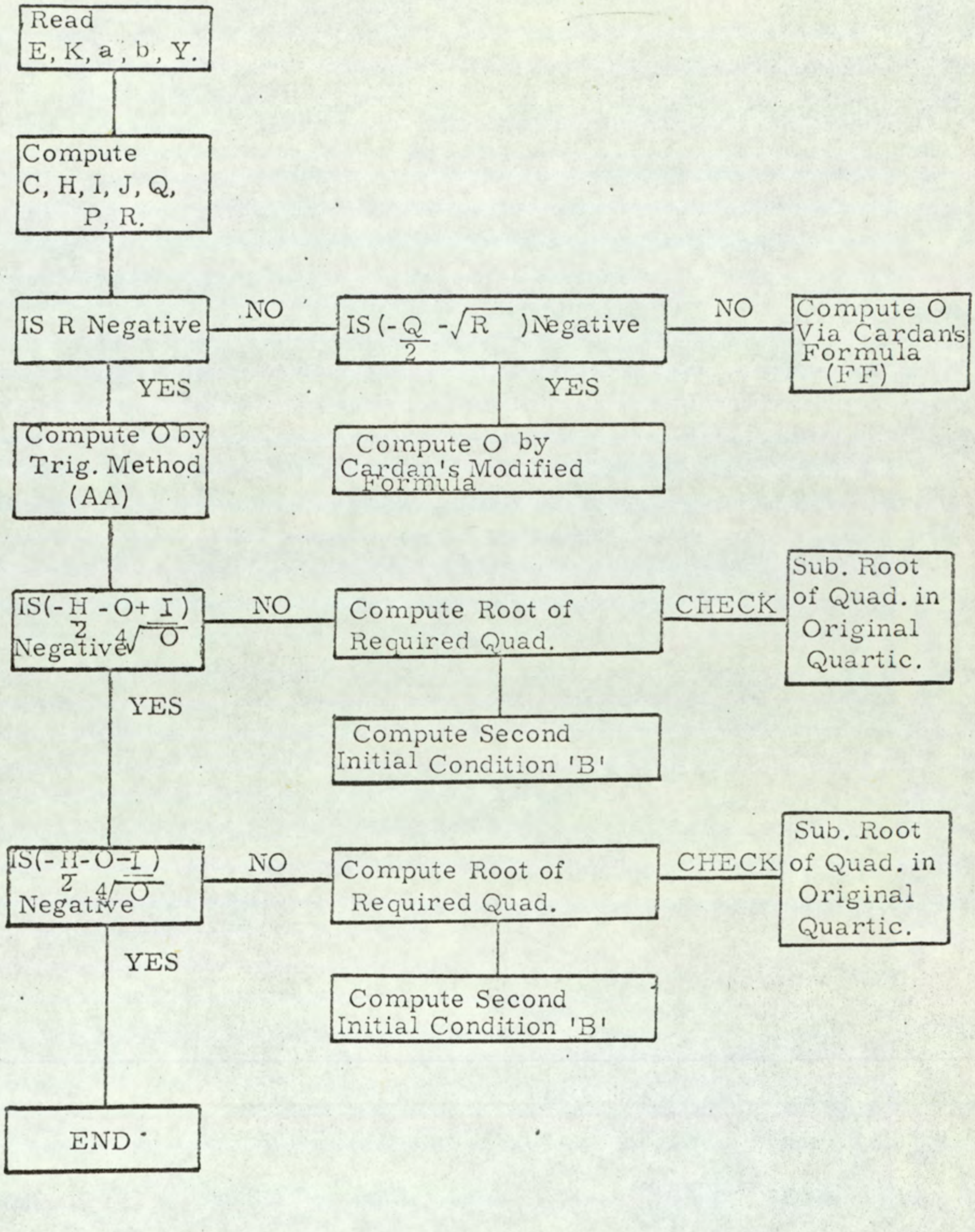


FIG. 7.2.1.

Flow diagram for solution of quartic equation.

A specimen of the results given by the quartic programme are shown in figure 7.2.2. The results show, along with many others that were taken, that equation 7.1.22. always possesses at least two real roots. At least one of these roots is always positive and will produce. a positive value for the initial condition of P_2. This root was always observed to be Z3, (initial value of P_1 or A), which gave rise to Z3B, (initial value of P_2 or B). The digital programme may therefore be shortened so that only root Z3 is evaluated. It should be noted however, that Z3 may be derived by one of two routes, i.e. either by Cardan's Formula or by trigonometric means, either FF or AA. The actual computer calculating time for each set of results, depicted in fig.7.2.2, was less than 30 seconds. (Computation was carried out on an Elliott 803 machine).

7.3 Verification of calculated initial conditions

Verification of the formulae derived in the previous section was achieved by running third order systems such as depicted in figure 7.1.1. (with a=1, b=3 and Y=10) on a digital computer with the initial conditions calculated via the formulae. The results and conclusions drawn from these runs were similar, though more exaggerated, to those obtained for the second order system. i.e. initial region of transient response acceptable, slight overshoot on the output and system unstable in the region where all the vectors should have maintained steady state conditions. A typical set of optimum trajectories, obtained by digital simulation, are shown in figures 7.3.1.A and B.

7.4 Trajectory error in proposed steady state regions

The instability invoked in the proposed steady state regions of the trajectories may be accounted for by similar reasoning to that for the second order system, namely error in integrating

E = 1.0 K = .10 Y = 1.00 a = 1.0 b = 2.0

FF z3 = 3.41386 z4 = -1.54945 c = .632456

EQU3 = .00000149 EQU4 = -.00000006

E3B = 1.88115 z4B = -.432255

E = 1.0 K = .10 Y = 10.0 a = 2.0 b = 1.0

FF z3 = 1.32155 z4 = -.158074 c = .632456

EQU3 = -.00000001 EQU4 = .00000001

z3B = 3.06134 z4B = -1.24244

E = 5.0 K = .10 Y = 40.0 a = .50 b = 3.0

FF z3 = 4.10155 z4 = -.365413 c = .632456

EQU3 = .00000685 EQU4 = .00000393

z3B = 32.0361 z4B = -1.34221

E = 1.0 k = .10 Y = 3.00 a = 3.0 b = .05

FF z3 = 2.13614 z4 = -.865854 c = .632456

EQU3 = .00000013 EQU4 = .00000012

z3B = 1.52419 z4B = -1.33584

E = 1.0 k = .10 Y = 1.00 a = 3.0 b = 2.0

AA z1 = -3.16681 z2 = -3.99307

EQU1 = .00000620 EQU2 = .00000763

z1B = -2.99019 z2B = -1.51121

z3B = 5.31123 z4 = 1.84866

EQU3 = 5.31123 EQU4 = .00000715

z3B = 1.55491 z4B = -4.64298

Results from 'Descarte's solution of the quartic'

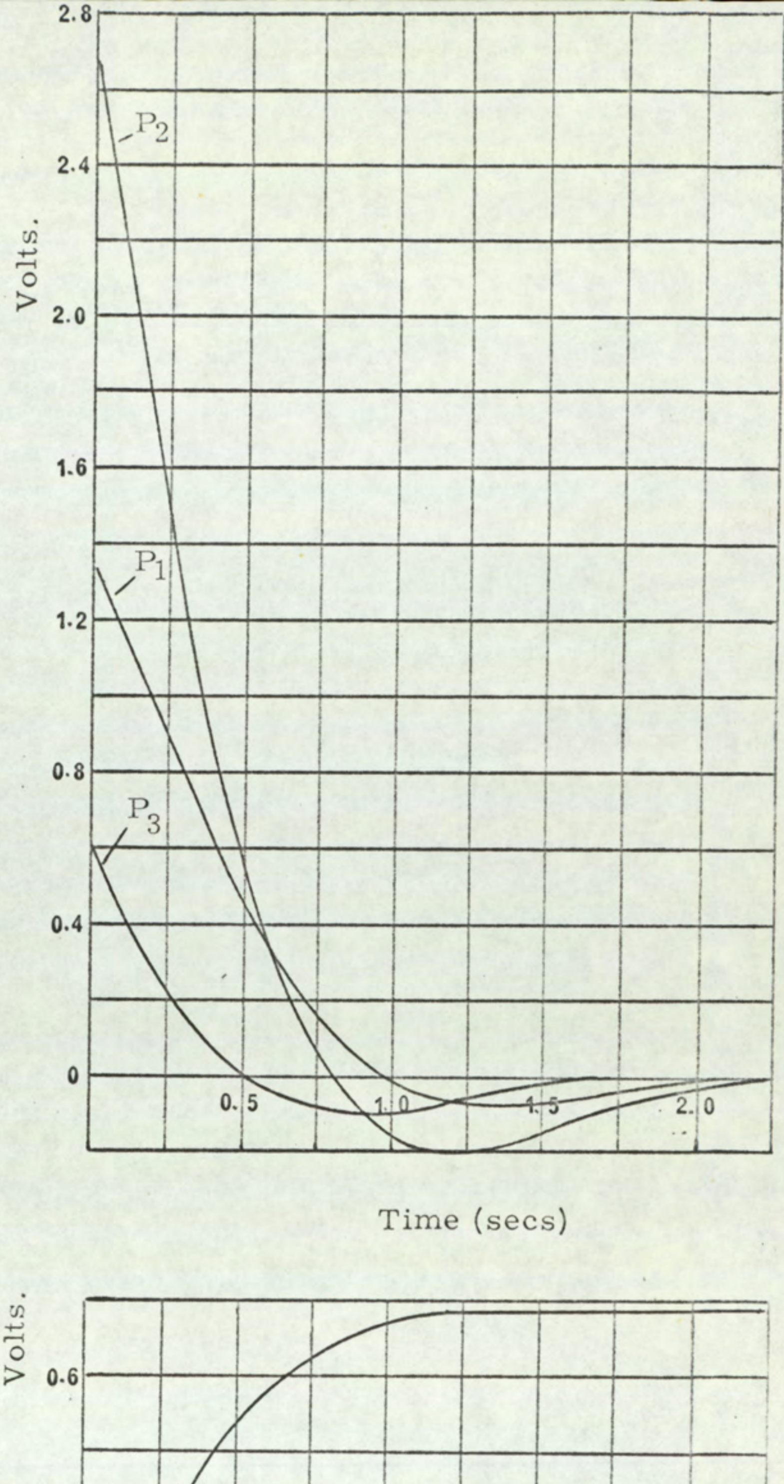
z1 to z4 - initial condition on P_1

z1B to z4B - initial condition on P_2

C - initial condition on P_3

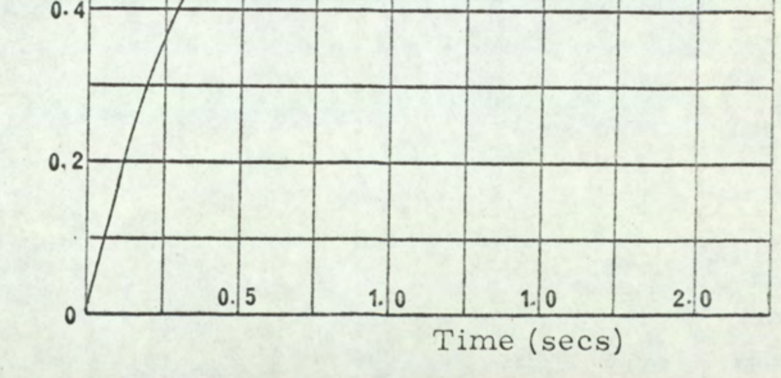
EQU1 to EQU4 value of quartic for factors z1 to z4

Fig. 7.2.2.



Co-state vector

trajectories.

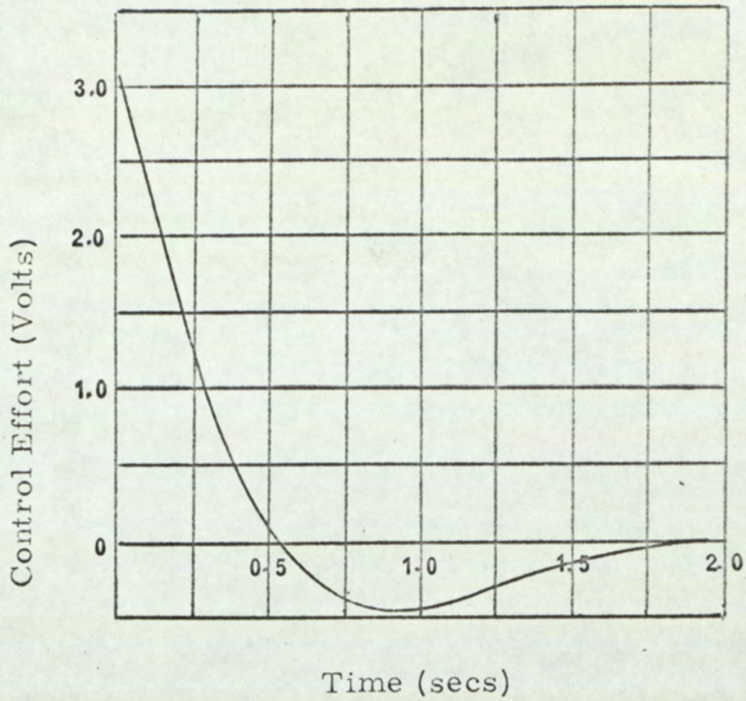


Index

FIG. 7.3.1(A).

Optimum trajectories

for the plant $\frac{10}{S(S+1)(S+3)}$



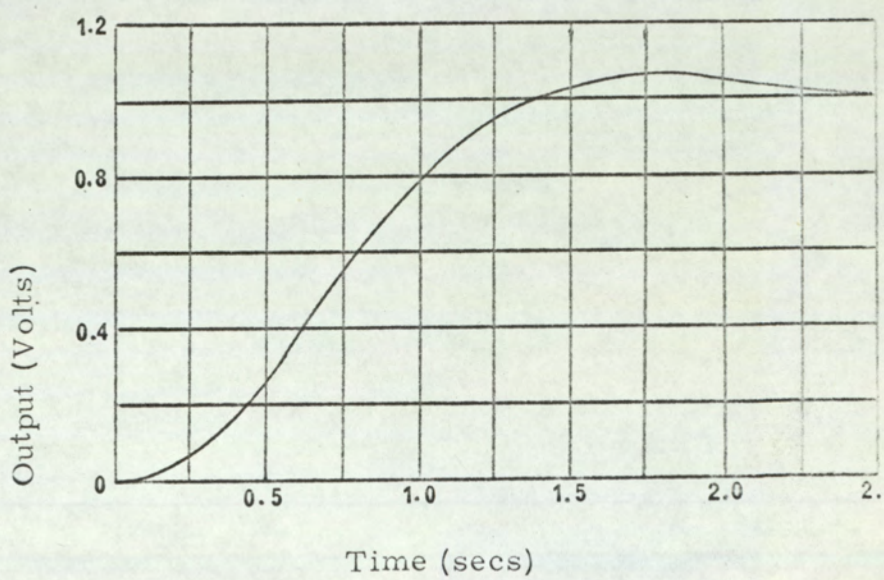


FIG. 7.3.1(B).

Optimum trajectories for a plant with an open open loop transfer function $\frac{10}{S(S+1)(S+3)}$

routine, error in the value of the initial conditions and noise.

Error in the integrating routine was observed not to be the significant factor as reducing the time interval of integration by a factor of ten did not produce a significant change in the system trajectories. The instability must therefore have been due to error in the actual value of the initial conditions and noise. On investigation it was determined that the sensitivity of the vector trajectories was critically dependant upon the value of the initial conditions. This is verified by fig. 7.4.1 which depicts the 'optimum' plant controlling efforts and resulting outputs for different initial conditions. It may be observed that slight adjustment of the sixth significant figure, of one of the co-state vector initial values or even terminating them to six significant figures, produces a totally different vector in the proposed steady state region. It may also be observed that the third order plant was much more sensitive to the actual value of the initial conditions of the co-state vectors than the second order plant. To overcome this inaccuracy in the setting of the initial conditions, the third order system programme was modified so that the initial conditions were first calculated and substituted into the system all on the same programme; i.e. the relevant part of the 'Quartic' programme was introduced at the beginning of the 'Third Order' programme. Figure 7.4.1. (curves 1), shows the results of such a programme, absolute steady state conditions were not obtained and ultimately instability was invoked. Since the controller, to be constructed from these results was only concerned with the initial transient response the region of instability was of no great importance. Further, the plant trajectories from switch on to the point the output attained its initial desired value, (i.e. region over which controller is effective), were not so sensitive to error in the initial values of the co-state vectors.

1. Initial conditions calculated to 8 significant figures.

2. Initial conditions adjusted to 6 significant figures.

A = 3.34748
B = 2.76753
C = 0.632456

3. A = 3.34750
B = 2.76753
C = 0.632456

4. A = 3.34748
B = 2.76750
C = 0.632456

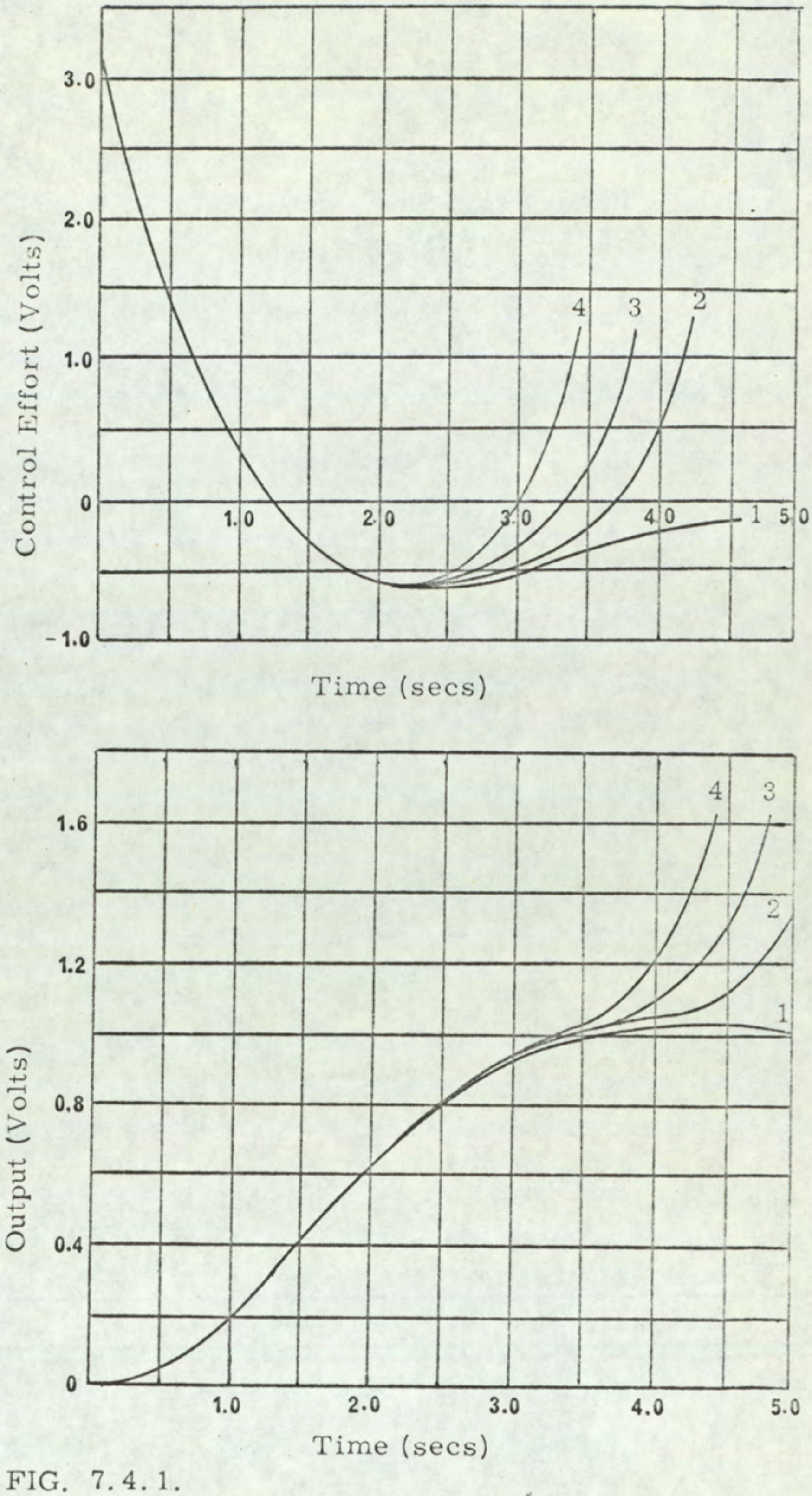


FIG. 7.4.1.

Optimum system trajectories for change in the co-state vector initial conditions. (Plant transfer function $\frac{1}{S(S+3)(S+.05)}$

7.5 Transient Gain of controller

As Pontryagin's Principle is devoted to minimising the value of the performance index a N.O.P., whose gain is sufficiently high as to invoke instability, may be perfectly stable when optimised by the modified Pontryagin approach for the infinite time interval. This is because Pontrygains Maximum Principle will minimize the performance index and this minimum value will not be infinity as for an unstable plant. Figure 7.5.1. depicts the plant output with and without optimising control, (according to equation 7.1.21 and 23), for a normally unstable plant. Since a stable plant when optimised inevitably goes unstable in the prepoised steady state region an originally unstable plant will be even more susceptible to instability. This however, is of no significance as the controller of section 5 allows the transient gain and steady state gain to be different and selected at will.

7.6 Comparison with Dynamic Programming

The state equations for the plant of figure 7.1.are:

$\dot{x}_1 = Yx_2$ 7.6.1

$\dot{x}_2 = x_3 - Ax_2$ 7.6.2

$\dot{x}_3 = m - Bx_3$ 7.6.3

Equations 7.6.1.-3 may be written in matrix rotation:

$$\dot{x}(t) = Bx(t) + Dm(t)$$

Where $B = \begin{bmatrix} 0 & Y & 0 \\ 0 & -A & 1 \\ 0 & 0 & -B \end{bmatrix}$ $D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Plant output
without control

Plant output with and
without control

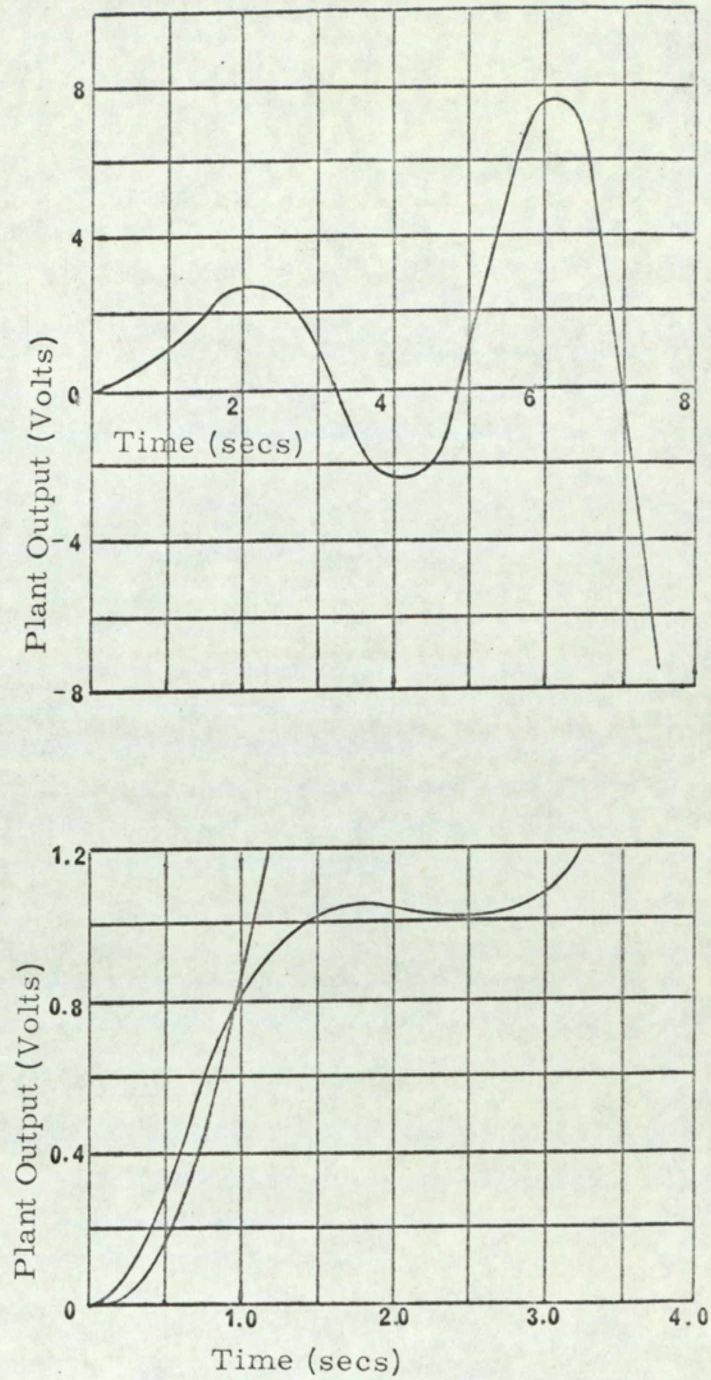


FIG. 7.5.1.

Plant output trajectories with and without optimising control for a plant with an open loop transfer function $\frac{10}{S(S+0.05)(S+3)}$

$$m_i^{\circ}(t) = \frac{dii(t)}{\lambda} \left[b_i(t) - \sum_{m=1}^{N} K_{im}(t)\, x_m(t) \right]$$

$$-\dot{k}_{mk}(t) = \sum_{n=1}^{N} (\alpha_n a_{nm} a_{nk} + b_{nm} k_{nk} + b_{nk} k_{nm} - d_n{}^2 k_{nm} k_{nk})$$

$$-\dot{k}_m(t) = \sum_{n=1}^{N} (\alpha_n x_n^d a_{nm} + b_{nm} k_n - d_n{}^2 k_n k_{nm})$$

Where the a's are elements of the matrix: $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

and the performance index is of the form

$$I = \int_0^{\infty} (\alpha_k (x_1{}^d - x_1)^2 + \lambda m^2)\, dt.$$

where $\alpha_1 = 1$ and $\alpha_2 = \alpha_3 = 0$

The resulting optimum control effort is:

$$\overset{o}{m}_3(t) = \frac{1}{\lambda} (k_3(t) - k_{31}(t)x_1(t) - k_{32}(t)x_2(t) - b_{33}(t)x_3(t)) \quad \ldots . 7.6.4.$$

and the Riccatian equations are:

$$-\dot{k}_{11}(t) = 1 - k_{31}(t)^2 \quad \ldots\ldots\ldots 7.6.5.$$

$$-\dot{k}_{12}(t) = Y.k_{11}(t) - A.k_{21}(t) - k_{31}(t)\, k_{32}(t) \quad \ldots\ldots\ldots 7.6.6.$$

$$-\dot{k}_{13}(t) = k_{21}(t) - B.k_{31}(t) - k_{31}(t)\, k_{33}(t) \quad \ldots\ldots\ldots 7.6.7.$$

$$-\dot{k}_{22}(t) = 2.Y.k_{12}(t) - 2.A.k_{22}(t) - k_{32}(t)^2 \quad \ldots\ldots\ldots 7.6.8$$

$$-\dot{k}_{23}(t) = Y.k_{13}(t) - A.k_{23}(t) + k_{22}(t) - B.k_{32}(t) - k_{32}(t).\, k_{33}(t) \quad \ldots . 7.6.9.$$

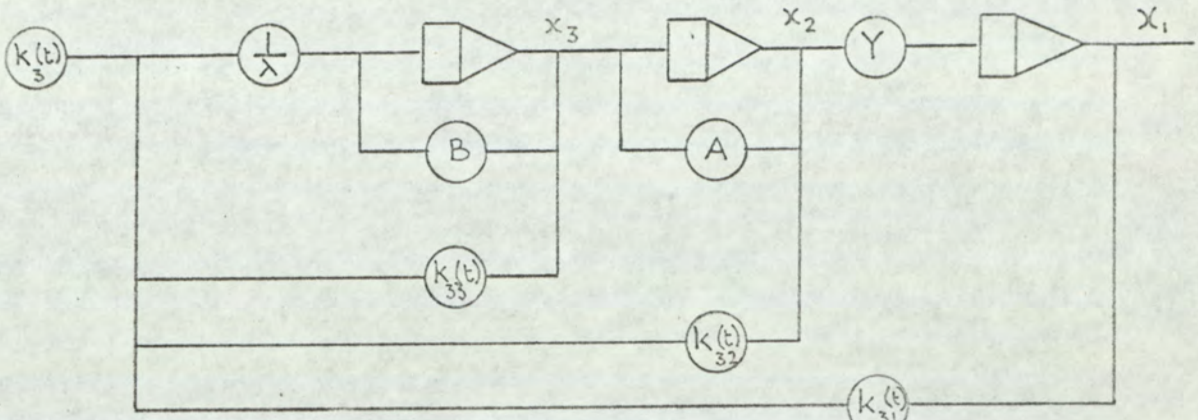
$$-\dot{k}_{33}(t) = 2.k_{23}(t) - 2.B.k_{33}(t) - k_{33}(t) \quad \ldots\ldots\ldots 7.6.10.$$

$$-\dot{k}_1(t) = -k_3(t).k_{31}(t) \quad \ldots\ldots\ldots 7.6.11.$$

$$-\dot{k}_2(t) = Y.k_1(t) - A.k_2(t) - k_3(t).k_{32}(t) \quad \ldots\ldots\ldots 7.6.12.$$

$$-\dot{k}_3(t) = k_2(t) - B.k_3(t) - k_3(t).k_{33}(t) \quad \ldots\ldots\ldots 7.6.13.$$

Equation 7.6.4 produces the optimum system of fig. 7.6.1. Thus, only the values of $K_3(t)$, $k_{33}(t)$, $k_{32}(t)$ and $k_{31}(t)$ are required.



Optimisation of $\frac{Y}{S(S+A)(S+B)}$ according to Dynamic Programming

FIG. 7.6.1.

$k_{ii}(\infty)$ will equal zero, and thus it is possible to solve the Riccation equations, for the required system gains, by pure algebraic means. All nine equations however, are interdependant and thus obtaining such a solution would be a lengthy and tedious process even though the value of k_3 is known to be equal to the required x_1 and k_{31} to be unity (Section 6). Since a digital computer was to have been used to simulate the optimum system it was also used to solve the Riccatian equations. The results for a system with unity input are shown in fig. 7.6.2. The digital programme concerned was that of appendix 9.5.

Similar conclusions were drawn as for the second order system, (section 6), namely: Dynamic Programming approach produced an optimum system whose control effort was much larger, settling time comparable and value of performance index 20% to 30% larger than a system designed by the formulae of section 7.1. The relavant magnitudes of the control effort were identical to those discussed for the second order system.

Plant output

Control effort

Index

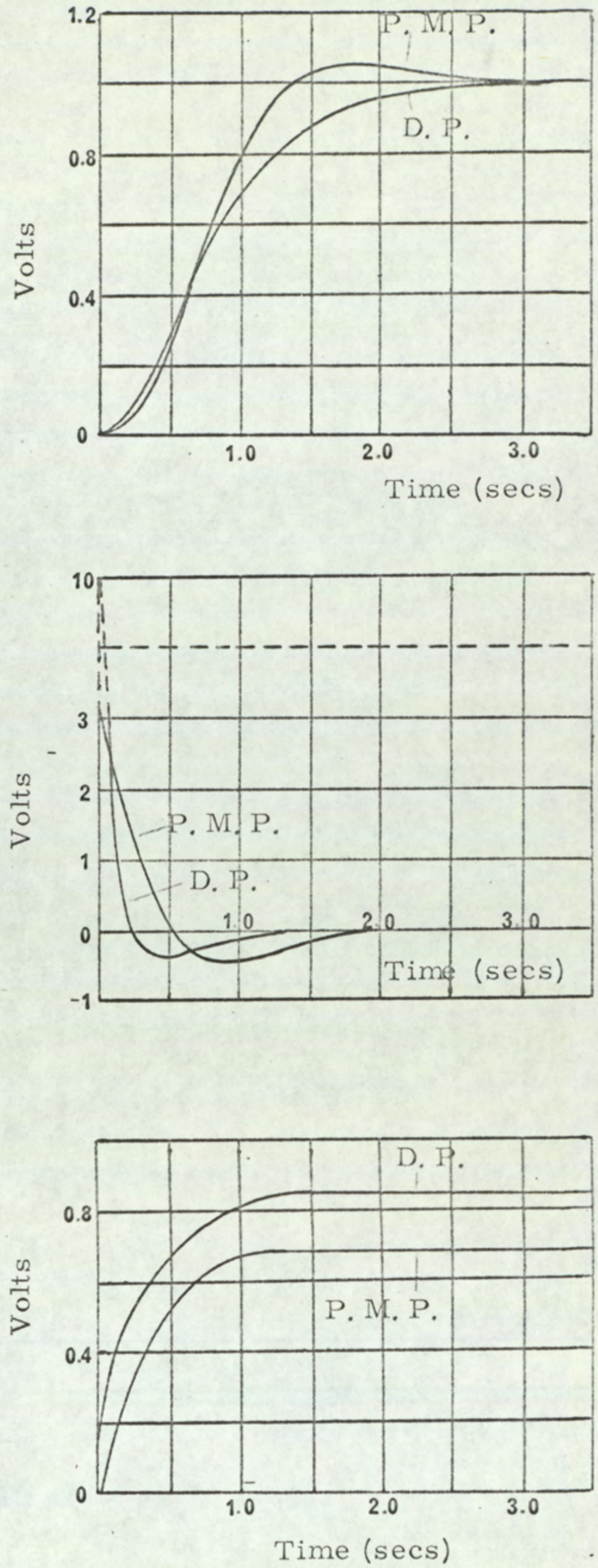


FIG. 7. 6. 2.

Comparison between Dynamic Programming and the modified Pontryagin Maximum Principle. Plant $\frac{10}{S(S+1)(S+3)}$

7.7 Control of an actual third order plant

A third order position control system, similar to the second order system controlled (fig. 5.1.1.) was modelled, again by comparing the output of the actual plant with that of a computer model. The model (fig. 7.7.1.) possessed two variable feedback gains, a and b, and a variable forward gain Y. A good approximation to the actual plant output for a unity step input (fig. 7.7.2.) was observed to be:

$$\frac{600}{s(s+10)\ (s+7.85)} \ . \ . \ . \ . \ . \ . \ . \ 7.7.1.$$

i.e. Y=600, a=10, b=7.85 (fig.7.7.1)

The plant state equations on open loop may be written

$$\dot{x}_1 = 600x_2 \ . \ . \ . \ . \ . \ . \ . \ 7.7.2.$$

$$\dot{x}_2 = x_3 - 7.85x_2 \ . \ . \ . \ . \ 7.7.3.$$

$$\dot{x}_3 = m - 10x_3 \ . \ . \ . \ . \ . \ . \ . \ 7.7.4.$$

The fourth state equation may be written

$$\dot{x}_4 = (E-x_1)^2 + \lambda m^2$$

The Hamiltonian will be given by

$$H = p_1(600x_2) + p_2(x_3-7.85x_2) + p_3(m-10x_3) - ((E-x_1)^2 + m^2\lambda)) \ . . 7.7.5.$$

The resulting co-state vector equations may be written

$$\dot{p}_1 = 2(x_1-E) \ . \ . \ . \ . \ . \ . \ . \ 7.7.6.$$

$$\dot{p}_2 = 7.85p_2 - 600p_1 \ . \ . \ . \ . \ 7.7.7.$$

$$\dot{p}_3 = 10p_3 - p_2 \ . \ . \ . \ . \ . \ . \ . \ 7.7.8.$$

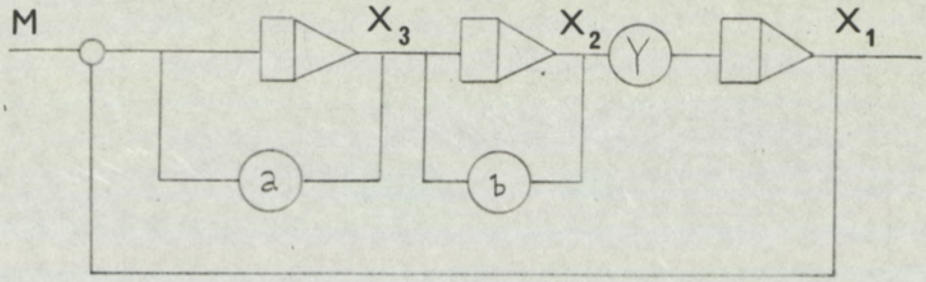


FIG. 7.7.1.

Computer model for third order plant $\frac{600}{S(S+7.85)(S+10)}$.

0.5 volts/cm.

0.5 secs/cm.

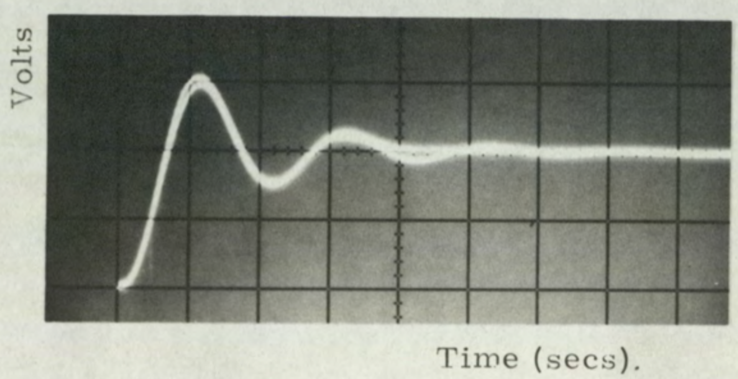


FIG. 7.7.2.

Actual plant and computer model output for the third order plant $\frac{600}{S(S+7.85)(S+10)}$

The optimum control effort may be written

$$m^o = \frac{p_3}{2\lambda} \quad . \ . \ . \ . \ . \ . \ . \ . \ 7.7.9.$$

The optimum system governed by equations 7.7.2.-9 with the required initial co-state vector values as given by equations, 7.1.18, 7.1.22. and 7.1.21, were simulated on a digital computer. The resulting trajectories and value of index are depicted in fig. 7.7.3.

The actual controller was identical to that used for control of a second order plant with the exception of the function generator. For identical reasons to that of the second order case, the function generator was constructed to reproduce as faithfully as possible, (appendix 3), the required m^o for system errors ranging from unity to zero (fig. 7.7.3.).

Fig. 7.7.4. shows the results obtained by controlling the computer model. The resulting value of index (0.16v) again being observed to be less than that obtained from the digital simulation. (0.18v).

The results obtained from the actual plant when controlled via the switched function generator are shown in fig. 7.7.5. with the value of index measured at 0.19v.

Figure 7.7.6. shows the output obtained from a digital simulation when optimisation was effected via Dynamic Programming. The setting time and value of index (0.2513), were observed to be greater than those obtained from the actual plant optimised by the modified Pontryagin Maximum Principles.

Plant Output

Control Effort

and Index

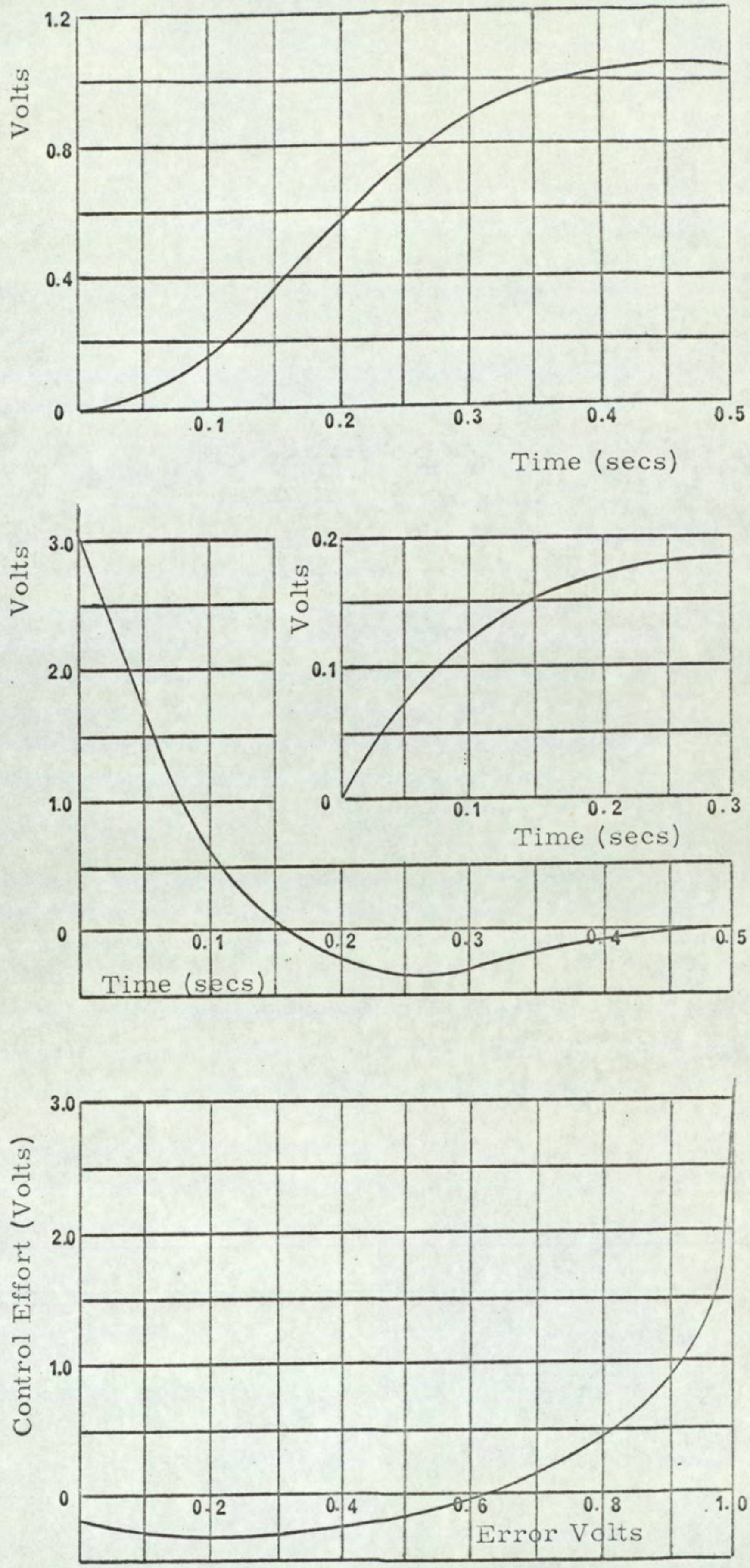


FIG. 7. 7. 3.
Optimum trajectories and function generator curve for plant

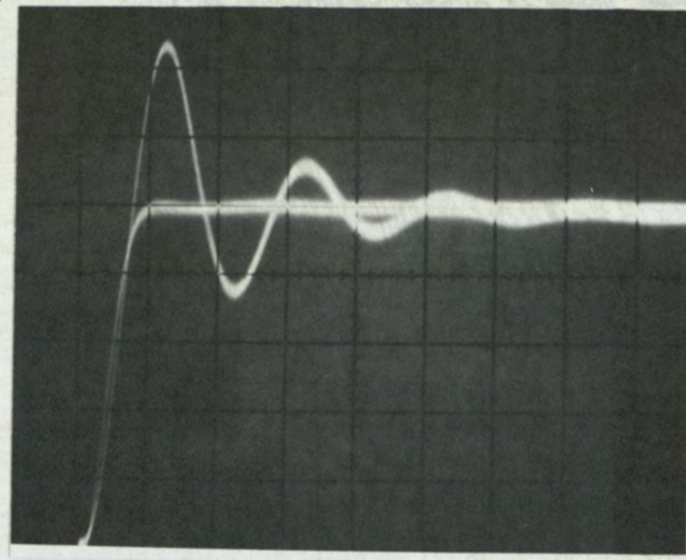
$$\frac{600}{S(S+7.85)(S+10)}$$

Output with and without control

0. 5 Secs/cm

0. 2 Volts/cm

Volts



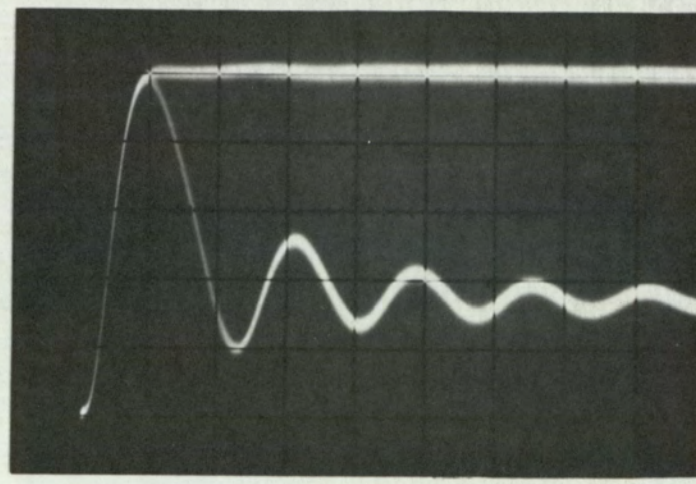
Time (secs)

Output with and without switching

0. 5 Secs/cm

0. 2 Volts/cm

Volts



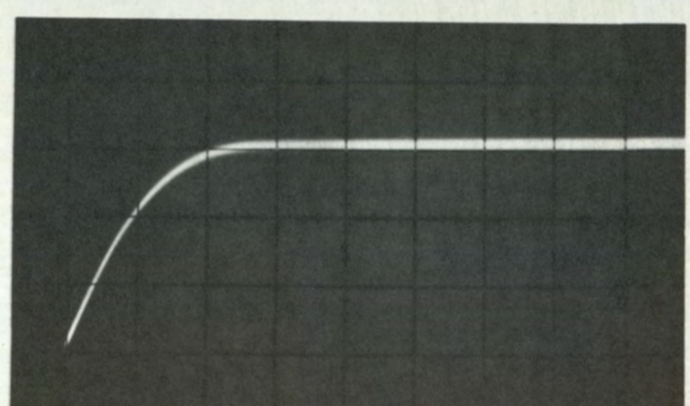
Time(secs)

Index

0. 1 Secs /cm

0. 05 Volts/cm

Volts



Time(secs)

FIG. 7. 7. 4.

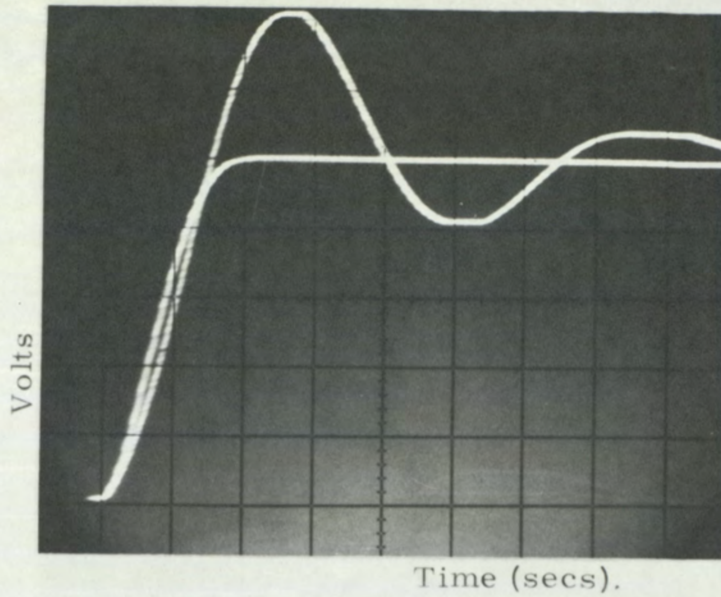
Optimum trajectories of computer model with transfer function

$$\frac{600}{S(S+10)(S+7.85)}$$

Actual plant output with and without control

0.2 secs/cm

0.2 Volts/cm



Index

0.05 Volts/cm

0.2 secs/cm

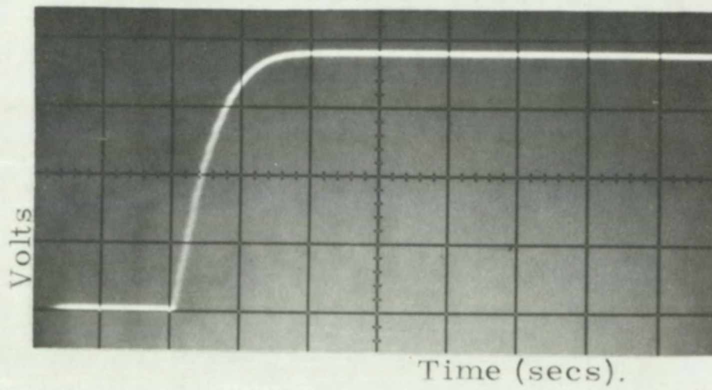


FIG. 7.7.5.

Optimum trajectories for actual plant $\dfrac{600}{S(S+10)(S+7.85)}$

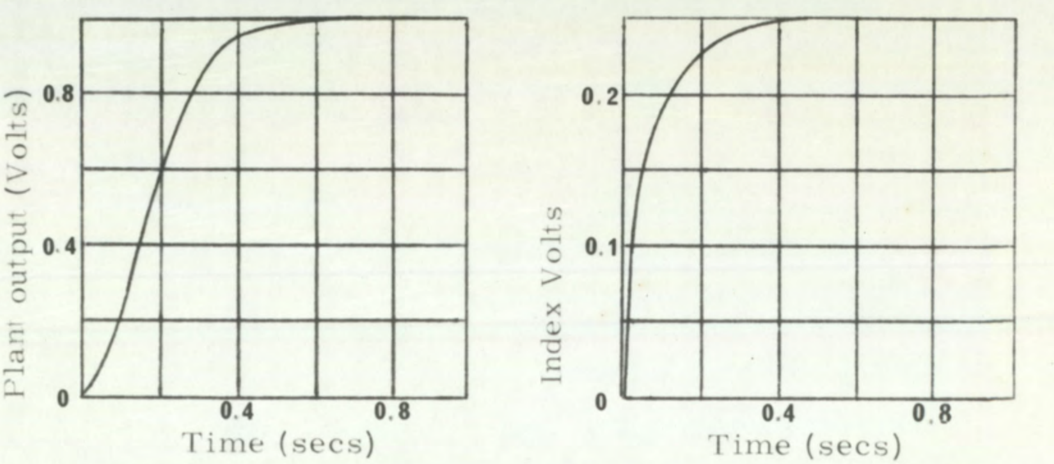


FIG. 7.7.6.

Trajectories of plant with open loop transfer function $\dfrac{600}{S(S+10)(S+7.85)}$ when optimised according to Dynamic Programming.

8. Conclusion

A method of continuous optimisation has been evolved for which the control equations are solved by purely algebraic manipulations. A controller to implement this control has been evolved which consists of a function generator and a small amount of logic switching. The controller is thus cheap to produce and is applicable for the optimal control of plants when operated for the infinite time interval. The control strategy, resulting plant trajectories and value of performance index are more desirable than those obtained by the application of Dynamic Programming (D.P.)

The method was evolved by extending Pontryagins Maximum Principle (P.M.P.), to the infinite interval case and eliminating, or replacing, the characteristic two point boundary value problem with algebraic formulae. The formulated advantages of P.M.P. were maintained, namely:

1) Mathematical manipulations required for producing the control equations are virtually purely algebraic.

2) Type of control (i.e. bang-bang or analogue), is determined by the integrand of the performance index.

3) The state vectors required for feedback purposes are completely dictated by the choice of the performance index.

4) The resulting controller and optimum trajectories are identical for plants which are originally open circuit or possess unity feedback links.

Additional advantages offered by the modified P.M.P. are

1) The maximum control effort is generally smaller than that produced when optimisation is effected by D.P. for a similar performance index.

2) The value of λ,(weight attached to the control effort in the performance index), may be easily calculated to produce the maximum value of control effort without introducing saturation, for any plant whose model possess at least one pure integrator.

The procedure to produce optimal control for the infinite time interval may be categorised:

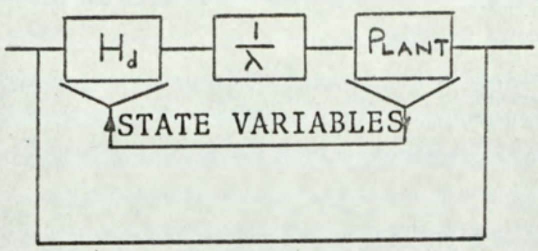
1) Calculate initial co-state vector conditions (sections4,7)
2) Calculate the optimal control effort and error (section 6)
3) Plot curve of control effort against error and piece wise linearise. (appendix 3)
4) Design function generator (appendix 3)
5) Add logic switching to function generator to produce complete controller (section 5)

The popularity of D.P. over P.M.P. for the application of optimisation was mainly due to the reduced computing time required for the solution of the Riccation equations, compared with that required for the solution of the two point boundary value problem of P.M.P. Optimisation by the modified P.M.P. has completely eliminated this computing time.

A disadvantage of D.P. is the requirement that all state vectors must be available for feedback purposes. The fact that only the state vectors appearing in the performance index in quadratic form are required for feedback, according to P.M.P., has not been exploited in the literature.

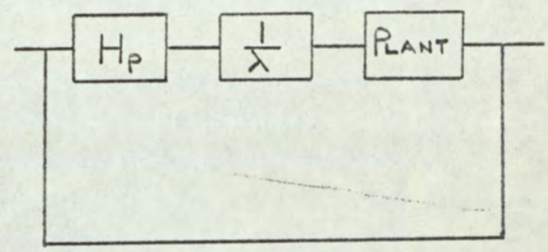
As it is unlikely that the value of λ would be greater than unity, the value of performance index, settling time and maximum value of control effort will generally be smaller for the modified P.M.P. than that obtained by the application of D.P. (It is most likely that λ will be less than unity as λ greater than unity would produce systems which would be more sluggish with control than without it). Besides the obvious advantages of reduction in error and cost, the modified P.M.P. may be used with a smaller value of λ than D.P. without producing plant saturation. This would enable a much shorter settling time to be obtained than possible with D.P.

A possible reason for the improved response is the greater dependance upon the performance index. i.e. the feedback gains of D.P, for the infinite time interval, are totally independant of the performance index. In comparison, the co-state vectors of the modified P.M.P. are totally dependant upon the index. A further reason is the non-linear control effected by the modified P.M.P. i.e. the infinite interval gains of D.P. are constants and Hd (fig. 8.1), consists of constant gains. The comparable controlling



Dynamic Programming

FIG. 8.1



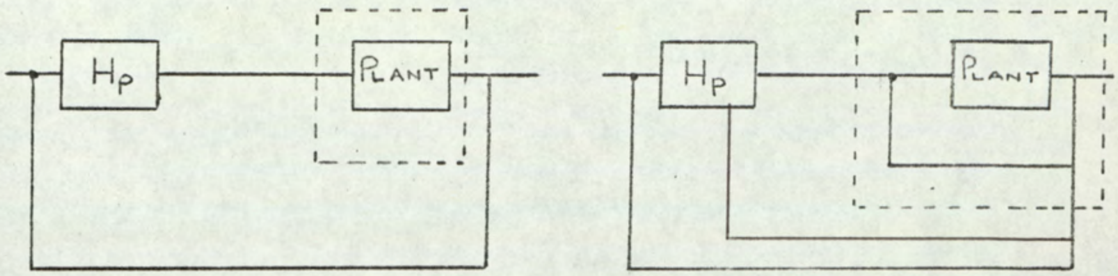
Pontryagins Maximum Principle

FIG. 8.2

element H_p of fig. 8.2 however, is non-linear (transfer function of adjoint system).

The application of the modified P.M.P. to an open loop plant requires exactly the same controller, producing similar trajectories, as when applied to a closed loop plant. This enables optimising control to be implemented on plants normally possessing feedback without removing the feedback link or even stopping the process. (fig. 8.3).

This would be advantageous for chemical plants, large



OPEN LOOP CLOSED LOOP

Control of open and closed loop plants

FIG. 8.3

furnaces and any process where shut down may be an extremely costly or lengthy procedure.

Freeman and Abbott's (ref.10) alternative method for the design of optimal linear systems based on P.M.P. for the infinite time interval has all the disadvantages of D.P. (except that a computer is not required for the solution of the control equations), plus the fact that the optimum plant output has overshoot. The modified P.M.P. evolved by this research offers all the advantages over D.P. without producing overshoot.

9. Appendix

9.1 Steepest Ascent of the Hamiltonian

The equation governing the co-state vector p (equation 1.3.12) is unstable. If, however, this equation is considered in reverse time, it will become stable, i.e. in reverse time equation 1.3.12 will transform to:

$$p_2(\eta) = Y.p_1(\eta) - a.p_2(\eta) \quad .. \quad .. \quad .. \quad (1.9.1)$$

In general, all the co-state vector equations will be unstable if the system being optimised is stable. The state equations for a stable system will of course be stable and therefore may be considered in forward time.

The initial conditions on the state equations will be known while their final conditions will generally not be known. The terminal conditions of the co-state vectors are known in reverse, i.e. their final values will be zero and the determination of their initial values is the object of the two point boundary problem. It is therefore evident that considering the reverse time equations of the co-state vectors, besides producing stable equations, produces equations with known initial conditions of zero and unknown final conditions. This enables the 'p' equations to be integrated in reverse time from known initial condtions and the state equations to be integrated in forward time from known initial condtions.

The object of Pontryagin's Principle is to maximise the Hamiltonian (equation 1.3.5) with respect to the control effort m. This would require a knowledge of both the terminal conditions on the co-state vectors. If, however, an estimate of the unknown terminal condition is made to compute a corresponding value of the control effort m, and a correction made to m so that the Hamiltonian is increased, a more accurate terminal value may be obtained. To achieve this, a law which corrected the value of m at each stage in the numerical integration would be required. Such a law to proceed from the ith to the (i + 1)th stage may be of the form

$$m_{i+1}(t) = m_i(t) + \frac{K\partial H_i}{\partial m} \qquad \ldots \qquad \ldots \qquad \ldots \qquad (1.9.2)$$

Where $m_{i+1} = m_i(t)$ when the correct condition for the p vectors has been attained, i.e. for this condition H will be a maximum and thus $\frac{\partial H}{\partial m}$ will be zero. K is a suitable positive constant.

The sequence of computations for one iteration would then be as follows:

a) Given the ith estimate of m(t) stored in the computer (originally a guess), the equations of the state vectors (x) are integrated with the specified initial conditions and the x-variables are stored. In this case only x_1 is required.

b) Given the stored values of x_1 , the equations

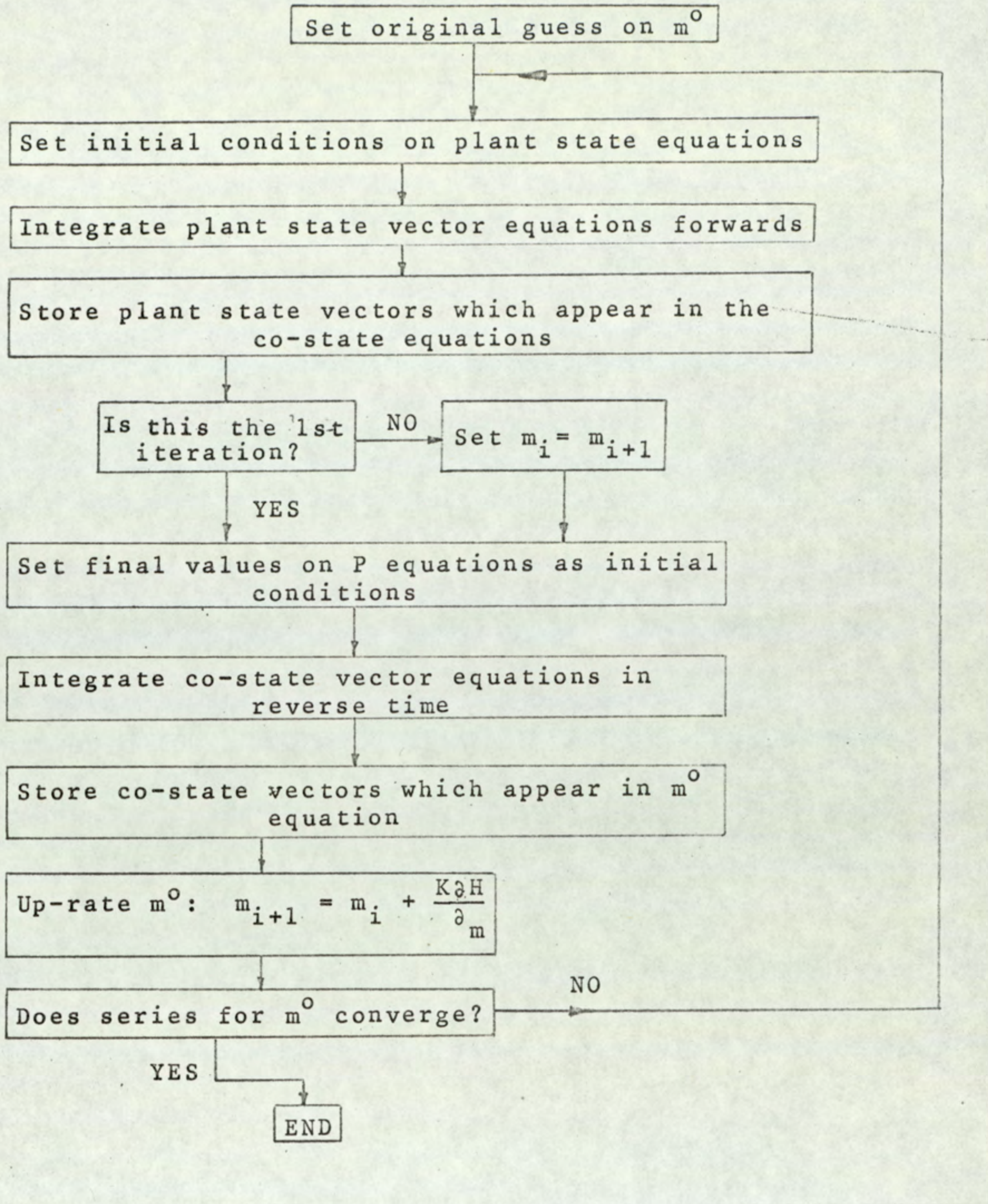


Fig. 9.1.1

Flow chart for 'steepest ascent of the Hamiltonian' for the co-state vectors are integrated in reverse time, viz.

$$p_1 = 2(E - x_1)$$

and $$p_2 = Yp_1 - a.p_2$$

are integrated with the initial conditions $p_1 = p_2 = 0$.

c) At each stage of numerical integration, as p becomes available, m(t) is up-dated according to the law of equation 1.9.2.

The resulting computer programme for the plant whose open loop transfer function was $\frac{1}{s(s + 2)}$ is shown for an optimising period of two seconds. (Appendix 9.5)

9.2 Calculation of Maximum Overshoot

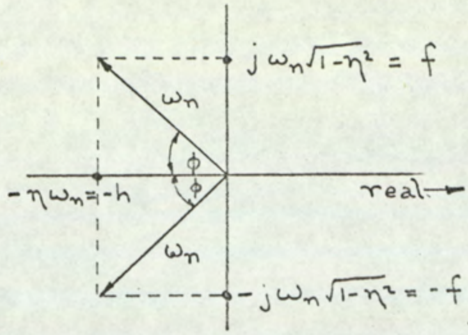
The partial fraction expansion of equation 4.4.4 may be written:

$$x_1(s) = \frac{E}{S} + \frac{K_1}{s + \eta\omega_n - j\omega_n\sqrt{1 - \eta^2}}$$

$$+ \frac{K_2}{s + \eta\omega_n + j\omega_n\sqrt{1 - \eta^2}}$$

where K_1 and K_2 are constants.

The location of the conjugate complex poles in the complex plane will be as in fig. 9.2.1, where $\phi = \cos^{-1}\eta$ or $\sin\phi = \sqrt{1 - \eta^2}$.



Location of Conjugate Complex Poles

Fig. 9.2.1

The time domain response of x_1 will be:

$$x_1(t) = E + \frac{Ee^{-\eta\omega_n t}}{\sqrt{1 - \eta^2}} \; \text{Sin}\left[(\omega_n\sqrt{1 - \eta^2})t - \phi\right] \tag{9.2.1}$$

The frequency of oscillation (ω_o) may be deduced as

$$\omega_o = \omega_n \sqrt{1 - \eta^2}$$

producing a period of oscillation (T_o) of

$$T_o = \frac{1}{f_o} = \frac{2\pi}{\omega_n\sqrt{1 - \eta^2}}$$

The first, and peak overshoot, occurs at $t_o/2$. Therefore, the time to the first overshoot (t_1) may be written

$$t_1 = \frac{\pi}{\omega_n\sqrt{1 - \eta^2}}$$

Substituting for t_1 in equation 9.2.1 produces:

$$x_1(t) = E + \frac{E.e^{-\frac{\eta\pi}{\sqrt{1-\eta^2}}}}{\sqrt{1-\eta^2}} \sin(\pi - \phi)$$

which may be written:

$$x_1(t) = E + E.e^{-\frac{\eta\pi}{\sqrt{1-\eta^2}}}$$

The maximum per cent overshoot may thus be expressed

$$100.e^{-\frac{\eta\pi}{\sqrt{1-\eta^2}}}$$

when = 0.707, the overshoot is given by:

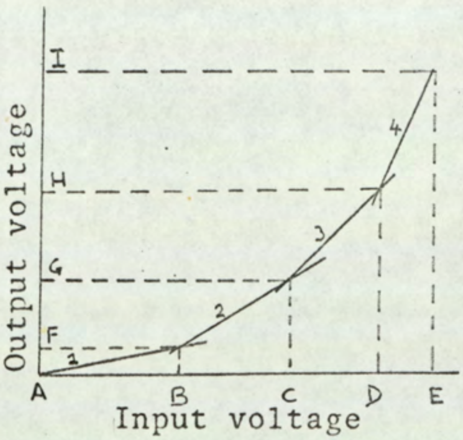
$$100.e^{-\frac{0.707.\pi}{\sqrt{1-.707^2}}}$$

$$= 100.e^{-\pi}$$

$$= \underline{\underline{4.3\%}}$$

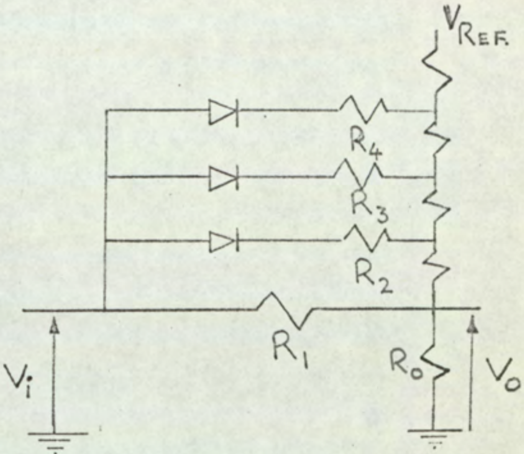
9.3 Function Generation

A continuous function when approximated as being piece-wise linear (fig. 9.3.1.) may be simulated by a non-linear impedance (fig. 9.3.2.). The resulting resistance



Piece wise-linear function

FIG. 9.3.1.



Function generator (Non-linear impedance)

FIG. 9.3.2.

values may be calculated from measurements taken from the linearised curve:

$$\frac{FA}{AB} = \frac{R_o}{R_1 + R_o} \quad \ldots\ldots \quad 9.3.1.$$

$$\frac{FG}{BC} = \frac{R_o}{R_o + R_2'} \quad \ldots\ldots \quad 9.3.2.$$

where $R_2' = \dfrac{R_1 R_2}{R_1 + R_2}$ ...... 9.3.3.

$$\frac{GH}{CD} = \frac{R_o}{R_o + R_3'} \quad \ldots\ldots \quad 9.3.4.$$

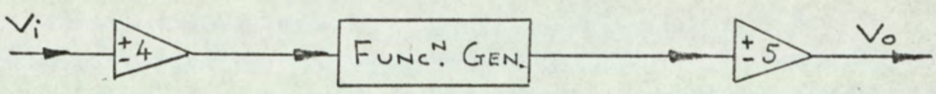
where $R_3' = \dfrac{R_2' R_3}{R_3 + R_2'}$ ...... 9.3.5.

$$\frac{HI}{DE} = \frac{R_o}{R_o + R_4'} \quad . \; . \; . \; . \; . \; . \; . \; 9.3.6.$$

$$\text{where } R_4' = \frac{R_4 R_3'}{R_3 + R_4'} \quad . \; . \; . \; . \; . \; 9.3.7.$$

The required function for the plant $\frac{71}{S(S+6.6)}$ is given in fig. 9.3.3. To accommodate for positive and negative outputs from the same function generator and to enable its gain to be less than unity the arrangement of fig. 9.3.4. was employed.

The function to be reproduced is shown in fig. 9.3.5.



Complete function generator

FIG. 9.3.4.

from which table 9.3.1. was derived

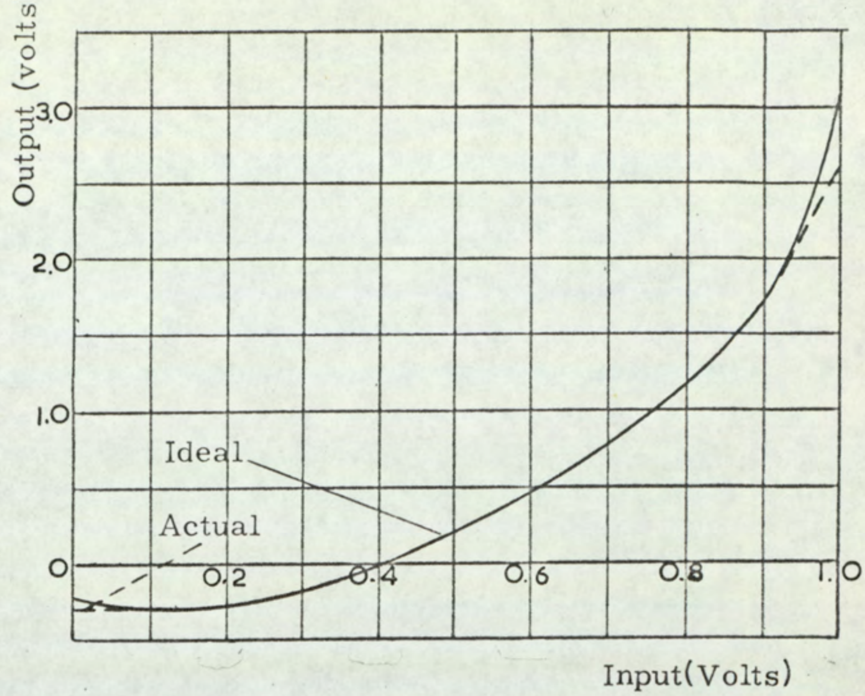
Section	Slope	Brake point
1	0(const.output of -0.6v)	0v
2	0.1	0.96v
3	0.17	2.24v
4	0.33	3.28v
5	0.87	3.76v

Table 9.3.1.

The circuit diagram for the function generator used is given in fig. 9.3.6. To ensure that the potential chain governing the brake points did not constitute extra loading

Typical function generator characteristic

FIG 9.3.3



Linearised Characteristic for an input gain of 4 and an output gain of 5.

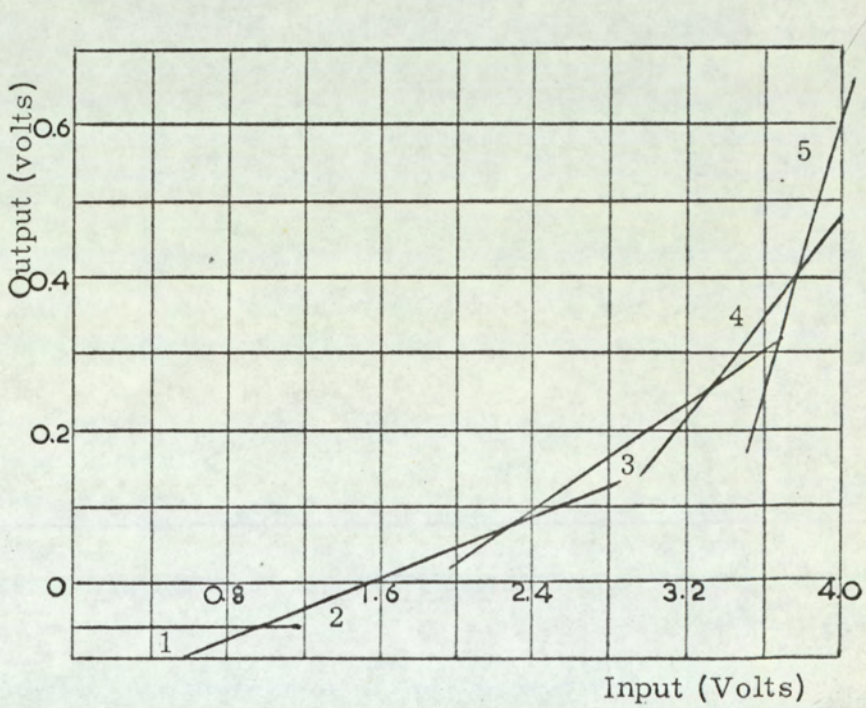
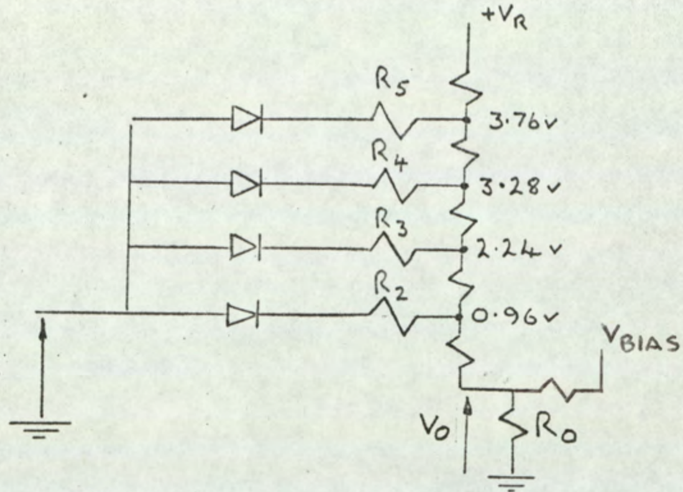


FIG. 9.3.5.

Function generator characteristic.

its total resistance was small (300Ω). An approximate



Function generator

Fig. 9.3.6.

value for the resistors R_2-R_5 were calculated from equations 2.3.1.-7. These were later modified along with slight adjustment of the break points to accommodate the non-linearity of the diodes. Fig. 9.3.3. shows the resulting generated function with error occurring at the extremities of the input. (This error, due to the small amount of power involved, did not produce significant errors in the plant output.)

9.4 Algebraic Solution of Quartic Equations (ref. 8)

Descarte's Solution.

To be solved: $x^4+bx^3+cx^2+dx+e=0$ 9.4.1

By replacing x with $Z - \frac{b}{4}$ we obtain a 'reduced' quartic equation lacking the term Z^3

i.e. $Z^4+qZ^2+rZ+S=0$ 9.4.2

(Equation 9.4.2. is the general form of the quartic in A, i.e. equation 7.1.22.)

Equation 9.4.2. may be expressed as the product of two quadratic factors:

$$(Z^2+2KZ+\ell)(Z^2-2KZ+m) = Z^4+(\ell+m-4K^2)Z^2+2K(m-\ell)Z+\ell m$$

The conditions are

$$\ell+m-4K^2 = q$$
$$2K(m-\ell) = r$$
$$\ell m = S$$

If $K \neq 0$, the first two give

$$2m = q+4K^2+\frac{r}{2K}, \quad 2\ell = q+4K^2-\frac{r}{2K}$$

Then $\ell m = S$ gives

$$64K^6+32qK^4+4(q^2-4s)K^2-r^2=0 \quad . . . . 9.4.3$$

Equation 9.4.3. may be solved as a cubic equation for K^2. Any root $K^2=0$ gives a pair of quadratic factors of equation 2 as

$$Z^2 \mp 2KZ+\frac{q}{2}+2K^2 \pm \frac{r}{4K}$$

The four roots of these two quadratic functionsare the four roots of equation 9.4.2.

Algebraic Solutions of Cubic Equation

If in the general cubic equation:

$$x^3+bx^2+cx+d = 0 \quad . \; . \; . \; . \; . \; . \; . \; . \quad 9.4.4.$$

the substitution $x=y-\frac{b}{3}$ is made then a reduced cubic equation is obtained:

$$y^3+py+q = 0 \quad . \; . \; . \; . \; . \; . \; . \; . \quad 9.4.5.$$

Where $p= c-\frac{b^2}{3}$, $q=d-\frac{bc}{3} + \frac{2b^3}{27}$ . . . . 9.4.6.

If the roots of equation 9.4.5. are y_1, y_2, y_3 then the roots of equation 9.4.4. will be

$$x_1=y_1-\frac{b}{3} , \; x_2 = y_2-\frac{b}{3}, \quad x_3=y_3-\frac{b}{3}$$

Algebraic Solution of a Reduced cubic equation:

Substituting $y = Z - \frac{p}{3Z}$ . . . . 9.4.7.

in equation 9.4.5. produces

$$Z^3 - \frac{p^3}{27Z^3} + q=0$$

$$\therefore Z^6 +qZ^3- \frac{p^3}{27} = 0 \quad . \; . \; . \; . \; . \; . \; . \; . \quad 9.4.8$$

Solving equation 9.4.8. as a quadratic equation in Z^3

$$Z^3= -\frac{q}{2} \pm \sqrt{R}$$

Where $R = (\frac{p}{3})^3 + (\frac{q}{2})^2$

Any number has three cube roots, two of which are the products of the remaining one and either

$$w = -0.5+j0.5.\sqrt{3} \quad \text{or} \quad w^2=-0.5-j0.5.\sqrt{3}$$

Since $(-\frac{q}{2} + \sqrt{R})(-\frac{q}{2} - \sqrt{R}) = (-\frac{P}{3})^3$

particular cube roots may be chosen:

$$A = \sqrt[3]{-q/2 + \sqrt{R}} \quad ; \quad B = \sqrt[3]{-q/2 - \sqrt{R}}$$

such that $AB = -\frac{P}{3}$. The six values of Z may then be written

$$a,\ wA,\ w^2A,\ B,\ wB,\ w^2B$$

These can be paired so that the product of the two in each pair is $-\frac{p}{3}$

$$AB = \frac{P}{3},\ wA.w^2B = -\frac{P}{3}\ ,\ w^2A.wB = -\frac{P}{3}$$

Hence with any root Z another root $-\frac{P}{3Z}$ may be paired. From equation 9.4.7, the sum of the two gives the value y. Thus the roots of equation 9.4.5 are

$$y_1 = A+B\ ,\ \ y_2 = wA+w^2B,\ y^3 = w^2A+wB$$

These are known as Cardan's formulae for the roots of a reduced cubic.

Irreducible Case

When the roots of a cubic equation are all real and distinct, R is negative, so that Cardans formulae present their values in a form involving cube roots of imaginaries. This is called the irreducible case.

Trigonometric Solution of a Cubic Equation

In the irreducible case Cardan's formulae may be avoided. This is based upon the trigonometric identity

$$\text{Cos } 3x = 4\ \text{Cos}^3x - 3\text{Cos}x$$

This may be written in the form

$$Z^3 - \frac{3}{4}Z - \frac{1}{4}\text{Cos}3x = 0 \quad . . . . \ 9.4.9.$$

To transform cubic 9.4.5. into equation 9.4.9, set $y = nZ$. Equation 9.4.5 may then be written

$$Z^3 + \frac{P}{n^2}Z + \frac{q}{n^3} = 0 \quad . . . . \ 9.4.10.$$

The two cubic equations 9.4.9. and 10 are identical if

$$n = \sqrt{\frac{-4P}{3}} \;;\quad \text{Cos}3x = -\frac{q}{2}\cdot\sqrt{\frac{27}{-p^3}}$$

As R<0, p<0 and the value of Cos3x is real and numerically <1 the value of 3x may be obtained from a table of cosines. The three values of Z may thus be written

$$\text{Cos}x,\quad \text{Cos}(x+120^{\circ}),\ \text{Cos}(x+240^{\circ}).$$

Multiplying these by n, the three roots y are obtained.

Construction of digital Programme.

The equation to be solved is:

$$A^4 - A^2\left[\frac{2C^2a^2b^2}{Y^2}\right] - A\left[\frac{8C^3}{Y\lambda}\right] + \frac{C^4a^4b^4}{Y^2} - \frac{4C^4}{Y^2\lambda}(a^2+b^2)$$

This has the general form

$$A^4 - HA^2 + IA + J$$

The deduced cubic may be written (equation 9.4.3.).

$$(64)K^3 + (32H)K^2 + 4(H^2-4J)K - I^2 = 0 \quad . \; . \; . \; . \; . \; . \; . \quad 9.4.11$$

i.e. $K^3 + \frac{H}{2}K^2 + \left(\frac{H^2-4J}{16}\right)K - \frac{I^2}{64} = 0$

This is of the form

$$x^3 + bx^2 + cx + d = 0$$

By setting $x = y - \frac{b}{3}$ or $x = y - \frac{H}{6}$ we obtain the reduced cubic

$$y^3 + Py + Q = 0$$

where $P = c - \frac{b^2}{3}$ $\quad Q = d - \frac{bc}{3} + \frac{2b^3}{27}$

or $P = \frac{H^2-4J}{16} - \frac{H^2}{12}$

and $Q = -\frac{I^2}{64} - \frac{H(H^2-4J)}{96} + \frac{H^3}{108}$

(In the programme x was represented by the letter O)

$$R = (\frac{P}{3})^3 + (\frac{Q}{2})^2$$

If R is negative then

$$N = \sqrt{\frac{-4P}{3}} \quad , \qquad \text{Cos } 3x = -\frac{Q}{2} \div \sqrt{-\frac{P^3}{27}}$$

In the programme this is represented as

$$T = \text{Cos}3x = -\frac{Q}{2} \div \sqrt{\frac{-P^3}{27}}$$

and U = ARCCOS(T), X = $^{U}/3$, V = N.COS(X)

Therefore

$$O = V - \frac{H}{6}$$

and the roots of the reduced cubic are N.Cos(x) producing the roots of equation 9.4.11 as

$$K = N.\text{Cos}(x) - \frac{H}{6} \equiv V - \frac{H}{6} \text{ in the programme.}$$

If R is positive then a root of the reduced cubic is

$$\sqrt[3]{-\frac{Q}{2} + \sqrt{R}} + \sqrt[3]{-\frac{Q}{2} - \sqrt{R}}$$

The roots of the quartic will be given by the roots of:

$$Z^2 + (2L)Z + \frac{H}{2} + 2(O) - \frac{I}{4L}$$

and $Z^2 - (2L)Z + \frac{H}{2} + 2(O) + \frac{I}{4L}$

$$\text{i.e. } Z = -L \pm \sqrt{-\frac{H}{2} - O + \frac{I}{4L}}$$

and $Z = L \pm \sqrt{\frac{-H}{2} - O - \frac{I}{4L}}$

where $L = \sqrt{V - \frac{H}{6}}$

and in the programme $M = O - \frac{H}{2} + \frac{I}{4L}$

$$\text{and} \quad G = O - \frac{H}{2} - \frac{I}{4L}$$

The programme is shown in Appendix 9.5. Additional information to that given for the calculation of the other initial conditions B and C, is also included in the programme.

9.5 Specimen Digital Programmes

```
STEEPEST ASCENT OF THE HAMILTONIAN'

BEGIN REAL IP1,IP2,IX1,IX2,AX1,AX2,M,BX1,CX1,BX2,CX2,FX1,FX2,AP1,AP2,
BP1,CP1,BP2,CP2,FP1,FP2,T,V,N,F'
ARRAY J(1:201),Y(1:201),H(1:201),W(1:201)'
INTEGER U,K,E,B,D,L,Q,C,I,Z'
SWITCH S:=AA,BB,CC,DD,EE,FF'
T:=0.01'
READ V'
PRINT ££L1S7?TIME(SECS)£S8?ITERATION?'
FOR U:=1 STEP 1 UNTIL 20 DO
BEGIN K:=E:=D:=L:=Q:=B:=C:=0'
F:=0'
N:=2.0'
IP1:=0.0'
IP2:=W(1):=0.0'
IX1:=Y(1):=0.0'
IX2:=0.0'
BEGIN
AA: K:=K+1'
F:=F+0.01'
L:=L+1'
C:=C+1'
AX1:=IX2'
IF U=1 THEN M:=1 ELSE M:=J(202-K)'
AX2:=-2*IX2+M'
IF K=1 THEN
BEGIN CX1:=BX1:=AX1'
CX2:=BX2:=AX2'
END'
FX1:=IX1+T*((2*AX1)-(1.5*BX1)+(0.5*CX1))'
FX2:=IX2+T*((2*AX2)-(1.5*BX2)+(0.5*CX2))'
Y(K+1):=FX1'
CX1:=BX1'
BX1:=AX1'
CX2:=BX2'
BX2:=AX2'
IX2:=FX2'
IX1:=FX1'
IF U=20 AND C=1 THEN
BEGIN PRINT££L1S2?T£S8?X1£S8?X2?,££L1??,SAMELINE,ALIGNED(1,3),F,
££S2??,FREEPOINT(6),Y(K+1),££S2??,FREEPOINT(6),FX2'
END'
IF U=20 AND L=20 THEN
BEGIN PRINT££L1??,ALIGNED(1,3),F,££S2??,
SAMELINE,FREEPOINT(6),Y(K+1),££S2??,FREEPOINT(6),FX2'

L:=0'
END'
```

```
IF K LESS 200 THEN GOTO AA
END'
FOR I:=1 STEP 1 UNTIL 201 DO
BEGIN
 H(I):=J(I)'
END'
BB:N:=N-0.01'
D:=D+1'
B:=B+1'
E:=E+1'
Q:=Q+1'
AP1:=-2*(Y(202-E)-1)'
AP2:=IP1-2*IP2'
IF E=1 THEN'
BEGIN CP1:=BP1:=AP1'
CP2:=BP2:=AP2'
END'
FP1:=IP1+T*((2*AP1)-(1.5*BP1)+(0.5*CP1))'
FP2:=IP2+T*((2*AP2)-(1.5*BP2)+(0.5*CP2))'
W(E+1):=FP2'
IF U=1 THEN H(E):=1'
J(E):=H(E)+V*(W(E)-0.2*H(E))'
IF E=200 THEN
BEGIN J(201):=H(201)+V*(W(201)-0.2*H(201))'
END'
IF D=20 THEN BEGIN
IF U=1 THEN GOTO CC'
IF U=5 THEN GOTO CC'
IF U=10 THEN GOTO CC'
IF U=15 THEN GOTO CC'
IF U=18 THEN GOTO CC'
IF U=19 THEN GOTO CC'
END'
GOTO DD'
CC:PRINT££L1S9??,SAMELINE,ALIGNED(1,3),N,££S11??,FREEPOINT(6),J(E)'
D:=0'
DD:IF B=1 AND U=20 THEN
BEGIN PRINT££L1S2?T£S8?P1£S8?P2£S8?M?'
END'
IF D=20 AND U=20 THEN
BEGIN PRINT££L1??,SAMELINE,ALIGNED(1,3),N,££S2??,FREEPOINT(6),FP1,
££S2??,SAMELINE,FREEPOINT(6),W(E+1),££S2??,FREEPOINT(6),J(E)'
D:=0'
END'
IF Q=1 THEN BEGIN
IF U=1 THEN GOTO EE'
IF U=5 THEN GOTO EE'
IF U=10 THEN GOTO EE'
IF U=15 THEN GOTO EE'
IF U=18 THEN GOTO EE'
```

```
IF U=19 THEN GOTO EE'
END'
GOTO FF'
EE:PRINT££L1S26??,SAMELINE,DIGITS(2),U,££L1S9??,SAMELINE,ALIGNED(1,3),N,
££S11??,FREEPOINT(6),J(E)'
FF:CP1:=BP1'
BP1:=AP1'
CP2:=BP2'
BP2:=AP2'
IP1:=FP1'
IP2:=FP2'
ELLIOTT(0,6,0,0,7,0,0)'
ELLIOTT(1,6,Z,0,0,0,0)'
IF Z LESS 0 THEN DUMP'
IF E LESS 200 THEN GOTO BB
END'
END OF PROGRAMME'
```

THIRD ORDER SYSTEM (DYNAMIC PROGRAMMING)'

```
BEGIN REAL C,XD,A,B,Y,T,CB1,BB1,AB1,CB2,BB2,AB2,CB3,BB3,AB3,CB11,
BB11,AB11,CB12,BB12,AB12,CB13,BB13,AB13,CB22,BB22,AB22,CB23,BB23,AB23,
CB33,BB33,AB33,FB1,FB2,FB3,FB11,FB12,FB13,FB22,FB23,FB33,B1,B2,B3,
B11,B12,B13,B22,B23,B33'
INTEGER U,H,L'
SWITCH S:=ZZ'
READ XD,A,B,Y,T,U'
PRINT ££L1?A=?,SAMELINE,FREEPOINT(4),A,££S3?B=?,FREEPOINT(4),B,££S3?Y=?,
FREEPOINT(4),Y,££S3?INPUT=?,FREEPOINT(4),XD,££S3?TIME INTERVAL=?,
FREEPOINT(4),T,££L5?TIME£S8?K3£S8?K31£S8?K32£S8?K33?'
H:=L:=0'
C:=0'
B1:=B2:=B3:=B11:=B12:=B13:=B22:=B23:=B33:=0'
CB1:=BB1:=AB1'
CB2:=BB2:=AB2'
CB3:=BB3:=AB3'
CB11:=BB11:=AB11'
CB12:=BB12:=AB12'
CB13:=BB13:=AB13'
CB22:=BB22:=AB22'
CB23 :=BB23:=AB23'
CB33:=BB33:=AB33'
ZZ: H:=H+1'
C:=C+T'
L:=L+1'
AB11:=1-B13*B13'
AB12:=Y*B11-A*B12-B13*B23'
AB13:=B12-B*B13-B13*B33'
AB22:=2*Y*B12-2*A*B22-B23*B23'
AB23:=Y*B13-A*B23+B22-B*B23-B23*B33'
AB33:=2*B23-2*B*B33-B33*B33'
AB1:=XD-B3*B13'
AB2:=Y*B1-A*B2-B3*B23'
AB3:=B2-B*B3-B3*B33'
FB1:=B1+T*((2*AB1)-(1.5*BB1)+(0.5*CB1))'
FB2:=B2+T*((2*AB2)-(1.5*BB2)+(0.5*CB2))'
FB3:=B3+T*((2*AB3)-(1.5*BB3)+(0.5*CB3))'
FB11:=B11+T*((2*AB11)-(1.5*BB11)+(0.5*CB11))'
FB12:=B12+T*((2*AB12)-(1.5*BB12)+(0.5*CB12))'
FB13:=B13+T*((2*AB13)-(1.5*BB13)+(0.5*CB13))'
FB22:=B22+T*((2*AB22)-(1.5*BB22)+(0.5*CB22))'
FB23:=B23+T*((2*AB23)-(1.5*BB23)+(0.5*CB23))'
FB33:=B33+T*((2*AB33)-(1.5*BB33)+(0.5*CB33))'
```

```
CB1:=BB1'
BB1:=AB1'
B1:=FB1'
CB2:=BB2'
BB2:=AB2'
B2:=FB2'
CB3:=BB3'
BB3:=AB3'
B3:=FB3'
CB11:=BB11'
BB11:=AB11'
B11:=FB11'
CB12:=BB12'
BB12:=AB12'
B12:=FB12'
CB13:=BB13'
BB13:=AB13'
B13:=FB13'
CB22:=BB22'
BB22:=AB22'
B22:=FB22'
CB23:=BB23'
BB23:=AB23'
B23:=FB23'
CB33:=BB33'
BB33:=AB33'
B33:=FB33'
IF L=20 THEN
BEGIN PRINT ££L1??,SAMELINE,ALIGNED(1,2),C,££S4??,FREEPOINT(5),
FB3,££S4??,FREEPOINT(5),FB13,££S4??,FREEPOINT(5),FB23,
££S4??,FREEPOINT(5),FB33'
L:=0'
END'
IF H LESS U THEN GOTO ZZ
END OF PROGRAMME'
```

DESCARTES SOLUTION OF THE QUARTIC'

```
BEGIN REAL E,K,A,B,Y,C,H,I,O,J,Q,P,R,V1,N,T,U,X,V,L,M,Z1,Z2,Z3,Z4,
EQU1,EQU2,EQU3,EQU4,Z1B,Z2B,Z3B,Z4B,G'
INTEGER W'
SWITCH S:=AA,FF,HH,GG,BB,CC,DD,EE'
FOR W:=1 STEP 1 UNTIL 10 DO
BEGIN READ E,K,A,B,Y'
PRINT ££L5?E=?,SAMELINE,FREEPOINT(2),E,££S3?K=?,FREEPOINT(2),K,
££S3?Y=?,FREEPOINT(3),Y,££S3?A=?,FREEPOINT(2),A,££S3?B=?,
FREEPOINT(2),B'
C:=2*E*(SQRT(K))'
H:=-2*((C*A*B/Y)**2)'
I:=-8*((C)**3)/(Y*K)'
J:=((C*A*B)/Y)**4-(4*A*A*((C)**4))/(Y*Y*K)-4*((C**4)*(B**2))/(Y*Y*K)'
Q:=(-H*(H*H-4*J))/96-I*I/64+(H**3)/108'
P:=(H*H-4*J)/16-H*H/12'
R:=(P/3)**3+(Q/2)**2'
PRINT ££L1?Q=?,SAMELINE,FREEPOINT(6),Q,££S3?P=?,FREEPOINT(6),P,
££S3?R=?,FREEPOINT(6),R,££S3?C=?,FREEPOINT(6),C,££L1?H=?,
SAMELINE,FREEPOINT(6),H,££S3?I=?,FREEPOINT(6),I,
££S3?J=?,FREEPOINT(6),J'
IF R LESS 0 THEN GOTO AA ELSE GOTO BB'
AA: N:=SQRT(-(4*P)/3)'
T:=(-Q/2)/SQRT((-P**3)/27)'
U:=ARCCOS(T)'
X:=U/3'
V:=N*COS(X)'
O:=V-H/6'
PRINT ££L1?AA?'
GOTO CC'
BB: V1:=-Q/2-SQRT(R)'
IF V1 LESS 0 THEN GOTO FF ELSE GOTO GG'
FF: PRINT ££L1?FF?'
V:=(-Q/2+SQRT(R))**(1/3)-(-V1)**(1/3)'
GOTO HH'
GG: V:=(-Q/2+SQRT(R))**(1/3)+(-Q/2-SQRT(R))**(1/3)'
HH: O:=V-H/6'
CC: L:=SQRT(O)'
M:=-H/2-O+I/(4*L)'
IF M LESS 0 THEN GOTO DD'
Z1:=-L+SQRT(M)'
EQU1:=Z1**4+(Z1**2)*H+Z1*I+J'
Z1B:=Z1*Z1*Y*(SQRT(K))/(2*C)-A*C-(C*A*A*B*B*SQRT(K))/(2*Y)'
Z2:=-L-SQRT(M)'
EQU2:=Z2**4+(Z2**2)*H+Z2*I+J'
Z2B:=Z2*Z2*Y*(SQRT(K))/(2*C)-A*C-C*A*A*B*B*(SQRT(K))/(2*Y)'
```

```
PRINT ££L1?Z1=?,SAMELINE,FREEPOINT(6),Z1,££S6?Z2=?,FREEPOINT(6),Z2,
££L1?EQU1=?,SAMELINE,FREEPOINT(8),EQU1,££S6?EQU2=?,FREEPOINT(8),
EQU2,££S3?Z1B=?,FREEPOINT(6),Z1B,££S3?Z2B=?,FREEPOINT(6),Z2B'
DD: G:=-O-H/2-I/(4*L)'
IF G LESS 0 THEN GOTO EE'
Z3:=L+SQRT(G)'
EQU3:=Z3**4+(Z3**2)*H+Z3*I+J'
Z3B:=Z3*Z3*Y*(SQRT(K))/(2*C)-A*C-C*A*A*B*B*(SQRT(K))/(2*Y)'
Z4:=L-SQRT(G)'
EQU4:=Z4**4+(Z4**2)*H+Z4*I+J'
Z4B:=Z4*Z4*Y*(SQRT(K))/(2*C)-A*C-C*A*A*B*B*(SQRT(K))/(2*Y)'
PRINT ££L1?Z3=?,SAMELINE,FREEPOINT(6),Z3,££S6?Z4=?,FREEPOINT(6),Z4,
££L1?EQU3=?,SAMELINE,FREEPOINT(8),EQU3,££S6?EQU4=?,FREEPOINT(8),EQU4,
££S3?Z3B=?,FREEPOINT(6),Z3B,££S3?Z4B=?,FREEPOINT(6),Z4B'
EE: END'
END OF PROGRAMME'
```

9.6 Suggestions for further work

The value of the initial state vectors producing optimisation when plotted against the optimizing interval, produce regular curves. The general shape of these curves resembling exponentials. Equations governing their maximum values have been attained. It is therefore suggested that the equations governing the complete curve may be determined to replace the two point boundary value problem of Pontryagin's Maximum Principle for the finite time interval.

A method of obtaining these equations may be by substitution of the required vectors into the performance index as in section 6. The minimum value of this integral, in general terms, being acquired and the equations governing the initial co-state vectors to achieve this extracted.

9.7. REFERENCES

1. Bellman, R. 'Dynamic programming', Princetown University Press. 1957.
2. Brennan, P.J. 'Use of an analogue computer in the application of Pontryagin's Maximum Principle to the design of control systems with optimum transient response'. J of Electronics and control 1963, pp. 345-352.
3. Bryson, A.E. and Denham, W.F. 'A steepest ascent method for solving optimum programming problems'. J.appl.Mech. 1962. 29. (No. 2), p.247.
4. Chang, J.'A problem in the synthesis of optimal systems using the maximum principle' Automat i Telemekh 1961 22, pp. 1302-1308.
5. Chang, S.S.L. 'Synethesis of optimum control systems' McGraw-Hill series 1961.
6. Chestnut, H. 'Bridging the gap in control status 1965' Trans. I.E.E.E. 1965 p.125.
7. Desoer, C.A. 'Pontryagins Maximum Principle and the principle of optimality' J. Franklin Inst. vol. 271, pp. 361-367, May 1961.
8. Dickson, L.E. 'Elementary theory of equations' John Wiley and Sons Inc. 1914.
9. Ellert and Merriam 11. 'Synthesis of feedback controls using optimization theory - An example'. Trans. I.E.E.E. Automatic Control 1963 pp. 89-103.
10. Freeman, E.A. and Abbott, K.M. 'Design of optimal linear control systems with quadratic performance indices.' Proc. I.E.E. vol. 114 No. 8 August 1967, pp. 1180-1183.

11. Elgerd, O. 'Control systems theory'. McGraw-Hill series 1967.

12. Grensted, P.E.W. and Fuller, A.T. 'Minimisation of integral square-error for non-linear control systems of third order and higher order.' Internat. J. Control 1965, 2, pp. 33-73.

13. Hosking, K.J.B. 'Dynamic programming and synthesis of linear optimal control systems.' Proc. I.E.E. vol. 113, No. 6, June 1966, pp.1087-1090

14. Kelley, H.J. 'Method of gradients' chap. 6, Optimization techniques (ed. Seitman) 1962 Academic Press, New York.

15. KOIVUNIEMI, A.J. 'A computational technique for the design of a specific optimal controller. Trans. I.E.E.E. vol. AC-12 No. 2 April 1967.

16. McCann, M.J. 'Variational methods for optimal control' Summer school on the mathematics of control, Hatfield College of Technology June 1961.

17. MASUBUCHI, M. KANOH, H. 'An optimal control system having no unique solution', I.F.A.C. 1965 Paper Sept.

18. MASUBUCHI, M., SEKIGUCHI, T., KANOH, H., KAWASHIMA, Y., MATSUI, M. 'Determination of the unique optimal trajectories in the second order optimal control systems', I.F.A.C. 1966 Paper 2KG.

19. Meditch, J.S. 'The Pontryagin Maximum Principle and some of its applications'. Advances in control systems (ed. C. T. Leondes), 1964.

20. Merriam, C.W. 'Optimisation theory and the design of feedback control systems', McGraw-Hill series 1964.

21. MIURA, TSUDA and IWATA, 'Hybrid computer solutions of optimal control problems by the maximum principle Trans. on Electronic Computers I.E.E.E. Oct. 1967 pp.666-670.

22. Noton, A.R.M. 'Optimal control and the two-point boundary problem, Advances in Automatic Control 1965 pp. 172-179.
23. Oldenburger, R. 'Optimal Control', Published by Holt, Rinehart and Winston Inc. 1966.
24. Pontryagin, L.S. 1957, Session of A.N.U.S.S.R. on fundamental problems of automatic control, 2, 107 (Moscow. A.N.U.S.S.R.).
25. Pontryagin L.S., Baltyanskii, V.G., Garnkrelidze, R.V. and Mischenki, E.F. 'Mathematical theory of optimal processes', Interscience Publishers, New York 1962.
26. Roberts, A.P. 'The optimum principle.' Control 1963 July pp. 10-13, August pp.75-78.
27. Roberts, S.M. and Shipman, J.S. 'Some results in the two point boundary value problem', I.B.M. J. July 1967 pp. 383-388.
28. Rosonoer, L.I. 'L.S. Pontryagin's Maximum Principle in the theory of optimum systems,' Automation and Remote Control, vol. 20, pp. 1288-1302, pp. 1405-1421, pp. 1517-1532, June, July, August, 1960.
29. Steel, G.K. 'Programming of digital computers for transient studies in control systems', I.E.E. Educ. vol. 3, pp. 261-278 (Pergamon Press 1965).
30. Sutton, R.W. 'Optimal control of a single-strand cold rolling mill', I.F.A.C. 1966, Paper 17c.
31. Sutton, R.W. and Tomlinson, N.R. 'Adaptive computer control of simulated second-order system', I.Mech.E. U.K. A.C. 1966.
32. Tou, J. 'Modern control theory', McGraw-Hill series 1964.
33. Westcott, J.H. 'Survey: The status of control theory I.F.A.C. 1966.

34. Westcott, J.H. 'An exposition of adaptive control', Pergamon Press 1962

35. Wilde, J.D. and Beightler, C.J. 'Foundations of optimization', Prentice Hall, Inc. 1967.

36. ZAHRADNIK, R.L., Archer D.H., Rothfus, R.R. 'Dynamic optimization of a cyclic catalytic gas reformer. I.F.A.C. 1966, paper 5A.

37. Elgerd,O. 'Control systems theory' McGraw-Hill series 1967.