ASPECTS OF SELF-ADAPTIVE CONTROL USING PARAMETER PERTURBATION TECHNIQUES

by A.J.R. Pawley, B.A. (Oxon)

A Thesis for Submission
to the University of Warwick
for the Degree of Doctor of
Philosophy

April 1969

ABSTRACT

ASPECTS OF SELF-ADAPTIVE CONTROL USING PARAMETER PERTURBATION TECHNIQUES

This thesis is concerned with the performance of a method of simultaneous identification and optimization of a control system, employing a parameter perturbation technique.

The work is divided into four parts:

(i) The evaluation of several techniques for the identification and gain-estimation of plants having one "input" and one "output", in the presence of noise, where the "input is a controllable parameter, and the "output" is a criterion of the cost of the plant, which is some function of the plant variables. The slope of the cost-function is then given by the "gain" of the plant.

(ii) The optimization of plants having more than one dynamic path between input and output.

(iii) The optimization of multi-parameter plants.

(iv) A general consideration of optimization strategies.

The first part of the thesis is based on the well known technique of obtaining the step response of a plant by applying a pseudo-random binary sequence (or "chain-code") at the input, cross-correlating this with the output, and integrating. A technique equivalent to this, using a multiplier and running-averager, is analysed, and it is shown that significant errors can be caused by any d.c. - bias inherent in the plant output. The performance of the system when band-limited noise is present at the output of the plant is determined, and it is found that if the parameters of the

identification process are chosen so as to reduce the variance of the gain-estimate to a minimum, the error due to d.c.-bias may be large. Practical results are given, which support the theory.

An alternative scheme is considered which enables the d.c. error to be effectively removed. Various methods of implementing this scheme in practice are proposed for use with and without a digital process-control computer. The scheme is analysed under the same noise conditions as before. The signal and noise components of the gain-estimate are evaluated in terms of the parameters of the system, and a design procedure for the choice of optimal values for these parameters is formulated. A method of gain-estimation using sine-wave perturbations is then analysed under similar noise conditions, and compared with the chain-code method.

The second part of the work is devoted to the optimization of plants having more than one dynamic path between input and output, with a different cost-function in each path. It is shown that the estimated position of the optimum varies with perturbation frequency when using sine-wave perturbation, ~~whereas the true optimum is found when using~~ or chain-codes. The theory is supported by the experimental analysis of a simplified model based on a steam-generating plant.

In the third part, various methods of identification of multi-channel plants are considered, based on the use of chain-codes. The extent of cross-coupling between channels is examined, and ways of reducing this are evaluated. It is found that the best solution is to use time-shifted versions of the same chain-code, applied to the different inputs of the plant. Practical methods of implementing this are discussed, and a multi-channel optimization program is built up for use with an on-line digital computer. The performance of this program is evaluated for a four-parameter system simulated on an analog computer, with and

without compensation for the effect that previous changes in parameter have on the estimate of the gradient of the cost-function.

In the last part of the thesis, the basic limitations on speed of optimization are investigated. Various strategies for rapid hill-climbing are discussed, and a technique for minimizing the number of optimization steps is developed.

ACKNOWLEDGEMENTS

The work described in this thesis was carried out at the University of Warwick from October 1965 until September 1968, under a Science Research Council Research Studentship grant.

The author would like to thank Professor Douce and the staff of the department of Electrical Science at the University for their help and co-operation, especially Dr. K.C. Ng, whose encouragement and useful comment while supervising this work was always welcomed. Thanks are also due to Mr. John Monk for many helpful discussions, both in the laboratory and on the bowling green, and to Mrs. Margaret Cannell for her patient deciphering of the manuscript and her competent typing of this thesis.

LIST OF SYMBOLS

Certain symbols used only locally are not listed here. In some cases they duplicate the symbols below, but the local definitions make the meanings clear. The last column below indicates the sections in which the symbols are defined.

Symbol	Definition	Section
a	chain-code amplitude scale factor	2.2
a_s	amplitude of sine-wave	5.2
A_n	amplitude of Fourier components of m	2.4(a)
B	scaled-expression for variance	2.4(b)
$c(t)$	unit chain-code (amplitude = ± 1)	2.2
$c_j(t)$	j-th chain-code in multi-parameter system	7.2
C	co-variance function	4.5
C_a	cost of actions applied to a system	1.2(b)
C_t	cost index for a process at time t	1.2(b)
$\underline{d}$	vector of disturbances	1.2(a)
d_e	error in d.c. estimate	4.4(a)
D	standard deviation of gain estimate using chain-code and d.c. compensation	4.2
D_q	scaled version of D_s	2.4(a)
D_s	standard deviation of gain estimate using chain-code and no d.c. compensation	2.4(a)
D_z	standard deviation of gain estimate using sine-wave	5.3(a)
$E[\ ]$	expected value of bracketed terms	4.2
$f(K)$	performance function of single-path plant	6.2(a)
$f_j(K)$	performance function of j-th parallel path	6.2(a)
f_N	normalized frequency of sine-wave	6.3(b)
f_s	frequency of sine-wave	6.3(b)
f_{ν}	double-integral used in evaluating D^2	4.2

Symbol	Definition	Section
$f(\underline{k})$	function relating P_t to $\underline{k}$	1.2(b)
$f^1(\underline{k})$	function relating C_a to $\underline{k}$	1.2(b)
$f_t(\underline{k})$	function relating V_t to $\underline{k}$	1.2(b)
F_j	value of F(w) for j-th parallel path	6.2(a)
F(w)	ratio of gain estimate to true gain using sine-wave	5.2
g	true gain	4.4(a)
$\hat{g}$	estimate of gain using chain-code	4.4(a)
g_e	total error in estimate of gain	4.4(a)
g_{e1}	gain error due to periodicity	4.4(a)
g_{e2}	gain error due to d.c. bias estimation	4.4(a)
g_{e3}	gain error due to non-finite plant settling time	4.4(a)
$\hat{g}_s$	estimate of gain using sine-wave	5.2
$(g_i)_t$	slope of performance function in i-th direction at time t	1.2(b)
$h(\tau)$	impulse response of plant	2.2
$h(\tau,t)$	time-dependent impulse response of plant	2.2
$h_j(s)$	impulse response of j-th parallel path	6.2(a)
$\hat{h}_j(s)$	primary estimate of $h_j(s)$	7.3
$\hat{h}'_j(s)$	secondary estimate of $h_j(s)$	7.3
$h_r(\tau)$	impulse response of running summer	A2.2
$H_r(jw)$	power spectrum of running averager	A2.2
$[H]$	matrix of transfer functions	1.2(a)
$\mathcal{H}[\ ]$	Heaviside function (=0 if $[\ ] \leq 0$, = $[\ ]$ otherwise)	4.3(a)
$\underline{i}$	vector of intermediate variables of process	1.2(a)
k	number of bits averaged for d.c. estimation	3.2
$\underline{k}$	vector of plant parameters	1.2(a)
$\underline{k}_i$	i-th parameter of plant	1.2(b)
K	plant parameter	1.2(c)

Symbol	Definition	Section
K_e	error in estimate of K_{opt}	6.3(b)
K_n	value of K at n-th clock time	10.2
K_o	present value of K	1.2(c)
K_{opt}	optimum value of K	6.3(b)
$\hat{K}_{opt}$	estimated value of K_{opt}	6.3(b)
m_s	number of test steps/operative step	1.4
$m(t)$	output of multiplier	2.3(a)
$m_\nu(t)$	noise component of output of multiplier	2.4(a)
M	number of bits in a chain-code ($\neq$ N)	7.4
n_c	number of parameters of plant	1.2(b)
n_q	ordinate at start of q-th impulse response	8.3
n_s	number of shift-register stages	A1.1
n_o	ordinate of impulse response reduced to zero	4.4(b)
n_1	first zero-crossing of impulse response	4.4(b)
N_c	number of channels of system	7.4
p	number of bits corresponding to T_r	2.3(a)
p_i	output of i-th performance function	6.2(a)
p_q	ordinate of step response of q-th channel	8.3
$P(t), P_t$	performance index at time t	1.2
$\hat{P}(t)$	estimate of P(t)	1.2(c)
$q(t)$	output of basic identification system	2.3(a)
$q_{av}(t)$	average value of $q_\nu(t)$	2.4(a)
$q_\nu(t)$	noise component of q(t)	2.4(a)
$r(t)$	running average of chain-code	2.3(a)
R	ratio of power of noise to power of chain-code	2.4(b)
R_s	ratio of power of noise to power of sine-wave	5.3(b)
$S_e(w)$	slope of performance curve at K_{opt}	6.3(b)
S_N	normalized slope-error	6.3(b)
$S(t)$	true step response	2.2

Symbol	Definition	Section
$\hat{S}(t)$	estimate of step response using chain-code	2.2
$\hat{S}_c(t)$	correction term due to periodicity	2.3(b)
$\hat{S}_d(t)$	step response estimate without d.c. correction	2.3(b)
$S_e(t)$	true step response with d.c. error	2.3(d)
$S_j(p)$	step response estimate due to j-th path	6.2(b)
S_N	normalized slope error	6.3(b)
$\hat{S}^1_q(p_q)$	step response estimate of q-th channel	8.3
$\hat{S}^1_q(p_q, j)$	step response estimate of q-th channel at j-th clock interval	8.3
$\hat{S}_\nu(t)$	noise component of gain estimate	4.2
$\hat{S}_1(t)$	step response estimate due to one period	2.3(b)
T	chain-code period	1.5
T_i	integrator time constant	2.3(a)
T_j	time constant of j-th path	6.3(b)
T^1_j	settling time of j-th channel	7.4
T_r	running-averager period	2.3(a)
T_s	system time constant	2.3(c)
T_ν	noise time constant	2.4(b)
$u(t)$	scaled version of chain-code ($= ac(t)$)	2.2
$u_{-2}(\tau)$	unit ramp function	A1.1
$U(\tau)$	triangular function composed of ramp functions	A1.1
V_t	value of products	1.2(b)
$V(\alpha)$	scaled expression for variance	5.3(b)
$\underline{x}$	vector of plant inputs	1.2
$x_1(t)$	perturbation applied to plant parameter	1.2(c)
$x_2(t)$	signal to be correlated with plant output	5.2
$\underline{y}$	vector of plant outputs	1.2
$y(t)$	plant output	2.2
$\hat{y}(t)$	estimated plant output	A3.1

Symbol	Definition	Section
$y_j(t)$	output of j-th path	6.2(a)
y_n	value of y at n-th clock time	10.2
z_j	gradient estimate due to j-th path	6.2(a)
$Z(T)$	output of sample-and-hold using sine-wave	5.2
α	ratio of T_γ to T	2.4(b)
β	ratio of T_r to T	2.4(b)
γ	constant in variance formula	4.3(b)
$\delta(\tau)$	delta function (area = 1 if $\tau = 0$, zero otherwise)	A1.1
Δ	averaging period for d.c. estimation	3.2
ΔK	step-change in parameter	1.2(c)
η	scale factor for step response	3.3(a)
η_B	bias-error ratio	2.3(d)
η_C	signal-to-noise ratio for system of Ch.2	2.4(c)
η_D	signal-to-noise ratio for system of Ch.4	4.4(c)
η_I	impulse-shape error	2.3(d)
η_p	periodicity-error ratio	2.3(d)
η_s	signal-to-noise ratio for sine-wave system	5.3(a)
η_{sN}	ratio of signal-to-noise ratios	5.4
η_T	integration-time error	2.3(d)
η_v	ratio of variances	5.4
λ	clock-interval of chain-code	1.5
μ	scale factor for step response	3.3(a)
μ_c	scale factor for covariance	4.5
μ_f	feedback scale factor	10.2
μ_i	scale factor	7.3
μ_q	scale factor for system output using chain-code	2.3(b)
μ_r	gain of the running averager	2.3(a)
μ_s	scale factor for system output using sine-wave	5.2

Symbol	Definition	Section
$\nu(t)$	noise at output of plant	2.4(a)
ρ_w	weighting function	4.5
$\rho_{Tr,\Delta}(t)$	compound running sum	3.3(a)
$\sigma_c(q:r)$	running sum of c from q to r	3.3(a)
σ_ν^2	(mean-squared) power of noise	2.4(b)
$\hat{\phi}$	estimate of impulse response	4.4(a)
ϕ_n	phase of Fourier components of $m_y(t)$	2.4(a)
$\phi_{cc}(\tau)$	auto-correlation function of chain-code	1.5
$\phi_{cy}(\tau)$	cross-correlation function of c and y	2.2
Φ	power/unit bandwidth of noise	4.3(b)
$\Phi_c(w)$,etc.	power spectrum of chain-code, etc.	A2.2
$*$	convolution operator	1.2(c)

CONTENTS

page number

Page number

Appendices

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 THE GENERAL PROBLEM

Many industrial plants, such as chemical works,[1] steam-generating boilers[2,3] and paper-making mills, require constant supervision of the output materials to ensure that the plant is working within allowable tolerances, and, if not, control action must be applied to correct any deficiencies. The ideal control system continuously examines the whole plant, assesses the performance in terms of overall cost and makes any adjustments necessary to bring the system to the optimum state at which the financial return on investment is greatest.

A straightforward feedback system[4] compares the output of the plant with some desired state, and adjusts the plant-parameters to reduce the error to a minimum. For such a system the desired state must be decided before-hand. In an industrial process, this state will not coincide with the optimum state in general, due to fluctuations in the properties of the raw-materials, ageing of components, leakages, changes in the ambient conditions, and so on. It is therefore desirable to design a supervisory system that evaluates the optimum state for the conditions prevailing at any time and drives the parameters towards this state. By continually repeating this process, the optimum state can be reached, and maintained, despite changes in the operating conditions due to uncontrolled influences. This process is known as self-adaptive control, and is the dominant subject of this thesis.

In order to optimize a system it is necessary to assess the characteristics of the plant, by a process known as identification. In the context of the work presented in this thesis, identification involves the evaluation of the changes in performance due to changes in the parameters. This can be achieved by perturbing the parameters in a predetermined manner,

and investigating the effect these perturbations have on the overall performance of the plant. In this way, the slope of the curve of performance against parameter value can be evaluated for each parameter. This then enables the mean value of each parameter to be changed by an amount dependent on the appropriate estimate of slope. This is repeated until a state is reached where no further improvement in performance can be obtained.

A great deal of theory of self-adaptive control systems has appeared in the literature in recent years. It is the purpose of this thesis to examine several aspects of adaptive control, and to develop criteria for the design and evaluation of a class of adaptive systems, using periodic-perturbation techniques. Considerable attention has been paid to practical details, to enable potential users to develop their own systems for use with or without on-line computing facilities.

In this chapter several basic topics will be introduced:

(i) The concepts of performance and cost functions, with reference to practical processes.

(ii) The distinction between optimal and self-adaptive control.

(iii) The classification of adaptive systems.

(iv) The properties of pseudo-random binary sequences.

(v) The experimental facilities available at the University of Warwick.

1.2. THE CONCEPTS OF PERFORMANCE AND COST

In this section, the concepts of performance and cost will be examined with reference to general and particular processes.

1.2(a) The Parameter-controlling Loop

Consider first the general process shown schematically in figure 1.1. This is a continuous-flow process with a set of ingoing materials, such as liquids or gases, a set of outgoing materials, or products, which may also

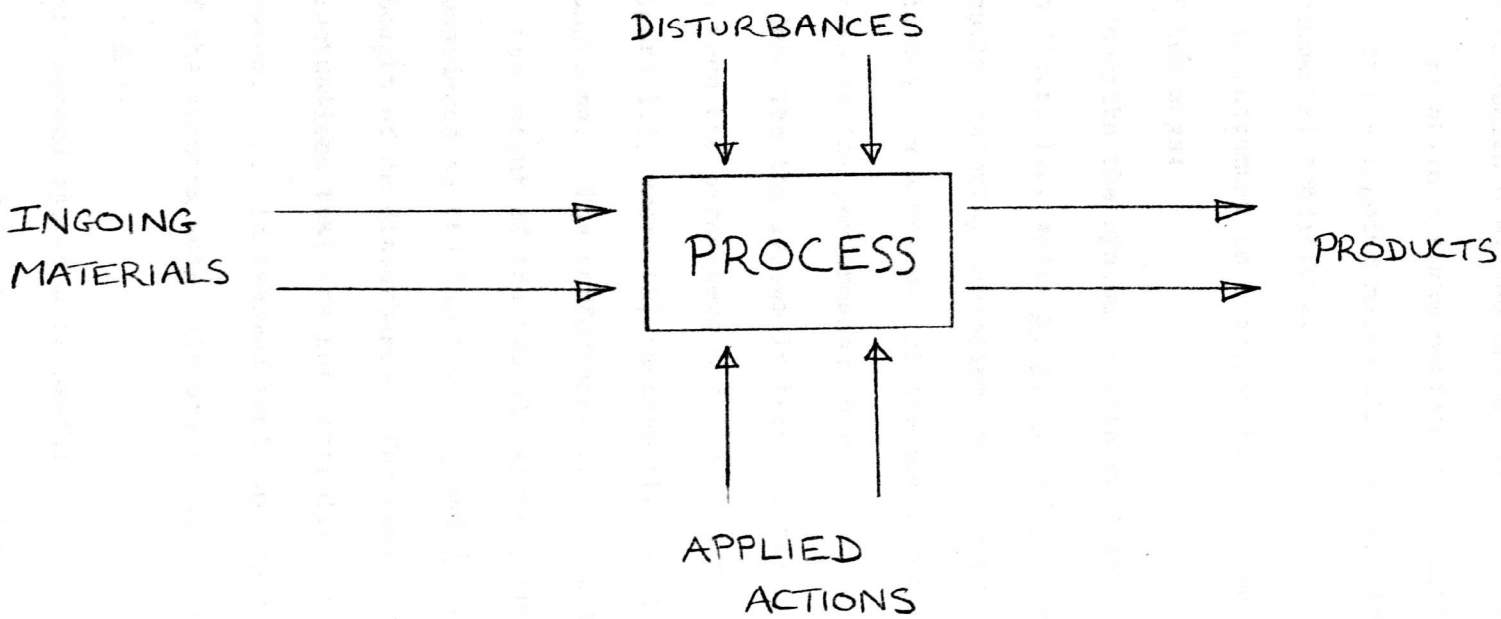


FIGURE 1.1 : A CONTINUOUS FLOW PROCESS

be liquids or gases, and a set of applied actions, such as heat. The control engineer is interested in maintaining the best properties of the products for the least overall cost. The optimization problem is therefore to design a system which controls some or all of the applied actions, to maintain optimum performance despite fluctuations in the properties of the ingoing materials and uncontrollable fluctuations of the environmental conditions.

It is customary to consider the problem of parameter optimization in one of two ways:

(i) To describe the system in straightforward terms as shown in figure 1.2, where $\underline{x}$, $\underline{y}$, $\underline{k}$, and $\underline{d}$ are vectors of the inputs, outputs, parameters and disturbances respectively, and $P(t)$ is a measure of the performance of the system known as the performance index.

(ii) To describe the system in terms of the transfer functions between the parameters and the outputs, as shown in figure 1.3, where [H] denotes the matrix of such transfer functions. The performance index can then be considered as the output of the overall system, and the parameters are considered as the inputs. $\underline{x}$ and $\underline{d}$ can then both be thought of as disturbances, for they are both subject to fluctuations that are not controlled by the optimizing system. (It is assumed that any controlled properties of the ingoing materials are included in the parameter set $\underline{k}$).

This second approach is useful for the analysis of the parameter-controlling loop, and is the representation most frequently used in this thesis.

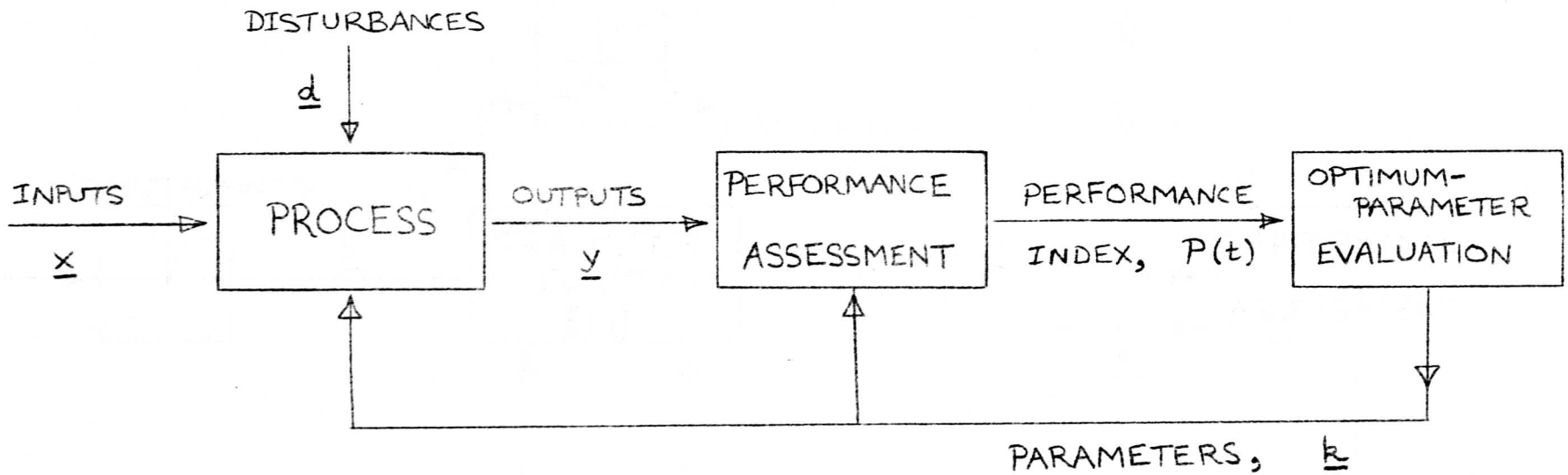


FIGURE 1.2 : THE OPTIMIZATION PROCESS

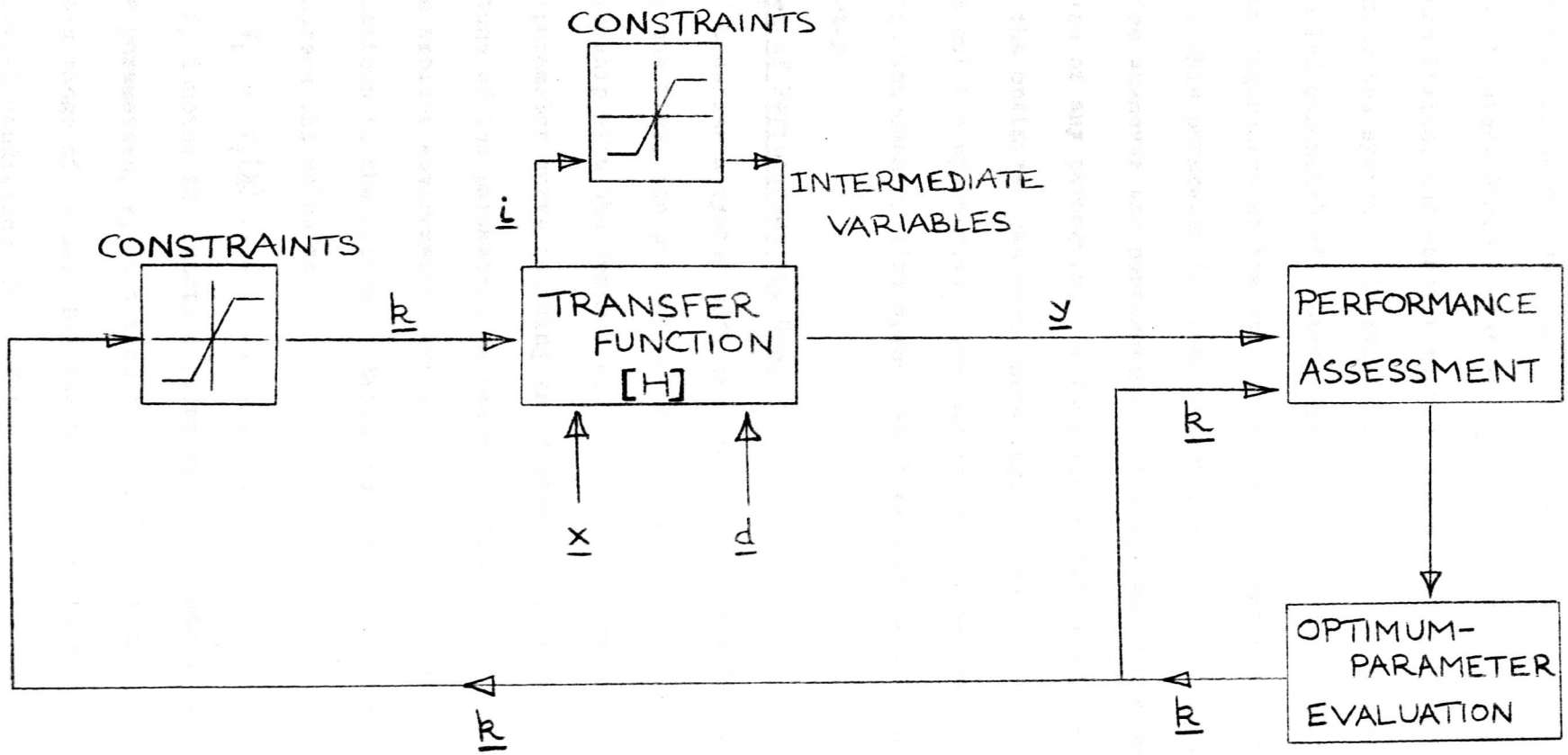


FIGURE 1.3 : AN ALTERNATIVE REPRESENTATION OF OPTIMIZATION PROCESS

In general there will be limits on the ranges over which the parameters can be varied, and there may also be limits on the permissible ranges of intermediate variables, $\underline{i}$, such as maximum temperature or pressure limits, and account must be taken of these constraints when designing the system. In particular, it must be checked that no controlled parameter will obviously take on its maximum value at the optimum regardless of the values of other parameters, for then a loop to control this parameter is clearly worthless. Such a situation arises in practice whenever the performance of the system is a steadily increasing function of any parameter. A function of this kind may not be discovered until the optimizer has been operating; it is then clearly advisable to switch out the appropriate loop and fix the parameter valve at its maximum, to avoid unnecessary time spent searching for the optimum value of this parameter.

1.2(b) Indices of Performance and Cost

Consider a system with one controlled parameter and one product. At any one time, the properties of the product will have some definite relationship with the controlled parameter, and we can therefore define a value-parameter curve relating the financial value of the product to the magnitude of the parameter, as shown in figure 1.4. This curve will depend on the ambient environment, and will therefore be influenced by any fluctuations in the system. Extending the concept to n parameters, we can express the value as

$$V_t = f_t(\underline{k}) \qquad \text{.....(1.1)}$$

where f_t denotes the function relating the financial value of the product to the parameters, $\underline{k}$, at a time t. This function will be bounded within a usable range of values, but due to constraints on the parameters, imposed by physical conditions, part of the usable range may be unavailable in a practical system.

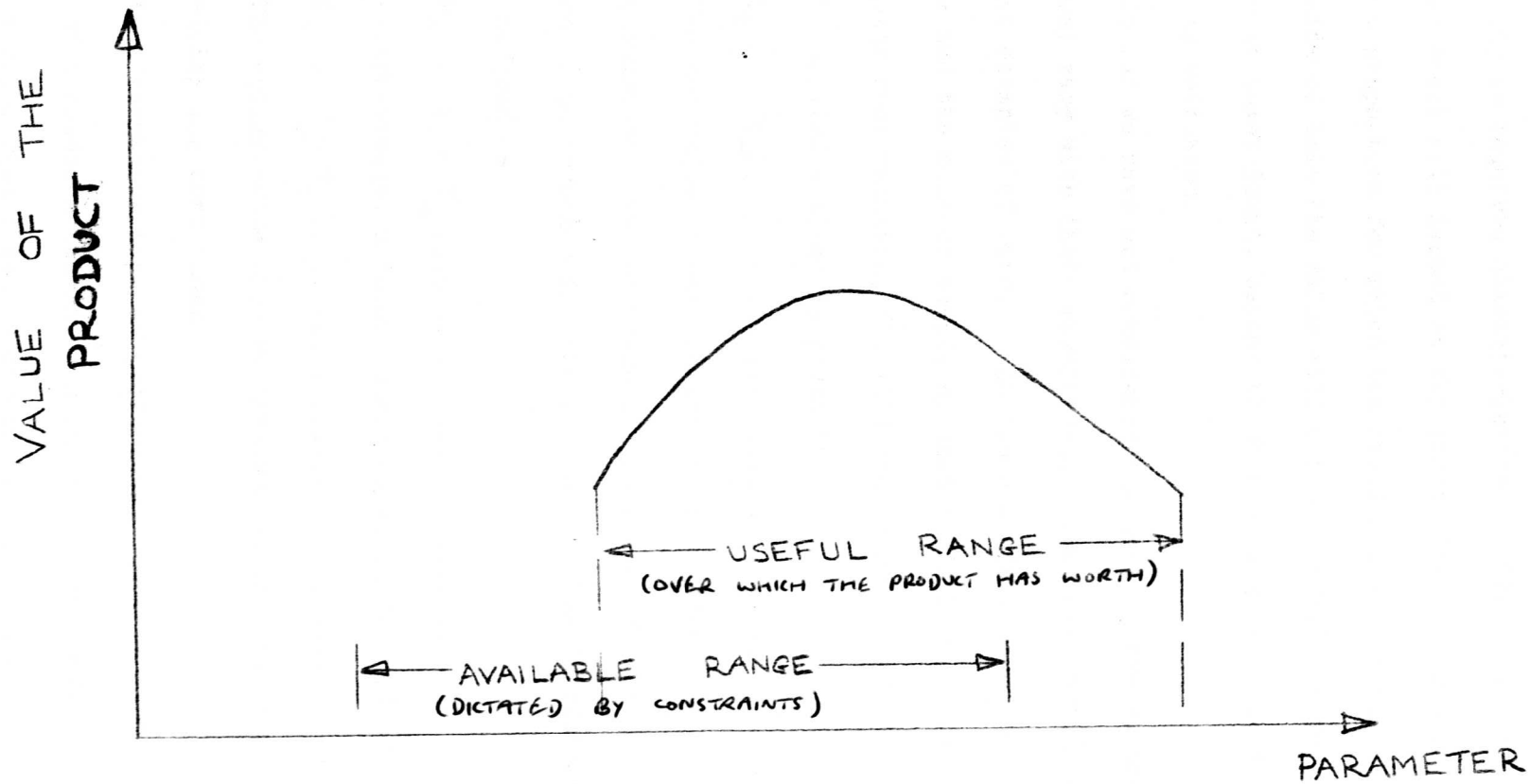


FIGURE 1.4: A TWO-DIMENSIONAL VALUE CURVE

An illustration of a value curve is provided by the production of stainless steel, in which ordinary steel is impregnated with chromium to give it the required characteristics. The financial value of the stainless steel will depend on the proportion of chromium used. There will be a proportion for which the steel has its maximum value, and on either side of this the value will decrease until the proportion reaches an upper or lower limit, beyond which the steel will be worthless and will have to be retreated.

So far we have not considered the cost of the parameters, which will in general vary with their magnitudes; power consumption and raw material costs are examples of this. In the stainless steel process the cost of the chromium and the cost of supplying heat would have to be taken into account. A parameter cost function, f^1, will therefore be defined, such that the cost, C_a, of the applied actions is given by

$$C_a = f^1(\underline{k}) \qquad (1.2)$$

The net value of the process is then given by subtracting the cost of the parameters from the value of the products. This can be used as a criterion of performance for the process. The 'performance index' can then be defined as

$$P_t = V_t - C_a \qquad (1.3)$$

Alternatively, a 'cost index' could be defined as

$$C_t = C_a - V_t \qquad (1.4)$$

The optimization problem involves maximizing the performance index or minimizing the cost index.

An alternative type of performance index is used to describe the ability of a feedback control system to follow changes in desired output. In such a case, rise time, overshoot or mean squared error can be used to describe the effectiveness of the system. The problem is to devise a supervisory system to maximize the performance index despite fluctuations in component values and environmental conditions.

The topics discussed in this thesis are applicable to either of the types of performance index mentioned above. In either case, the performance index can be written in terms of the parameters as

$$P_t = f(\underline{k}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1.5)$$

at any one time t. The slope of the performance surface in the direction of the i-th parameter, $\underline{k}_i$, is then

$$(g_i)_t = \left(\frac{\partial P}{\partial \underline{k}_i}\right)_t \frac{\underline{k}_i}{|\underline{k}_i|} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1.6)$$

The optimum value of P is then given by the point where $g_i=0$ for all parameters, and the matrix of partial derivatives is negative definite. If several such points exist, the absolute maximum of these will give the optimum value. (The corresponding equations are denoted (1-7).)

1.2(c) Dynamic Representation of the Parameter Path

It will now be shown how the overall transfer function of the system from parameter to estimated performance index is related to the gradient of the performance hill.

Consider a process-control plant with a single controlled parameter. In any practical system the dynamics of the path from parameter to estimated performance index will be distributed in some way throughout the plant. In many cases there is no means of assessing this distribution, and in such a case it is difficult to obtain a realistic model of the plant. Monk[5] has shown how the estimate of the slope of the hill using a parameter perturbation technique depends on the relative positions of the hill and the dynamics. Some idea of the performance function of a plant can however be gleaned from the somewhat over-simplified model, adopted in much of the literature[6,7] of a performance function followed by lumped dynamics, as represented in Figure 1.5, where P(t) is the true performance index at time t, $\hat{P}(t)$ is the estimate of performance index under dynamic conditions, and $h(\tau)$ is the

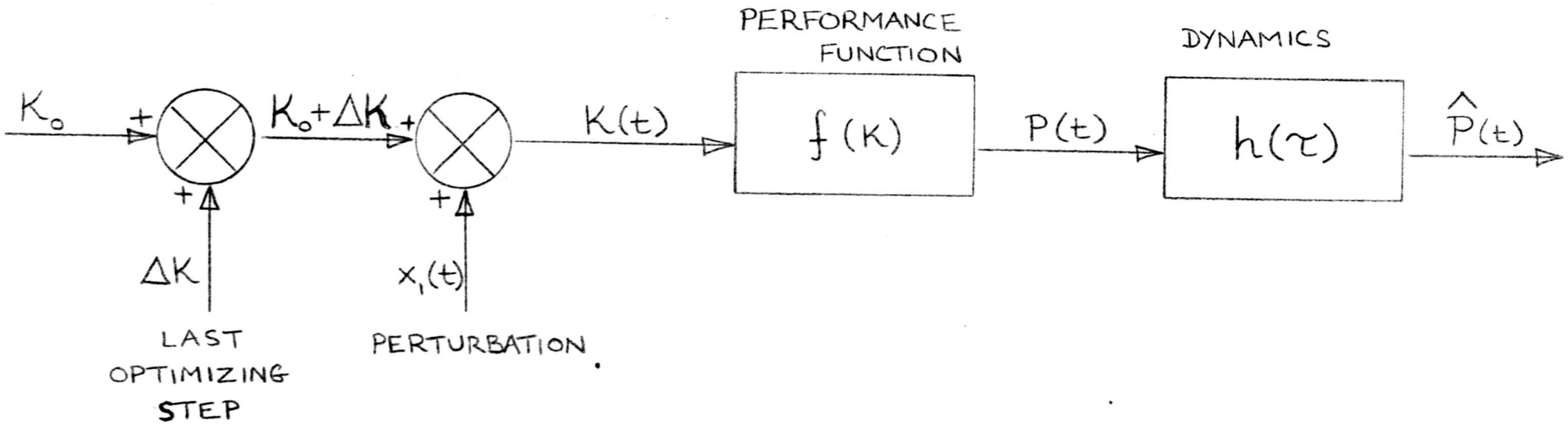


FIGURE 1.5: A PERFORMANCE-FUNCTION MODEL

response of the system to a unit impulse. Errors arising from using this model may severely hamper attempts to optimize a practical system[5,8]; the extent of the errors can only be found experimentally, as the result of a pilot scheme applied to the actual system.

From the model of Figure 1.5, the output of the performance function generator is given by

$$P(t) = f\left[K_o + \Delta K + x_1(t)\right] \qquad \text{.....(1.8)}$$

where K_o is the previous value of the parameter K, ΔK is the change in parameter as a result of the latest optimization step, and $x_1(t)$ is a perturbation applied to the parameter.

If we assume the performance function can be considered linear over a small range about $K = K_o$, then P(t) can be represented by a truncated Taylor series[9] as

$$P(t) \simeq f(K_o) + \left(\frac{\partial f}{\partial K}\right)_{K = K_o} \cdot \left[\Delta K + x_1(t)\right] \qquad \text{.....(1.9)}$$

If the effect of the previous step change in parameter is small, and the effect of any d.c. bias on the parameter can be eliminated in the identification process, P(t) can be approximated as

$$P(t) \simeq \left(\frac{\partial f}{\partial K}\right)_{K = K_o} x_1(t) \qquad \text{.....(1.10)}$$

Hence the overall relationship from parameter perturbation to estimated performance index is

$$\hat{P}(t) = P(t) * h(\tau) \simeq x_1(t) * \left[\left(\frac{\partial f}{\partial K}\right)_{K = K_o} h(\tau)\right] \qquad \text{.....(1.11)}$$

where '*' denotes convolution.

For a multi-parameter system, a similar relationship holds for any one parameter, providing the other parameters are held constant. The problem when all the parameters are perturbed simultaneously is examined in Chapter 7.

From equation (1.11) we can see that the slope of the performance function is equivalent to the gain of the path from the parameter to the estimated performance index. Therefore the $(g_i)_t$ terms of equation (1.6) correspond exactly to the gains of the parameter paths. It is the purpose of this thesis to examine various ways of estimating the gains, and to use these estimates to bring the system towards the optimum.

It is not intended to consider capital and maintenance costs in this work, beyond noting that these are influenced by such considerations as the maximum values of intermediate variables, such as temperature, pressure and flow rate. At the design stage these factors must be considered, and the limits on the parameters must be chosen, bearing in mind the likely performance characteristics of the plant, assessed from pilot schemes or by experience.

1.3. Optimal and Self-Adaptive Control

In the past few years, control engineers have developed theories along two distinct tracks, termed optimal control and self-adaptive control. Both these techniques are referred to as optimization, which can lead to great confusion. We will now consider what these terms have come to mean, and what the basic differences are. Figure 1.6 shows the basic classification.

Optimal control is the computation of the control system configuration and parameter values that will result in a given plant operating in a region as close to the optimum state as possible, under given environmental conditions. In terms of the performance function, this implies an operating point where the conditions of equations (1.7) are approximately satisfied at all times. Optimal control is essentially an off-line design problem. It requires pre-determined knowledge of the transfer functions of the process from parameters to performance index, and a knowledge of the statistics of likely disturbances and fluctuations.

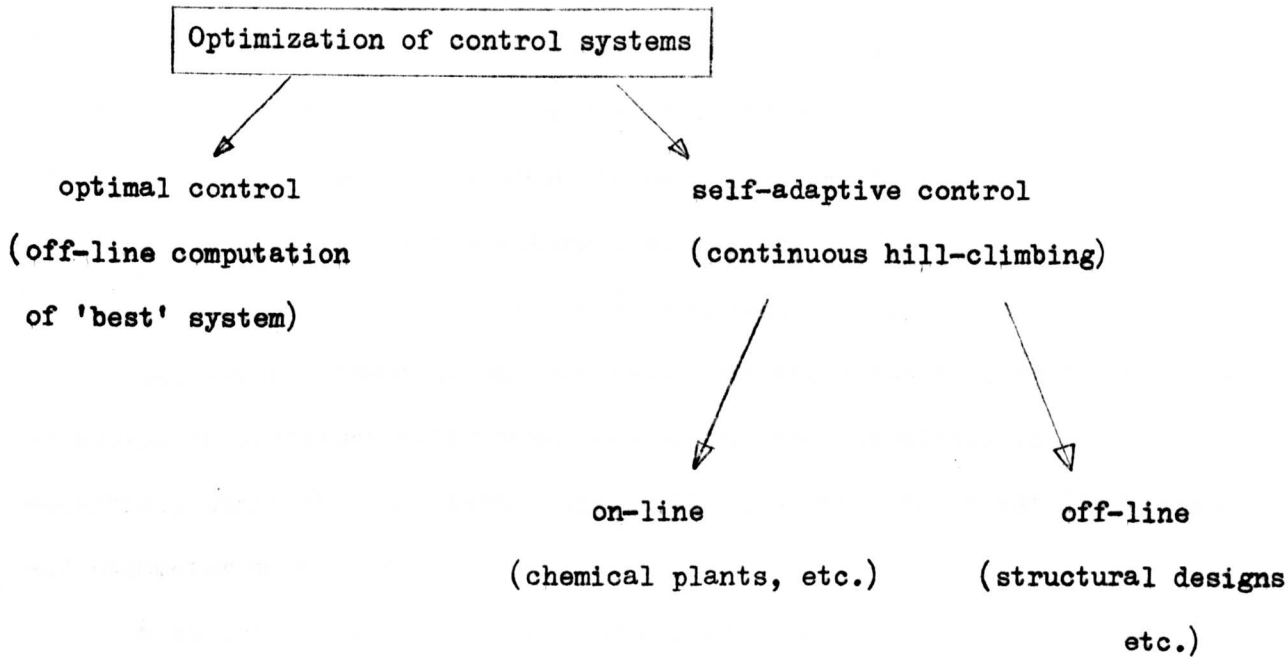


Figure 1.6. Classification of Optimization Techniques

Self-adaptive control, on the other hand, is a technique where the slope of the performance 'hill' is continuously assessed and steps are taken to drive the parameters towards their optimum values, that is, to the state where the slope of the hill is zero in all parameter directions. Self-adaptive systems are also referred to as hill-climbing systems, self-adjusting systems, extremum-seeking systems, or, in certain cases, parameter-tracking systems. For such a system, any prior knowledge of the performance function and the system dynamics can be used to advantage to design an efficient optimizing system, but such knowledge is not essential, apart from an overall appreciation of the likely settling times and parameter magnitudes.

A self-adaptive technique should always be used on-line to the process to be optimized, whenever it is economically and physically feasible. An example of on-line control is the continuous optimization of a steam-generating boiler[2,3], where the problem is to maintain the maximum efficiency at all times, despite natural changes in steam flow rate, fuel quality, leakage of air and other uncontrolled factors. Off-line self-adaptive analysis can be applied to such problems as the design of portal-frame structures to give maximum load-bearing capacity for minimum cost. By calculating the changes in permissible load due to changes in the dimensions of the frame members, the optimum state can be found. This type of problem does not lend itself to on-line analysis, for physical reasons. In such circumstances, the mathematical equations of the structure can be simulated on a digital computer, and a variational hill-climbing technique employed to find the optimum. This type of analysis has two basic differences compared with the on-line self-adaptive procedure, namely:

(i) There are no disturbances or uncontrolled fluctuations to be considered, and

(ii) there are no dynamic factors.

For these reasons optimization by performance index gradient estimation can be performed using step changes in the parameters, rather than sophisticated statistical techniques, and the gradient can be assessed without having to wait for the system to settle down.

The on-line control problem is the aspect of self-adaptive control to be mainly considered in this thesis.

1.4 Types of Self-Adaptive Control Systems

There are many types of self-adaptive systems[10,11], and these can be classified in several different ways. One classification scheme is outlined in figure 1.7. To avoid confusion, the relationship between parameters and performance index will be referred to as the 'plant', the external controller being the 'system'.

Self-adaptive systems can be divided into two main types:

(i) Model reference systems.[1,12,13,14,15] These involve a two-fold process:

(a) Model building, in which a model of the plant is constructed, either from variable analogue components or as data in a digital computer. The complexity of the model is decided from an overall consideration of the plant, and the values for the parameters are chosen either off-line or by an on-line self-adaptive technique to give a performance as close to that of the plant as possible.

(b) Plant control, in which the optimum values for the parameters are calculated from the model, so that any hill-climbing is undertaken on the model and not directly on the plant. The plant parameters can then be adjusted accordingly. The set-up is shown schematically in figure 1.8.

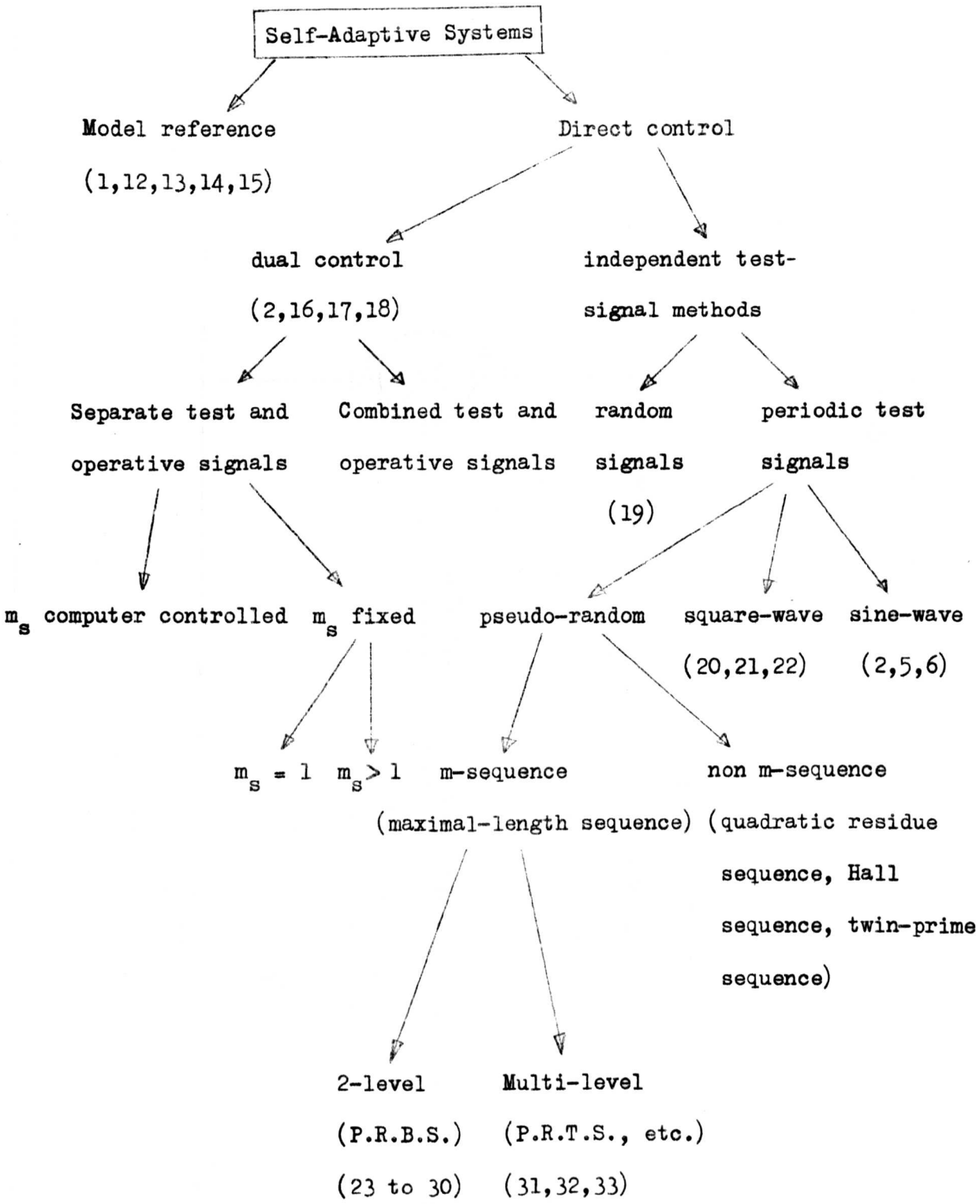


Figure 1.7. Some Types of Self-Adaptive System (with References in Brackets)

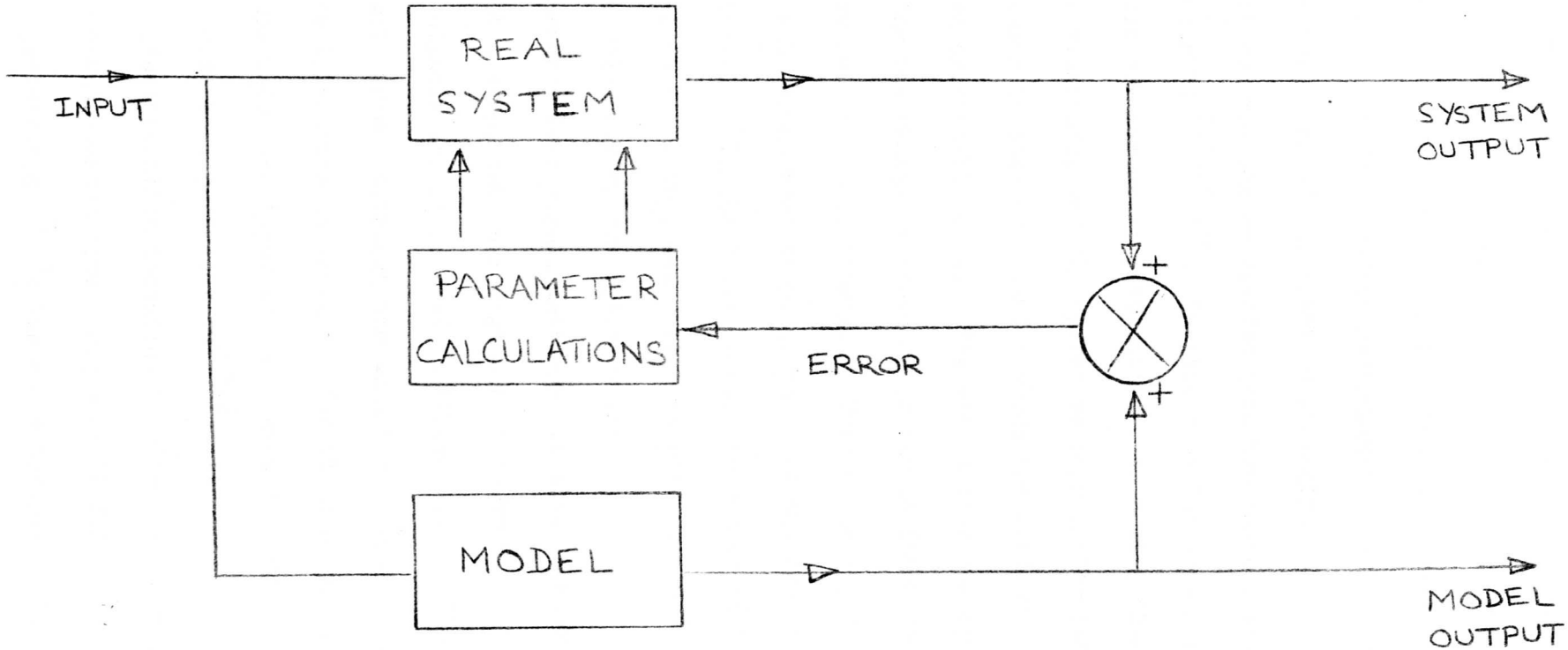


FIGURE 1.8 : MODEL-REFERENCE SYSTEM

(ii) Direct Control Systems, in which no model of the plant is explicitly used. Instead, the hill-climbing process is applied directly to the plant, by perturbing the parameters, deciding whether this improves the performance or not, and changing the operating point of the plant accordingly.

Direct control can be divided into two categories:

(i) Dual-control[2,16,17,18]: in this case the test signal (the exploratory parameter perturbation) is a step change, usually of fixed magnitude, and the operative signal (the parameter change found necessary to take the plant towards the optimum) is also a step, of fixed or variable size. The method involves applying the test signal, noting the change it produces in the performance, and choosing an operative step accordingly. Each operative step can be used as the subsequent test step, or the test and operative steps can be distinct. In the latter case, the number of test steps to each operative step, m_s, can either be predetermined or be computer-controlled during the optimization.

(ii) Independent test signal methods: in these systems, the test signal is not a step, but is a signal with known statistical properties. Such signals are classified in figure 1.7. These techniques are all well-known, although the amount of comparative study work appearing in the literature is small. For the most part, this thesis considers methods using two types of test perturbation:

(a) sine-wave perturbation[2,5,6]: the noise analysis of a sine-wave identification technique is given in Chapter 5.

(b) pseudo-random binary sequence (P.R.B.S., or 'chain-code') perturbation[23,24]: the noise analysis of two P.R.B.S. identification techniques is given in Chapters 2, 3 and 4, and the design and performance of multi-parameter optimization systems is examined in Chapters 7, 8 and 9.

The optimization of plants having more than one dynamic path between parameter and performance index is examined in Chapter 6 for both sine-wave and chain-code perturbation systems. The performance of systems using other control techniques is discussed in Chapter 10, where their relative merits are examined, and various strategies for rapid hill-climbing are considered.

1.5 Chain-Codes

A brief description of the most significant properties of chain-codes will now be given. A binary chain-code is a periodic signal which possesses at any one time one of two states, and can switch from one state to the other only at multiples of a basic interval, λ, known as the clock-interval, clock-time, or bit-interval. A chain-code is characterized by the nature of its auto-correlation function, defined by

$$\phi_{cc}(\tau) = \int_0^T c(t)\, c(t-\tau)\, dt \qquad \text{.....(1.13)}$$

where $c(t)$ is the chain-code, T is its period, and τ is the shift of one code relative to the other. The 'pseudo-random' property of the chain-code implies that $\phi_{cc}(\tau)$ approximates to the auto-correlation function of white noise, which is an impulse at zero shift. In fact $\phi_{cc}(\tau)$ consists of a train of approximate impulses at intervals of T.

It is well-known that white noise can be applied to the input of a plant as a test signal without it being correlated with other signals occurring in the plant, and that the result of correlating the output of the plant with the white noise gives the impulse response of the plant. Using a chain-code as the test signal instead of the white noise yields almost the same result, but has the advantage that the averaging time needed for the correlation is finite, namely the chain-code period T.

Further properties of chain-codes are given in Appendix A1.1.

1.6 The Experimental Facilities Used

The experimental work for this thesis was aided by the use of three computing facilities:

(i) a G.E.C. 92 digital process-control computer

(ii) a Solartron 247 analogue computer

(iii) an Elliott 4130 digital data processing computer.

The G.E.C. 92 computer enables on-line computing to be carried out on practical systems, or on models thereof. It has a memory of 8,000 12-bit words; the most significant bit of each word is used to denote sign, negative numbers being stored in two's complement form. The word-length is adequate for most control problems, but double-length working can be used if greater accuracy is required. The memory cycle time is 1.75μ sec. Digital-to-analogue and analogue-to-digital converters are provided, which can be switched to appropriate analogue lines by the programmer. These lines come up on an interface connection panel, where they can either be routed to other parts of the building, such as the engine test calls or the fatigue laboratory, or linked directly to the analogue computer. Such lines can also be used for monitoring purposes, using an oscilloscope or graph-plotter.

The Solartron 247 Analogue Computer is used for simulation of practical systems. It has facilities for multiplication and non-linear function generating in addition to normal summing and integrating elements. Facilities for iterative computation are provided, whereby successive switching between the 'problem check', 'compute ' and 'hold' modes can be performed continuously, under the control of the internal timer or of signals sent externally from the digital computer.

The Elliott 4130 computer is a high-speed data processing computer for off-line calculations. The memory cycle-time has recently been upgraded to 2μsec., and the main-store capacity to 64,000 words of 24-bits. There is also a magnetic tape backing store facility and a graph-plotter.

All these computing facilities have come into being during the course of the work described in this thesis, some of which was completed before the computers became available. Some of the methods described in the early part of the thesis can now be performed in the department in a much simpler manner using the computers, but the methods are still applicable in an environment where computers are not available or are considered uneconomic for solving such problems. Considerable time could however be saved, by replacing the transistorized circuits, such as chain-code generators and running summers, by their integrated circuit equivalents. Commercial ready-made equipment has come onto the market recently which can perform some of the tasks of code-generation and cross-correlation, but this equipment is very expensive at present.

1.7 Aims and Contributions of the Thesis

When the work described in this thesis was started, the author was invited to study two aspects of self-adaptive control using chain-codes:

(i) The performance in the presence of noise of a particular single-parameter system postulated in the literature.

(ii) The development of a multi-parameter optimizer which, it was thought, might overcome the problems of interactions between parameter paths, by using on-line estimates of the interactions as corrections for the gain estimates.

During the investigations into (i), it was found that the system

postulated could yield serious errors due to d.c. bias, even with no noise present. As a result, an improved system was developed and analysed. Following on from this, an optimizer using sine-waves was analysed under noisy conditions and compared with the chain- code system. The application of each system to plants with single and multiple dynamic paths between parameter and output was then examined.

Investigation into (ii) above showed that the suggested compensation techniques will not work, and any similar methods would also fail. Effort was then turned to the choice of multi-parameter optimizer to give the best performance, in terms of short identification time, small interaction errors and ease of implementation.

The contributions of this thesis can be summarized as:

(i) An exact mathematical analysis, supported by experimental evidence, of the basic chain-code system, with and without noise, with an assessment of the errors, and a design procedure for choosing the parameters of the identification system.

(ii) The derivation of an improved system that overcomes the major error of the original one.

(iii) Discussion of methods of implementing the improved system, with and without recourse to a process-control computer.

(iv) Detailed analysis of the improved system, with and without noise, yielding design criteria for the system, and showing that the inherent errors in the system are not large enough to prevent the system from being used as part of an optimizer.

(v) Analysis of the sine-wave system in the presence of noise, and quantitative comparison with the improved chain-code system, showing the benefits to be gained from each type of system.

(vi) Mathematical analysis, supported by experimental evidence, to

show that errors arise in estimating the optimum of a plant with parallel dynamic paths between parameter and output when using sine-wave or chain-code perturbations.

(vii) Analysis of various compensation techniques for multi-parameter identification, and a physical explanation of why all such methods will fail.

(viii) A critical comparison of the methods described in the literature for multi-parameter optimization.

(ix) Development of a method using shifted versions of one code as parameter perturbations, for use with and without a process-control computer.

(x) Realization of an on-line computer program for optimization using shifted codes and a proportional-to-gradient optimization strategy, with application to four-parameter plant models.

(xi) An analysis of the errors arising during optimization due to the changing d.c. level of the plant output and to the changing magnitude of the envelope of plant output values.

(xii) A survey of several of the available identification and optimization techniques.

Although this thesis has been primarily concerned with chain-code methods of optimization, it is not suggested that these are always better than other techniques. In the last chapter it is shown that some of the advantages of using chain-codes claimed in the literature, such as good noise rejection and short identification time, do not always survive after close examination. A quantitative comparison of all the available techniques is beyond the scope of this work.

PART I

SINGLE-PARAMETER IDENTIFICATION

PART I : SINGLE-PARAMETER IDENTIFICATION

CHAPTER 2

SINGLE-PARAMETER IDENTIFICATION USING CHAIN-CODES, WITH AND WITHOUT NOISE PRESENT

2.1 Introduction

In this chapter the performance of an identification system using chain-codes will be analysed[12,23 to 28]. The system is based on the well-known result (an extension of the Wiener-Hopf relation[14]) that states that if a chain-code is applied to the input of a plant and cross-correlated with the output, the resulting function, when scaled suitably, is approximately the same as the impulse response of the plant. The technique may be used to evaluate the gradient of the performance function in an optimization loop[47]. In such a case, the parameter to be controlled can be considered as the 'input' to the 'plant', and the output of the performance function generator as the 'output' of the 'plant'. The gradient of the performance function is then given by the gain from 'input' to 'output' which can be deduced by integrating the impulse response to give the step response, and choosing a value of this which is representative of its value after an infinite time.

The basic principle is described in Section 2.2. An alternative method, using a running summer (or averager), multiplier and integrator, is described and analysed under noise-free conditions in Section 2.3. The practical implementation of the method is discussed, and experimental results are given to support the theory. The behaviour of the system under noisy conditions is examined in Section 2.4. from theoretical and experimental observations. The conclusions are presented in Section 2.5.

2.2. The Basic Principle

It has been seen in Chapter 1 how a two-level chain-code is characterised by having an auto-correlation function that approximates to an impulse. If the code is applied to the input of a control system and cross-correlated with the output,[14,37,38,39] as shown in Figure 2.1, the result is

$$\phi_{cy}(\tau) = \frac{1}{T} \int_0^T c(t-\tau)\, y(t)\, dt \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.1)$$

where c(t) is a 'unit chain-code', that is, one having levels of $\pm$ 1.

The convolution integral gives

$$y(t) = \int_0^\infty u(t-s)\, g\, h(s)\, ds \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.2)$$

where u(t) = ac(t), 'a' being a scaling factor dictated by practical considerations of the particular system to be identified. ('s' is dummy variable)

(Implicit in equation (2.2) is the assumption that $h(\tau)$ is constant with respect to time. For the present it will be assumed that the characteristics of the plant are static. If this is not so, $gh(\tau)$ must be replaced by $g(t)h(\tau,t)$, making equation (2.2) difficult to manipulate.)

Substituting for y(t) in equation (2.1), and interchanging the order of integration, gives

$$\phi_{cy}(\tau) = ag \int_0^\infty \phi_{cc}(\tau - s)\, h(s)\, ds \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.3)$$

This is an exact relation. It will now be shown that, by making certain approximations about $\phi_{cc}(\tau)$, $\phi_{cy}(\tau)$ is proportional to the impulse response of the plant, h(s).

The approximation to be used is that given by equation (A1.1.3) of Appendix A1.1. We then have

$$\phi_{cy}(\tau) \triangleq a \frac{N+1}{N} \lambda g \int_0^\infty \delta(\tau - s)\, h(s)\, ds$$

$$\triangleq \begin{cases} a \frac{N+1}{N} \lambda g\, h(\tau) & \text{if } \tau > 0 \\ \frac{1}{2}\, a \frac{N+1}{N} \lambda g\, h(0) & \text{if } \tau = 0 \end{cases} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.4)$$

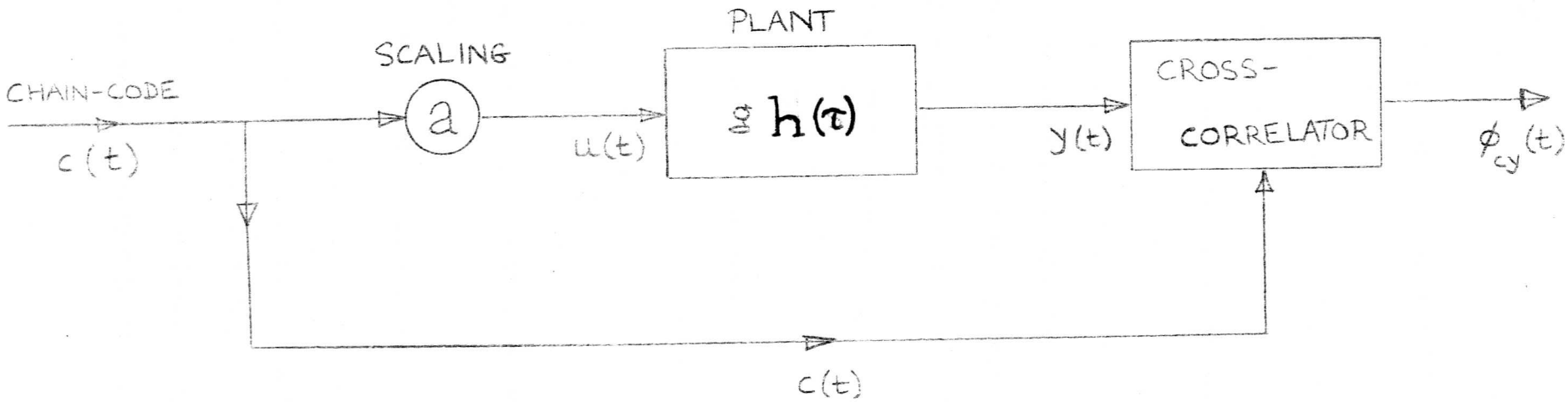


FIGURE 2.1 : BASIC CROSS-CORRELATING TECHNIQUE

(Note that $\int_0^T \delta(\tau)\, d\tau = \frac{1}{2}$, since the impulse can then be considered to straddle the lower limit of integration, being equally disposed about it. Some researchers have presented results where the first point of $\phi_{cy}(\tau)$ has twice the value given above, which leads one to suspect their technique, unless they have taken steps to double the value of $\phi_{cy}(o)$ automatically. The resulting error in step response estimate is of course very small.)

$\phi_{cy}(\tau)$ is therefore roughly proportional to the impulse response of the plant. An estimate of the step response at time T_r is given by integrating this, and scaling appropriately, so that

$$\hat{S}(T_r) = \frac{N}{a(N+1)\lambda} \int_o^{T_r} \phi_{cy}(\tau)\, d\tau \qquad \text{.....(2.5)}$$

whereas the true step response is

$$S(T_r) = \int_o^{T_r} g\, h(\tau)\, d\tau \qquad \text{.....(2.6)}$$

The true gain of the system is given by

$$g = S(\infty) \qquad \text{.....(2.7)}$$

so that an estimate of gain is

$$\hat{g} = \hat{S}(T) = \frac{N}{a(N+1)\lambda} \int_o^{T} \phi_{cy}(\tau)\, d\tau$$

If the impulse response has almost died away to zero in one period of the chain-code,

$$\int_o^T h(\tau)\, d\tau \doteq 1 \qquad \text{.....(2.8)}$$

so that $\hat{g} \doteq g$ .....(2.9)

If the impulse response has a long settling time compared with the chain-code period, equation (2.8) cannot be deemed to hold. In addition to this, errors will be brought forward from previous periods of the chain-code, due to the periodic nature of its auto-correlation function.

In the next section the exact value of $\hat{S}(T_r)$ will be derived from consideration of a practical method of identification based on the above theory but avoiding the use of a cross-correlator. The errors due to the approximations made in this section about $\phi_{cc}(\tau)$ and $h(\tau)$ will then be assessed.

2.3 Analysis of the Running Averages Technique under noise-free conditions

2.3(a) General Principles

The technique that will be analysed in this section was first proposed by Douce[24,47,48]. The system is shown in figure 2.2. The analysis will be restricted here to a single-parameter identifier; the multi-parameter case is discussed in Chapter 8.

A chain-code is fed to the 'input' of the plant, after scaling. In optimization work this will represent a perturbation on the current parameter value, so that there will be a steady state parameter value term entering the calculations, which will be referred to as the d.c. bias (see note in Appendix A1.1). This will be assumed zero at this juncture. The chain-code is also fed to a 'running averager' which calculates the average value of the chain-code over the previous p clock-intervals. The product $p\lambda$ is known as the 'running average time'. The running average is a periodic multi-level sequence, given by

$$r(t) = \frac{\mu_r}{T_r} \int_o^{T_r} c(t-\tau)\, d\tau \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.10)$$

where μ_r is the gain of the running averager and $T_r = p\lambda$.

We will now examine the value of the output of the sample-and-hold under noise-free conditions, that is for $\nu(t) \equiv o$.

The output of the multiplier is

$$m(t) = y(t) \times r(t) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.11)$$

and the output of the sample-and-hold is

$$q(T_r) = \frac{1}{T_i} \int_o^{T} m(t)\, dt \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.12)$$

where T_i is the integrator time-constant.

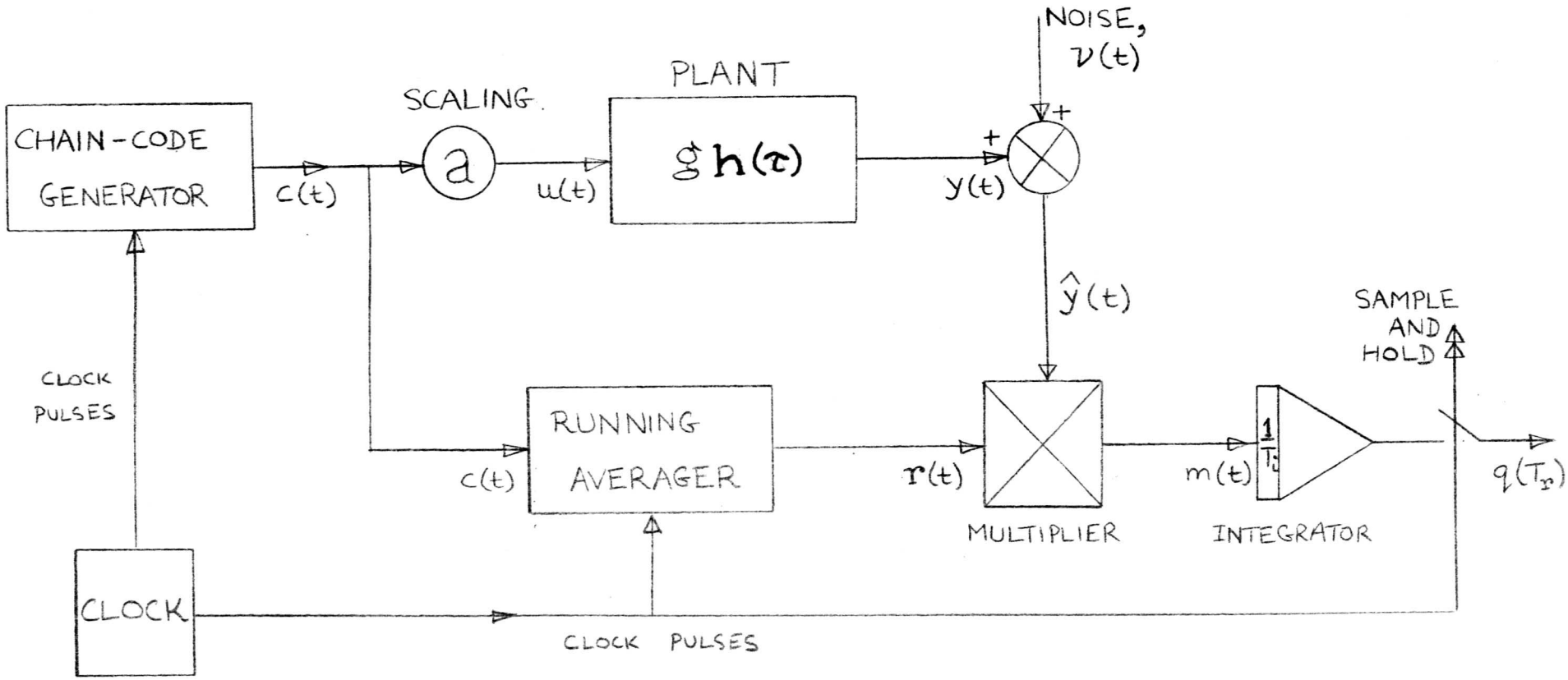


FIGURE 2.2 : SINGLE-CHANNEL IDENTIFICATION TECHNIQUE

Combining equations (2.2),(2.10), (2.11) and (2.12), and interchanging the order of integration gives

$$q(T_r) = \frac{a\mu_r T}{T_i T_r} g \int_o^{T_r} \int_o^{\infty} \phi_{cc}(\tau - s)\, h(s)\, ds d\tau \quad \ldots\ldots\ldots (2.13)$$

Comparing this with equations (2.3) and (2.5) we see that

$$q(T_r) = \mu_q \hat{S}(T_r) \quad \ldots\ldots\ldots (2.14)$$

where

$$\mu_q = \frac{a\mu_r (N+1)\lambda^2}{T_i T_r} \quad \ldots\ldots\ldots (2.15)$$

so that an estimate of step response is given by a scaled version of $q(T_r)$, for the values of T_r from 0 to T.

It will be noted that if T_r is set equal to T, the step response estimation process reduces to integrating the output of the plant over one period and scaling, as the running average then has a constant value. This incredibly simple technique, dispensing with running averager and multiplier, is of little value, however, as the d.c. bias error will be very large (see Section 2.3.(d)). It is also interesting to observe that, although the amplitude range of the running sum plotted against T_r is roughly symmetric about $T_r/2$, the step response estimate is not. In fact, with the approximation made in this section that the d.c. error is zero, the step response estimate of a first-order system will be a steadily increasing function of T_r. The apparent discrepency is explained when the phase of the running sum with respect to the output of the system is considered (see Appendix A3.4).

2.3(b) The Exact Solution

The exact evaluation of $\hat{S}(T_r)$ will now be derived, by substituting the exact form of $\phi_{cc}(\tau)$ into equation (2.3). It will be shown that the error due to the fact that $\phi_{cc}(\tau)$ is not a true impulse is usually small, but the error due to the d.c. bias of the code may be large, particularly if there is a bias superimposed on the input in addition to the inherent bias of the code, or if there is a d.c. offset somewhere in the plant. The

error due to the periodic nature of the code may also be large, but this can be reduced at the expense of speed of identification.

The value of $\hat{S}(T_r)$ will be evaluated for one period of the chain-code, and then the effect of the periodicity will be calculated.

From equations (2.3) and (2.5), the exact value of the step response estimate calculated from one period of the chain-code is

$$\hat{S}_1(T_r) = \frac{N}{(N+1)\lambda} g \int_{\tau=0}^{T_r} \int_{s=0}^{\infty} h(s)\, \phi_{cc}(s-\tau)\, ds\, d\tau \quad \ldots\ldots(2.16)$$

The range of τ can be split into two ranges, and the range of s into three ranges, so that substituting from equation (A1.1.1) gives

$$\begin{aligned}\hat{S}_1(T_r) = \frac{N}{(N+1)\lambda} g \Bigg[& \int_0^{\lambda} \left\{ \int_0^{\tau} \frac{N+1}{N}\left(1+\frac{s-\tau}{\lambda}\right) h(s)\,ds \right. \\ & \left. + \int_{\tau}^{\tau+\lambda} \frac{N+1}{N}\left(1+\frac{\tau-s}{\lambda}\right) h(s)\,ds - \frac{1}{N}\int_0^{\infty} h(s)\,ds \right\} d\tau \\ & + \int_{\lambda}^{T_r} \left\{ \int_{\tau-\lambda}^{\tau} \frac{N+1}{N}\left(1+\frac{s-\tau}{\lambda}\right) h(s)\,ds + \int_{\tau}^{\tau+\lambda} \frac{N+1}{N}\left(1+\frac{\tau-s}{\lambda}\right) h(s)\,ds \right. \\ & \left. - \frac{1}{N}\int_0^{\infty} h(s)\,ds \right\} d\tau \Bigg] \quad \ldots\ldots(2.17)\end{aligned}$$

This can be evaluated for any given values of $h(s)$. Without the bias terms, the step response estimate is

$$\hat{S}_d(T_r) = \hat{S}_1(T_r) + \frac{T_r}{(N+1)\lambda} \quad \ldots\ldots(2.18)$$

Consider now the effect that other periods of the auto-correlation function of the chain-code have on the step response estimate. The effect for a general function of $h(s)$ is illustrated in figure 2.3. The convolution process implies that the results of signals occurring at times in the past are given by considering positive values of s, so that as $t \to -\infty$, $s \to \infty$. Hence the effect of the periodicity is to introduce 'spikes' at $s = \tau + T$, $s = \tau + 2T$, etc., as well as at $s = \tau$. The step response estimate for all periods is therefore

$$\hat{S}(T_r) = \hat{S}_1(T_r) + \hat{S}_c(T_r) \quad \ldots\ldots(2.19)$$

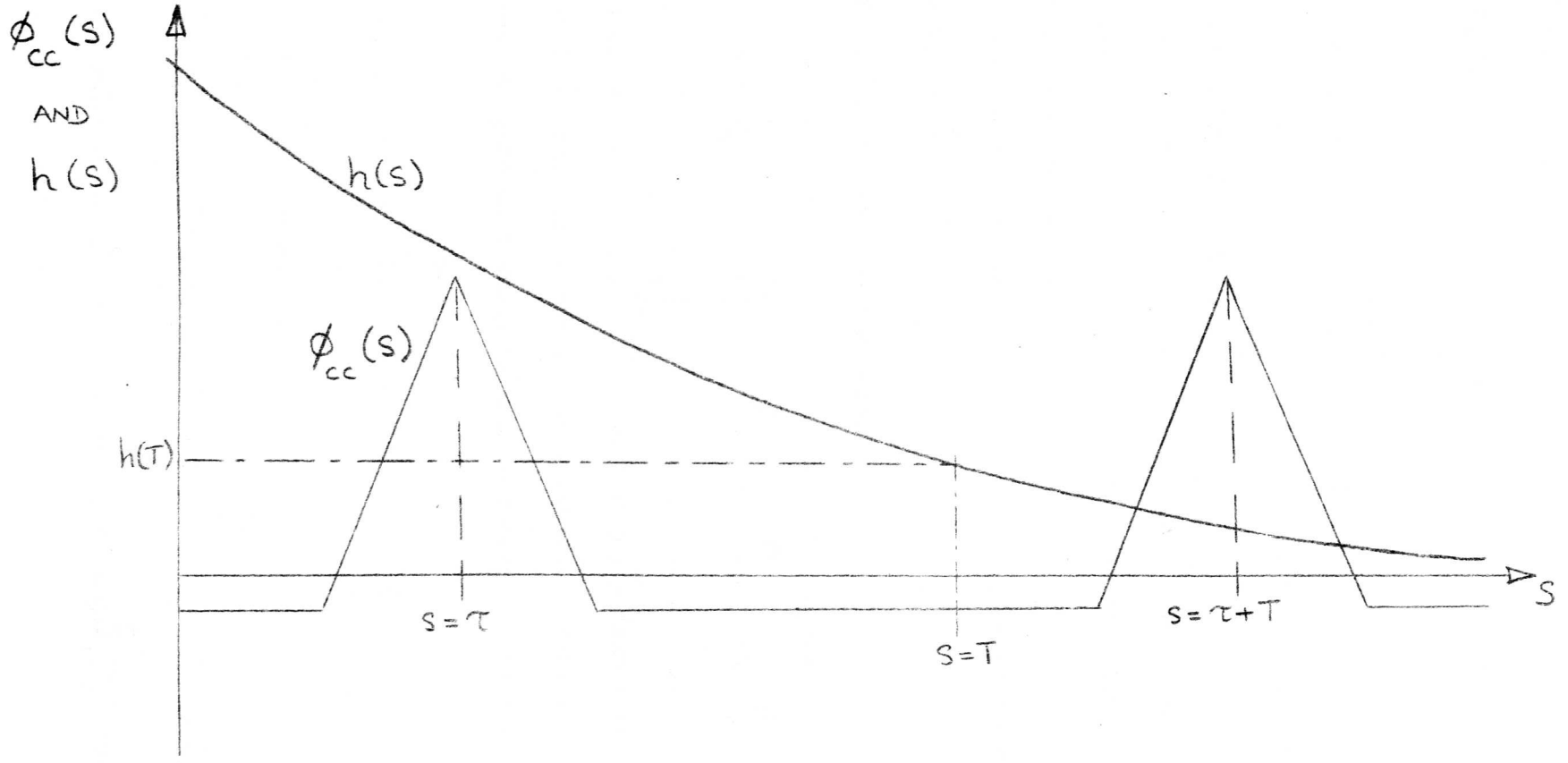


FIGURE 2.3 : THE EFFECT OF PERIODICITY

where $\hat{S}_c(T_r)$ is the correction term given by

$$\hat{S}_c(T_r) = \frac{g}{\lambda} \sum_{j=1}^{\infty} \int_{\lambda}^{T_r} \left\{ \int_{\tau-\lambda+jT}^{\tau+jT} \left(1 + \frac{s-\tau-jT}{\lambda}\right) h(s)\, ds + \int_{\tau+jT}^{\tau+\lambda+jT} \left(1 + \frac{\tau+jT-s}{\lambda}\right) h(s)\, ds \right\} d\tau \quad \ldots\ldots(2.20)$$

This correction term can be evaluated for any given values of $h(s)$.

2.3(c) Application to a Particular Problem

In order to illustrate the magnitude of $\hat{S}(T_r)$ given by equations (2.17) and (2.20), a particular system will be analysed, and the errors in step response estimate assessed.

Consider a plant whose transfer function is of first order, so that

$$h(s) = \frac{1}{T_s} \varepsilon^{-\frac{s}{T_s}} \quad \ldots\ldots(2.21)$$

where T_s is the plant time constant.

Substituting for $h(s)$ in equation (2.17) gives

$$\hat{S}_1(T_r) = g \left\{ \left(\frac{T_s}{\lambda}\right)^2 \left[\varepsilon^{-\frac{T_r}{T_s}} \left(2 - \varepsilon^{\frac{\lambda}{T_s}} - \varepsilon^{-\frac{\lambda}{T_s}}\right) + \left(\varepsilon^{-\frac{\lambda}{T_s}} - 1\right) \right] + \frac{1}{2} + \frac{T_s}{\lambda} - \frac{T_r}{(N+1)\lambda} \right\} \quad \ldots\ldots(2.22)$$

Since $h(s + jT) = \varepsilon^{-\frac{jT}{T_s}} h(s)$ for any j from 1 to ∞, the impulse response is of the same shape in every period. This is a special case, for which the general formula is not necessary. The correction term is then

$$\hat{S}_c(T_r) = g \frac{1}{\lambda} \left\{ \sum_{j=1}^{\infty} h(s + jT) \right\} \int_0^{T_r} \left\{ \int_{\tau-\lambda}^{\tau} \left(1 + \frac{s-\tau}{\lambda}\right) h(s)\, ds + \int_{\tau}^{\tau+\lambda} \left(1 + \frac{\tau-s}{\lambda}\right) h(s)\, ds \right\} d\tau$$

i.e. $$\hat{S}_c(T_r) = \frac{g}{\varepsilon^{T/T_s} - 1} \left(\frac{T_s}{\lambda}\right)^2 \left(\varepsilon^{-\frac{T_r}{T_s}} - 1\right)\left(2 - \varepsilon^{\frac{\lambda}{T_s}} - \varepsilon^{-\frac{\lambda}{T_s}}\right) \quad \ldots(2.23)$$

The value of $\hat{S}(T_r)$ is then given by adding $\hat{S}_1(T_r)$ and $\hat{S}_c(T_r)$ from equations (2.22) and (2.23).

Figures 2.4(a) and 2.4(b) show the true and estimated step responses, with and without the bias terms, for a system with $g = 1$, $\lambda = 1$, $N = 127$, and $T_s = 127$ and $127/3$, that is, for a chain-code period equal to the system time constant and three times the system time constant. These curves will be discussed in terms of the various errors arising.

2.3(d) The Magnitude of the Errors

There are four types of error inherent in the identification system. These can be classified as:

(i) The impulse-shape error, due to the fact that the 'spike' of the auto-correlation function is not a true impulse.

(ii) The periodicity error, due to the repetitive nature of the chain-code.

(iii) The integration-time error, due to the finite nature of the integration process.

(iv) The d.c. bias error, due to d.c. at the input of the plant, or inherent elsewhere in the process.

The magnitudes of these errors will be considered in turn.

(i) The impulse-shape error:

It has been shown in equations (2.17) and (2.18) how the estimate of step response can be calculated for the case where the d.c. bias error and periodicity error are not present. The impulse-shape error ratio can then be defined as:

$$\eta_I = \frac{\hat{S}_d(T_r)}{S(T_r)} - 1 \qquad \text{.....(2.24)}$$

for all values of T_r. This is the ratio of the error due to the impulse-shape alone to the true step response. This ratio, and the others to be introduced in this section, have been chosen to have a positive value when the practical result is greater than the true result .

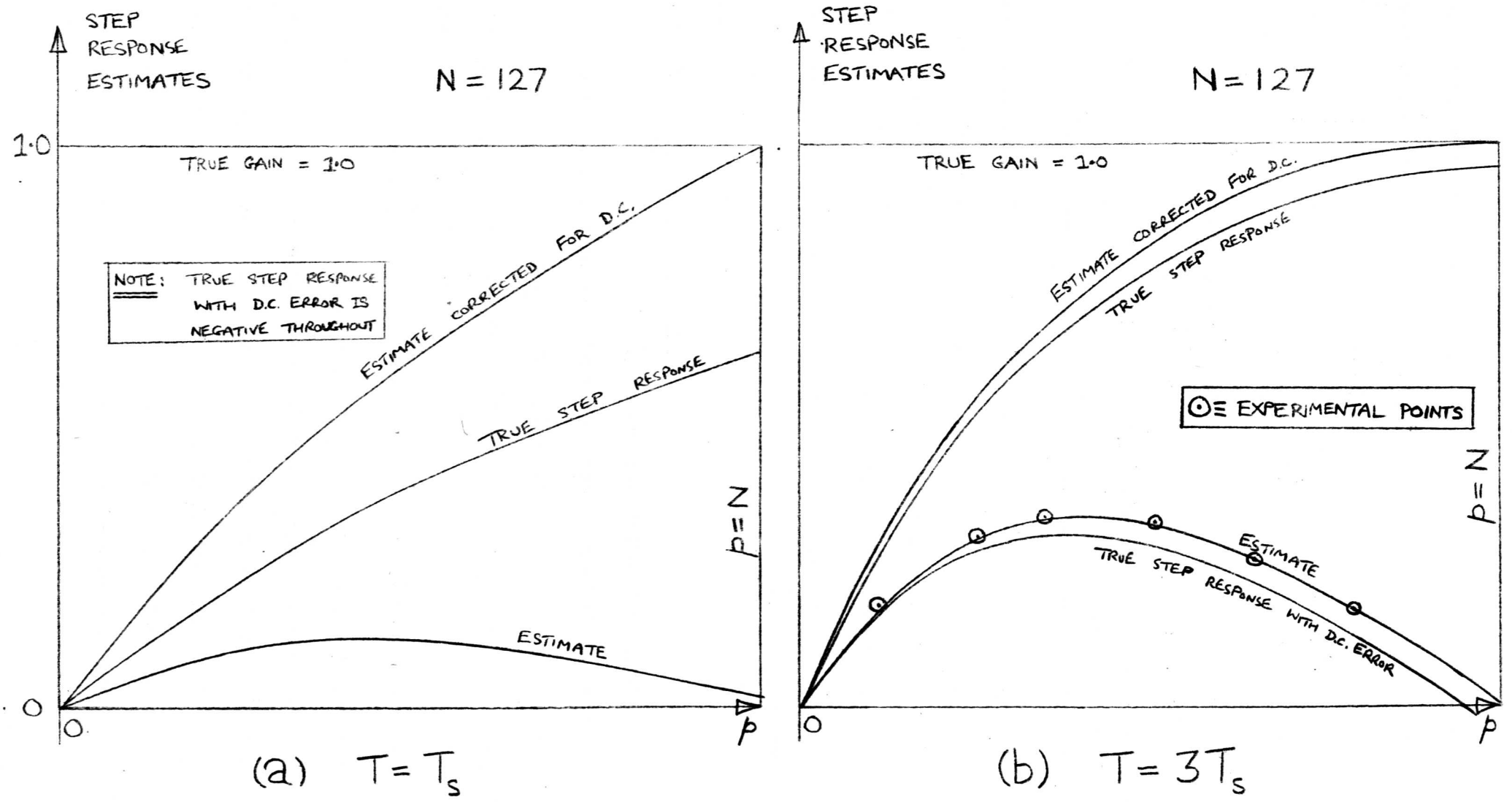


FIGURE 2.4 : STEP RESPONSE ESTIMATES FOR A FIRST-ORDER SYSTEM

For the first-order system of the previous section, $|\eta_I|$ can be shown to be under 1% over most of the range. Clearly, for a given value of T, as N increases, the 'spike'will tend to a true impulse, and $|\eta_I|$ will decrease. The error is usually so small, however, that such considerations become insignificant.

(ii) The periodicity error

The error due to past 'spikes' of the auto-correlation function is given by equation (2.20). The periodicity error ratio can be defined as

$$\eta_P = \frac{\hat{S}_c(T_r)}{S(T_r)} \qquad \text{.....(2.25)}$$

for all values of T_r. This is the ratio of the error due to past triangular 'impulses' to the true step response.

For the first-order system, η_P can be shown to have the value 0.582 when $T = T_s$, over the whole range, and 0.052 when $T = 3T_s$, over the whole range. It is constant because of the fact that the experimental impulse response has a similar shape over all given periods of its length. This will not be true for a system of higher order, or where there are transport lags.

(iii) The integration-time error

It is not possible to choose an integration time greater than T, and in some cases a time less than this may be preferable from considerations of the likely disturbances. This gives rise to an error in gain estimate given by $S(T_r) - S(\infty)$, if the other errors are ignored. The integration-time error ratio is then defined as

$$\eta_T = \frac{S(T_r)}{S(\infty)} - 1 \qquad \text{.....(2.26)}$$

For the first-order system, η_T can be visualized from figures 2.4(a) and 2.4(b) by subtracting 1 from the true step response. It clearly decreases as p increases, having final values of -.37 and -.05 for the two cases. It can be seen that the periodicity error and the integration time error are complementary, so that the signal that is lost by using a finite integration time, equal to the period of the chain-code, is regained by

the addition of the other step responses at periodic intervals, so that the step response estimate at $T_r = T$ with the periodicity term added is unity. This only applies to a first-order system, where the impulse response has a similar shape for all periods. In general the effect of periodicity may tend to counteract the effect of truncating the step response; whether it enhances or reduces the total error depends on the step response.

(iv) The d.c. bias error

Consider the d.c. inherent in the ± 1 chain-code. An equation analogous to equation (2.18) can be deduced to relate the true step response with d.c. error, S_e, to the true step response without error, namely

$$S_e(p) = S(p) - \frac{p}{N+1} \qquad \ldots\ldots(2.27)$$

The bias error ratio can be defined as

$$\eta_B = \frac{S_e(p)}{S(p)} - 1 \qquad \ldots\ldots(2.28)$$

For a plant of any order,

$$\eta_B = - \frac{p/(N+1)}{S(p)} \qquad \ldots\ldots(2.29)$$

For optimization work, the sign of the step response estimate is all important to ensure convergence to the optimum. (The magnitude affects the rate of convergence, but is not as crucial as the sign). Ignoring the impulse-shape error, the estimate of the step response with d.c. error is of the same sign as the step response without error if

$$\hat{S}(p) - \frac{p}{N+1} > 0 \qquad \ldots\ldots(2.30)$$

assuming $S(p) > 0$. It is therefore essential to choose a value for p for which this condition is satisfied at all times.

Since the bias error is proportional to p, it is advisable to work in the region where p is small, providing the value of $\hat{S}(p)$ is reasonable in this region. There are many cases occurring in practice where $\hat{S}(p)$ for small values of p is not representative of the final value given by $S(\infty)$. Two examples of this are provided by systems involving:

(i) a transport lag, such as might occur in a process involving the transmission of fluid from one part of the plant to another, or in measuring and transducing elements; and

(ii) non-minimum phase elements, in which the step response has the opposite sign at small values of p to its value at $p \to \infty$.

For the first-order system, η_B is shown in figure 2.5. For $T = T_s$, $\eta_B < -1$ over the whole range of p, indicating that $S_e(p)$ and $S(p)$ have opposite signs for all values of p. For $T = 3T_s$, $\eta_B > -1$ for all values of p less than 120. In either case $|\eta_B|$ is very large, showing that the d.c. can swamp the true result.

In an optimizing system, there will be a d.c. bias on the parameter due to its unperturbed value. There may also be d.c. present at other stages of the process. In such cases, the effect of the d.c. will be to produce an error in the step response estimate. It has been seen how even the small d.c. level inherent in the chain-code can create a large error, so it can be deduced that additional bias will create an even more significant error. It is therefore essential to consider ways of eliminating the bias error. These will be considered in the next two chapters.

The system chosen for analysis in this section has been of first-order. For any higher-order system, a similar analysis can be carried out if the likely step response is known. Owing to the diversity of possible systems, illustrative examples are not given here, but an idea of the performance of any system can be gleaned from a comparison with the first-order system discussed above. For example, for a damped second-order system, with a given settling time, the errors will be of roughly the same order as for the first-order system with the same settling time.

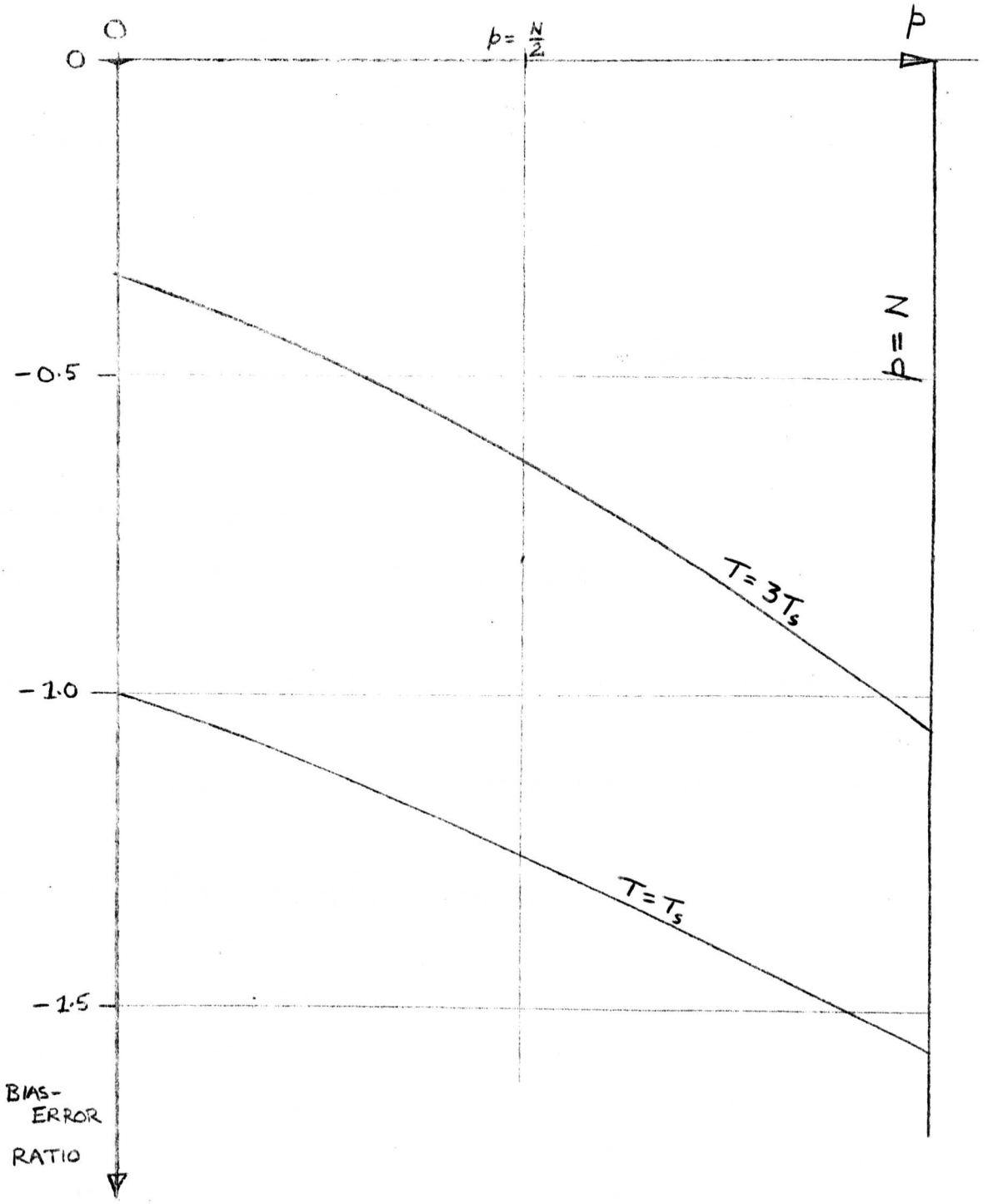


FIGURE 2.5 :

THE BIAS-ERROR RATIO FOR A FIRST-ORDER SYSTEM

2.3(e) Experimental Results

The arrangement of Figure 2.2 was implemented experimentally. At the time that this work was carried out, no digital on-line computer was available, and so the chain-code generator, running averager and multiplier were built in the laboratory. The plant was simulated on the Solartron 247 analogue computer, which also handled the integrating and sample-and-hold elements. The practical details are given in Appendix A2.1.

The system chosen was of first-order, with $T = 3T_s$. The results are plotted on Figure 2.4(b). It will be observed that the practical points are close to the theoretical curve, showing the validity of the theory. In particular, the significance of the d.c. error is verified.

2.4 The Behaviour of the System Under Noisy Conditions

In this section, the performance of the identification system will be analysed under noisy conditions. In practice, noise will always be present and may be very significant; if it were not so, there would be no need to use correlation techniques, and the step response could be found by simpler methods.

2.4(a) Derivation of the variance formula in the general case

The variance of the step response estimate will now be evaluated for the general case where noise is present at the output of the plant. The most useful concepts[50,51,52], for random signals analysis are those of power spectra and correlation functions. The power spectrum of the output of the multiplier is derived in Appendix A2.2, where it is seen that the resulting function is not amenable to analysis. The correlation function approach leads to a simpler solution, and is the method to be described now.

Referring to figure 2.2, the effective output of the plant is given by

$$\hat{y}(t) = y(t) + \nu(t) \qquad (2.31)$$

The output of the multiplier is

$$m(t) = r(t) \times \hat{y}(t) \quad \ldots\ldots(2.32)$$

$$= \left[r(t) \times y(t)\right] + \left[r(t) \times \nu(t)\right] \quad \ldots\ldots(2.33)$$

$m(t)$ can therefore be considered to have two components, one due to $y(t)$, the other due to $\nu(t)$, which can be referred to respectively as the signal and noise components of $m(t)$. Defining the noise component as $m_\nu(t)$, the noise component of $q(T_r)$ is given by

$$q_\nu(T_r) = \frac{1}{T_i}\int_o^T m_\nu(t)dt \quad \ldots\ldots(2.34)$$

Expressing $m_\nu(t)$ in terms of its Fourier components gives

$$m_\nu(t) = \sum_{n=1}^{\infty} A_n(\cos nw_o t + \phi_n) \quad \ldots\ldots(2.35)$$

Substituting in equation (2.34) we have

$$q_\nu(T_r) = \frac{1}{T_i}\int_o^T \left[\sum_{n=1}^{\infty} A_n(\cos nw_o t + \phi_n)\right] dt = q_{av} \pm D_q \quad \ldots\ldots(2.36)$$

where q_{av} is the average value and D_q is the standard deviation.

It has been assumed that the noise has zero mean, so that $q_{av} = o$.

The variance is then given by the phase-average of q^2, so that

$$D_q^2 = \frac{1}{2\pi}\int_o^{2\pi} q_\nu^2 \, d\phi \quad \ldots\ldots(2.37)$$

Substituting for q_ν gives

$$D_q^2 = \frac{2}{T_i^2}\int_o^T (T-\tau)\,\phi_{m_\nu m_\nu}(\tau)\,d\tau \quad \ldots\ldots(2.38)$$

It is shown in Appendix A2.3 that if $r(t)$ and $\nu(t)$ are mutually uncorrelated then

$$\phi_{m_\nu m_\nu}(\tau) = \phi_{rr}(\tau) \times \phi_{\nu\nu}(\tau) \quad \ldots\ldots(2.39)$$

(In practice there is bound to be some correlation between $y(t)$ and $\nu(t)$. This will give rise to cross terms involving 4th-order integration. It is not unreasonable to neglect the cross-correlation for the purposes of an approximate analysis.)

Hence the variance in step response is

$$D_s^2 = \frac{D_q^2}{\mu_q^2} = \frac{2}{\mu_q^2 T_i^2}\int_o^T (T-\tau)\phi_{rr}(\tau)\phi_{\nu\nu}(\tau)\,d\tau \quad \ldots\ldots(2.40)$$

It is therefore necessary to know the auto-correlation functions of the output of the running averager and the noise. $\phi_{rr}(\tau)$ is derived in Appendix A2.4, with the assumption that $\phi_{cc}(\tau)$ is a true impulse with d.c. bias. $\phi_{\nu\nu}(\tau)$ can be estimated in practice from the likely properties of the noise. The solution for a particular type of noise will now be given.

2.4(b) Variance of estimate for band-limited noise

Consider the case where the noise at the output of the plant is filtered white noise, where the filter is a single lag of time constant T_ν. This will take account of:

(i) band-limited noise added to the output of the plant, and

(ii) white noise elsewhere in the plant, where the plant can be considered to act as a low-pass filter to this noise.

For such a case

$$\phi_{\nu\nu}(\tau) = \sigma_\nu^2 \, \varepsilon^{-\frac{|\tau|}{T_\nu}} \qquad \text{.....(2.41)}$$

where σ_ν^2 is the mean-squared value of the filtered noise, which can be considered as its power.

The expression for $\phi_{rr}(\tau)$ can be derived from Figure A2.4.2 and substituted, with the expression for $\phi_{\nu\nu}(\tau)$, into equation (2.40). This can then be integrated by dividing the range of τ into three, namely

$$0 < \tau \leq T_r$$

$$T_r \leq \tau \leq T - T_r$$

$$T - T_r \leq \tau < T$$

Putting $\alpha = \frac{T_\nu}{T}$ and $\beta = \frac{T_r}{T} = \frac{p}{N}$, the resulting expression for the variance is

$$D_s^2 = 2R \, \frac{N^2}{N+1} \, B \qquad \text{.....(2.42)}$$

where $R = \frac{\sigma_\nu^2}{a^2}$ = the ratio of the power of the noise to the power of the chain-code

and $$B = \alpha\left[\beta - \alpha\beta + \frac{N}{N+1}\beta^2\ (\alpha-1)\right] + \alpha^2\left[2\alpha-1+\varepsilon^{-\frac{\beta}{\alpha}}(-\beta+1-2\alpha)\right.$$
$$\left. + \varepsilon^{\frac{\beta-1}{\alpha}}(\beta-2\alpha) + \varepsilon^{-\frac{1}{\alpha}}(\beta+2\alpha-\frac{N}{N+1}\beta^2)\right] \quad \ldots\ldots\ldots(2.43)$$

It will be noticed from these equations that:

(i) The variance is proportional to the power ratio, R, as expected.

(ii) The variance is roughly proportional to N if α and β are fixed.

The function B is illustrated in Figure 2.6, from which we observe that:

(i) For small values of α the variance has a maximum value when $\beta \simeq 0.5$, tending to zero as β tends to zero or to infinity. The approximate symmetry about $\beta = 0.5$ is due to the nature of the variation of the running average amplitude with β.

(ii) For large α the variance increases steadily with β.

(iii) It can be shown that, for $\beta = 0.5$, B has a maximum value when $\alpha \simeq 0.2$. As $\alpha \to 0$, $B \to 0$, indicating that the high frequency components of noise are eliminated by the correlation process. Also $B \to 0$ as $\alpha \to \infty$, showing that low frequency components of noise are filtered out by the d.c. elimination process.

2.4(c) The Choice of Parameter Values

The choice of the values of β and N to achieve the "best" estimate under noisy conditions will now be considered.

The most useful concept for this analysis is the ratio of the amplitude of the signal component of the gain estimate to the amplitude of the noise component of the estimate (given by the standard deviation). This ratio can be termed the signal-to-noise amplitude ratio. Alternatively the power ratio could be used, but for the present analysis only the amplitude ratio will be considered.

It is given by

$$\eta_c = \frac{\hat{S}\ (T_r)}{D_s} \quad \ldots\ldots\ldots\ldots\ldots\ldots(2.44)$$

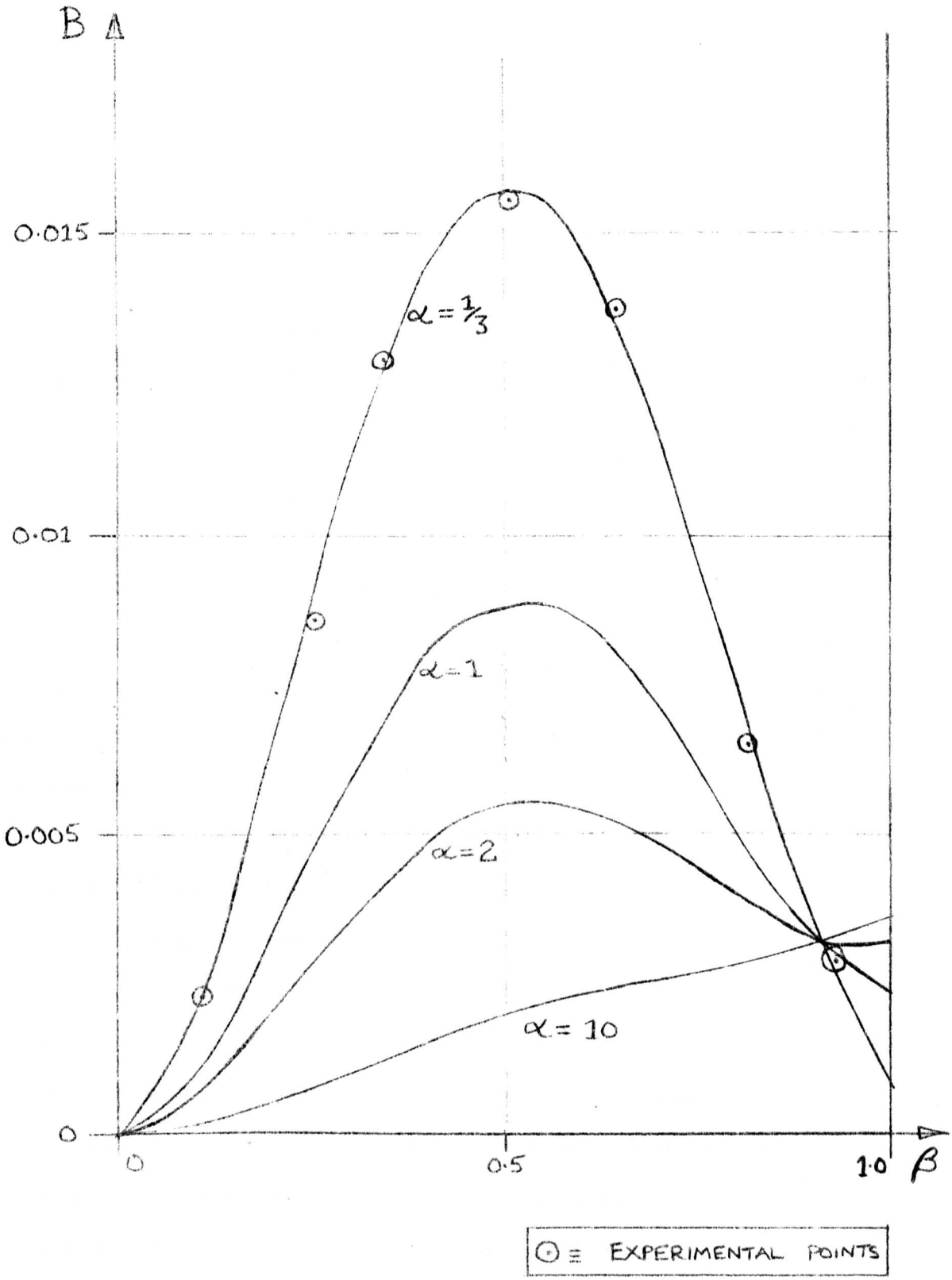


FIGURE 2.6:

VARIATION OF B WITH α AND β FOR N=127

For any plant, with any kind of noise at the output, η_c can be evaluated using equations (2.17), (2.19), (2.20),(2.40) and (2.44).

It has been shown that the variance is proportional to N. As the signal component has been scaled to be independent of N, η_c is proportional to the square-root of N. N should therefore be chosen to be as small as possible, providing enough ordinates are used to adequately define the response of the plant. Clearly for high-order plants a larger N will be necessary than for a 1st-order plant.

For the particular case of the first-order system discussed in Section 2.3(c) with band-limited noise, η_c can be evaluated from equations (2.22), (2.23) and (2.42). Figure 2.7 shows η_c plotted against β for $\alpha = 1$ and $N = 127$, for two values of T/T_s. From this it can be seen that the signal-to-noise ratio is greatest for small values of β. But it has been noted in Section 2.3(d), subsection (iv), that in practice it is often necessary to choose a large value for β to overcome errors due to the shape of the impulse response at low values of its argument. There are two possible solutions:

(i) Make a compromise by choosing an intermediate value for β and hope that neither the noise nor the d.c. bias error will prove too significant, or

(ii) Develop a system to eliminate the d.c. bias error. This will reduce the discrepancy in the signal component of the gain estimate, especially for large values of β, and thereby improve the signal-to-noise ratio, enabling a larger value of β to be used than was previously possible. Development and analysis of such a system is the subject of the next two chapters.

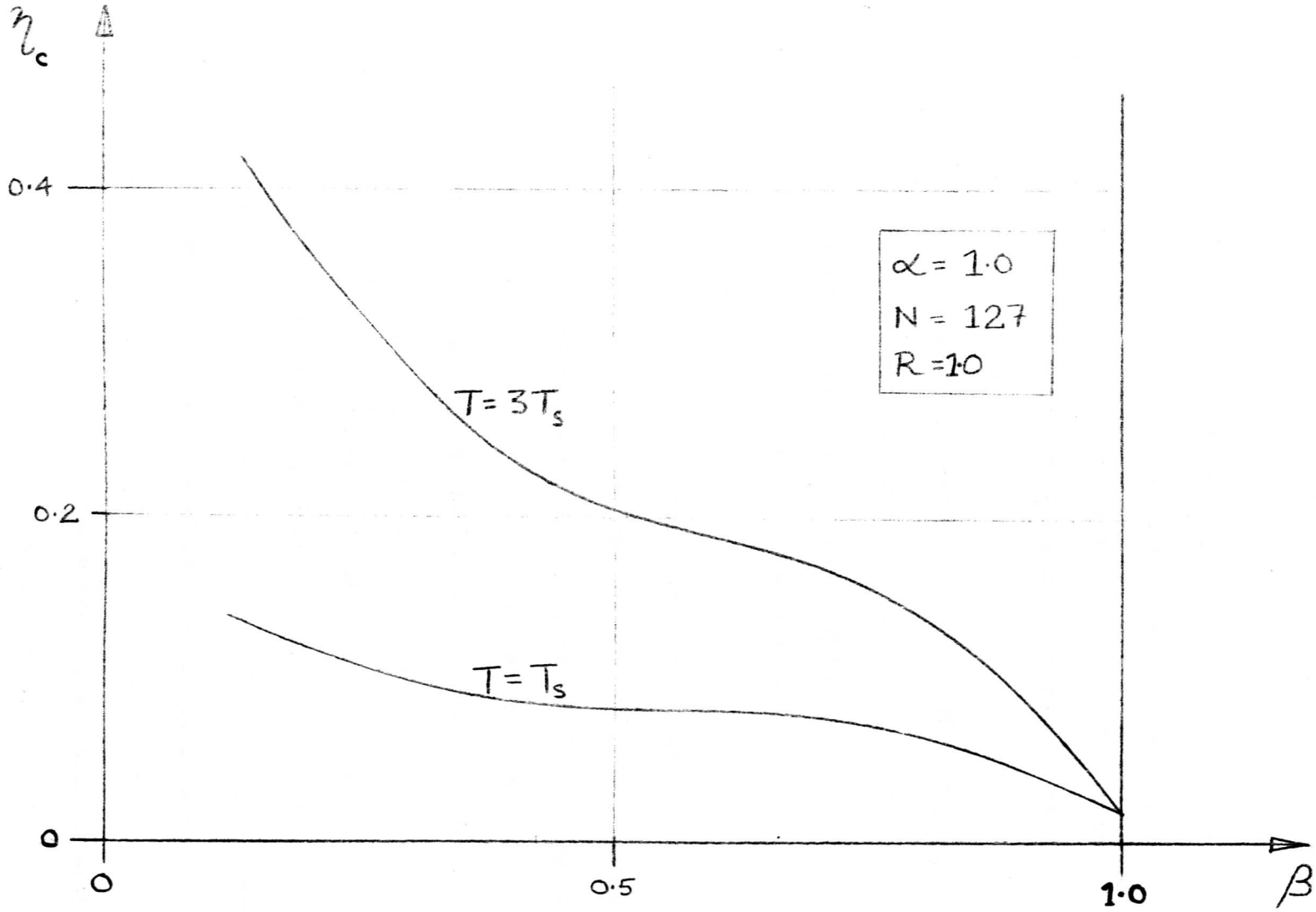


FIGURE 2.7 : SIGNAL-TO-NOISE RATIO FOR BAND-LIMITED NOISE

2.4(d) Experimental results

The system described in Section 2.3(c) was studied under noisy conditions. Details of the noise generator are given in Appendix A2.5. The analog computer was switched to an iterative mode in order to perform the integrate, sample-and-hold, reset series of operations repeatedly, and the output of the integrator was recorded on a printer during each 'hold' operation. In this way, a large number of results was obtained, in order to give a statistically reasonable estimate of the variance, which was then calculated off-line.

The results are shown on figure 2.6. The case chosen to verify the theory was with N = 127, $T = 3T_s$ and $T_\nu = T_s$, so that $\alpha = \frac{1}{3}$. It is seen that the experimental points agree well with the theoretical curves, and thus support the predictions made in this chapter.

2.5 Conclusions

It has been shown that the estimate of the gain of the path between a parameter of a plant and the performance index, obtained by a chain-code cross-correlation technique, may be seriously in error if the d.c. bias inherent in the system is not estimated and removed. It was also shown that the periodicity of the chain-code and the necessarily finite integration time may both significantly affect the estimate of gain, but not usually to the extent that the d.c. bias error does.

The system was examined under noisy conditions, and the variance of the gain estimate was evaluated. It is seen that the variance may be large when the parameters ofthe identification system are chosen to give the noise-free estimate that is closest to its correct value, and, conversely, that for the parameter values where the variance is small, the mean estimate may be grossly in error due to the d.c. bias. It is concluded that the method is seriously handicapped by the bias error, and that it is very advisable to consider methods of removing it. Such methods are discussed in the next Chapter.

CHAPTER 3

MODIFIED METHODS OF SYSTEM IDENTIFICATION

3.1 Introduction

The analysis of the previous chapter illustrated the extent to which d.c. bias at the output of theplant can affect the estimate of step response when using pseudo-random binary perturbations. When attempting to optimize the plant parameters, a reasonable estimate of the gain may be possible in particular cases, but in general the magnitude of the gain estimate will be severely in error, and in some cases its sign will be incorrect, leading to a false adjustment in the controlled parameters of the plant.

Possible ways of eliminating the d.c. bias will therefore be considered. Direct methods, such as filtering the plant output with a high-pass filter, lead to errors in the estimate of impulse and step responses, and an indirect method of d.c. elimination is preferable. Several such methods are developed in this chapter, for use in various environments.

Section 3.2. deals with the analysis of the d.c. elimination problem, for a single-parameter plant, and a formula for the unbiassed gain is given. Several methods for implementing this formula in a practical system are outlined in Section 3.3, for use with and without a digital process-control computer. The conclusions are presented in Section 3.4.

3.2 Analysis of the d.c. bias elimination problem

Consider the basic technique for estimating system gain for a single parameter plant, outlined in Section 2.2. This technique involves cross-correlating the plant output with the plant input to give the impulse response, and integrating this over a predetermined time to give the value of the step response at this time.

It is shown in Appendix A.3.1 that the effect of d.c. on the output of the plant (see Figure 3.1) is equivalent to adding some value to each ordinate of the impulse response. The following method for removing the effect of the d.c. can therefore be postulated:

(i) Ensure that the true impulse response dies away to nearly zero within one period of the input signal. (This is a normal requirement for identification methods in any case, as it effectively eliminates errors that would otherwise be carried over from one period to the next, and also yields a gain estimate close to the value of the step response at infinite time).

(ii) Average the last k ordinates of the estimated impulse response, where k is some predetermined number. This will give an estimate of the d.c. level of the impulse response.

(iii) Subtract this estimated d.c. level from each of the ordinates of the estimated impulse response.

(iv) Integrate the new estimate of impulse response to give the unbiassed step response.

Expressing this method in integral form gives

$$\hat{S}(T_r) = \left(\frac{T}{T+\lambda}\right) \frac{1}{a\lambda} \int_0^{T_r} \left[\phi_{cy}(\tau) - \frac{1}{\Delta} \int_{T-\Delta}^{T} \phi_{cy}(s)\,ds \right] d\tau \quad ..(3.1)$$

where $\hat{S}(T_r)$ = the estimated step response at time T_r,

and $\Delta = k\lambda$

(Clarke[53] uses a similar expression for the case where $T_r = (T - \Delta)$. He derives it from a least-squares estimate).

3.3 Practical methods of bias elimination

We will now consider methods of implementing equation (3.1) or its equivalent. General expressions for the gain estimating process, for use with or without a digital computer, are derived and a scheme

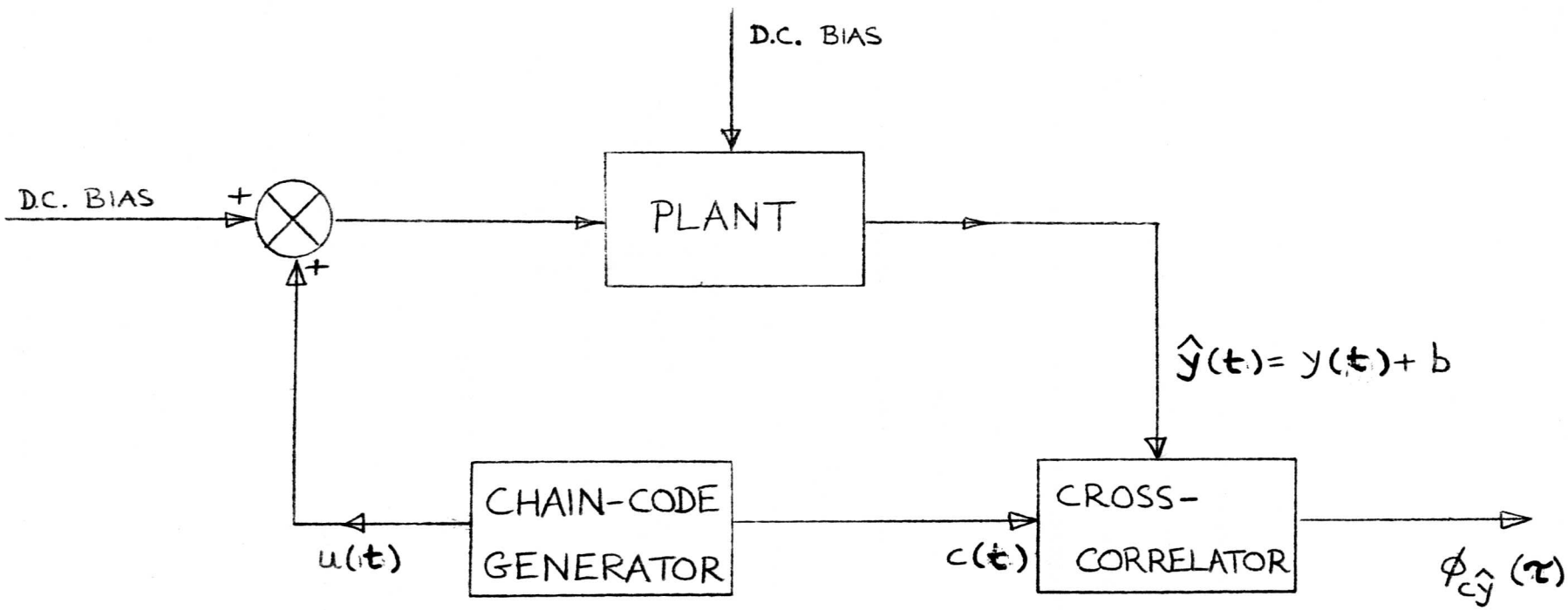


FIGURE 3.1: BASIC CORRELATION TECHNIQUE FOR D.C. BIAS ANALYSIS

that is particularly suitable when a digital computer is not available is then postulated, with a description of the necessary hardware. Evidence for the validity of the methods is given at the end of the section.

3.3(a) General Techniques

Equation (3.1) can be evaluated directly using a digital computer, by carrying out the cross-correlation, evaluating the d.c. shift, shifting the cross-correlation accordingly, and summing over p ordinates. (If dynamic compensation (see Chapter 9) is not required, it is not necessary to shift the cross-correlation; it is merely necessary to subtract p times the estimated bias from the biassed value of $\hat{S}(p)$). This is not necessarily the best method when using a digital computer, and is not easily implemented by what can be termed "independent hardware", that is, purpose-built equipment for use in environments where a digital computer is not available.

An alternative form of equation (3.1), suitable for applications with or without a digital computer, is derived in Appendix (A3.2) as

$$\hat{S}(T_r) = \frac{1}{(T+\lambda)a\lambda} \int_o^T y(t)\left[\sigma_c(t - T_r : t) - \frac{T_r}{\Delta} \times \sigma_c(t : t+\Delta)\right] dt \quad \ldots\ldots(3.2)$$

where $\sigma_c(q:r) = \int_q^r c(t)dt$

In the special case where $T_r = T$, equation (3.2) reduces to

$$\hat{S}(T) = \frac{1}{(T+\lambda)a\lambda} \int_o^T y(t)\left[1 - \frac{T}{\Delta}\ \sigma_c\ (t : t+\Delta)\right] dt \quad \ldots\ldots(3.3)$$

involving only one running sum, but for any other value of T_r, two running sums are required.

Equation 3.2 can be written as

$$\hat{S}(T_r) = \mu \int_o^T y(t)\ \rho_{T_r,\Delta}(t)\ dt$$

where $\mu^{-1} = (T+\lambda)\ a\lambda$ $\quad \ldots\ldots(3.4)$

and $\rho_{T_r,\Delta}(t) = \sigma_c\ (t - T_r : t) - \frac{T_r}{\Delta}\ \sigma_c(t : t+\Delta)$

ρ is a cyclical function of t whose N values depends on T_r and Δ.

Equation (3.4) is easier than equation (3.1) to implement on a digital computer because:

(i) ρ can be calculated and stored at the start of the program, before on-line working begins. It will be noted that $\Delta\rho$ takes on only integer values, thus simplifying computation. The whole of equation (3.4) can then be evaluated in fixed point.

(ii) Only N multiplications are needed before the summation, whereas equation (3.1) requires N cross-correlations, each with N shift-multiply-add computations.

For digital computing work, equation (3.4) can be expressed in its equivalent discrete form as

$$\hat{S}(p) = \eta \sum_{m=1}^{N} y(m)\,\rho(m) \qquad \text{.....(3.5)}$$

where $\eta^{-1} = (N+1)a$ and $p = T_r/\lambda$

Computing time can be saved if the latest estimate of $\hat{S}(p)$ is related to the estimate made one clock time earlier, assuming $\hat{S}(p)$ is required after every clock pulse. Denoting the estimate made after the j-th clock pulse by $\hat{S}(p,j)$, we have, from Appendix A3.3,

$$\hat{S}(p,j) = \hat{S}(p,j-1) + \eta\left[y(j) - y(j-N)\right]\rho(j) \qquad \text{.....(3.6)}$$

This is considerably quicker to compute than equation (3.4), as only one multiplication is required, once ρ has been found. An alternative form for ρ, which may prove easier to implement in certain cases, is discussed in Appendix A3.4.

3.3(b) Experimental evaluation of gain without using a digital computer

When a digital process-control computer is not available, equation (3.1) is not easily represented in practice, but we can implement equation (3.6) with the system of Figure 3.2

The "compound running summer" evaluates $\rho_{p,k}(t)$ continuously. It can be made up from two running summers (electronic or electro-mechanical) of one of the types discussed in Appendix A2.1; it is represented in Figure 3.3. (Alternatively, equation (A3.4) can be simulated by an

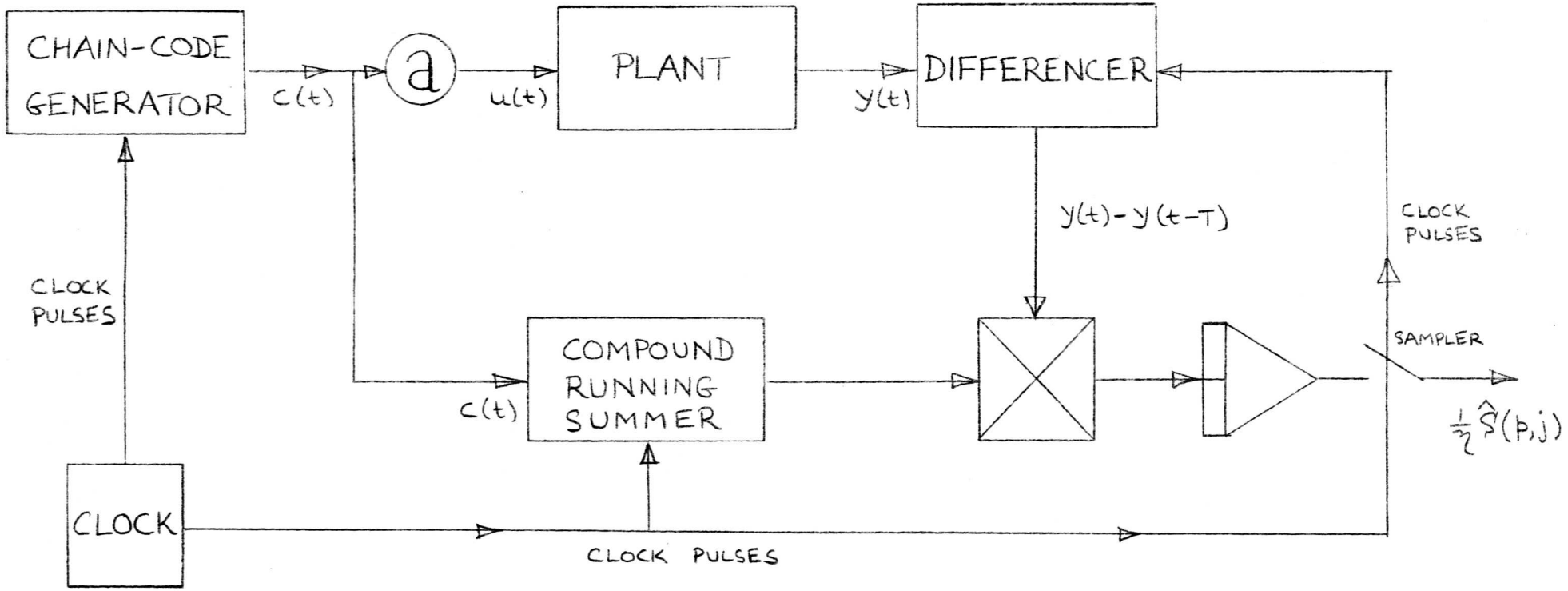


FIGURE 3.2 : "INDEPENDENT HARDWARE" FOR GAIN ESTIMATE WITH D.C. REMOVAL

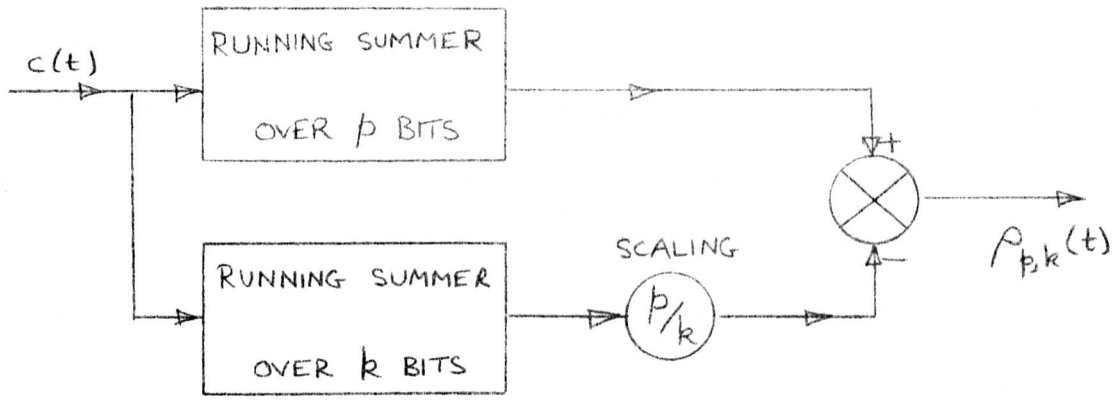


FIGURE 3.3: COMPOUND RUNNING SUMMER

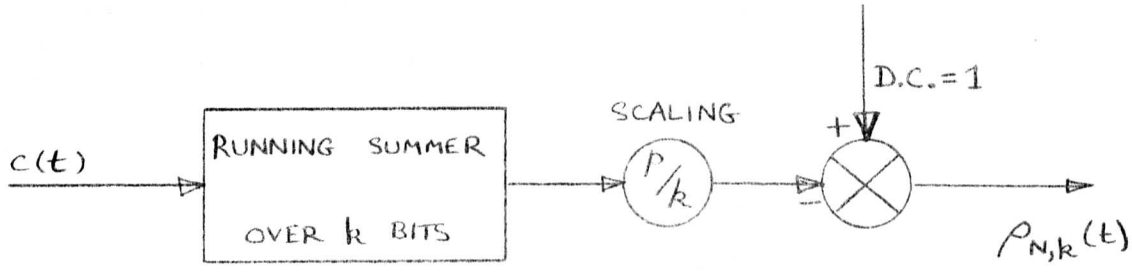


FIGURE 3.4: RUNNING SUMMER FOR $\hat{S}(N)$

equivalent system). The correct phasing of each running summer is essential.

The "differencer" is an N-word storage device giving an output of $\left[y(t) - y(t - T)\right]$. An electro-mechanical capacitor - drum device[49] is well suited to this application; at any time the past N values of $y(t)$ will be stored on the drum; when the latest value of $y(t)$ is available, $y(t - T)$ is read from the drum, subtracted from $y(t)$, and then replaced on the drum by $y(t)$.

The integrator performs the summing operation of equation 3.6, and is sampled at each clock pulse to give the latest estimate of $\hat{S}(T_r)/$.

For evaluating $\hat{S}(T)$, using equation (3.3), the compound running summer can be simplified by replacing one running summer by a d.c. term of unity, as shown in figure (3.4).

(The integrator and differencer correspond to the second running averager used by Ng, et al,[48], at the output of the multiplier in place of the integrator. The two methods are equivalent, one corresponding to evaluation of equation (3.6), the other to the continuous evaluation of equation (3.2).)

3.3(c) The validity of the estimation methods

Successful off-line computation on the Elliott 4130 computer (see Chapter 9), based on equation (3.2), indicates that the theory presented above is sound. In addition to this, the experiments of Chapter 2 indicate that an extension of the running summer principle to the experimental evaluation of equations (3.2) and (3.6) is reasonable.

It is therefore fair to conclude that the method is theoretically correct and physically realizable. Its limitations, in terms of errors in gain estimates, are discussed in the next chapter for noisy and noise-free conditions.

3.4 Conclusions

Methods for eliminating the errors in step response estimates due to d.c. bias at the output of the plant are available. Several such methods have been described in this chapter, some for use with a digital process-control computer and some for use with special-purpose hardware where no computer is available.

CHAPTER 4

NOISE ANALYSIS OF ESTIMATION METHODS OF CHAPTER 3

4.1 Introduction

The methods of identification of plants using pseudo-random binary sequences postulated in Chapter 3 are all based on the estimate of the gain of the plant given by equation (3.1), and are equivalent to one another from the point of view of theoretical analysis. We shall consider now the effect that noise will have on the performance of these systems.

The results of this analysis form an extension to that of Chapter 2; this time the effect of noise on the estimate of d.c. will be taken into account, in addition to its effect on the "signal" component of the gain estimate. The theoretical technique is, however, somewhat different to that of Chapter 2, as explained in section 4.2, where a general formula for the variance due to noise is derived. Section 4.3 illustrates the application of this formula to two particular types of noise. The choice of parameter values for the gain-estimation process is considered in Section 4.4, where it is shown that signal (as well as noise) is dependent on parameter values. The shortcomings of a scheme for averaging step response estimates are discussed in Section 4.5. The conclusions are presented in Section 4.6.

4.2 Derivation of variance formula in the general case

We shall consider the general case where noise is added at the output of the plant, as in Chapter 2, and calculate the variance in the gain-estimate due to this noise.

The technique of considering the auto-correlation function of the output of the multiplier, used in Section 2.4, becomes very combersome when applied to the system of figure 3.2. Appendix A4.1 shows how the auto-correlation function of the compound running summer can be derived, and it can be seen that the form of this function is not amenable to analysis.

An alternative method will therefore be considered, starting from the basic concepts of equation (3.1). Noise, $\nu(t)$, is introduced at the output of the plant, as shown in Figure 4.1.

Then $\hat{y}(t) = y(t) + \nu(t)$

Now since $\phi_{c\hat{y}}(\tau) = \frac{1}{T}\int_o^T c(t-\tau)\,\hat{y}(t)\,dt$,

the noise component of $\phi_{c\hat{y}}(\tau)$ is

$$\phi_{c\nu}(\tau) = \frac{1}{T}\int_o^T c(t-\tau)\,\nu(t)\,dt \qquad \ldots\ldots(4.1)$$

so that the noise component of the gain estimate is, from equation (3.1),

$$\hat{S}_\nu(T_r) = \frac{T}{T+\lambda}\,\frac{1}{a\lambda}\int_o^{T_r}\left[\phi_{c\nu}(\tau) - \frac{1}{\Delta}\int_{T-\Delta}^{T}\phi_{c\nu}(s)\,ds\right]d\tau \qquad \ldots\ldots(4.2)$$

Now the variance of the gain estimate due to the noise is given by

$$D^2 = E\left[\hat{S}_\nu^2(T_r)\right] \qquad \ldots\ldots(4.3)$$

where E denotes the expected value. It will be assumed that the noise has zero mean, so that $E\left[\hat{S}_\nu(T_r)\right] = 0$, as any mean value would be taken account of in the d.c.-elimination process.

Equation (4.3) is expanded in Appendix A4.2, to give, on rearranging,

$$D^2 = \frac{1}{(T+\lambda)a^2\lambda^2}\left[\int_o^{T_r}\int_o^{T_r} f_\nu\,d\tau_1 d\tau_2 + \frac{T_r^2}{\Delta^2}\int_{T-\Delta}^{T}\int_{T-\Delta}^{T} f_\nu\,d\tau_1 d\tau_2 - \frac{2T_r}{\Delta}\int_{\tau_2=T-\Delta}^{T}\int_{\tau_1=0}^{T_r} f_\nu\,d\tau_1 d\tau_2\right] \qquad \ldots\ldots(4.4)$$

where

$$f_\nu = \int_o^T\int_o^T \phi_{\nu\nu}(t_1 - t_2)\;\phi_{cc}(t_1 - \tau_1 - t_2 + \tau_2)\,dt_1 dt_2 \qquad \ldots(4.5)$$

4.3 Evaluation of Variance for Particular Cases

4.3(a) Band-Limited Noise

D^2 will now be evaluated for the case where the noise $\nu(t)$ is considered as filtered white noise, where the filter is a single lag of time constant T_ν. (This type of noise was discussed in Section 2.4(b)).

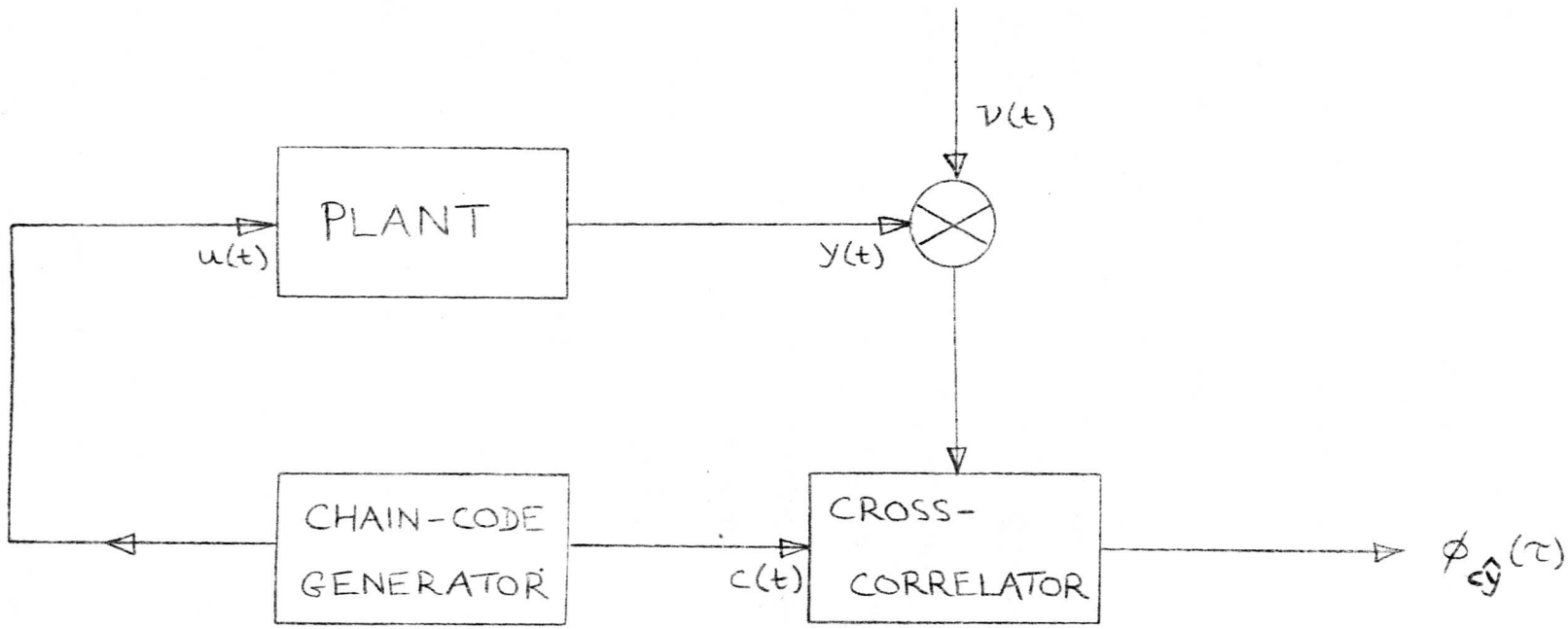


FIGURE 4.1: CROSS-CORRELATION IN PRESENCE OF NOISE

The auto-correlation function of the noise is

$$\phi_{\nu\nu}(\tau) = \sigma_\nu^2 \varepsilon^{-\frac{|\tau|}{T_\nu}} \quad \text{.....(4.6)}$$

where σ_ν^2 is the power of the noise.

The chain-code auto-correlation function will be approximated to an impulse. It can be shown that d.c. bias on the code has no effect on the variance, so we shall take

$$\phi_{cc}(\tau) = \frac{N+1}{N} \lambda \delta(\tau) \quad \text{.....(4.7)}$$

Substituting in equation (4.5) gives

$$f_\nu = \sigma_\nu^2 \frac{N+1}{N} \lambda \int_0^T \int_0^T \varepsilon^{-\frac{|t_1-t_2|}{T_\nu}} \delta(t_1-\tau_1-t_2+\tau_2)\, dt_1 dt_2 \quad \text{.....(4.8)}$$

which becomes, with the aid of Appendix A4.3,

$$f_\nu = \sigma_\nu^2 \frac{N+1}{N} (T - |\tau_1 - \tau_2|) \varepsilon^{-\frac{|\tau_1-\tau_2|}{T_\nu}} \quad \text{.....(4.9)}$$

Substituting in equation (4.4) and using Appendix A4.4 gives

$$\begin{aligned} D^2 = 2\frac{\sigma_\nu^2}{a^2} \frac{T_\nu}{(T+\lambda)T\lambda} \Bigg\{ & TT_r - T_\nu T_r - TT_\nu + 2T_\nu^2 + (TT_\nu - 2T_\nu^2 - T_\nu T_r)\varepsilon^{-\frac{T_r}{T_\nu}} \\ & + \frac{T_r^2}{\Delta^2}\left[T\Delta - T_\nu \Delta - TT_\nu + 2T_\nu^2 + (TT_\nu - \Delta T_\nu - 2T_\nu^2)\varepsilon^{-\frac{\Delta}{T_\nu}} \right] \\ & - \frac{T_r}{\Delta}\Bigg[(2T - 2T_\nu) \,\text{ff}\, (T_r - T + \Delta) - 2T_\nu^2 \varepsilon^{-\frac{T}{T_\nu}} + (2T_\nu^2 - \Delta T_\nu)\varepsilon^{\frac{\Delta - T}{T_\nu}} \\ & + (2T_\nu^2 - T_\nu T_r)\varepsilon^{\frac{T_r - T}{T_\nu}} + T_\nu (T - 2T_\nu - |T_r - T + \Delta|)\varepsilon^{-\frac{|T_r - T + \Delta|}{T_\nu}} \Bigg] \Bigg\} \quad \text{.....(4.10)} \end{aligned}$$

where $\text{ff}\,(\text{function}) = \begin{cases} \text{function} & \text{if function} \geq 0 \\ 0 & \text{if function} \leq 0 \end{cases}$

Equation (4.10) gives the variance of step response estimate due to band-limited noise. It is proportional to the ratio of the power of the noise to the power of the chain-code.

Note that D^2 is proportional to N, if T, T_r and T_ν are fixed. Hence the signal-to-noise (amplitude) ratio decreases as N increases. Thus the smallest value for N must be chosen, consistent with:

(i) defining the noise-free step response adequately, and

(ii) keeping the impulse-shape error small.

A scaled form of D^2 is plotted in Figure 4.2, for N = 31, for various values of the parameters. Note that D^2 decreases as Δ increases, for a given T_ν, but increases over most of the range as T_r increases, for a given Δ. It also increases as the band-width of the noise increases. To evaluate the value of Δ and p to give the best signal-to-noise ratio, it is necessary to consider Figure 4.2 in conjunction with a graph of the noise-free step response estimate. To facilitate the choice of optimum values for the parameters, the variance will now be evaluated for a simpler case, that of white noise at the output of the plant. This results in a less complex equation, which yields further insight into the problem.

4.3(b) White Noise

The case where the noise added to the output of the plant is white will now be considered. This situation is not as realistic as the case considered in the last section, but the resulting equation can be analysed more easily than equation (4.10), and will yield further information on how to choose values for Δ and T_r.

White noise has an auto-correlation function given by

$$\phi_{\nu\nu}(\tau) = \overline{\phi}\ \delta(\tau) \qquad \text{.....(4.11)}$$

where $\overline{\phi}$ is the power/unit bandwidth.

Reiterating the mathematics from equation (4.4) gives

$$D^2 = \frac{\overline{\phi}}{a^2\lambda}\,\frac{T_r}{T+\lambda}\left[1 + \frac{T_r}{\Delta} - 2\,\delta\left(\frac{T_r-T+\Delta}{\Delta}\right)\right] \qquad \text{.....(4.12)}$$

(Clarke[53] has derived a similar formula for the case where $T_r = T-\Delta$, using matrix least-squares analysis based on the (N-k) equations available assuming that the last k ordinates of $\phi_{xy}(n)$ are zero).

The following observations can be made:

(i) D^2 is proportional to $\overline{\phi}$ and inversely proportional to a^2.

(ii) D^2 is proportional to N if the other parameters are fixed, so that a small value should be chosen for N to maximize the signal-to-noise ratio.

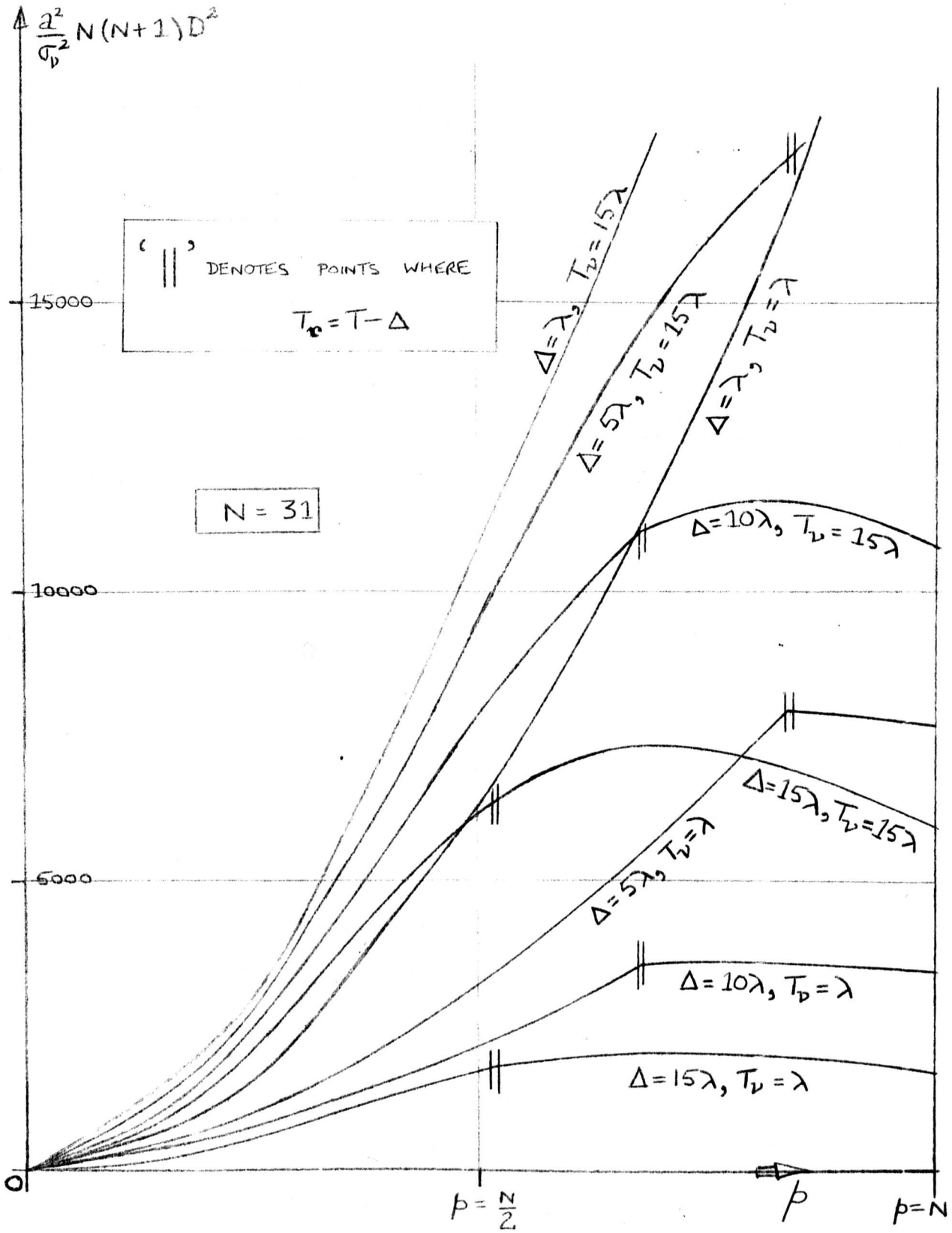


FIGURE 4.2:

VARIANCE FOR BAND-LIMITED NOISE

(iii) If all parameters but Δ are fixed, then

$$\left.\begin{aligned} D^2(\Delta) &= \gamma^2 T_r \left(\frac{\Delta + T_r}{\Delta}\right) \quad \text{for } \Delta \leq T - T_r \\ &= \gamma^2 T_r \left(\frac{2T - [\Delta + T_r]}{\Delta}\right) \text{for } \Delta \geq T - T_r \end{aligned}\right\} \quad \ldots\ldots\ldots\ldots (4.13)$$

where γ^2 is a constant under these conditions. The dependence of D^2 on Δ is shown in Figure 4.3(a), from which it can be seen that

(a) The variance increases as Δ decreases

(b) As $\Delta \rightarrow 0$, $D^2 \rightarrow \infty$

(c) The slope of the curve has a discontinuity at $\Delta = T - T_r$.

(iv) If all parameters but T_r are fixed, D^2 is given by equations (4.13), and is illustrated in Figure 4.3(b). It can be seen that

(a) The variance increases as T_r increases until $T_r = (T - \Delta/2)$ and then decreases. The maximum value is

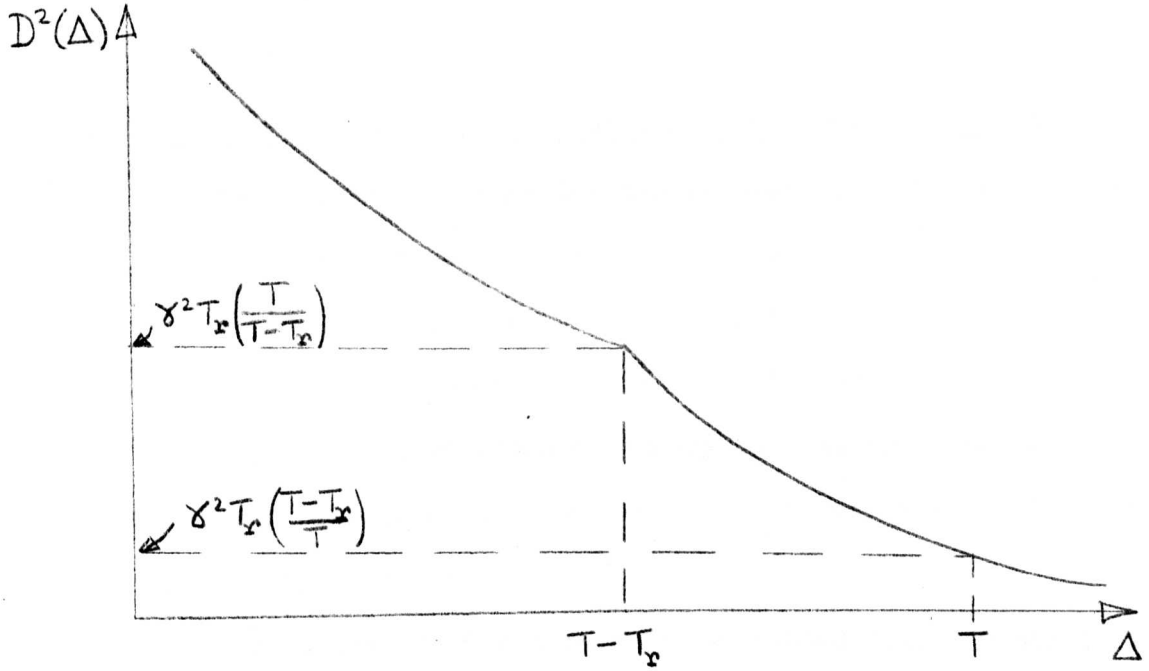
$$\frac{\gamma^2}{\Delta} \left(T - \frac{\Delta}{2}\right)^2$$

(b) D^2 has the same value at $T_r = (T - \Delta)$ as at $T_r = T$. This value is

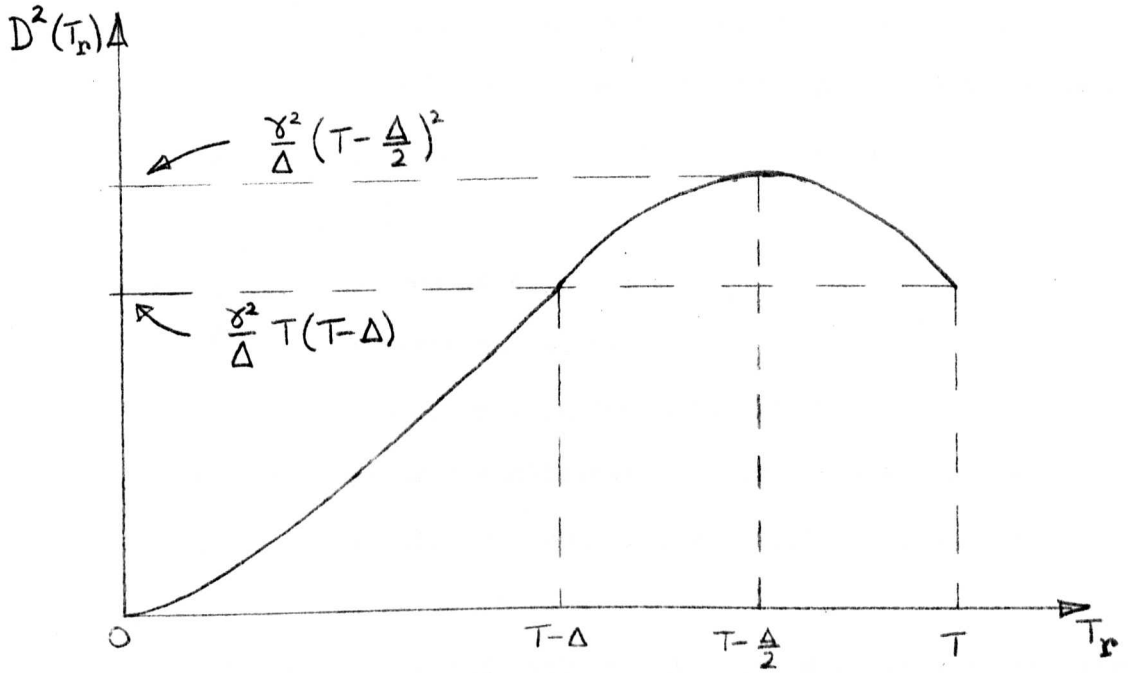
$$\gamma^2 \frac{T}{\Delta} (T - \Delta)$$

(c) The slope of the curve has a discontinuity at $T_r = T - \Delta$.

It has been shown how the variance of gain estimate depends on the parameters of the identification system. It will be noted that, for the cases considered, the variance is independent of the characteristics of the plant. It is now necessary to consider the noise-free estimate of gain, as this also depends on the parameters of the identification system.



(a) DEPENDENCE ON Δ



NOT TO SCALE

(b) DEPENDENCE ON T_r

FIGURE 4.3 : VARIANCE OF GAIN ESTIMATE DUE TO WHITE NOISE AT PLANT OUTPUT

4.4 Choice of parameter values for the identification system

The choice of values for the parameters of the gain estimation process, to achieve the "best" estimate in the presence of noise, will now be considered. The mean of the gain estimate, as well as its variance, has to be taken into account, in order to assess the optimal parameter values. It is then found necessary to compromise between a good estimate of mean, with large variance, and an estimate with small variance, but mean far from the true gain.

The nature of this problem is discussed with the aid of two criteria

(i) the range of probable estimates,

(ii) the signal-to-noise ratio.

4.4(a) The noise-free estimate of gain

The estimate of gain for a (theoretical) noise-free plant depends on the characteristics of the plant and on the parameters of the estimating process. The difference between the true gain and the estimated gain will be defined by the error term, g_e, given by

$$g_e = g - \hat{g}$$

where g = the true gain

and $\hat{g}$ = the estimated gain.

There are three principle components of g_e. (The impulse-shape error will be assumed negligible). These are all caused by the non-zero nature of the impulse response after a finite time. The components of g_e will be considered in turn:

(i) Due to the periodicity of the code, $\hat{\phi}(n)$ will be the sum of $h(n)$ and delayed versions of $h(n)$. The corresponding error in gain estimate for $p = p_1$ is

$$g_{e1} = S_c(p_1)$$

(ii) The d.c. bias estimation process is equivalent to reducing a point on the estimated ~~step~~ response $\hat{\phi}(n)$ to zero. This point is given by $n = n_o$ where

$$\hat{\phi}(n_o) = \frac{1}{k} \sum_{N-k}^{N-1} \hat{\phi}(n)\, dn \qquad \ldots\ldots(4.14)$$

This is illustrated in Figure 4.4(a). The true d.c. bias is given by $h(\infty)$. The error in d.c. estimate is therefore

$$d_e = \hat{\phi}(n_o) - h(\infty) \qquad \ldots\ldots(4.15)$$

which gives an error in gain estimate for $p = p_1$ of

$$g_{e2} = p_1 d_e = p_1 \left[\hat{\phi}(n_o) - h(\infty)\right] \qquad \ldots\ldots(4.16)$$

(iii) The gain estimate without the periodicity and bias terms, given by $S(p_1)$ differs from the true gain, given by $S(\infty)$, due to the non-finite settling time of the plant.

The error in gain estimate is

$$g_{e3} = S(\infty) - S(p_1) \qquad \ldots\ldots(4.17)$$

This is illustrated in Figure 4.4(b). Note that the periodicity term tends to cancel out some of the error due to finite integration time, but increases the d.c. error, since $\left|\hat{\phi}(N)\right| > \left|h(N)\right|$ due to periodicity.

The total error in noise-free estimate of gain is

$$\begin{aligned} g_e &= -g_{e1} + g_{e2} + g_{e3} \\ &= -S_c(p_1) + p_1\left[\hat{\phi}(n_o) - h(\infty)\right] + \left[S(\infty) - S(p_1)\right] \end{aligned}$$

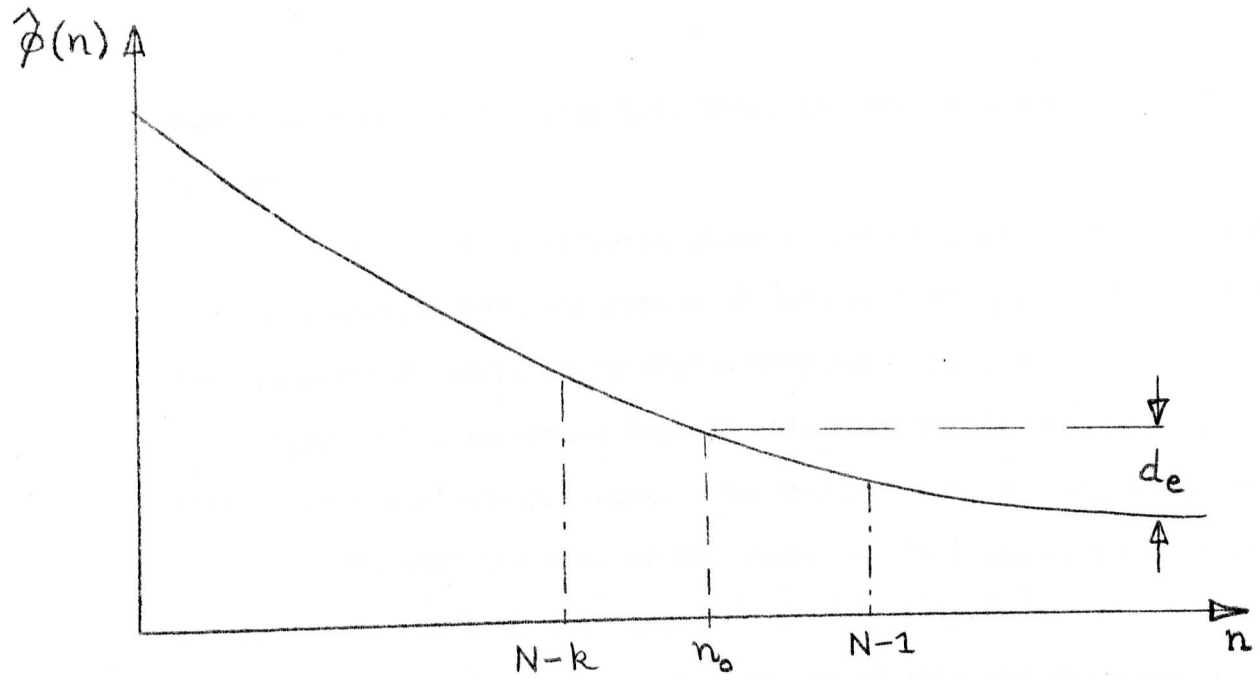
so that

$$g = \hat{g} - S_c(p_1) + p_1\left[\hat{\phi}(n_o) - h(\infty)\right] + \left[S(\infty) - S(p_1)\right] \qquad \ldots\ldots(4.18)$$

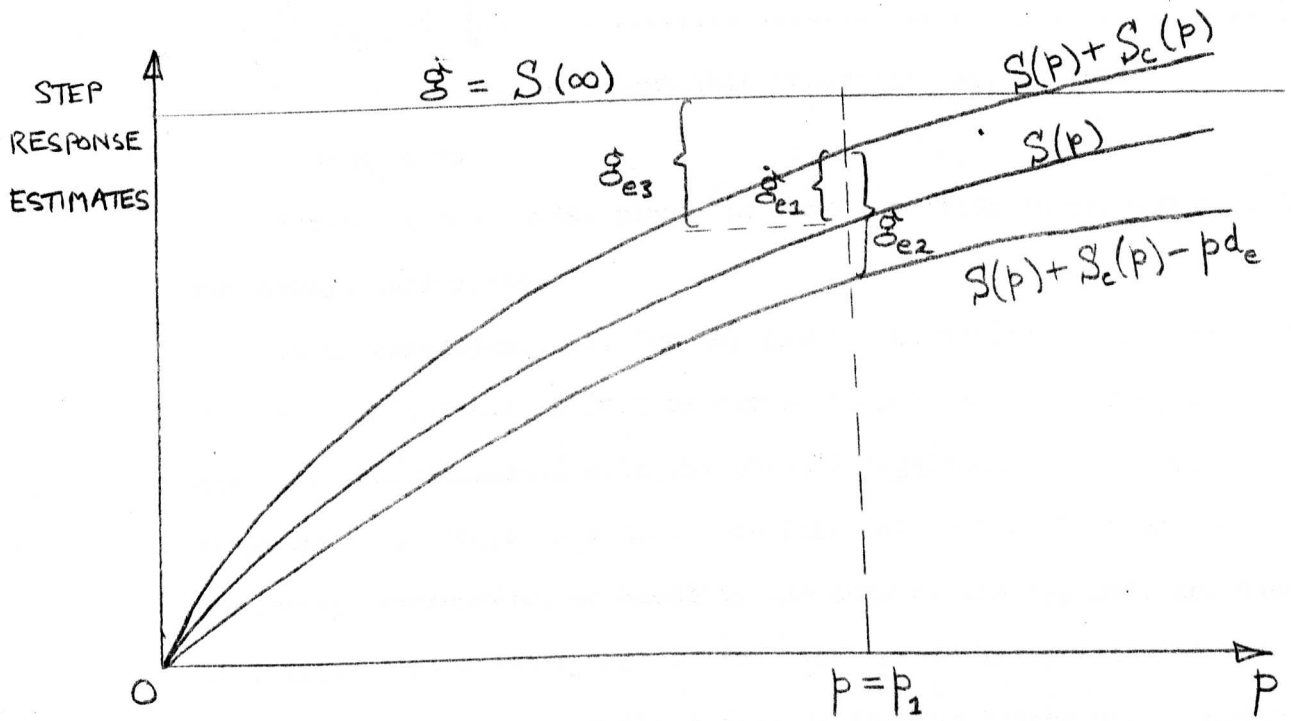
for a plant of any order.

4.4(b) Comparison of estimation methods with and without d.c. removal, under noise-free conditions

The system proposed in Chapter 3, where the d.c. bias is allowed for, will generally give a better mean estimate of gain than will the system of Chapter 2. If, however, the d.c. estimate has an error greater in



(a) D.C. ERROR



(b) TOTAL ERROR

FIGURE 4.4: ERRORS IN NOISE-FREE GAIN ESTIMATE

magnitude than the original d.c. bias, the system of Chapter 2 will be better.

The error in d.c. estimate given by equation (4.15) may be larger than that arising from the system of Chapter 2 if the impulse response has not settled sufficiently when averaging takes place.

Figure 4.5 illustrates typical estimated impulse responses, for first- and second-order plants. We shall assume, for the purposes of this section, that the d.c. at the input to the cross-correlator is due solely to the non-zero d.c. level of the ± 1 chain-code.

For a plant of any order, the system of this and the previous chapter produces less d.c. error than the method of Chapter 2 provided

$$\left| \hat{\phi}(n_o) - \frac{1}{N} \right| < \frac{1}{N} \qquad \text{.....(4.19)}$$

that is, if

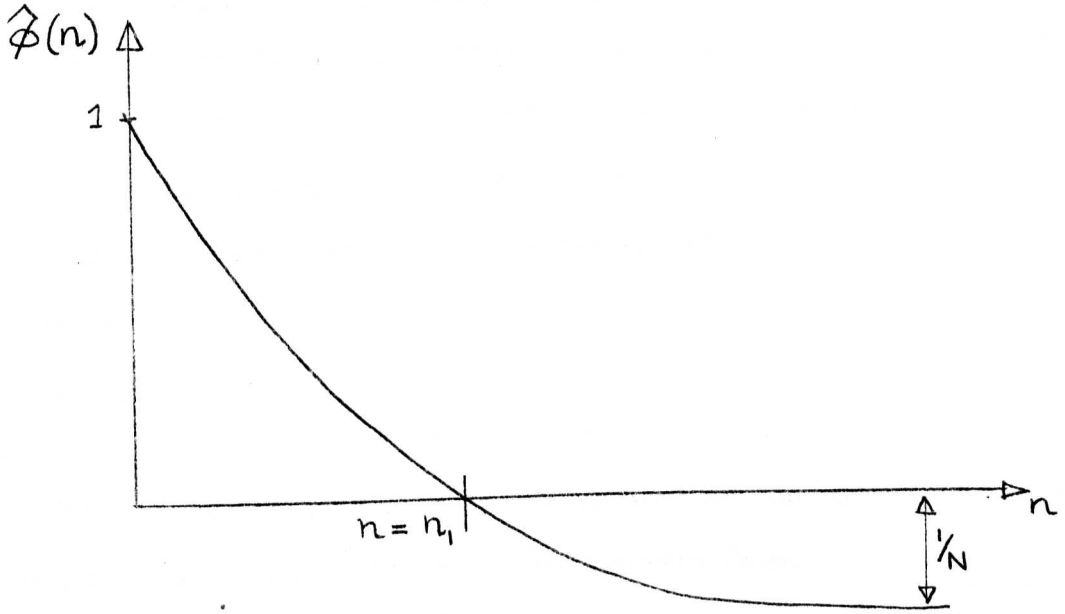
$$\left| \hat{\phi}(n_o) \right| < \frac{2}{N} \qquad \text{.....(4.20)}$$

For the first-order plant this condition implies that

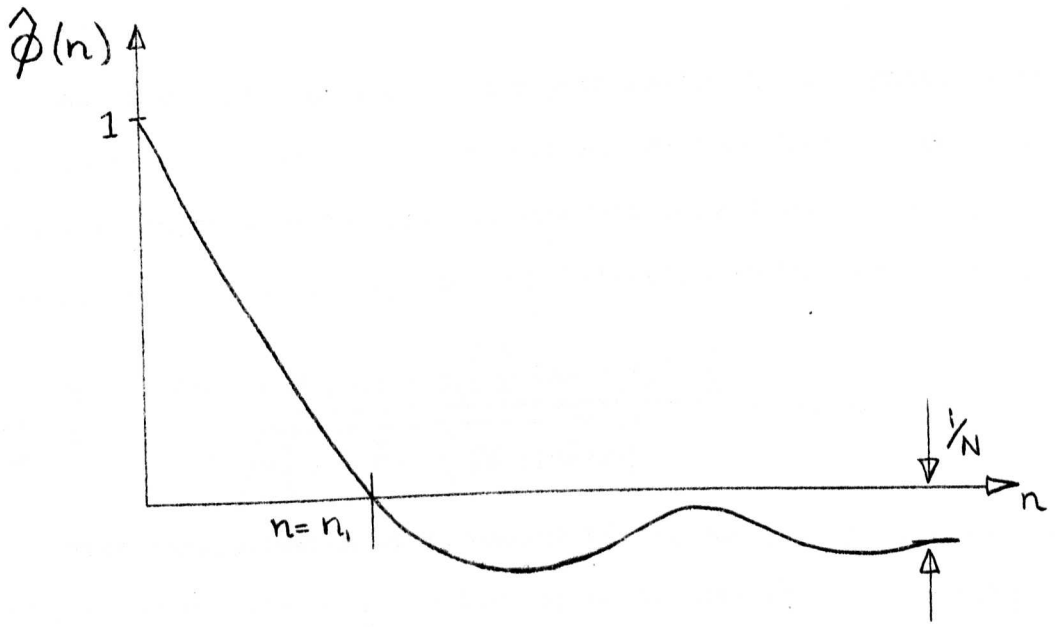
$$n_1 < n_o < \infty$$

For the second-order plant the same condition is necessary, but not always sufficient.

It is concluded that, for any plant, the period of the chain-code, $N\lambda$, and the value of k must be chosen to give a value of n_o such that $h(n_o)$ is small compared with the overall magnitude of the impulse response. If there is a d.c. bias inherent in the plant or the measuring instruments, or added to the code at the 'input', the system of Chapter 3 eliminates most of the bias before integration of the impulse response, whereas the system of Chapter 2 does not. Under such circumstances the d.c.-elimination system is to be preferred.



(a) First-Order



(b) Second-Order

FIGURE 4.5:

ESTIMATED IMPULSE RESPONSES (NORMALIZED)

4.4(c) The estimate of gain under noisy conditions

In any practical plant, noise will be present, and therefore the choice of parameters must be considered under such conditions. It will be shown that, in general, the error term, g_e, tends to increase as the variance decreases. We shall consider the region of probable estimates bounded by

$$\hat{g} \pm D$$

and compare this with the true gain, g.

For the system with white noise present at the output of the plant, equations (4.12) and (4.18) give, in discrete form,

$$\hat{g} \pm D = S(p_1) + S_c(p_1) - p_1\left[\hat{\phi}(n) - h(\infty)\right] \pm \gamma\sqrt{p\left[1 + \frac{p}{k} - \frac{2}{k}\hat{\phi}\hat{\phi}(p-N+k)\right]} \quad \ldots\ldots\ldots(4.21)$$

whereas $g = S(\infty)$

This formula enables the bounds to be calculated for a plant of any order.

Another useful measure of the performance of the system is the signal-to-noise (amplitude) ratio of the gain estimate. This is given by the ratio of expected gain to standard deviation. From equations (2.20), (4.12) and (4.18), the signal-to-noise ratio can be written as

$$\eta_D = \frac{\hat{g}}{D} = \frac{S(p_1) + S_c(p_1) - p_1\left[\hat{\phi}(n) - h(\infty)\right]}{\gamma\sqrt{p\left[1 + \frac{p}{k} - \frac{2}{k}\hat{\phi}\hat{\phi}(p-N+k)\right]}} \quad \ldots\ldots\ldots\ldots\ldots\ldots(4.22)$$

From consideration of equations (4.21) and (4.22), the optimal values of p, k and N can be found for any particular plant. In addition to these, a value must be found for N, bearing in mind that, since $\hat{S}(p)$ has been scaled to be independant of N, and D^2 is proportional to N, η_D is inversely proportional to $\sqrt{N}$. A value of 15 for N would be reasonable. Below this, the impulse-shape error would be significant.

4.4(d) Analysis of a particular plant

The signal and noise components of the gain estimate for a particular plant will now be considered. The plant has a first-order transfer-function, with time constant T_s and gain unity. The period of the code is $T = 2T_s$, and N is taken as 31 for the purposes of quantitative analysis. The noise at the output of the plant is assumed to be white.

Figure 4.6 shows the bounds of probable gain estimates for the plant, calculated from equation (4.21), together with the true step response, for values of p from 0 to 30 and values of k of 2, 10 and 20. (In practice, several samples of the output of the plant will be averaged for each bit of the chain-code, and the estimated step response will not be exactly as shown, but the discrepancy is small. Also, as the impulse response is, in practice, a discrete function of n, for values of n from 0 to (N - 1), the step response is, strictly, a discrete function, for values of p from 0 to (N - 1), and not a continuous function. The variance formulae given in equations (4.10) and (4.12) are calculated for continuous functions, but again, the discrepancy is small).

From Figure 4.6 it can be seen that

(i) The expected gain is considerably less than the true gain, when assessing the d.c. by averaging over even a small number of impulse-response ordinates, due to the non-zero value of h(N-1). This error can only be reduced by increasing T.

(ii) For any given value of p, as k increases the expected value decreases slowly whereas the standard deviation decreases rapidly (see equation (4.12)). It is therefore advisable to choose a fairly large value for k, as the inaccuracy in $\hat{g}$ will be counteracted by the smaller standard deviation.

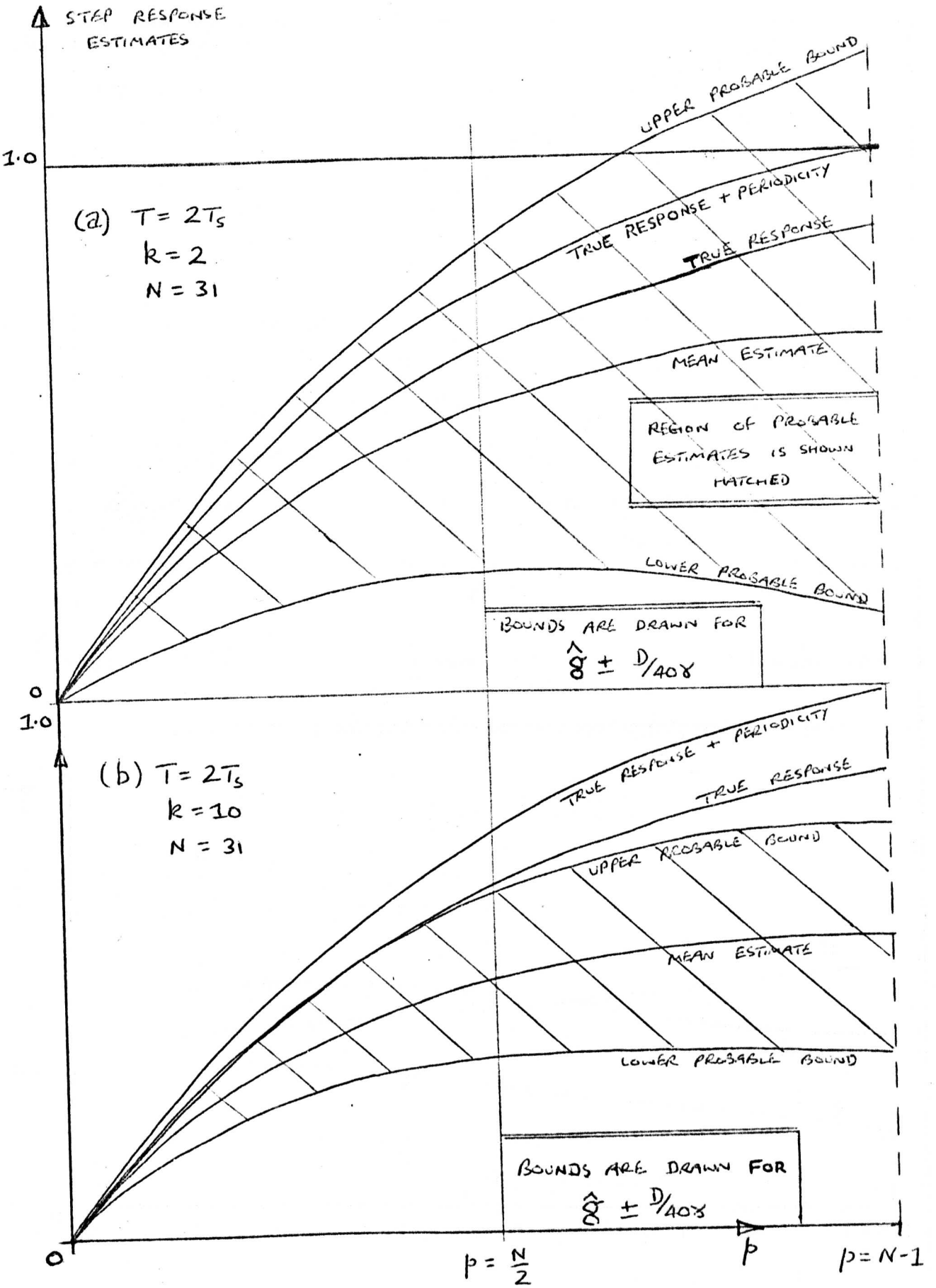


FIGURE 4.6:

SIGNAL AND NOISE FOR FIRST-ORDER PLANT

(CONTINUED OVERLEAF →)

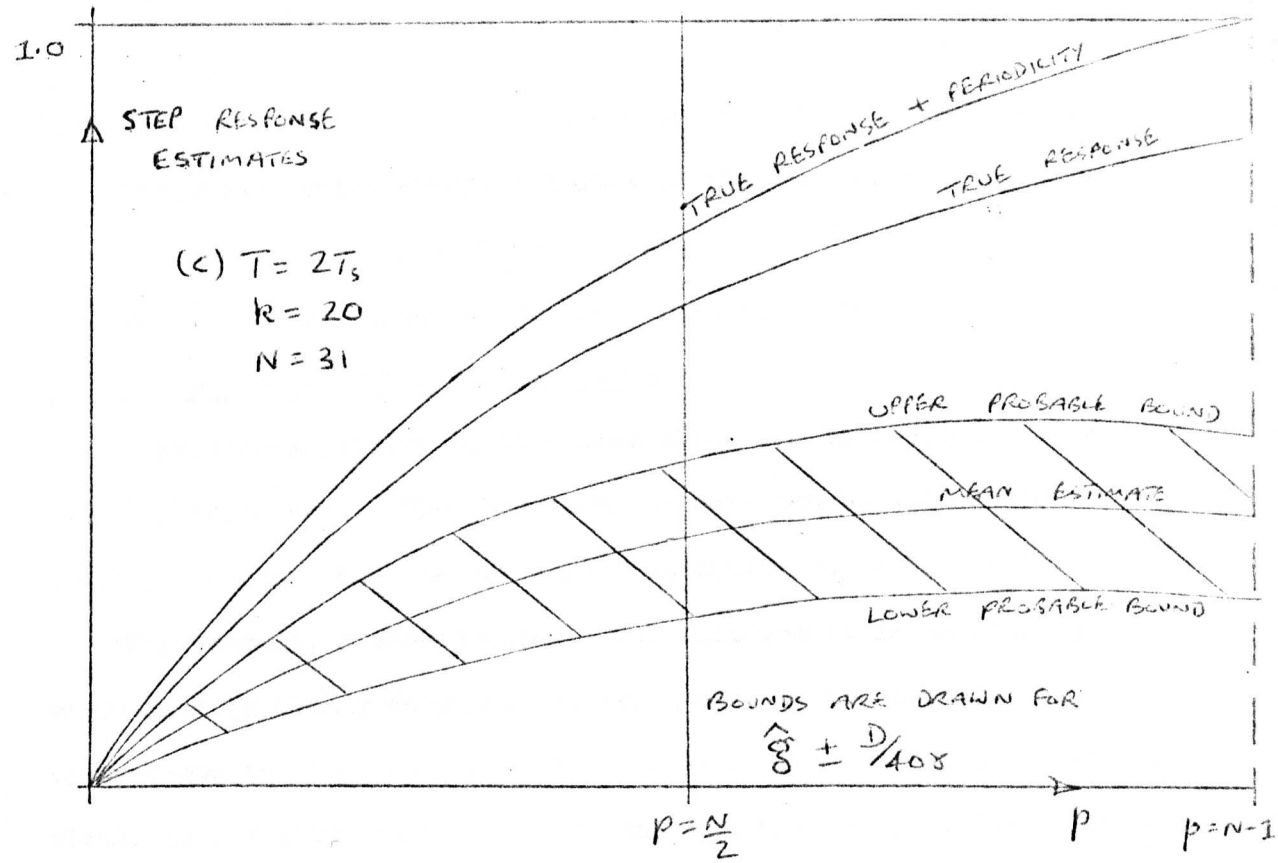


FIGURE 4.6 : SIGNAL AND NOISE (CONTINUED)

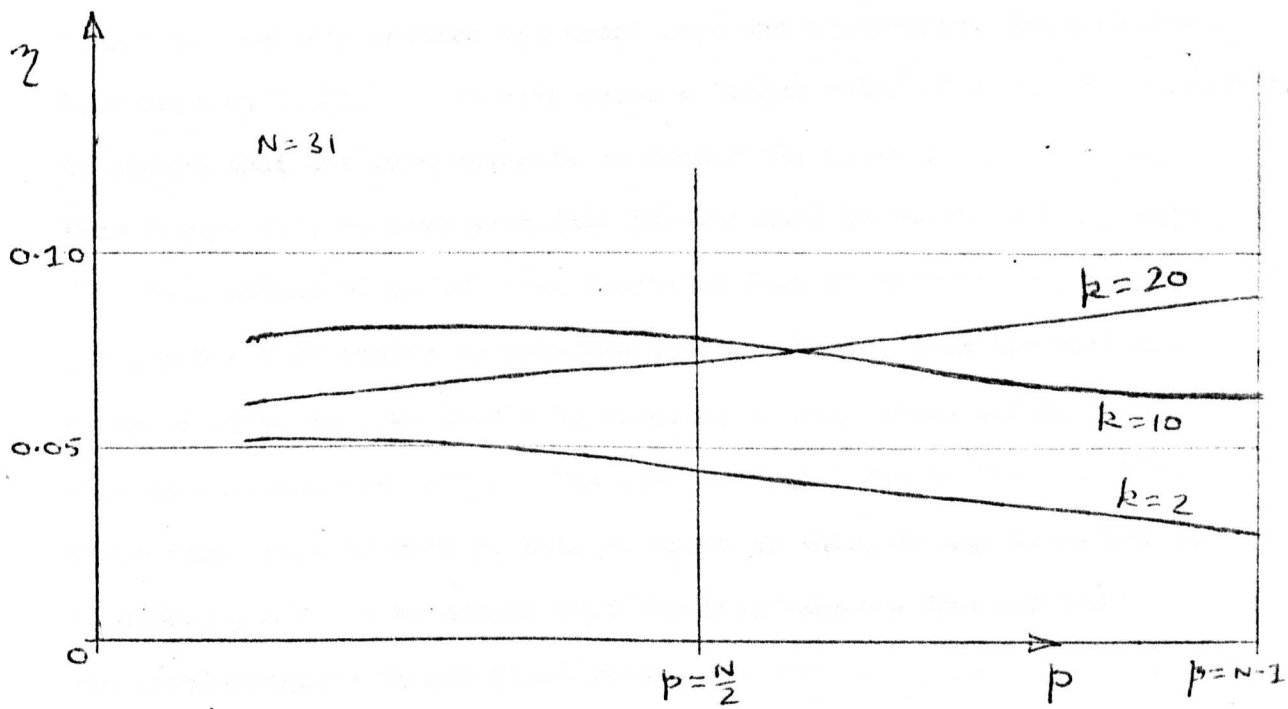


FIGURE 4.7:

SIGNAL-TO-NOISE RATIO FOR FIRST-ORDER PLANT

(iii) For any given value of k, as p increases the expected value increases butthe standard deviation also increases.

Figure 4.7 shows how η_D varies with p and k. From this we observe that large values should be chosen for both p and k.

4.4(e) Analytic procedure for a general plant

A procedure similar to that used above can be applied to a known plant of any order. The mean gain estimate can be deduced from the impulse response, and the variance from equations (4.10) or (4.12).

For example, a second-order plant will result in an estimate whose mean is highly dependent on which part of the impulse response is averaged to find the d.c. The analysis of second- and higher-order plants is not given here, owing to the diversity of possible responses; each plant must be taken on its own merits.

We have seen how a relatively small value of p is optimal for certain cases. In general, however, the plant dynamics will be of high order and may involve transport lags and non-minimum phase elements (See Section 2.3(d)). In such cases a larger value of p will be necessary, to ensure that the gain estimate is reasonably close to the true gain. From Figure 4.3, we have seen that the variance increases rapidly with p for small values of p, but then increases less rapidly and, finally, for $p > N - k/2$, starts to decrease again. Hence, from the variance point of view, we lose little by choosing a large value for p, rather than an intermediate value. The loss in signal due to the d.c. bias error does increase with p, but, in spite of this, it may be advisable to choose $p = N - 1$ to ensure that the step response has settled reasonably closely to its final value.

The design procedure for an identification system is therefore

1. Carry out a preliminary investigation to find out how the system is likely to behave, in terms of the settling-time and the probable magnitude of oscillations.

2. Choose the product $N\lambda$ so that the plant has settled a reasonable amount in one period, in order to give a small bias error and a good estimate of the final value of step response (i.e. to make g_{e1} and g_{e2} both small).
3. Choose a large enough value for p to overcome any pecularities that might occur in the initial part of the step response. As the choices of p and N are based on similar considerations, it will generally be found that p should have a value close to N. In particular cases, however, where the step response is of the same sign at small values of p as at $p \to \infty$, a small value of p is allowable.
4. Choose k so that the best signal-to-noise ratio is obtained for the value of p chosen.

The very nature of the identification process implies that we do not have precise knowledge of the plant dynamics. The parameters of the identification system must therefore be chosen to allow for wide variations in the dynamics, but it has been shown in this section how we may make some attempt at choosing the optimal values for these parameters.

4.5 The Shortcomings of Averaging Step Responses

The previous sections have shown how to choose the value for T_r that gives the highest signal-to-noise ratio. It might be thought that an estimate of gain derived from the weighted average of estimates of the step response obtained at several values of T_r, at the same instant in time, would give a higher signal-to-noise ratio. It will now be shown that there is usually no advantage to be gained by using an average of estimates instead of the best single estimate.

Consider first an estimate, $\hat{S}$, given by the weighted average of two estimates, made at $T_r = T_1$, $T_r = T_2$, with $T_2 > T_1$.

Then $\hat{S} = \frac{1}{1+\rho_w}\left[\hat{S}(T_1) + \rho_w \hat{S}(T_2)\right]$ .....(4.23)

where ρ_w = a weighting factor. Denoting error terms due to noise by the suffix ν, the error in $\hat{S}$ is

$$S_\nu = \frac{1}{1+\rho_w}\left[S_\nu(T_1) + \rho_w S_\nu(T_2)\right]$$

The variance of $\hat{S}$ is then

$$D_S^2 = E\left[S_\nu^2\right] = \frac{1}{1+\rho_w^2+2\rho_w}\left(D_1^2 + \rho_w^2 D_2^2 + 2\rho_w C\right) \quad \text{.....(4.24)}$$

where D_1^2 and D_2^2 are the variances of $\hat{S}(T_1)$ and $\hat{S}(T_2)$ and C is their covariance.

Assuming the noise is white, it can be shown, by analysis similar to that of Sections 4.2 and 4.3, that

$$C = \mu_c\left[\Delta(T_1 + T_2) - |T_1 - T_2|) + T_1T_2 - T_1\phi\phi(T_2 - T + \Delta) - T_2\phi\phi(T_1 - T + \Delta)\right]$$

where μ_c is a scale factor that is independent of T_1 and T_2.

Consider now the values of C and D_1^2 for the possible ranges of T_1 and T_2:-

(i) For $T_1 \leq T - \Delta$ and $T_2 \leq T - \Delta$:

$$C = \mu_c(\Delta T_1 + T_1T_2)$$

and $D_1^2 = \mu_c(\Delta T_1 + T_1^2)$

Hence $C - D_1^2 = \mu_c T_1(T_2 - T_1)$

so that $C > D_1^2$

In this range $D_2^2 > D_1^2$, so that equation (4.24) gives

$D_S^2 > D_1^2$

(ii) For $T_1 \leq T - \Delta$ and $T_2 \geq T - \Delta$:

$C = \mu_c TT_1$

Hence $C - D_1^2 = \mu_c T_1(T - \Delta - T_1)$

so that $C > D_1^2$

Therefore, as for (i), $D_S^2 > D_1^2$

(iii) For $T_1 \geq T-\Delta$ and $T_2 \geq T-\Delta$:

In this range it is possible for C to be less than D_1^2 and D_2^2. But if D^2 is the variance for $T_r = T$ then

$$C-D^2 = (T-T_2)(T_1-T+\Delta)$$

so that $$C > D^2$$

In this range $D_1^2 > D^2$ and $D_2^2 > D^2$, so equation (4.24) gives

$$D_{\hat{s}}^2 > D^2.$$

Hence in cases (i) and (ii), a lower variance is obtained by using a single estimate at $T_r = T_1$ than the average of estimates at T_1 and T_2. In case (iii), a single estimate at $T_r = T$ has smaller variance than the average of any two estimates in the range $(T-\Delta, T)$. This discussion can clearly be extended to cover averages over more than two estimates, with any constant weighting factors. If it is assumed that the averages are taken over the settled portion of the step response, it can be concluded that no improvement in signal-to-noise ratio can be expected by taking a weighted average rather than the best single-estimate.

4.6 Conclusions

It has been shown that the variance in the estimate of gain using chain-code perturbation is highly dependent on the parameters of the identification system. The noise-free estimate is also dependent on these parameters, but will usually be close enough to the true estimate for optimization purposes. A design procedure for a general plant has been postulated, and a particular plant analysed. It has been found that no advantage is to be gained from averaging several step response ordinates at the same instant in time.

CHAPTER 5

NOISE ANALYSIS OF A GAIN-ESTIMATION SYSTEM USING SINE-WAVE PERTURBATION

5.1 Introduction

The previous chapter was concerned with the performance of a chain-code perturbation identification system in the presence of noise. An alternative identification system uses sine-wave perturbation[2,6,7]. Expressions will now be derived for the signal and noise components of the gain estimate for the sine-wave system, and the results will be compared with those obtained in the previous chapter for the chain-code system.

5.2 The Noise-free Estimate of Gain

The sine-wave perturbation system is illustrated in Figure 5.1, where the plant is represented, as before, by a transfer function $gH(s)$ where g is the gain and $H(s)$ describes the dynamics. An expression will now be derived for the output ofthe integrator in the noise-free case.

Assuming $h(\tau)$ is constant with respect to time, the output of the plant due to an input $x_1(t)$ is given by the convolution integral as

$$y(t) = \int_0^\infty x_1(t-\tau)\, gh(\tau)\, d\tau \qquad \ldots\ldots(5.1)$$

Now $x_1(t)$ is a sine-wave given by

$$x_1(t) = a_s \sin wt \qquad \ldots\ldots(5.2)$$

so that $y(t) = a_s g \int_0^\infty \sin(wt - w\tau) h(\tau)\, d\tau$

The output of the sample-and-hold is

$$Z(T) = \frac{1}{T_i} \int_0^T y(t) \sin wt\, dt$$

where T is the period of the sine-wave, so that

$$Z(T) = \frac{1}{T_i} a_s g \int_0^T \int_0^\infty \sin(wt - w\tau) \sin wt\, h(\tau)\, d\tau\, dt$$

Interchanging the order of integration,

$$Z(T) = \frac{a_s}{2T_i} g \int_{\tau=0}^\infty h(\tau) \left\{ \int_{t=0}^T \left[\cos(-w\tau) - \cos(2wt - w\tau) \right] dt \right\} d\tau$$

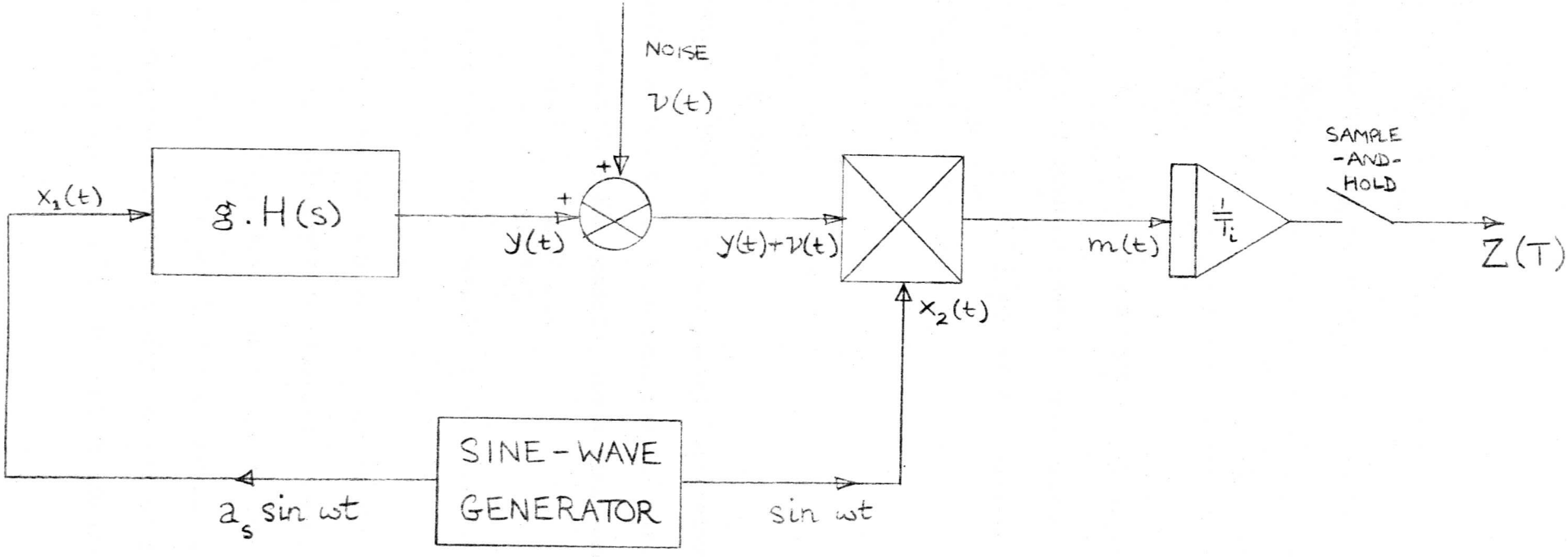


FIGURE 5.1 : GAIN-ESTIMATOR USING SINE-WAVE PERTURBATIONS

Therefore

$$Z(T) = \frac{a_s T}{2T_i} g \int_{\tau=0}^{\infty} h(\tau) \cos w\tau \, d\tau \quad \ldots\ldots(5.3)$$

Hence an estimate of the gain of the plant can be obtained by taking

$$\hat{g}_s = \frac{Z(T)}{\mu_s} \quad \ldots\ldots(5.4)$$

where $\mu_s = \frac{a_s T}{2T_i}$ $\ldots\ldots(5.5)$

so that $\hat{g}_s = g \int_{\tau=0}^{\infty} h(\tau) \cos w\tau \, d\tau \quad \ldots\ldots(5.6)$

Defining the ratio of gain estimate to true gain as F(w), we have

$$F(w) = \frac{\hat{g}_s}{g} = \int_0^{\infty} h(\tau) \cos w\tau \, d\tau \quad \ldots\ldots(5.7)$$

As $w \to 0$, $F(w) \to \int_0^{\infty} h(\tau) \, d\tau \to 1$

so that $\hat{g}_s \to g$

At any non-zero frequency, $F(w) < 1$

so that $\hat{g}_s < g$

To obtain a good estimate of gain, the frequency of the sine-wave must be chosen so that the impulse response has settled almost to zero in well under one period. For example, for a first-order system of time constant T_s, $F(w) = \frac{1}{1+w^2 T_s^2}$

In order to make $F(w) > 0.9$, say, it is necessary to chose T to be greater than about $20T_s$.

It will be seen that any d.c. present at the output of the plant will be eliminated by the multiply-and-integrate process, providing the other multiplicand has zero d.c., which can always be arranged. Some authors recommend the use of a band-pass filter at the output of the plant, to attenuate the frequency components that are not of frequency w, but no conclusive evidence has been given to justify the use of such a filter.

5.3 The Behaviour of the System Under Noisy Conditions

5.3(a) Derivation of Variance Formula in General Case

The variance of the gain estimate will now be evaluated for the general case where noise is present at the output of the plant. The system lends itself to analysis by the auto-correlation function approach of Chapter 2. Assuming the noise, $\nu(t)$, is uncorrelated with the sine-wave and has zero mean, the variance of Z(T) can be deduced, by analogy with equations (2.39) and (2.40), to be

$$D_Z^2 = \frac{2}{T_i^2} \int_o^T (T - \tau)\, \phi_{x_2x_2}(\tau)\phi_{\nu\nu}(\tau)\, d\tau \qquad \ldots\ldots\ldots\ldots\ldots\ldots (5.8)$$

for the sine-wave system,

$$x_2(t) = \sin wt$$

So that the auto-correlation function of $x_2(t)$ is

$$\phi_{x_2x_2}(\tau) = \frac{1}{2}\cos w\tau$$

Substituting in equation (5.8) then gives

$$D_Z^2 = \frac{1}{T_i^2} \int_o^T (T - \tau)\, \phi_{\nu\nu}(\tau) \cos w\tau\, d\tau \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.9)$$

The variance in gain estimate is given by

$$D_s^2 = \frac{D_Z^2}{\mu_s^2} = \frac{4}{a_s^2 T^2} \int_o^T (T - \tau)\phi_{\nu\nu}(\tau) \cos w\tau\, d\tau \qquad \ldots\ldots (5.10)$$

An expression for $\phi_{\nu\nu}(\tau)$ can be deduced from the likely properties of the noise, and substituted in equation (5.10) to find the variance.

The mean and standard deviation of the gain estimate can therefore be derived from equations (5.6) and (5.10) respectively, and the performance of the system thereby assessed.

Defining the signal-to-noise ratio as

$$\eta_s = \frac{\hat{g}_s}{D_s} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5.11)$$

gives

$$\eta_s = \frac{g \int_o^\infty h(\tau) \cos w\tau\, d\tau}{\left(\frac{2}{a_s T}\right)\sqrt{\int_o^T (T-\tau)\phi_{\nu\nu}(\tau) \cos w\tau\, d\tau}} \qquad \ldots\ldots\ldots\ldots\ldots (5.12)$$

5.3(b) Variance of Estimate for Band-limited Noise

The variance of the gain estimate will now be evaluated for the case where the noise, $\nu(t)$, is considered as filtered white noise, where the filter is a single lag of time constant T_ν. This type of noise was discussed in Section 2.4(b).

The auto-correlation function of the filtered noise is given by equation (2.41). Substituting this in equation (5.10) and putting $\alpha = T_\nu/T$ gives

$$D_s^2 = 2R_s.V(\alpha) \quad \ldots\ldots\ldots\ldots\ldots\ldots (5.13)$$

where $R_s = \dfrac{2\sigma_\nu^2}{a_s^2}$ = the ratio of the power of the noise to the power of the sine-wave

and $$V(\alpha) = \alpha\left[\frac{(1+4\pi^2\alpha^2) + \alpha(1-4\pi^2\alpha^2)\ (\varepsilon^{-1/\alpha} - 1)}{(1+4\pi^2\alpha^2)^2}\right] \quad \ldots\ldots\ldots (5.14)$$

It will be observed that:

(i) The variance is proportional to the power ratio, R_s, and

(ii) The variance is independent of the absolute values of T_ν and T, depending only on their ratio.

Figure (5.2) shows the variation of $V(\alpha)$ with α. The variance is a maximum when $T_\nu = 0.2\ T$, and tends to zero either side of this. This corresponds exactly to the situation of Section 2.4(b), with the very high and very low frequency noise components being filtered out.

The signal-to-noise ratio follows directly from equations (5.12) and (5.13), and is

$$\eta_s = \frac{g\ F(\omega)}{\sqrt{2R_s V(\alpha)}} \quad \ldots\ldots\ldots\ldots\ldots (5.15)$$

It has been shown that for minimum variance, T should be chosen to be very small or very large, but for maximum signal T should be as large as possible. The signal-to-noise ratio will therefore be maximised by choosing a large value for T. This however results in slow identification

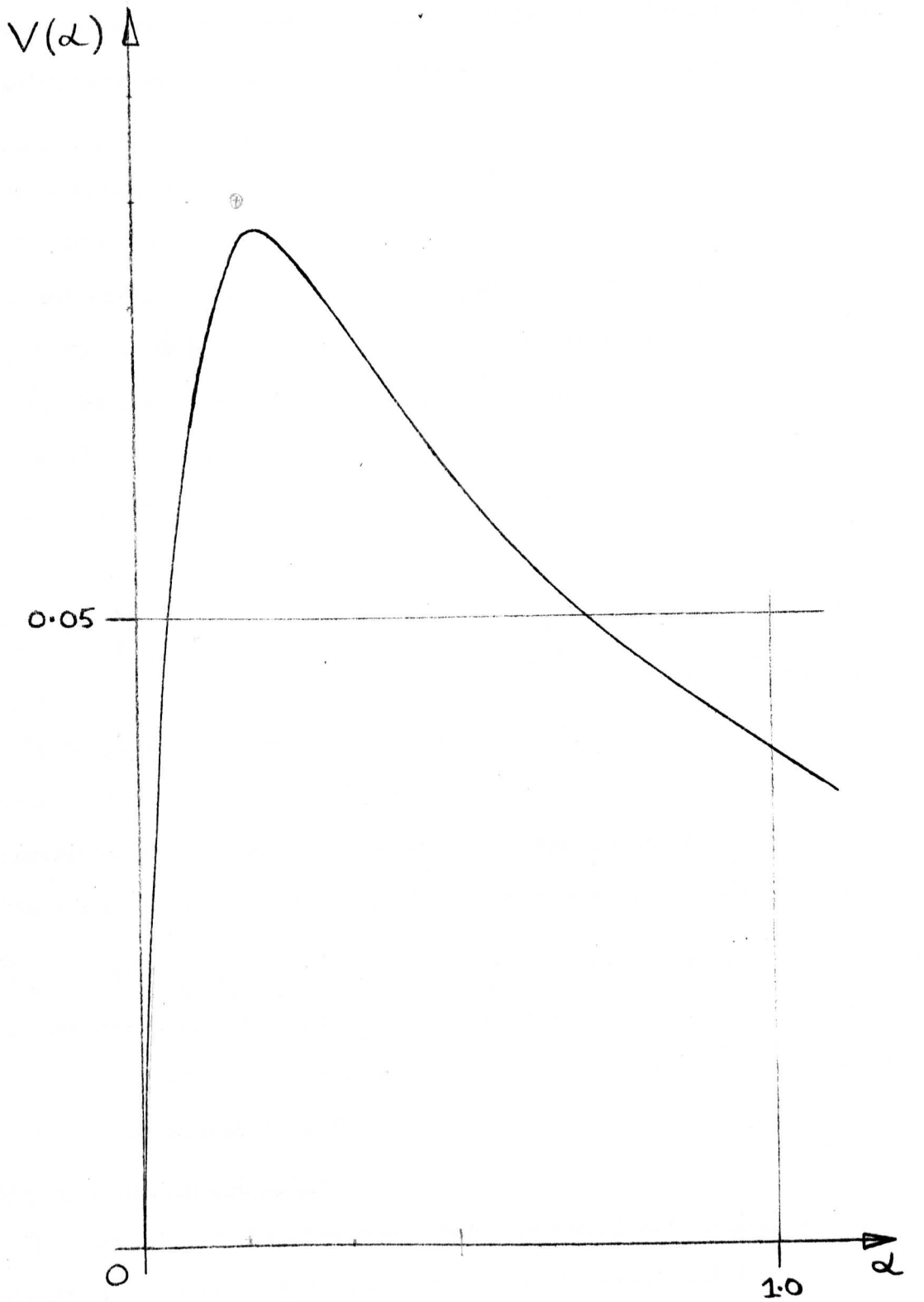


FIGURE 5.2: VARIATION OF $V(\alpha)$ WITH α

and in practice a compromise must be made between fast identification and poor noise rejection, and slow identification and good noise rejection.

5.3(c) Variance of estimate for white noise

The variance of the gain estimate will now be evaluated for the case where the noise is white.

The auto-correlation function of the noise is given by

$$\phi_{\nu\nu}(\tau) = \Phi\,\delta(\tau) \qquad \text{.....(5.16)}$$

where Φ is the power/unit band-width of the noise. Substituting this in equation (5.10), and noting that

$$\int_0^T \delta(\tau)\, f(\tau)\, d\tau = \frac{1}{2} f(0)$$

where $f(\tau)$ is any function of τ, gives

$$D_s^2 = \frac{R_s'}{T} \qquad \text{.....(5.17)}$$

where R_s' = power/unit bandwidth of the noise divided by power of the sine-wave. It will be noted that the variance is

(i) proportional to the power ratio, and

(ii) inversely proportional to the period of the sine-wave.

The signal-to-noise ratio, given by equations (5.12) and (5.17), is

$$\eta_s = gF(w)\sqrt{\frac{T}{R_s'}} \qquad \text{.....(5.18)}$$

which is maximized by choosing a large value for T, as in the case where the noise is filtered. A compromise solution will therefore have to be made in a similar manner to that for the filtered noise.

5.3(d) Analysis of a particular plant

The analysis of the previous Subsections will now be applied to the estimation of gain of the plant considered in Section 4.4(d).

The impulse response of the plant is given by

$$h(\tau) = \frac{1}{T_s}\,\varepsilon^{\frac{-\tau}{T_s}} \qquad \text{.....(5.19)}$$

Equation (5.7) then gives

$$F(w) = \int_0^\infty \frac{1}{T_s} \varepsilon^{-\frac{\tau}{T_s}} \cos w\tau \, d\tau$$

so that $F(w) = \frac{1}{1+T_s^2 w^2}$ .....(5.20)

For band-limited noise, equation (5.15) gives

$$\eta_s = \frac{1}{\left[1 + 4\pi^2\left(\frac{T_s}{T}\right)^2\right]\sqrt{2R_s V\left(\frac{T_\nu}{T}\right)}} \quad \text{.....(5.21)}$$

for white noise, equation (5.18) gives

$$\eta_s = \frac{1}{\left[1 + 4\pi^2\left(\frac{T_s}{T}\right)^2\right]}\sqrt{\frac{T}{R_s'}} \quad \text{.....(5.22)}$$

For the case of the band-limited noise, the signal-to-noise ratio is plotted against T/T_s in figure 5.3., for various values of T_ν/T_s. The range of probable estimates bounded by

$$\hat{g}_s \pm D_s$$

is shown in figure 5.4 for the case where $T_\nu = T_s$. For other values of T/T_s, this range can be deduced from Figures 5.3 and 5.4.

From these curves it will be observed that

(i) The signal component of the gain estimate is highly dependent on T for values of T/T_s less than about 100, but for higher values a small change in T does not appreciably alter the signal.

(ii) The signal-to-noise ratio invariably increases with increasing values of T/T_s.

(iii) η_s increases most rapidly with T/T_s when T_ν/T_s is small.

(iv) If T_ν/T_s is large, it may be necessary to choose a very large value for T/T_s. The best value to choose will depend on the allowable signal-to-noise ratio bearing in mind the expected value of R_s.

The corresponding curves for the white noise case can be deduced from the signal component shown in Figure 5.4 and the noise component given

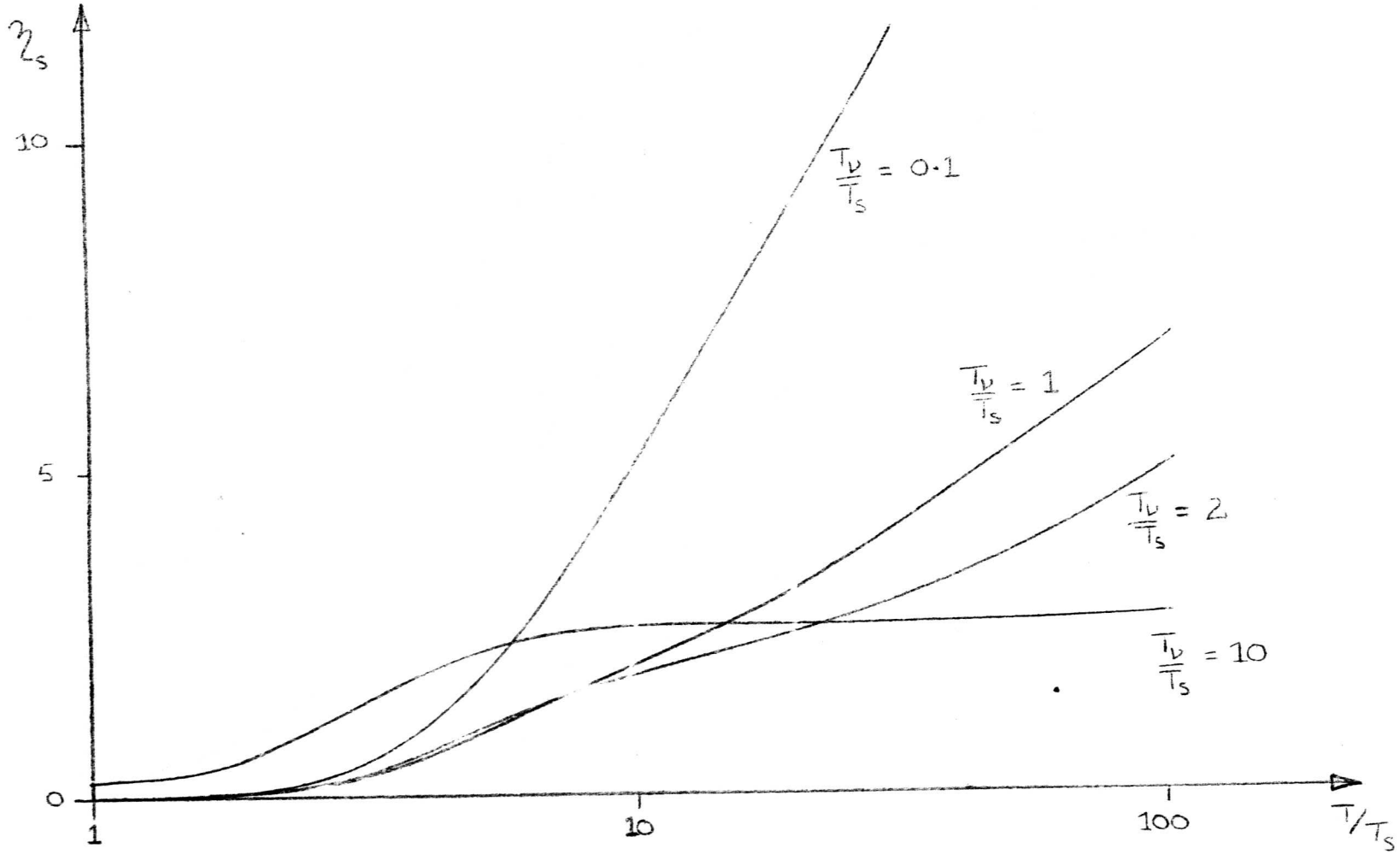


FIGURE 5.3 : SIGNAL-TO-NOISE RATIO FOR BAND-LIMITED NOISE

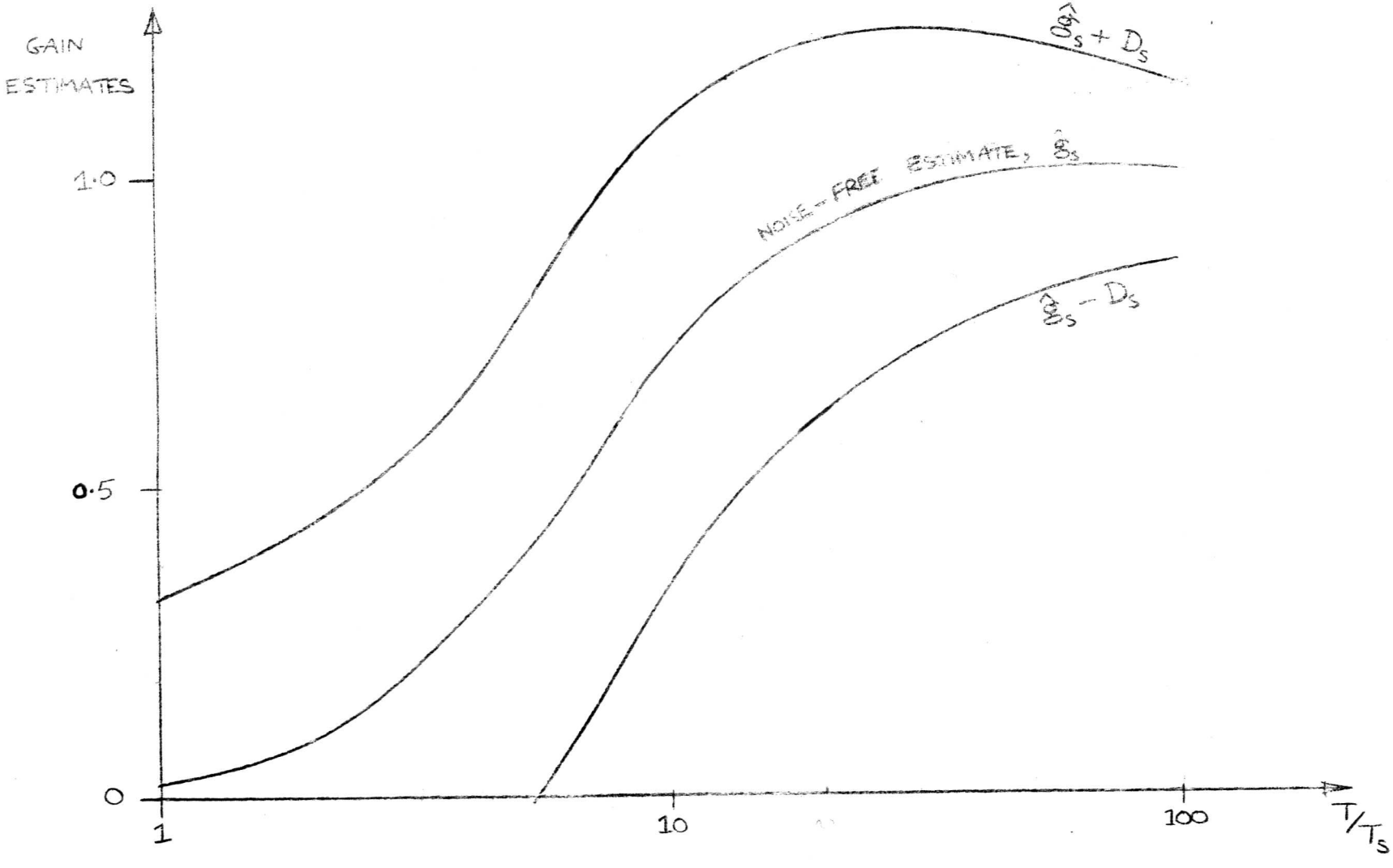


FIGURE 5.4: PROBABLE GAIN ESTIMATES WITH BAND-LIMITED NOISE

by equation (5.17). Clearly the signal-to-noise ratio increases rapidly with T.

Similar results can be obtained for any given plant dynamics, by using equation (5.6) in conjunction with equations (5.13), (5.14) and (5.15) or equations (5.17) and (5.18).

5.4 Comparison with the Chain-Code Perturbation Method

The variance in gain estimate using the sine-wave system will now be compared with the variance using the chain-code system of Chapters 3 and 4. The variance due to white noise at the output of the plant is given by equation (4.12) for the chain-code system, and by equation (5.17) for the sine-wave system. For the same period and power ratio, the ratio of the variances is

$$\eta_V = \frac{D^2}{D_s^2} = \frac{T_r T}{\lambda(T+\lambda)} \left[1 + \frac{T_r}{\Delta} - 2 \left(\frac{T_r - T + \Delta}{\Delta}\right)\right]$$

which reduces to

$$\eta_V \simeq p \left[1 + \frac{p}{k} - 2 \left(\frac{p - N + k}{k}\right)\right] \qquad \text{.....(5.23)}$$

The order of magnitude of this function is illustrated by Figure 5.5, where N = 15 and k = 5. For all usable values of p, $\eta_V > 1$, and for p = N, $\eta_V = 30$. Thus the sine-wave system always gives lower variance than the chain-code system under these conditions. It has been shown earlier, however, that the mean gain estimate for the sine-wave system is significantly in error at low perturbation frequencies. Therefore to obtain a realistic comparison between the two methods, consider the ratio of the signal-to-noise ratios given by

$$\eta_{SN} = \frac{\eta_D}{\eta_s} \qquad \text{.....(5.24)}$$

In terms of the gain estimates,

$$\eta_{SN} = \frac{\hat{g}}{\hat{g}_s} \frac{1}{\sqrt{\eta_V}} \qquad \text{.....(5.25)}$$

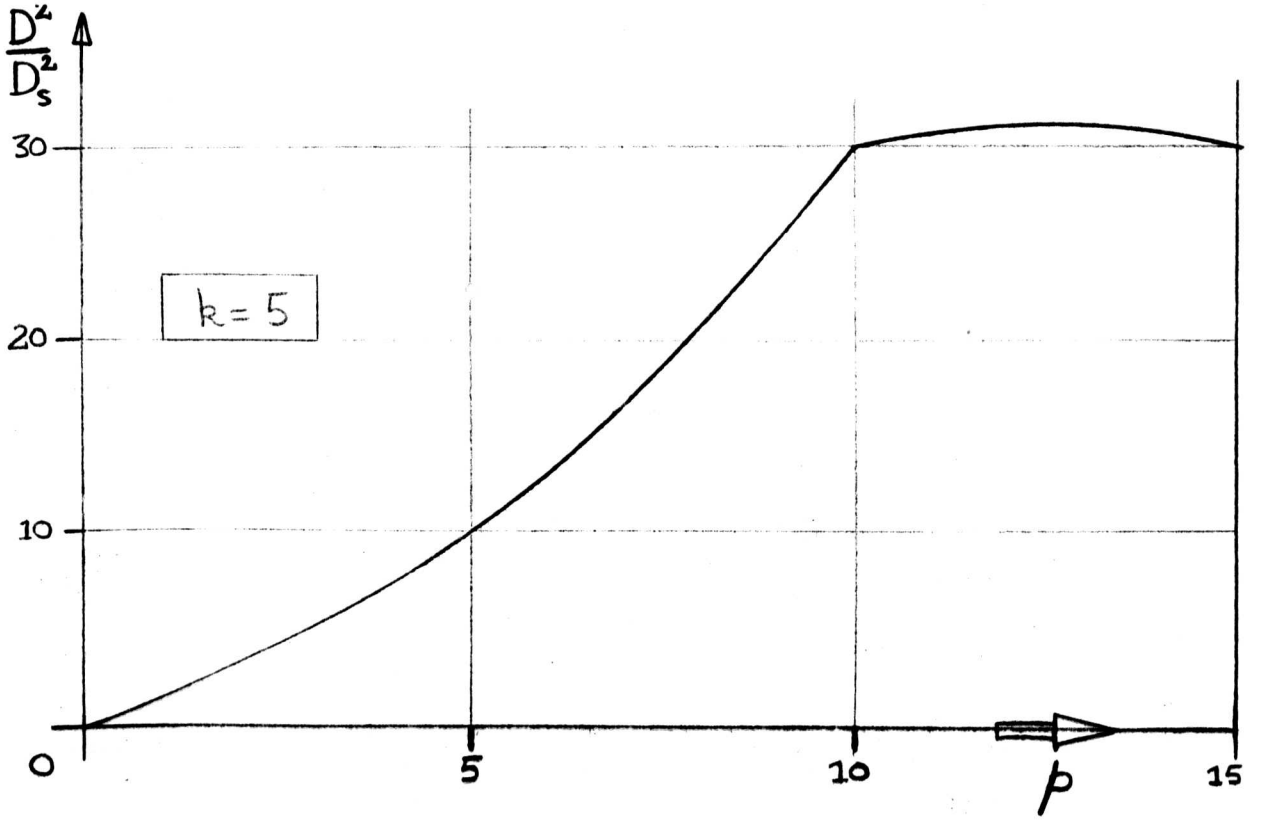


FIGURE 5.5: VARIANCE RATIO FOR N=15

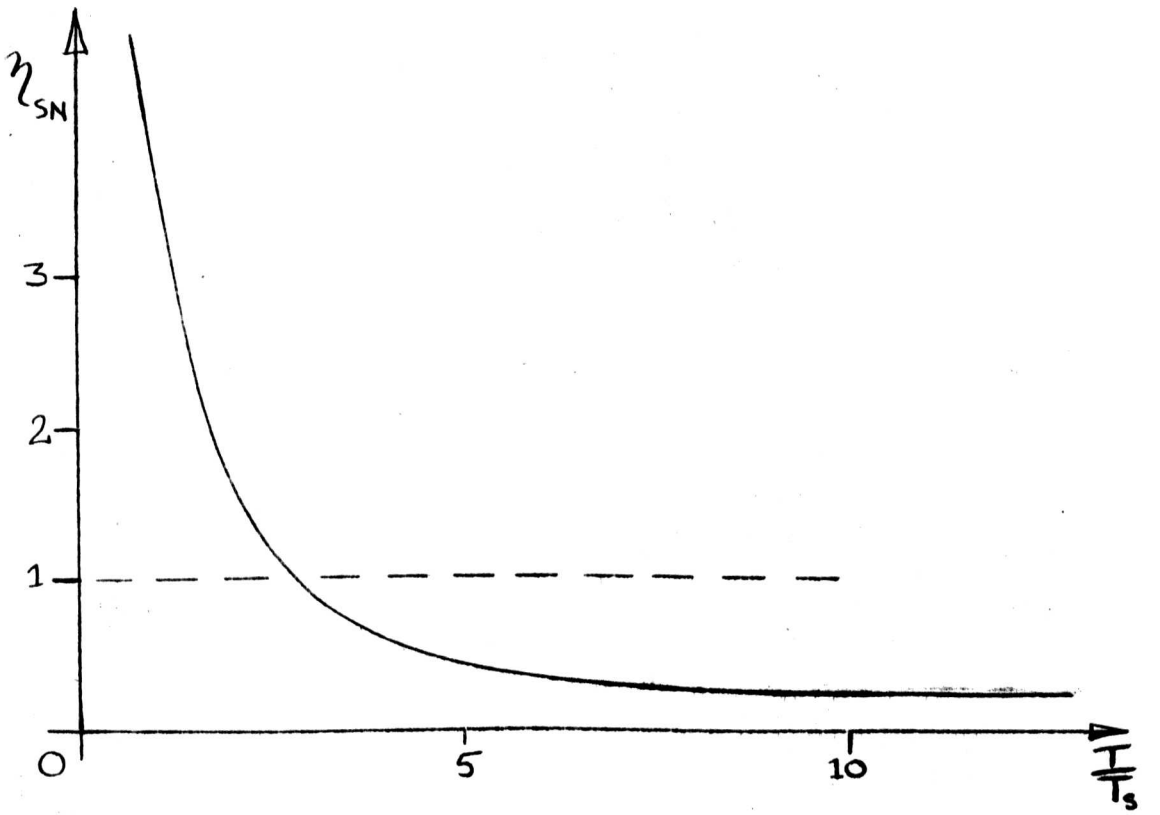


FIGURE 5.6: η_{SN} FOR $\eta_V = 30$ FOR 1ST-ORDER PLANT

Figure 5.6 shows η_{SN} as a function of T/T_s for the system of Section 5.3(d), with $\eta_V = 30$, as calculated from results in Sections 4.4(a) and 5.2. The systems give comparable signal-to-noise ratios: if a long perturbation period is tolerable, the sine-wave system is preferable; otherwise the chain-code system is slightly better. Of course if η_V is increased appreciably by increasing N or decreasing k, the sine-wave system will give better results under noisy conditions.

5.5 Conclusions

It has been shown that choosing a low frequency for a sine-wave perturbation system gives good signal-to-noise properties, which deteriorate as the frequency is increased. Formulae have been derived for the signal-to-noise ratios for the cases of band-limited and white noise, and it has been seen that the sine-wave system and the chain-code system give similar performances when operating under noisy conditions. For example, for a first-order plant, the sine-wave system gives a higher signal-to-noise ratio than a chain-code system with $N = 15$, $p = 15$, $k = 5$ if the perturbation period is more than 3 times the plant time-constant. For rapid optimization, where a short period is required, the chain-code system is preferable. If however N is increased appreciably or k decreased appreciably, the sine-wave system is better. It is not possible to summarize comparative results concisely, as they depend on the expected step response. The comparative procedure involves assessing the optimum values for N, k and p for the chain-code system, and then evaluating equation (5.25) using equations (4.18), (5.6) and (5.23). (The expected time to climb a hill using each system has not been evaluated, as the factors affecting this, as outlined in Section 9.3, would involve complex mathematical treatment that was considered to be beyond the scope of this work.)

PART II

PARALLEL-PATH OPTIMIZATION

PART II PARALLEL-PATH OPTIMIZATION

CHAPTER 6

OPTIMIZATION OF PLANTS WITH PARALLEL PERFORMANCE FUNCTION PATHS

6.1 Introduction

Various methods for optimizing plants using periodic test signals have been proposed in the literature. In this chapter, two such methods, one using sine waves and the other using chain-codes, will be compared when applied to plants with several parallel paths between any one input and the output. It is shown that, in general, both systems yield a false optimum when so applied, due to the dynamics of the parallel paths.

Section 6.2 contains an analysis for a general plant with M parallel paths, using the two types of perturbation. Section 6.3 gives quantitative theoretical results for a particular system, based on a model of a steam-generating plant. Practical results are given in Section 6.4. The conclusions are presented in Section 6.5.

6.2 Analysis of the Optimizing Methods

The general arrangement for optimization of one variable using periodic test signals is shown in Figure 6.1.

The method involves estimating the gradient of the performance function and changing the parameter $K_0(t)$ by amounts $\Delta K(t)$ until the gradient estimate, $Z(T)$, is zero. (It is assumed that the performance function has only one stationary point, and that this is a maximum of the function.) A discrete-adjustment technique, utilizing a sample-and-hold element after the gradient estimator[8,54], will be considered. This allows an adjustment of the parameter to be made only every T seconds.

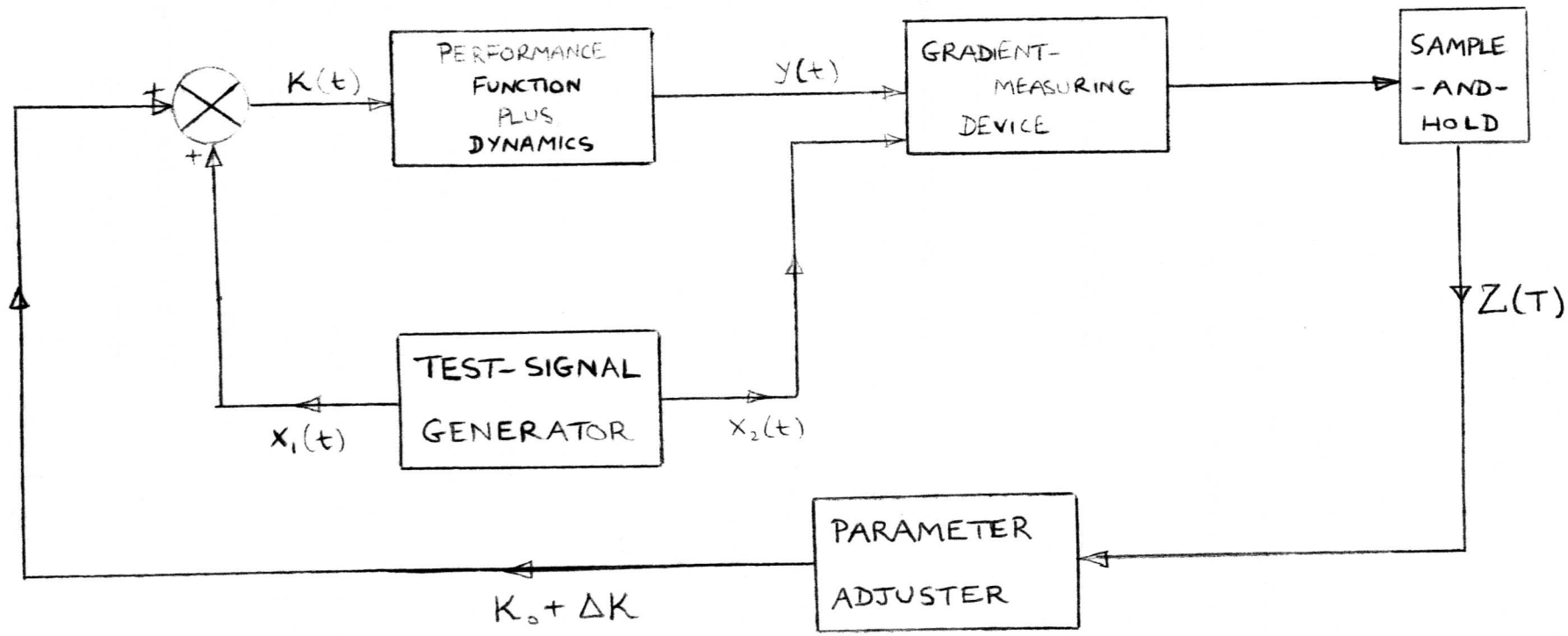


FIGURE 6.1: A GENERALIZED OPTIMIZER

It is well known that the dynamics of the adaptive loop affect the performance of the system.[47,48,55,56] By making ΔK small and T large, the dynamic effects can be reduced, at the expense of optimization speed, and eventually the system will settle, in the absence of disturbances, in a state where the gradient estimate is zero. This will only be the true optimum, however, if the estimated gradient is equal to the true gradient in the region of the optimum. We must therefore analyse the gradient estimation techniques for this region.

6.2(a) System With Sine-Wave Perturbation

Consider first the single-path sine-wave perturbation gradient estimator shown in Figure 6.2. Using a result derived in Section 1.2(c) in conjunction with equation (5.3), the output of the sample-and-hold is

$$Z(T) = \frac{a_s T}{2T_i} \left(\frac{\partial f}{\partial K}\right) \int_0^\infty h(\tau) \cos w\tau \, d\tau \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6.1)$$

assuming $h(\tau)$ is constant with respect to time, and using once again the somewhat unrealistic model of a performance 'hill' followed by dynamics.

Consider now the plant of Figure 6.3, with M parallel paths, each containing a performance function and associated dynamics. Once again this is not a very realistic model of any practical plant, but it will help in the study of the performance of the optimizer. A preliminary investigation of a 2-path first-order plant carried out by Thomas[57] has shown that errors arise in the sine-wave gradient estimation process, due to the dynamics of each path. A general theory will now be developed to evaluate the magnitude of such errors for a plant with any number of parallel paths, with dynamics of any order in each path.

The gradient estimate is given by

$$Z(T) = \frac{1}{T_i} \int_0^T x_2(t).y(t)\,dt$$

$$= \frac{1}{T_i} \int_0^T x_2(t). \left[\sum_{j=1}^{M} y_j(t)\right] dt$$

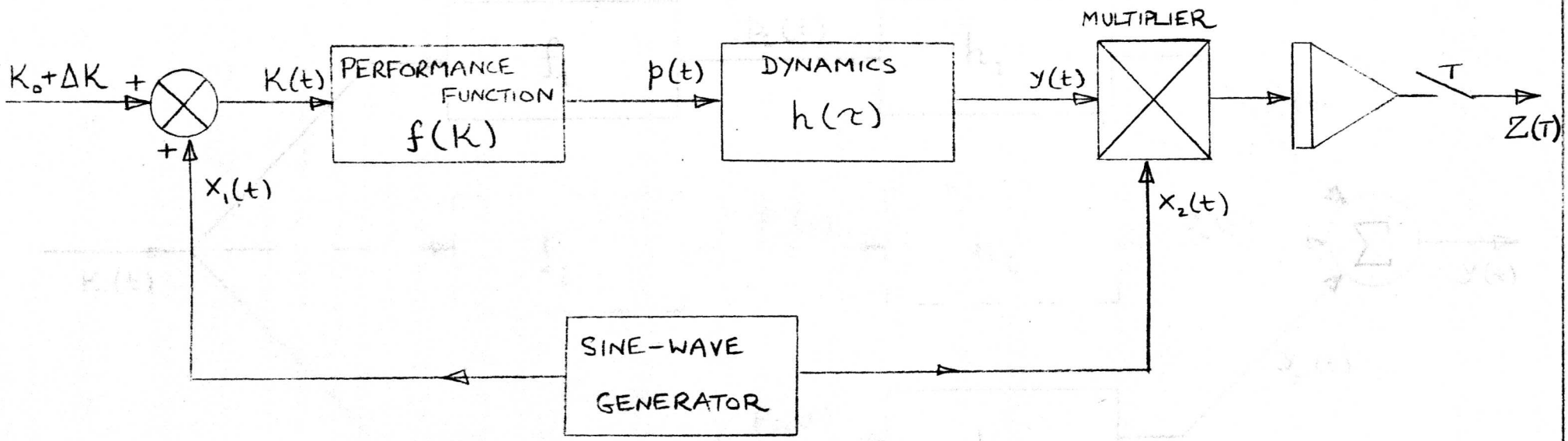


FIGURE 6.2: GRADIENT ESTIMATION USING SINE-WAVE PERTURBATIONS

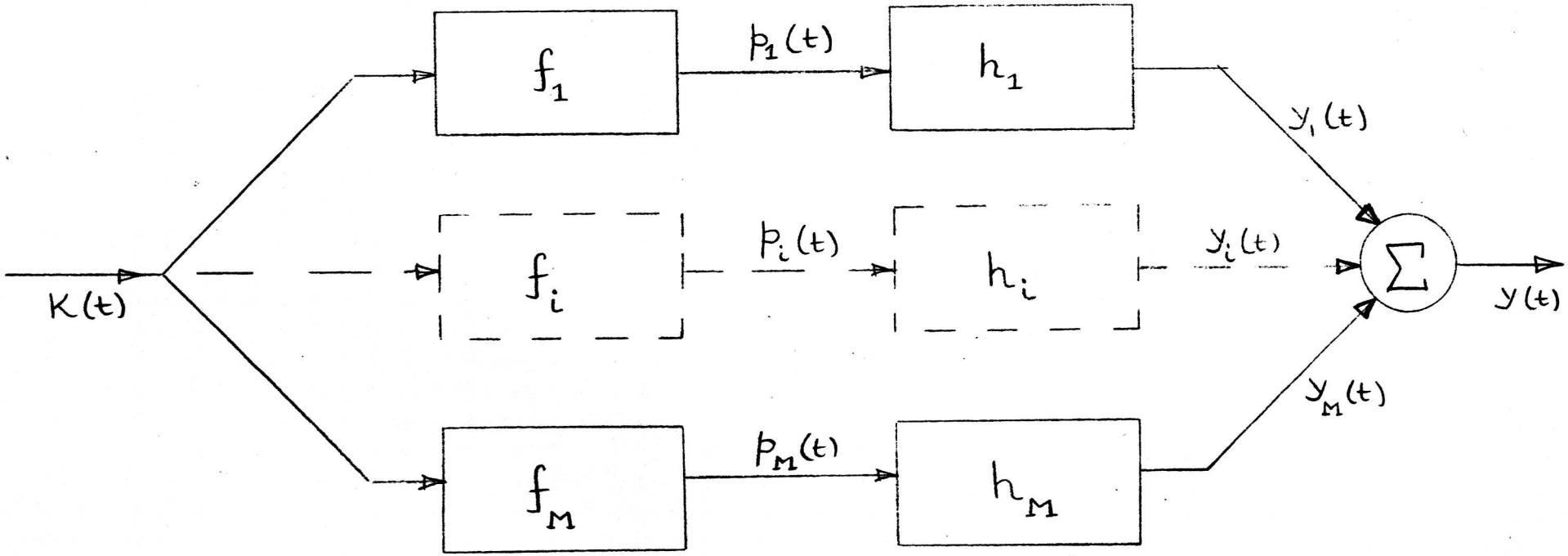


FIGURE 6.3: PLANT WITH M PARALLEL PATHS

$$= \sum_{j=1}^{M} \left[\frac{1}{T_i} \int_o^T x_s(t)\ y_j(t)\,dt \right]$$

$$= \sum_{j=1}^{M} z_j$$

where z_j is the gradient estimate due to the j-th path with the other paths absent.

Hence, from equation (6.1),

$$Z(T) = \frac{a_s T}{2T_i} \sum_{j=1}^{M} \left[\frac{\partial f_j}{\partial K} \int_o^\infty h_j(\tau) \cos w\tau\, d\tau \right]$$

At the estimated optimum, $Z(T) = 0$.

Then $$\sum_{j=1}^{M} \left[\frac{\partial f_j}{\partial K} \int_o^\infty h_j(\tau) \cos w\tau\, d\tau \right] = 0 \qquad (6.2)$$

i.e. $$\sum_{j=1}^{M} \frac{\partial f_j}{\partial K} F_j(h_j, w) = 0 \qquad (6.3)$$

where $$F_j(h_j, w) = \int_o^\infty h_j(\tau) \cos w\tau\, d\tau \qquad (6.4)$$

But the true optimum is given by $\frac{\partial y}{\partial K} = 0$.

And $$\frac{\partial y}{\partial K} = \frac{\partial}{\partial K} \left[\sum_{j=1}^{M} y_j \right] = \frac{\partial}{\partial K} \left[\sum_{j=1}^{M} f_j(K) \right] = \sum_{j=1}^{M} \frac{\partial f_j}{\partial K}$$

Therefore at the true optimum $$\sum_{j=1}^{M} \frac{\partial f_j}{\partial K} = 0 \qquad (6.5)$$

Equation (6.3) only reduces to equation (6.5) if

$$F_j(h_i, w) = F_j(h_j, w)$$

for all i and j between 1 and M. But this is not true by hypothesis. Hence the sine-wave perturbation system yields a false estimate of the optimum, given by equation (6.3), when applied to a plant with several parallel paths.

Note that for M = 1, the estimated optimum is given by

$$\frac{\partial f_1}{\partial K} F_1(h_1, w) = 0$$

which returns to $\frac{\partial f_1}{\partial K} = 0$, which is the correct equation for the optimum. Thus although the gain estimate for the single-path system may be substantially

in error at some distance on either side of the optimum (see Section 5.2), at the optimum it is identically zero as required.

6.2(b) System with Chain-Code Perturbation

Consider the relevant part of the single-path chain-code perturbation gradient estimator, shown in Figure 6.4. In this system the second running averager enables the d.c. component to be eliminated.

The gain estimate for this system can be expressed as a sum of terms (as for the sine-wave analysis), viz.:

$$\hat{S}(T_r) = \mu\left[\sum_{j=1}^{M}\frac{\partial f_j}{\partial K}\int_{\tau=0}^{\infty} h_j(\tau)\int_{t=0}^{T} c(t-\tau)\rho(t)\,dt\,d\tau\right] \quad \ldots\ldots\ldots\ldots(6.6)$$

The estimated optimum is then given by

$$\sum_{j=1}^{M}\frac{\partial f_j}{\partial K}\int_{\tau=0}^{\infty} h_j(\tau)\int_{t=0}^{T} c(t-\tau)\,(t)\,dt\,d\tau = 0 \quad \ldots\ldots\ldots\ldots(6.7)$$

Although

$$\int_{\tau=0}^{\infty} h_j(\tau)\,d\tau = 1$$

for all j by hypothesis,

$$\int_{\tau=0}^{\infty} h_i(\tau)\int_{t=0}^{T} c(t-\tau)\rho(t)\,dt\,d\tau \neq \int_{\tau=0}^{\infty} h_j(\tau)\int_{t=0}^{T} c(t-\tau)\rho(t)\,dt\,d\tau$$

if $h_i(\tau)$ and $h_j(\tau)$ are distinct. Equation (6.7) does not therefore reduce to equation (6.5), and a false estimate of the optimum is obtained.

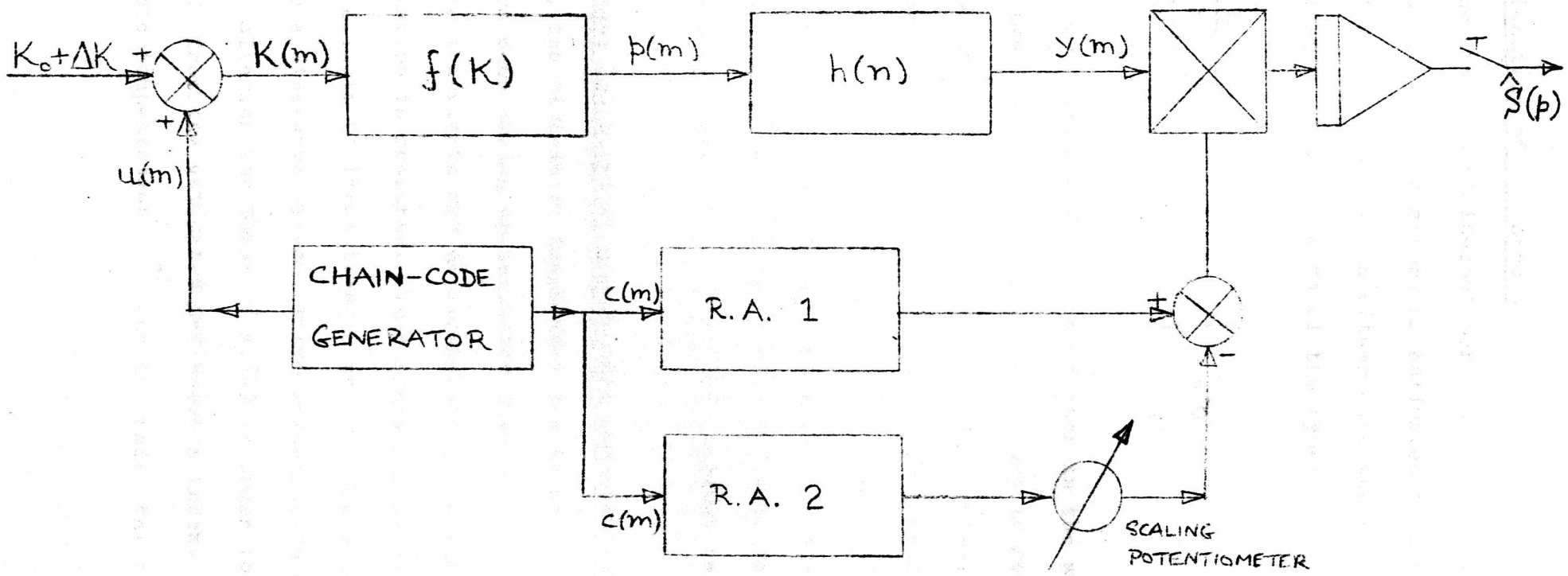


FIGURE 6.4: GRADIENT ESTIMATION USING PSEUDO-RANDOM PERTURBATIONS

6.2(c) Physical Explanation of the Errors

The sine-wave and chain-code perturbation methods have both been shown to give rise to errors in estimated optimum for plants with parallel dynamic paths. In either case the estimate of optimum is given by an expression of the type:

$$\sum_{j=1}^{M} \left[\frac{\partial f_j}{\partial K} \; F_j(h_j , T) \right] = 0$$

The function F_j represents an error factor in the assessment of the gain for the j-th path. It is clear that in general this factor will be different for different dynamic paths, in which case, the above equation will not reduce to that required for a true estimate of the optimum as given by equation (6.5). In order to evaluate the estimated optimum using perturbation techniques, it is necessary to calculate the error factors, F_j, for each path. Section 6.3 considers various cases.

6.2(d) The Use of Phase Compensation and Dynamic Compensation

So far, the effects of transients due to step changes in the parameters made during optimization have been ignored. In practice these transients may be significant, particularly if rapid optimization is required. Techniques are available for reducing the effects of these transients for single-path systems. For a sine-wave system, phase compensation can be used; this involves altering the phase of $x_2(t)$ in order to remove the terms involving ΔK from the estimated performance index. For a chain-code system, dynamic compensation[47,48] can be used: the change in system

output due to ΔK is assessed from the latest estimate of step response, and subtracted from the actual system output before cross-correlating. This is discussed more fully in Chapter 9.

Consider now attempts to apply compensation to multi-path systems. Phase compensation requires a priori knowledge of the phase-shift of the plant, from parameter perturbation to performance index. This phase-shift depends on the relative gains of the paths in a multi-path system, and therefore changes during optimization. If phase compensation is applied, it will only be of the correct magnitude at certain times during optimization; at other times it may be grossly in error. Hence phase compensation may be of little or no value in a sine-wave multi-path system. For a chain-code multi-path system, on the other hand, dynamic compensation can be used, as the effective step response of the overall parameter path is continually assessed, so that the effect of a step change in parameter on the estimated performance index can be calculated with some accuracy at all times during the optimization process. (See Section 9.2, paragraph (vi), for a description of the errors that arise in this process).

6.3 Application to a Particular Problem

The error in estimated optimum for a particular plant model, using the ~~sine-wave~~ perturbation methods will now be examined.

6.3(a) The Plant Model

An approximate model of a steam-generating plant will be used, based on work by Moran, et al.[2] The plant configuration is shown in Figure 6.5.

The variables that we are interested in are:-

(i) the air/fuel ratio at entry to the furnace, K(t), and

(ii) the heat transferred to the water/steam throughout the plant, y(t).

We wish to find the value of K(t) that maximizes y(t).

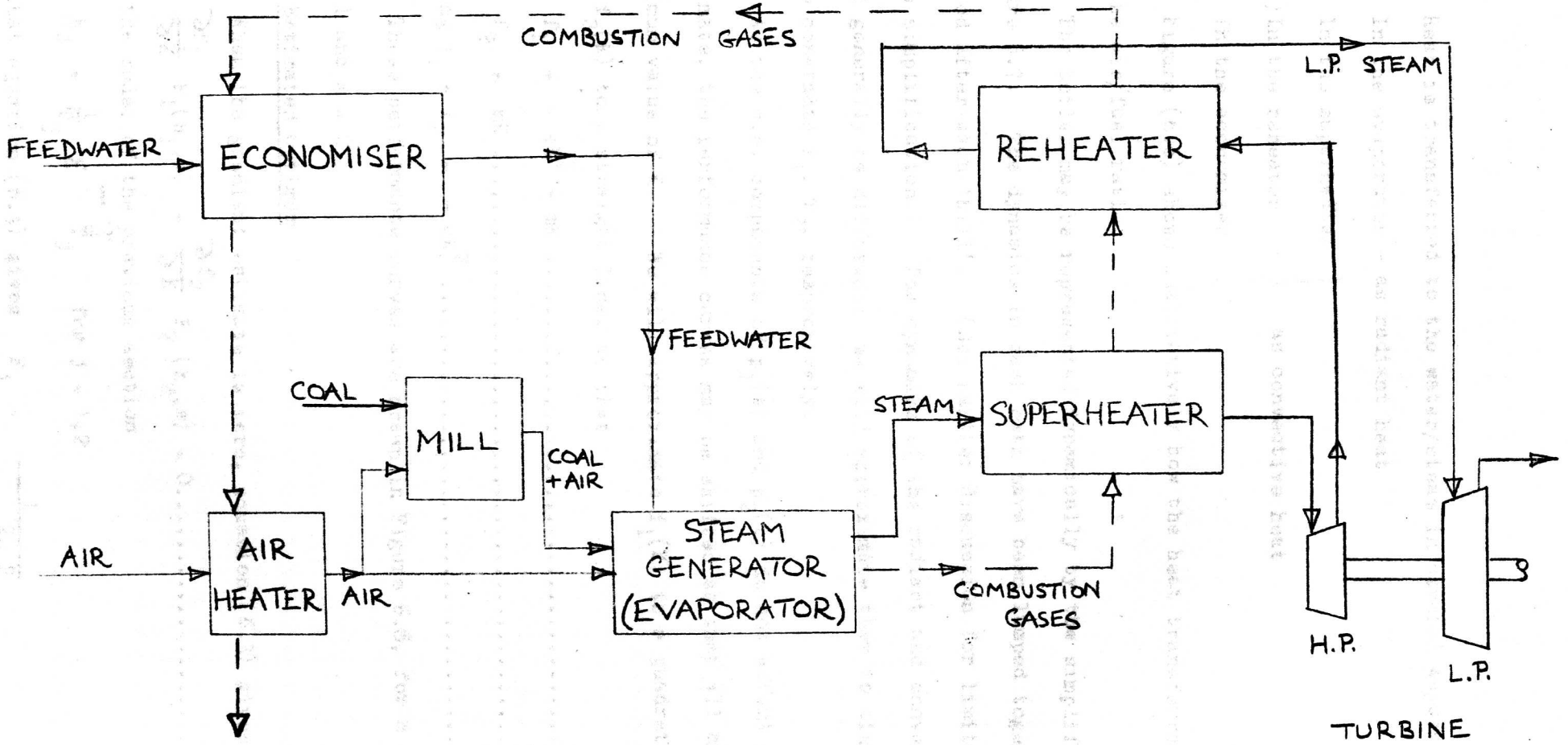


FIGURE 6.5 : BOILER BLOCK DIAGRAM (FROM REF. 2)

Heat is transferred to the water/steam in several ways:-

(i) In the evaporator - as radiant heat

(ii) In the superheater)
(iii) In the reheater) as convective heat
(iv) In the economizer)

Figure (6.6) shows qualitatively how the heat transferred depends on the air/fuel ratio.

The boiler may be represented dynamically by the simplified model of Figure 6.7. The dynamics in each path have been lumped together, and placed after each 'hill'. (See earlier discussion for limitations of this simplification.) The dynamics of the radiant and convective paths will generally be different; we will approximate them to single lags with time constants T_1, T_2, respectively.

As the d.c. components of $p_1(t)$ and $p_2(t)$ do not affect the gradient estimate, the performance curves may be shifted so that all are zero at the optimum value of K. We will approximate $f_1(K)$ to a quadratic function and $f_2(K)$ to a straight line, so that

$$f_1 = -cK^2 - mK \quad \ldots\ldots(6.8)$$

and $$f_2 = mK \quad \ldots\ldots(6.9)$$

giving $$f_1 + f_2 = -cK^2 \quad \ldots\ldots(6.10)$$

These performance curves are drawn in Figure 6.8, for c = 1 with m = 1 and m = 2.

6.3(b) The Estimated Optimum

When the gradient estimate is zero, equation (6.3) gives

$$\frac{\partial f_1}{\partial K} F_1(h_1,w) + \frac{\partial f_2}{\partial K} F_2(h_2,w) = 0 \quad \ldots\ldots(6.11)$$

For the model of the previous section

$$h_j = \frac{1}{T_j} \varepsilon^{-\frac{\tau}{T_j}} \quad \text{for } j = 1,2$$

So that equation (6.4) gives $$F_j = \frac{1}{1 + w^2 T_j^2}$$

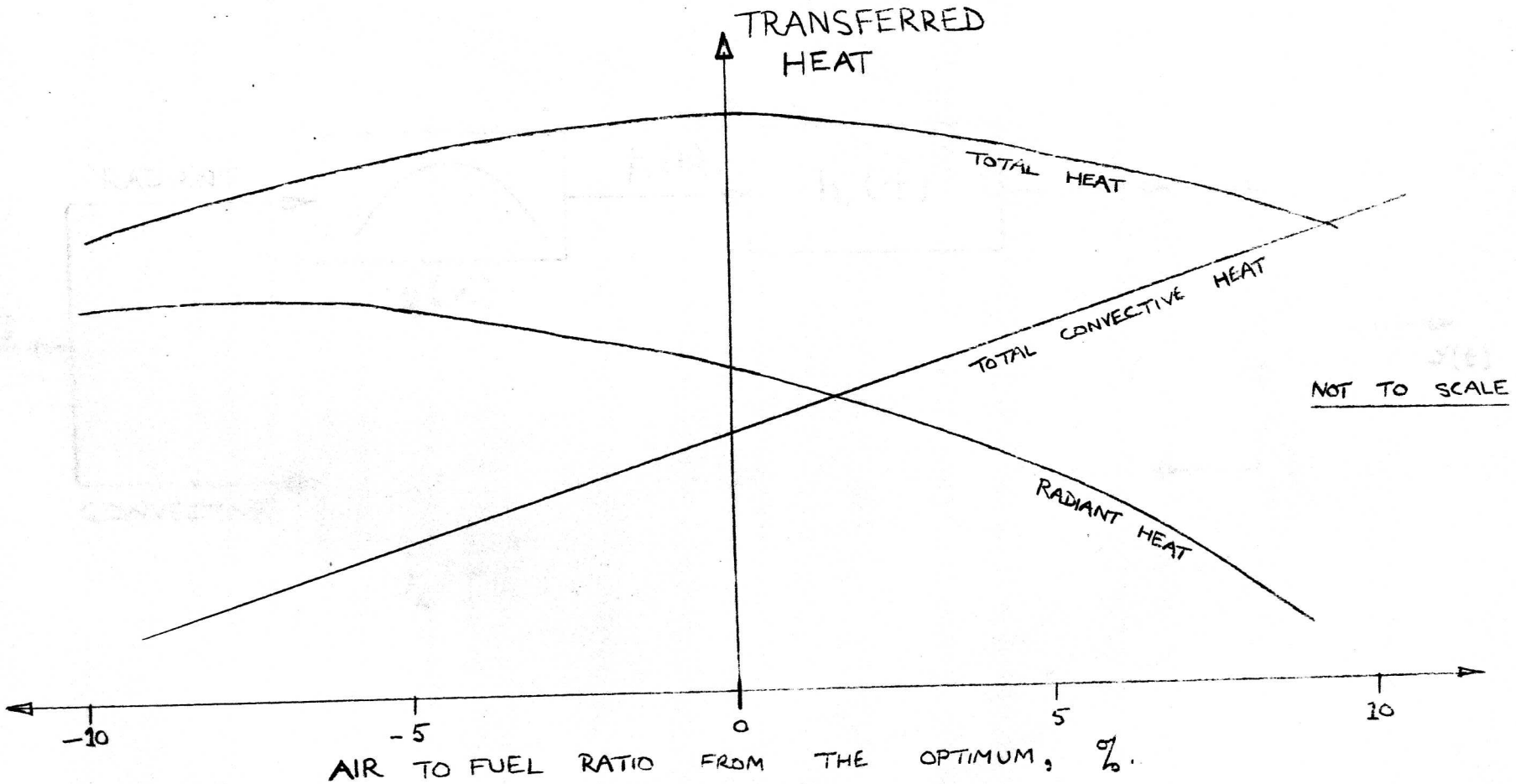


FIGURE 6.6 : HEAT TRANSFER CHARACTERISTICS

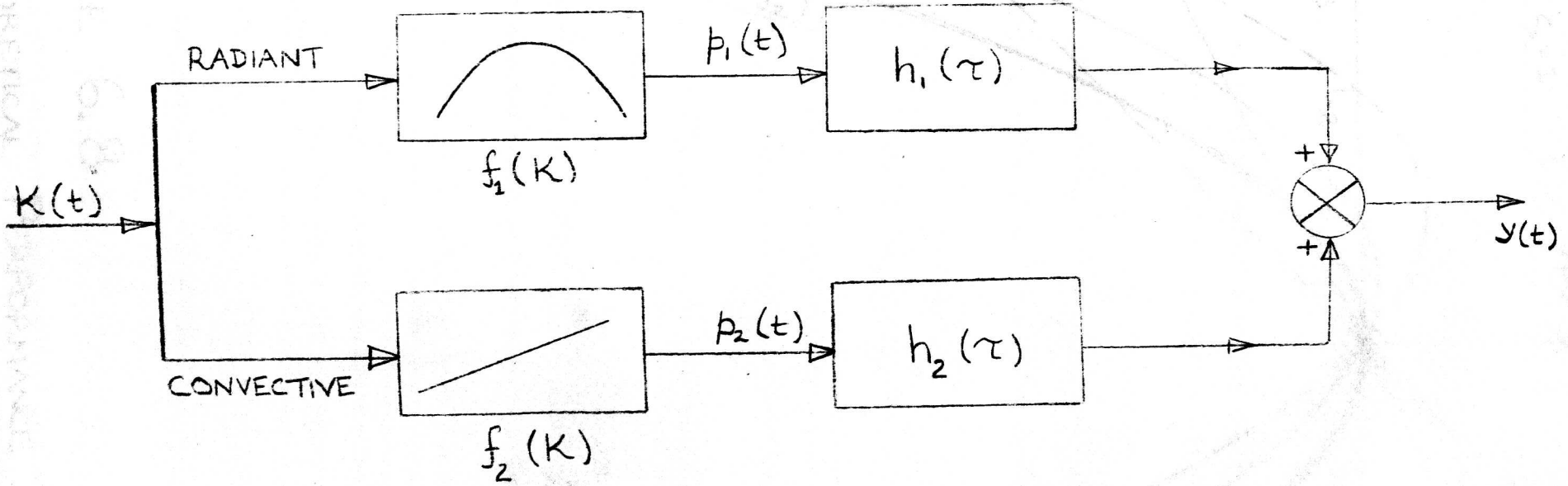


FIGURE 6.7 : DYNAMIC MODEL OF BOILER

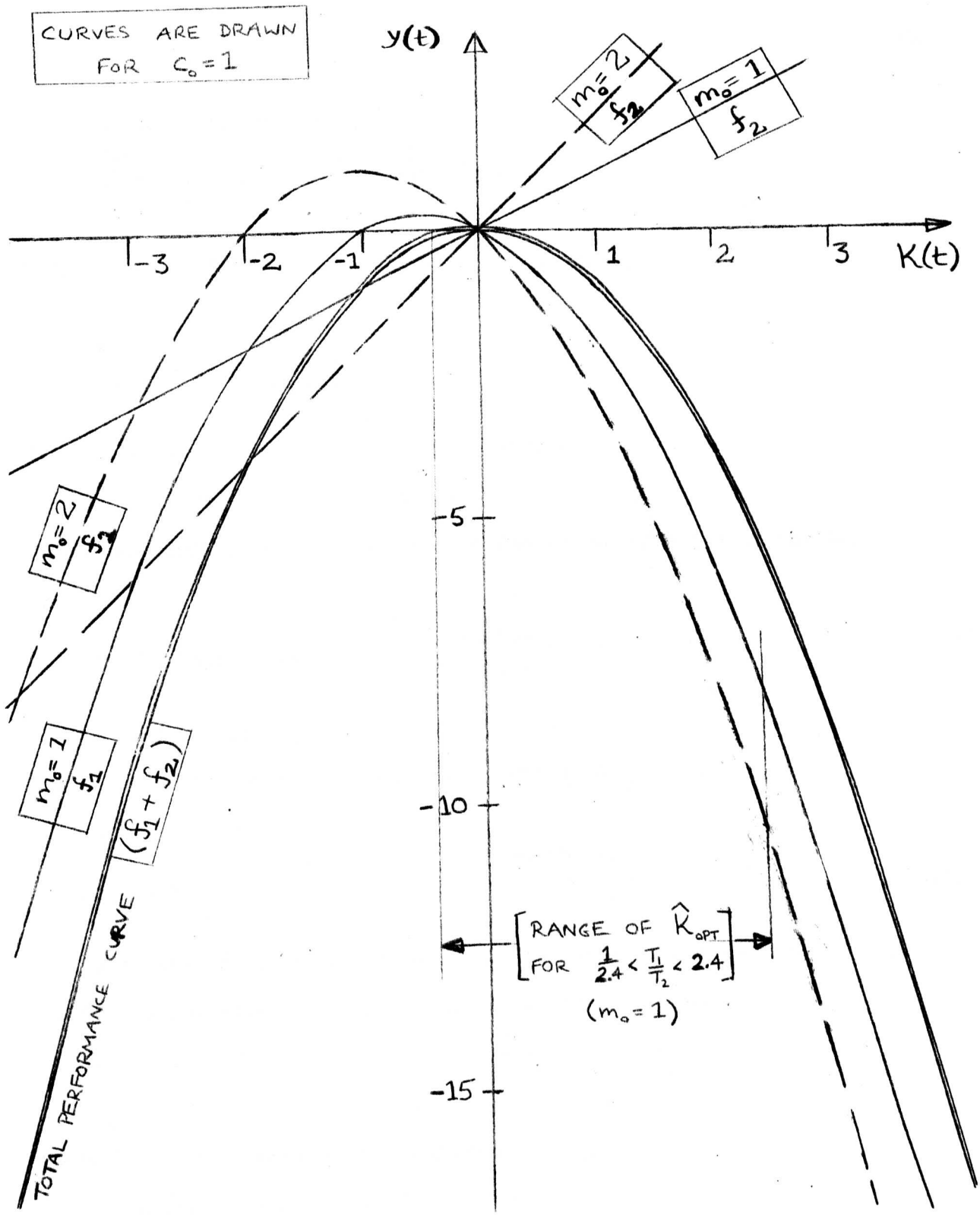


FIGURE 6.8:

THEORETICAL PERFORMANCE CURVES FOR BOILER

Substituting in equation (6.11)

$$\frac{\partial f_1}{\partial K}\left(\frac{1}{1+w^2T_1^2}\right) + \frac{\partial f_2}{\partial K}\left(\frac{1}{1+w^2T_2^2}\right) = 0 \qquad \text{.....(6.12)}$$

From equations (6.8), (6.9) and (6.12), the estimated optimum value for K is

$$\hat{K}_{opt} = \frac{mw^2(T_1^2-T_2^2)}{2c\,(1+w^2T_2^2)} \qquad \text{.....(6.13)}$$

But, by hypothesis, the true optimum value is

$$K_{opt} = 0$$

Hence the shift in optimum value of K due to the dynamics of the model is

$$K_e = \hat{K}_{opt} - K_{opt} = \frac{mw^2(T_1^2-T_2^2)}{2c\,(1+w^2T_2^2)} \qquad \text{.....(6.14)}$$

A useful criterion for measuring the effect of this shift is to evaluate the slope of the overall performance curve at $\hat{K}_{opt}$, that is

$$S_e(w) = \left(\frac{\partial f}{\partial K}\right)_{\hat{K}_{opt}}$$

In this case $S_e(w) = -2c\,\hat{K}_{opt} = \frac{mw^2(T_2^2-T_1^2)}{1+w^2T_2^2}$ .....(6.15)

Note that S_e is independent of c, and hence of the magnitude of the overall performance curve, and that it is directly proportional to m.

We shall define the normalized frequency of the sine-wave as

$$f_N = f_s.T_2$$

where f_s = frequency of the sine-wave = $w/2\pi$

Defining the normalized slope-error as

$$S_N = \frac{S_e}{m}$$

and using equation (6.15) then gives

$$S_N(f_N) = \frac{4\pi^2 f_N^2}{1+4\pi^2 f_N^2}\left[1-\left(\frac{T_1}{T_2}\right)^2\right] \qquad \text{.....(6.16)}$$

Curves of S_N versus f_N are plotted in figure 6.9 for various values of T_1/T_2.

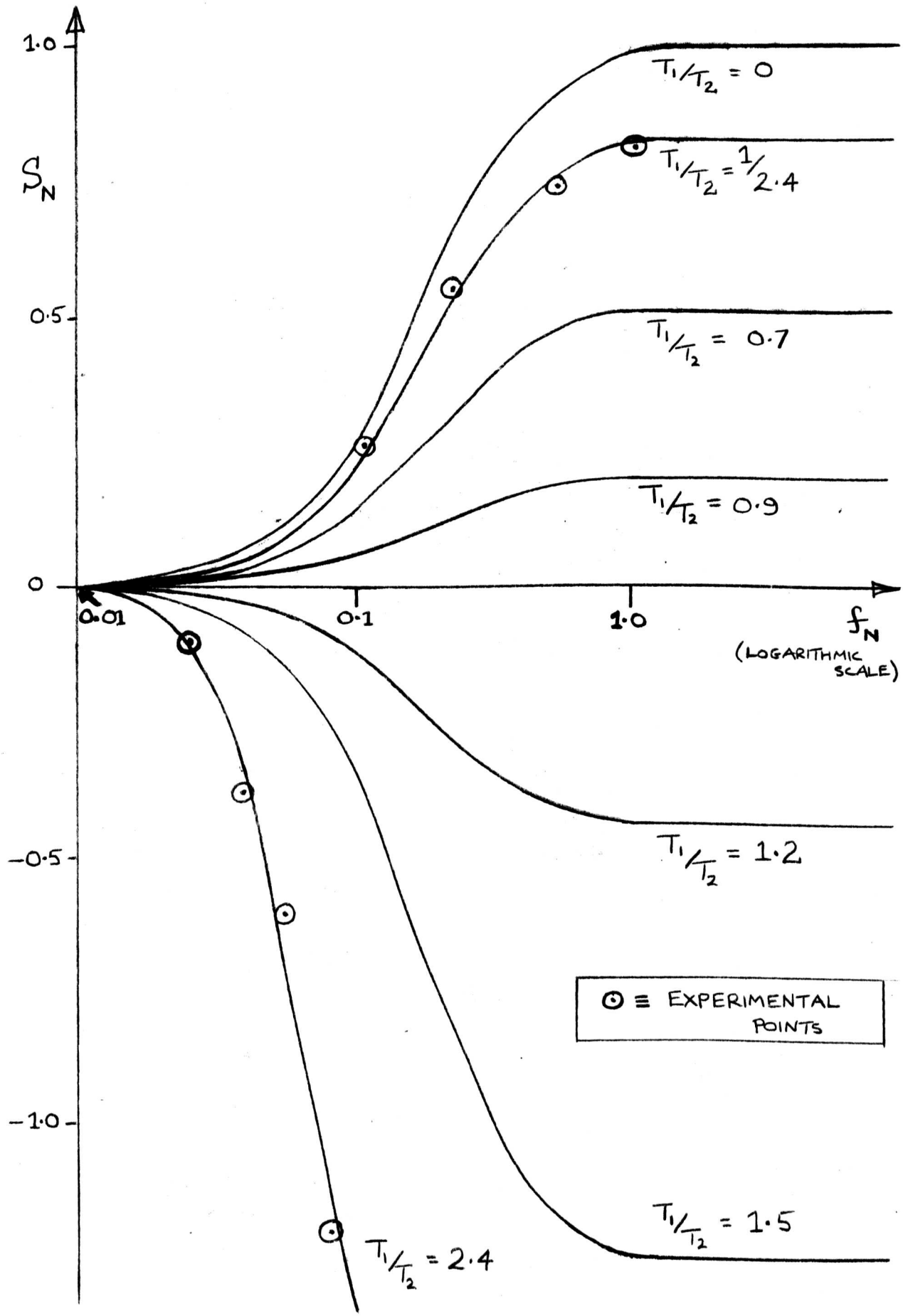


FIGURE 6.9: NORMALIZED SLOPE ERROR CURVES (SINE-WAVE)

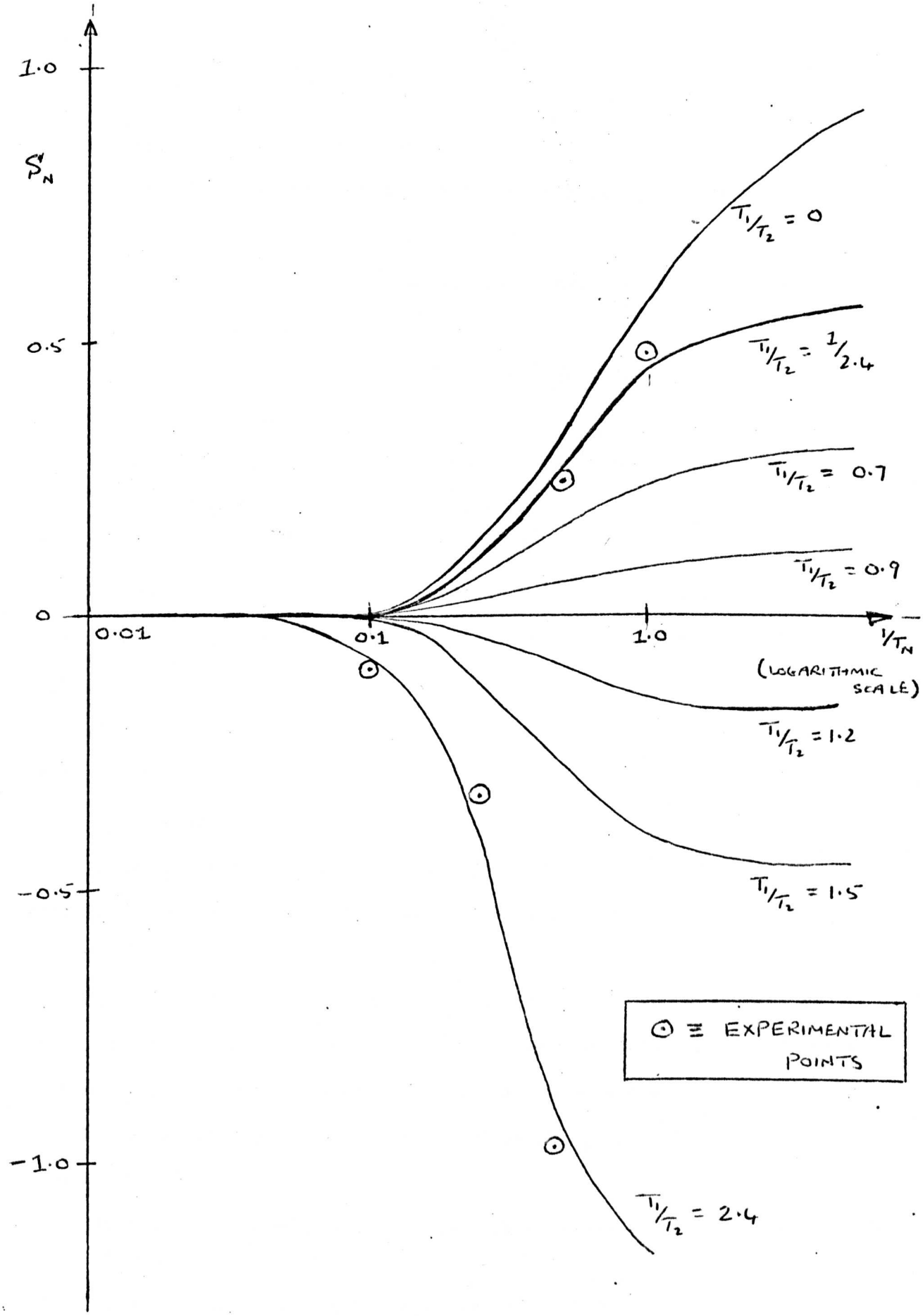


FIGURE 6.9 (a): NORMALIZED SLOPE ERROR CURVES

(CHAIN – CODE)

The following observations can be made:

(i) The direction of the shift in optimum depends on the sign of $\left(1 - \frac{T_1}{T_2}\right)$.

(ii) For any given frequency, $|S_N|$ increases as $\left|1 - \frac{T_1}{T_2}\right|$ increases.

(iii) For any given value of T_1/T_2, $|S_N|$ increases with frequency.

(iv) If q be any positive integer greater than unity,

$$\left|S_N(f_N)\right|_{\frac{T_1}{T_2} = \frac{1}{q}} < \left|S_N(f_N)\right|_{\frac{T_1}{T_2} = q}.$$

In practice, a limit on the permissible excursions of the optimum will be set by the designer. Suppose, for our particular plant, that this implies that

$$|S_N| < 0.2$$

Suppose also that

$$T_2 \simeq 2T_1$$

(This corresponds to a suggestion by Moran that the time constant for convectice heat transfer in the plant is roughly twice that for radiant transfer.)

Then it will be observed that the period of the sine-wave must be slightly longer than ten times the longer time constant. This imposes a severe restriction on the speed of optimization possible.

It is of interest to consider the value of S_N as $f_N \to \infty$. From equation (6.16) we have

$$\text{Limit}_{f_N \to \infty} S_N = 1 - \left(\frac{T_1}{T_2}\right)^2 \quad \text{.....(6.17)}$$

The values of this function are plotted in Figure 6.10, over the range of T_1/T_2 for which $|S_N(\infty)| < 1$. It will be seen that $|S_N(\infty)|$ increases rapidly as T_1/T_2 increases from unity. This restricts the use of high-frequency perturbation techniques[21] to systems where $T_1/T_2 \simeq 1$ (when a single-path representation would be adequate anyway).

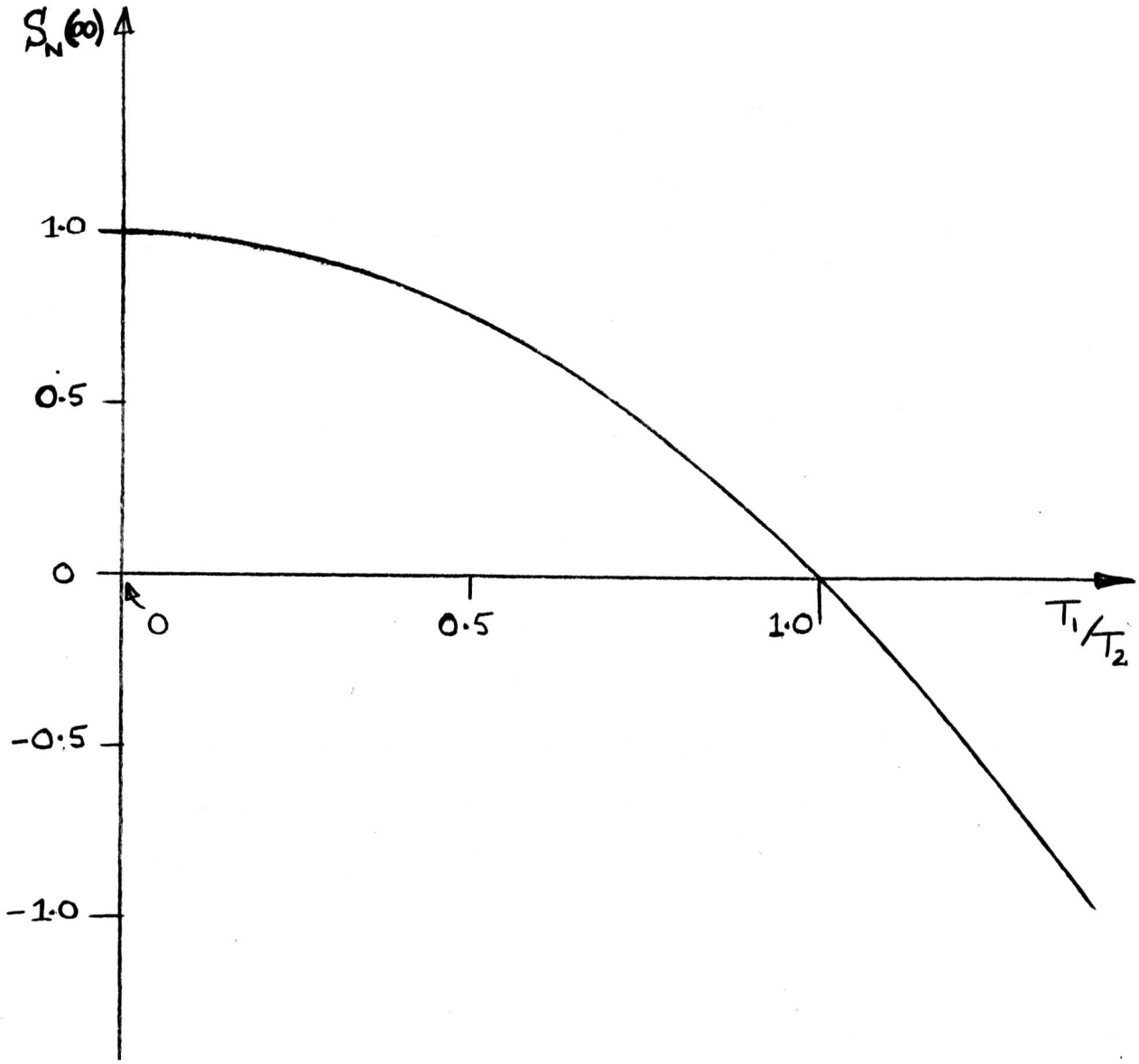


FIGURE 6.10:

LIMITING VALUES OF NORMALIZED SLOPE ERROR

(SINE WAVE)

Turning now to the errors that will arise when using chain-codes, consider the general expression for equation (6.16), applicable to a plant of any order:

$$S_n = \frac{F_1 - F_2}{F_1}$$

The analysis for chain-code perturbations runs parallel to that for sine-waves, and the equation above holds then also. The values for F_1 and F_2 can be deduced from equation (4.18). For the first-order plant, with p=N-1, k=2 as example, equation (4.18) reduces to:

$$\hat{S}\,(30) \simeq \frac{\partial f}{\partial K}\left\{1 + 30\left[\hat{\phi}(n_c) - h(\infty)\right]\right\}$$

so that the only significant error is that due to the bias-estimation process.

The slope-error curves for this case are shown in Figure 6.9(a), over the same range of values as for the sine-wave system. T_n is the normalized period of the chain-code, corresponding directly with f_n. The curves show no well-defined pattern for large values of $1/T_n$, but under such circumstances the individual step responses are so much in error that their ratio becomes meaningless, and is therefore not considered further.

It is interesting to note that the curves for the chain-code system are very similar to those for the sine-wave system, but, for any given values of the parameters, the sine-wave system gives a larger error in estimating the optimum than the equivalent chain-code system. As the estimate of optimum depends on the relative errors in estimating the gain for each path, the results of this chapter tie in with those of Chapter 5, in which it was shown that the error in estimating gain for a single path is greater for the sine-wave

system than for the chain-code system in the absence of noise. However, it was also shown in Chapter 5 that, when noise is present, the variance is lower for the sine-wave system than for the chain-code system. It is reasonable to infer that when noise is present there is not much to choose between the two systems when applied to plants having parallel dynamic paths, as in the case of a single path plant, but that for fast optimization the chain-code may be better. To decide on the best approach in any given situation, it is necessary to have some idea of the plant and noise characteristics, and hence to evaluate the expected estimate of optimum using the results of Chapters 4,5 and 6. (In this thesis, time has limited the analysis to the noise-free case, but extension to the noisy case would be straightforward, using the results of Chapters 4,5 and 6.

We have thus demonstrated the restrictions imposed by the dynamics of a process-control plant with more than one dynamic path between parameter and output, when attempting to optimize the plant using sine-wave or chain-code perturbations.

6.4 Practical Results

The model of the steam generating plant, analysed in the previous section, was simulated on the analogue computer.

6.4(a) The Experimental Configurations

Figure 6.11 shows the patching arrangement, and Figures 6.12 and 6.13 illustrate the configurations for the two perturbation methods. M1 and M2 are quarter-squares multipliers, and F is an integrator. For the sine-wave method, the timing facility of the analogue computer enabled F to be used in a sample-and-hold mode. In order to simulate the two lags continuously, the normal computer integrators could not be used, as they would operate in the sample-

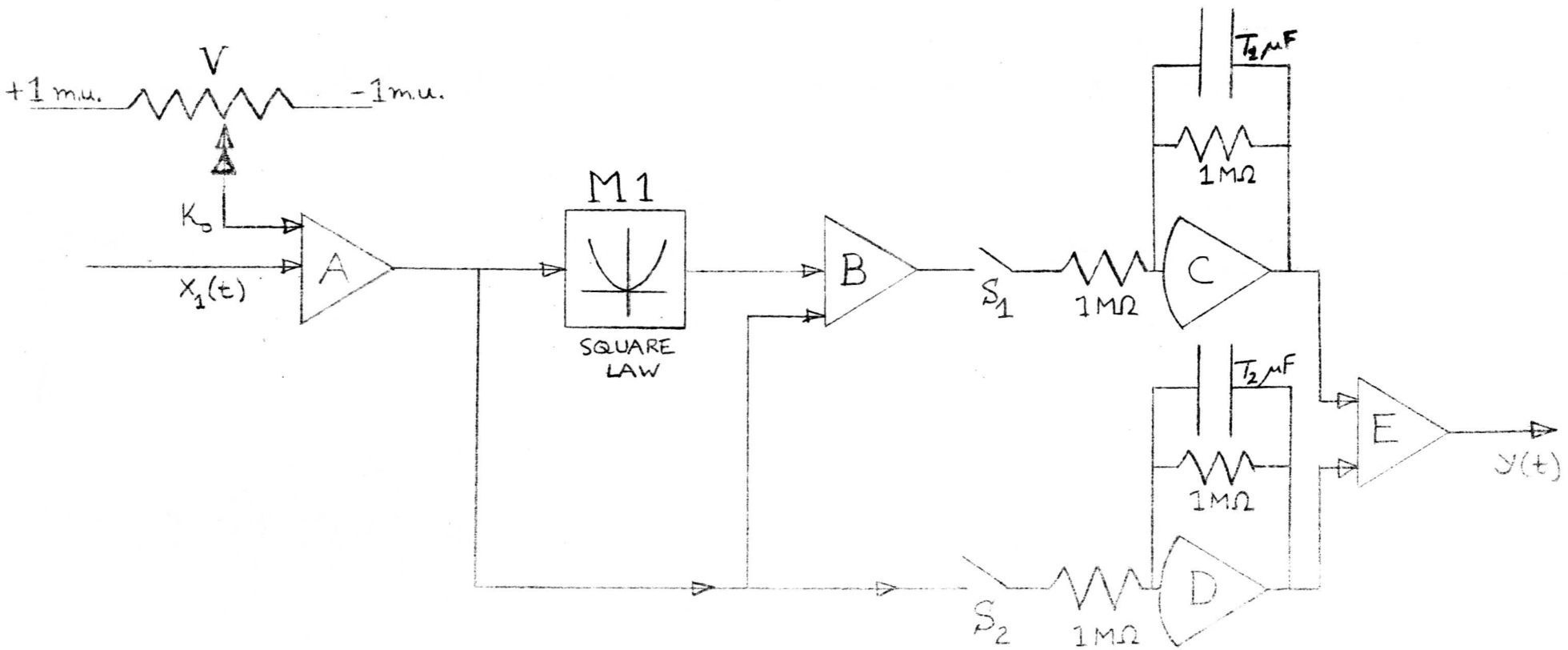


FIGURE 6.11. ANALOGUE COMPUTER SIMULATION OF BOILER MODEL

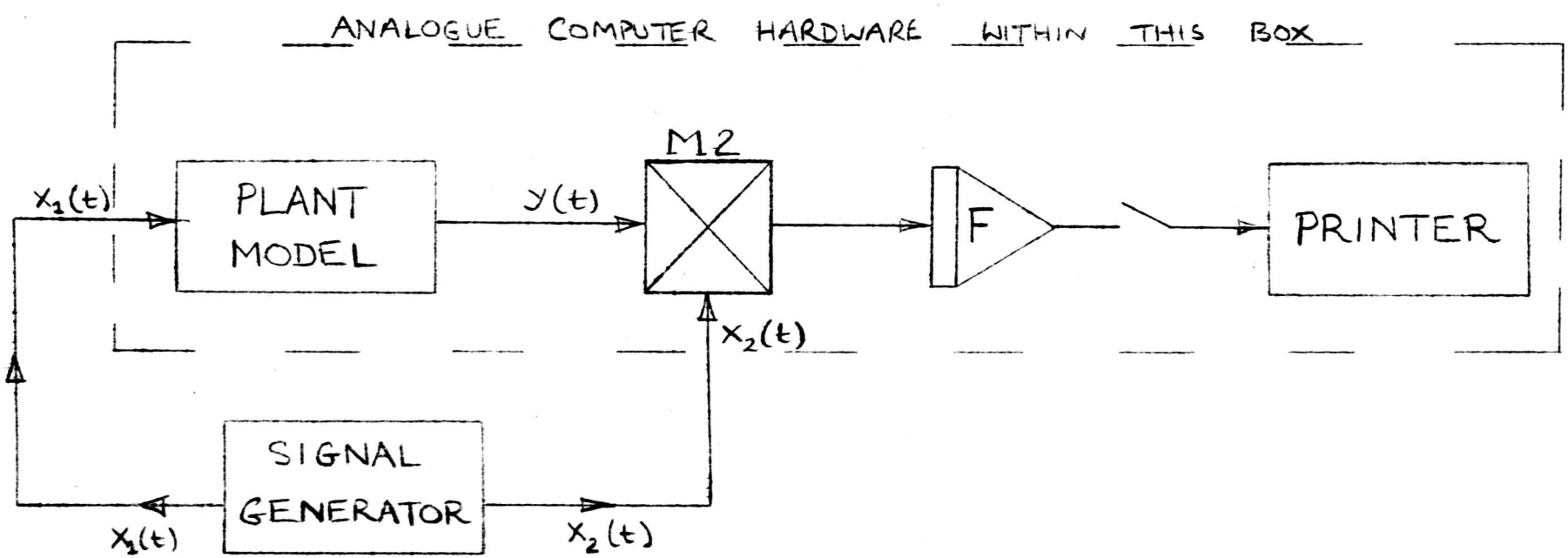


FIGURE 6.12 : CONFIGURATION FOR SINE-WAVE PERTURBATION EXPERIMENT

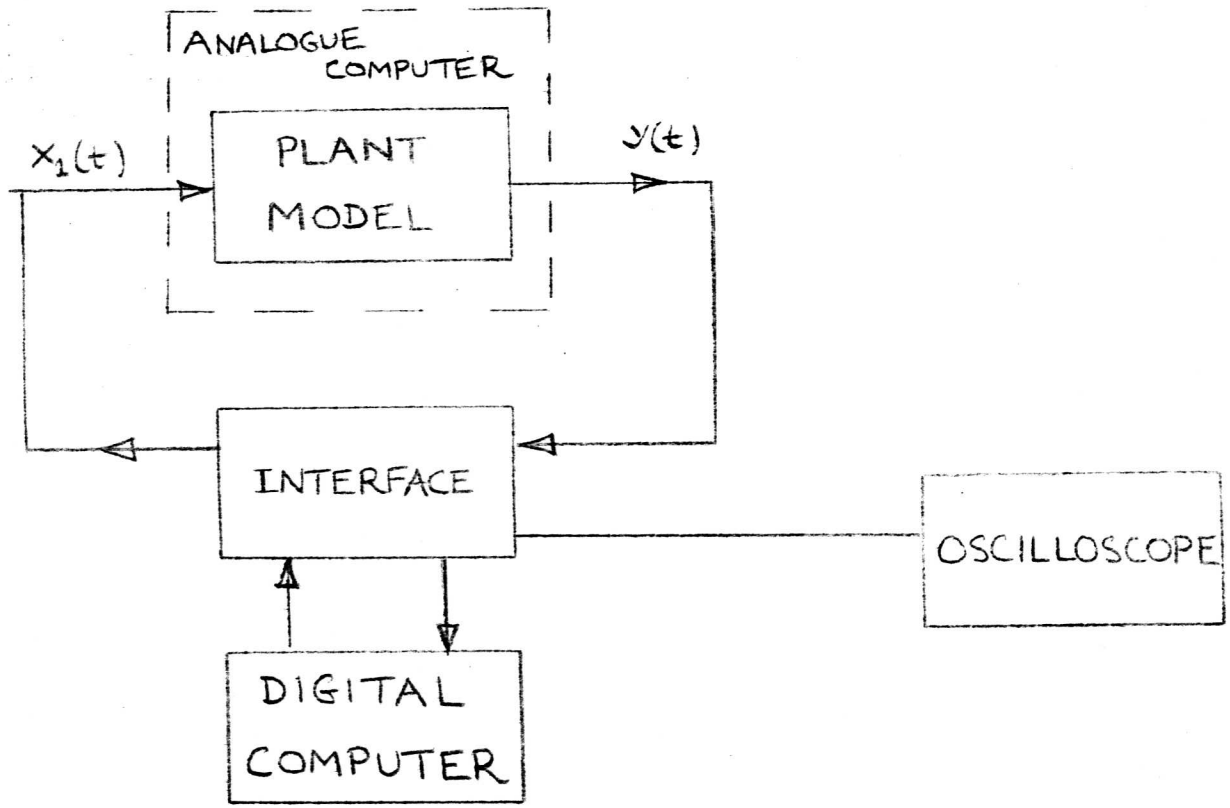


FIGURE 6.13 : CONFIGURATION FOR PRBS EXPERIMENT

and-hold mode also. The lags were therefore simulated with summing amplifiers with capacitors connected between the summing junctions and the outputs. The potentiometer, V, enabled the operating point to be varied by varying K_0 between ± 1 machine unit.

From Figure 6.11 we see that

$$p_1 = -K^2 - K$$

$$p_2 = K$$

and $$p_1 + p_2 = -K^2$$

so that the model of the previous section is simulated for the values $c = m = 1$, resulting in the performance curves shown in Figure 6.8 by unbroken lines.

6.4(b) The Tests Carried Out

Five separate tests were carried out:

(i) A static test to verify that the model was behaving correctly. For this test, using the arrangement of Figure 6.11 on its own, K_0 was varied with the other input to amplifier A earthed. The output of the model was shown to match the appropriate performance curves of Figure 6.8.

(ii) A dynamic test using the system of Figure 6.12 with $T_1 = T_2$. This gave estimated optimums (i.e. those values of K_0 for which $Z(T) = 0$) as follows:

path 1 only: $\hat{K}_{opt} = 0.5$

path 2 only: constant slope for all K_0

both paths: $\hat{K}_{opt} = 0.$

This verifies that the true optimum is given by the sine-wave method when applied to a plant with only one dynamic path. It was observed that the gradient estimate was very dependent on the accuracy of the timer controlling the integration, which necessitated careful synchronization of the integration period with the period of the sine-wave.

(iii) Two dynamic tests using the system of Figure 6.12 to verify equation (6.16). For the first test $T_1 = 2.4\ T_2$; for the second test $T_2 = 2.4\ T_1$. Various values of f_s from 0.05 to 1.00 were used. The results are plotted on Figure 6.9, and are shown to be in close agreement with the theory.

(iv) A test similar to (ii), but using chain-code perturbation instead of sine-waves. This verified that the true optimum is given when the plant has only one dynamic path.

(v) A test similar to (iii), but using chain-codes. The digital computer program of Chapter 9 was used. The results are plotted on Figure 6.9(a), and verify the theory.

6.5 Conclusions

It has been shown that attempts to optimize plants with multiple dynamic paths using sine-wave or chain-code perturbations will result in false optimums.

Under noise-free conditions, the chain-code system is preferable, but it has been deduced that when noise is present the two methods will give very similar performances. As in Chapter 5, it is not possible to summarize the results concisely. For any particular case, it is necessary to estimate the likely performance of the plant and substitute into the appropriate equations in this chapter. It will then be possible to decide which system is preferable, or indeed if either system is worthwhile in the given situation.

PART III

MULTI-PARAMETER IDENTIFICATION

AND OPTIMIZATION

PART III MULTI-PARAMETER IDENTIFICATION AND OPTIMIZATION

CHAPTER 7

MULTI-PARAMETER IDENTIFICATION SYSTEMS

7.1 Introduction

Up to this point, the analysis in this work has been confined to systems with only one variable parameter, but in practical systems there will usually be more than one such parameter.[59] ~~A43, A119, A114.~~ In this chapter several techniques will be considered for the identification and optimization of multi-parameter systems using pseudo-random sequences. The problem is to analyse the transfer function of a system between each parameter and the performance index, and thereby assess the gain, and hence the gradient of the performance function, for each parameter path. The parameters are then adjusted, as for a single-parameter system, according to some predetermined hill-climbing strategy. The operation may be sequential, in that each parameter transfer function is identified in turn, but ideally a system should be developed in which the parameter loops, or "channels", are operated simultaneously, as this should enable the optimum to be reached in a much shorter time than with a sequential system. Unfortunately, attempts to simultaneously identify the parameter paths lead to errors due to the interaction of the parameter perturbations.

The extent of the interaction is considered in the next section. Methods for compensating for the interaction are developed in Section 7.3, and in Section 7.4 methods of avoiding the interaction effects by choosing suitable parameter perturbations are compared. The conclusions are presented in Section 7.5.

7.2 The Interaction Problem

Consider the plant shown in Figure 7.1. It is a two-channel system. The number of output variables is not relevant at this juncture, as they can be combined to give one function, namely the performance index. The simplified model of the performance 'hill' followed by the dynamics will

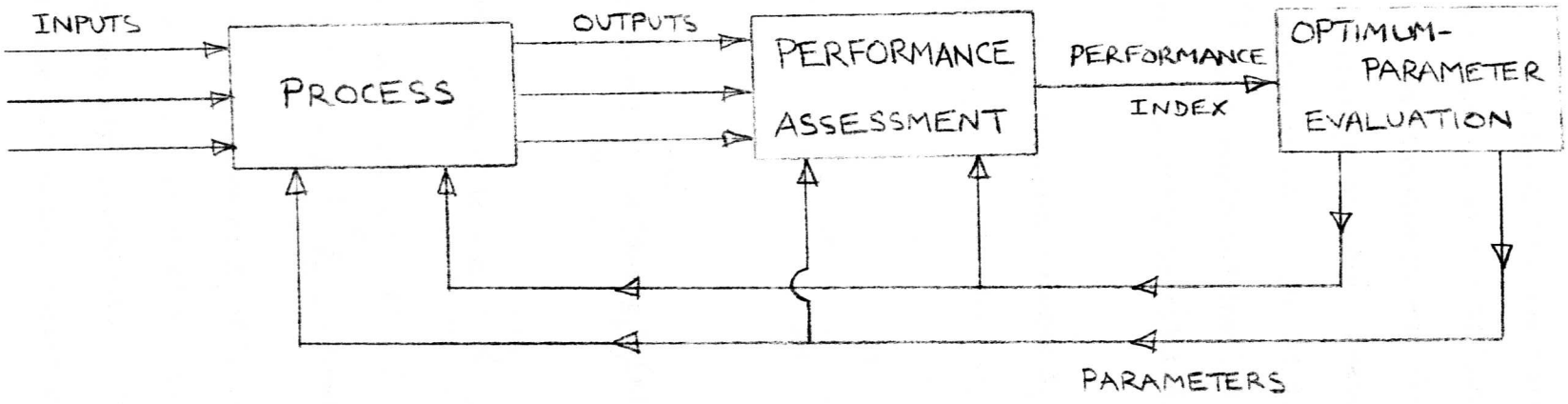


FIGURE 7.1: A TWO-CHANNEL OPTIMIZER

again be used. There will be a transfer function between each parameter and the performance index, and the system can be represented by the model of Figure 7.2, where g_1, g_2 are the performance function slopes in the directions of K_1 and K_2, and $h_1(s)$ and $h_2(s)$ are the respective impulse responses.

Consider the identification system of Figure 7.3, where chain-codes are applied to both inputs simultaneously, and the sum of the outputs is cross-correlated with each parameter to give a measure of the response of the system to impulses applied to each input in turn. Due to the interaction of the two channels[60], both estimated impulse responses will be in error if the input chain-codes are mutually correlated. This will now be analysed mathematically.

The cross-correlation function between c_1 and y is

$$\phi_{c_1 y}(\tau) = \frac{1}{T}\int_o^T c_1(t-\tau)y(t)\,dt \qquad (7.1)$$

But $y(t) = y_1(t) + y_2(t)$

$$= \int_o^\infty c_1(t-s)\,g_1h_1(s)\,ds + \int_o^\infty c_2(t-s)g_2h_2(s)\,ds$$

by convolution.

Therefore

$$\phi_{c_1 y}(\tau) = \frac{1}{T}\left\{\int_o^T c_1(t-\tau)\int_o^\infty c_1(t-s)g_1h_1(s)\,ds\,dt + \int_o^T c_1(t-\tau)\int_o^\infty c_2(t-s)g_2 h_2(s)\,ds\,dt\right\}$$

$$= g_1\int_o^\infty \phi_{c_1c_1}(\tau-s)\,h_1(s)\,ds + g_2\int_o^\infty \phi_{c_1c_2}(\tau-s)h_2(s)\,ds \qquad (7.2)$$

Using the approximation of equation (A1.1.3) gives

$$\phi_{c_1 y}(\tau) = \frac{N+1}{N}\lambda g_1h_1(\tau) + g_2\int_o^\infty \phi_{c_1c_2}(\tau-s)h_2(s)\,ds \qquad (7.3)$$

Hence $\phi_{c_1 y}(\tau)$ is a measure of $h_1(\tau)$, but is in error due to a term dependent on the cross-correlation of the two chain-codes. The analysis for $\phi_{c_2 y}(\tau)$ is clearly exactly equivalent. It is well known[60] that no two m-sequences of the same period and clock-time can have zero mutual correlation for all shifts, so the error terms can never be zero under these circumstances.

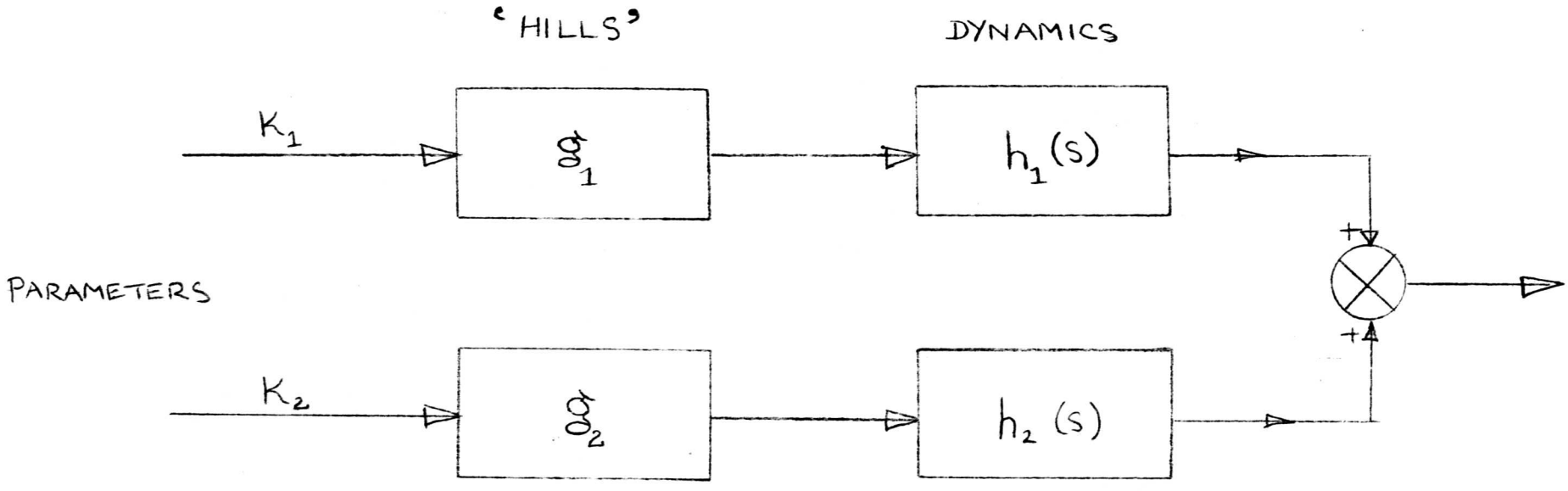


FIGURE 7.2: MODEL OF A TWO-CHANNEL SYSTEM

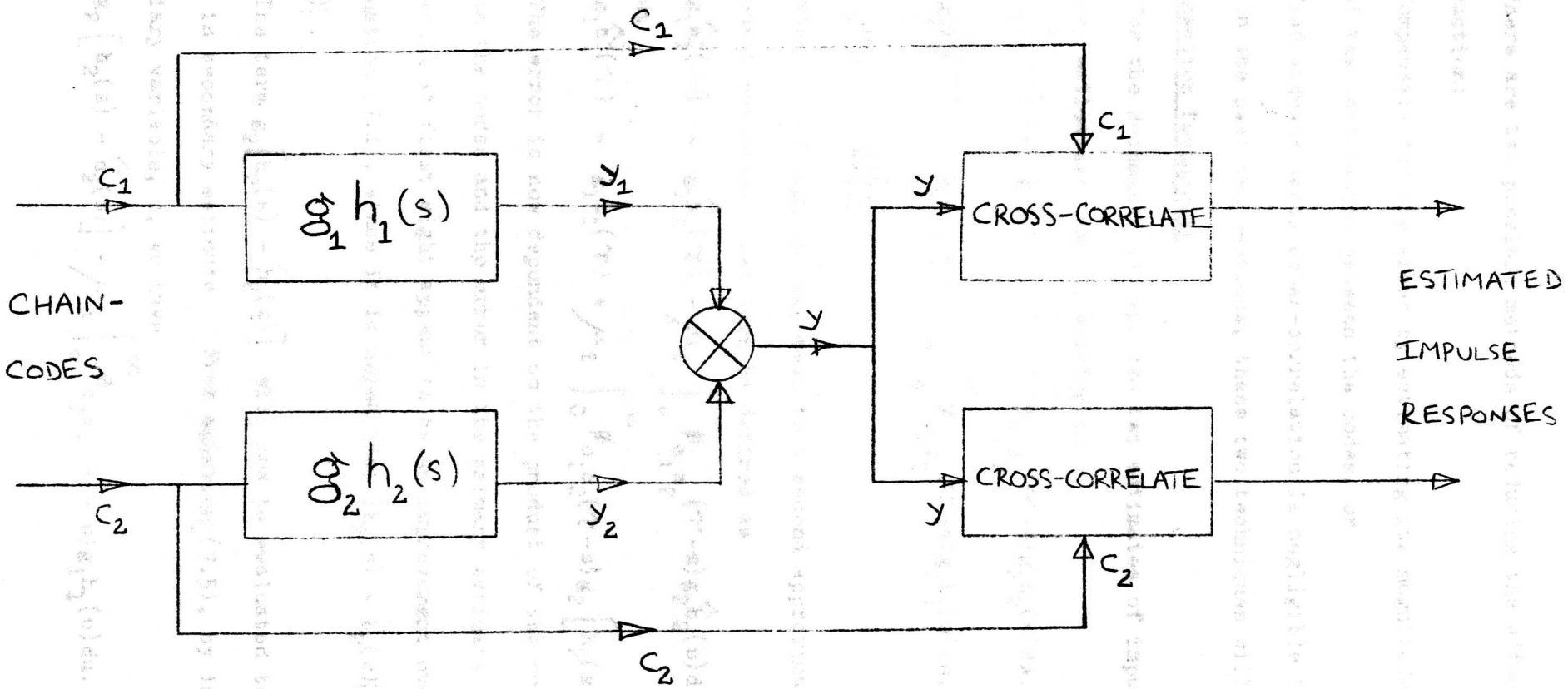


FIGURE 7.3: TWO-CHANNEL IDENTIFICATION SYSTEM

There are two possible methods of reducing the effects of the interaction:

(i) Compensate for the error by estimating its magnitude from a knowledge of the correlation between the codes, or

(ii) find signals whose cross-correlation is negligible for all shifts.

In the next two sections, these two techniques will be examined.

7.3 Compensation Techniques

For the 2-channel system, the two estimates of impulse response (the 'primary estimates') can be written as

$$g_1\hat{h}_1(\tau) = g_1h_1(\tau) + \mu_i \int_0^\infty \phi_{c_1c_2}(\tau-s)g_2h_2(s)ds \qquad (7.4)$$

$$g_2\hat{h}_2(\tau) = g_2h_2(\tau) + \mu_i \int_0^\infty \phi_{c_2c_1}(\tau-s)g_1h_1(s)ds \qquad (7.5)$$

where $\mu_i = \frac{N}{(N+1)\lambda}$

Estimates of the impulse responses to a second approximation (the 'secondary estimates') can then be defined as

$$g_1\hat{h}_1^1(\tau) = g_1\hat{h}_1(\tau) - \mu_i \int_0^\infty \phi_{c_1c_2}(\tau-s)g_2\hat{h}_2(s)ds \qquad (7.6)$$

so that

$$g_1\hat{h}_1^1(\tau) = g_1h_1(\tau) + \mu_i \int_0^\infty \phi_{c_1c_2}(\tau-s)g_2\left[h_2(s)-\hat{h}_2(s)\right]ds \qquad (7.7)$$

The error is now dependent on the product of the cross-correlation between the codes and the error in the primary estimate for the second channel. This would at first sight appear to be an improvement over the estimate given by equation (7.4), since it is hoped that $\left|h_2(s) - \hat{h}_2(s)\right|$ is smaller than $\left|h_2(s)\right|$.

The term $g_2\left[h_2(s) - \hat{h}_2(s)\right]$ will now be evaluated to ascertain the error in secondary estimate. From equation (7.5), by introducing an auxiliary variable, u, we have

$$g_2\left[h_2(s) - \hat{h}_2(s)\right] - \mu_i \int_0^\infty \phi_{c_2c_1}(s-u)g_1h_1(u)du \qquad (7.8)$$

so that equation (7.7) gives

$$g_1\hat{h}_1^1(\tau) = g_1h_1(\tau) - \mu_i^2 \int_o^\infty \phi_{c_1c_2}(\tau - s) \int_o^\infty \phi_{c_2c_1}(s-u)\, g_1h_1(u)\, du\, ds \quad ..(7.9)$$

so that the error term depends on the auto-correlation function of the cross-correlation between the codes. Equation (7.9) can be expanded to give

$$g_1\hat{h}_1^1(\tau) = g_1h_1(\tau) - \frac{\mu_i^2}{T^2} \int_{s=0}^\infty \int_{u=0}^\infty \int_{t_1=0}^T \int_{t_2=0}^T c_1(t_1-\tau)c_2(t_1-s)$$

$$\times\ c_2(t_2-s)c_1(t_2-u)\ g_1h_1(u)\ dt_2\ dt_1\ du\ ds$$

$$= g_1h_1(\tau) - \frac{\mu_i^2}{T} \int_{u=0}^\infty \int_{t_1=0}^T \int_{t_2=0}^T c_1(t_2-u)\ c_1(t_1-\tau)$$

$$\times\ \phi_{c_2c_2}(t_1-t_2)\ g_1h_1(u)\ dt_2\ dt_1\ du \quad .....(7.10)$$

Now $\phi_{c_2c_2}(t_1-t_2) \simeq \frac{1}{\mu_i}\ \delta(t_1-t_2)$ .....(7.11)

Substituting in equation (7.10) and integrating with respect to t_1 gives

$$g_1\hat{h}_1^1(\tau) = g_1h_1(\tau) - \frac{\mu_i}{T} \int_{u=0}^\infty \int_{t_2=0}^T c_1(t_2-u)\ c_1(t_2-\tau)\, g_1h_1(u)\, dt_2\ du$$

$$= g_1h_1(\tau) - \mu_i \int_o^\infty \phi_{c_1c_1}(\tau-u)\ g_1h_1(u)\ du$$

But $\phi_{c_1c_1}(\tau - u) \simeq \frac{1}{\mu_i}\ \delta(\tau - u)$

so that

$$g_1\hat{h}_1^1(\tau) \equiv 0 \quad .....(7.12)$$

Therefore the error in estimating $h_2(s)$ before substituting in equation (7.6) is such as to exactly cancel out the required term, $g_1h_1(\tau)$. (The d.c. bias estimation process discussed in Chapter 3 has not been considered here, but its inclusion does not alter the inherent failure of the compensation technique.)

The result of equation (7.12) was verified by using the process control computer coupled to a plant model simulated on the analogue computer, where it was observed that:

(i) without compensation, the estimates of the impulse responses obtained by simultaneous cross-correlation were grossly in error:

(a) On both channels, if the plant characteristics were of the same order of magnitude in each channel, or

(b) on one channel only, if the other channel was dominant.

(ii) with the compensation mooted above, the impulse response estimates vanished completely.

It is thus concluded that this method of compensation is of no use.

An alternative scheme would be to estimate the value of $y_2(t)$, say, from the estimate of $h_2(s)$ by convolution, subtract this from $y(t)$ to estimate $y_1(t)$, and then correlate this with $c_1(t)$ to obtain a secondary estimate of $h_1(\tau)$. Unfortunately, this technique will clearly fail in the same manner as that proposed above. It can be concluded that any similar techniques for compensation will also fail. The explanation for these failures is that the method of cross-correlation involves estimating a number of ordinates of impulse responses from a number of equations. For a single-channel there are sufficient equations if one value is assumed, such as that the impulse response is zero at the end of the period of the code (see Chapter 4), but for more than one channel there is not enough information to estimate the impulse responses with any degree of accuracy.

As all the compensation methods have failed, other schemes for multi-parameter identification will now be considered.

7.4 Uncorrelated Perturbation Signals

The problem of searching for uncorrelated pseudo-random perturbation signals will now be investigated. Clearly, if the mutual correlations between parameter perturbations are all zero, the interaction errors will also be zero, and identification will be possible on all channels simultaneously. The settling time of the identification process will then be equal to the longest settling time of the plant, plus one period of the code for the output of the plant to become correct throughout one period

(just as for a single-parameter system the corresponding time is equal to the plant settling time plus one period of the code, which equals at least twice the plant settling time).

It has already been noted that all m-sequences are mutually correlated if they have the same clock-time and length.[60] A great deal of work has been undertaken by engineers and mathematicians[31,61-65] to find uncorrelated signals based on four types of pseudo-random sequence (m-sequences, Hall sequences, quadratic-residue codes and twin-prime sequences), and some of the results are now summarized:

(i) Consider the 2-channel system once again. Briggs and Godfrey[31] have observed that the cross-correlation between codes is negligible if the codes have the same clock-times but different lengths, M and N bits, providing averaging is performed over MN bits, if M and N are coprime (as is often the case, for example if M and N are 31 and 63, or 63 and 127). Such long averaging times make this technique unacceptable in practice. Changing one clock-time relative to the other does not improve the situation[62]. In particular, codes of the same period but different clock-times have large mutual correlation.

(ii) Another possibility for a 2-channel system is to use a code C followed by its amplitude-inverse on one channel, and two periods of C (or of any other code of the same period as C) on the other. This clearly results in zero correlation for an averaging time of twice the basic period. Using C and a stretched version of C with twice the bit-length also has this correlation property. Both these methods are possible, but have two drawbacks:

(a) They result in very long codes when applied to more than two channels; for N_C channels, the length is $2^{(N_C-1)}.T$. Where T is the period of the basic code.

(b) They require a certain amount of logic to generate the modified codes from the basic code.

(iii) Codes generated by applying a Hadamard matrix[66] to a basic code have the required auto- and cross-correlation functions[61]. (A Hadamard matrix is an orthogonal matrix with all terms ± 1). The length of the resulting codes is $N_C.T$. As T has to be chosen to exceed the longest settling time of the plant, there will be some redundancy in the identification of the other channels if they have shorter settling times. For use with an on-line computer, where the codes can be generated internally, the method is straightforward in execution; without a computer, a certain amount of logic hardware is required to generate the modified codes from the basic code.

(iv) Codes with more than two levels can be used. A 3-level code, of length $(3^n - 1)$ bits where n is the number of shift-register stages, can be generated fairly simply by modulo-3 hardware. The auto-correlation function of a 3-level code has 2 'spikes' per period, and therefore has to be twice as long as the plant settling time for one channel. Unfortunately, the correlation between 3-level codes of the same period is no smaller than that between 2-level codes of the same period.

(v) A combination of code-types is more primising. For example, a 3-level m-sequence is antisymmetric, that is, the second half is the amplitude-inverse of the first half, so that any signal whose period is half that of the 3-level sequence will be uncorrelated with it (as for the two-level system of paragraph (ii)). In particular, the second signal can be a 2-level code. Such a system is more complicated to set up experimentally than a system using codes of the same modulus. It also has no advantage over other systems as far as identification time is concerned, and cannot be extended to more than two channels. A combination of 3-level codes can be applied to a system of any order by 'nesting' the codes, but the total length of $2^{(N_C-1)}.T$ makes the method impractical.

(vi) The final technique to be considered, and the one favoured for experimental work, makes use of that part of the auto-correlation function of a long 2-level code for which the impulse response of the plant can be considered zero. The length of the code is given by

$$T = \sum_{j=1}^{N_c} T'_j$$

where T'_j is the settling time of the j-th channel. The code is used to perturb all the parameters, with a phase-shift between each parameter perturbation, such that the j-th parameter is perturbed by a signal that is shifted an amount T'_{j-1} from the (j-1)-th parameter perturbation. The result of cross-correlating the output with the code gives an estimate of all the impulse responses in phase-shifted form. Figure 7.4 shows a typical result for a 4-channel system. The scheme will be referred to as the 'shifted codes' system.

Williams has given experimental results that show that using a code c and a code derived from c by inverting every third bit gives slightly better results for a 2-channel system than using shifted codes, which in turn gives better results than using c and a code derived by taking c followed by its amplitude-inverse. He also gives purely empirical results[63] for a multi-channel system using codes derived from a Hadamard matrix, where the d.c. bias is known beforehand and need not be estimated by the identification system. However, there is no evidence to show that the Hadamard matrix method is appreciably superior to the shifted codes system, and there are two definite advantages of the shifted code system:

(i) The identification time is no longer than that required by any of the other systems mentioned above, and because of the ability to adjust the phase intervals between the perturbation signals, the overall length of the code can be made smaller than that needed for the Hadamard matrix technique, for example, if the settling times of the channels are not all the same.

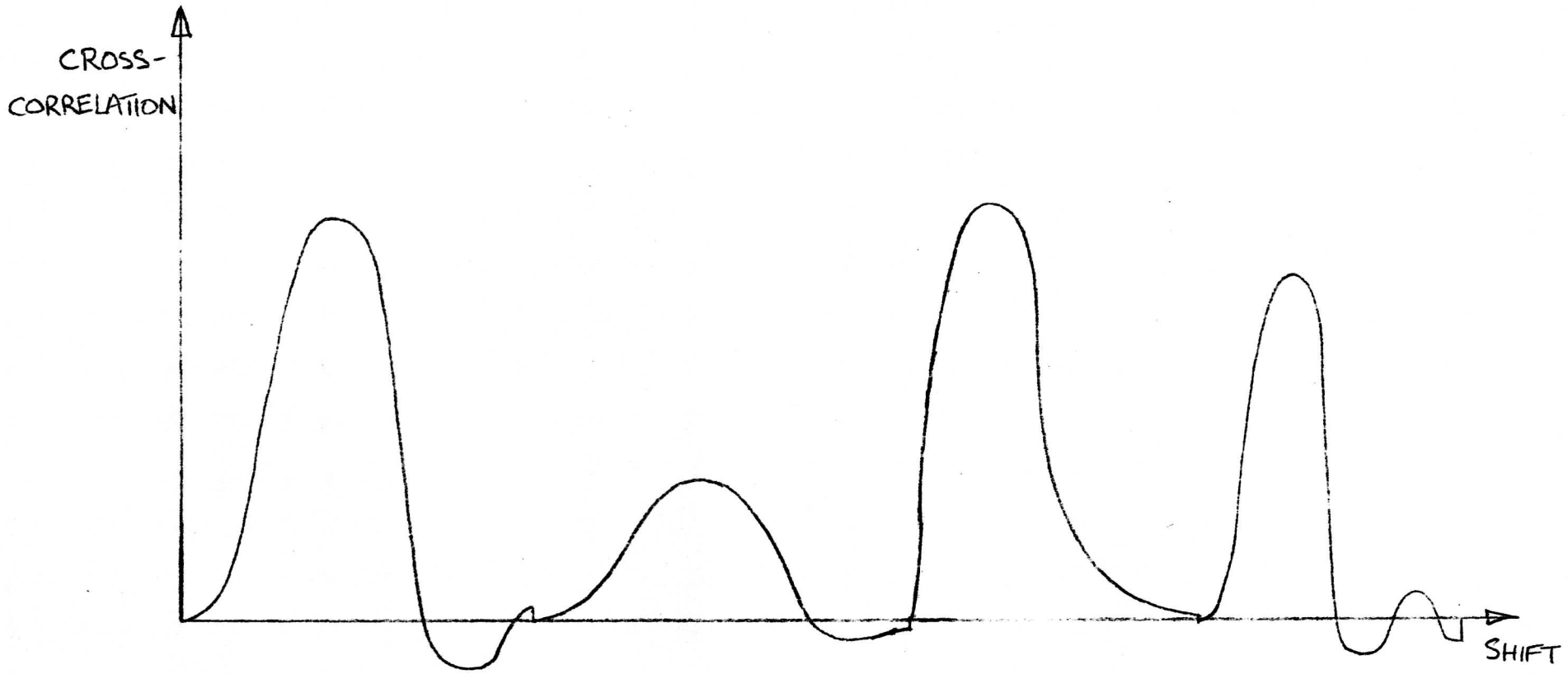


FIGURE 7.4: TYPICAL IMPULSE RESPONSE ESTIMATES FOR A FOUR-CHANNEL PLANT

(ii) It is simple to implement in practice.

The shifted code technique might be referred to as a 'quasi-simultaneous' identification scheme: the identification is performed for all channels at the same time, but is not in the category of truly simultaneous systems, for which the period of the perturbations would be of the order of the longest settling time of the plant. It is, however, much more rapid than a system which identifies each parameter path separately.

In the next chapter, the development and performance of a shifted codes system is examined, and Chapter 9 outlines the structure of a digital computer program to execute the system on-line.

7.5 Conclusions

It has been shown that the estimates of the impulse responses for a multi-parameter plant using two-level chain-codes of the same length as input perturbations will be seriously in error due to interaction between the channels. Techniques for compensating for the interaction are shown to fail. Codes designed to reduce or eliminate the interaction must therefore be used, and several combinations of codes have been compared. It is concluded that a system with phase-shifted versions of the same code as input perturbations is preferred, in which case the code must be at least as long as the sum of the settling times of the channels.

CHAPTER 8

THE DEVELOPMENT OF MULTI-PARAMETER SYSTEMS USING SHIFTED CODES

8.1 Introduction

In the last chapter is was concluded that a system based on the use of phase-shifted versions of a binary chain-code was the most suitable for multi-parameter identification. The practical development of such a system will now be examined, for uses with and without a process-control computer.

8.2 Assessment of the D.C. Bias

The shifted codes technique is a straight forward extension of the system of Chapter 3. As for the single-parameter system, the d.c. has to be assessed. All d.c. bias on the parameter values, or inherent in the plant, will appear as a bias on the performance index, and so there is only one d.c. level to be assessed for all the parameters. There are two ways of assessing it:

(i) from the last few bits of the cross-correlation, or

(ii) from a few bits at the end of each estimated impulse response.

For the first method, the relative phase-shifts of the perturbations can be arranged to give one long gap between two of the impulse responses; for the second method, the gaps can be distributed so that there is a small one between each impulse response. In either case, running summers can be used. For the first method, the form of the running summer can be deduced from equation (3.2); for the second, a running summer will have to be designed to generate the sum of bits of the chain-code situated in small groups along it.

8.3 Assessment of the Step Responses

The non-d.c. terms of the identification system will now be examined. Denoting the p_q-th ordinate of the step response of the q-th channel, as estimated without the d.c. correction term, by $\hat{S}^1_q(p_q)$, gives, from equation (3.1) and Figure (8.1),

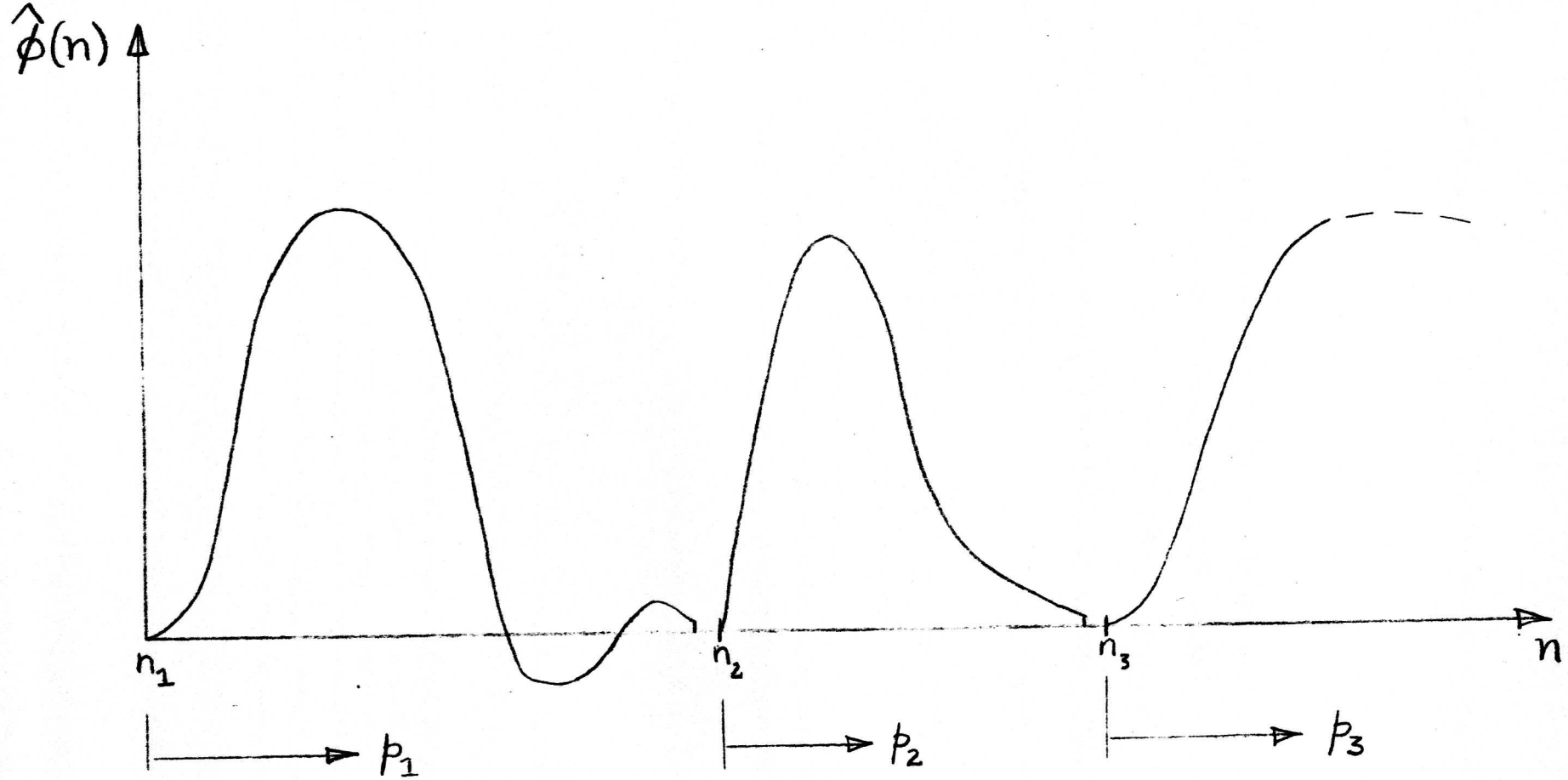


FIGURE 8.1: CROSS-CORRELATION FUNCTION FOR A MULTI-PARAMETER SYSTEM

$$\hat{S}_q^1(p_q) = \frac{N}{a(N+1)} \sum_{n=n_q}^{n_q+p_q-1} \hat{\phi}(n) \qquad (8.1)$$

for $0 < p_q \leq n_{q+1} - n_q$, where n_q is the value of n at the start of the q-th impulse response estimate.

Expanding equation (8.1) and referring to Appendix A3.2 gives

$$\hat{S}_q^1(p_q) = \frac{N}{a(N+1)} \sum_{m=1}^{N} \{ y(m)\, \sigma_c(m - \overline{n_q + p_q - 1} : m - n_q) \} \qquad (8.2)$$

So that the step response can be estimated by multiplying each of the last N values of y(m) by an appropriate value of the running sum over p_q bits of c(m), and summing these products.

Alternatively, the 'differencer' method of equation (3.6) can be considered. The latest estimate of step response is then assessed from that made one clock-interval earlier. The estimate of the q-th channel at the j-th clock-interval is

$$\hat{S}_q^1(p_q, j) = \hat{S}_q^1(p_q, j-1) + \frac{1}{a(N+1)} \left[y(j) - y(j-N) \right] \sigma_c(m - \overline{n_q + p_q - 1} : m - n_q) \qquad (8.3)$$

If the d.c. is assessed from the last k bits of the cross-correlation function, the d.c.-corrected step response can be estimated from one of the following equations:

$$\hat{S}_q(p_q) = \frac{N}{a(N+1)} \sum_{n=n_q}^{n_q+p_q-1} \left\{ \hat{\phi}(n) - \frac{1}{k} \sum_{\ell=N-k}^{N-1} \hat{\phi}(\ell) \right\} \qquad (8.4)$$

$$\hat{S}_q(p_q) = \frac{1}{a(N+1)} \sum_{m=1}^{N} \left\{ y(m) \left[\sigma_c(m - \overline{n_q + p_q - 1} : m - n_q) - \frac{p}{k} \sigma_c(m+1 : m+k) \right] \right\} \qquad (8.5)$$

$$\hat{S}_q(p_q, j) = \hat{S}_q(p_q, j-1) + \frac{1}{a(N+1)} \left[y(j) - y(j-N) \right] \times \left[\sigma_c(m - \overline{n_q + p_q - 1} : m - n_q) - \frac{p}{k} \sigma_c(m+1 : m+k) \right] \qquad (8.6)$$

8.4 Hardware for Multi-Parameter Identification

The design of the hardware for a multi-parameter identification system when no process-control computer is available will now be considered. In such circumstances, equation (8.4) is not easily evaluated practically, owing to the cross-correlation and storage elements required. To implement

equation (8.5) requires N multiplications, and the generation of the running averages, either by operating continuously on the chain-code or by reading from storage elements charged at the start of the experiment. The details are explained in Appendix A2.1. In addition, another running averager[48] can be used at the output of the multiplier in order to give an estimate of step response at every clock-interval. The system then corresponds to a modified form of equation (8.6). Alternatively, equation (8.6) can be evaluated directly by using the 'differencer' device of Section 3.3(b), in which $\left[y(j) - y(j-N)\right]$ is calculated using an N-word storage device.

Two ways of evaluating all the step responses using the differencer method are shown in Figures 8.2 and 8.3. It will be noted that only one differencer is required for the whole system, whereas the second running averager method requires a second running averager for each channel. The differencer method is therefore to be preferred. In Figure 8.2, a multiple running summer is used. This generates the required N_C running sums from the chain code. Each of these is then modified by the output of the running summer used for bias-evaluation, and the resulting signals are multiplied by the output of the differencer in N_C multipliers, the outputs of which are integrated and sampled-and-held, with the sampler operating at the clock-frequency. The parameter perturbations are obtained from the chain-code by appropriate delay circuits; the easiest way is to tap-off appropriate outputs of the chain-code generator stages and combine them with logic elements to give the required delayed versions (see Appendix A1.1). Alternatively, the chain-code generator could be replaced by an electro-mechanical storage device[49] of the capacitor-drum type, on which the whole code is stored at the start of the experiment, or by an electronic device based on a shift-register connected in a 'ring', the stages of which are present at the start of the experiment. With either device, any delayed versions of the code can be read-off from the appropriate points.

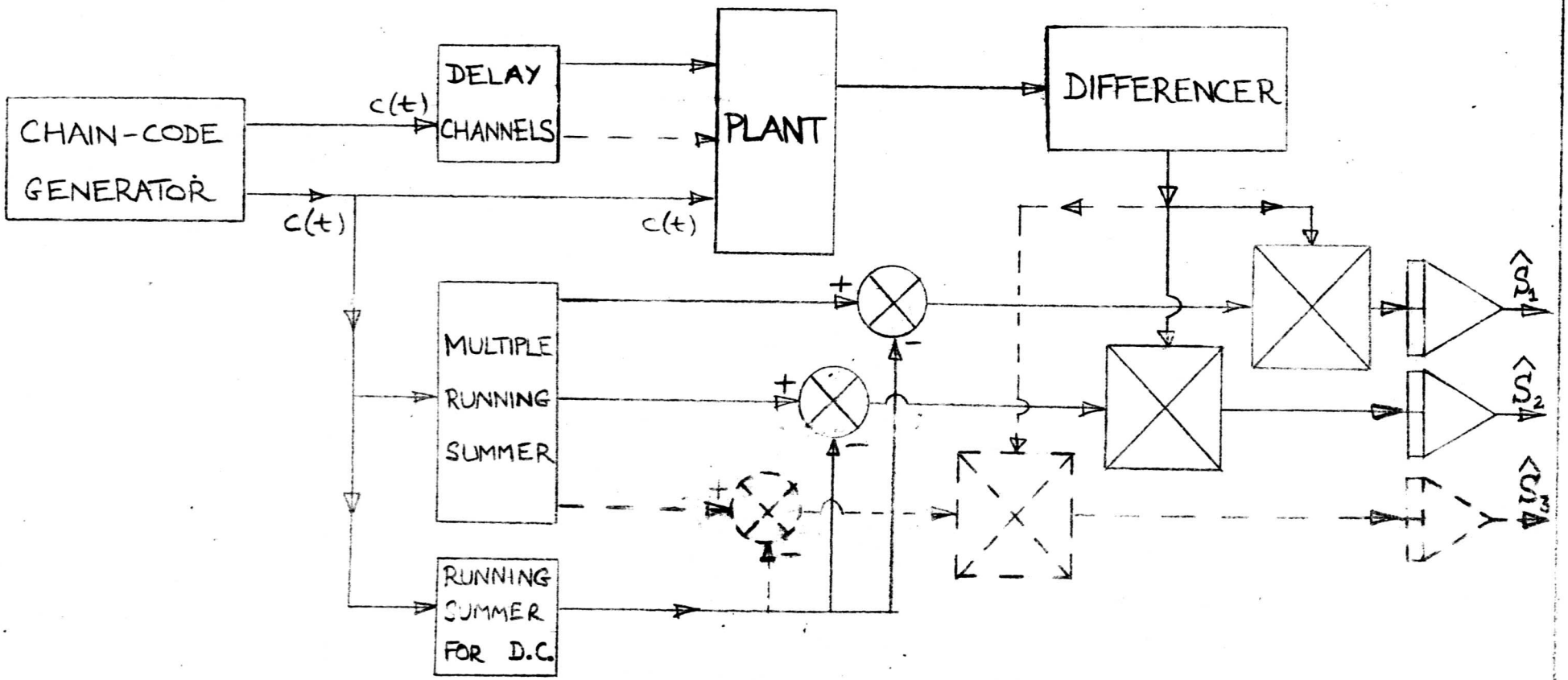


FIGURE 8.2: MULTI-PARAMETER GAIN ESTIMATION USING 'INDEPENDENT HARDWARE'

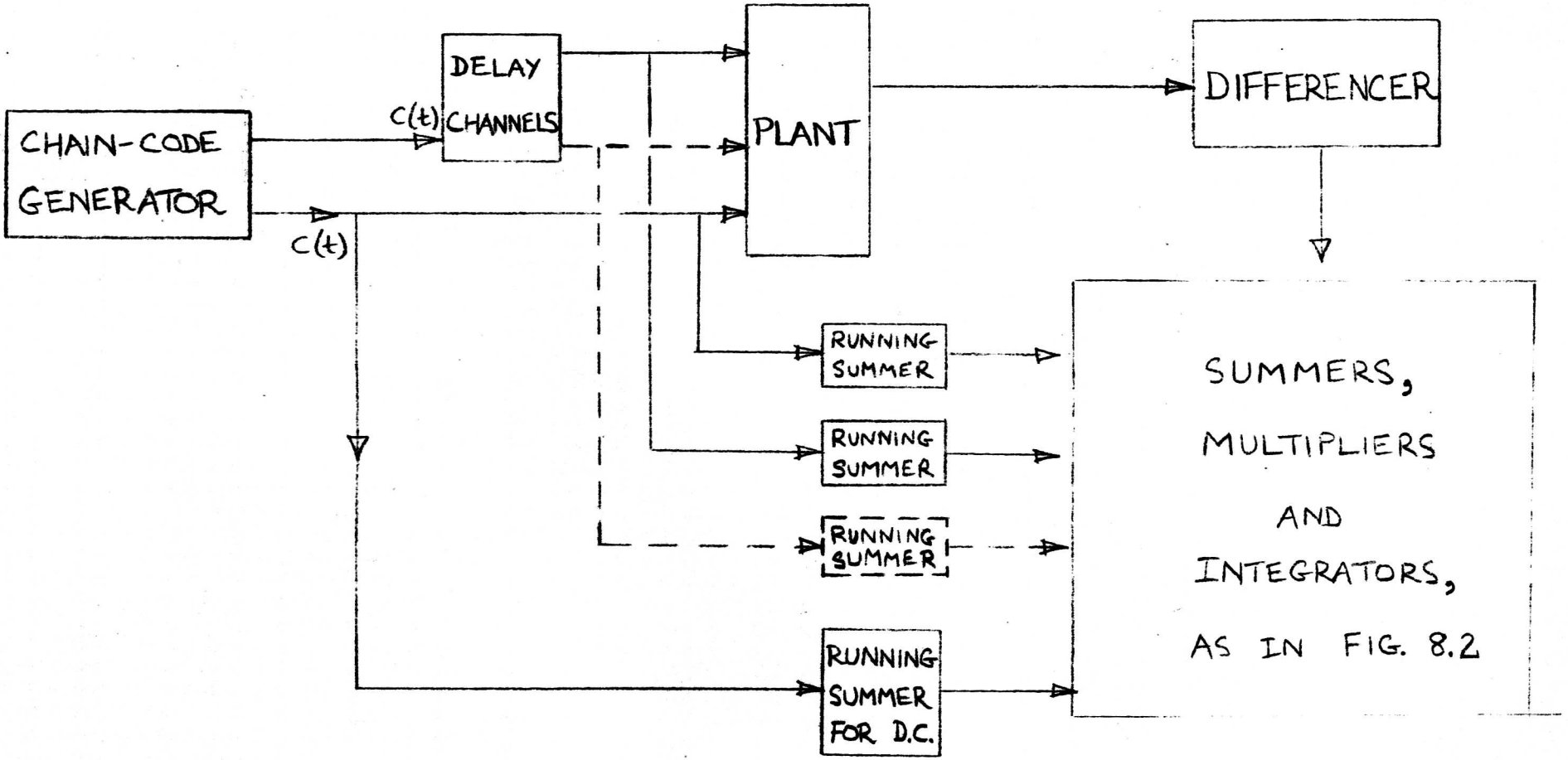


FIGURE 8.3: ALTERNATIVE MULTI-PARAMETER GAIN ESTIMATION PROCESS

In Figure 8.3, the multiple running summer is replaced by N_C running summers of the single-output type. The input of each running summer is taken from the appropriate output of the delay generator. The system is otherwise identical to that of Figure 8.2, but has the advantage that the running summers are all similar units, differing only in the summation periods required.

It is concluded that a system for multi-parameter identification can be built up using ordinary running summers and a differencer, in much the same way as for a single-parameter system.

8.5 Computer Software for Multi-Parameter Identification

Methods of multi-parameter identification using a digital process-control computer will now be evaluated. There are two alternative approaches:

(i) use of the running summer technique of equation (8.6), or

(ii) use of the cross-correlation techniques of equation (8.4).

The advantage of the running summer technique is that the running sums can all be evaluated and stored at the start of the program. The disadvantages are:

(i) The storage needed for the running sums is $(N.N_C + 1)$ words if they are all different. (In practice some may be duplicated, with different phases). This may be too large to be practicable.

(ii) Dynamic compensation (see Chapter 9) cannot be used, as all the points on the step response estimates must be known for this, and to obtain them from equation (8.6) is too cumbersome.

(iii) The manipulation of equation (8.6) for a system with many parameters is more complex than the evaluation of the step responses by integration of the cross-correlation function.

It is concluded that the cross-correlation technique is preferable when using a digital computer, especially if there are many parameters and if storage is at a premium.

8.6 Conclusions

Methods for identifying multi-parameter systems using the shifted codes technique are available for uses with and without a process-control computer. Purpose-built hardware can be developed, using conventional running summers, multipliers and a 'differencer', or a computer program can be developed, using the straight forward cross-correlation technique. Such a program is the subject of the next chapter, in which a multi-channel optimization routine is derived for on-line application.

CHAPTER 9. AN ON-LINE COMPUTER PROGRAM FOR MULTI-PARAMETER OPTIMIZATION

9.1 Introduction

A program for the optimization of a multi-parameter process using the shifted codes technique has been written for use on an on-line process-control computer. The program falls into two parts:

(i) the identification process,

and (ii) the optimization process.

In this program the optimization strategy is a simple one: each parameter is changed at each clock interval by an amount proportional to the estimate of the appropriate slope of the performance function. This results in reasonably fast optimization with small overshoot, but the constants of proportionality have to be fixed by the operator at the start of the program. (Other optimization strategies are considered in Chapter 10). This chapter describes the development of the program, and the results obtained from applying it to various plant models.

9.2 The Multi-Channel Optimization Program

The Optimization program is written for the G.E.C. 92 process - control computer, with facilities as outlined in Appendix A9.1. The operation of the program is illustrated for one channel in Figure 9.1, where 'feedback' denotes the latest parameter value, as determined by the optimizing loop. The chain-code perturbation is added to the feedback inside the computer.

The following items are at the discretion of the operator:

(i) The number of parameters. Two of the six available analogue outputs are used for monitoring; the program has therefore been written to operate with any number of channels up to 4.

(ii) The length of the code, up to 511 bits, and the shift-register feedback connections.

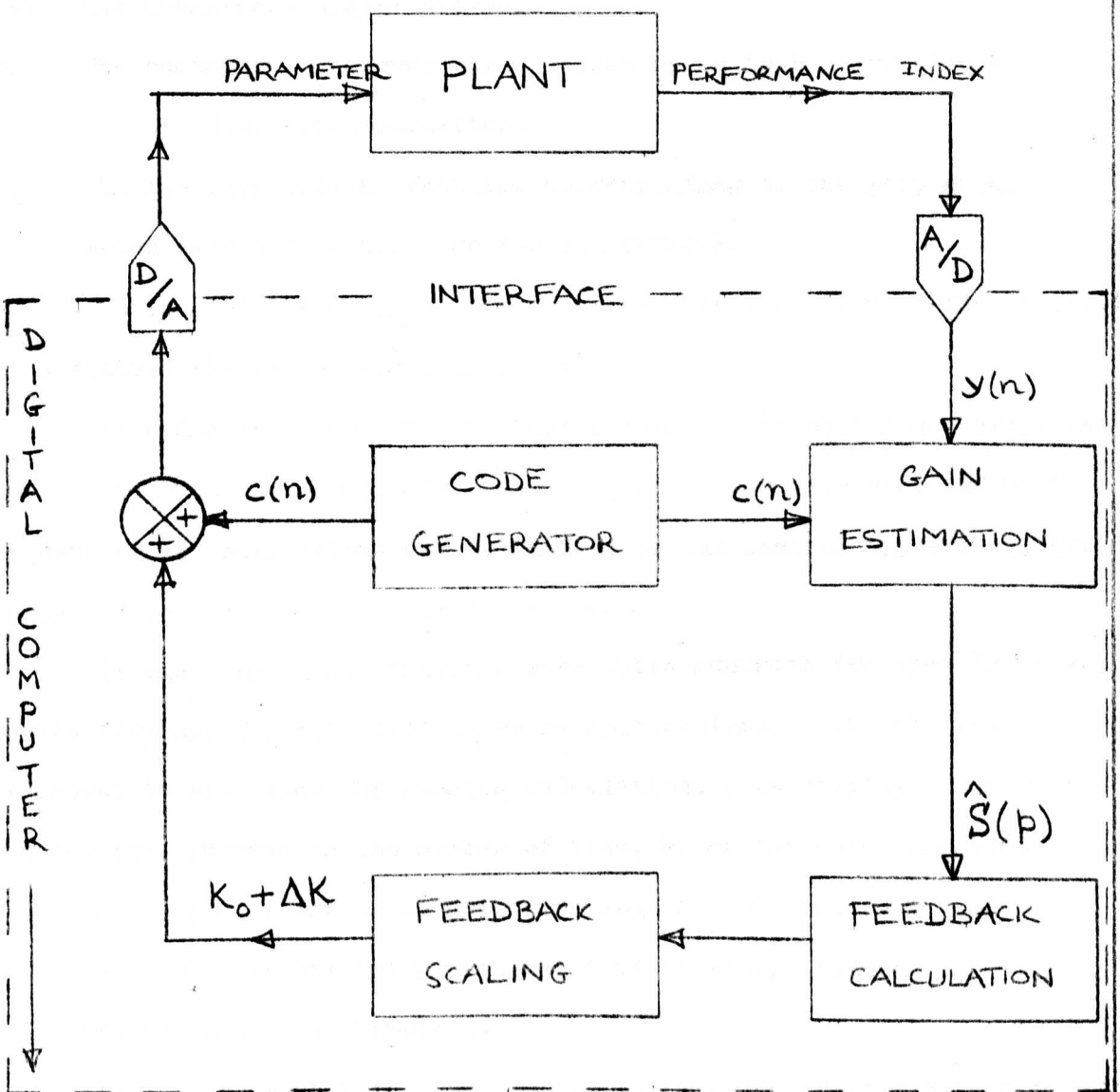


FIGURE 9.1: ONE CHANNEL OF THE ON-LINE OPTIMIZATION PROCESS

(iii) The phase separation between the parameter perturbations. This allows allocation of a large number of bits to channels with long settling times, and a smaller number of bits for faster channels.

(iv) The chain-code clock-interval.

(v) The number of bits over which averaging is to be carried out for the d.c. bias estimation.

(vi) The feedback scaling factors, corresponding to the loop gain, which need not be the same for all channels.

On loading the program, preset values are allocated to the variables which dictate the above characteristics.

In order to change any of these values, it is merely necessary to press buttons on the operator's panel. (For a more sophisticated system, the setting up instructions can be entered on the control typewriter, after making a simple modification to the program).

It was found that 32 points were quite adequate for specifying each impulse response for most hill-climbing applications. The bit interval was chosen to give time for running calculations plus display. It will be noted that increasing the number of bits, N, of the code increases

(i) the length of the calculations during each bit interval, and

(ii) the number of bit intervals that have to elapse before identification is complete.

Hence doubling N more than doubles the minimum identification time attainable (if the settling time of the plant is short, so that the calculation time is critical).

The listing of the program, which is known as OPTIM4, appears in Appendix A9.2, together with a description of the program storage allocation. The object was to write a concise program; a program with slightly shorter run-time could be written, but would be less neat. OPTIM4 uses 3 subroutines:

(i) PRBS, which is a logic routine for generating the next state of a shift-register from the present state, with given feedback connections.

(ii) INPUT, which handles the input, conversion and averaging of the plant output.

(iii) SCOPE, which displays any block of up to 512 words of storage, together with x- and y- axes.

(These programs were all written by the author of this thesis, with the exception of PRBS, which was written by a colleague, John Monk, to whom the author would like to express his thanks.) The listings of INPUT and SCOPE appear in Appendix A9.2.

Flow charts for OPTIM4 and INPUT appear in Figures 9.2 to 9.8. (Details of PRBS and SCOPE are straightforward, and are omitted here.). The programs will now be broken down into blocks for analysis:-

I. The Starting Routine (Figure 9.2)

Having ensured that the interrupts are disabled, certain storage blocks are dynamically allocated, and certain indices are evaluated. This is necessary because variable phase-separation is to be allowed. Certain working stores are then cleared, and the chain-code is generated by repeated use of PRBS. The code is stored in two forms

(i) in CC as ± 1, and

(ii) in CODE as $\pm$01000 (octal)

(Storage could be saved at the expense of a slightly longer run-time by omitting CODE.) The interrupts are then enabled, and control is routed to Display.

II. The Display Routine (Figure 9.3.)

This is a self-contained, continuous loop, which is serviced whenever the running calculations are not being carried out. Exit from the loop only occurs when an interrupt signal is received, and control returns to the loop after the interrupt routine has been serviced. Available options, chosen by breakpoint selection, are:

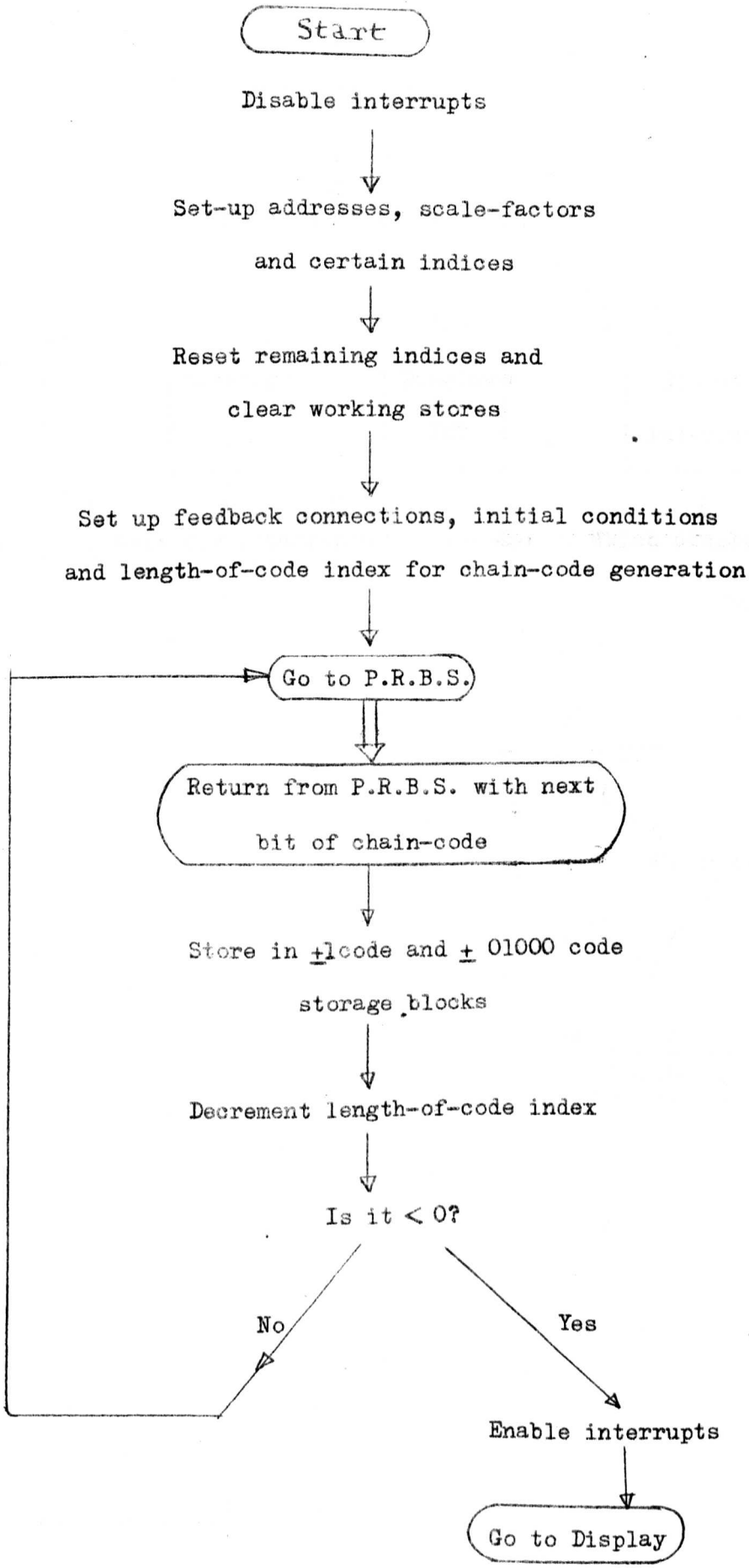


Figure 9.2. Starting Routine

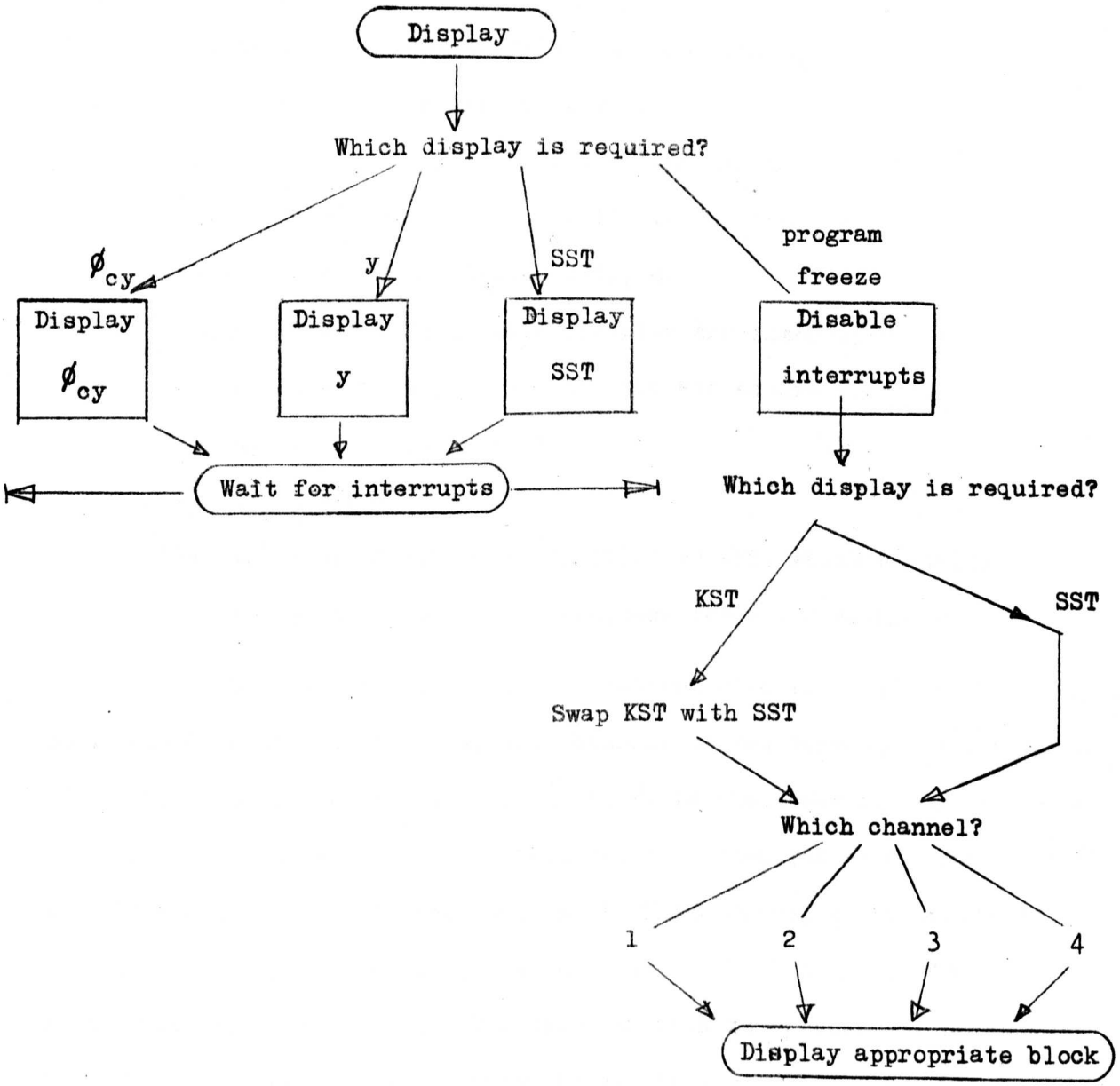


Figure 9.3. Display Routine

(i) display of the cross-correlation function, ϕ_{cy},

(ii) display of the plant outputs, y,

(iii) display of the progress of the optimization, in terms of the past 512 values of gain estimate for the first channel (part of the block labelled SST), or

(iv) freezing the program; this disables the interrupts, so that on-line working ceases and the stores are frozen.

Display options are then

(a) any channel of SST, or

(b) any channel of the block labelled KST, which contains the past 512 values of feedback for all 4 channels.

With the proportional-to-gradient strategy used in this program, each sub-block of KST is the scaled integral of the corresponding block of SST. SCOPE only operates with data blocks in the lower half of the store for simplicity, as addresses are then not more than one word long. As KST and SST cannot both be in the lower half of the store, the contents of KST (in the top half) can be swapped with those of SST (in the bottom half) if display of KST is required. The swap can then be reversed if necessary.

III. The Interrupt Routines. (Figures 9.4 to 9.6)

On receiving a clock-interrupt, control goes to TICK, where the latest value of y is read from the analogue input into the converter. As soon as y has been read, control returns to Display while the conversion takes place, at the end of which an interrupt is generated and control passes to INPUT; INPUT is responsible for averaging 8 consecutive values of y during the bit interval. When the eighth value occurs, control goes to OPTIM4 at location IPDONE; otherwise control returns to Display.

IV. On-Line Routine. (Figure 9.7)

The output to the buffer of code-plus-feedback for each channel is performed by a routine common to all channels, using the indirect addressing

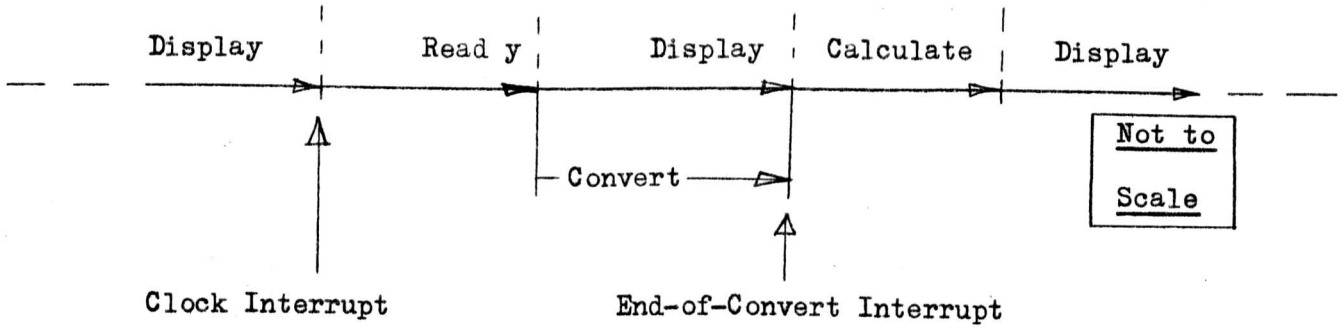


Figure 9.4 Time Sequence for Interrupts

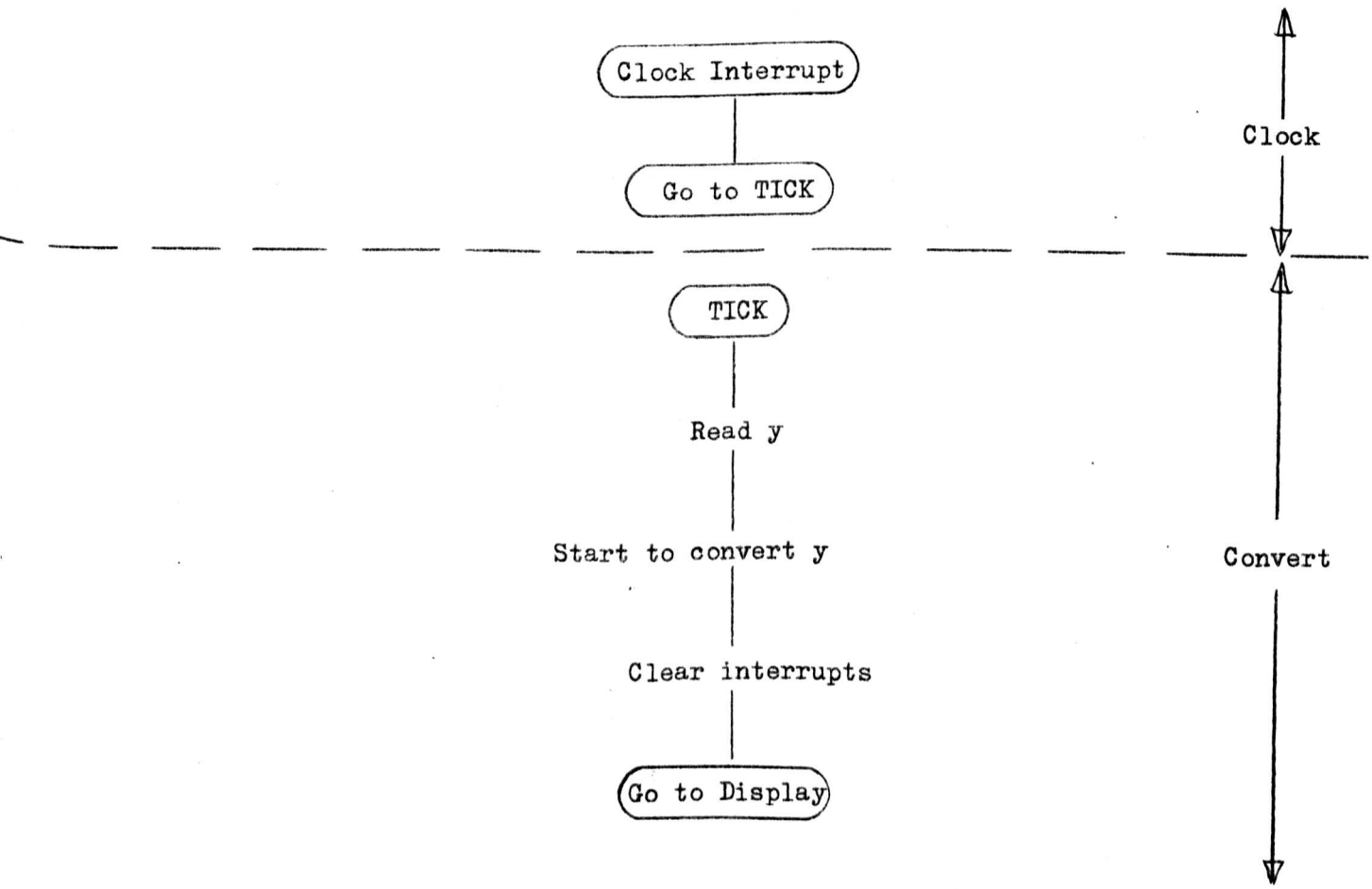


Figure 9.5 Analogue-to-Digital Conversion Routines

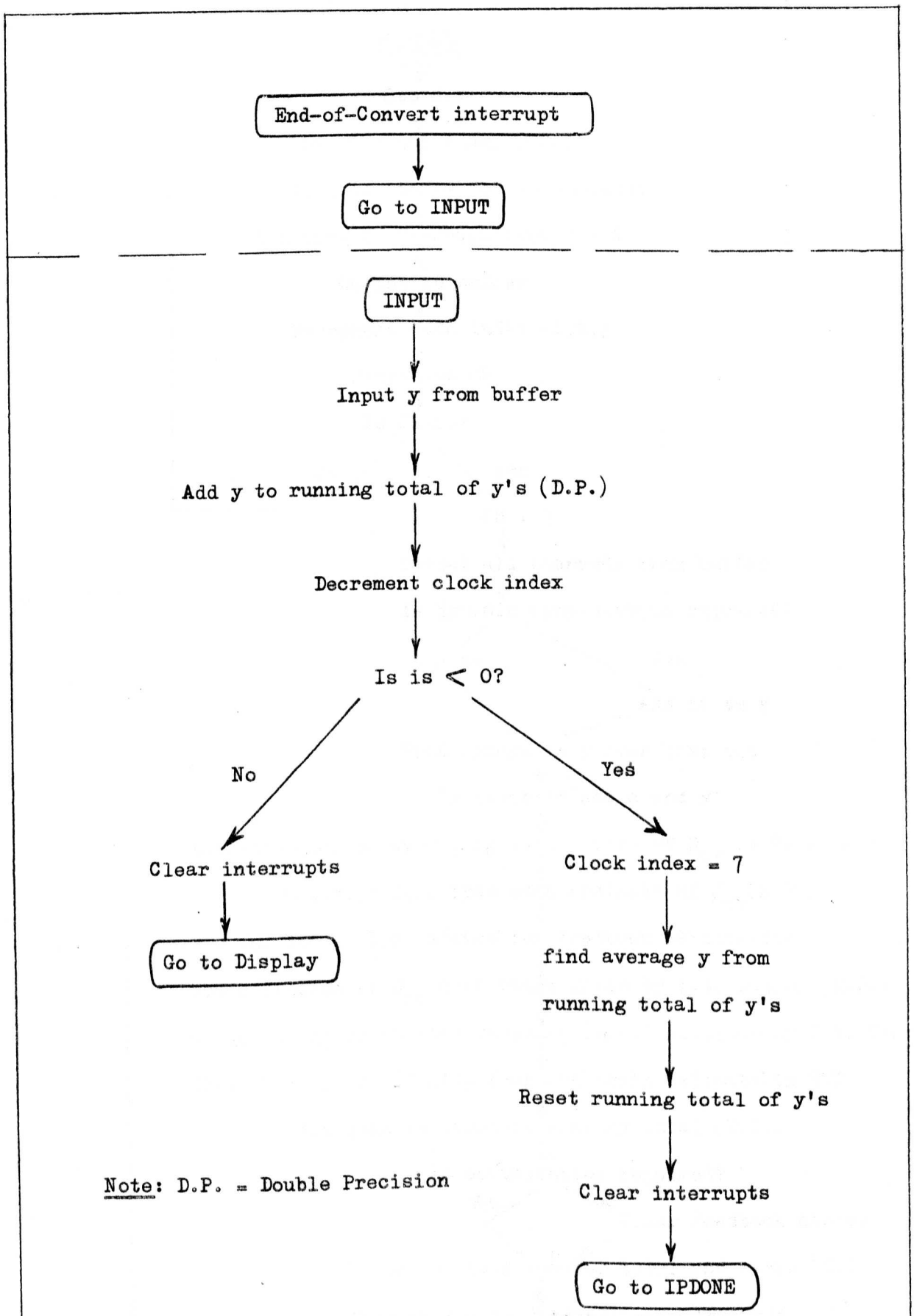


Figure 9.6 Input Routines

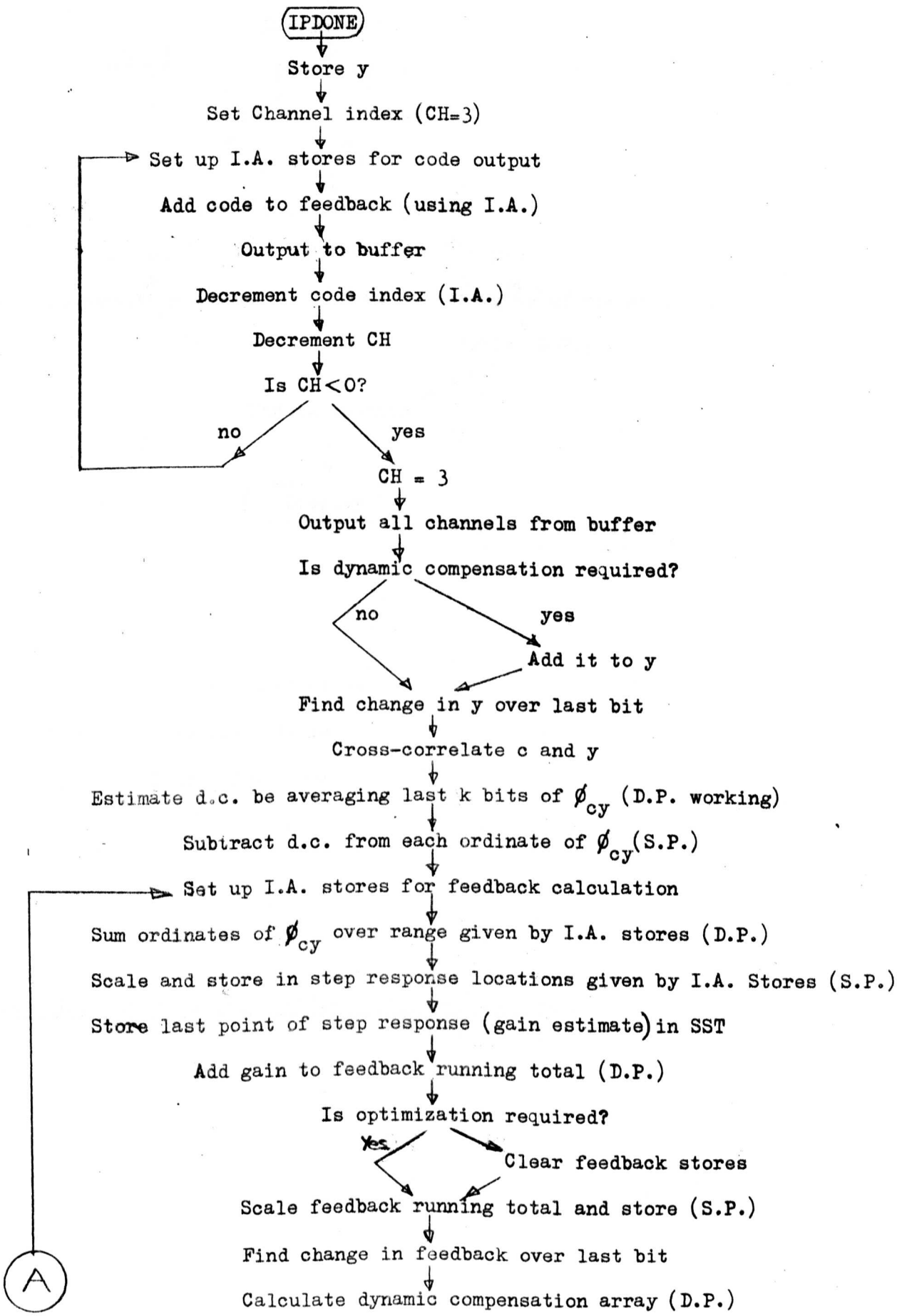
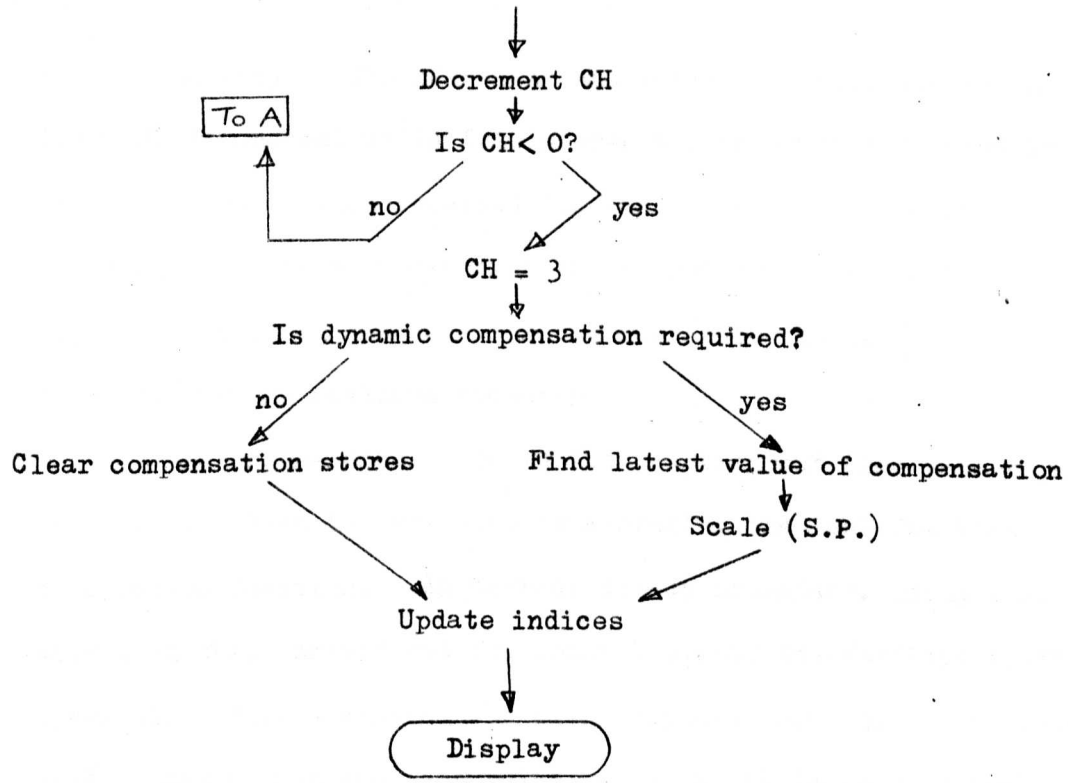


Figure 9.7. Main Program (OPTIM4) On-Line Routine. Sheet 1



Note: I.A. = Indirect Addressing

S.P. = Single-Precision

D.P. = Double-Precision

Figure 9.7. Main Program (OPTIM4) On-Line Routine, (Sheet 2)

(I.A.) facility. The addresses and indices appropriate to channel 4 (with CH=3) are set up in I.A. stores and the output routine is serviced. The procedure is then repeated for each of the other channels. When all 4 outputs have been transferred to the buffer in this way, they are simultaneously transferred from the buffer to the process, represented in this case by the analogue computer.

The cross-correlation is then carried out (see paragraph V below), and the d.c. bias is estimated by averaging the last few bits of the cross-correlation function. An address set-up procedure, using I.A. Stores as above, is then carried out for channel 4, and the feedback routine is serviced. This evaluates the step response from the appropriate sub-block of ϕ_{cy}, and stores the last point on it, which is used as the gain estimate, in the appropriate sub-block of SST. This gain estimate is then added to a double-length store (K, KP) which holds the total of gain estimates to date. K,KP is then scaled to single precision (S.P.), ready for adding to the code for output next bit, whenever optimization is required (as signified with breakpoint 3). If optimization is not required, K and KP are zeroed. The change in feedback is calculated, and then used by the dynamic compensation array routine (see paragraph VI below). This completes the feedback routine, which is then serviced for the other 3 channels. The dynamic compensation to be used next bit is then calculated if required from the compensation array, scaled to single precision and stored. If not required, the array is cleared. Indices are then updated and control returned to Display.

V. Cross-Correlation Algorithm

The cross-correlation of c and y is performed by an updating method, utilizing the change in y over the past chain-code period. The mathematical development is as follows: the n-th ordinate of ϕ_{cy} at the (i-1)-th interval of time is

$$\phi_{cy}(n,i-1) = \sum_{m=i-N}^{i-1} c(m-n)\ y\ (m) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (9.1)$$

so that

$$\phi_{cy}(n,i) = \sum_{m=i-N+1}^{i} c(m-n)\ y\ (m)$$

Hence

$$\phi_{cy}(n,i) = \phi_{cy}(n,i-1) + c(i-n)\ \left[y(i) - y(i-N)\right] \quad \ldots\ldots\ldots (9.2)$$

To implement this algorithm requires one multiplication and one addition for each ordinate, and is therefore considerably superior to evaluating ϕ_{cy} from equation (9.1), which requires N multiplications and N additions for each ordinate.

VI. Dynamic Compensation Routine (Figure 9.8)

During optimization, the output of the plant, y, will have a drift due to the changes in d.c. level of the controlled parameters. Dynamic compensation can be used to reduce the effect of this drift to a minimum, by estimating the change in output due to changes in the parameters from the latest step response estimates. (The convolution integral operating on the impulse response estimates could be used, but the step response method yields the required cumulative compensation directly.)

The compensation is stored as an array of 256 double-precision numbers. The computation is in two parts:

(i) The calculation of the array (Figures 9.8 and 9.9). This is calculated for each channel in turn, at the appropriate point in the feedback routine. As the channel phase-separations are less than the length of the compensation array, after the available step response has been used to update part of the array, the last value of step response is used to update the remainder of the array.

(ii) The derivation of the value of compensation to be used for the next program cycle. This is extracted from the array using the index DC, scaled to S.P. and stored. Its place in the array is then cleared ready for the next program cycle.

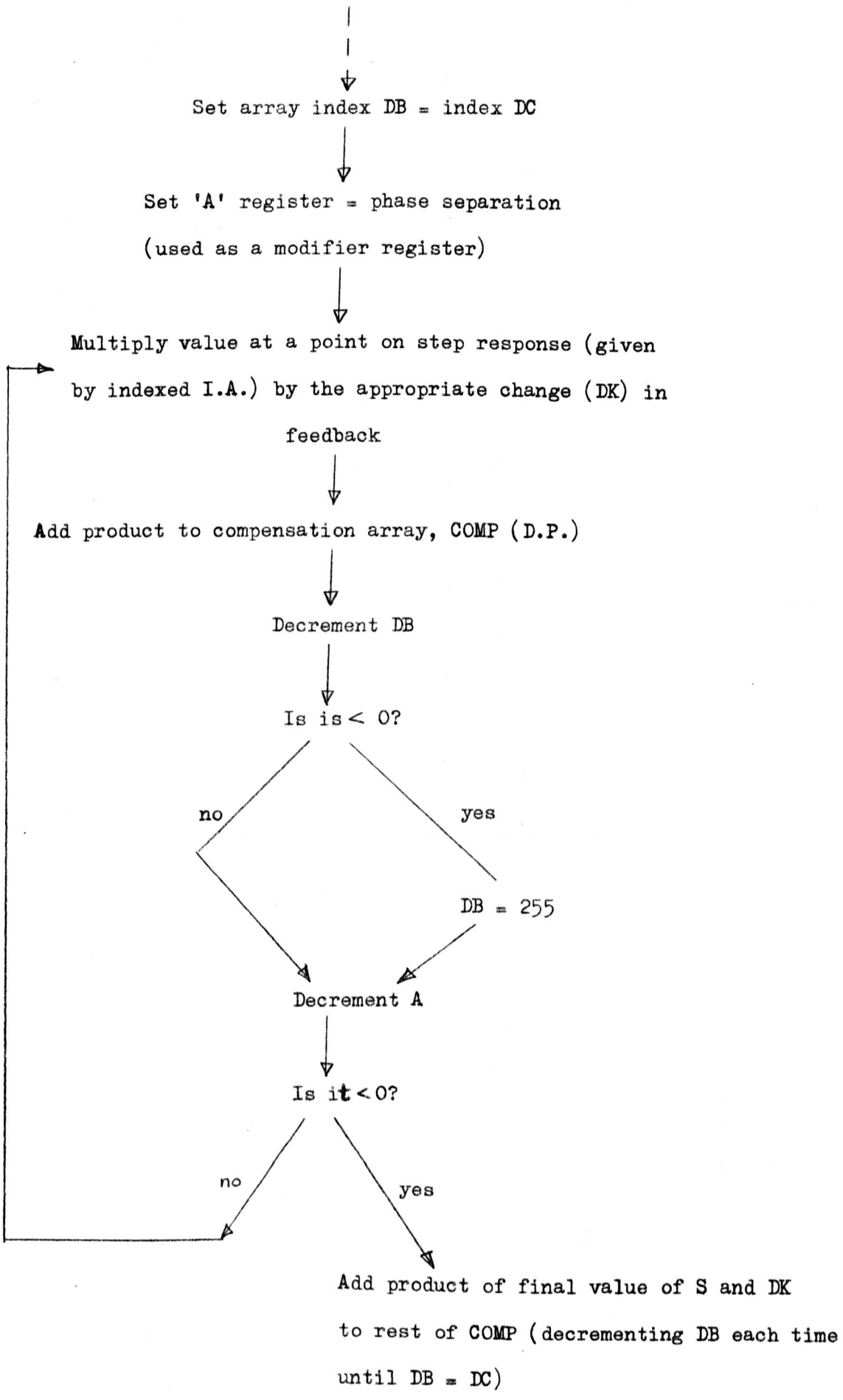


Figure 9.8 Dynamic Compensation for Any One Channel

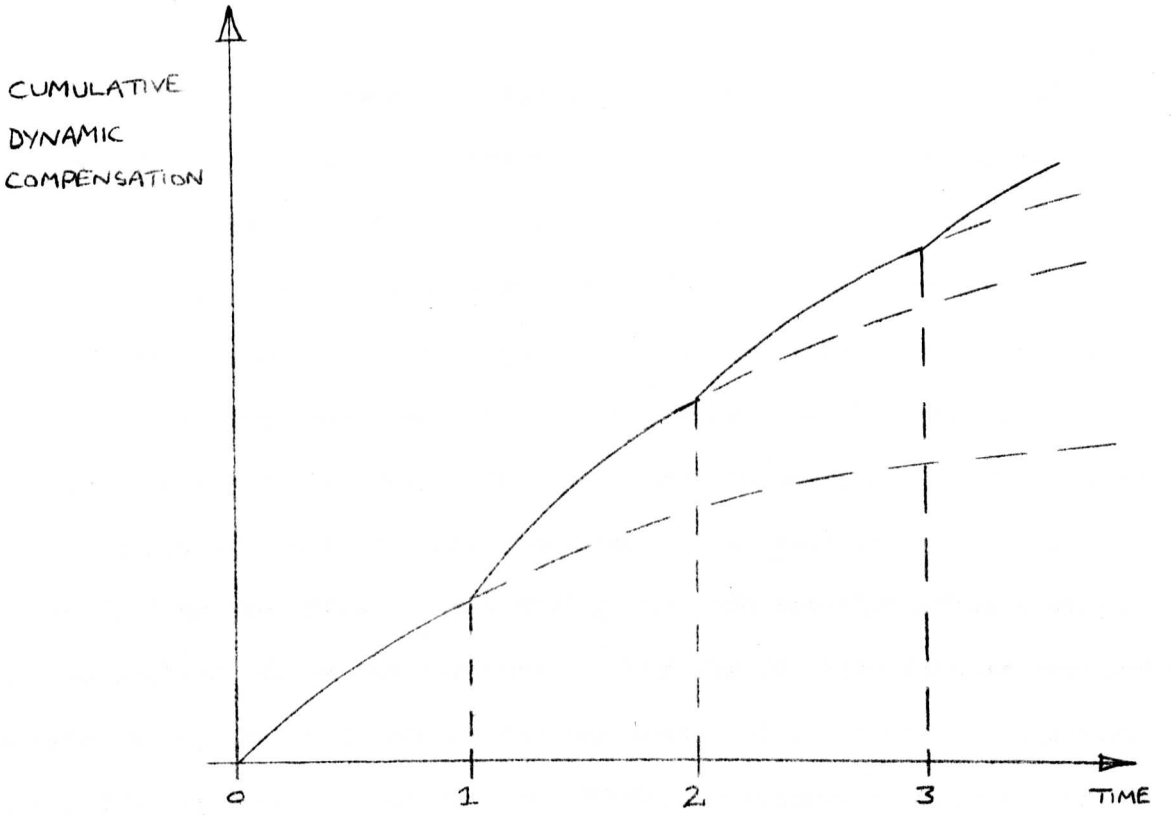


FIGURE 9.9:

BUILD-UP OF DYNAMIC COMPENSATION ARRAY

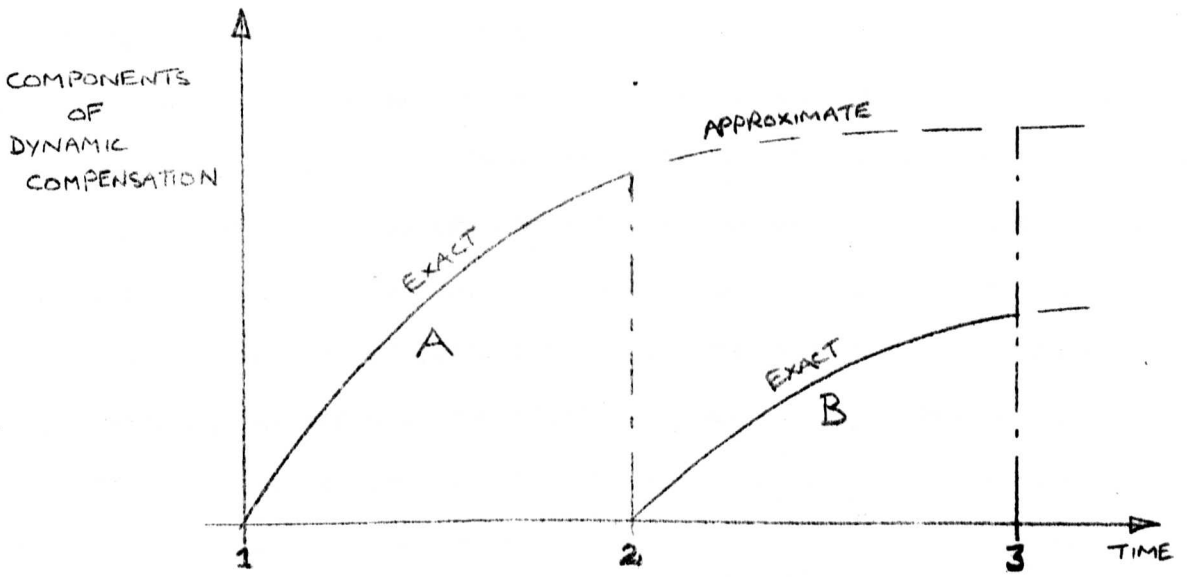


FIGURE 9.10:

ERRORS IN DYNAMIC COMPENSATION TECHNIQUE

Note that this dynamic compensation technique is not strictly accurate, as the gain of the plant will be changing. The error is illustrated in Figure 9.10 for the case where the gain is assumed to drop at point 2 to half its original value. This change is taken care of in future components of the compensation array, such as curve B, but curve A (and earlier ones) will be in error from point 2 onwards. This error is directly equivalent to that mentioned in Section 2.2 with regard to convolution of non-stationary functions. To avoid the error would involve finding the effect that changing the gain sometime after a step has been applied has on the response. This may be calculated if regression analysis is applied to determine the parameters of the transfer function, but would be extremely lengthy. In OPTIM4, no attempt is made to rectify the error; the effect of this is discussed in the next section (paragraph (vi)).

9.3 Experimental Results

OPTIM4 has been applied successfully to analogue computer models of plants with up to four controlled parameters. The 'hills' were square-law non-linearities, 'upstream' of first or second order dynamics.

Some of the results obtained are shown in Figures 9.11 to 9.13. The following observations were made:

(i) During the settling-down period after a chain-code is first applied to a single-parameter plant, the estimate of step response fluctuates, with many zero crossings and with excursions of up to 150% of its final value on either side of zero. This of course would lead to hopeless results if optimization is attempted too soon. As noted earlier, the minimum identification time is approximately equal to one settling time of the plant plus one period of the code.

(ii) Optimization of a one-channel plant yields reasonable results (Figure 9.11). For practical simplicity, the 4-channel program was used, with channels 2 to 4 open-circuit; the code was therefore much longer than the plant settling time. Figure 9.11 shows

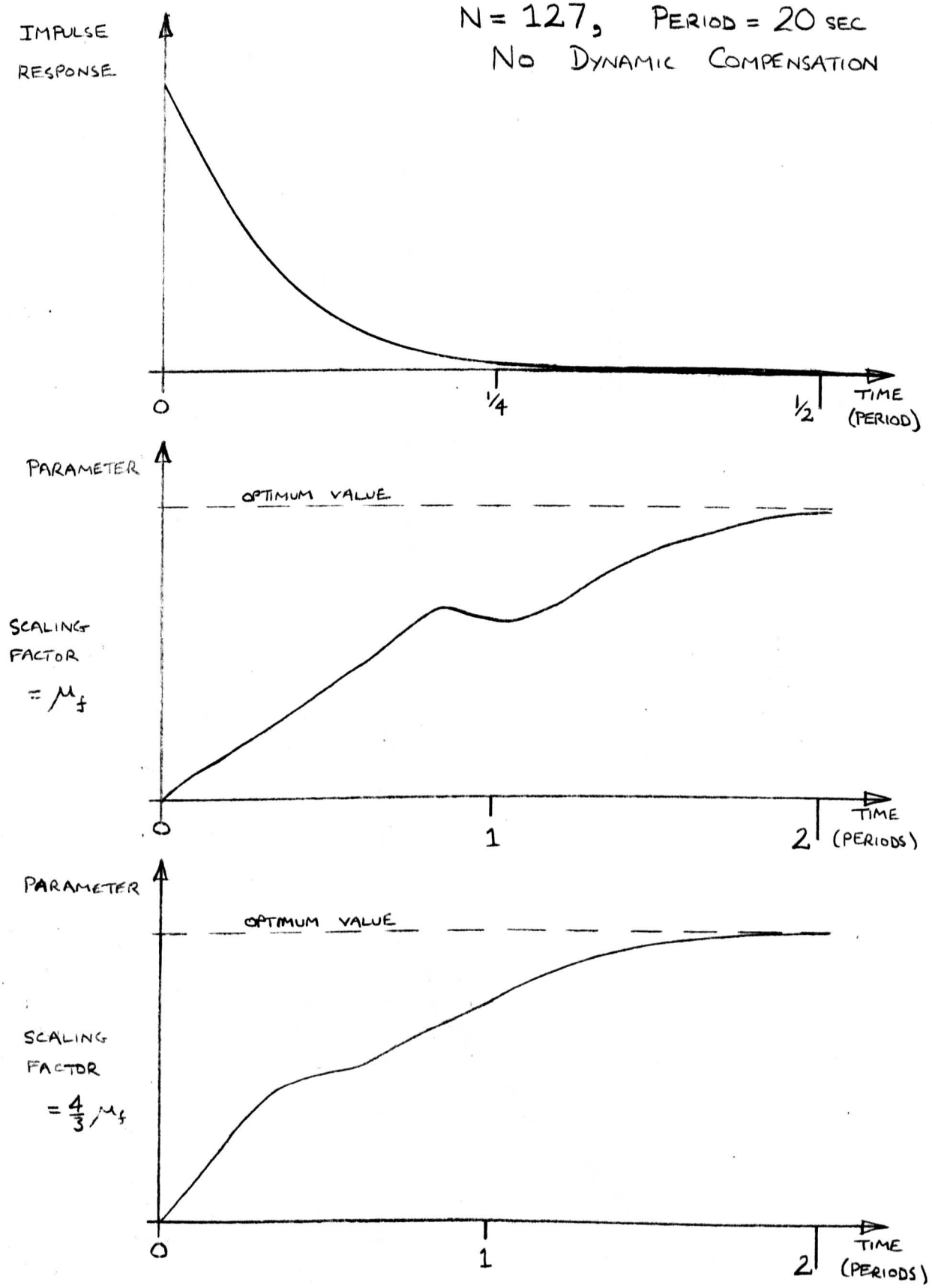


FIGURE 9.11: OPTIMIZATION OF ONE CHANNEL

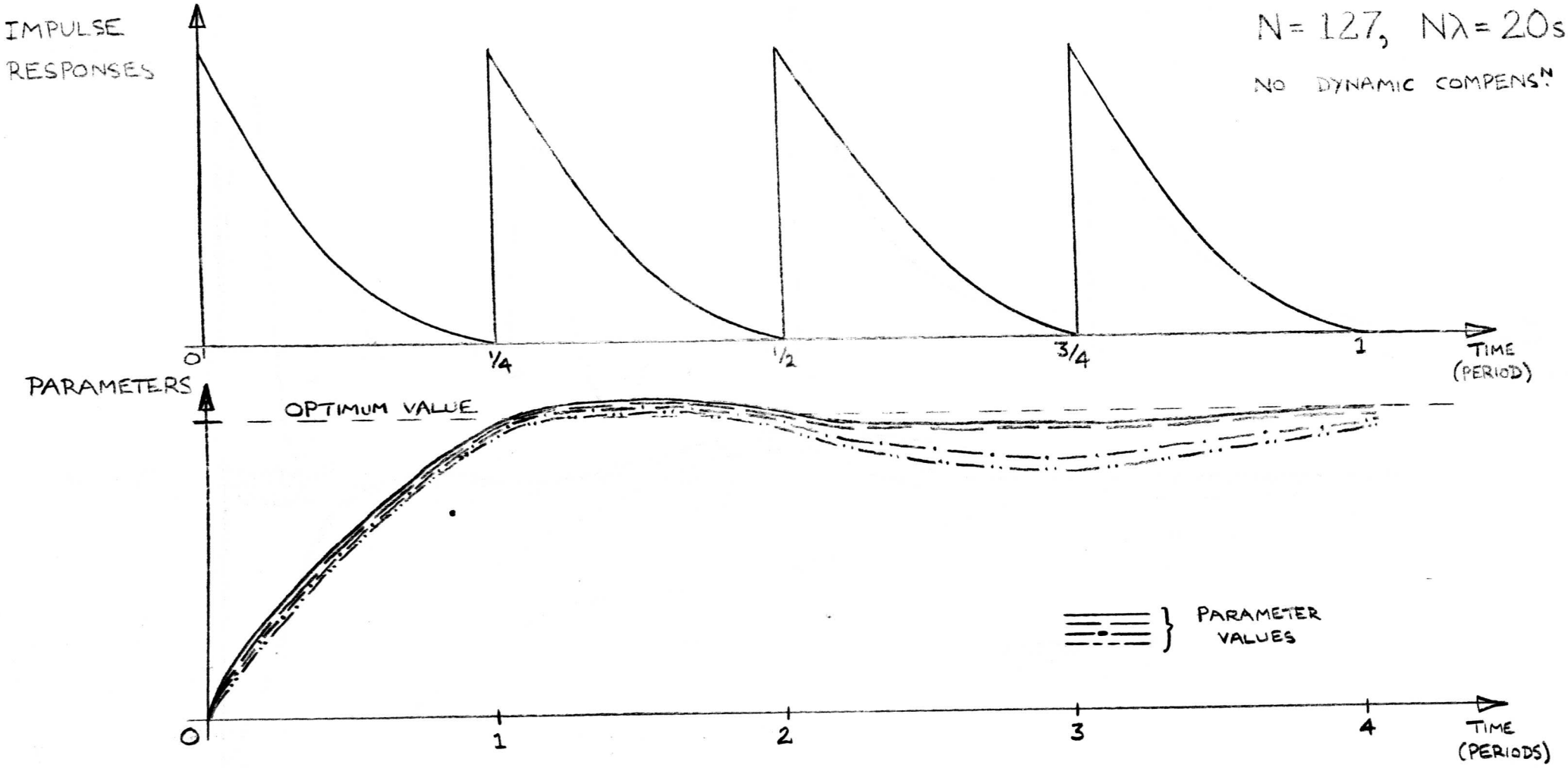


FIGURE 9.12: OPTIMIZATION OF 4 CHANNELS

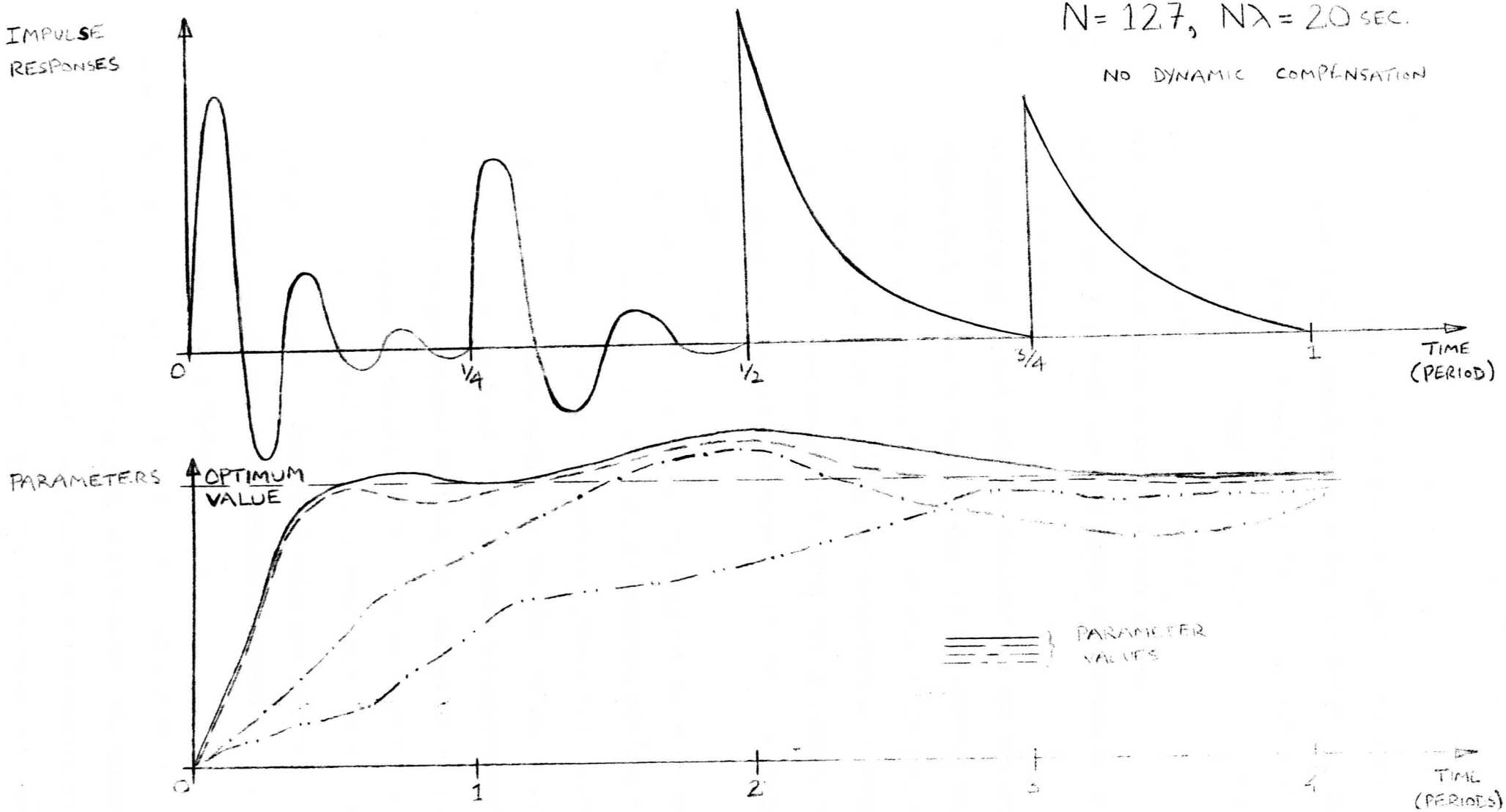


FIGURE 9.13: OPTIMIZATION OF 4 CHANNELS

the mean parameter value (given by the optimizing feedback), for two values of the feedback scale factor, denoted by μ_f and $4/3\mu_f$. For higher values, the system sometimes went unstable, depending on the starting-point (in time) of the optimization. Under the fastest stable conditions, the parameter can be brought to within 75% of its final value in one chain-code period.

(iii) Optimization of a plant with four first-order channels (Figure 9.12) gives good results.

(iv) Optimization of a plant with two 2nd-order and two 1st-order channels (Figure 9.13) is not as rapid as when all the channels are almost identical, as in the case above, but is still reasonably fast. For the example shown, three of the parameters were brought close to the optimum in one period, the fourth taking somewhat longer. After three periods, the performance deteriorated slightly, but picked-up soon after.

(v) If conditions suddenly change by any appreciable amount while the system is attempting to optimize, the cross-correlation function will no longer represent the impulse responses with any degree of accuracy, due to sudden changes in the plant output values. The increments in feedback will then no longer be representative of the slopes of the performance function, and instability will generally result. (This is analogous to the performance of the system if optimization is begun during the settling-down period mentioned in paragraph (i) above). The system will, however, be able to track relatively slow changes in the ambient conditions, or relatively slow movements of the 'hills'.

(vi) The dynamic compensation routine was tested with an artificial case; the non-linearities of the plant were removed, the optimization loop was left open, and a staircase function was generated for each of the feedback variables. In this way the effects of changes in

parameter values on the plant output could be tested for a plant with static gain characteristics. Without compensation, a steadily increasing bias was superimposed on the plant output, due to the changing mean values of the parameters. With compensation, the plant output after correction was only slightly altered while the parameters were changing, showing that the dynamic compensation routine worked well under static gain conditions.

In the normal optimization mode, the plant gains are continuously changing, and, as observed before, the dynamic compensation routine yields errors. At the start of optimization the error is small, but the process

adjust latest y → cross-correlate → integrate →
find next adjustment of y

is cumulative, and errors tend to build up. From the experiments it was observed that:

(a) There was no evidence to show that the maximum speed of optimization was appreciably improved by including compensation in any of the cases tried.

(b) Compensation tended to improve the estimates of impulse response at the start of optimization, but after a relatively short time a rapidly increasing divergence from the optimum occurred, resulting in permanent instability.

Further practical work would have to be done before definite conclusions can be drawn, but it can be stated that the power of the compensation technique may be severely hampered by the changing gains of the plant, and compensation may prove worthless in such cases.

(vii) In addition to the error described above, the estimates of impulse response are in error during optimization due to the changing gains of the plant. For example, for a single channel with a square-law non-linearity, the envelope of the amplitudes of the plant output

will decrease roughly exponentially during optimization, giving errors in the cross-correlation process. The output of the plant will then be

$$y(t) = \varepsilon^{-K_g t} \int_0^{\infty} u(t - s)\, h(t)\, ds$$

Substituting in equation (3.2) gives the corresponding estimate of step response, which will be a function of the starting point of the code. The standard deviation in estimate due to a varying starting point can then be deduced. The expression for this has been evaluated off-line for certain cases, and the results are shown in Figure 9.14. It can be seen that a rapidly decaying gain gives a large standard deviation and very inaccurate mean.

It might be thought that compensation could be used to counteract the effect of the changing gain. This would involve estimating the change in gain over the last bit, and adjusting the stored values of y accordingly. This would assume that the estimate of the change in gain over one bit is accurate. This cannot be true, since the settling time of the plant is many times longer than the bit interval. What is ideally required is an estimate of gain at the current time, which is of course physically unrealizable, regardless of the identification system used.

(viii) Desirable features for a computer for on-line optimization would include:

(a) A wide dynamic range for the computer input so that large ranges of parameter variations can be allowed.

(b) The provision of hardware signed arithmetic, as this would considerably ease the programming load.

(c) An overflow indicator would prove invaluable. (On the G.E.C.92 several instructions are required to test for overflow of signed numbers.)

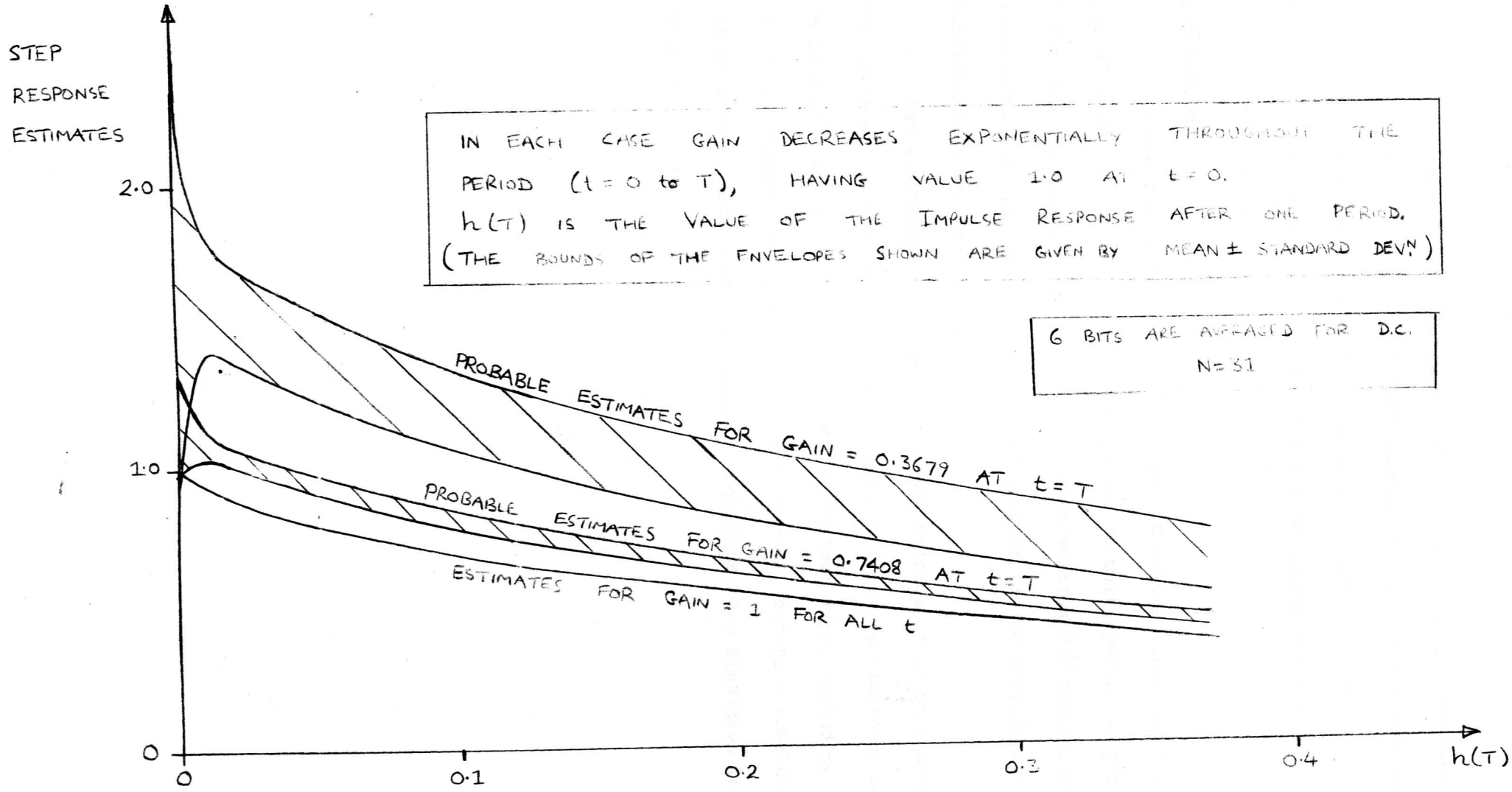


FIGURE 9.14: RANGE OF PROBABLE STEP RESPONSE ESTIMATES DURING OPTIMIZATION

(d) Lack of floating-point hardware is not a disadvantage for control work, but a 12-bit word-length is somewhat restricting, as double-precision may then be necessary whenever a cumulative total is required. A 15- or 16-bit word-length would be a considerable advantage.

9.4 Conclusions

It has been seen that an optimization system utilizing the shifted codes technique can be successfully applied to multi-parameter plants. A flexible on-line computer program has been written for feasibility assessment and development work. This program can be considerably simplified in any specific application.

The advantages of using statistical methods of plant identification and optimization are not as great as might at first be imagined, and it will be necessary to carry out a great deal of further work, including an extensive comparative study of available methods, before a complete appraisal can be made. Some other optimization techniques are discussed in the next chapter.

CHAPTER 10

A Survey of Identification Methods and Optimization Strategies

10.1 Introduction

Some aspects of system identification using sine-wave and pseudo-random perturbations have been discussed in this thesis. There are, however, many other techniques available, and some of these will now be considered. Furthermore, strategies for optimization are not limited to the proportional-to-gradient method, and other systems for on-line optimization are outlined below. A brief mention of off-line techniques is also included for completeness.

10.2 The Available Techniques

The two components of an optimization system are identification and parameter adjustment, and as their effects usually interact, it is essential to consider them together. Identification methods can be divided into two classes:

(i) those involving step test signals, and

(ii) those involving random or periodic test signals.

In the order of increasing practical complexity, step inputs require little hardware, sine- and square-waves involve some form of phase-detection equipment, and pseudo-random signals require logical code generators and cross-correlators.

It will be observed that all these methods involve some form of correlation of output with input. If a pseudo random perturbation is used, a great deal of information will be input to the plant, and cross-correlation eliminates a large proportion of errors due to noise in the plant. Suppose that a step input is used instead, and that the output of the plant is averaged over several intervals of width λ before the step is applied, and over several intervals λ after the plant has substantially

settled down after receiving the step. This is equivalent to cross-correlation, but in this case the input has a constant value, in the absence of noise, before the step is applied, and another constant value afterwards. In a discussion, D.W. Clarke of Oxford University described results of an analysis that showed that a step input gave somewhat better noise rejection than a chain-code, for the same output power. This suggests that the apparent advantage to be gained by choosing a chain-code is, to say the least, nebulous, and that the most important aspect of noise rejection ability is the cross-correlation process, and not the form that the input takes. Furthermore, the time taken to estimate a plant gain under noisy conditions is roughly the same for the two techniques. After one period of a chain-code, about one settling time is required before all the output measurements are reliable; with a step, time to average output values is necessary after one settling time has elapsed.

Other perturbations that could be used include sine-waves and square-waves. It was shown in Chapter 5 that sine-waves and chain-codes have comparable noise-rejection properties, the chain-code being somewhat preferable if rapid identification is required. There is no evidence in the literature to suggest that square-waves are any better. It is thus reasonable to conclude that the step input has a good deal to recommend it, including the simplicity of practical implementation, but a large amount of comparative work is necessary before any definite rules can be laid down as to which system is to be preferred.

Turning now to parameter adjustment, there are again many different methods available. There are three types of application:

(i) On-line use, applied to a plant with unknown characteristics, where the plant output is monitored for different input levels,

(ii) on-line use of an auxiliary network whose parameters are chosen

off-line[67]; this applies to plants with known transfer characteristics but variable input or disturbance factors, and

(iii) Off-line use, where the function between 'input' and 'output' is explicitly known, and the output or gradient have to be found analytically; this is the classical function-maximization problem.

A secondary division is between systems using knowledge of the plant output alone, and those utilizing first or higher derivatives of the output directly. For example, in on-line applications, a step input yields only the output directly, a binary chain-code or a sine-wave yield the gradient directly, and a ternary chain-code yields the gradient and second derivative directly. In off-line applications, the number of derivatives to be calculated depends on the complexity of the function to be maximized.

Consider first on-line applications. Feldbaum[18] has collated several step-input methods, which can be classified as:

(i) Those using combined test and operative steps ('dual control'), with either

  (a) constant step size, or

  (b) variable step size.

(ii) Those using separate test and operative steps, using either

  (a) one test step for each operative step, or

  (b) the average of several test steps for each operative step.

For group (ii), the test step is normally of fixed size, and the operative steps may be of fixed or variable size.

As examples, the most fundamental algorithm[17] for group (i) (a) is

$$K_{n+1} - K_n = (K_n - K_{n-1}) \operatorname{sign}(y_n - y_{n-1}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10.1)$$

where K_n and y_n are the parameter value and plant output at the n-th iteration, whereas a typical strategy for group (i) (b) is

$$K_{n+1} - K_n = (K_n - K_{n-1})(y_n - y_{n-1}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10.2)$$

It has been suggested[68] that a worthwhile modification to equation (10.1) is to weight a 'success' by more than unity and a 'failure' by less than unity. The step size in all cases must be chosen to give the best compromise, with reasonable speed and local stability, and small offset. Some control engineers prefer a peak-holding method[3], where small deviations from the optimum are corrected, whereas others prefer a dead-zone near the optimum[2,16], so that high-frequency hunting is prevented. In any system it is advisable to include a saturating non-linearity to limit runaway should it occur.

An algorithm suggested by Moran[2] is

$$K_{n+1} - K_n = (K_n - N_{n-1})\{(y_n - y_{n-1})\, W_o + (y_{n-1} - y_{n-2})\, W_1 + \text{---}\} \quad \ldots(10.3)$$

where the W_i are weighting factors chosen to reduce the effects of transients to a minimum. For example, for a first-order plant, W_o and W_1 can be chosen so as to eliminate transients.

A pilot program was run on the G.E.C.92 computer to optimize a first-order plant using step inputs given by equation (10.2). The results are shown in Figure 10.1, from which it can be seen that the method compares favourably with the chain-code method under noise-free conditions. It is suggested that a full-scale program be written to evaluate the various aspects of dual control optimization.

Considering now systems where the primary measurement is gradient of performance function rather than plant output, we see that it is hard to derive any comprehensive comparisons from the available literature, in spite of the large number of papers written on the subject. For example, Jacobs develops a stability analysis for sine-wave systems in the presence of noise and disturbances, but the assumptions made severely limit the usefulness of the results. Limitations also apply to the stability analysis of chain-code systems derived by Murthy[47], and the exhaustive analysis given by Gupta[26] of the transient performance of several systems is hard to apply in practical situations. In all these cases, the proportional-to-gradient method is

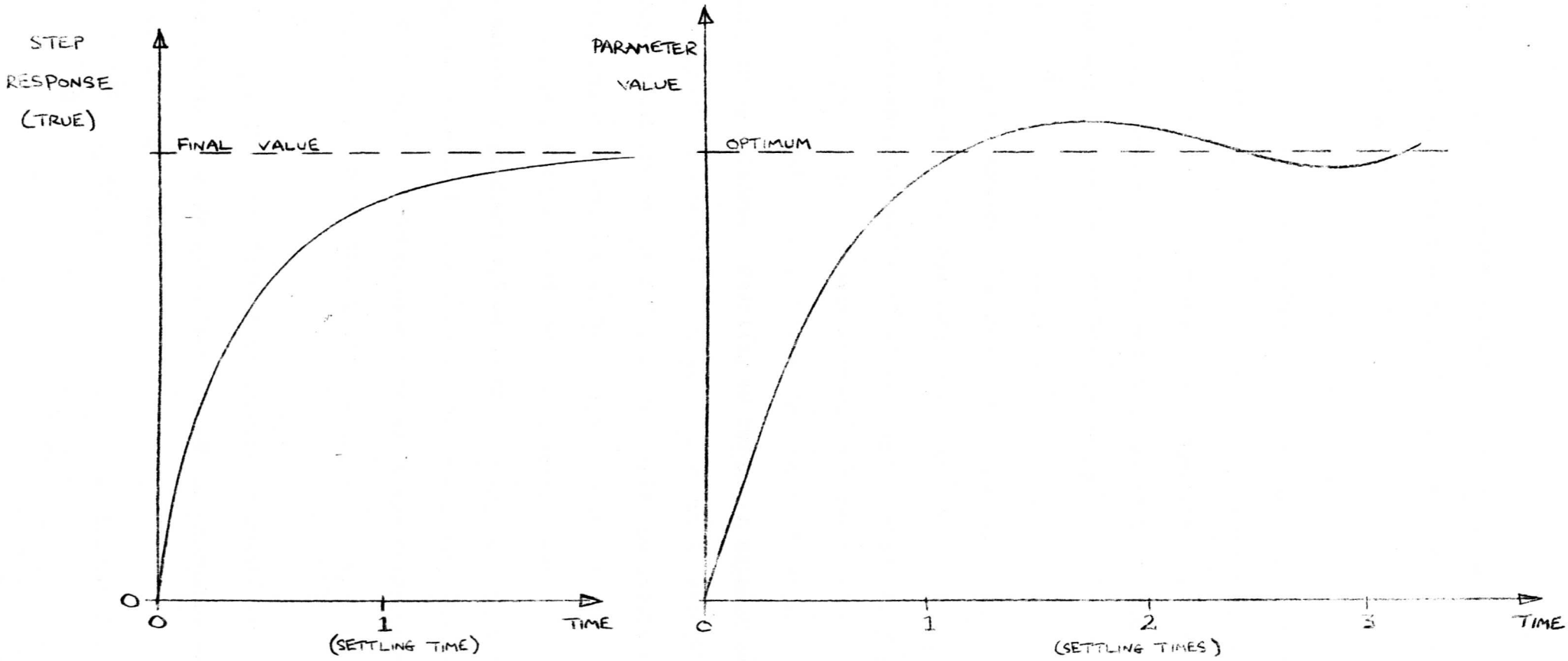


FIGURE 10.1: OPTIMIZATION OF FIRST-ORDER PLANT USING DUAL CONTROL

used. Figure 10.2 illustrates the application to a quadratic hill. After one iteration, the gain at point A, say, is found. The next step to be taken is given by

$$\Delta K_A = \mu_f g_A \quad \text{.....(10.4)}$$

where μ_f is an arbitrary scale factor. Although this technique has the useful property that, for a convex hill, the further the point is from the optimum, the larger the parameter adjustment made, it gives no indication of the ideal step size, and μ_f has to be chosen by trial and error. For example, consider the hill given by

$$P = -K^2 \quad \text{.....(10.5)}$$

If $\mu_f = \frac{1}{2}$, the system will optimize in one step; if $\mu_f > 1$, the system will diverge from the optimum; for other values of μ_f, the system will optimize, the number of steps being governed by μ_f.

An obvious improvement on this method, that can be applied if the hill does not move significantly during the course of two gradient measurements, is as follows. Starting as before, a measurement at A is made, and, by choosing a relatively small μ_f, a small step ΔK_A is made. Another gradient measurement is made, at B. This now completely specifies the quadratic function, and the second adaptive step will be exact, its value being directly extrapolated from the gradient graph. This method of fitting curves to gradient estimates and finding the appropriate zero-crossing can be extended to functions of any order, but, due to inevitable movements of the hill, it may be better to apply the quadratic-convergence technique repeatedly, rather than utilize a high-order approximation occasionally.

Should high-order curve fitting be thought desirable, the analysis is straightforward. The gradient function can be approximated to a polynomial of order u, so that

$$g(K) = \sum_{i=0}^{u} a_i K^i \quad \text{.....(10.6)}$$

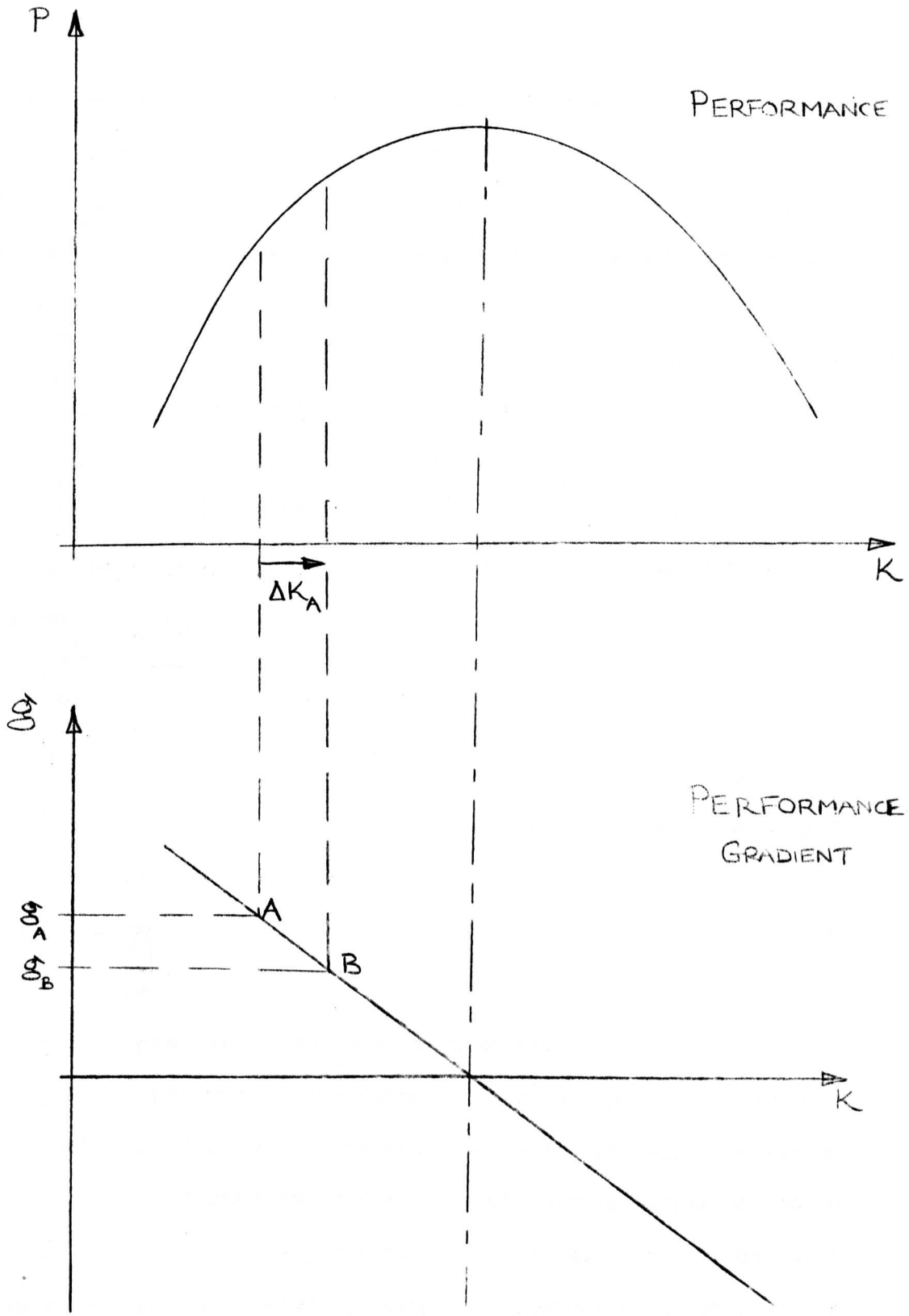


FIGURE 10.2:

OPTIMIZATION OF A QUADRATIC FUNCTION

where the a_i are constants. By estimating gradients for (u + 1) distinct values of K, the a_i can be found from the matrix equation.

$$\underline{A} = [K]^{-1} \underline{G} \quad \text{.....(10.7)}$$

where $\underline{A}$ and $\underline{G}$ are column vectors of the a_i and g estimates, and $[K]$ is a square matrix of the form

$$[K] \equiv \begin{bmatrix} 1 & K_o & K_o^2 & \cdots & K_o^u \\ 1 & K_1 & K_1^2 & \cdots & K_1^u \\ \vdots & & & & \vdots \\ 1 & K_u & K_u^2 & \cdots & K_u^u \end{bmatrix} \quad \text{.....(10.8)}$$

where K_j is the value of K at the j-th point. Evaluating equation (10.7) gives

$$a_i = \sum_{j=0}^{u} \frac{\sigma_j^{(u-1)} g_j}{p_j} \quad \text{for } i = 0, 1 \,..\, u$$

where $\sigma_j^{(k)}$ = sum of products of K_r taken k at a time, excluding those with non-distinct suffices and terms involving K_j.

$$\sigma_j^{(0)} = 1$$

$$p_j = \prod_{\substack{r=0 \\ r \neq j}}^{u} (x_r - x_j)$$

g_j = gradient estimate at j-th point.

Having estimated the a_i, the appropriate root can be found by one of the usual methods. This then represents the best estimate of optimum deducible from the information available. Of course, equivalent to (u+1) estimates of gradient are (u+2) estimates of output, using a step input, or one estimate of (u+1) derivatives, using a chain-code with (u+2) levels. The measurement time in all cases is at least (u+2) settling times.

In on-line applications, the speed of optimization of any system depends on the dynamics of the plant. It will be appreciated that the effect of a parameter change cannot be accurately measured until at least one settling time has elapsed, whatever method of identification is used.

However, it may often be found that a system will optimize quickly when using operative steps at intervals that are only a fraction of the settling time. This is possible whenever the plant response soon after a step has been applied is a representative measure of the response if the plant were given time to settle. This is often the case, and explains the rapid optimizations obtained in the experiments described in Chapter 9, but it does not apply to many practical plants, for example those where the response is non-minimum phase. The speed of optimization therefore depends on the shape of the step response, rather than on the settling time, and, for any specific applications, it is necessary to examine the shape of the response before determining the frequency to be used for the operative steps.

A "self optimizing optimizer" using chain-code indentification can be postulated, in which the period of the code is adjusted continuously from the cross-correlation, so that the period is always a minimum. However, this method will fail, because cross-correlation carried out whilst the period is changing will be seriously in error, as a chain-code with unequal clock-intervals does not have an impulsive auto-correlation function.

Throughout this thesis, we have considered a particular class of optimization, where the performance index is a function of the parameters and the plant output. It should be pointed out that there is a related class of problems, in which the performance is measured by such a criterion as the speed of response to a step input, in which case a somewhat different approach is needed to that used here.

A brief outline of off-line optimization strategies, derived for the maximization of analytical functions, will now be given, as they can also prove useful for certain on-line applications. In analytical work, it is desirable to minimize the total amount of calculation, even though the number of steps to the optimum may not be the absolute minimum. Most of the techniques applicable to multi-dimensional functions[69,70] involve:

(i) choosing the "best" direction in which to move, and then

(ii) moving along that direction until a maximum is found.

These steps are repeated until the optimum is reached. The "best" direction is often along the line of steepest ascent.[68,71] (The direction utilized in the proportional-to-gradient technique of Chapter 9 can be shown to be the line of steepest ascent.) For a two-parameter system, this results in a series of steps in one of two directions at right angles, the directions being dictated by the starting direction. An alternative method is to find the maximum in the current direction and then track back part of the way before choosing the next direction.[68] The conjugate-gradient method[72,73,74] is an alternative means of allocating the "best" direction. In this case the steepest descent vector, r_o, is used initially, and a maximum, X_o, is found on it. The next direction is then given by the plane through X_o conjugate to r_o. This procedure is then repeated until the optimum is reached.

10.3 Conclusions

It has been seen that the variety of optimization problems and of their possible solutions is enormous. Several on-line systems have been discussed, with their relative advantages and disadvantages. The off-line problem has also been mentioned. At this stage in the history of process optimization, no system can be postulated as definitely "the best". Before this can be done, more must be known of the practical factors influencing a real plant, including the effects of distributed dynamics.

SUGGESTIONS FOR FUTURE WORK

AND

CONCLUSIONS TO THE THESIS

SUGGESTIONS FOR FUTURE WORK

A great deal of analytical work on hill-climbing methods appears in the literature, but very few comparative studies have been carried out to ascertain the best optimization system for any given conditions. Such comparisons should include relative assessments in terms of

(i) time to optimize; for example, the time from the initial state to the first crossing of the optimum,

(ii) stability,

(iii) steady-state error.

It is suggested that a process-control computer be used to implement several on-line techniques, applying them in turn to each member of a set of rigidly-defined plant models, with noise of known statistical properties added at various points. Once the system is set up, a comparative study can rapidly be made.

Programs discussed in this thesis have been written for chain-code and step perturbations, using the simplest optimization strategies, and these can be extended or modified to cover many other types of optimizer. Aspects to be analysed should include

(i) the best clock frequencies,

(ii) the number of operative steps for each test step; for example, whether parameter changes in a chain-code system should be made once per clock interval, once per period, or somewhere in between,

(iii) techniques for compensation for transients,

(iv) ideal step sizes, to satisfy the conflicting requirements of avoiding perching on local maxima, and missing the top peak.

Theoretical analyses should also be extended, but practical comparisons are needed to cover aspects beyond the scope of the theory.

When the most suitable system has been found for any class of problems, a concise computer program should be written and documented to enable others to use it. In this way, a library of optimization programs can be built up.

Many attempts have been made in the literature to analyse stability of optimizers, but without much success. One problem is that the stability is dependent on the disturbances, and the analysis has therefore to be restricted to a well defined disturbance, making it of little practical use. A second problem is that the analysis has to be restricted to a particular plant model, with the associated difficulty of assessing the dynamic performance with respect to the "hill", remembering that the relevant transfer function is that between parameter and output under given, and usually varying, input conditions, which function may well prove hard to find. Much more needs to be done, to enable an engineer to assess the probable stability of his particular practical configuration.

Also necessary is the extensive application of optimizers to real plants. So far, applications have generally revealed that noise in the plant and the transducers yields intolerable errors. It may prove best to use some kind of self-learning process based on knowledge of past situations, or pattern-recognition techniques to extract signals from noise. The possibilities available with a digital computer are almost boundless, and will increase still further when present-day developments of components with operating times of only a few nano-seconds are fully realized. Against the advantages of a fast general purpose computer must be set the cost, which may be relatively high due to components in the computer that are not utilized in a specific application, and it may well prove cheaper, with the advent of versatile large scale integration of field effect transistors, to devise special purpose hardware for inprocess optimization. The performance and economics of each technique should be evaluated, to enable efficient optimization to be introduced into industrial environments as soon as possible.

CONCLUSIONS TO THE THESIS

The analysis of a method of simultaneous identification and optimization of a control system employing pseudo-random perturbations has shown that large errors can arise in the estimate of gain if the d.c. level of the plant output is ignored. Techniques are derived in which the d.c. level is allowed for, and a method of practical implementation involving two running summers and a differencing technique is developed, together with methods for use with an on-line computer. It is shown how to choose the parameters of the identification system to give the best signal-to-noise ratio under various conditions of noise, with any plant dynamics.

An identification system using sine-waves is analysed under similar conditions, and it is shown that the signal-to-noise ratio is highly dependent on the parameters of the system, but is usually of the same order of magnitude as that for the chain-code system.

The chain-code and sine-wave perturbations are applied in turn to plants having more than one dynamic path between parameter and output, and it is shown that the estimate of the position of the optimum ~~derived using sine-waves~~ varies with frequency. ~~whereas the chain-code system yields the correct solution at all frequencies.~~ The theory is supported by applications to a model based on the type of situation arising in a steam-generating plant.

It is shown that applications of chain-codes to multi-parameter identification yields problems due to the cross-coupling between channels. The best solution is found to be the use of time-shifted versions of one code, applied to the various parameters. Practical methods of implementation are discussed, and a multi-channel program is developed for optimization using a process-control computer. This program was successfully applied to a model of a four-channel plant set up on an analogue computer.

The results prove the validity of the method, but further work is required to compare it with alternative techniques. The advantage of using compensation for transients due to previous parameter changes is not as great as might be hoped, due to the time-varying gain of the plant and the cumulative nature of the compensation process.

Various identification and optimization techniques are discussed, and it is noted that a great deal of further work is necessary in order to compare the performances of possible systems effectively. It is far from self-evident that the chain-code method is superior to other methods, even under noisy conditions. A basic limitation of any system is that the exact effect of a parameter change cannot be determined under noisy conditions until up to two settling times have passed. It is, however, possible when optimizing certain plants to utilize gain estimates made in a much shorter time, thereby speeding optimization.

It is hoped that a great deal of rationalization of both the theory and practice of on-line optimization will take place, so that industrial engineers can apply techniques to practical plants to a much greater extent than is possible at present.

APPENDICES

APPENDIX A1.1

PROPERTIES OF CHAIN-CODES

The auto-correlation function of a chain-code[34] with levels $\pm$ 1 is

$$\phi_{cc}(\tau) = \frac{N+1}{N} \sum_{j=-\infty}^{\infty} U(\tau - jT) - \frac{1}{N} \qquad \text{(A1.1.1)}$$

where $N = T/\lambda$ and $U(\tau)$ is a triangular function given by

$$U(\tau) = u_{-2}(\tau - \lambda) - 2u_{-2}(\tau) + u_{-2}(\tau + \lambda) \qquad \text{(A1.1.2)}$$

where $u_{-2}(\tau)$ is a unit ramp function.

$\phi_{cc}(\tau)$ is illustrated in Figure A1.1.1.

For certain purposes $\phi_{cc}(\tau)$ can be approximated to an impulse, such that

$$\phi_{cc}(\tau) \triangleq \frac{N+1}{N} \lambda \, \delta(\tau) \qquad \text{(A1.1.3)}$$

where $\delta(\tau)$ has an area of 1 if $\tau = 0$ and of zero otherwise. In making this approximation, three aspects of the true auto-correlation function are ignored, namely

(i) the width of the true 'spike' (which is 2λ at the base),

(ii) the periodic nature of the true auto-correlation function, and

(iii) the d.c. level of the true auto-correlation function ($-\frac{1}{N}$).

(Note that the term d.c. is used here to denote any zero-frequency component of a signal, whatever its nature, as is the customary, though often illogical, notation).

Chain-codes have three fundamental advantages over white noise for control-system identification purposes:

(i) They have finite repetition times, so that the correlation is performed over a finite time, rather than the theoretically-infinite time necessary for white noise.

(ii) They are deterministic, that is they have time-invariant statistics.

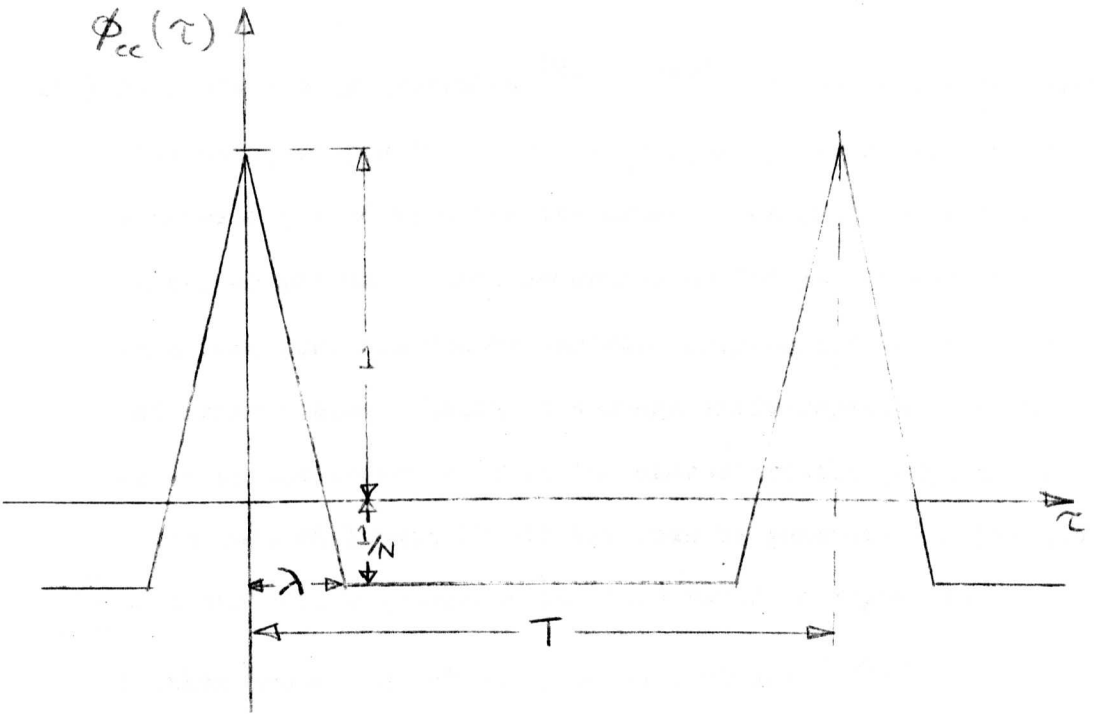


FIGURE A1.1.1:

AUTO-CORRELATION FUNCTION OF CHAIN-CODE

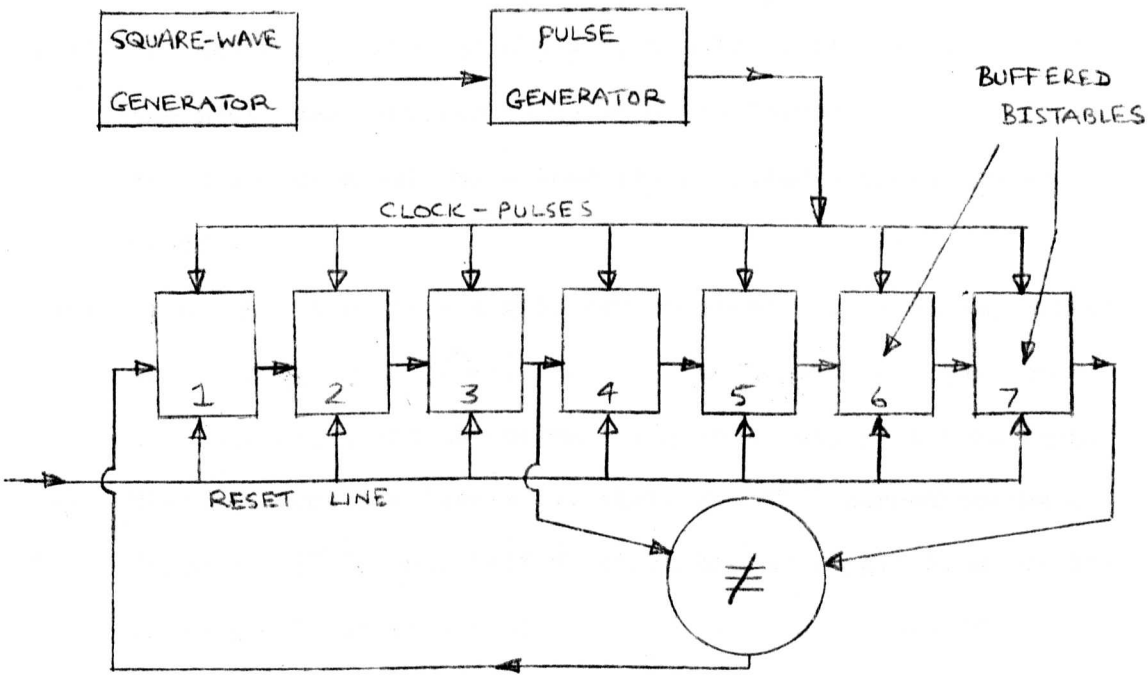


FIGURE A1.1.2:

LAYOUT OF A CHAIN-CODE GENERATOR

(iii) They are easily generated.[29,30,35,36] A generating function, consisting of powers of the unit-delay operator, D, can be defined. A necessary condition fow the codes to be used in this thesis (known as maximal length sequences or 'm'-sequences) is that this function, the "characteristic polynomial", must be primitive and irreducible. Using an n-stage shift-register with feedback given by the powers of D in the characteristic polynomial, a chain-code of length (2^n-1) bits can be generated. The layout of a chain-code generator is illustrated in Figure A1.1.2.

Further properties of chain-codes include:[37,38,39].

(i) If an m-sequence is added, modulo-2, to a delayed version of the same sequence, the result is another delayed version of the same sequence.

(ii) The time-inverse of an m-sequence derived using feedback from stages f_1, f_2.....f_r, n of a shift-register is given by using feedback from stages $n-f_1$, $n-f_2$.....$n-f_r$,n.

(iii) For any small value of n($n \neq 8$, $n \leq 11$) there exists at least one m-sequence derived using only two feedback connections. For $n = 8$ or $n > 11$, more than two feedback connections are needed.

(iv) If any n consecutive digits are considered as a binary number in the range 0 to (2^n-1), then all such numbers appear once and once only, except the null sequence, which is forbidden.

(v) When the code has levels ± 1, there are 2^{n-1} zero-crossings.

(vi) There are 2^{n-1} runs, half of which are of length 1, a quarter of length 2, an eighth of length 3, and so on, providing the number of runs so indicated is greater than 1. There are equal numbers of runs of either state, except that there is a run of n ones but not of n zeros (for a 0,1 code), and a run of (n-1) zeros but not of (n-1) ones.

Various methods of evaluating the feedback necessary to generate particular delayed versions of m-sequences have been described in the literature[40-45]. The technique used for the practical work of Chapter 2 of this thesis is due to A.C. Davies[44]. The delay operator raised to the power n is divided, modulo-2, by the characteristic polynomial, using the method of detached coefficients. This long-division process is continued for as long as is necessary. The remainder polynomial after each division represents the feedback necessary to generate a version of the code delayed by an amount deducible from the long-division process.

This thesis is mostly confined to work using two-level chain-codes. Considerable work appears in the literature on the subject of 3-level codes. It is shown in Chapter 2 that the slope of a performance function can be assessed using a 2-level code; if a 3-level code is used the second-derivative of the function can also be assessed. This is discussed more fully in Chapter 10, where it is shown that the apparent advantage of 3-level codes is not as great as might at first be thought.

APPENDIX A2.1

EXPERIMENTAL CONFIGURATION FOR SYSTEM IDENTIFICATION

The apparatus is shown in Figure A2.1. The chain-code generator consisted of a seven-stage shift-register, with feedback from the 3rd and 7th stages to the 1st stage, via a non-equivalent gate (or modulo-2 adder). Clock-pulses for the register were derived from a specially built unit driven from a square-wave generator. A reset line was provided, to set the bistables to represent 1 000 000 (binary) on pressing a button.

Two lines carried the output of the chain-code generator, one going to the plant model, the other to the running summer. The plant was represented by a single lag, using a summing amplifier of the analogue computer, with capacitative external feedback. (It is shown later why an integrator with resistive feedback could not be used.) The basic unit of the running summer is a bi-directional counter, on which the running sum is stored, in scaled form. At each clock pulse, the latest value of the code, taken as ± 1, is added to the counter, and another value, derived from a delayed version of the code, is subtracted. The delayed version is out of phase with the original code by the running sum period, and is obtained by suitable logic applied to the shift-register (see Appendix A1.1). The counter is a four-stage binary counter, utilizing 'AND' and 'OR' gates and bistables. Before the clock-pulses are switched on, the bistables are preset to give the scaled running sum corresponding to the initial state of the shift-register and the running sum period. After switching on the clock-pulses, the counter registers the running sum divided by 2^{λ}.

The output of the running summer consists of lines from each bistable. Each line is passed to a voltage generator with two outputs, one + 30 volts and the other - 30 volts, polarized according to the logic level of the

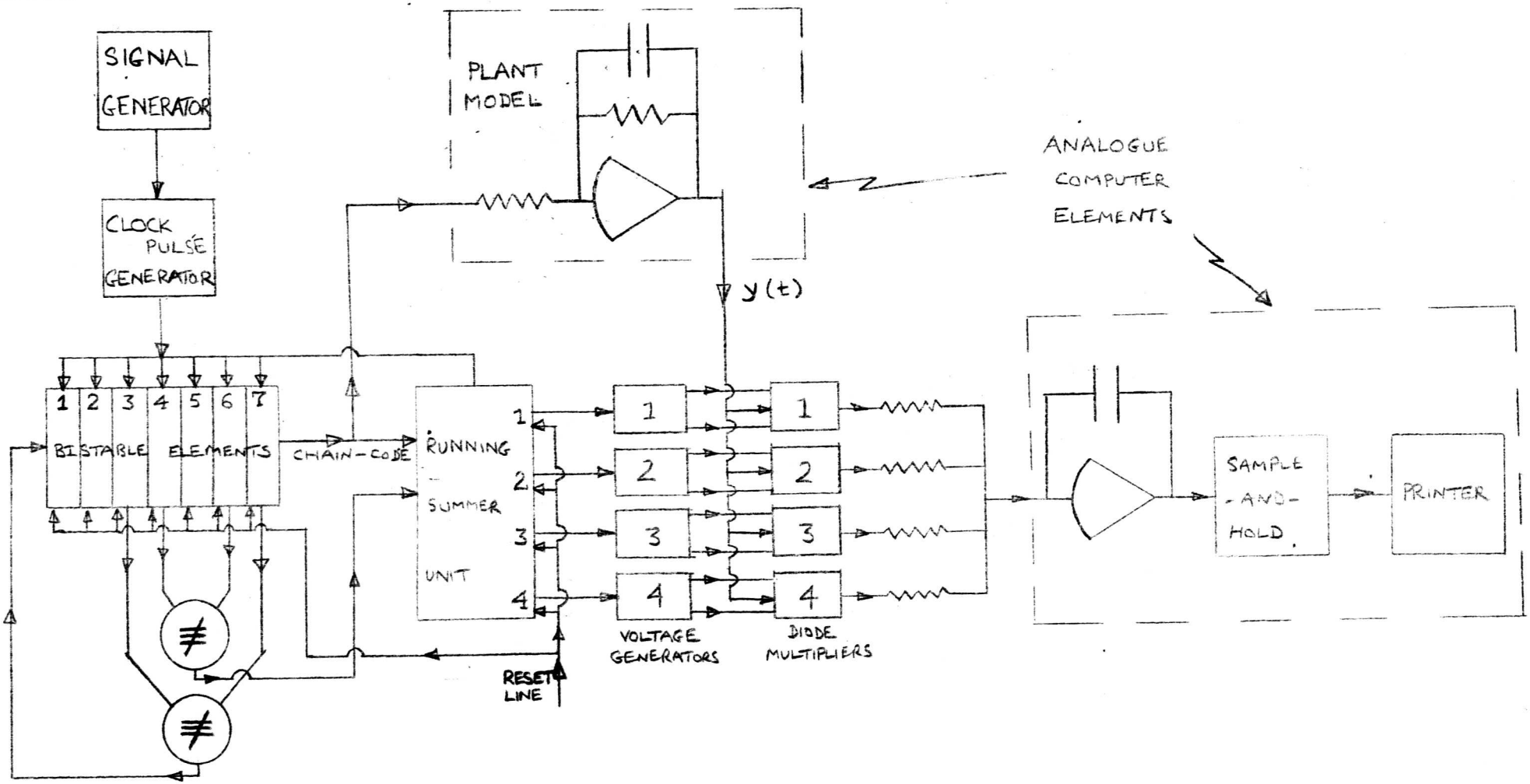


FIGURE A2.1: EXPERIMENTAL CONFIGURATION

input. Each pair of lines is then used for controlling a 6-diode bridge, the input of which is the output of the plant model. By taking the weighted sum of the outputs of the four diode networks, a signal is obtained which represents, in scaled form, the product of the running sum and the plant output. The summation is performed by an amplifier which also integrates the sum. The analogue computer is put into an iterative mode of problem-check, compute, hold, under the control of a built-in timer, such that the output of the integrator represents the required step response estimate. Its value is printed out automatically, in decimal form, during each hold operation. (The iterative computing mode operates on all the integrators in the computer. Therefore, as it is essential to let the plant settle before taking any measurements, the plant model must not use an integrator).

The purely electronic running summer was preferred to the electro-mechanical capacitor-drum device, as it involves no moving parts or rubbing contacts. When several running summers are required, the electro-mechanical device may prove simpler, as only one motor and synchronising circuit are needed, with several drums mounted on one shaft. However, the advent of large scale integration of microcircuits will probably encourage the use of purely electronic running summers for all purposes.

APPENDIX A2.2

POWER SPECTRUM ANALYSIS

The power spectrum of the output of the multiplier in the adaptive loop will now be derived, with reference to Figure A2.2. The impulse response of the running summer is

$$h_r(\tau) = u_1(\tau) - u_1(\tau - T_r)$$

where $u_1(\tau)$ is a unit step applied at $\tau = 0$.

Transforming to the frequency domain gives

$$H_r(jw) = \int_{-\infty}^{+\infty} h_r(\tau)\, \varepsilon^{-jw\tau}\, d\tau$$

$$\therefore\ H_r(jw) = \frac{1}{jw}\,(1 - \varepsilon^{-jwT_r}) \qquad \text{(A2.2-1)}$$

$$\text{Now } \Phi_r(w) = \Phi_c(w).H_r(jw).H_r^{*}(jw) \qquad \text{(A2.2-2)}$$

If the code has values $\pm \frac{1}{2}$, and the d.c. bias is ignored,

$$\Phi_c(w) = \frac{N+1}{N}\,\frac{\sin^2\left(\frac{\pi i}{N}\right)}{\left(\frac{\pi i}{N}\right)^2}\,\delta(w - iw_o) \qquad \text{(A2.2-3)}$$

for i = 1,2,3 etc.

Combining equations (A2.2-1), (A2.2-2) and (A2.2-3) gives

$$\Phi_r(w) = \frac{N+1}{2N}\,T_r^2.\,\frac{\sin^2\left(\frac{T_r\pi i}{T}\right)}{\left(\frac{T_r\pi i}{T}\right)^2}\cdot\frac{\sin^2\left(\frac{\pi i}{N}\right)}{\left(\frac{\pi i}{N}\right)^2}\,\delta(w - iw_o) \qquad \text{(A2.2-4)}$$

Suppose y(t) is multiplied by $c_s \sin w_s t$; the spectrum of the resulting signal is

$$\Phi_m^1(w) = \frac{c_s^2}{4}\left\{\Phi_y(w + w_s) + \Phi_y(w - w_s)\right\} \qquad \text{(A2.2-5)}$$

Hence when y(t) is multiplied by r(t), the spectrum is

$$\Phi_m(w) = \frac{N+1}{4N}\sum_{i=1}^{\infty}\frac{\sin^2\left(\frac{T_r\pi i}{T}\right)}{\left(\frac{T_r\pi i}{T}\right)^2}\cdot\frac{\sin^2\left(\frac{\pi i}{N}\right)}{\left(\frac{\pi i}{N}\right)^2}\cdot\left\{\Phi_y(w+iw_o) + \Phi_y(w-iw_o)\right\} \qquad \text{(A2.2-6)}$$

The analysis can be continued, by substituting for $\Phi_y(w)$, but rapidly becomes unmanageable.

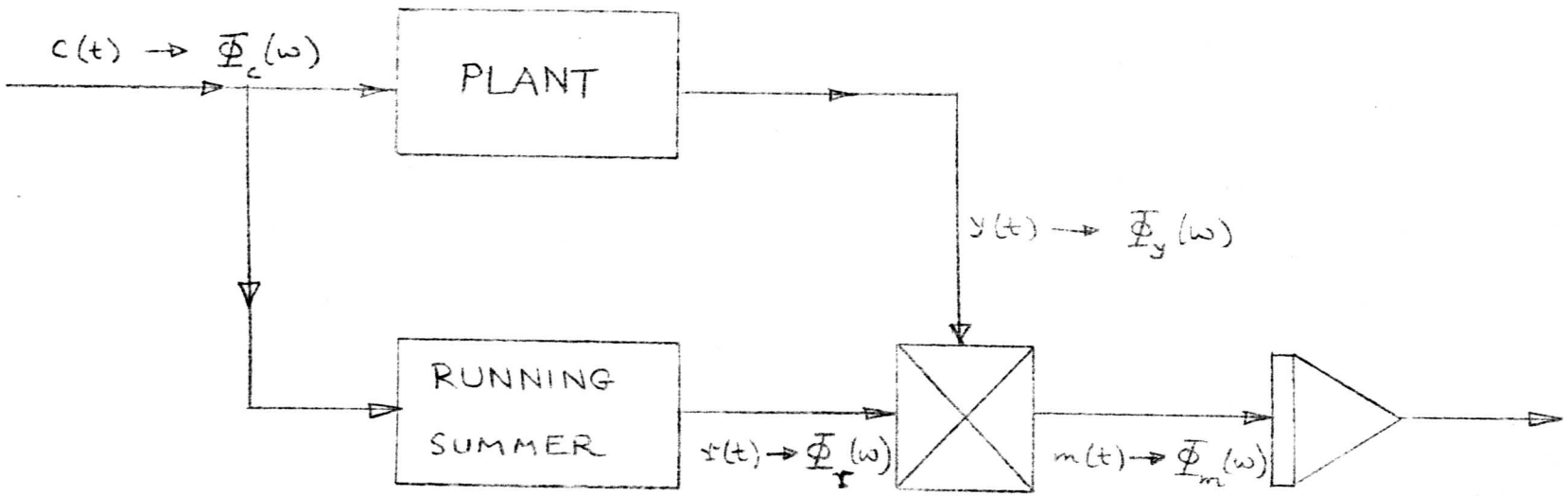


FIGURE A2.2: DIAGRAM FOR POWER SPECTRUM ANALYSIS

APPENDIX A2.3

Auto-Correlation Function of a Product

Laning and Battin[52] have shown that, if X_i (i = 1 to 4) are random variables with a joint normal distribution,

$$E\left[X_1X_2X_3X_4\right] = E\left[X_1X_2\right] E\left[X_3X_4\right] + E\left[X_1X_3\right] E\left[X_2X_4\right] + E\left[X_1X_4\right] E\left[X_2X_3\right]$$

In terms of equation (2.39),

$X_1 = r(t)$, $X_2 = \nu(t)$, $X_3 = r(t+\tau)$, $X_4 = \nu(t+\tau)$.

If $r(t)$ and $\nu(t)$ are assumed to be uncorrelated,

$$E\left[X_1X_2X_3X_4\right] = E\left[X_1X_3\right] E\left[X_2X_4\right]$$

that is

$$\phi_{m_\nu m_\nu}(\tau) = \phi_{rr} \times \phi_{\nu\nu}(\tau)$$

APPENDIX A.2.4.

AUTO-CORRELATION FUNCTION OF THE OUTPUT OF THE RUNNING AVERAGER

The auto-correlation function of the output of the running averager will now be evaluated. Consider the system shown in Figure A2.4.1(a). Using the convolution integral,

$$y(t) = \int_0^{\infty} h(s)\ x(t - s)\ ds$$

Then

$$\phi_{yy}(\tau) = Lt_{T\to\infty}\ \frac{1}{2T} \int_{-T}^{+T} y(t)\ y(t+\tau)\ dt$$

$$= Lt_{T\to\infty}\ \frac{1}{2T} \int_{-T}^{+T} \int_{-\infty}^{+\infty} h(s_1)\ x(t-s_1)\, ds_1 \int_{-\infty}^{+\infty} h(s_2)\ x(t+\tau-s_2)\, ds_2\, dt$$

Interchanging the order of integration gives

$$\phi_{yy}(\tau) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \phi_{xx}(\tau - s_2 + s_1)\ h(s_1)\ h(s_2)\ ds_1\ ds_2 \qquad \text{.....(A2.4.1.)}$$

This may be integrated directly. An alternative, and simpler, approach is to consider an equivalent system, shown in Figure A.2.4.1(b). Pass $x_1(t)$ through the system, record the output, reverse it in time, and pass it through the system again.

Then $$x_2(t) = \int_{-\infty}^{+\infty} h(s_2)\, x_1\,(t-s_2)\ ds_2$$

$$x_4(t) = \int_{-\infty}^{+\infty} h(s_1)\, x_3\,(t-s_1)\ ds_1$$

Since $x_3(t) = x_2(-t)$ then

$$x_4(t) = \int_{-\infty}^{+\infty} h(s_1)\ x_2(t+ s_1)\ ds_1$$

if $x_2(t)$ is an even function.

Hence

$$x_4(t) = \int_{-\infty}^{+\infty} h(s_1) \int_{-\infty}^{+\infty} h(s_2)\ x_1\,(t + s_1 - s_2)\ ds_2\ ds_1 \qquad \text{.....(A2.4.2)}$$

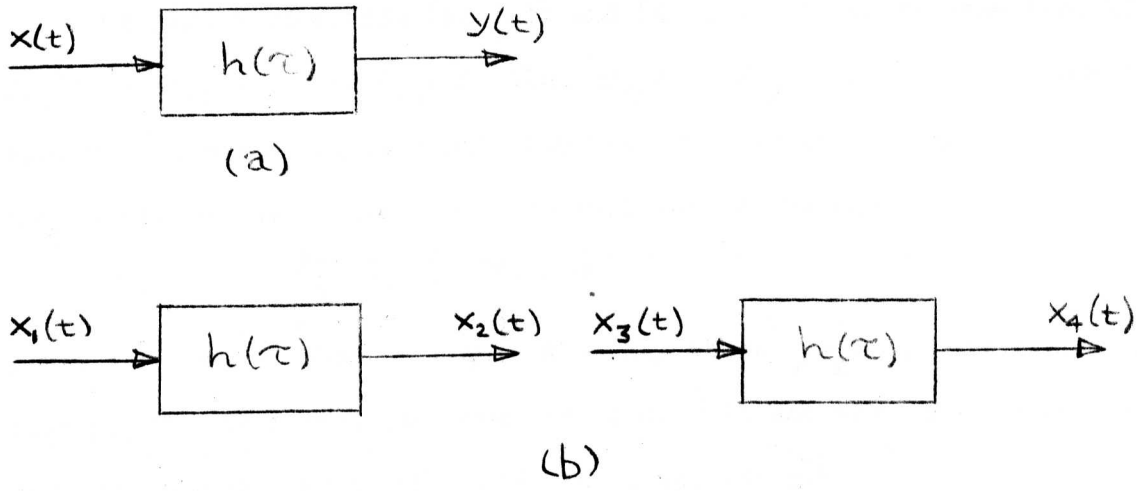


FIGURE A2.4.1:
SYSTEMS CONSIDERED IN APPENDIX A2.4

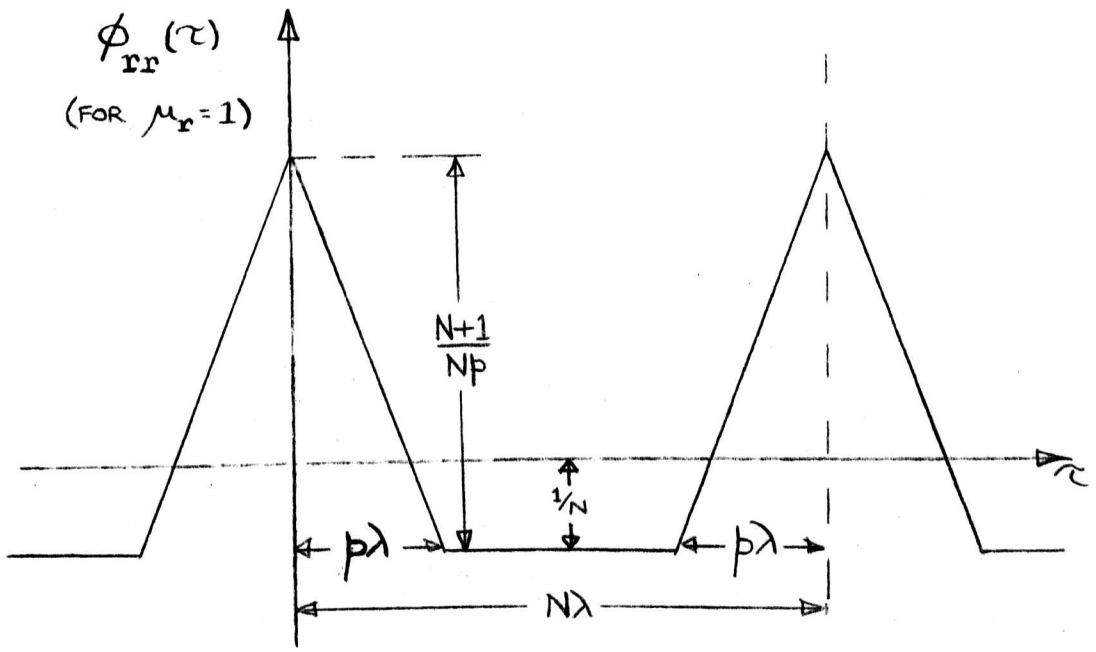


FIGURE A2.4.2: AUTO-CORRELATION FUNCTION OF OUTPUT OF RUNNING AVERAGER

Comparing equations (A2.4.1) and (A2.4.2) it can be seen that if $x_1(t)$ is equated with $\phi_{xx}(\tau)$ then $x_4(t) = \phi_{yy}(\tau)$. In this case the system is the running averager, and the input is the chain-code. Approximating the auto-correlation function of the code as

$$\phi_{cc}(\tau) = \frac{N+1}{N} \lambda \ \delta(\tau) - \frac{1}{N}$$

gives the result shown in figure A2.4.2., where $\mu_r = 1$. It can be seen that $\phi_{rr}(\tau)$ is a periodic even function. It has the same form for a running average time of $(N - p)\lambda$ as it has for $p\lambda$.

APPENDIX A2.5

THE NOISE GENERATOR

The noise was produced from a 20-stage shift-register, with feedback from the 17th and 20th stages to the 1st stage, with outputs of all stages summed in an operational amplifier. It can be shown that the output of any one stage is a pseudo-random sequence, and hence that the output of the amplifier has a Binomial amplitude probability distribution, which can be considered Gaussian for most purposes.

The clock-rate of the shift-register was 10kHz, so that the repitition period of the noise was over 100 seconds, which was very long compared with the time constants of the identification system. To eliminate low-frequency drift, a high-pass filter was used to cut off frequencies less than the fundamental frequency of the noise. The noise was then passed through a low-pass filter consisting of an operational amplifier with parallel capacitative and resistive feedback. In order to assess the power of the noise, and hence its auto-correlation function, it was passed through a square-law device and then integrated over one period. Dividing the result by the period then gave the mean-squared power of the noise.

APPENDIX A3.1

EFFECT OF D.C. ON THE ESTIMATE OF IMPULSE RESPONSE

Consider the system of Figure 3.1. The output is

$$\phi_{c\hat{y}}(\tau) = \frac{1}{T}\int_0^T c(t-\tau)\,\hat{y}(t)\,dt$$

where $\hat{y}(t)$ = estimated output of the plant.

If $\hat{y}(t) = y(t) + b$, where $y(t)$ is the output of the plant with the d.c. bias removed, and b is the bias, then

$$\phi_{c\hat{y}}(\tau) = \frac{1}{T}\int_0^T c(t-\tau)\,\left[y(t) + b\right]$$

$$= \phi_{cy}(\tau) + \frac{b}{T}\int_0^T c(t-\tau)$$

where $\phi_{cy}(\tau)$ is the correlation for the unbiassed case.

Now $\int_0^T c(t-\tau) = \lambda$ for all τ, so that

$$\phi_{c\hat{y}}(\tau) = \phi_{cy}(\tau) + \frac{b}{N}$$

Therefore a constant value must be subtracted from each ordinate of the estimated cross-correlation function to give the unbiassed cross-correlation function.

APPENDIX A3.2

DERIVATION OF RUNNING SUM METHOD OF IDENTIFICATION

Equation (3.1) may be expanded to give

$$\hat{S}(T_r) = \frac{T}{T+\lambda}\,\frac{1}{a\lambda}\int_{\tau=0}^{T_r}\left[\frac{1}{T}\int_{t=0}^{T} c(t-\tau)y(t)\,dt - \frac{T_r}{\Delta}\int_{S=T-\Delta}^{T}\frac{1}{T}\int_{t=0}^{T} c(t-s)y(t)\;dt\;ds\right] d\tau$$

Rearranging and interchanging the order of integration we have

$$\hat{S}(T_r) = \frac{1}{(T+\lambda)a\lambda}\int_{t=0}^{T} y(t)\left[\int_{\tau=0}^{T_r} c(t-\tau)\;d\tau - \frac{T_r}{\Delta}\int_{s=T-\Delta}^{T} c(t-s)\,ds\right] dt$$

Defining the running sum of c(t) over a period from q to r as

$$\sigma_c(q{:}r) = \int_{q}^{r} c(t)\;dt$$

and noting that $\sigma_c(q+T{:}r+T) \equiv \sigma_c(q{:}r)$ gives

$$\hat{S}(T_r) = \frac{1}{(T+\lambda)a\lambda}\int_{0}^{T} y(t)\left[\sigma_c(t-T_r{:}t) - \frac{T_r}{\Delta}\,\sigma_c(t{:}t+\Delta)\right] dt$$

The running sum is used here rather than the running average, as it is more appropriate for practical work.

APPENDIX A3.3

RELATIONSHIP BETWEEN SUCCESSIVE STEP RESPONSE ESTIMATES

Equation (3.5) can be written as

$$\hat{S}(p,j) = \eta \sum_{m=j-N+1}^{j} y(m)\rho(m)$$

so that

$$\hat{S}(p,j) = \hat{S}(p,j-1) + \eta\left[y(j)\rho(j) - y(j-N)\rho(j-N)\right]$$

But $\rho(j) = \rho(j-N)$

Therefore

$$\hat{S}(p,j) = \hat{S}(p,j-1) + \eta\left[y(j) - y(j-N)\right]\rho(j)$$

APPENDIX A3.4

ALTERNATIVE EXPRESSION FOR COMPOUND RUNNING SUM

We may write

$$\sigma_c(m+1:m+k) = \sum_{i=m+1}^{m+k} c(i)$$

$$= \sum_{i=m+1}^{m+N} c(i) - \sum_{i=m+k+1}^{m+N} c(i)$$

$$= 1 - \sigma_c(m+k+1:m+N)$$

so that the running sum over k bits is related to that over (N - k) bits, with a certain phase difference that must be taken into account for experimental work. Substituting in equation (A3.1) then gives

$$\hat{S}(p) = \frac{\eta}{k} \sum_{m=1}^{N} \left\{ y(m) \left[k\sigma_c(m - \overline{p-1}:m) + p\sigma_c(m+1-\overline{N-k}:m) - p \right] \right\} \quad \text{.....(A3.4-1)}$$

The advantages of this formula over equation (A3.1) are

(i) Both running sums have top-end points with the same phase; this is an advantage when using special-purpose hardware.

(ii) If several values of p and k are to be used, running sums will only have to be computed over large periods, since p and (N-k) will generally be large, instead of over one large and one small period, as is the case when using equation (3.5)

APPENDIX A4.1

AUTO-CORRELATION FUNCTION OF THE OUTPUT OF THE COMPOUND RUNNING SUMMER

The output of the compound running summer can be split into two components giving

$$r(t) = r_1(t) + r_2(t)$$

Then $$\phi_{rr}(\ell) = \phi_{r_1r_1}(\ell) + \phi_{r_2r_2}(\ell) + 2\phi_{r_1r_2}(\ell) \quad \text{.....(A4.1)}$$

The auto-correlation terms can be derived as in Appendix A2.4. The cross term is

$$\phi_{r_1r_2}(\ell) = -\frac{1}{N} \sum_{m=1}^{N} \left[\sum_{i=0}^{p-1} c(m-i) \right] \left[\frac{p}{k} \sum_{j=N-k}^{N-1} c(m-j+\ell) \right]$$

$$= \begin{cases} \frac{p^2}{N} - \frac{p}{k} \frac{N+1}{N} \sum_{i=0}^{p-1} 1 & \text{for } N-k-\ell \leq i \leq N-1-\ell \\ \frac{p^2}{N} & \text{otherwise} \end{cases}$$

assuming the width of the auto-correlation spike is negligible.

The shape of this function and the position of the 'breakpoints' depend on the relative values of the parameters. Three examples are shown in Figure A4.1.

It is possible to evaluate $\phi_{rr}(\ell)$ by substituting in equation (A4.1), but the resulting equation has too many and varied segments to be useful for a general analysis.

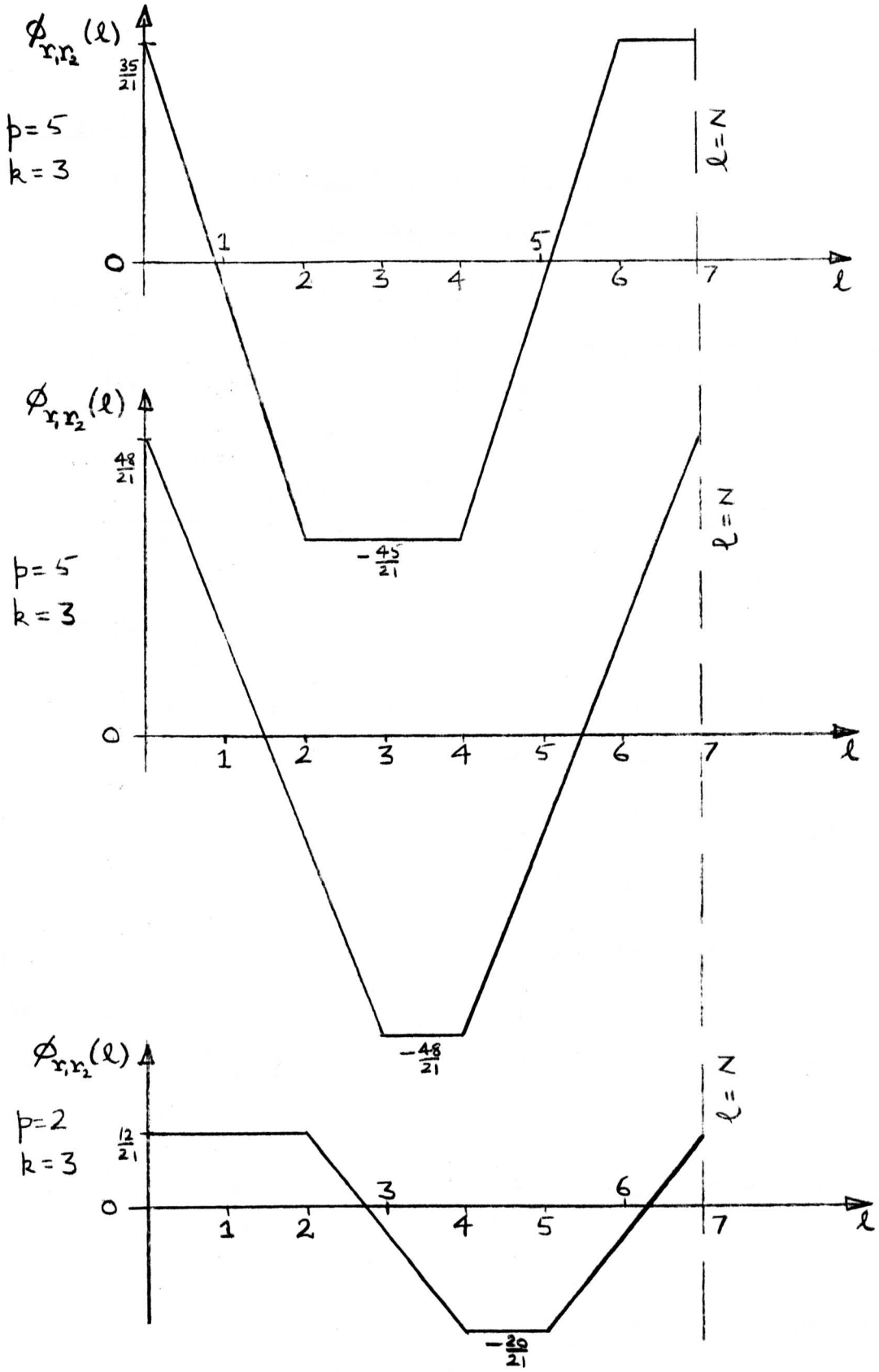


FIGURE A4.1:

CROSS-CORRELATION OF RUNNING SUMS (N = 7)

APPENDIX A4.2

VARIANCE DUE TO NOISE, IN TERMS OF AUTO-CORRELATION FUNCTIONS

From equations (4.1) and (4.2)

$$\hat{S}_{\nu}(T_r) = \left(\frac{T}{T+\lambda}\right) \frac{1}{a\lambda} \frac{1}{T} \int_{t=0}^{T} \nu(t) \left[\int_{\tau=0}^{T_r} c(t-\tau)\, d\tau - \frac{T_r}{\Delta} \int_{S=T-\Delta}^{T} c(t-s)\, ds \right] dt$$

so that, using equation (4.3),

$$D^2 = \frac{1}{[(T+\lambda)a\lambda]^2} E \left[\int_{t_1=0}^{T} \nu(t_1) \int_{\tau_1=0}^{T_r} c(t_1-\tau_1) \int_{t_2=0}^{T} (t_2) \int_{\tau_2=0}^{T_r} c(t_2-\tau_2) \right.$$

$$d\tau_2\, dt_2\, d\tau_1\, dt_1$$

$$+ \frac{T_r^2}{\Delta^2} \int_{t_1=0}^{T} (t_1) \int_{\tau_1=T-\Delta}^{T} c(t_1-\tau_1) \int_{t_2=0}^{T} (t_2) \int_{\tau_2=T-\Delta}^{T} c(t_2-\tau_2)\, d\tau_2 dt_2 d\tau_1 dt_1$$

$$\left. - \frac{2T_r}{\Delta} \int_{t_1=0}^{T} (t_1) \int_{\tau_1=0}^{T_r} c(t_1-\tau_1) \int_{t_2=0}^{T} (t_2) \int_{\tau_2=t-\Delta}^{T} c(t_2-\tau_2)\, d\tau_2 dt_2 d\tau_1 dt_1 \right]$$

Assuming $\nu(t)$ and $c(t)$ are uncorrelated, using the result of Appendix A2.3 gives

$$D^2 = \mu^2 \left[\int_{t_1=0}^{T} \int_{\tau_1=0}^{T_r} \int_{t_2=0}^{T} \int_{\tau_2=0}^{T_r} \phi_{\nu\nu}(t_1-t_2) \phi_{cc}(t_1-\tau_1-t_2+\tau_2)\, d\tau_2 dt_2 d\tau_1 dt_1 \right.$$

$$+ \frac{T_r^2}{\Delta^2} \int_{t_1=0}^{T} \int_{\tau_1=T-\Delta}^{T} \int_{t_2=0}^{T} \int_{\tau_2=T-\Delta}^{T} \phi_{\nu\nu}(t_1-t_2) \phi_{cc}(t_1-\tau_1-t_2+\tau_2)\, d\tau_2\, dt_2\, d\tau_1\, dt_1$$

$$\left. - \frac{2T_r}{\Delta} \int_{t_1=0}^{T} \int_{\tau_1=0}^{T_r} \int_{t_2=0}^{T} \int_{\tau_2=T-\Delta}^{T} \phi_{\nu\nu}(t_1-t_2) \phi_{cc}(t_1-\tau_1-t_2+\tau_2)\, d\tau_2 dt_2 d\tau_1 dt_1 \right]$$

APPENDIX A4.3

EVALUATION OF A PARTICULAR DOUBLE INTEGRAL

Consider the integral

$$I = \int_0^T \int_0^T f(t_1 - t_2)\, \delta(t_1 - t_2 - \tau)\, dt_1 dt_2$$

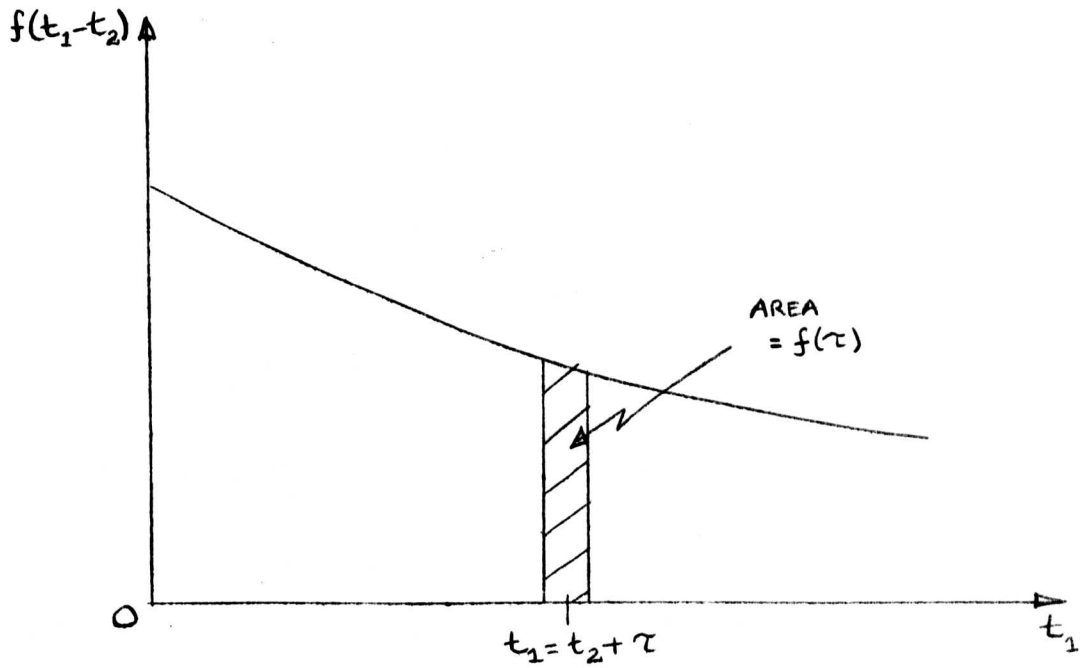
where f is any function.

The integrand can be expressed as a function of t_1, as shown in Figure A4.3(a), and it is seen that

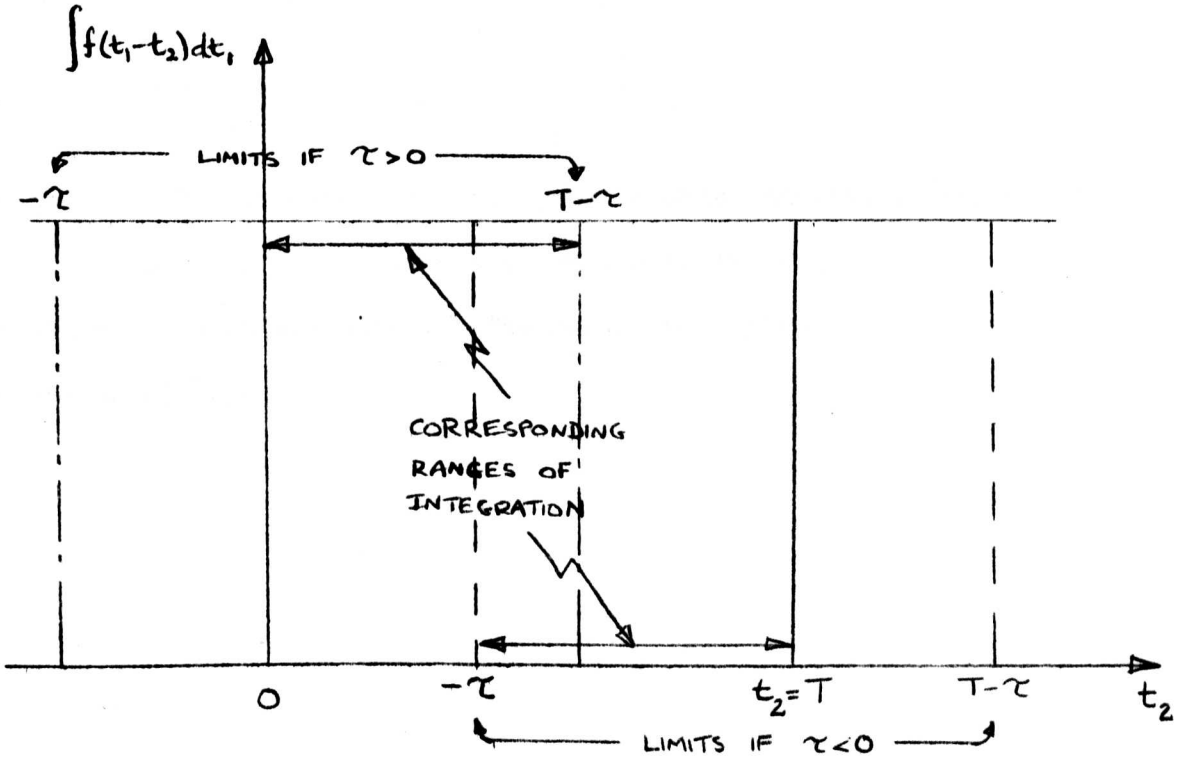
$$\int_0^T f(t_1 - t_2)\, \delta(t_1 - t_2 - \tau)\, dt_1 = \begin{cases} f(\tau) & \text{if } 0 < t_2 + \tau < T \\ 0 & \text{otherwise} \end{cases}$$

This is independent of t_2, and is shown in Figure A4.3(b). Possible positions for the limits of the function are shown, together with the corresponding ranges of integration. It is clear then that

$$I = (T - |\tau|)\, f(\tau) \qquad |\tau| < T$$



(a)



(b)

FIGURE A4.3:

GRAPHICAL REPRESENTATIONS FOR APPENDIX A4.3

APPENDIX A4.4

EVALUATION OF EQUATION (4.4) FOR BAND-LIMITED NOISE

f_ν, from equation (4.9), can be substituted into equation (4.4). Putting $\tau = \tau_1 - \tau_2$ then gives

$$D^2 = \frac{\sigma_\nu^2}{a^2} \frac{1}{(T+\lambda)T\lambda} \left[\int_0^{T_r} \int_{-\tau_2}^{T_r-\tau_2} \varepsilon^{-\frac{|\tau|}{T}} (T-|\tau|) d\tau d\tau_2 \right.$$
$$\left. + \text{two similar terms} \right]$$

To evaluate this expression, consider the graphical representation of the limits for the first double integral shown in Figure A4.4. From this it is seen that the first double integral is

$$\int_0^{T_r} \left[\int_{-\tau_2}^{0} \varepsilon^{\frac{\tau}{T_\nu}} (T+\tau) d\tau + \int_0^{T_r-\tau_2} \varepsilon^{-\frac{\tau}{T_\nu}} (T-\tau)\, d\tau \right] d\tau_2$$
$$= T_\nu \left[2TT_r - 2T_\nu T_r - 2TT_\nu + 4T_\nu^2 \right.$$
$$\left. + (2TT_\nu - 4T_\nu^2 - 2T_\nu T_r)\, \varepsilon^{-\frac{T_r}{T_\nu}} \right]$$

Repeating this procedure for the other double integrals, noting that the last one has two solutions depending on the sign of $(T_r - T + \Delta)$, enables D^2 to be evaluated. The resulting expression is given in equation (4.10).

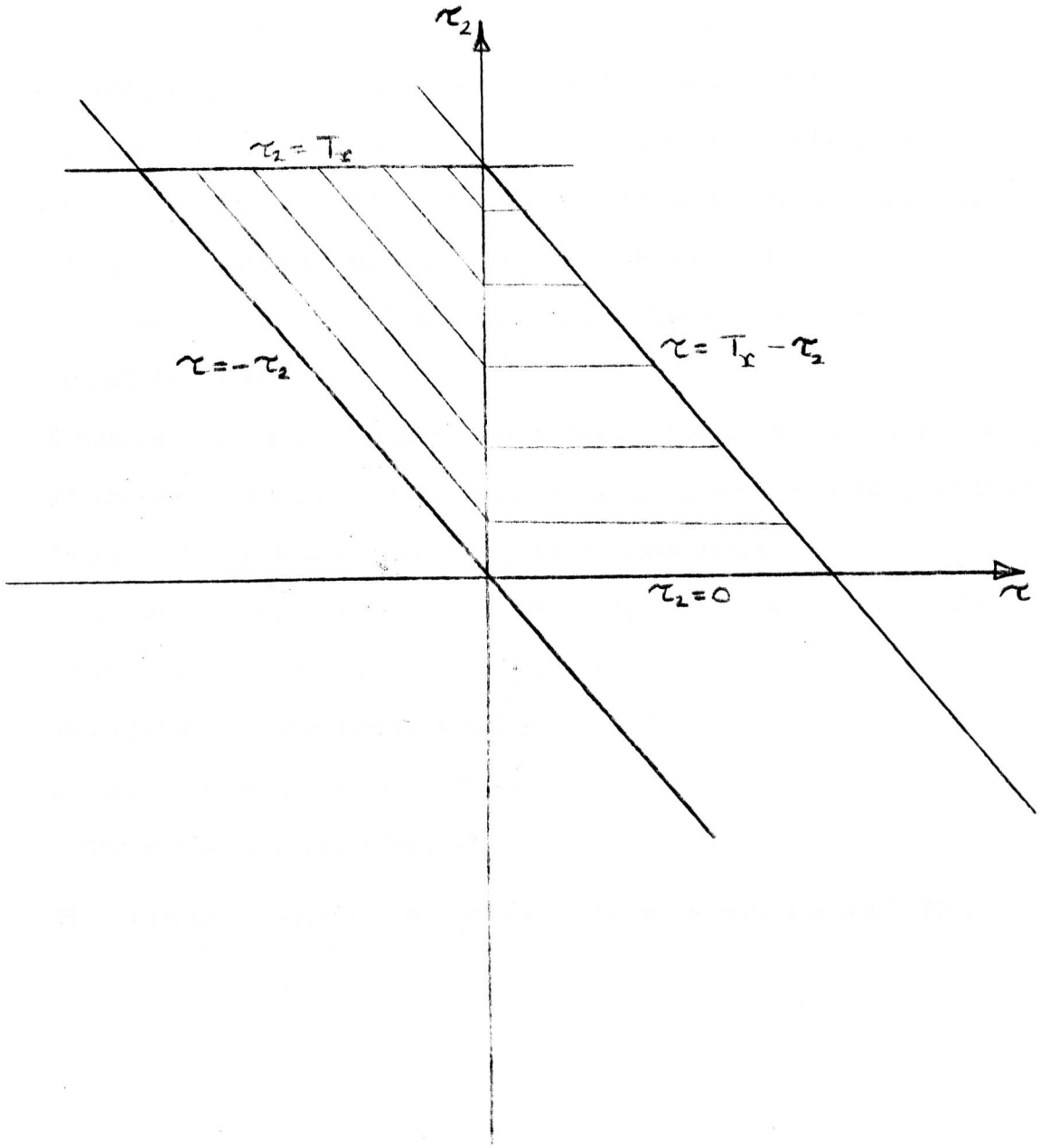


FIGURE A4.4:

LIMITS OF INTEGRATION FOR APPENDIX A4.4

APPENDIX A9.1

Facilities Available on the G.E.C.92 Computer

The hardware of the computer includes:

(i) A priority-interrupt system, by which the program can be interrupted, while running, by a signal generated internally or externally. When such a signal is detected, control is routed to a pre-assigned location, providing the latest interrupt signal is of higher priority then the routine being serviced.

(ii) An internal clock, which generates interrupt signals every 20 milliseconds.

(iii) 6 analogue outputs. These have a resolution of 8 bits plus sign.

(iv) 24 analogue inputs. These have a resolution of 10 bits plus sign.

(v) Paper tape input and output; control typewriter.

(vi) 4 externally-set breakpoint switches, which can be tested in the program and control routed accordingly.

(vii) Two arithmetic registers, A and B.

(viii) Indirect and indexed addressing.

(ix) 8,000 words of 12 bits each.

The program is written in a symbolic machine code called SYMBOL.

APPENDIX A9.2

PROGRAM STORAGE ALLOCATION AND LISTING

Storage	
Data and Working Stores	*Preset values of program parameters
	Working stores (indices and running totals)
	Interrupt locations
	Working stores (variable arrays)
	*Address blocks
	*Indirect-address stores
	Chain-code stores.
Executable Program (Setting up Procedure)	*Setting up of addresses
	Clearance of working stores
	Chain-code generation
	*Setting up of counters
Executable Program (On-line Routine)	Setting up addresses for code output
	Code output
	Evaluation of change in y
	Update of ϕ_{cy} and bias removed
	Setting up addresses for feedback routine
	Step response and feedback calculations
	‡Update of dynamic compensation array
	‡Extraction of latest value for compensation
	Update of indices
Display Routines	Display with interrupts enabled
	Display with program frozen

Subroutines
- P.R.B.S. - chain-code generation
- INPUT - analogue input routine
- SCOPE - oscilloscope display routine

Extra store Feedback graphs.

Note:-

(1) If the progress displays (SST and KST) are omitted, 4000 (octal) words of store are saved.

(2) If dynamic compensation is not required, items marked above with ** can be omitted.

(3) If dynamic allocation of storage is not required, part or all of each item marked with * may be omitted.

(4) If the display and dynamic compensation routines are omitted, the on-line routine can be reduced to well under 300 words of executable statements. (The remaining statements are serviced once only, at the start of the program.)

```
                    *
                    *OPTIM4 - OPTIMIZATION OF 4 CHANNELS USING SHIFTED CODES
                    *
00000                REF PRBS,INPUT,SCOPE [CODE GENERATION,ANALOG INPUT,DISPLAY]
00100                AORG 0100
00100 0104          FB DATA 0104 [FEEDBACK FOR SHIFT-REGISTER]
00101 0177          LENGTH DATA 0177 [LENGTH OF CODE]
00102 0040          LEN4 DATA 040 [NO. OF BITS FOR CHANNEL 4]
00103 0040          LEN3 DATA 040
00104 0040          LEN2 DATA 040
00105 0037          LEN1 DATA 037
00106 0017          DELTA DATA 017 [NO. OF BITS TO AVERAGE OVER FOR D.C.]
00107 0400          SC DATA 0400,0400,0400,0400 [FEEDBACK SCALE FACTORS]
      0400
      0400
      0400
                    *
00113               DLEN RES 1 [LENGTH MINUS 1]
00114               YC RES 1 [DYNAMIC COMPENSATION]
00115               CH RES 1 [CHANNEL NO.]
00116               SR RES 2 [D.C. AND STEP RESPONSE RUNNING TOTALS]
00120               K RES 4 [FEEDBACK RUNNING TOTALS - MOST SIGNIFICANT HALVES]
00124               KP RES 4 [DITTO - LEAST SIG. HALVES]
00130               DK RES 4 [CHANGE IN FEEDBACK OVER PAST BIT]
00134               NK RES 4 [FEEDBACK - SCALED DOWN]
00140               I RES 4 [INDICES FOR CODE OUTPUT]
                    *
00200                AORG 0200
* 00200 7700         BRM TICK [CLOCK INTERRUPT LOCATION]
       0000
00202                RES 2
* 00204 7700         BRM INPUT [END-OF-CONVERT INTERRUPT LOCATION]
       0000
                    *
00206                RES 511
01204               Y EQU $-1 [SYSTEM OUTPUTS OVER ONE PERIOD]
01205                RES 511
02203               BIPHI EQU $-1 [IMPULSE RESPONSES]
02204                RES 511
03202               S EQU $-1 [STEP RESPONSES]
03203                RES 256
03602               COMP EQU $-1 [DYNAMIC COMPENSATION ARRAY]
03603                RES 2048
07602               SST EQU $-1 [GAINS OF EACH CHANNEL OVER PAST 512 BITS]
17777               KST EQU 017777 [FEEDBACKS DITTO]
```

```
07603          PAGE 2
               *ADDRESS BLOCK FOR FEEDBACK CALCULATIONS
07603 2203     BIPHI4 DATA BIPHI [ADDRESS OF TOP END OF IMP. RESP. OF CHAN.4]
07604          BIPHI3 RES 1
07605          BIPHI2 RES 1
07606          BIPHI1 RES 1
07607 3202     S4 DATA S [DITTO FOR STEP RESP.]
07610          S3 RES 1
07611          S2 RES 1
07612          S1 RES 1
07613 7602      DATA SST,SST-512,SST-1024,SST-1536,KST,KST-512,KST-1024,KST-1536
      6602
      5602
      4602
      7777
      6777
      5777
      4777
07623 3202     S4P DATA S [IDENTICAL TO S4]
07624          S3P RES 1
07625          S2P RES 1
07626          S1P RES 1
07627 0120      DATA K,K+1,K+2,K+3,KP,KP+1,KP+2,KP+3,DK,DK+1,DK+2,DK+3
      0121
      0122
      0123
      0124
      0125
      0126
      0127
      0130
      0131
      0132
      0133
07643          DLEN4 RES 4 [NO.OF BITS PER CHANNEL MINUS 1]
07647 0107     SC4 DATA SC,SC+1,SC+2,SC+3
      0110
       0111
      0112
07653 0134     NK4 DATA NK,NK+1,NK+2,NK+3
      0135
      0136
      0137
               *ADDRESS BLOCK FOR CODE OUTPUT
07657 0137      DATA NK+3,06003
      6003
07661 0143     LIST DATA I+3
```

```
07662          PAGE 3
               *INDIRECT-ADDRESS STORES
07662 0010     XBIPHI DATA 010,0
      0000
07664 0010     XS DATA 010,0
      0000
07666 0010     XSST DATA 010,0
      0000
07670 0011     XKST DATA 011,0
      0000
07672 0000     IS DATA 0,0
      0000
07674 0000     IK DATA 0,0
      0000
07676 0000     IKP DATA 0,0
      0000
07700 0000     IDK DATA 0,0
      0000
07702          IDLEN RES 2
07704 0000     ISC DATA 0,0
      0000
07706 0000     INK DATA 0,0
      0000
07710 0003     ECM DATA 3,0
      0000
07712 0000     II DATA 0,0
      0000
               *CODE IN +1/-1 AND +01000/-01000 FORMS
07714           RES 511
10712          CC EQU $-1
10713           RES 511
11711          CODE EQU $-1
```

```
11712          PAGE 4
              *STARTING ROUTINE
11712 0050    START DIR
11713 0041     HLT
11714 6440     LDA =3
      0003
11716 4400     STA CH
      0115
11720 2410     LDB LEN1,1
      0105
11722 2040     SUB =1
      0001
11724 0410     STB DLEN4+3,1
      7646
11726 7001     BDA $-6
      1720
11730 2400     LDB LENGTH
      0101
11732 2040     SUB =1
      0001
11734 0400     STB DLEN
      0113
11736 2000     SUB DLEN4
      7643
11740 0400     STB I
      0140
11742 2000     SUB LEN3
      0103
11744 0400     STB I+1
      0141
11746 2000     SUB LEN2
      0104
11750 0400     STB I+2
      0142
11752 2000     SUB LEN1
      0105
11754 0400     STB I+3
      0143
11756 0640     CEB =0 [CHECK INPUT DATA FOR CONSISTENCY]
      0000
11760 3101     BFF $+3
      1763
11762 0041     HLT
11763 2440     LDB =1
      0001
* 11765 0400    STB INDEX
      0000
              *
```

```
 11767                PAGE 5
                     *SCALING FOR STEP RESPONSE AND D.C.
 11767 6400          LDA LEN1
       0105
 11771 2440          LDB =0267
       0267
 11773 4740          CMA =33
       0041
 11775 3101          BFF $+10
       2007
 11777 2240          ADB =1
       0001
 12001 4740          CMA =65
       0101
 12003 3101          BFF $+4
       2007
 12005 2240          ADB =1
       0001
*12007 0400          STB SB
       0000
                     *
 12011 6400          LDA DELTA
       0106
 12013 2440          LDB =0271
       0271
 12015 4640          CEA =63
       0077
 12017 3101          BFF $+10
       2031
 12021 2040          SUB=1
       0001
 12023 4640          CEA=31
       0037
 12025 3101          BFF $+4
       2031
 12027 2040          SUB=1
       0001
*12031 0400          STB AL
       0000
                     *
```

```
12033          PAGE 6
              *ADDRESSES OF IMPULSE AND STEP RESPONSES
12033 6400     LDA BIPHI4
      7603
12035 6000     SUA LEN4
      0102
12037 4400     STA BIPHI3
      7604
12041 6000     SUA LEN3
      0103
12043 4400     STA BIPHI2
      7605
12045 6000     SUA LEN2
      0104
12047 4400     STA BIPHI1
      7606
12051 6400     LDA S4
      7607
12053 6000     SUA LEN4
      0102
12055 4400     STA S3
      7610
12057 4400     STA S3P
      7624
12061 6000     SUA LEN3
      0103
12063 4400     STA S2
      7611
12065 4400     STA S2P
      7625
12067 6000     SUA LEN2
      0104
12071 4400     STA S1
      7612
12073 4400     STA S1P
      7626
              *
```

```
12075                PAGE 7
                    *CLEAR STORES
12075 2440    LDB =0
      000000000
12077 6440    LDA =3836
      7374
12101 0410    STB SST,1
      7602
12103 7001    BDA $-2
      2101
12105 6440    LDA =2047
      3777
12107 0411    STB KST,1
      7777
12111 7001    BDA $-2
      2107
12113 6440    LDA =510
      0776
12115 2440    LDB =1
      0001
12117 0411    STB CC,1
      0712
12121 2440    LDB =01000
      1000
12123 0411    STB CC,1
      0712
12121 2440    LDB =01000
      1000
12123 0411    STB CODE,1
      1711
12125 7001    BDA $-8
      2115
```

```
  12127                  PAGE 8
                        *GENERATE CODES
  12127 2400    0100     LDB FB
* 12131 0400    0000     STB AD
  12133 2440    0001     LDB =1
  12135 0401    2146     STB $+9
  12137 6400    0113     LDA DLEN
  12141 4401    2150    AA STA $+7
* 12143 7700    0000     BRM PRBS
  12145 0000    0001    AD DATA 0,1
  12147 6440    0000     LDA =0
  12151 0540    0001     COB =1
  12153 3101    2165     BFF $+10
  12155 2440    7777     LDB =-1
  12157 0411    0712     STB CC,1
  12161 2440    7000     LDB =-01000
  12163 0411    1711     STB CODE,1
  12165 7001    2141     BDA AA
                        *SET UP COUNTERS
  12167 6440    0377     LDA =255
* 12171 4400    0000     STA DC
  12173 6440    0777     LDA =511
  12175 4440    0000     STA =0
  12176                  TIME EQU $-1
                        *ENABLE INTERRUPTS AND GO TO DISPLAY
  12177 0051             EIR
* 12200 7300    0000     BRU BP1
                        *
```

```
12202          PAGE 9
               *ACTION ON RECEIVING INTERRUPT
12202          TICK RES 2
12204 0003      EOM 030021
      0021
12206 3221      BRC *TICK
      2202
               *RETURN FROM INPUT
12210 0003     IPDONE EOM 036004 [MOVE DISPLAY SPOT TO EDGE OF SCREEN]
      6004
12212 1040      POT =03770
      3770
12214 0003      EOM 037000
      7000
12216 0440      STB =0
      0000
               *OUTPUT ROUTINE: SETTING UP ADDRESSES
12220 6400      LDA CH
      0115
12222 4400     SETAD1 STA CH
      0115
12224 6440      LDA =2
      0002
12226 2410      LDB LIST,1
      7661
12230 2000      SUB CH
      0115
12232 4256      CYA 1
12233 0410      STB II+1,1
      7713
12235 4244      CYA 11
12236 7001      BDA $-8
      2226
               *
```

```
12240          PAGE 10
              *OUTPUT CODES WITH FEEDBACK
12240 6420     LDA *II
      7712
12242 2411     LDB CODE,1
      1711
12244 0146     BPT 3
12245 3101     BFF $+4 [TEST IF FEEDBACK TO BE ADDED-IN]
      2251
12247 2220     ADB *INK [IF HILL INVERTED, CHANGE TO SUB *INK]
      7706
12251 0440     STB =0
      0000
12253 7200     EXU ECM
      7710
12255 1001     POT $-3
      2252
12257 7001     BDA $+4
      2263
12261 6400     LDA DLEN
      0113
12263 4420     STA *II
      7712
              *TEST IF ALL CHANNELS OUTPUT
12265 6400     LDA CH
      0115
12267 7001     BDA SETAD1
      2222
12271 6440     LDA =3
      0003
12273 4400     STA CH
      0115
              *
12275 0003     ECM 037000
      7000
```

```
 12277          PAGE 11
                *FIND CHANGE IN Y
 12277 6440     LDA =0
       0000
 12300          INDEX EQU $-1
*12301 4400     STA IA
       0000
 12303 2401     LDB IPDONE+7
       2217
 12305 0147     BPT 4 [TEST IF DYNAMIC COMP. TO BE ADDED IN]
 12306 3101     BFF $+4
       2312
 12310 2200     ADB YC [IF HILL INVERTED, CHANGE TO SUB YC]
       0114
 12312 3410     XMB Y,1
       1204
 12314 2010     SUB Y,1
       1204
 12316 0440    AK STB =0
       0000
 12320 2640     EOB =-1
       7777
 12322 2240     ADB =1
       0001
 12324 0440     STB =0
       0000
```

```
12326          PAGE 12
               *IMPULSE RESPONSE
12326 6400     LDA DLEN
      0113
12330 7440     XMA =0
      0000
12331          IA EQU $-1
12332 7001     BDA $+4
      2336
12334 6400     LDA DLEN
      0113
12336 2411     LDB CC,1
      0712
12340 0540     COB =04000 [TEST IF CODE +VE OR -VE]
      4000
12342 2401     LDB AK+1
      2317
12344 3101     BFF $+4
      2350
12346 2401     LDB AK+7
      2325
12350 7401     XMA IA
      2331
12352 4440     STA =0
      0000
12354 6400     LDA DLEN
      0113
12356 6001     SUA $-3
      2353
12360 3610     MPB BIPHI,1 [UPDATE IMPULSE RESPONSE]
      2203
12362 6401     LDA $-7
      2353
12364 7001     BDA IA-1
      2330
```

```
12366          PAGE 13
              *REMOVE DC
12366 6440     LDA =0
      0000
12370 4400     STA SR
      0116
12372 4400     STA SR+1
      0117
12374 6400     LDA DELTA
      0106
12376 2410    BH LDB BIPHI,1
      2203
12400 3600     MPB SR [SIGNED ADDITION TO DOUBLE-LENGTH NUMBER]
      0116
12402 5600     MPF SR+1
      0117
12404 0540     COB =04000
      4000
12406 7101     BFT $+6
      2414
12410 2440     LDB =-1
      7777
12412 3600     MPB SR+1
      0117
12414 7001     BDA BH
      2376
12416 6400     LDA SR [SCALE D.C.]
      0116
12420 2400     LDB SR+1
      0117
12422 0277    AL CYD 0
12423 2640     EOB =-1
      7777
12425 2240     ADB =1
      0001
12427 6400     LDA DLEN
      0113
12431 3610     MPB BIPHI,1 [ADD MINUS D.C. TO IMP. RESP.]
      2203
12433 7001     BDA $-2
      2431
               *
```

```
12435          PAGE 14
              *CALCULATION OF SYSTEM GAIN AND DESIRED FEEDBACK: ADDRESS SET-UP
12435 6400     LDA CH
      0115
12437 4400    SETAD2 STA CH
      0115
12441 6440     LDA =10
      0012
12443 4255     CYA 2
12444 6200     ADA CH
      0115
12446 2410     LDB NK4+3,1
      7656
12450 6000     SUA CH
      0115
12452 4244     CYA 11
12453 0410     STB INK+1,1
      7707
12455 4244     CYA 11
12456 7001     BDA SETAD2+4
      2443
              *STEP RESPONSE
12460 2440     LDB =0
      0000
12462 0400     STB SR
      0116
12464 0400     STB SR+1
      0117
12466 6400     LDA IDLEN+1 [NO. OF BITS / CHAN. -1]
      7703
12470 4440    SA STA =0
      0000
12472 2420     LDB *XBIPHI [SUM ORDINATES OF BIPHI]
      7662
12474 3600     MPB SR
      0116
12476 5600     MPF SR+1
      0117
12500 0540     COB =04000
      4000
12502 7101     BFT $+6
      2510
12504 2440     LDB =-1
      7777
12506 3600     MPB SR+1
      0117
```

```
12510          PAGE 15
12510 6400     LDA SR [SCALE STEP RESPONSE]
      0116
12512 2400     LDB SR+1
      0117
12514 0277   SB CYD 0
12515 6401     LDA SA+1
      2471
12517 0420     STB *XS
      7664
12521 7001     BDA SA
      2470
             *FEEDBACK
12523 2420     LDB *IS [ADD GAIN TO K,KP]
      7672
12525 6401     LDA TIME
      2176
12527 0420     STB *XSST [STORE GAIN IN SST]
      7666
12531 3620     MPB *IKP
      7676
12533 5620     MPF *IK
      7674
12535 0540     COB =04000
      4000
12537 7101     BFT $+6
      2545
12541 2440     LDB =-1
      7777
12543 3620     MPB *IK
      7674
             *
12545 0146     BPT 3 [TEST IF FEEDBACK REQUIRED]
12546 7101     BFT $+10
      2560
12550 2440     LDB =0 [CLEAR FEEDBACK STORES]
      0000
12552 6440     LDA =7
      0007
12554 0410     STB KP+3,1
      0127
12556 7001     BDA $-2
      2554
             *
```

```
12560          PAGE 16
12560 6420     LDA *IK [SCALE UP FEEDBACK IF NECESSARY]
      7674
12562 2420     LDB *IKP
      7676
12564 0275     CYD 2
12565 4540     COA =04000 [SCALE-DOWN FEEDBACK IF NECESSARY]
      4000
12567 7101     BFT $+4
      2573
12571 6220     ADA *ISC
      7704
12573 5220     DVA *ISC
      7704
12575 6401     LDA TIME
      2176
12577 0420     STB *XKST
      7670
12601 3420     XMB *INK [STORE FEEDBACK IN NK]
      7706
12603 2020     SUB *INK
      7706
12605 0420     STB *IDK [STORE CHANGE IN FEEDBACK OVER PAST BIT IN DK]
      7700
               *DYNAMIC COMPENSATION ARRAY
12607 2440     LDB =0
      0000
12610          DC EQU $-1
* 12611 0400    STB DB
      0000
12613 6400     LDA IDLEN+1
      7703
12615 4440     DA STA =0
      0000
12617 2420     LDB *XS [SIGNED-MULTIPLICATION OF S BY DK]
      7664
12621 0440     STB =0
      0000
12623 0540     COB =04000
      4000
12625 5320     MUB *IDK
      7700
12627 7101     BFT $+4
      2633
12631 6020     SUA *IDK
      7700
```

```
12633          PAGE 17
12633 5720     LDF *IDK
      7700
12635 3101     BFF $+4
      2641
12637 6001     SUA $-13
      2622
12641 4440     STA =0
      0000
12643 0440     STB =0
      0000
12645 6440     LDA =0
      0000
12646          DB EQU $-1
12647 3610     MPB COMP,1 [ADD S*DK TO COMP (DOUBLE-LENGTH)]
      3602
12651 2401     LDB DB-4
      2642
12653 2340     ACB =0
      0000
12655 7001     BDA $+2
      2657
12657 3610     MPB COMP,1
      3602
12661 7001     BDA $+4
      2665
12663 6440     LDA =255
      0377
12665 4401     STA DB
      2646
12667 6401     LDA DA+1
      2616
12671 7001     BDA DA
      2615
12673 6401     LDA DB [ADD FINAL VALUE OF S*DK TO REST OF COMP]
      2646
12675 4601     CEA DC [TEST IF ALL ADDED]
      2610
12677 2401     LDB DB-2
      2644
12701 7101     BFT DB+1
      2647
```

```
12703          PAGE 18
               *TEST IF FEEDBACK FOUND FOR ALL CHANNELS
12703 6400     LDA CH
      0115
12705 7001     BDA SETAD2
      2437
12707 6440     LDA =3
      0003
12711 4400     STA CH
      0115
               *DYNAMIC COMPENSATION : PRESENT VALUE
12713 0147     BPT 4 [TEST IF COMP TO BE ZEROED]
12714 7101     BFT $+10
      2726
12716 6440     LDA =255
      0377
12720 2440     LDB =0
      0000
12722 0410     STB COMP,1
      3602
12724 7001     BDA $-2
      2722
12726 6401     LDA DC [FIND AND SCALE PRESENT VALUE OF COMP]
      2610
12730 2410     LDB COMP,1 [LEAST SIG. HALF]
      3602
12732 7001     BDA $+2
      2734
12734 6410     LDA COMP,1
      3602
12736 0273     CYD 4
12737 4400     STA YC
      0114
12741 6401     LDA DC
      2610
12743 6240     ADA =1
      0001
12745 4640     CEA =256
      0400
12747 7101     BFT $+4
      2753
```

```
12751          PAGE 19
12751 6440     LDA =0
      0000
12753 2410     LDB COMP,1
      3602
12755 0440     STB =0
      0000
12757 6240     ADA =1
      0001
12761 2410     LDB COMP,1
      3602
12763 6401     LDA DC
      2610
12765 0410     STB COMP,1
      3602
12767 7001     BDA $+2
      2771
12771 2401     LDB $-11
      2756
12773 0410     STB COMP,1
      3602
12775 7001     BDA $+4
      3001
12777 6440     LDA =255
      0377
13001 4401     STA DC
      2610
               *UPDATE INDEX AND TIME
13003 6401     LDA INDEX
      2300
13005 7001     BDA $+4
      3011
13007 6400     LDA DLEN
      0113
13011 4401     STA INDEX
      2300
13013 6401     LDA TIME
      2176
13015 7001     BDA $+4
      3021
13017 6440     LDA =511
      0777
13021 4401     STA TIME
      2176
               **********END OF MAIN PART*********
```

```
13023                PAGE 20
                     *DISPLAY ROUTINES
                     *DISPLAY: INTERRUPTS ENABLED
13023 0144           BP1 BPT 1
13024 7101           BFT $+19
      3047
13026 0145           BPT 2
13027 7101           BFT $+9
      3040
13031 6440           LDA =510
      0776
* 13033 7700         BRM SCOPE
      0000
13035 2203           DATA BIPHI
13036 7301           BRU BP1
      3023
13040 6440           LDA =510
      0776
* 13042 7701         BRM SCOPE
      3033
13044 1204           DATA Y
13045 7301           BRU BP1
      3023
13047 0145           BPT 2
* 13050 7100         BFT DIS [BRANCH TO PROGRESS DISPLAY]
      0000
13052 6440           LDA =511
      0777
* 13054 7701         BRM SCOPE
      3042
13056 7602           DATA SST
13057 7301           BRU BP1
      3023
13061                DEF IPDONE
                     *
```

```
13061          PAGE 21
               *PROGRESS DISPLAY: INTERRUPTS DISABLED
13061 0050     DIS DIR
13062 0147      BPT 4 [TEST IF SST OR KST TO BE DISPLAYED]
* 13063 3100    BFF DIS1
      0000
13065 0144      BPT 1
13066 2440      LDB =0
      0000
13070 2340      ACB =0
      0000
13072 0540      COB =1
      0001
13074 7101      BFT DIS
      3061
13076 6440      LDA =2047 [INTERCHANGE CONTENTS OF SST AND KST]
      3777
13100 2410      LDB SST,1
      7602
13102 3411      XMB KST,1
      7777
13104 0410      STB SST,1
      7602
13106 7001      BDA $-6
      3100
13110 1610      MPO DIS+6
      3067
13112 6440     DIS1 LDA =511
      0777
13114 0144      BPT 1
13115 3101      BFF$+7
      3124
* 13117 7701    BRM SCOPE
      3054
13121 4602      DATA SST-1536
13122 7301      BRU DIS
      3061
13124 0145      BPT 2
13125 3101      BFF $+7
      3134
* 13127 7701    BRM SCOPE
      3117
13131 5602      DATA SST-1024
13132 7301      BRU DIS
      3061
```

```
  13134          PAGE 22
  13134 0146     BPT 3
  13135 3101     BFF $+7
        3144
* 13137 7701     BRM SCOPE
        3127
  13141 6602     DATA SST-512
  13142 7301     BRU DIS
        3061
* 13144 7701     BRM SCOPE
        3137
  13146 7602     DATA SST
  13147 7301     BRU DIS
        3061
  13151          END START

  SUBROUTINES USED

  SCOPE
  INPUT
  PRBS
```

```
00000          *
               *INPUT - SAMPLES ANALOG INPUT AND PUTS MEAN VALUE IN B
               *
00000           REF IPDONE
00000          INPUT RES 2
00002 4440      STA =0
      0000
00004 0003      EOM 034024
      4024
00006 1440      PIN =0
      0000
00010 6400      LDA $-1
      0007
00012 7640      MPA =0
      0000
00014 5640      MPF =0
      0000
00016 4540      COA =04000
      4000
00020 7100      BFT $+6
      0026
00022 6440      LDA =07777
      7777
00024 7600      MPA $-7
      0015
00026 1640      MPQ =-8
      7770
00030 7100      BFT $+6
      0036
00032 6400      LDA $-23
      0003
00034 3220      BRC *INPUT
      0000
00036 3200      BRC $+2
      0040
               *RESET SAMPLE COUNTER
00040 6440      LDA =-8
      7770
00042 4400      STA $-11
      0027
               *AVERAGE THE SAMPLES
00044 6400      LDA $-25
      0013
00046 2400      LDB $-25
      0015
00050 0266      CYD 9
```

```
00050          PAGE 2
               *RESET RUNNING TOTAL
00051 6440     LDA =0
      0000
00053 4400     STA $-32
      0013
00055 4400     STA $-32
      0015
* 00057 7300   BRU IPDONE
      0000
00061          DEF INPUT
00061          END

IPDONE
```

```
                    *SCOPE (AJP)
                    *
00000                RORG 0
00000               ADDR RES 2
00002               SCOPE RES 2
00004 2420           LDB *SCOPE
      0002
00006 0400           STB ADDR+1
      0001
00010 2440           LDB =04000
      4000
00012 0400           STB $+13
      0027
00014 2440           LDB =010
      0010
00016 0400           STB ADDR
      0000
00020 0400           STB $+33
      0061
00022 3600          AC MPB $+5
      0027
00024 0003           EQM 036004
      6004
00026 1040           POT =0
      0000
00030 0003           EQM 036005
      6005
00032 1020           POT *ADDR
      0000
00034 0003           EQM 037000
      7000
                    *X-AXIS
00036 0003           EQM 036005
      6005
00040 1040           POT =0
      0000
00042 0003           EQM 037000
      7000
00044 7000           BDA AC
      0022
```

```
00046           PAGE 2
                *Y-AXIS
00046 0003      EOM 036004
      6004
00050 1040      POT =04000
      4000
00052 6440      LDA =255
      0377
00054 3600    AB MPB $+5
      0061
00056 0003      EOM 036005
      6005
00060 1040      POT =0
      0000
00062 0003      EOM 037000
      7000
00064 7000      BDA AB
      0054
00066 1600      MPO SCOPE+1
      0003
00070 5600      MPF SCOPE
      000
00072 7320      BRU *SCOPE
      0002
00074           DEF SCOPE
00074           END
```

REFERENCES

REFERENCES

1. "Automation of a Portland Cement Plant Using A Digital Control Computer" by Phillips, R.A. (Proc. I.F.A.C. Conference, Basle, 1963, p.347)
2. "Development and Application of Self-Optimizing Control to Coal-Fired Steam-Generating Plant" by Moran, F., Berger, CS. and Xirokostas, D. (Proc. I.E.E., 115, February 1968, p.307)
3. "An Optimalizing Control of Boiler Efficiency" by Fujii S. and Kanda N. (Proc. I.F.A.C. Conference, Basle, 1963, p.380)
4. "Analytical Design of Feedback Systems" by Newton G.C., Gould L.A., and Kaiser J.F. (Wiley, 1957)
5. Paper to be published by Monk J., Institute of Engineering Control, University of Warwick, Coventry.
6. "General Stability Analysis of Sinusoidal Perturbation Extremum Searching Adaptive Systems" by Eveleigh V.E. (proc. I.F.A.C. Conference, Basle, 1963, p.472)
7. "Design of a Single-Input Sinusoidal-Perturbation Extremum Control System" by Jacobs O.L.R. and Shering G.C. (Proc. I.E.E., 115, January 1968, p.212)
8. Discussion on Reference 5 by Douce J.L. (ibid Reference 5, p.478)
9. "Automatic Optimization by Continuous Perturbation of Parameters" by Hammond P.H., and Duckenfield M.J. (Automatica, 1, 1963, p.147)
10. "The Behaviour of Adaptive Controllers" by Douce J.L. (Proc. I.F.A.C. Conference, Basle 1963, paper 346)
11. "Hill Climbing" by Jacobs O.L.R. (Control, February 1962, p.92)
12. "Determination of System Dynamics by the Use of Adjustable Models" by Blandhol E. and Balchen J.G. (Proc. I.F.A.C. Conference, Basle 1963, Theory p.602).
13. "Parameter-Tracking Models for Adaptive Control Systems" by Bohn E.V., Butler, R.E. and Mukerjee M.R. (Proc. I.E.E., 113, February 1966, p.378)

14. "Introduction to Statistical Identification Methods in Control Systems" by Briggs P.A.N., Clarke D.W. and Hammond P.H. (Control, March 1968)

15. "Analysis of a Type of Model-Reference Adaptive Control System" by Dymock A.J., Meredith J.F., Hall A, and White K.M. (Proc. I.E.E., 112, April 1965, p.743)

16. "Digital Adaptive Control System Employing a Normalized Performance Index" by Nightingale J.M., (Proc. I.E.E., 111, January 1964, p.165)

17. "Extremum Control in the Presence of Noise" by Jacobs O.L.R., and Wonham W.M. (J. Electronics Control, 11, 1961, p.193)

18. "Problems in the Statistical Theory of Systems of Automatic Optimization" by Feldbaum A.A. (Proc. I.F.A.C. Conference, Moscow, 1960, Vol.II, p.547)

19. "Extremum or Hill-Climbing Regulation: a Statistical Theory Involving Lags, Disturbances and Noise" by Roberts J.D. (Proc. I.E.E., 112, January 1965, p.137)

20. "A Self-Optimizing Non-Linear Control System" by Douce J.L., and King R.E. (Proc. I.E.E., 108B, July 1961, p.441).

21. "High-Frequency Perturbation in Hill-Climbing Systems" by Ng K.C. (Proc. I.E.E., 111, November 1964, p.1907)

22. "Six-Channel Adaptive Computer" by Douce J.L., and Ng K.C. (Proc. I.E.E. 111, October 1964, p.1757)

23. "The Application of Random Test Signals in Process Identification" by Van der Grinten, P.M.E.M. (Proc. I.F.A.C. Conference, Basle, 1963)

24. "The Use of Pseudo-Random Signals in Adaptive Control" by Douce J.L., and Ng. K.C. (Proc. I.F.A.C. Symposium, Teddington, 1965)

25. "Correlation Analysis of Process Dynamics Using Pseudo-Random Binary Test Perturbations" by Briggs P.A.N., Hammond P.H., Hughes M.T.G. and Plumb G.O. (I.Mech.E. Advances in Automatic Control Convention, Nottingham, April 1965)

26. "Application of Cross-Correlation Equipment to Linear System Identification" by Hazelrigg A.D.G. and Noton A.R.M. (Proc. I.E.E. 112, December 1965, p.2385).

27. "Equipment for Investigating P.R.B.S. Applications" by Darnell M, and Nicoll G.R. (I.E.E. Control and Automation Colloquium on P.R.S. Applied to Control Systems, 27th January 1967)

28. "Application of P.R.S. to process plants - a case history" by Davies W.D.T. and Sinclair P.A. (ibid.)

29. "Error-Correcting Codes" by Peterson W.W. (M.I.T. Press 1961)

30 "Linear Recurring Sequences" by Zierler N. (J. Soc. Ind. Appl. Maths, 7, 1959, p.31)

31. "P.R.S. for the Dynamic Analysis of Multivariable Systems" by Briggs P.A.N. and Godfrey K.R. (Proc. I.E.E., 113, July 1966, p.1259)

32. "Simultaneous Estimation of First and Second Derivatives of a Cost Function" by Clarke D.W., and Godfrey K.R. (Electronics Letters, September 1966).

33. "Three-level m-sequences" by Godfrey K.R. (Electronics Letters, July 1966).

34. "Statistical Properties of Smoothed Maximal-Length Linear Binary Sequences" by Roberts P.D. and Davis R.H. (Proc. I.E.E., 113, January 1966, p.190.)

35. "Several Binary Sequence Generators" by Zierler N. (M.I.T. Lincoln Laboratory Technical Report No. 95, 12th September 1955).

36. "The Theory of Autonomous Linear Sequential Networks" by Elspas B. (Trans. I.R.E., CT6, March 1959, p.45)

37. "Input-Transducer Errors in Binary Cross-Correlation Equipment" by Godfrey K.R., and Murgatroyd W. (Proc. I.E.E., 112, March 1965, p.565).

38. "Input-Transducer Errors in Binary Cross-Correlation Equipment-2" by Godfrey K.R., Everett D. and Bryent P.R. (Proc. I.E.E., 113, January 1966, p.185)

39. "Input-Transducer Errors in Binary Cross-Correlation Equipment - 3" by Godfrey K.R. (Proc. I.E.E., 113, June 1966).

40. "Generation of Delayed Replicas of Maximal-Length Linear Binary Sequences by Tsao, S.H. (Proc. I.E.E., 111, November 1964, p.1803)

41. Discussion on Reference 39 by Roberts P.D. (ibid., 112, April 1965, p.702).

42. Discussion on Reference 39 by Davies A.C. (ibid., p.703).

43. Discussion on Reference 39 by Davies W.A. (ibid., 113, February 1966, p.295).

44. "Delayed Versions of Maximal-Length Linear Binary Sequences" by Davies A.C. (Electronics Letters, May 1965, p.61)

45. "Further Notes on Delayed Versions of Maximal-Length Linear Binary Sequences" by Davies A.C. (ibid., September 1965, p.190).

46. "An Introduction to the Mathematics of Servomechanisms" by Douce, J.L. (EUP, 1963)

47. "Some Studies in Optimum and Self-Adaptive Control Systems" by Murthy K.K. (Phd. Thesis, University of Warwick, Coventry, 1967)

48. "Recent Advances in a Hill-Climbing Technique" by Ng, K.C., Murthy K.K., Stockwell D.H. and Morris K.R. (Advances in Automatic Control, 2nd UKAC Control Convention, April 1967).

49. "An Electromechanical Running Averager" by Ng K.C., and Murthy K.K. (Electronic Engineering, March 1967, p.169)

50. "Mathematical Analysis of Random Noise" by Rice S.O.("Selected Papers on Noise and Stochastic Processes", Dover, 1954).

51. "Principles and Applications of Random Noise Theory" by Bendat J.S. (Wiley 1958)

52. "Random Processes in Automatic Control" by Laning, J.H. and Battin R.H. (McGraw Hill, 1956).

53. "Self-Optimizing Systems Involving Estimation of Cost Function Slope Using Pseudo-Random Binary Perturbations" by Clarke D.W. (N.P.L. Autonomics Division Report No.20, 1966).

54. "The Development and Performance of a Self-Optimizing System" by Douce J.L. and Bond A.D. (Proc. I.E.E., 110, March 1963, p.619)

55. "Dynamics of the Parameter-Perturbation Process" by Douce J.L., Ng K.C. and Gupta M.M. (Proc. I.E.E., 113, June 1966, p.1077).

56. "Dynamics of the Parameter Perturbation Process" by Gupta M.M. (Ph.D. Thesis, University of Warwick, Coventry, 1967).

57. "A Comparison of Test Signals in Parallel-Path Profit Measurement" by Thomas D.K. (M.Sc. Project Report, University of Warwick, Coventry, 1967).

~~58. "Automatic Optimization by Continuous Perturbation of Parameters" by Hammond P.H. and Duckenfield M.J. (Automatica, 1, 1963, p.147).~~

59. "The Response of Multi-Channel Adaptive Systems" by Douce J.L. and Ng K.C. (I.E.E.E. International Convention, New York, March 1964, part I, p.55).

60. "Cross-Correlation of Pairs of Binary m-Sequences of the Same Length" by Ream, N. (Battersea College of Advanced Technology, Dept. of Elect. and Control Engineering Report, January 1966).

61. "Application of Modified P.R.S. to multi-input processes" by Briggs P.A.N., and Williams B.J. (I.E.E. Control and Automation Colloquium on P.R.S. Applied to Control Systems, 27th January 1967).

62. "An Examination of Some P.R.S. for Multivariable System Dynamic Analysis" by Godfrey K.R. and Briggs P.A.N. (N.P.L. Autonomics Division Report No.14, April 1966).

63. "Multivariable System Identification Using P.R.B.S." by Williams, B.J. (N.P.L. Autonomics Division Report No.23, November 1966)

64. "Multi-Input/Multi-Output System Identification Using P.R.B.S." by Williams B.J. (Electronics Letters, October 1966, p.382).

65. "Comparison of Three Methods of Multi-Variable System Identification Using P.R.B.S." by Williams B.J. (Electronics Letters, October 1965, p.235)

66. "Digital Communications with Space Applications" by Golomb S.W. (Prentice Hall, 1964).

67. "On the Theory of Self-Tuning Systems with a Search of Gradient by the Method of Auxiliary Operator" by Kazekov I.E. and Evlanov L.G. (Proc. I.F.A.C. Conference, Basle 1963).

68. "An Automatic Method for Finding the Greatest or least Value of a Function" by Rosenbrock H.H. (Computer Journal, 1960, Vol.3, p.175).

69. "Method of Conjugate Gradients Applied to Self-Adaptive Digital Control Systems" by Davis R.H. and Roberts P.D. (Proc. I.E.E., Vol.115, No.4, p.562, April 1968).

70. "Function Minimization by Conjugate Gradients" by Fletcher R and Reeves, C.M. (Computer Journal, 1964, Vol. 7, p.149.)

71. "The Method of Steepest Descent for Non-Linear Minimization Problems" by Curry H.B. (Quarterly Journal of Applied Mathematics, 1944, Vol.2, p.258)

72. "A Rapidly Convergent Descent Method for Minimization" by Fletcher R and Powell M.J.D. (Computer Journal, 1963, Vol.6, p.163.)

73. "An Iterative Method for Finding Stationary Values of a Function of Several Variables" by Powell M.J.D. (Computer Journal, 1962, Vol.5, p.147)

74. "Iterative Methods for Linear Equations with Symmetric Positive Definite Matrix" by Martin D.W. and Tee G.J. (Computer Journal, 1961, Vol.4, p.242.)