STATE VARIABLE AND PARAMETER ESTIMATION.

—

A thesis submitted for the degree of

Doctor of Philosophy

by

Robert F. Kilbride-Newman

Department of Chemical Engineering

University of Aston in Birmingham

October 1974

Summary.

The development of methods designed to produce continuous solutions to linear, ordinary differential equations is described. These solutions are based on a set of orthogonal polynomials. This work is then incorporated into state estimation theory and a continuous filter is developed.

A new sequential adaptive filter is then developed which effectively compensates for errors in the mathematical description of the process. This adaptive filter finds the mean and covariance of 'fictitious inputs' and uses these parameters to compensate for the model errors.

The results show the application of the above topics to some simple linear and non-linear systems and demonstrate the effectiveness of the adaptive filter in situations involving poor models. The adaptive filter also provides information concerning the nature of the model errors which may be used to improve the model formulation.

Previous Publications.

'The Use of Orthogonal Polynomials in Simulation and State Estimation.'

R.F. Kilbride-Newman, B. Gay.

Presented at:

The European Symposium 'Computer Application in Process Development'.

DECHEMA, Erlangen, Germany (April 1974).

A copy of this paper will be found at the end of the thesis.

ACKNOWLEDGEMENTS.

I should like to express my thanks to:

The Staff and Postgraduates of the Department of Chemical Engineering, University of Aston in Birmingham, in particular Dr. B. Gay, for his helpful supervision.

The University of Aston in Birmingham, for financial support.

M.G. Kilbride-Newman, for the typing of this thesis.

TO AOIFE.

Nomenclature.

Generally, higher case letters denote Matrices or Operators (e.g. E denotes the expectation operator; V denotes a covariance matrix).

Lower case letters denote vectors or scalars.

The dot notation (e.g. $\dot{x}$) denotes differentiation with respect to time.

The dash notation (e.g. $f_x^1(x,y)$) denotes partial differentiation with respect to the subscripted variable.

CONTENTS.

CHAPTER (1).

INTRODUCTION.

1. Introduction.

Physical systems are designed and built to perform certain defined functions. Chemical process plants, electric power systems, etc. must meet their respective load demands. Submarines, aircraft and spacecraft must navigate their respective environments. In order to determine whether a system is performing properly, and ultimately to control the system performance, the engineer must know what the system is 'doing' at any instant of time. That is, the engineer must know the state of the system. In navigation, the state consists of position and velocity of the craft in question; in chemical processes the state consists of the various flows, pressures, temperatures, concentrations, etc. that exist throughout the particular system. Physical systems are often subject to random disturbances, so that the system state may itself be random. In order to determine the state of a system, the engineer builds a measurement device (e.g. flowmeter, thermometer, pressure gauge, etc.) and takes measurements or observations on the system. These measurements are generally contaminated with noise caused by the electronic and mechanical components of the measuring device.

The problem of determining the state of a system from noisy measurements is called estimation, or filtering, and is the main subject of this thesis. It is of central importance in engineering, since state estimates are required in monitoring, and for the control of systems. Furthermore, a large class of system identification problems can be regarded as problems of filtering.

This thesis discusses the development of filter theory from its origins in the work of K. Gauss, to the important

contribution made by R.E. Kalman. A discussion of the present state of the art reveals problem areas and introduces the concept of 'adaptive filtering'. The attempts to construct adaptive filters which are discussed, however, are not totally satisfactory and outstanding problems remain. A new adaptive filter is described, which overcomes many of these problems.

The approach taken in this thesis is that the mathematical model of a system and the actual dynamics of that system rarely match. This fact is accepted at the outset and the new adaptive filter, while retaining the sequential structure of the basic Kalman filter, compensates effectively for these model errors. Furthermore, this new filter provides useful information concerning the type of model error that exists. This information is particularly helpful to any work aimed at improving the model.

It must be remembered that the ultimate aim of filtering is to provide estimates of the system's state, and statistical parameters describing the state estimation errors, that can be used to control the system. Therefore, a modification to the Kalman filter is described which produces continuous state estimates from discrete measurements. These continuous estimates can be used in a control scheme to produce continuous control functions, and thus achieve finer control than would be possible with the standard discrete form of the Kalman filter.

CHAPTER (2).

LITERATURE SURVEY.

(2.1) Identification Techniques.

(2.1.1) Estimation of Step Response and Impulse Response.

(2.1.2) Correlation Analysis.

(2.1.3) Frequency Response Estimation.

(2.2) Statistical Parameter Estimation Theory.

(2.2.1) Historical Developments.

(2.2.2) More Recent Theoretical Developments.

(2.3) Kalman Filtering.

(2.3.1) Linear Filtering.

(2.3.2) Non-linear Problems.

2.1 Identification Techniques.

Over the past fifteen years, increasing attention has been given to the problem of determining, from records of experimental input/output data, useful mathematical descriptions of dynamical systems. The study of such problems has received great impetus more recently as a result of investigations on large scale process plant and adaptive systems, and the ready availability of digital and analogue computers has facilitated the implementation of sophisticated data processing operations. It is largely as a result of these developments that System Identification, or (in alternative and probably preferable terminology), Parameter Estimation, in a control engineering context is currently recognised as a substantial topic for research and development.

It is advisable at this stage to examine what is implied by the term 'Identification'. A little thought will indicate that the term is something of a misnomer, although its use is fairly commonly accepted, and perhaps 'Parameter Estimation' would be a more precisely descriptive term to use. This is because even in the simplest system no useful progress can be made unless we are willing to make some a-priori assumptions about the structure of the unknown system. It is the nature of these basic structural assumptions which distinguishes the various identification techniques in use at the present time. They amount to a choice of a basic type of mathematical model of the system, with a (finite) number of unknown parameters which are to be determined by experiment. Some of the best known examples are:

(a) Weighting sequence models. (39)

(b) Frequency response models (39,9,22,2)

(c) Differential equation models. (2,11,30,27)

(d) Difference equation models. (39,1,28)

(e) Orthogonal function expansions (40,29,49)

Once the choice of structural model has been made (e.g. as above) the problem reduces to a more readily definable one of parameter estimation.

In order to asses the merits of a proposed scheme for identification, it is necessary to consider a number of related aspects. The main points for consideration are:

(1) The usefulness of the selected model structure.

(2) The size and cost of computer hardware required to estimate the model parameters.

(3) The amount of time and data which is required to achieve the desired confidence in the model parameters.

(4) The types and nature of test signals which are to be used, and the extent to which they are likely to disturb normal operation of the system.

All of these factors interact, and (1) and (2) can only be evaluated subject to engineering judgements. It is possible however, to obtain some quantitative criteria in connection with (3) and (4) for some of the schemes listed above, on the basis of statistical analysis.

2.1.1 Estimation of Step Response and Impulse Response

(64)

Here the basic weighting sequence model (a) listed previously is used. That is for a single input/single output system represented by a linear stationary model the output z(t) can be expressed in terms of the input x(t) via the well known convolution integral

$$z(t) = \int_0^t h(\tau)\ x(t-\tau)\ d\tau$$

Where $h(\tau)$ is the impulse response function for the system. In the discrete case this equation is replaced by the following:

$$z(t_i) = \sum_{j=1}^{\infty} a_j\ x(t_{i-j})$$

where the a_j is a weighting sequence which is intimately related to the impulse response function $h(\tau)$ above.

An attempt is made to estimate values of the impulse response function, or the step response function which will best fit the observed conditions.

Given complete freedom of the choice of input signals x(t), the problem would be very simple in principle: One would simply apply narrow pulses or step disturbances at the input to the system, and observe the resulting response z(t), at appropriate time intervals. It must be remarked however, that the use of narrow pulse inputs and even step disturbances as test signals in many systems is just not feasible. This is because if the input is to contain sufficient energy to excite the system its amplitude must be very large. Frequently this makes it quite impossible to obtain reliable results with such test signals

since the likelihood is high that certain parts of the system (e.g. actuating valves, or amplifiers) will be driven into saturation. The observed responses thus would not be truly representative of the normal modes of operation.

2.1.2 Correlation Analysis (39, 23, 18).

This technique has been developed to overcome the problems outlined in the above section. By considering correlation functions of the inputs and outputs of a system and using as a test signal an approximation to white noise (e.g. pseudo-random binary sequences) good estimates of the impulse and step response functions can be obtained.

2.1.3 Frequency Response Estimation (39, 9).

The estimation technique here is based on the use of a sinusoidal test signal. Although techniques are available which will work with random input/output data (22, 25) Sine wave testing is a very powerful technique, and is very well established both in theory and in practice. However, it is not always convenient to use such a technique for an accurate determination of the frequency response of the system under examination. Very often, it is simply not feasible to inject sinusoidal disturbances at the input which is of most interest in a system. For such situations several alternative procedures have been investigated with a fair degree of success (64). These modifications are based on using the random disturbances which occur during normal operation of the system.

The identification techniques which have been considered so far could be termed 'linear black-box techniques' (7,64,65), in that the only a-priori assumptions made about the systems to be tested were that they were linear and time invariant. Within this broad framework it is usually necessary to estimate the values of a large number of parameters in order to arrive at an adequate model of the system's characteristics. This can often lead to an inconvenient commitment of time to test a particular system in practice. Also the development of these techniques has been based on single input/output systems and although they can be extended to multi-input systems (usually with a considerable increase in complexity) the extension to the multi-output case is completely prohibitive in practice. (7,64).

From this point of view, there is considerable incentive to find ways of using more a-priori information and to attack the multi dimensional case directly. The obvious way to do this is to study the underlying mechanics, physics and chemistry of the system in the first instance; and to obtain a set of differential or difference equations to represent the essential features of the system's structure. After simplifying these (e.g. by reducing their order as far as possible, and usually, by linearising them about some nominal state), experience has shown that one is left with the problem of estimating, from the measurable data, the 'best' values for the unknown parameters and coefficients which appear in the assumed equations. This is essentially a problem of statistical estimation (13).

2.2 Statistical Parameter Estimation Theory.

2.2.1 Historical Developments.

Almost synonymous with estimation and smoothing is the 'least-squares' principle. Through a period of over 160 years since the pioneering work of Legendre (36) and the fundamental publication of Gauss (24), the stage has been reached in which, whenever confronted with a data set suspected of containing random errors, the 'most probable' or 'best' estimate of the desired parameters is computed by means of some variation of the method of least-squares estimation.

Perhaps the first major advance in estimation theory since the introduction of the least-squares principle was the 'method of moments' formulated by Pearson (47, 48). Although the method of moments is no longer widely used, one often encounters situations in which the method is applicable when other estimators exibiting greater theoretical attributes can be employed only with a large amount of labour. The main disadvantage with the method of moments is that it has been established that the estimates found with this technique are not the best possible from the view point of efficiency (17).

The present firm foundations of estimation theory are attributed to Fisher. His contributions are contained in a series of fundamental papers (17, 19, 20). Fisher demonstrated that the method of maximum likelihood was usually superior to the method of moments and that estimates derived by the likelihood technique could not be improved essentially. To students of scientific history it comes as no great surprise that Gauss had, at least in particular cases, anticipated the important general method of estimation introduced by Fisher as the method of maximum

likelihood.

Fisher also introduced a set of definitions that have been adopted for describing estimators. He introduced such terms as unbiased, sufficiency, efficiency and asymptotic efficiency into the theory of estimation. These concepts motivated a whole series of statistical theories and publications dealing with the detailed theoretical aspect of estimators. In particular, general -isation of the theory led to the relatively recent development of non parametric estimation in which the exact nature of the probability distribution for the random variables need not be specified. By carefully examining the foundations of estimation and introducing new concepts, Fisher freed estimation theory from the tight confines that had existed since the work of Gauss.

Almost concurrently with Fisher's statistical investigation was the rapid development of communication theory by engineers and physicists. Communication theory, as originally conceived, was applied to the transmission of intelligence by electrical means. By the very nature of the transmission media, communications were found to be perturbed by a random process, or noise, introduced by thermal motion in resistors, electron motion in vacuum-tubes, galactic and ionospheric noise in propagation, etc. Communication engineers were vitally concerned with the affect of these noise sources on the intelligibility of signals within communication channels. The first attempts to reduce the effects of unwanted noise introduced filters designed to estimate the power-frequency spectrum of the desired signal. These attempts were in the proper direction but were severely limited because of the lack of an estimation theory that could be used to synthesise the required noise-separation filters.

A fresh approach to the study of information transmission in the presence of perturbing noise is generally attributed to N.Wiener (1942). However an independent and similar theory had already been published by A.Kolmogorov in 1941 (35,56). Wiener made two important contributions. First he demonstrated that estimation theory could be applied to synthesise an electrical filter that would provide a 'best' separation of a desired signal in the presence of undesired noise. Secondly, Wiener emphasised the viewpoint of treating signals and noise as stochastic processes rather than viewing them in terms of their frequency spectra.

Wiener's work coincided with the birth of a new branch of science: Information theory, or statistical communication theory. This coincidence resulted in the almost immediate adoption of Wiener's technique. Moreover, as usual with each significant advance, a number of modifications of the original technique were formulated.

At first glance, Wiener's theory appears to be essentially a least squares estimation process. This is indeed the case. However Wiener made elegant use of the fact that he was estimating parameters from input data in the form of a stochastic process. Thus by leaning strongly upon the known properties of stochastic processes, he formulated an optimum estimator which makes the best separation between the desired signal and the undesired noise. An important point is that Wiener demanded that the optimum estimator be a linear, physically realisable filter. The concept not only proves the existence of such filters, but also, in theory, permits one to synthesise the optimal filters. Of course because the theory provides more specific properties of the optimum estimator than are attainable from a conventional least

squares calculation, one must be prepared to pay a penalty for the increased output. The cost is in the requirement for more a-priori knowledge of the probability distributions of both the desired signal and undesired noise. These additional requirements are the reasons why filter theory has become classed as a facet of probability theory.

2.2.2. More Recent Theoretical Developments.

The copious present-day theories for smoothing and prediction of time series, with few exceptions, owe their genesis to the original papers of Wiener (56) and Kolmogorov (35). These papers appeared during World War II, a period during which many serious and urgent smoothing and prediction problems were being attacked by military systems designers. Although the mathematical treatments in these papers were serious stumbling blocks, engineers were willing to overcome these hurdles because they realised that for the first time an analytical synthesis technique was available for the systematic design of filters. Unfortunately, the abstract formalisms coupled with almost unsurmountable difficulties of solving a certain basic integral equation (Wiener-Hopf equation) deterred many readers from a full understanding of the rather simple underlying principles of the Wiener-Kolmogorov theory.

The number of papers which have generalised, modified and extended the original Wiener-Kolmogorov theory is far too large to allow an adequate reference list to be compiled. However, extensive bibliographies on this subject are given in references (6, 52,58). Some extensive developments of the Wiener-Kolmogorov theory are contained in the publications of Darlington (14)

Bendat (4), Wainstein and Zubakov (55), Yaglom (57), and Lanning and Battin (37).

The Wiener-Kolmogorov theory has not been universally accepted, and various criticisms have been directed at the theory. The criticisms have been partly directed at the underlying assumptions and partly toward the often unsurmountable practical problems of actually synthesising the optimal filters. These criticisms have been the motivation for several modifications as well as generalisations of the original concept.

At first, before the theory had been well assimilated almost everyone was content with the hypothesis that both the signal and noise were stationary processes; However, as engineering technology became more refined and systems became more complex, the stationarity hypothesis was questioned. For this reason the Wiener-Kolmogorov theory has been extended to encompass non stationary signals (15, 45, 46, 53). The cases of sampled data systems with and without the stationarity hypothesis, have also received their due share of attention (21). Theories have also been formulated for situations in which the linear filters are permitted to have time varying elements (4).

The most severe problems encountered in the application of the Wiener-Kolmogorov theory to specific applications arise from the problems in synthesising the theoretically optimum filter responses. Except for a relatively few, special and simple examples, the synthesis of the filter is an extremely labourious task and frequently one must resort to approximations and certain compromises (54).

The Wiener-Kolmogorov methods have been criticised

because the transfer responses are dictated completely by a minimal-error criterion and the computation thus ignores the filter's transient response (44). Wiener filters frequently result in underdamped systems which exhibit oscillatory transient responses that can be inconvenient.

The Wiener-Kolmogorov theory has served as both an end in itself and as the motivation for related theories which are designed to avoid the problems encountered in solving the Wiener-Hopf equation, as well as the practical problem of synthesising the theoretically optimum filter from its impulse response. An alternative approach to signal filtering and prediction has been suggested, which essentially avoids the Wiener-Hopf integral equation by substituting an equivalent differential equation. Of more practical interest, is the fact that the differential equation technique has the property that the optimum filter can be synthesised in a sequential fashion and, thus, is often readily implemented. Although these differential equation techniques were independently developed, they are in most instances equivalent or intimately related to recursive least squares estimators (12, 13).

The primary impetus for the current activity in the theory of sequential estimation stems from the work of Kalman and Bucy (31, 32, 33, 34). Both Kalman and Bucy independently recognised that, rather than attack the Wiener-Hopf equation directly with the attendant problems of factorisation, it is frequently desirable to convert the integral equation into a non linear differential equation whose solution yields the covariance matrix of the minimum filtering error. In turn, this matrix contains all the necessary information for the design of the optimum filter.

2.3 Kalman Filtering.

2.3.1 Linear Filtering.

The paper by Kalman (31) in 1960 introduced a different approach to the problem of Wiener and Kolmogorov for random sequences. In 1961 Kalman and Bucy (34) generalised the results to random processes. These results are closely related to those obtained for sequential least squares estimation (12, 13), but it would be wrong to allow this similarity to be construed as a criticism of their work. Their development is much more rigorous and general, and they have generated a new and powerful technique for both estimation and control problems.

The practical implications of linear filtering theory were soon recognised by the engineering community. Whereas classical least squares methods involve simultaneous processing of batches of observation with attendant data storage requirements, the filter operates on the data sequentially, requiring no data storage. The filter generates new estimates as new observations become available, thus opening the possibility of real-time estimation. As a by-product, the filter generates the estimation error covariance matrix, which measures the uncertainty in the estimate.

Other important advantages accrue from the filter theory. It is possible, for example, to perform a complete error analysis without actually simulating the filter. The filter also offers advantages over least squares in applications to non linear problems. Thus, although the linear filter is completely equivalent to least squares when the latter is properly interpreted, it offers numerous advantages in applications.

The structure of the discrete linear time invariant Kalman filter can be briefly summarised in the following relationships:

Given a Mathmatical Model:

$$x(i+1) = \emptyset\, x(i) + u$$

where $x(i)$ is the n dimensional state vector at the i^{th} time increment, and $\emptyset$ is an n x n transition matrix; and given an estimate $\hat{x}(i)$ for $x(i)$ with a known error covariance matrix,

$$\hat{V}(i) = E(x(i) - \hat{x}(i))\,(x(i) - \hat{x}(i))^T$$

and knowing that,
$Eu = 0$ and $Euu^T = Q$ (a known covariance matrix), then a prediction $\bar{x}(i+1)$ of $x(i+1)$ can be calculated via $\bar{x}(i+1) = \emptyset\, \hat{x}(i)$ and the prediction error covariance matrix

$$\bar{V}(i+1) = E(x(i+1) - \bar{x}(i+1))\,(x(i+1) - \bar{x}(i+1))^T$$

can be calculated via

$$\bar{V}(i+1) = \emptyset\, \hat{V}(i)\, \emptyset^T + Q$$

When the measurement $z(i+1)$, which is related to the state via

$$z(i+1) = H\, x(i+1) + v$$

Where z is an h-vector of measurements

H is an h x n matrix and v is an h-vector of random variables

with $Ev = 0$

and $Evv^T = R$

becomes available an estimate $\hat{x}(i+1)$ of $x(i+1)$ can be calculated.

$$\hat{x}(i+1) = \bar{x}(i+1) + K(i+1)\,(z(i+1) - H\,\bar{x}(i+1))$$

where $K(i+1) = \bar{V}(i+1)\,H^T\,(H\bar{V}(i+1)H^T + R)^{-1}$

is a gain matrix calculated to minimise the estimation error covariance matrix which can now be calculated as

$$\hat{V}(i+1) = (I - K(i+1)H)\,\bar{V}(i+1)$$

This whole process is repeated for the subsequent time increments.

The practicality of the Kalman approach to the estimation problem was first exploited in aerospace applications (38,43). Following these early pioneering investigations, a number of authors reported interesting applications of linear filter theory (3, 59, 62, 63, 66, 60, 61).

The experience gained in applying the Kalman filter to physical systems has been invaluable in highlighting various problem areas. Considerable engineering experience is needed to properly identify the system to which the filter is to be applied, to model adequately that system, and then to develop a practical programme that mechanises the filter. The optimisation of the

filter must include many factors which are difficult or impossible to describe mathematically, such as the trade off between performance and computer size. The statistical parameters are rarely based on the actual statistics of the physical systems, because these statistics are either too complicated or are not well known. The problems encountered in Kalman filtering in general fall into the following categories:

(a) Loss of positive definiteness in the covariance matrix resulting from numerical errors.

(b) Improper mathematical model, leading to a divergence of the estimates from the states.

(c) Non-linear phenomena generally aggravated by a poor selection of the starting estimates.

Item (a) can be removed reasonably efficiently by basing the filter on square root algorithms (50), by simply rearranging the basic Kalman equations to ensure positive definiteness (43), or by accounting for numerical error as input noise to the system (43, 50). Item (c) will be considered in the section on Non-linear estimation.

The problem of filter divergence (51, 43, 50) is essentially caused by the filter being constructed on the basis of an erroneous model. The filter therefore learns the wrong state too well when it operates over many observations. The problem is particularly acute when the noise inputs to the system are small and when measurement noise is small. Eventually, the error covariance matrix becomes very small, the filter gain is therefore small, and subsequent observations have little effect on the estimate. But the dynamical system model in the filter is different from the actual system model, so that the estimate and

the state can diverge.

A number of techniques have been proposed for overcoming the divergence problem.

(1) Directly increasing the Prediction-error Covariance Matrix.

Arguments for using this approach are as follows: State equations which are invalid have been used to update the estimates. One should therefore increase the prediction-error covariance matrix in accordance with the errors involved in the time updating of the estimates. The difficulties in using this approach lie in defining the real error source. Their formulation can also become extremely complex. Hence, for practical usage it is better to say that fictitious-errors are introduced to cause an increase in the prediction error a-priori covariance matrix. (43, 50,38).

(2) Overweight the most recent data.

In this approach it is also recognised that the a-priori covariance matrix may be overly optimistic. This matrix, however, is not modified on the basis of adding the effects produced by fictitious-error in the dynamical model. Instead, a non-optimal filter algorithm is adopted which attaches a greater significance to the recent observations than the optimal filter does. The a-posteriori covariance matrix is modified to conform with the non-optimal algorithm. (50, 43, 38).

(3) Modelling of Parameter Uncertainties.

In this approach those portions of the dynamics (and/or measurement functions) which are poorly known are parametrised, and these parameters are considered as unknown quantities.

Instead of estimating these parameters, the filter can be redeveloped to include their uncertainty in the state equations. The parameter uncertainty will always degrade the state estimates, since we are not improving our estimate of these parameters themselves. This is a desirable feature that will tend to keep the covariance matrix sensitive to incoming information (43,50).

(4) Limited Memory Filtering.

In this approach conditioning of the estimate on old data is discarded in batches. This has the effect of keeping the estimates locked on to the more recently acquired observations. (41, 43, 50, 60).

All the approaches described above will prevent divergence, but they suffer from over estimating the error covariance matrix so that the estimates are 'noisier' than perhaps they need be. Another bad feature of these methods is that they contain various 'fudge'factors that require 'cut and try' experiments to fix.

(5) Adaptive Filtering

Other attempts to solve the divergence problem are the so-called 'adaptive filter' techniques. These approaches look at the residuals between the predictions and the measurements. These residuals should be small, random and should posses statistical properties consistent with the statistical parameters defined in the filter algorithm. The basic idea of adaptive filtering is to let the residuals themselves determine appropriate noise input levels (10, 41, 42, 43, 63, 67, 68).

Adaptive filtering is a more general and powerful technique than the methods described above, but again it usually over estimates the error covariance matrix. This is because it treats the input noise that is supposed to represent the model errors as a zero mean stochastic process and this need not be a true assumption.

The adaptive filter of Coggan and Wilson (10) attempts to alleviate this problem by removing bias, but this causes further problems in that only state-variable estimation is possible while bias is being removed.

2.3.2 Non-linear Problems.

Exact equations of the evolution of the conditional probability distribution function of the estimates of non-linear systems can be formulated (43). These equations, in fact, require the precise knowledge of an infinite set of parameters (e.g. all the statistical moments) and therefore only approximate solutions are practically possible (43).

An alternative approach to non-linear problems is to linearise the equations about a nominal state trajectory using a Taylor series expansion. A set of linear equations is then obtained in the state-deviations from the nominal. These equations can then be used in the Linear Kalman filter described in the previous section. Jaswinski (43) describes this 'Extended Kalman filter' and also some modifications based on iteration to improve the estimate. He also shows some comparisons between the approximate non-linear filters and the Extended Kalman filters.

His conclusions are that the differences between the approximate non-linear filters and the iterated Extended Kalman filter are negligible but the Extended Kalman filter, although showing bias, is probably the most useful because of its simplicity and speed of computation.

The problems of the Extended Kalman filters can be looked at in the framework of the divergence problems of the previous

section. Once a nominal trajectory has been chosen and the equations linearised, we are faced with a linear filtering problem, if divergence (or bias) occurs it is because the linear model does not accurately represent the real situation. Various techniques for dealing with this problem have been discussed and criticisms made (10, 41, 42, 43, 50, 63, 67) . Clearly the problem of adequately compensating for errors in the mathematical model is essentially unsolved.

The discussion above has shown that the Kalman approach to estimation possesses theoretical and practical advantages over alternative methods. It is for this reason that the basic Kalman approach is used in the development of a piece-wise continuous filter in the next section. The divergence problem which, as has been mentioned, is the most serious problem associated with Kalman filtering applications, is then examined in detail and an algorithm is developed to prevent the estimates diverging from the true states.

CHAPTER (3).

THEORETICAL DEVELOPMENTS.

PIECE-WISE CONTINUOUS FILTER AND ADAPTIVE FILTER.

(3.1) Prediction.

(3.1.1) Linear-stationary Case.

(3.1.2) Linear non-stationary Case.

(3.1.3) Non-linear Equations.

(3.1.4) State Variable Estimation.

(3.2) Non-linear Estimation.

(3.3) Adaptive Filtering.

3. Theoretical Developments.

Piece-wise Continuous Filter and Adaptive Filter.

The filter to be developed can be thought of as a prediction-correction procedure. Starting from a state of known mean and covariance, predictions are calculated for the time evolution of the mean andthe error covariance matrix.As observations become available these predictions are corrected to provide updated estimates of the state variables and the error covariance matrix.

The computation of the predictions is essentially the solution of a set of differential equations. At present the standard methods of solving differential equations on digital computers yield discrete solutions. However, solutions in continuous form have considerable advantages, especially in a control context. These advantages are that many standard mathematical tools (e.g. differentiation, int e gration, interpolation, extrapolation, etc.), become available.

3.1. Prediction.

Given a set of differential equations

$$\dot{x} = f(x,u) \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

where x is an n-dimensional state-vector and u is an m-dimensional vector of known parameters (e.g. forcing functions, control functions, etc.)

and given an initial condition

$x(t_o) = x_o$

Then find the solution $x(t)$ for $t_o \leq t \leq t_f$

The solution is to be continuous over this range.

3.1.1. Linear-Stationary Case.

Here the differential equations (1) can be simplified:

$$\dot{x} = F_1 x + F_2 u \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Where F_1 is an n x n known constant matrix and F_2 is an n x m known constant matrix.

Now approximating x by orthogonal polynomials (appendix 1) gives

$$x \simeq AP \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3).$$

Where A is an n x (p+1) matrix of coefficients and P is a (p+1) vector $(p_o ; p_1 ; \ldots\ldots p_i ; p_p)^T$

Where the p_i are a set of orthonormal polynomials.

Defining a matrix D such that

$DP = dP/dt$ and substituting this and (3) into (2)

gives $ADP = F_1 AP + F_2 u$

For this differential equation to be approximated optimally

$(ADP - F_1 AP - F_2 u, P) = 0$ (Appendix 1)

Where the left hand side is an inner product in the polynomial function space.

i.e. $$AD = F_1A + (F_2u,P) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

Incorporating the initial conditions, the polynomial approximation must satisfy the following equations.

$$AP(t_o) = x_o$$

$$ADP(t_o) = F_1x_o + (F_2u,P)\ P(t_o)$$

$$AD^2P(t_o) = F_1^{\,2}\ x_o + F_1(F_2u,P)\ P(t_o) + (F_2u,P)DP(t_o)$$

$$AD^3\ P(t_o) = F_1^{\,3}\ x_o + F_1^{\,2}(F_2u,P)\ P(t_o) + F_1(F_2U,P)D\ P(t_o) + (F_2u,P)D^2\ P(t_o)$$

etc.

These equations can be written in matrix form:

$$M = (P(t_o) \vdots DP(t_o) \vdots D^2\ P(t_o) \vdots \ldots\ldots \vdots D^p\ P(t_o))$$

$$N = (x_o \vdots F_1x_o + (F_2u,P)\ P(t_o) \vdots \text{etc})$$

and then the coefficient matrix for the optimal p^{th}-order polynomial approximation is:

$$A = N\ M^{-1} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5)$$

The matrix M depends only on the polynomials being used so that M^{-1} can be predefined and held in the computer as a constant matrix. The solution then is found by forming the matrix N, and performing the matrix multiplication NM^{-1}.

The computation of N involves the term $(F_2 u, P)$, this exists if u can be approximated by polynomials over the range

$t_o \leq t \leq t_f$

The exact solution may be written as an infinite series thus:

$$x = \sum_{i=o}^{\infty} b_i \, p_i \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

where b_i are n-dimensional vectors of coefficients, and because of the orthogonality of the p_i the error of the approximation can be estimated as,

$$e \simeq b_{p+1} \, p_{p+1}$$

therefore the total square error of the approximation is,

$$e_{tot} = \int_{t_o}^{t_f} e^T e \, dt \approx // \, b_{p+1} //^2 \qquad \ldots\ldots\ldots\ldots\ldots (7)$$

and if it is assumed that the series is monotonically convergent after some number of terms less than p then,

$$e_{tot} \approx // \, b_{p+1} //^2 < // \, A_{(p)} //^2 \qquad \ldots\ldots\ldots\ldots\ldots (8)$$

Where A(p) is the p^{th} column of A.

Then if

$// \, A_{(p)} // \not< (e_s)^{\frac{1}{2}}$ where e_s is some preset error bound, define k_s such that,

$$k_s // \, A_{(p)} // = (e_s)^{\frac{1}{2}} \qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (9)$$

$(k_s)^{1/p}$ then specifies a time scale factor which will define the range to ensure convergence of a p^{th}-order approximation to within the error bound. Note that this allows the time scale factor to increase if a larger range is acceptable.

3.1.2 Linear Non-Stationary Case.

Here the differential equations can be written,

$$\dot{x} = F_1(t)\, x + F_2(t)\, u \qquad \text{.....(10)}$$

Where $F_1(t)$ is now an n x n matrix of functions of t, and $F_2(t)$ is an n x m matrix of functions of t.

Assume the solution can be approximated by,

$$x \approx x^c + T \sum_{i=o}^{q} a_i\, q_i(t) \qquad \text{.....(11)}$$

a_i are n-vectors of coefficients

$q_i(t)$ are i^{th}-order scalar polynomials in t

T is an n x n diagonal matrix with

$T_{(i,i)} = t - t_i$ where t_i is the time at which the condition $x^c_{(i)}$ is given.

Usually the $t_i = t_o$ (i.e. initial value problems) but this method can be generally applied to the situation where the t_i are not all equal (i.e. split boundary condition problems).

Substituting (11) into (10) gives,

$$\dot{x} = T \sum_{o}^{q} a_i\, \dot{q}_i(t) + \sum_{o}^{q} a_i\, q_i(t)$$

$$= F_1(t) x^c + F_1(t)\, T \sum_{o}^{q} a_i\, q_i(t) + F_2(t)\, u$$

Therefore $\sum_{o}^{q} (T\dot{q}_i (t) + Iq_i (t) - F_1 (t)\, Tq_i (t))a_i$

$$= F_1 (t)\, x^c + F_2 (t)u \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (12)$$

i.e.

$$\sum C_i (t)\, a_i = B(t) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (13)$$

Where $C_i (t) = T\dot{q}_i (t) + Iq_i (t) - F_i (t)\, Tq_i (t)$

and $B (t) = F_1 (t)\, x^c + F_2 (t)\, u$

Making equation (13) exact at q+1 collocation points (appendix 2) gives;

$$C_o (t_1)\, a_o + C_1 (t_1)\, a_1 + \ldots\ldots\ldots = B (t_1) \quad \ldots\ldots\ldots\ldots (14)$$

$$C_o (t_2)\, a_o + C_1 (t_2)\, a_1 + \ldots\ldots\ldots = B (t_2)$$

" " " " " " " "

" " " " " " " "

" " " " " " " "

$$C_o (t_{q+1})\, a_o + C_1 (t_{q+1})\, a_1 + \ldots\ldots\ldots = B (t_{q+1})$$

These equations could in theory be solved for the nx(q+1) unknowns a_i , however, in practise this would require an enormous amount of storage. To overcome this problem nominal solutions, satisfying the specified conditions, are assumed for all the x_i except x_1 , this reduces the simultaneous differential equations (10) to a single differential equation for x_1. The solution of this equation by the method outlined above, provides the nominal trajectory of x_1. This is then used to update the nominal solutions of all the x_i in turn. This procedure is repeated until

all the solutions converge to the approximation defined by equation (11). To ensure that this approximation suitably represents the solution to the differential equation (10); the approximation error can be linked to the range in an analogous way to that defined by equations (7), (8), (9).

3.1.3 Non-Linear Equations.

Here the differential equations remain in their most general form defined by equation (1).

Using nominal trajectories x^*, that satisfy the specified conditions for x, these equations can be linearised.

$$f(x^*+dx,u) = f(x^*,u) + f_x^1(x^*,u)\,dx$$

Where dx are deviations from the nominal trajectory

$$dx = x - x^*$$

then

$$d\dot{x} = f_x^1(x^*,u)\,dx + f(x^*,u) - \dot{x}^* \quad \ldots\ldots\ldots (15)$$

Equation (15) is now linear and can be solved for dx by either of the above methods depending on whether the x^* are constant or not.

If the solution is to be obtained via the non-stationary method (collocation) then the nominal trajectories are updated via

$$x^* = x^* + dx/2$$

These nominal solutions will converge to the true solutions for a wide variety of situations (59, 70).

If the stationary method of solution is to be used (this has the advantage of being faster) then the nominal trajectories are updated via:

$$x^* = x^* + \overline{dx}/2$$

where $$\overline{dx} = \frac{1}{t_f - t_o} \int_{t_o}^{t_f} dx \, dt$$

i.e. $\overline{dx}$ is the time average of dx over the range $t_o \leq t \leq t_f$.

These nominal trajectories x^*, and therefore the deviations dx, will converge (59, 70) . When this convergence occurs the solution x is calculated as,

$$x = x^* + dx$$

This will not be the true solution but only the closest approximation that can be obtained by stationary equations.

3.1.4 State Variable Estimation.

The problem in its most general form is:
Given a mathematical model for the evolution of the state variables; and given a measurement model for the connection between the measurements and the states; then devise an algorithm for the evolution of the optimum state estimates and the evolution of the covariance matrix of the estimation errors.

In mathematical terms this can be expressed as follows:
Process Model:

$$\dot{x} = f(x,u,y) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (16)$$

Where x is the n-dimensional state vector.
u is an m-dimensional vector of known parameters (e.g. forcing functions, control functions, etc.).
and y is a r-dimensional vector of random disturbances.

Measurement Model:

$$z = h(x,v) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (17)$$

Where z is an h-dimensional vector of measurements
and v is an h-dimensional vector of random variables.
For the linear stationary case these equations can be simplified:

$$\dot{x} = F_1 x + F_2 u + F_3 y \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (18)$$

Where F_1 is nxn known constant matrix.
F_2 is nxm known constant matrix .
F_3 is nxr known constant matrix.

$$z = Hx + v \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (19)$$

Where H is h x n known constant matrix.

To solve this problem various initial conditions need to be specified.

Consider that:

$$E(y) = 0$$

and, $$\frac{1}{t_1 - t_o} \int_{t_o}^{t_1} E(y, yT)\, dt = Q$$

Assume that an estimate exists over the time range $t_o \leq t \leq t_1$ such that,

$\hat{x}(t)$ is an estimate of $x(t)$ $\quad t_o \leq t \leq t_1$

and $\tilde{x}(t) = x(t) - \hat{x}(t)$

and $\hat{V}(t_1) = \frac{1}{t_1 - t_o} \int_{t_o}^{t_1} E(\tilde{x}(t)\, \tilde{x}(t)^T)\, dt$ is known,

then a prediction $\bar{x}(t)$ can be found for $x(t)$ $\quad t_1 \leq t \leq t_2$

via $\dot{\bar{x}} = F_1 \bar{x} + F_2 u$ $\qquad$ with $\bar{x}(t_1) = \hat{x}(t_1)$

by the method discussed previously.

Now let

$$\bar{V}(t_2) = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} E(x - \bar{x})\, (x - \bar{x})^T\, dt$$

represent the average covariance matrix of the errors in the prediction $\bar{x}$ of x over the range $t_1 \leq t \leq t_2$

Now,

$$\bar{x}(t) = \emptyset\, (t,\ t - t_2 + t_1)\ \hat{x}(t - t_2 + t_1) + \Delta(t, t - t_2 + t_1)\ F_2\ u(t - t_2 + t_1)$$

where $\phi(t,\tau)$ is the fundamental matrix of the system defined by,

$$\frac{d}{dt}\,\phi(t,\tau) = F_1\,\phi(t,\tau)$$

with $\phi(\tau,\tau) = I$

and $\Delta(t,\tau) = \int_{\tau}^{t} \phi(t,\tau_1)\,d\tau_1$

For the linear time invariant situation $\phi(t,\tau)$ is only dependent on the value of $(t-\tau)$

Therefore,

$$\phi(t,t-t_2+t_1) = \phi(t_2,t_1)$$

Therefore,

$$x(t)-\bar{x}(t) = \phi(t_2,t_1)\,\tilde{x}(t-t_2+t_1)+\Delta(t_2,t_1)\,F_3\,y(t-t_2+t_1)$$

Then assuming $y(t-t_2+t_1)$ and $\tilde{x}(t-t_2+t_1)$ are independent random variables,

$$\bar{V}(t_2) = \phi\,\hat{V}(t_1)\,\phi^T + \Delta F_3\,Q\,F_3^T\Delta^T$$

Where $\phi = \phi(t_2,t_1)$

and $\Delta = \Delta(t_2,t_1)$

to simplify the terminology.

Now assume that measurements are available as continuous functions $z(t)$ for the time range $t_1 \leq t \leq t_2$ and consider that

$$\frac{1}{t_2-t_1}\int_{t_1}^{t_2} (vv^T)\,dt = R \quad \text{is given}$$

and that,

$$Ev = 0$$

Then the optimal estimate $\hat{x}(t)$ for $x(t)$ over the time range $t_1 \leq t \leq t_2$, assuming that $v(t)$ and $x(t)-\bar{x}(t)$ are independent Gaussian random processes, is given by,

$$\hat{x}(t) = \bar{x}(t) + K(z(t)-H\bar{x}(t))$$

Where K is an n x h matrix chosen to minimise the covariance matrix $\hat{V}(t_2)$ (appendix 3)

i.e. $$K = \bar{V}(t_2)\, H^T(H\bar{V}(t_2)\, H^T + R)^{-1}$$

and $$\hat{V}(t_2) = (I-KH)\, \bar{V}(t_2)$$

To summarise:

The piece-wise continuous estimator described above consists of the following operations.

Prediction:

Predict $\bar{x}(t)$ for $t_i \leq t \leq t_{i+1}$ via

$$\dot{\bar{x}} = F_1\bar{x} + F_2u \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(20)$$

with initial condition,

$$\bar{x}(t_i) = \hat{x}(t_i)$$

$$\bar{V}(t_{i+1}) = \phi\, \hat{V}(t_i)\, \phi^T + \Delta F_3\, Q\, F_3{}^T \Delta^T \quad \ldots\ldots\ldots\ldots(21)$$

$$K_{(i+1)} = \bar{V}(t_{i+1})\, H^T(H\, \bar{V}(t_{i+1})\, H^T + R)^{-1} \quad \ldots\ldots\ldots(22)$$

Estimation.

$$\hat{x}(t) = \bar{x}(t) + K_{(i+1)} (z(t)-H\bar{x}(t)) \text{ for } t_i \leq t \leq t_{i+1} \quad \ldots\ldots\ldots\ldots(23)$$

$$\hat{V}(t_{i+1}) = (I-K_{(i+1)}H)\, \bar{V}(t_{i+1}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots(24)$$

3.2. Non-Linear Estimation.

The equations for the process model cannot, in the non-linear situation, be simplified and must be treated in their general form (equation 16).

Using the same technique for linearisation described previously (equation 15) enables the non-linear system to be filtered by the linear filter just developed.

Clearly the estimation of unknown parameters can be treated in exactly the same way merely by regarding them as state variables and augmenting the state-vector.

The filters described above bear very close relationships to the Kalman filter and the Extended Kalman filter (31, 33, 43). The main difference is that the continuous and the differentiable matrix function $E(\tilde{x}(t)\ \tilde{x}(t)^T)$ has been discretised and replaced by the step function $\hat{V}(t)$. The result of this is to remove an awkward non-linear matrix differential equation (34, 38,43) from the algorithm. The avoidance of this matrix Riccati differential equation has immediate benefits in the context of continuous non-linear estimation because in this case the computation of the

state estimates and the error covariance matrix are coupled and the matrix Riccati equation would have to be solved in real time. Further benefits of the avoidance of the Riccati equation will become obvious as the adaptive filter is developed.

The similarities with the Kalman filters, however, are sufficient to allow the stability theorems developed for the latter (34, 38, 43)to be directly applied to this situation.

3.3 Adaptive Filtering.

The most serious problem encountered in practical applications of state variable and parameter estimation is that of the divergence of the estimates from the true values. This divergence is caused by the use of an inaccurate mathematical model.

Consider the true system equations to be represented by:

$$\dot{x}_s = f_s(x_s, u_s, y_s) \quad \text{.....(25)}$$

and the model to be represented by,

$$\dot{x}_m = f_m(x_m, u_m, y_m) \quad \text{.....(26)}$$

Clearly equation (25) can be written as,

$$\dot{x}_s = f_m(x_s, u_s, y_s) + f_s(x_s, u_s, y_s) - f_m(x_s, u_s, y_s)$$

and defining

$$F_4 w = f_s(x_s, u_s, y_s) - f_m(x_s, u_s, y_s)$$

Where F_4 is an n x ℓ matrix of the form $F_4(i,j)=0$ except for at most, one element in each row and column which may be unity; and w is an ℓ-vector, where ℓ is the dimensionality of the model errors.

Then,

$$\dot{x}_s = f_m(x_s, u_s, y_s) + F_4 w \quad \ldots\ldots\ldots\ldots\ldots\ldots (27)$$

and comparing this with equation (26) shows that the model represented by the function $f_m(\cdot)$ can produce the true state if it is disturbed by the 'fictitious inputs' $F_4 w$. In general of course $F_4 w$ will be unknown, but approximating this model compensation term by a Gaussian random process will at least make the errors in using the inaccurate model random and unbiased.

The problem of the adaptive filter is then to discover the mean (Ew) and covariance matrix $(E(w-Ew)(w-Ew)^T)$ and the matrix F_4 for the random approximation to the model errors. A realisation of the above factors will be called a 'Model Error Compensation' strategy. Clearly the optimum strategy will produce a well tuned filter, that is a filter in which the estimation error covariance matrix is minimised while at the same time prevented from becoming over optimistic.

Considering $f_m(\cdot)$ to be linear and stationary and approximating $F_4 w$ by a random process,

$$\dot{x} = F_1 x + F_2 u + F_3 y + F_4 Ew$$

where the subscripts m and s have been dropped to simplify the terminology in the following developments.

Now Ew is an unknown vector of parameters and could, in theory, be estimated by forming an augmented state vector,

$$\begin{bmatrix} \dot{x} \\ \dot{Ew} \end{bmatrix} = \begin{bmatrix} F_1 & F_4 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ Ew \end{bmatrix} + \begin{bmatrix} F_2 \\ 0 \end{bmatrix} u + \begin{bmatrix} F_3 \\ 0 \end{bmatrix} y \quad \text{.....(28)}$$

In practice this increases the dimensionality of the filter problem, which causes a disproportionate increase in computation time, and is therefore to be avoided if possible. The technique used follows that described by Schmidt (50) and Jazwinski (43) and consists of writing out formally the filter equations for the augmented system equation (28) but then only to take the estimates of x leaving the Ew constant. This means that the value of Ew will not be improved by the filter but the error committed by not improving the estimate will be modelled so that its effect on the estimates of x will be taken into account.

The modification to the filter just proposed produces the following equation: (appendix 4).

Prediction:

for the range $t_i \leq t \leq t_{i+1}$ via

$$\dot{\bar{x}} = F_1 \bar{x} + F_2 u + F_3 y + F_4 Ew \quad \text{.....(29)}$$

with the initial condition $\bar{x}(t_i) = \hat{x}(t_i)$

$$\bar{V}(t_{i+1}) = \phi \hat{V}(t_i) \phi^T + \Delta F_3 Q F_3^T \Delta^T + \phi \hat{C}(t_i) + \hat{C}^T(t_i) \phi^T$$

$$+ \Delta F_4 V_w F_4^T \Delta^T \quad \text{.....(30)}$$

where

$$\hat{C}(t_i) = \frac{1}{t_i - t_{i-1}} \int_{t_{i-1}}^{t_i} E(x-\hat{x})(w-Ew)^T \, dt \; F_4^T \Delta^T$$

$$\bar{C}(t_{i+1}) = \bar{\phi} \, \hat{C}(t_i) + \Delta F_4 \, V_w \, F_4^T \Delta^T \quad \text{.....(31)}$$

where

$$\bar{C}(t_{i+1}) = \frac{1}{t_{i+1} - t_i} \int_{t_i}^{t_{i+1}} E(x-\bar{x})(w-Ew)^T \, dt \; F_4^T \Delta^T$$

and

$$V_w = \frac{1}{t_{i+1} - t_i} \int_{t_i}^{t_{i+1}} E(w-Ew)(w-Ew)^T \, dt$$

Estimation:

$$K(i+1) = \bar{V}(t_{i+1}) \, H^T \, (H \, \bar{V}(t_{i+1}) H^T + R)^{-1} \quad \text{.....(32)}$$

$$\hat{x}(t) = \bar{x}(t) + K(i+1) \, (z(t) - H \, \bar{x}(t)) \qquad t_i \leq t \leq t_{i+1} \quad \text{.....(33)}$$

$$\hat{V}(t_{i+1}) = (I - K(i+1) \, H) \, \bar{V}(t_{i+1}) \quad \text{.....(34)}$$

$$\hat{C}(t_{i+1}) = (I - K(i+1) \, H) \, \bar{C}(t_{i+1}) \quad \text{.....(35)}$$

These equations will work if Ew and V_w define consistent stochastic processes. That is V_w must accurately represent the covariance matrix of the errors (w-Ew). This means that an error in the value of Ew will deteriorate the accuracy of the state estimates by requiring V_w to increase, which will result in a larger estimation error covariance matrix $\hat{V}(t)$.

If the values of Ew and V_w define consistent stochastic processes then the residuals $(z-H\bar{x})$ will be a zero mean white noise process with,

$$\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} E(z-H\bar{x})\,(z-H\bar{x})^T\,dt = H\,\bar{V}(t_{i+1})H^T+R \quad .....(36)$$

This equation provides a consistency check on the value of V_w. Assume that V_w can be represented by a diagonal matrix Ic where c is a scalar; let $\bar{c}$ be an a-priori guess for c, and define $\tilde{c}=c-\bar{c}$ then,

$$\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} E(z-H\bar{x})\,(z-H\bar{x})^T\,dt = H\,\bar{V}\,H^T+R+(H\Delta F_4)(H\Delta F_4)^T\tilde{c}$$

Therefore:

$$\tilde{c} = \frac{\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} E(z-H\bar{x})^T(z-H\bar{x})\,dt-\mathrm{Trace}(H\,\bar{V}\,H^T+R)}{\mathrm{Trace}((H\Delta F_4)\,(H\Delta F_4)^T)} \quad .....(37)$$

Whence $\bar{c}$ is updated to $(\bar{c}+\tilde{c})$ unless this is negative in which case $\bar{c}=o$.

Equation (37) requires a value for

$$\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} E(z-H\bar{x})^T\,(z-H\bar{x})\,dt$$

representing this by $g(t_{i+1})$, and using an exponential filter gives,

$$g(t_{i+1}) = g(t_i) + \alpha\left(\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} (z-H\bar{x})^T(z-H\bar{x})\,dt - g(t_i)\right)$$

.....(38)

where $0 \leq \alpha \leq 1$

Equation (38) is a recursive relation for the scalar quantity g(t) which is an estimate of the trace of the covariance matrix of the residuals. This equation along with equation (37) can be implemented alongside the filter algorithm discussed previously. This will ensure that the statistics used in the filter are consistent with the statistics of the residuals.

The value found for V_w from equations (37) and (38) will be consistent with whatever value of Ew is used, but clearly to optimise the performance of the adaptive filter some method is needed to obtain an accurate estimate of Ew. It is obvious from its definition that V_w will be minimised when Ew is known precisely and from equation (37) this means that an accurate Ew will minimise the quantity g(t) defined in equation (38). Therefore the computation of a value for Ew is similar to optimal control problems designed to minimise an objective function of the form

$$J = \frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} E(z-H\bar{x})^T(z-H\bar{x})\,dt$$

Let $\bar{w}$ = some a-priori prediction of Ew and,

let $\tilde{w} = Ew - \bar{w}$.

Let $\bar{\bar{x}}$ represent the prediction of x with Ew replaced by $\bar{w}$ then,

$$\bar{\bar{x}} = \bar{x} + \Delta F_4 \tilde{w}$$

Then:

$$J = \frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} E(z-H\bar{x})^T(z-H\bar{x})-(z-H\bar{x})^T H\Delta F_4\tilde{w}$$

$$-(H\Delta F_4\tilde{w})^T(z-H\bar{x})+(H\Delta F_4\tilde{w})^T(H\Delta F_4\tilde{w})\ dt$$

Setting $\frac{\partial J}{\partial w_i} = 0$ for all i gives,

$$\frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} E(z-H\bar{x})^T\ dt\ H\Delta F_4 = (H\Delta F_4\tilde{w})^T H\Delta F_4$$

From which,

$$\tilde{w} = \frac{1}{t_{i+1}-t_i}((H\Delta F_4)^T(H\Delta F_4))^{-1}(H\Delta F_4)^T\int_{t_i}^{t_{i+1}} E(z-H\bar{x})dt \quad ..(39)$$

and the prediction of Ew is updated to be $\bar{\bar{w}}=\bar{w}+\tilde{w}$.

To calculate $\frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} E(z-H\bar{x})\ dt$ represent this by

$\gamma(t_{i+1})$ then use an exponential filter

$$\gamma(t_{i+1}) = \gamma(t_i)+\beta(\frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} (z(t)-H\bar{x}(t))dt-\gamma(t_i))$$

$$\ldots\ldots\ldots\ldots(40)$$

where $0 \leq \beta \leq 1$

Equation (40) is a recursive relation for the mean of the residuals and can be simply implemented along with the filter.

To summarise, the adaptive filter developed so far consists of the prediction/estimation equations (29 to 35) along with the following relationships.

$$E\underline{w} = \bar{\underline{w}} \qquad \ldots\ldots(41)$$

$$\underline{\gamma}(t_{i+1}) = \underline{\gamma}(t_i) + \beta\left(\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} (\underline{z}(t)-H\bar{\underline{x}}(t)\,dt - \underline{\gamma}(t_i)\right) \qquad \ldots\ldots(42)$$

$$\tilde{\underline{w}} = ((H\Delta F_4)^T(H\Delta F_4))^{-1}(H\Delta F_4)^T\underline{\gamma}(t_{i+1}) \qquad \ldots(43)$$

$$\bar{\underline{w}} = \bar{\underline{w}} + \tilde{\underline{w}} \qquad \ldots\ldots(44)$$

$$g(t_{i+1}) = g(t_i) + \alpha\left(\frac{1}{t_{i+1}-t_i}\int_{t_i}^{t_{i+1}} (\underline{z}-H\bar{\underline{x}})^T(\underline{z}-H\bar{\underline{x}})\,dt - g(t_i)\right) \qquad \ldots\ldots(45)$$

$$V_{\underline{w}} = I\,\bar{c} \qquad \ldots\ldots(46)$$

$$\tilde{c} = \frac{g(t_{i+1}) - \text{Trace}\,(H\,\bar{V}(t_{i+1})H^T + R)}{\text{Trace}\,((H\Delta F_4)\,(H\Delta F_4)^T)} \qquad \ldots\ldots(47)$$

$\bar{c} = \bar{c} + \tilde{c}$ unless this is negative when,

$\bar{c} = o$ .....(48)

The algorithm described above has similarities with the adaptive filters of Coggan and Wilson (10 , 38) and Jazwinski (43,63), but contains certain advantages. The main advantage is that both the mean and the covariance of the model error compensation term are adjusted. This means that the filter will converge with a smaller covariance term than the filter described by Jazwinski. This results in the covariance of the estimation errors being smaller. At the same time the consistency requirement (equation 36) ensures that this smaller covariance matrix is not over optimistic. The Coggan and Wilson algorithm also provides, in a way, for the updating of the mean of the compensation term, by removing measured bias from the estimates. But their method results in the inability to perform parameter estimation while bias is being removed. The filter described above has no such limitation provided the steps (equation 43 and 44) are performed only every θ^{th} time increment (where θ is as yet undetermined) this allows the filter to respond as much as it can to changing conditions, so that the compensation term is only conditioned on error caused by poor model.

There are a few remaining problems to be discussed before the theoretical development of this new adaptive filter is completely general. The most important of these is the determination of the matrix F_4. This matrix will, in general, be unknown. Although in some situations there may be some information available about the nature of F_4. Such a situation is when the true system equations (25) are known but considered too complicated

for 'on line' filtering. Therefore a simplified model (26) is selected, in which case there does exist knowledge of the model error committed.

In general situations, however, equation (25) is unknown, so that all that can be said about F_4 is that $F_4(i,j)=o$ except for, at most, one element in each row and column, which may be unity. The method proposed to find which particular matrix of this form represents F_4 is as follows. The dimension of the error compensation term (ℓ) is taken to be unity, this means F_4 is reduced to an n-dimensional column vector F_4^1 with each element zero except, at most, one. Each column vector of this type is tried in the filter and the vector that produces a minimum for g(t) equation (45) is selected as the first column of F_4. That is

$$F_4(j,1) = \overset{*}{F}_4^1(j) \qquad \text{for } j=1 \text{ to } n$$

Where $\overset{*}{F}_4^1$ is the vector that minimises g(t)

The dimension (ℓ) is then increased to two and the same process used to fix the second column of F_4. This is continued until $\ell = h$ the dimension of the measurement vector. This is the largest number of independent variables that can be computed via equation (43) and therefore restricts the method to finding model error compensation vectors with dimension less than or equal to h

The remaining problems are associated with the choice of the values of α and β in equations (38 and 40) and with the

number of time increments (θ) that must elapse before each implementation of the equations (43, 44). Clearly the choice of these parameters affect the optimality of the performance of the adaptive filter.

These points will be discussed in the following sections, as no general result can, at present, be developed from theoretical considerations.

The complete adaptive filter can now be summarised as follows:

Initially at time t_i specify,

$\hat{x}(t_i)$, $\hat{C}(t_i)$, $\hat{V}(t_i)$, $g(t_i)$, $\gamma(t_i)$, $\bar{c}$, $\bar{w}$, and F_4

Then predict for the range $t_i \leq t \leq t_{i+1}$ via,

$$\dot{\bar{x}} = F_1\bar{x} + F_2u + F_3y + F_4\bar{w} \quad \text{.....(49)}$$

with $\bar{x}(t_i)=\hat{x}(t_i)$

$$\bar{V}(t_{i+1}) = \phi\hat{V}(t_i)\phi^T + \Delta F_3 Q F_3{}^T\Delta^T + \phi\hat{C}(t_i) + \hat{C}^T(t_i)\phi^T + (\Delta F_4)(\Delta F_4)^T\bar{c} \quad \text{.....(50)}$$

$$\bar{C}(t_{i+1}) = \phi\hat{C}(t_i) + \Delta F_4 V_w F_4{}^T\Delta^T \quad \text{.....(51)}$$

Take measurements,

$$z(t_{i+1}) = H\, x(t_{i+1}) + v \quad \text{.....(52)}$$

then, $$\gamma(t_{i+1}) = \gamma(t_i) + \beta\left(\left(\frac{1}{t_{i+1}-t_i}\right)\int_{t_i}^{t_{i+1}} (z(t) - H\bar{x}(t))dt - \gamma(t_i)\right)$$

.....(53)

$$\tilde{w} = ((H\Delta F_4)^T(H\Delta F_4))^{-1}(H\Delta F_4)^T \gamma(t_{i+1}) \quad \ldots\ldots(54)$$

$$\bar{w} = \bar{w} + \tilde{w} \quad \ldots\ldots(55)$$

$$g(t_{i+1}) = g(t_i) + \alpha\left(\left(\frac{1}{t_{i+1}-t_i}\right)\int_{t_i}^{t_{i+1}} (z-H\bar{x})^T(z-H\bar{x})\,dt - g(t_i)\right) \quad \ldots\ldots(56)$$

$$\tilde{c} = \frac{g(t_{i+1}) - \text{TRACE}(H\bar{V}(t_{i+1})H^T+R)}{\text{TRACE}((H\Delta F_4)(H\Delta F_4)^T)} \quad \ldots\ldots(57)$$

$$\bar{c} = \bar{c} + \tilde{c} \text{ unless this is negative, when } \bar{c}=o \quad \ldots\ldots(58)$$

$$K = \bar{V}(t_{i+1})H^T(H\bar{V}(t_{i+1})H^T+R)^{-1} \quad \ldots\ldots(59)$$

Now estimate,

$$\hat{x}(t) = \bar{x}(t) + K(z(t)-H\bar{x}(t)) \quad t_i \leq t \leq t_{i+1} \quad \ldots\ldots(60)$$

$$\hat{V}(t_{i+1}) = (I-KH)\,\bar{V}(t_{i+1}) \quad \ldots\ldots(61)$$

$$\hat{C}(t_{i+1}) = (I-KH)\,\bar{C}(t_{i+1}) \quad \ldots\ldots(62)$$

Equations 49 to 62 are now repeated for ten samples and the final value of g(t) is stored. The matrix F_4 is then changed as described previously and the equations 49 to 62 repeated for a further ten samples. This is continued until all possible F_4's have been tried and the F_4 that minimises g(t) selected. The equations 49 to 62 are then used recursively for all subsequent samples.

CHAPTER (4).

DISCUSSION OF COMPUTER ALGORITHMS.

(4.1) Linear Stationary Prediction.

(4.2) Linear Non-stationary Prediction.

(4.3) State Variable and Parameter Estimation.

(4.4) Adaptive Estimation.

Chapter (4). Discussion of Computer Algorithms.

The programmes to be described here have all been written in 'BASIC' for use on the Department's Honeywell 316 computer. This language was chosen because of its ease of implementation and its facility for on-line programme development. The 'BASIC' compiler was augmented with subroutines written in 'FORTRAN'. These subroutines perform the following matrix manipulations: Matrix inversion; matrix transposition; matrix addition, subtraction and multiplication by a scalar; and matrix multiplication (appendix 14). The Programme Listings are shown in the appendices referred to in each section.

4.1 Linear-Stationary Prediction.

The set of orthogonal polynomials used in this programme were calculated from the relationship,

$$\int_{-1}^{+1} P\,P^T\,d\tau = I \qquad \text{.....(appendix 5)}$$

That is the p_i are orthonormal over the range $-1 \leq \tau \leq 1$ where,

$$\tau = \frac{2t - t_o - t_f}{t_f - t_o}$$

and equation (1) section 3.1.1 becomes,

$$\frac{dx}{d\tau} = \frac{dt}{d\tau} f(x,u)$$

i.e. $$\frac{dx}{d\tau} = \frac{(t_f - t_o)}{2} f(x,u) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (63)$$

The matrix D defined such that $DP = dP/d\tau = \dot{P}$ can be calculated from,

$$(DP,P) = (\dot{P},P)$$

i.e. $$D = (\dot{P},P) = \int_{-1}^{+1} \dot{P}\, P^T \, d\tau \quad \ldots\ldots\ldots\ldots\ldots\ldots (64)$$

(appendix 6).

The matrix M defined by,

$$M = (P(t_o) : D\,P(t_o) : D^2P(t_o) : \text{etc.})$$

is easily computed and its inverse found (appendix 7).

The programme can now be written following the theoretical development in section (3.1.1) for three separate cases:

1. u(t) is constant over the range $t_o \leq t \leq t_f$
2. u(t) is known in terms of the polynomials P for the range $t_o \leq t \leq t_f$.
3. u(t) is known at any point (t) in the range $t_o \leq t \leq t_f$. (figure 1) (appendix 8).

For (1) above the matrix N (see section 3.1.1) reduces to,

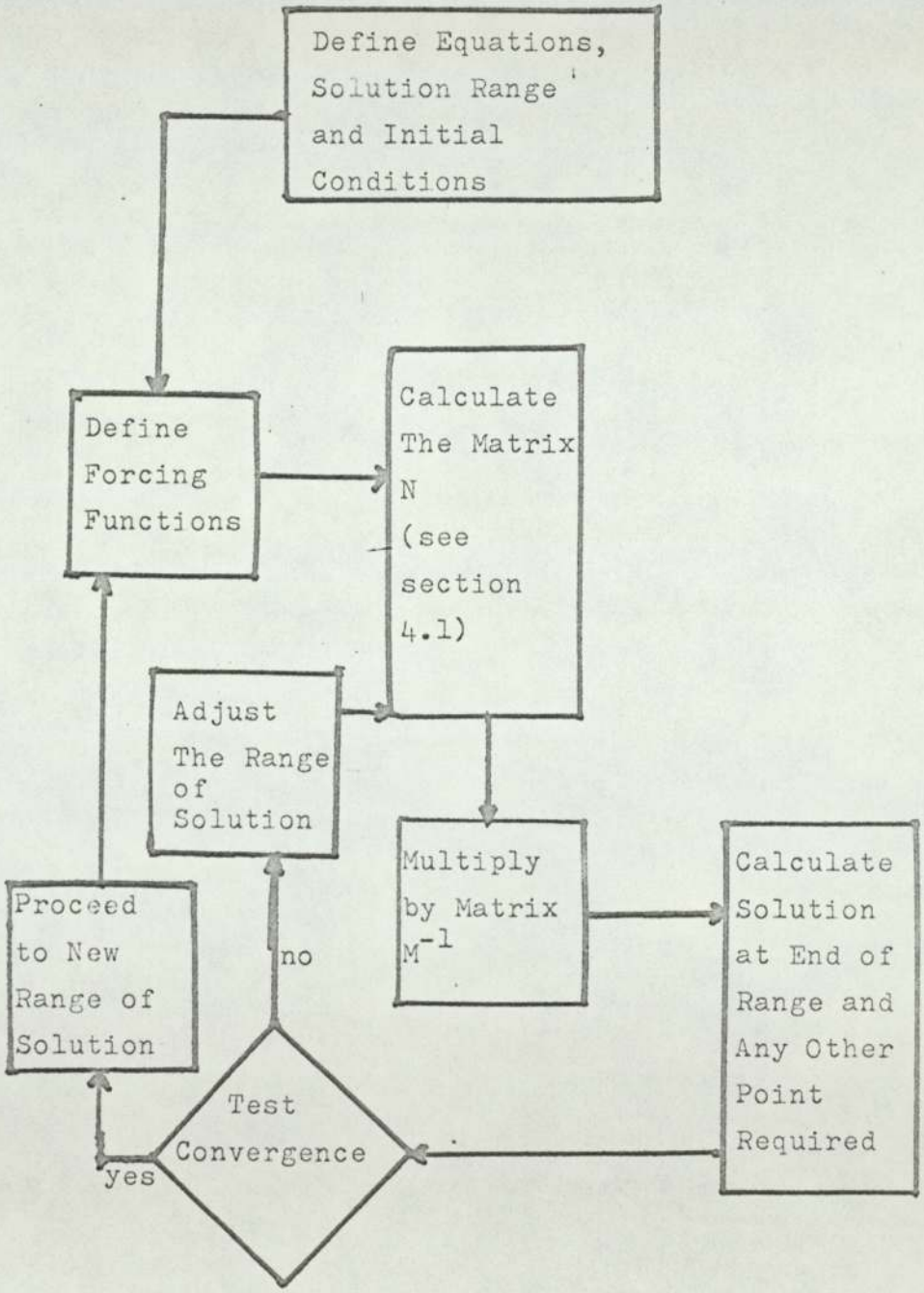
$$N = (x_o : F_1 x_o + (F_2 u, P)\, P(t_o) : F_1^2 x_o + F_1(F_2 u, P) P(t_o) \ldots\ldots)$$

because every differential coefficient of u with respect to t is zero. This is a special case of (2) above as a constant u can be readily expressed in terms of the polynomials p.

For (2) the matrix N is easily calculated because the

Figure (1).

Diagram of Linear Stationary Programme.



term (F_2u,P) is just the coefficient matrix of the polynomial representation of u.

For (3) it is clear that before N can be calculated it is necessary to compute the coefficient matrix (F_2u,P) for the polynomial representation of u. This is performed via Gaussian Quadrature.

$$(u,P) = \int_{-1}^{+1} uP^T \, d\tau$$

$$= \sum_{i=1}^{8} d_i \, u(\tau_i) \, P^T(\tau_i)$$

$$= (u(\tau_i) \; u(\tau_2) \; \ldots.) \begin{bmatrix} d_1 \; P^T(\tau_1) \\ d_2 \; P^T(\tau_2) \\ \text{etc} \end{bmatrix}$$

where the d_i and τ_i are the 8-point Guassian weights and abscissae (appendix 9). The 8-point formula was chosen to combine high accuracy with good computation speed.

The programmes just described were written to produce 10^{th} order polynomial approximations with the range adjustable to make the solutions accurate to within the preset error bound $e_s=10^{-4}$ (see section 3.1.1). The choice of 10 for the order of the polynomial approximation is somewhat arbitrary, its optimisation depends on a trade-off between the time of calculation of the coefficients, (which increases with the order), and number of steps taken to cover the full time-range of interest, (which decreases with the order). No general theoretical statements can be made about this balance as it depends upon the differential equations to be solved.

Results of the application of these programmes are shown and discussed in chapter 5.

4.2 Linear Non-Stationary Prediction.

From section 3.1.2 it can be seen that the polynomials used in this programme must be such that the zeroes of each $q_i(t)$ are real, distinct and lie within the range $t_o \leq t \leq t_f$.
Normalising this range by substituting,

$$\tau = \frac{2t-t_o-t_f}{t_f-t_o} \qquad t_o \leq t \leq t_f$$

shows that the zeroes of each $q_i(\tau)$ must lie within the range $-1 \leq \tau \leq +1$.
Now defining a variable s such that,

$$COS(s) = \tau$$

and defining functions

$$q_i(s) = COS(is)$$

It is clear from the following identity (69)

$$COS((i+2)s) = 2COS(s)\ COS((i+1)s)-COS(is)$$

i.e. $$q_{i+2}(s) = 2\tau q_{i+1}(s)-q_i(s)$$

that the $q_i(s)$ are i^{th} order polynomials in γ with

$$q_o(\gamma) = 1$$

$$q_1(\gamma) = \gamma = \cos(s)$$

$$q_2(\gamma) = 2\gamma^2 - 1 = 2\cos^2(s) - 1$$

etc.

What is more, it is obvious that $-1 \leq \gamma \leq +1$ for any value of s. Therefore the zeroes of $q_i(\gamma)$ are real, distinct and lie between -1 and +1 for all i; and are given by,

$$q_i(\gamma) = o$$

which implies,

$$\gamma = \cos\left(\frac{(2n-1)}{2i}\right) \qquad \text{for } n=1,2,3,\text{etc.}$$

These polynomials are 'Chebycheff Polynomials' and are orthogonal over the range -1 to +1 with respect to the weighting function

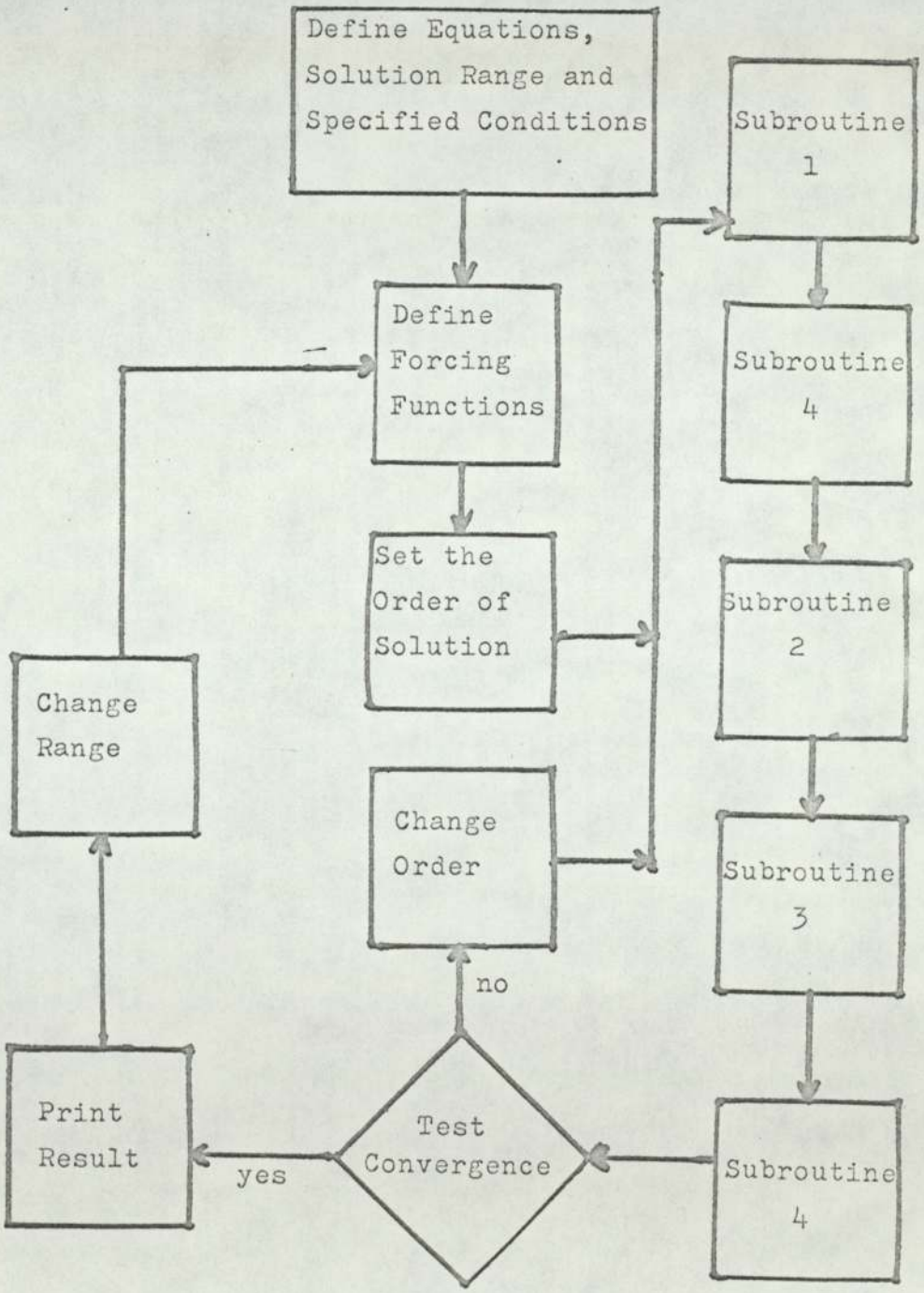
$$f_w(\gamma) = 1/\sqrt{1-\gamma^2}$$

i.e.
$$\int_{-1}^{+1} f_w(\gamma)\, q_i(\gamma)\, q_j(\gamma)\, d\gamma = o \quad i \neq j$$
$$= \frac{\pi}{2} \quad i = j$$

The programme to solve differential equations by this method consists of a BASIC programme which accesses four FORTRAN sub-routines (figure2) (appendix 10). These subroutines perform the following operations.

Figure (2).

Diagram of Collocation Programme.



Subroutine (1) Calculates the collocation points.

Subroutine (2) Sets up the simultaneous equations as described in section 3.1.2.

Subroutine (3) Solves these equations for the coefficients of the polynomial approximation.

Subroutine (4) Calculates the value of the approximation at some chosen points.

Results of the application of this programme are shown and discussed in chapter 5.

4.3 State Variable and Parameter Estimation.

The programme to achieve estimation was developed by adding the algorithm described in section 3.2 to the prediction programmes described above. (Figure 3). Only the estimation programme with linear stationary prediction has been developed. This is because it produces the prediction in a shorter computation time than the other methods. (appendix 11). Any error incurred by neglecting any non-stationary aspects of the equations over each range is absorbed into the statistics describing the random disturbances on the process.

Results of the application of the estimation programme are shown and discussed in chapter 5.

4.4 Adaptive Estimation.

This programme was developed from the estimation programme just described by adding the modifications discussed in section 3.3 (Figure 4) (appendix 12).

Figure (3).

Diagram of the Standard Filter.

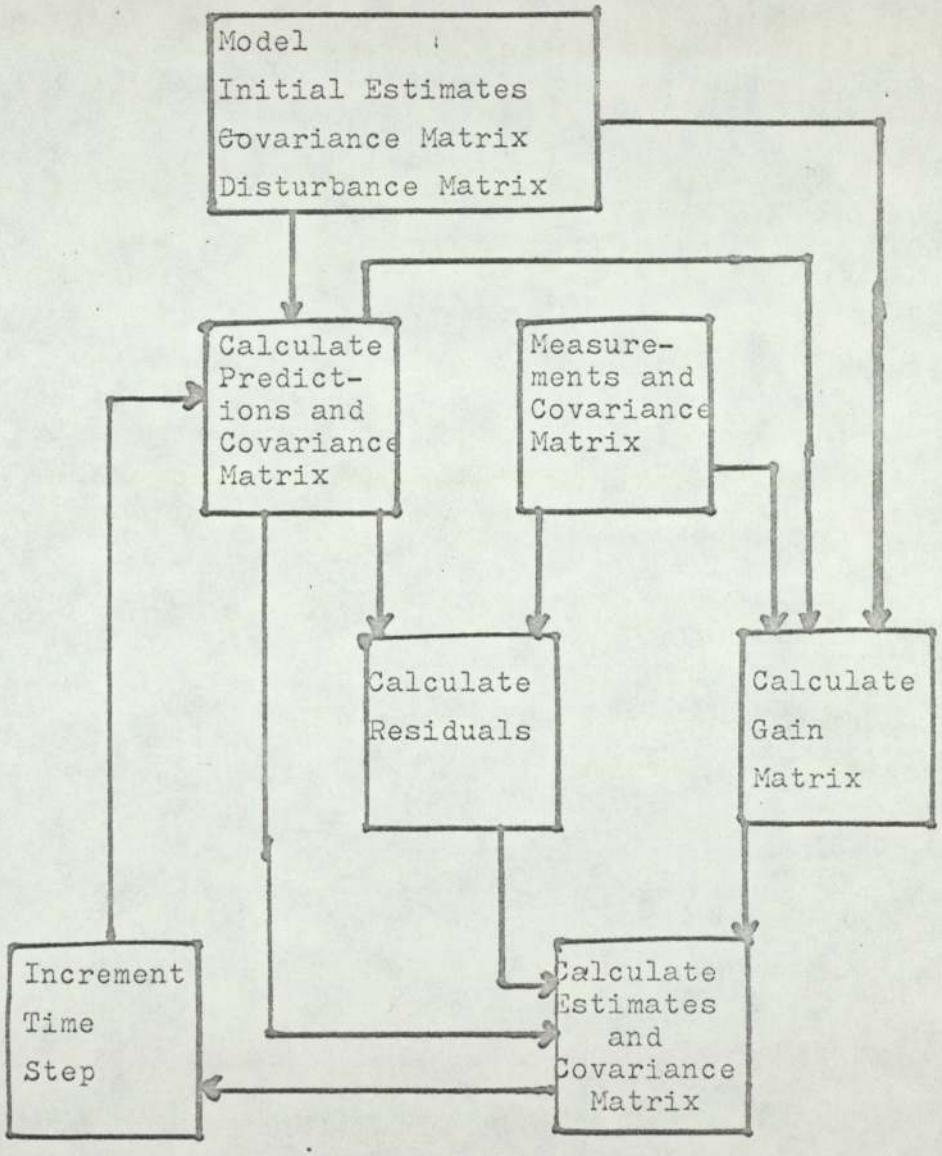
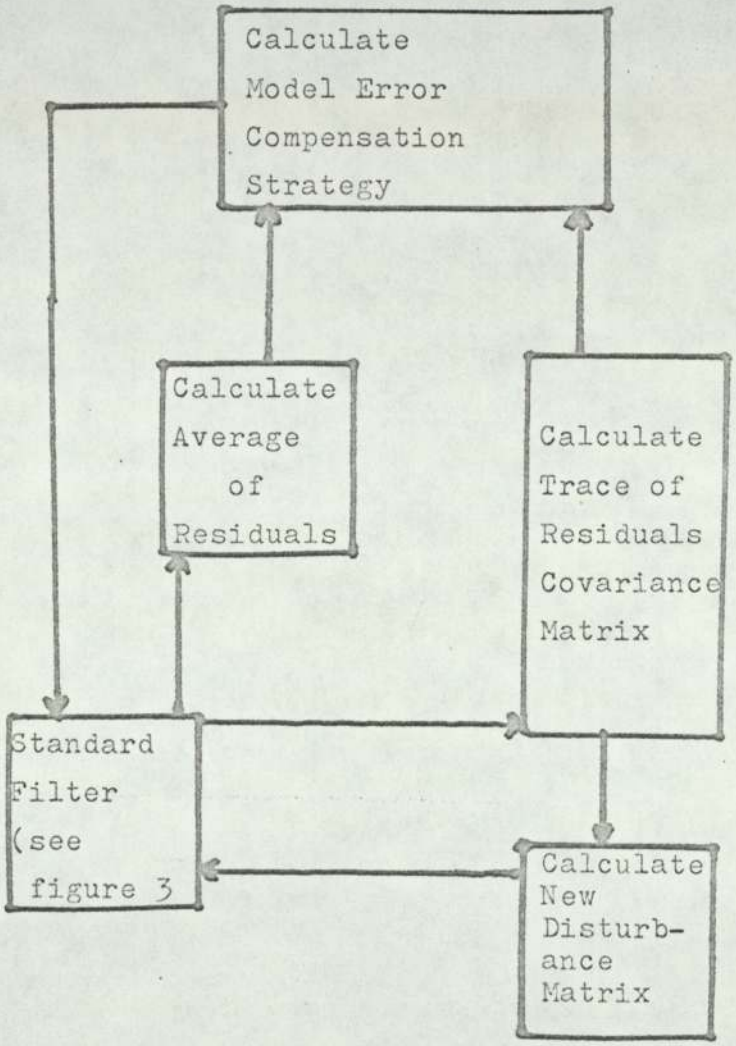


Figure (4).

Diagram of the Adaptive Filter.



Results of the application of this programme and investigations into the effects of different values of the parameters α, β, and θ are shown and discussed in chapter 5.

CHAPTER (5).

DISCUSSION OF RESULTS.

5. Discussion of Results.

5.1 Simulation.

The application of the simulation programmes described (4.1, 4.2) is quite straight forward. Results are shown for two systems.

5.1.1 Linear System (Figure 5).

$$\dot{A} = -(k_1 + q/v)A + k_2 B + q/v\, A_o$$

$$\dot{B} = k_1 A - (k_2 + k_3 + k_4 + q/v)B$$

$$\dot{C} = k_3 B - q/v C$$

$$\dot{D} = k_4 B - q/v D$$

with $k_1 = 0.538\ \text{min}^{-1}$; $k_2 = 0.385\ \text{min}^{-1}$; $k_3 = 0.062\ \text{min}^{-1}$; $k_4 = 0.246\ \text{min}^{-1}$; and $q/v = 0.308\ \text{min}^{-1}$.

Figure (6) shows the results obtained by simulating this system with the linear stationary programme described. (4.1). Figure (7) shows the results obtained with the collocation programme. (4.2).

5.1.2 Non Linear System (Figure 8).

$$\dot{x}_1 = -(k_1 + q/v_1)x_1 + q/v_1\, x_o$$

$$\dot{x}_2 = -(k_2 + q/v_2)x_2 + q/v_2\, x_1$$

$$\dot{y}_1 = -q/v_1 y_1 - (k_1 H/pc)x_1 + q/v_1\, y_o$$

$$\dot{y}_2 = -q/v_2 y_2 - (k_2 H/pc)x_2 + q/v_2\, y_1$$

with $k_1 = \text{EXP}(36.49 - (12100/y_1))\ \text{min}^{-1}$

$k_2 = \text{EXP}(36.49 - (12100/y_2))\ \text{min}^{-1}$

$H = 10$ cal/gm.mole

$q = 1$ litre/min

$v_1 = 3$ litres

$v_2 = 2$ litres

$c = 1$

$p = 1$

Figures (9, 10) show the results obtained by simulating this system with the linear stationary programme described (4.1). Figures (11, 12) show the results obtained with the collocation programme (4.2).

5.1.3 Discussion of Simulations.

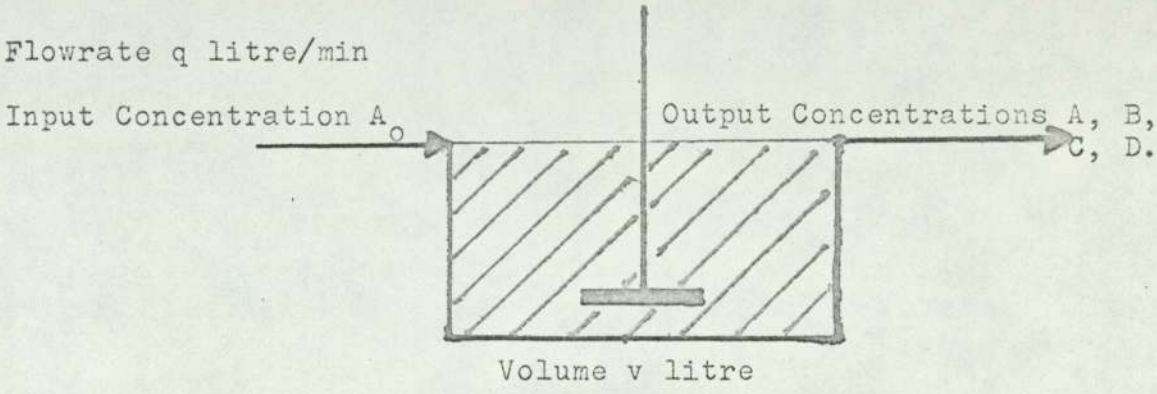
Figures (6, 7) show the results obtained when the system (5.1.1) was disturbed by the deterministic step function (A_o) shown. Comparison of Figures (6) and (7) show that both the stationary programme and the collocation programme produce identical results with the automatic range of solution effectively varying the range as the system changes from a steady state to a dynamic state.

The collocation programme, however, took very much longer to compute the solution than did the stationary programme. This is because it is a more general programme, capable of dealing with non-stationary problems and also capable of solving problems with a wide variety of specified conditions (e.g. split boundary value problems, or conditions in functional form).

Figure (6a) shows the results obtained by the stationary programme when the system was disturbed by the random function A_o

Figure (5).

Single Stage Stirred Tank Reactor With Linear Reaction.



Reaction In Tank.

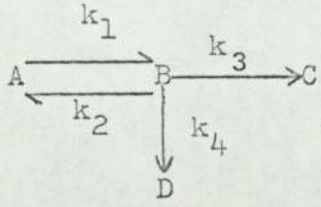


Figure (6).

Results of Simulation (Linear Stationary Programme).

(~)-Simulated : (x x)-Analytical Solution.

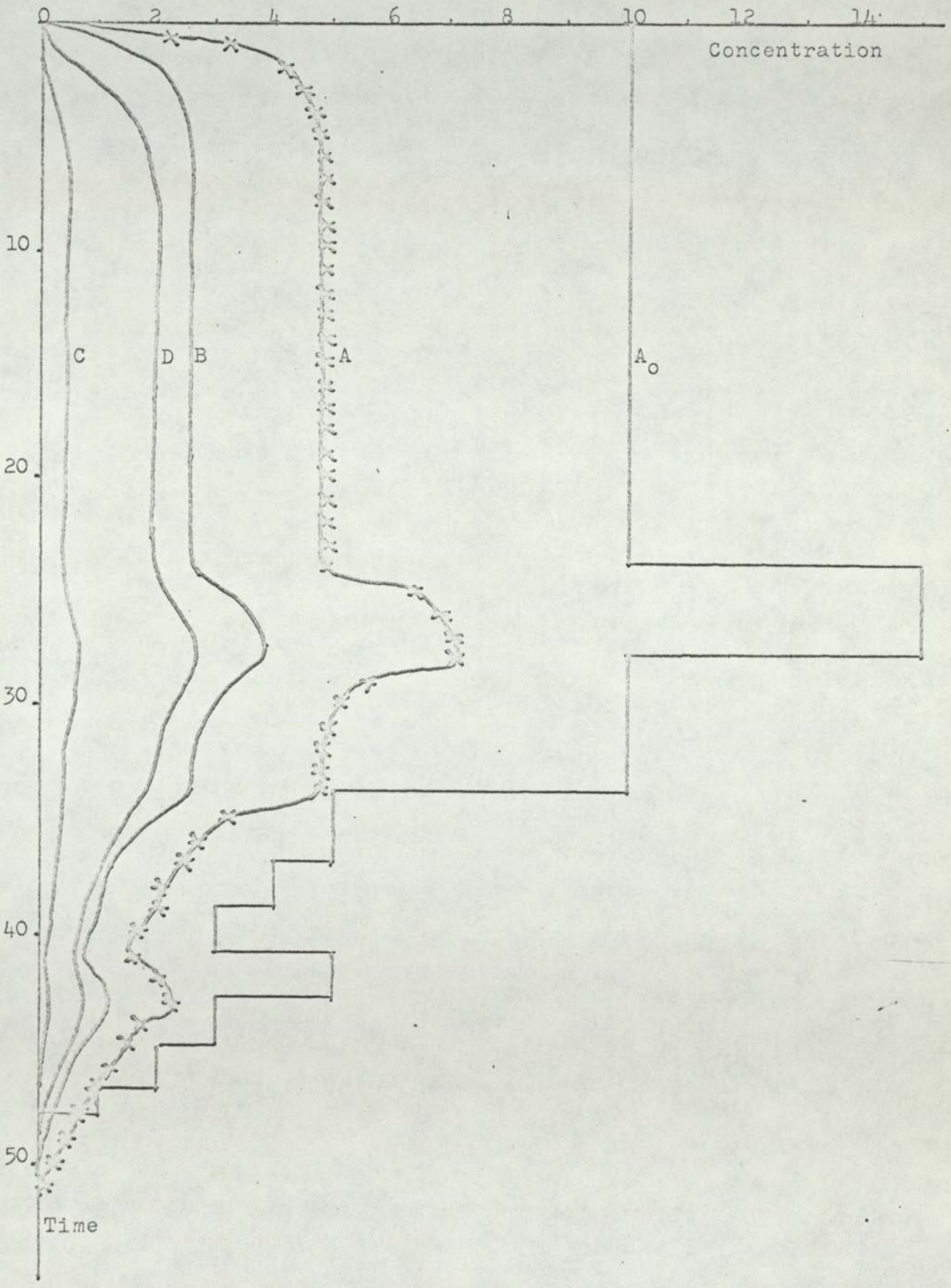


Figure (6a).

Results of Simulation with Random Input.

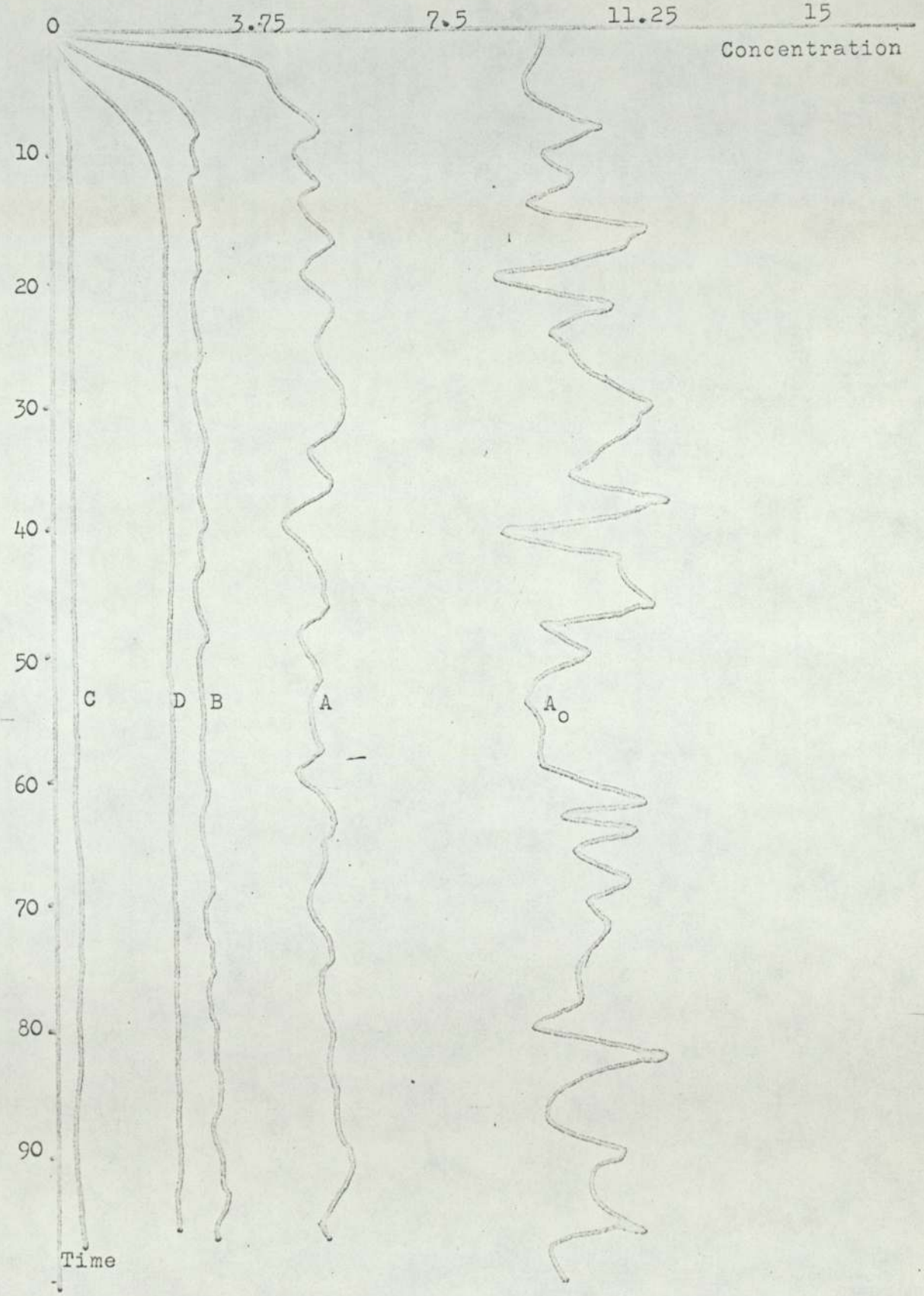


Figure (7).

Results of Simulation (Collocation Programme).

(~)-Simulated : ($z^x y$)-Analytical Solution.

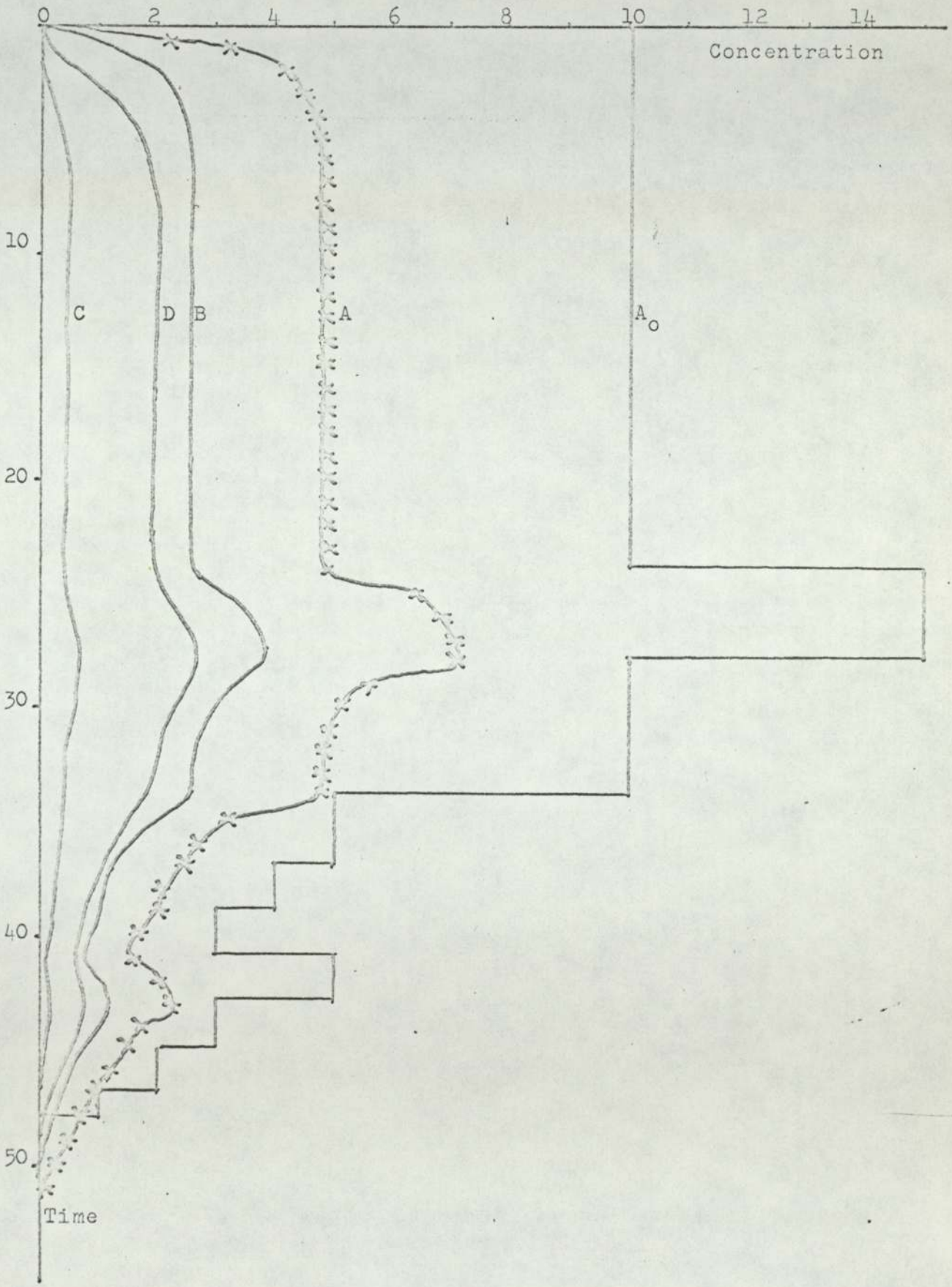
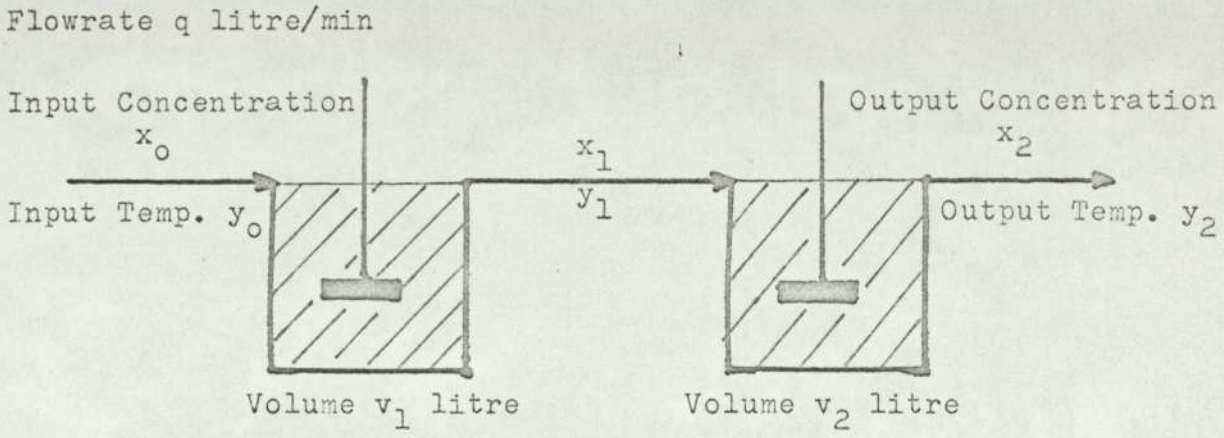


Figure (8).

Double Stage Stirred Tank Reactor With Non-linear Reaction.



Reaction In Tanks.

$$\dot{x} = f_1(x,y,x_o)$$

$$\dot{y} = f_2(x,y,y_o)$$

Figure (9).

Graph of Simulated Concentrations.

(•)-Input (x_o) : (*)-Intermediate (x_1) : (+)-Output (x_2)

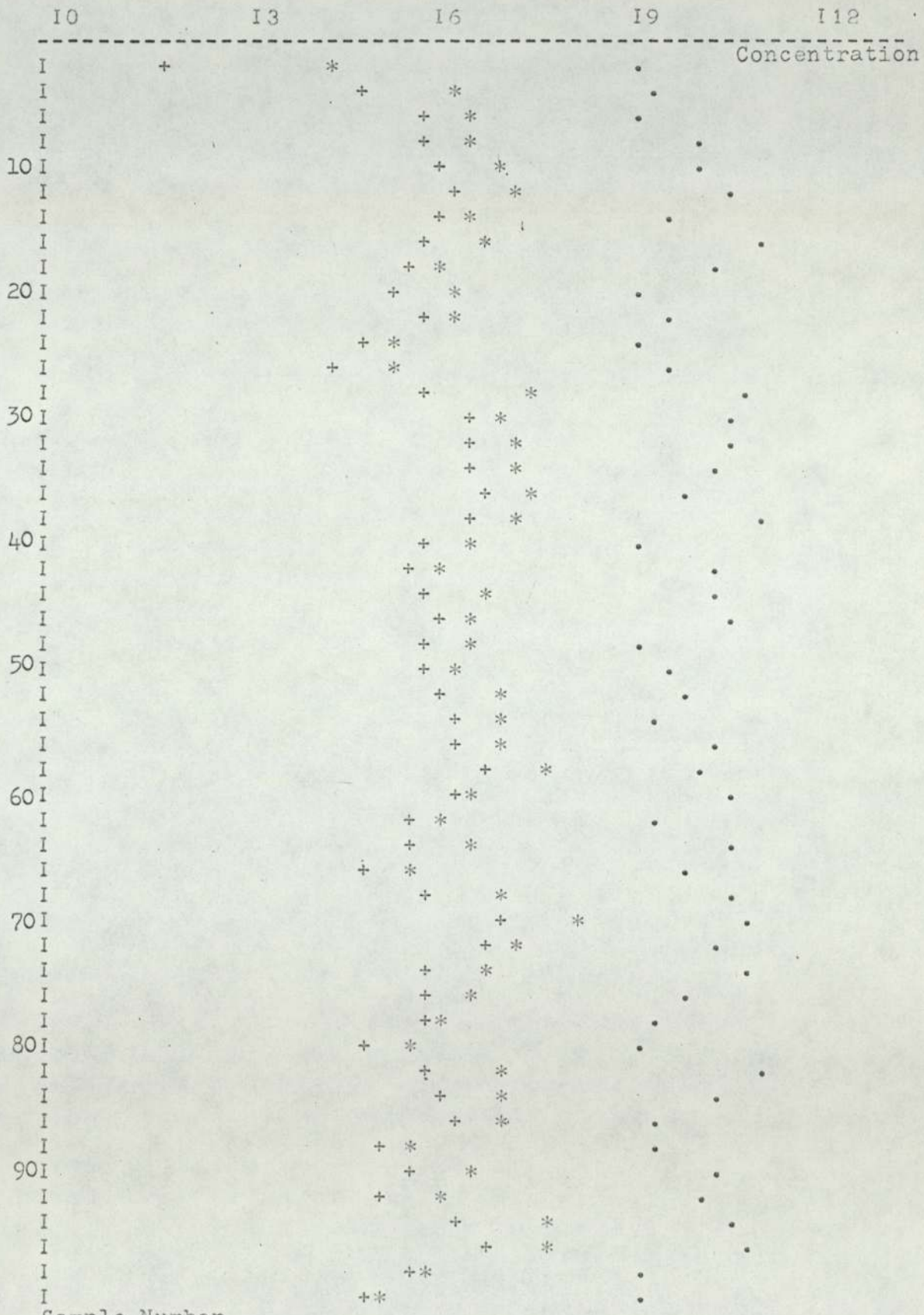


Figure (10).

Graph of Simulated Temperatures.

(.)-Input (y_o) : (*)-Intermediate (y_1) : (+)-Output (y_2)

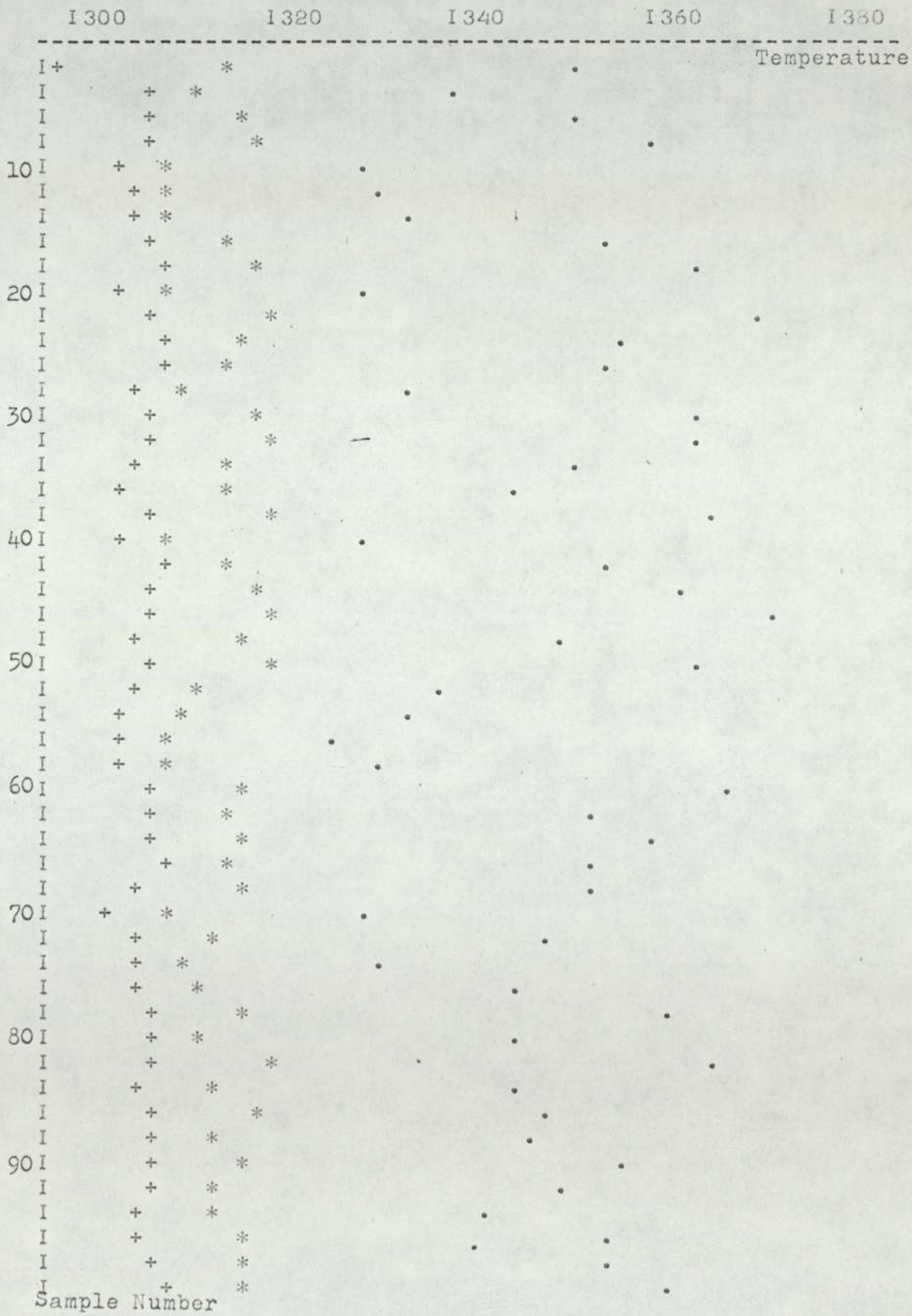


Figure (11)

Graph of Simulated Concentrations.

(•)-Input (x_o) : (*)-Intermediate (x_1) : (+)-Output (x_2)

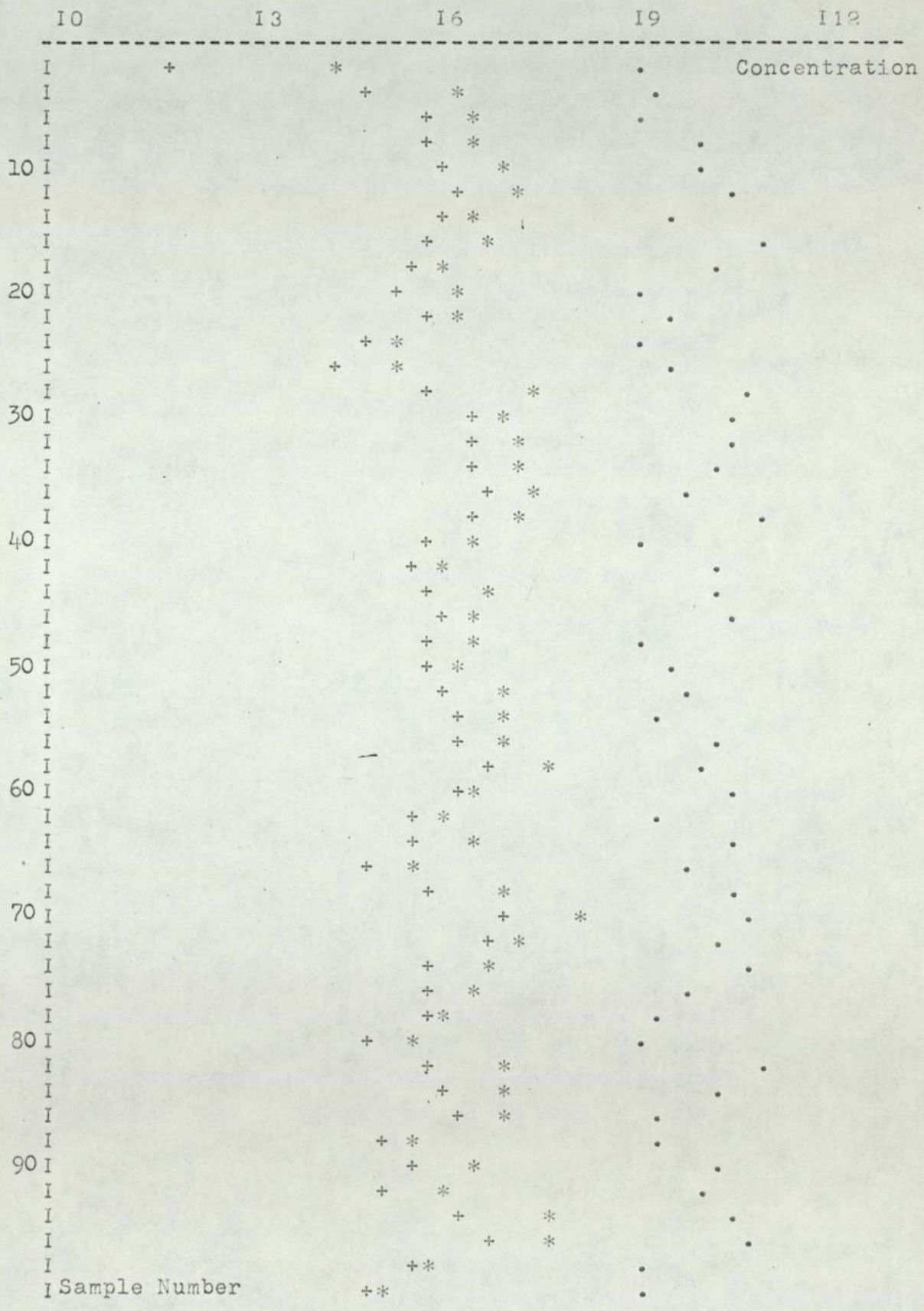
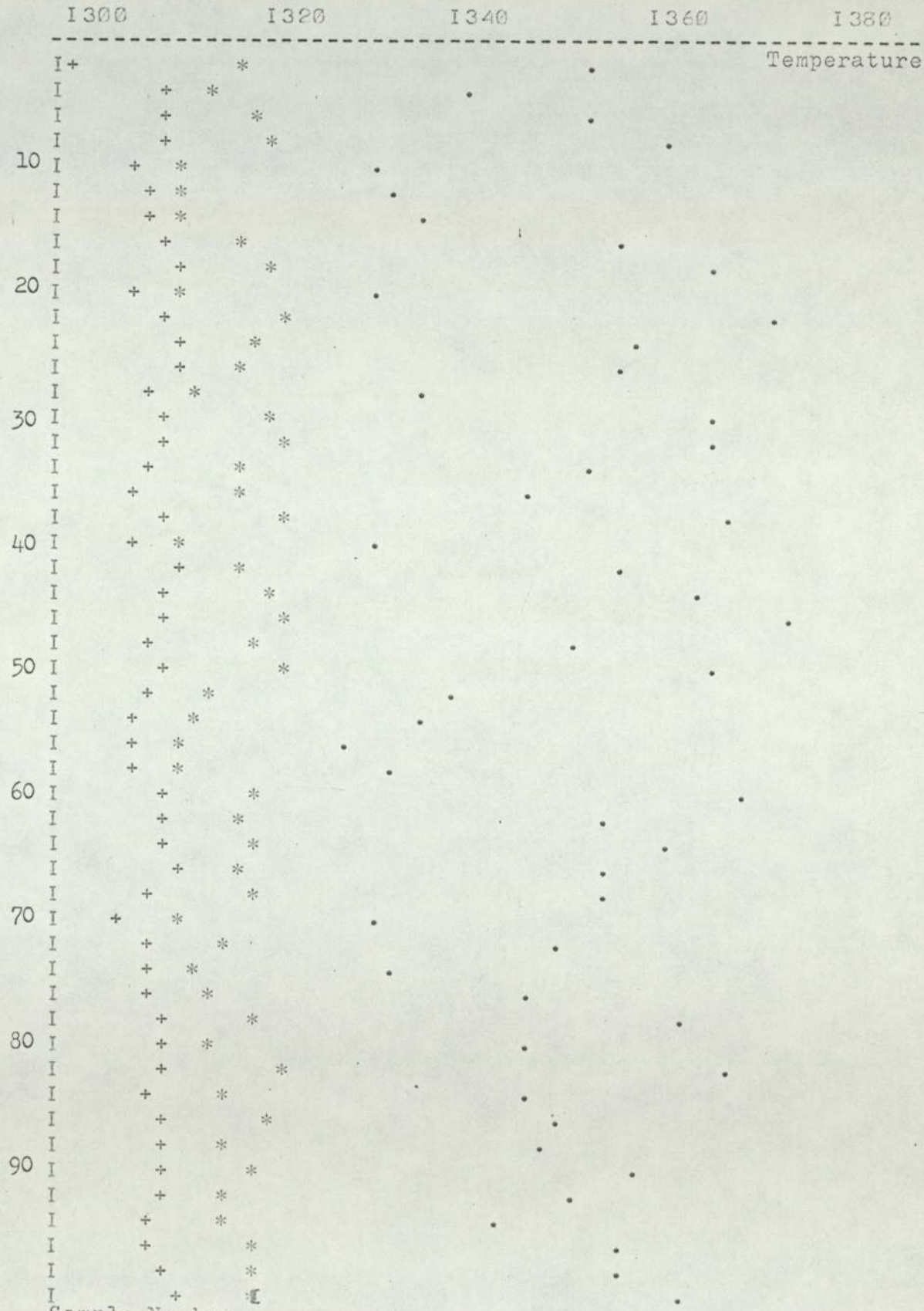


Figure (12).

Graph of Simulated Temperatures.

(•)-Input (y_o) : (*)-intermediate (y_1) : (+)-Output (y_2)



where $EA_o = 10$ and $E(A_o-10)^2 = 1.333$.

The automatic range of solution was, in this situation, suppressed and a constant range of 2 minutes was used. This was because the system was in a constant dynamic state so that no advantage was obtained by allowing the range to vary.

Figures (9, 10, 11, 12) show the results obtained when the system (5.1.2) was disturbed by the random functions x_o and y_o where $Ex_o = 10$ and $E(x_o-10)^2 = 0.333$

and $Ey_o = 350$ and $E(y_o-350)^2 = 133.33$

Comparison of Figures (9, 10) with Figures (11, 12) show that the results obtained by the stationary programme are identical to the results obtained by the collocation programme. This means that the non-linear system (5.1.2) can be closely approximated by a linear stationary model over the range of 2 minutes. The automatic variation of the range of solution was again supressed because of the random inputs.

The above results show that the stationary programme is an effective way of obtaining piece-wise continuous solutions to linear, and non-linear, differential equations. The results indicate however, that the collocation programme would be more suited to solving problems involving complicated specified conditions, as it is unable to compete with the stationary programme in simple initial value problems because of its larger computation time.

5.2 State Estimation.

Figure (13) shows the results obtained by using the filter described (4.3), the system (5.1.1) and the measurement model.

$$z(t_i) = H\,x(t_i)+v \quad \text{with;} \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

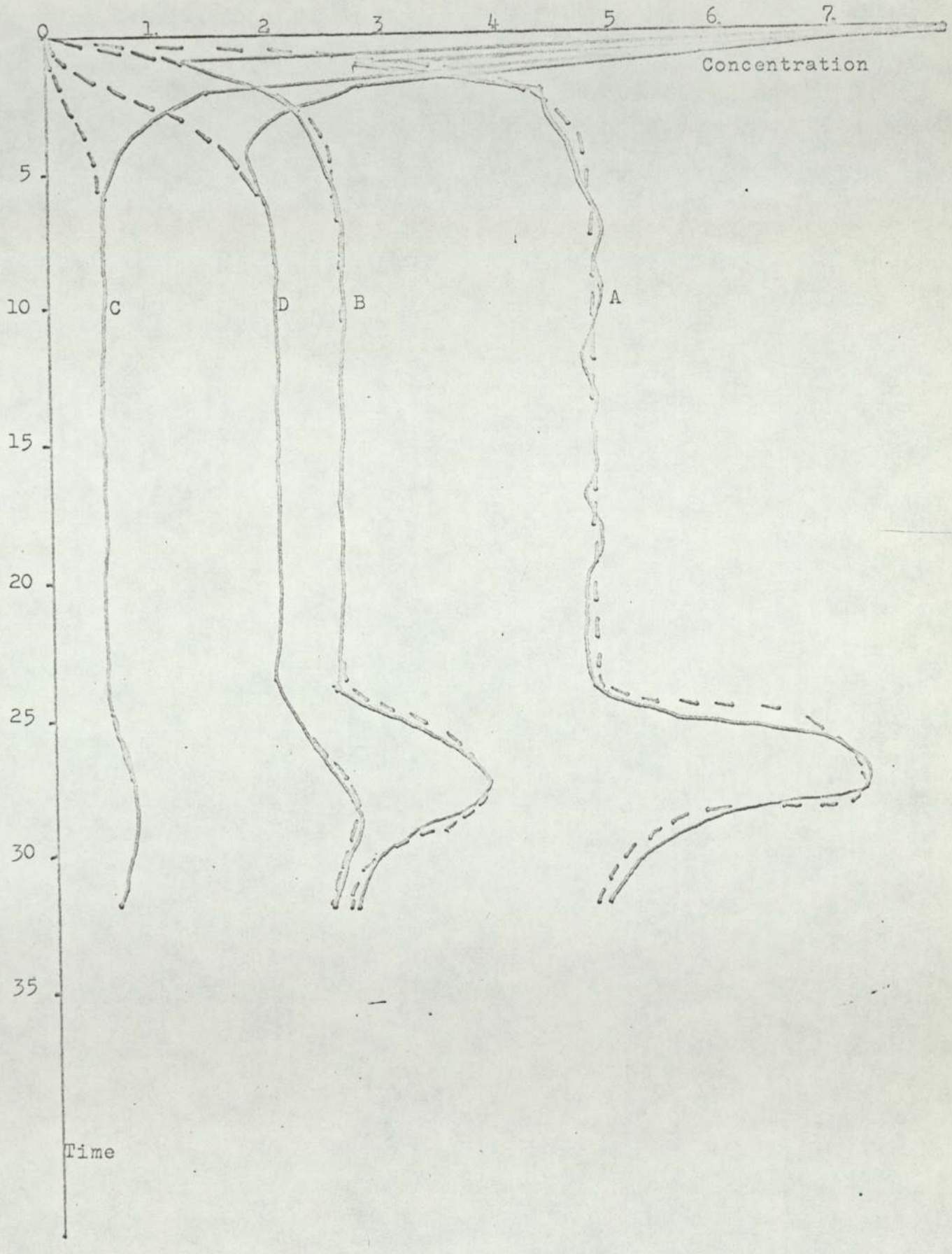
The measurements were obtained from the simulation results (Figure 6) by corrupting them with additive noise v such that,

$$Ev = 0 \quad \text{and} \quad Evv^T = 0.0833\ I$$

Figure (13).

Results Obtained by the Standard Filter.

(~)-Estimates : (- - -)-Simulated.



These measurements are, of course, discrete, whereas the filter requires continuous measurement functions. The choice of measurement function is somewhat arbitrary and its selection is dependent upon what is required from the estimates and what degree of complexity can be handled. For instance, probably the simplest measurement function is the step function defined by,

$$z(t) = z(t_{i+1}) \qquad t_i \leq t \leq t_{i+1}$$

This choice will result in the estimates $\hat{x}(t)$ themselves being functions with discontinuities at the sampling points.

The measurement function used in this work was chosen as the straight line from the previous estimates to the new measurements i.e.

$$z(t) = \hat{b}t + \hat{d} \qquad t_i \leq t \leq t_{i+1}$$

where, $$\hat{b} = (z(t_{i+1}) - H\hat{x}(t_i))/(t_{i+1} - t_i)$$

and, $$\hat{d} = (t_{i+1} H\hat{x}(t_i) - t_i\, z(t_{i+1}))/(t_{i+1} - t_i)$$

Now the linear approximation to the true state can be written,

$$Hx(t) \simeq bt + d$$

where, $$b = (Hx(t_{i+1}) - Hx(t_i))/(t_{i+1} - t_i)$$

and, $$d = (t_{i+1}\, Hx(t_i) - t_i\, Hx(t_{i+1}))/(t_{i+1} - t_i)$$

The measurement error can therefore be approximated as,

$$v(t) \simeq z(t)-Hx(t) = (\hat{b}-b)t+(\hat{d}-d) = \tilde{b}t+\tilde{d}$$

The covariance matrix of the errors in the measurement function

$$R = \frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} Ev(t)v^T(t)\, dt$$

is then approximated by,

$$R \simeq \frac{1}{t_{i+1}-t_i} \int_{t_i}^{t_{i+1}} E(\tilde{b}t+\tilde{d})(\tilde{b}t+\tilde{d})^T dt$$

which, if the range $t_i \leq t \leq t_{i+1}$ is normalised to (-1 to +1), reduces to,

$$R \simeq \frac{1}{3}(H\hat{V}(t_i)H^T + R_m)$$

where R_m is the error covariance matrix of the discrete measurements.

The use of this measurement function results in the estimates being continuous for all time. The differential coefficients of the estimates, however, show simple discontinuities at the sample points. These discontinuities can be removed by using more complicated measurement functions (e.g. quadratic or higher order polynomials).

The value of the disturbance covariance matrix (Q) in this situation represents the covariance matrix of the errors on the

value of A_o and was in this case 0.0833 . Poor initial estimates of the state variables were made and the initial covariance matrix of the estimation errors $\hat{V}(t_o)$ was taken as unity.

5.3 Parameter Estimation.

Figures (14, 15, 16) show the results obtained by using the same system as before (5.1.1) but with A_o and k_3 considered as unknown parameters. An augmented state vector was set up and the equations linearised as described (3.1.3).

Estimates were obtained for the state variables and for the unknown parameters using the same measurements as before (Figure 6). The initial estimates were taken to be zero for each component and the initial covariance matrix of the estimation errors was again taken as unity.

The value of the disturbance covariance matrix (Q), this time, represents the covariance of the unknown value of A_o and as such is a parameter that affects the performance of the filter. If the value of Q is too high the estimates are 'noisy ', if it is too low the estimates are unable to respond to any change in the input. Many experiments with different Q's would be necessary to decide which Q to use, but even then little could be decided without reference to the true value of the states. The value of Q used to obtain the results shown in Figures (14, 15, 16) was 0.0833, and the fact that the results are in fairly good agreement with the true values is the result of the use of information that would not normally be available in a real situation. These results therefore, demonstrate clearly the problems encountered while trying to estimate states and parameters in the presence of uncertainties in the statistical parameters of the system.

Figure (14).

Parameter Estimation Using the Standard Filter.

(~)-Estimates : (- - -)-Simulated

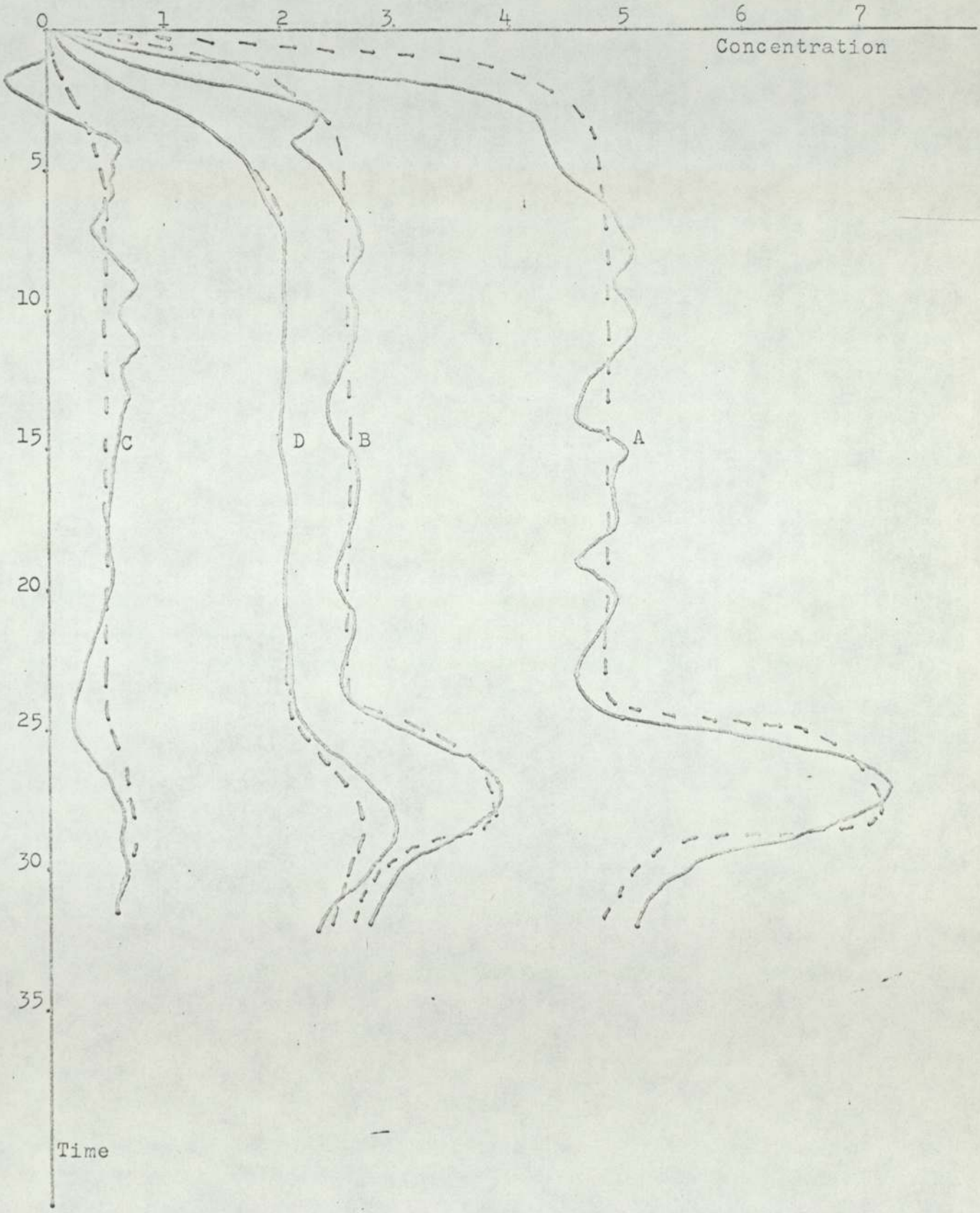


Figure (15).

Estimation of the Input Concentration. (A_o).

(~)-Estimates : (- - -)-Simulated

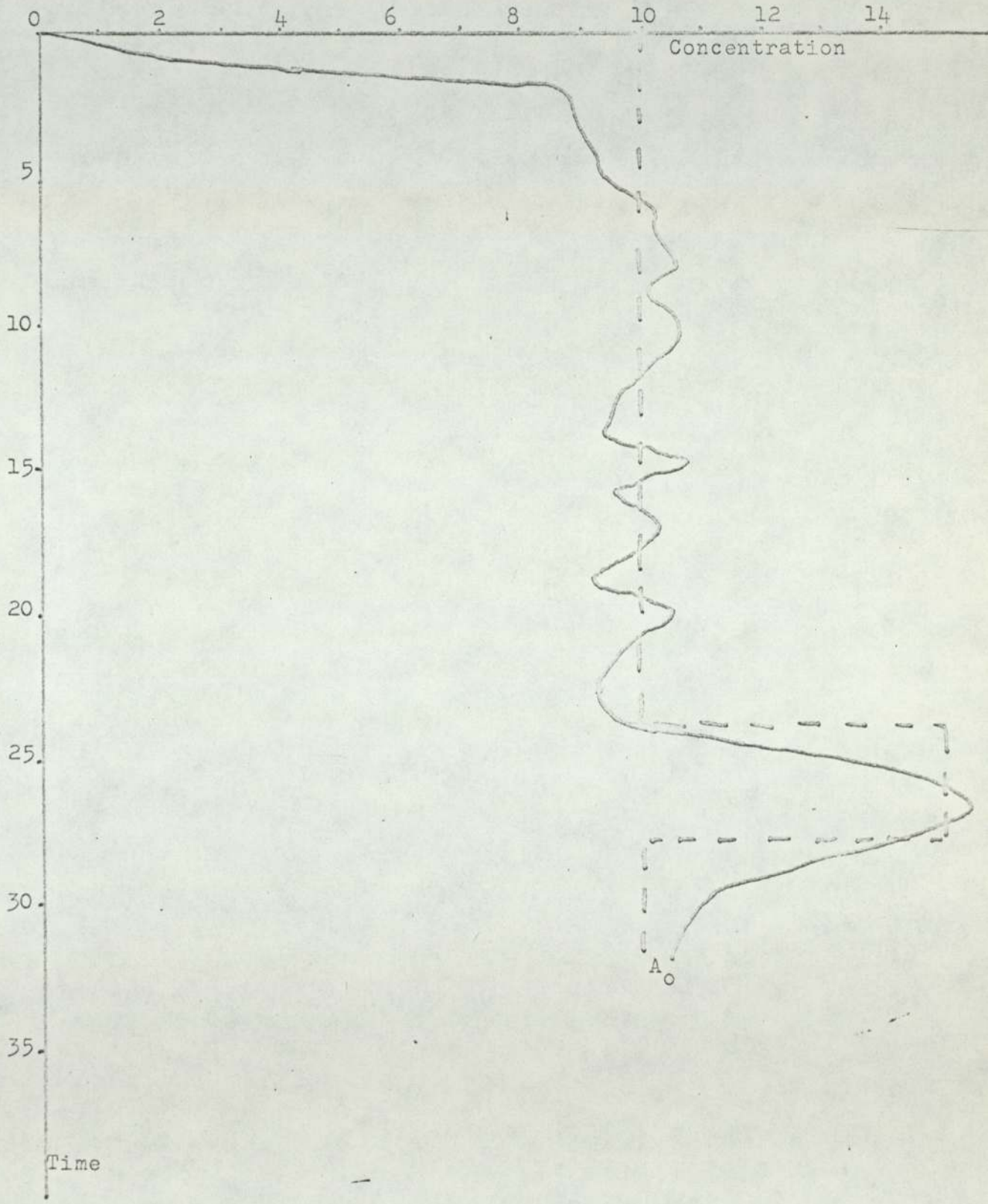
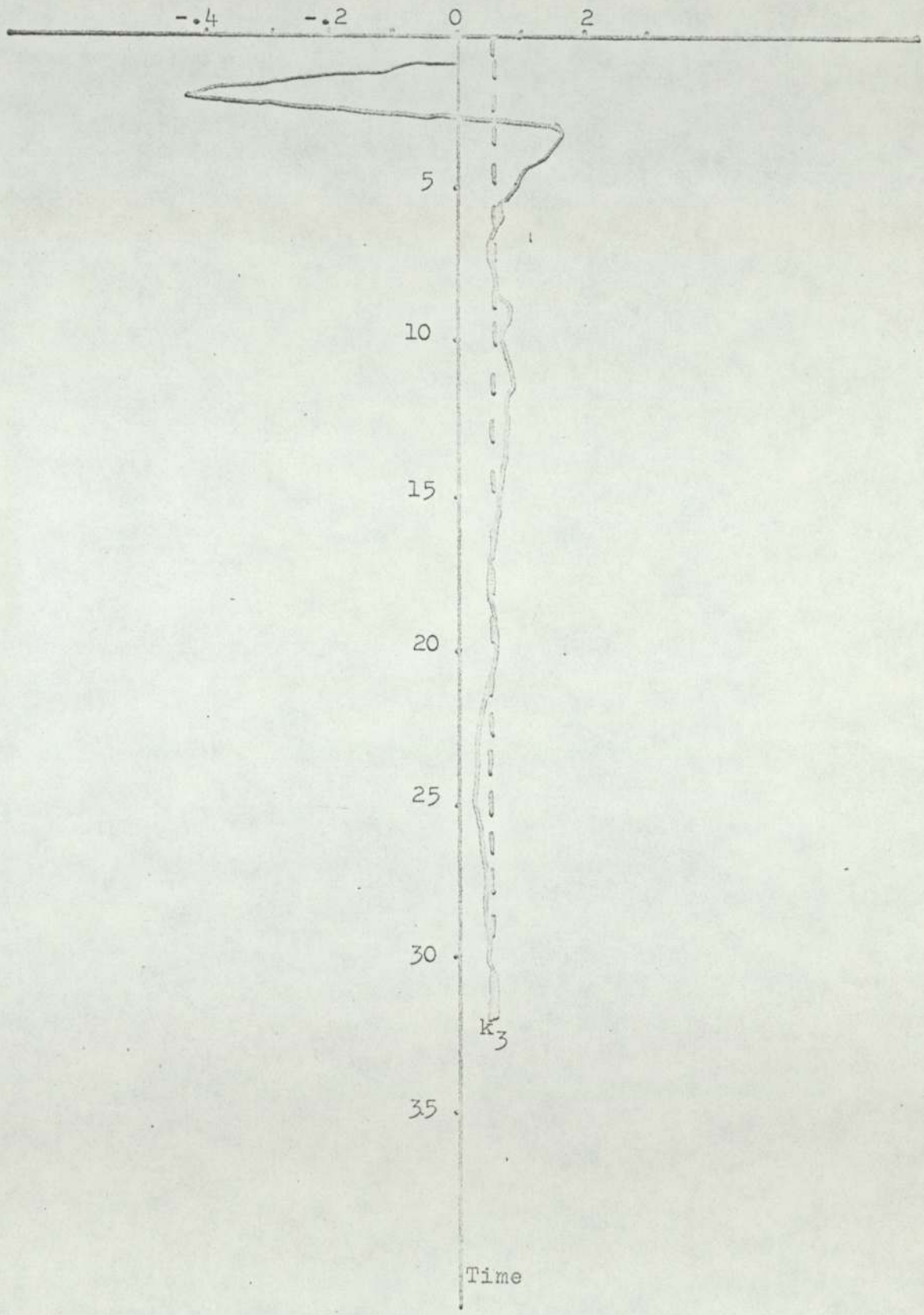


Figure (16).

Estimation of Rate Constant (k_3).

(~)-Estimate : (~)-Simulated



5.4 Adaptive Filtering.

At the end of chapter (3) it was noted that the parameters α, β, and θ could not be evaluated from theoretical considerations. To investigate the effects of varying these parameters, some initial guesses had to be made. Initially α was set = 1/i where i is the number of samples taken, and β was set = 0.25 . The effect of different values of θ (i.e. the number of time increments that must elapse before each implementation of equations (43, 44)), could now be investigated.

5.4.1 Experiment With A Poor Model. (The Effect of Varying θ)

The poor model used was:

$$\dot{A} = -(k_1 + 0.308)A + 0.385B + 0.308A_o$$

$$\dot{B} = k_1 A - B$$

$$\dot{A}_o = 0$$

$$\dot{k}_1 = 0$$

which is the same as system (5.1.1) but with the components C and D ignored. The measurements were taken from the simulation of the true system Figure (6a) and were linked to the state variables by,

$$z = Hx + v$$

with,
$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

and, $Ev = 0$ and $Evv^T = 0.0833\ I$

The value of Q was taken to be 0.01 with,

$$F_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

i.e. a disturbance with zero mean and variance of 0.01 on the value of A_o was expected. This was not the true value of the

input disturbance, which was unknown, and so represents a further error with this model.

The matrix F_4, which is the compensation incidence matrix, was restricted to be one of:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

because, by definition, A_o and k_1 are parameters and are therefore modelled correctly. To find which of these matrices is the best representation of F_4, each one was used in the filter for ten sample increments and the effect on the TRACE of the residual covariance matrix (g) recorded. The matrix that minimised (g) was then selected as F_4.

Figures (17, 18, 19, 20, 21, 22, 23) show the results obtained by filtering this poor model with θ=2, 4, and 6. The results obtained with θ=4 and 6 were quite close together and in good agreement with the true values. The matrix found for F_4 with θ=4 and 6 was:

$$F_4 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

and Figure (21) shows the compensation function (w). F_4 and w together form the optimal compensation strategy for this system and in fact show that the model is in error because it does not allow for B to leave the system except by changing to A, whereas in the true system B can also change to C and to D. Therefore w

Figure (17).

Graph of Output Concentration (A) (Estimates).

(*)-θ=2 : (+)-θ=4 : (•)-θ=6

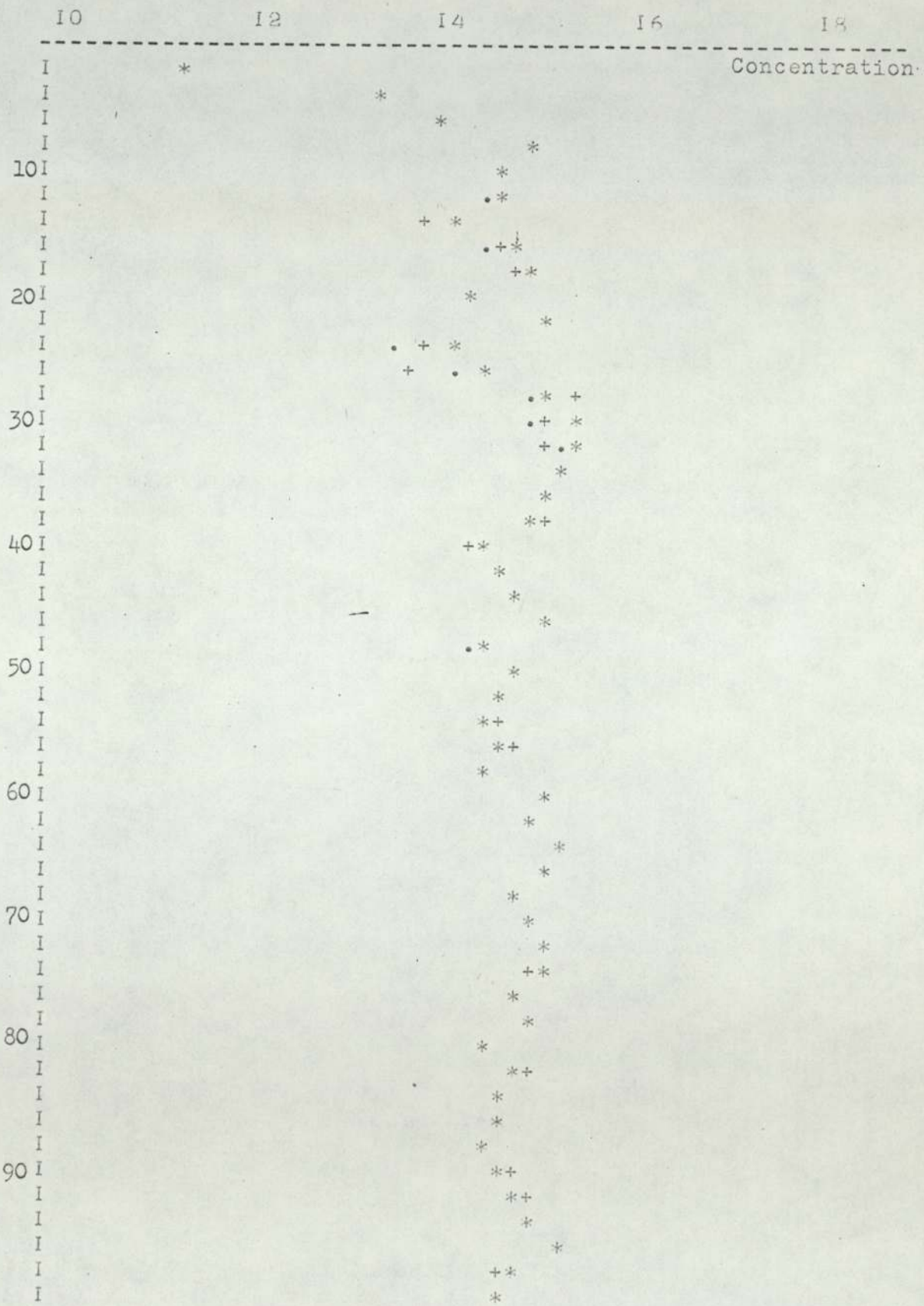


Figure (18).

Graph of Output Concentration (B) (Estimates).

(*)-θ=2 : (+)-θ=4 : (•)-θ=6

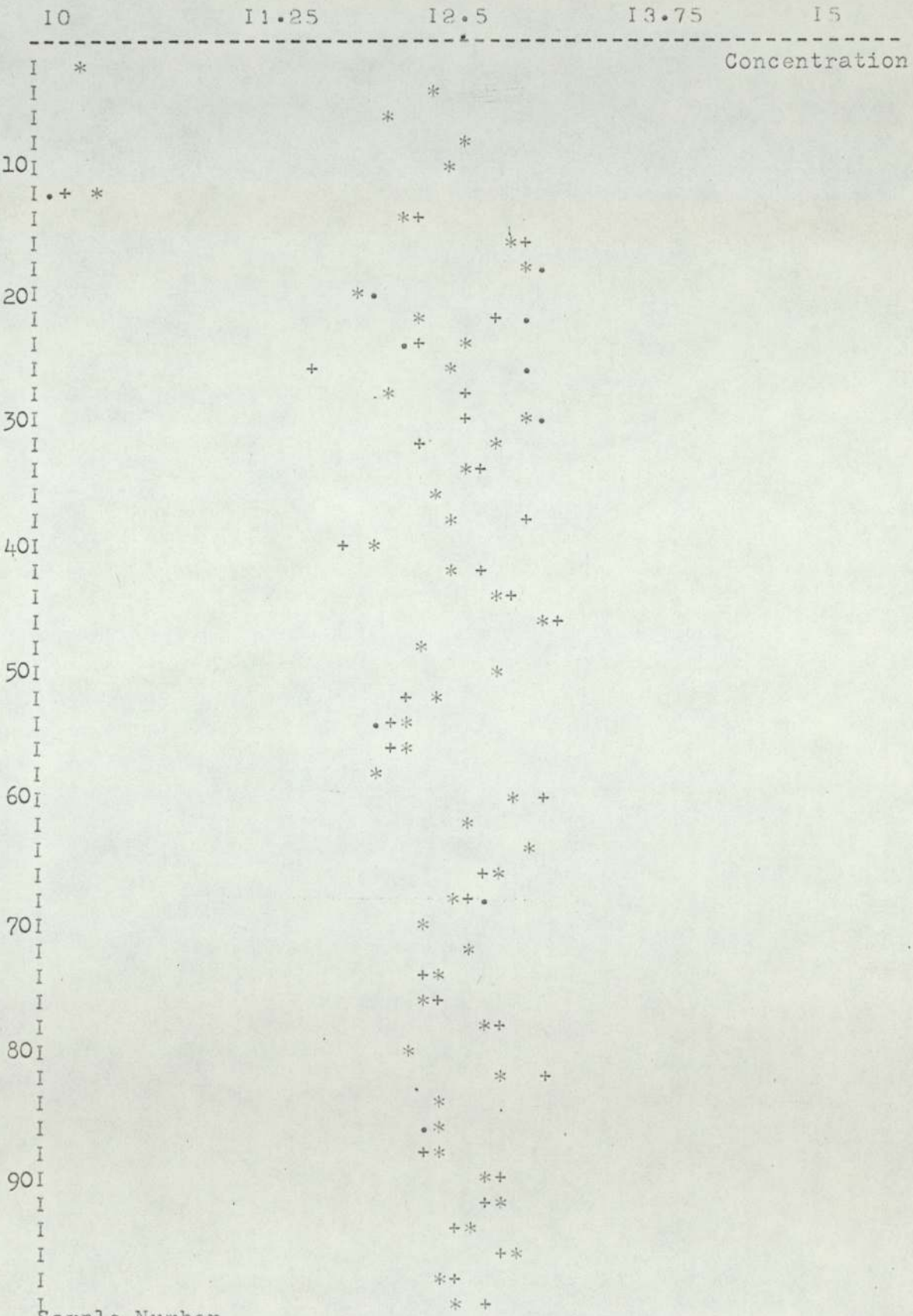


Figure (19).

Graph of Input Concentration (A_o) (Estimates).

(*)-θ=2 : (+)-θ=4 : (•)-θ=6

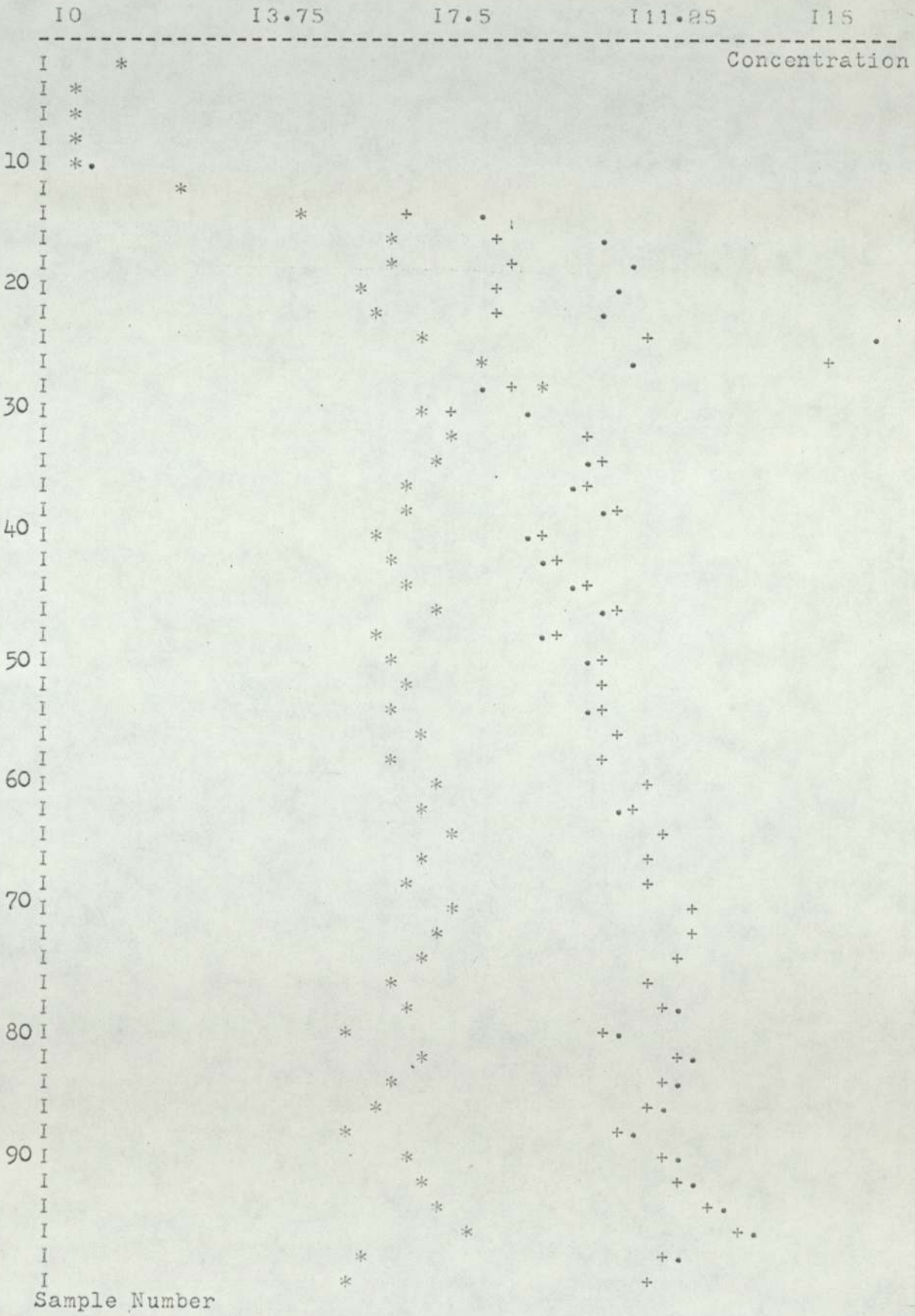


Figure (20).

Graph of Rate Constant (k_1) (Estimates).

(*)-θ=2 : (+)-θ=4 : (°)-θ=6

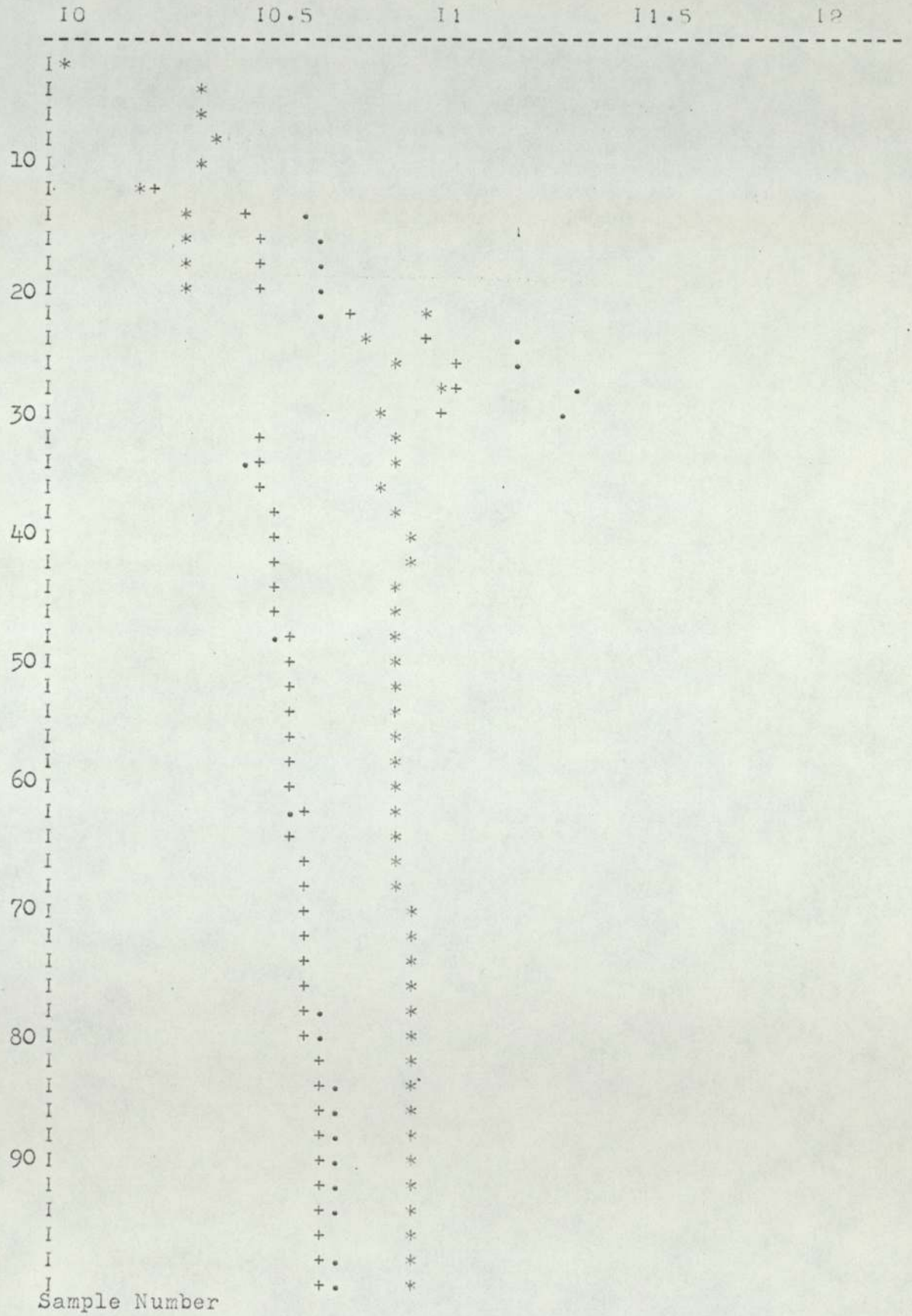


Figure (21).

Graph of Fictitious Input (w_1).

(*)-θ=2 : (+)-θ=4 : (°)-θ=6

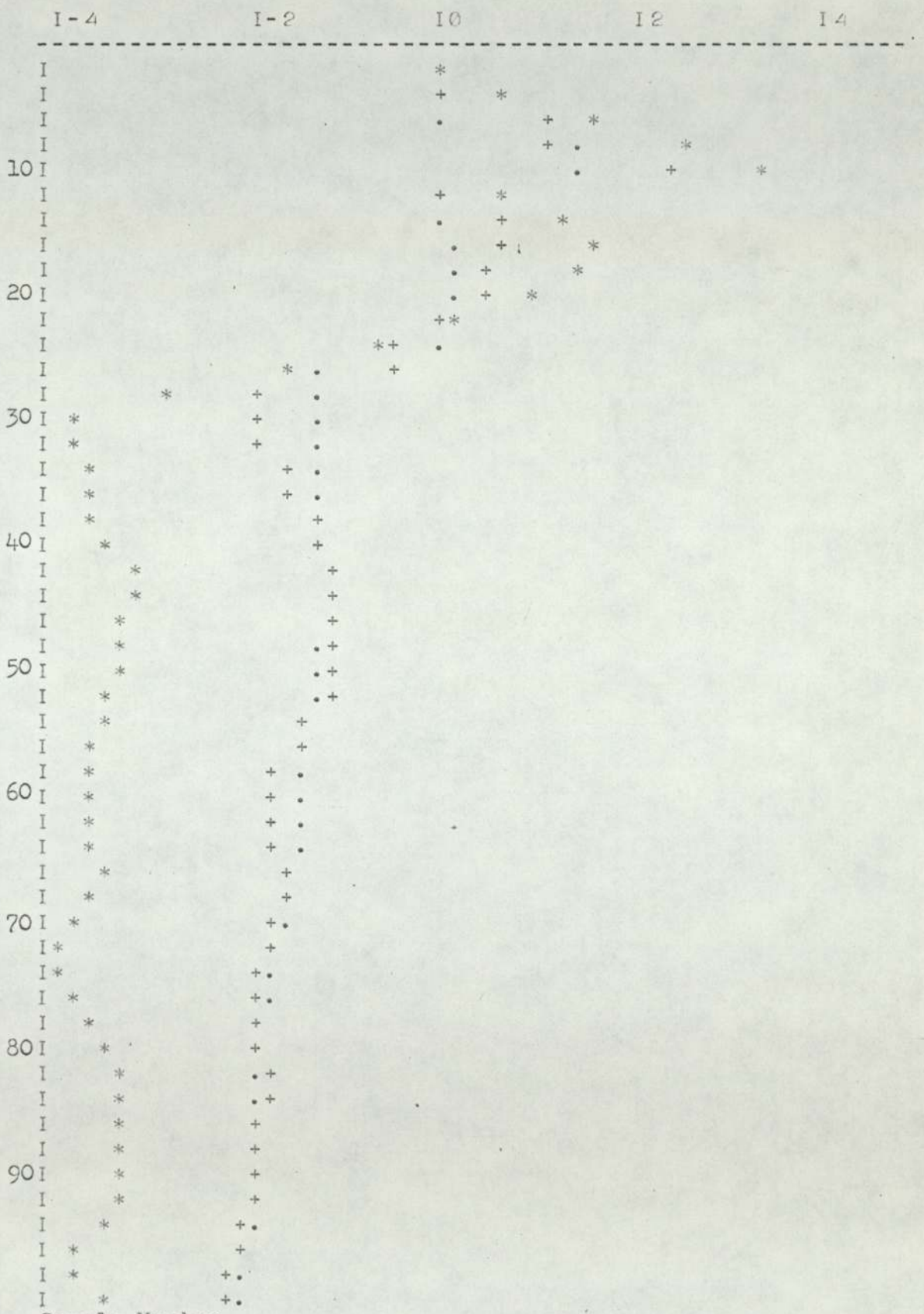


Figure (22).

Graph of Fictitious Input (w_2).

(*)-θ=2 : (+)-θ=4 : (•)-θ=6

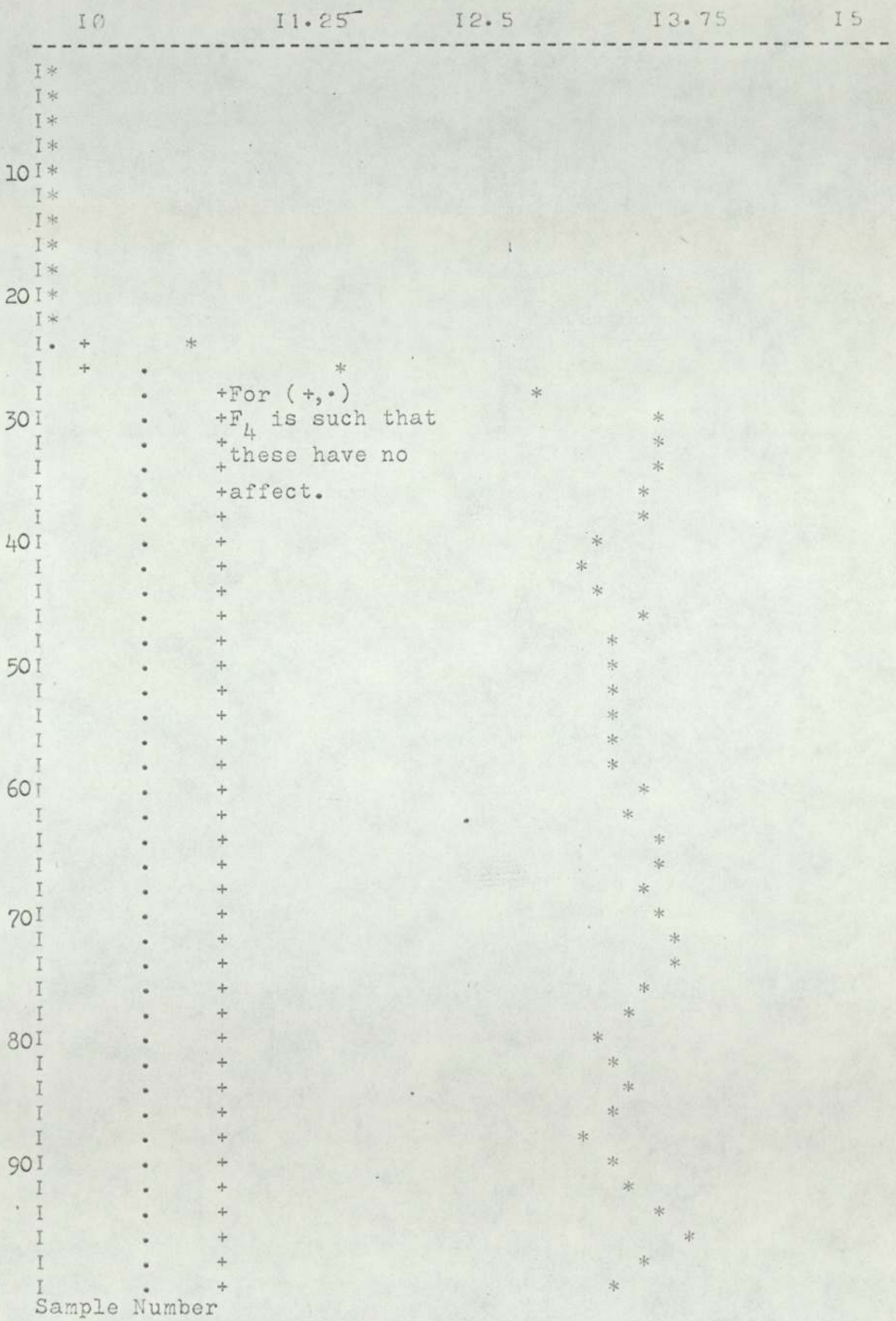
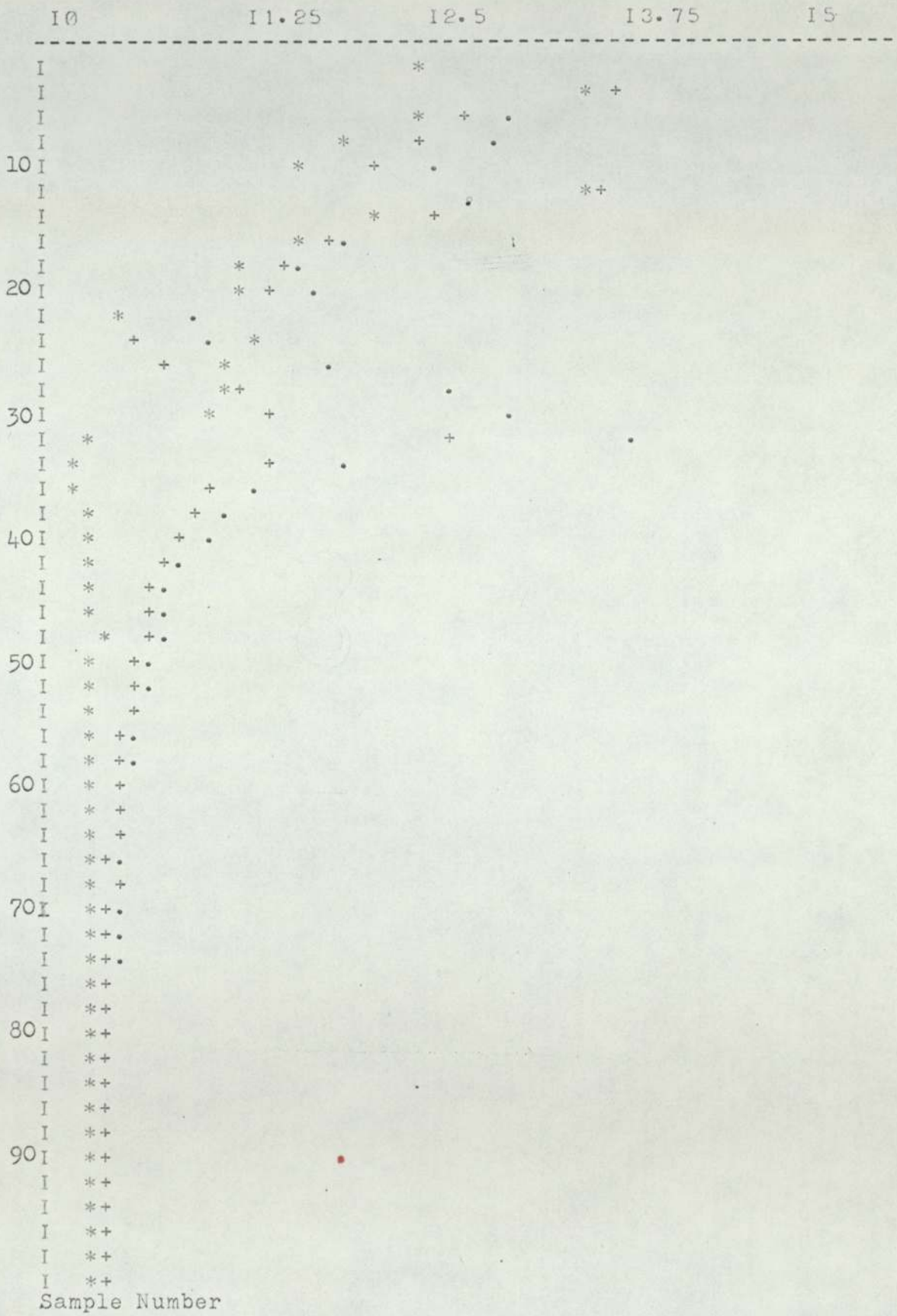


Figure (23).

Graph of The TRACE of The Residual Covatiance Matrix (g).

(*)-θ=2 : (+)-θ=4 : (•)-θ=6



provides a rate of loss of B which matches the rate of B changing to C and D in the true system.

The results obtained with $\theta=2$, however, were very different even though the TRACE of the residual covariance matrix (g) Figure (23) would seem to indicate that $\theta=2$ gave the best performance. The explanation for this is quite obvious, and indeed these results could have been intuitively expected. The compensation term defined by equations (43, 44) should represent errors resulting from the poor model i.e. errors that remain after the filter has converged. If these equations are used too frequently, then the filter is not allowed to converge and compensation is conditioned on random errors resulting from poor starting conditions or particularly poor measurements. This situation is aggravated further in this case by the fact that the fully augmented system equations are unobservable as, F_4 was found to be,

$$F_4 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$ (appendix 13).

5.4.2 Experiment with a Poor Model.(The Effect of Varying β).

Figures (24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34) show the results obtained by filtering the same poor model with $\theta=4$ and with $\beta=0.2$, 0.25, 0.3, 0.4, 0.5, 0.6. This means that compensation for bias is performed every 4^{th} measurement and is based on a value for the bias calculated from equation (40).

The results obtained with $\beta=0.2$, 0.25, 0.3 and 0.4 were all similar with $\beta=0.3$ giving the best performance (g minimised) Figure (29).

Figure (24).

Graph of Output Concentration (A) (Estimates).

(*)-β=0.2 : (+)-β=0.25 : (·)-β=0.3

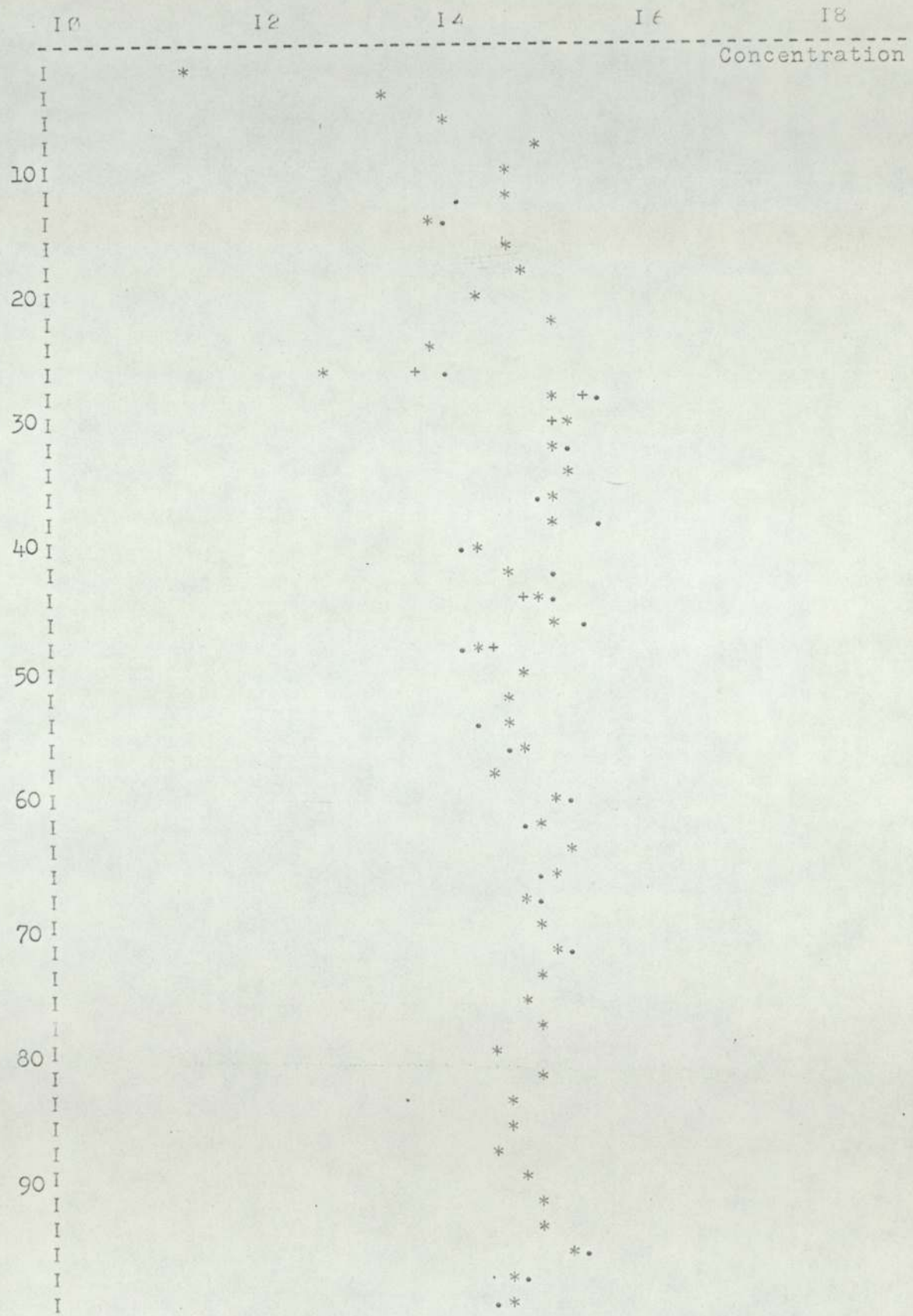


Figure (25).

Graph of Output Concentration (B) (Estimates).

(*)-β=0.2 : (+)-β=0.25 : (°)-β=0.3

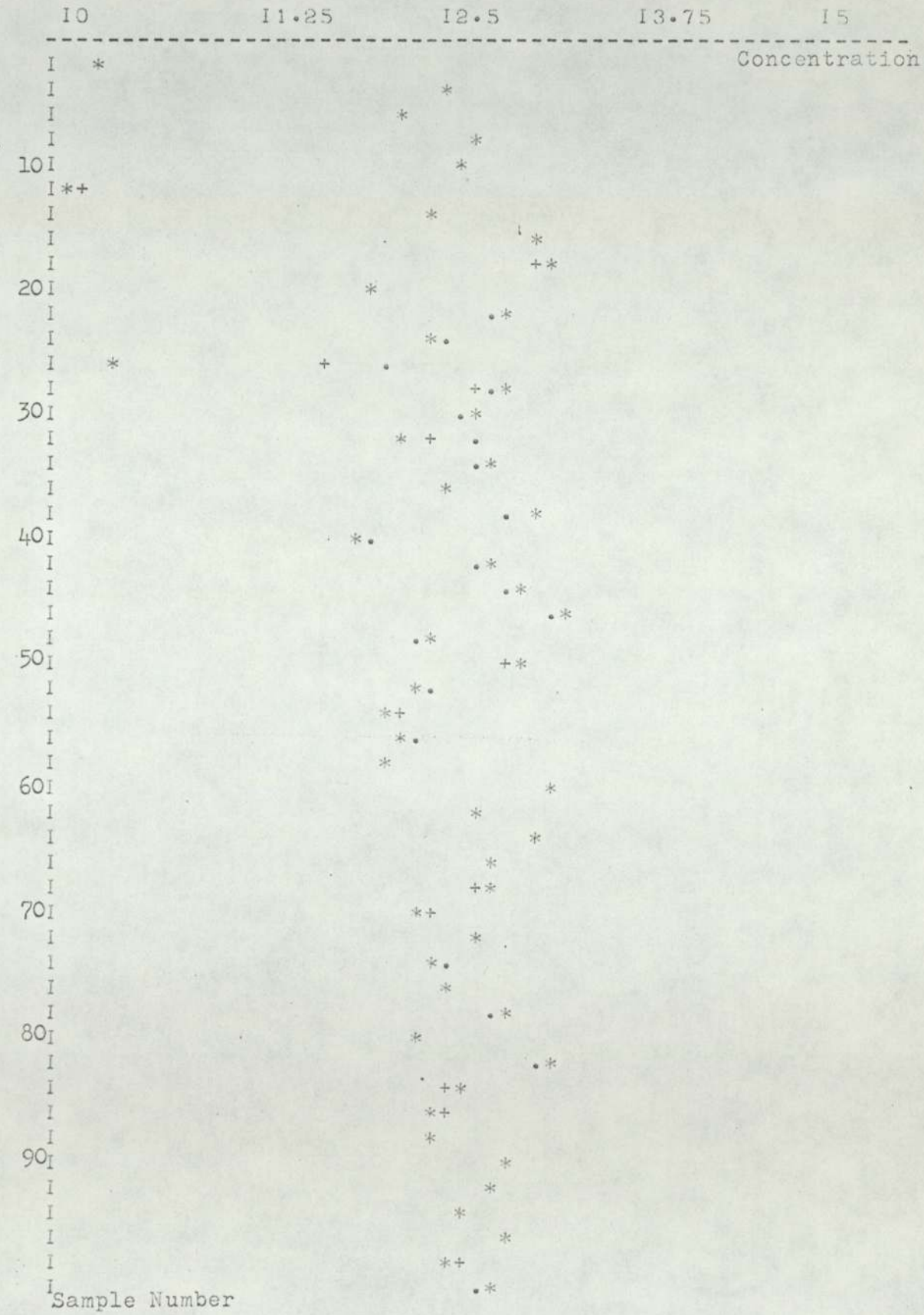


Figure (26).

Graph of Input Concentration (A_o) (Estimates).

(*)-β=0.2 : (+)-β=0.25 : (.)-β=0.3

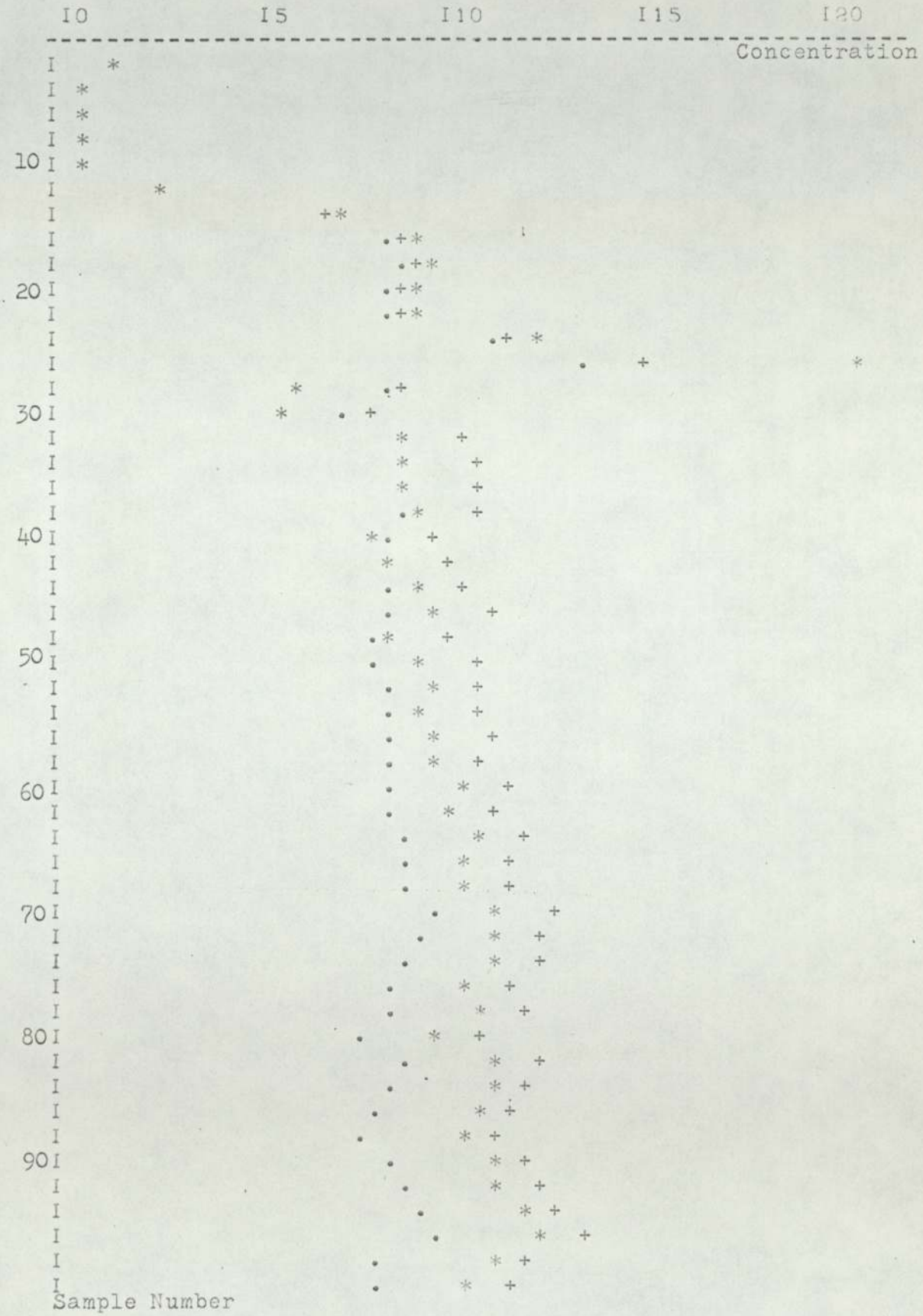


Figure (27).

Graph of Rate Constant (k_1) (Estimates).

(*)-β=0.2 : (+)-β=0.25 : (.)-β=0.3

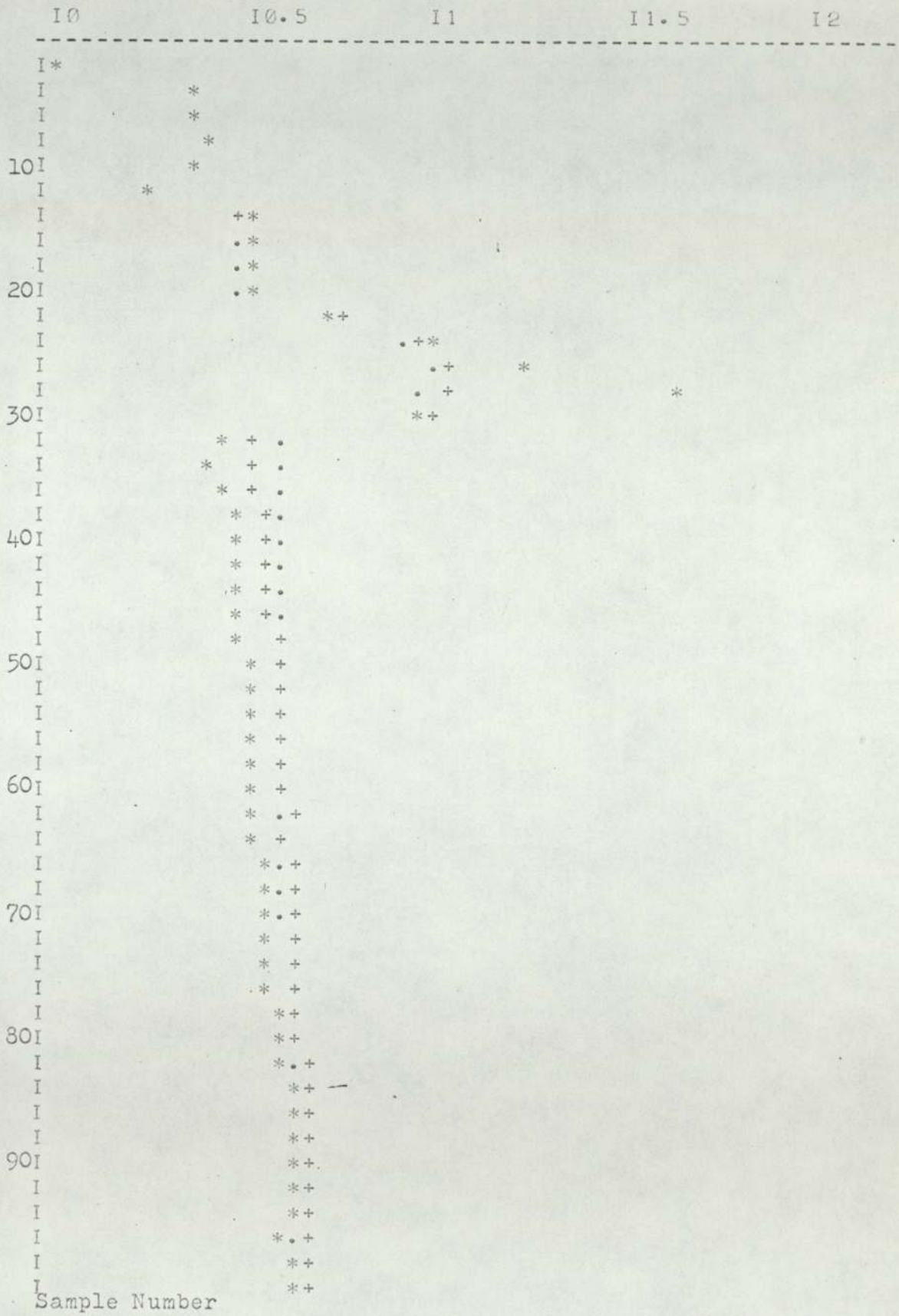


Figure (28).

Graph of Fictitious Input (w).

(*)-β=0.2 : (+)-β=0.25 : (.)-β=0.3

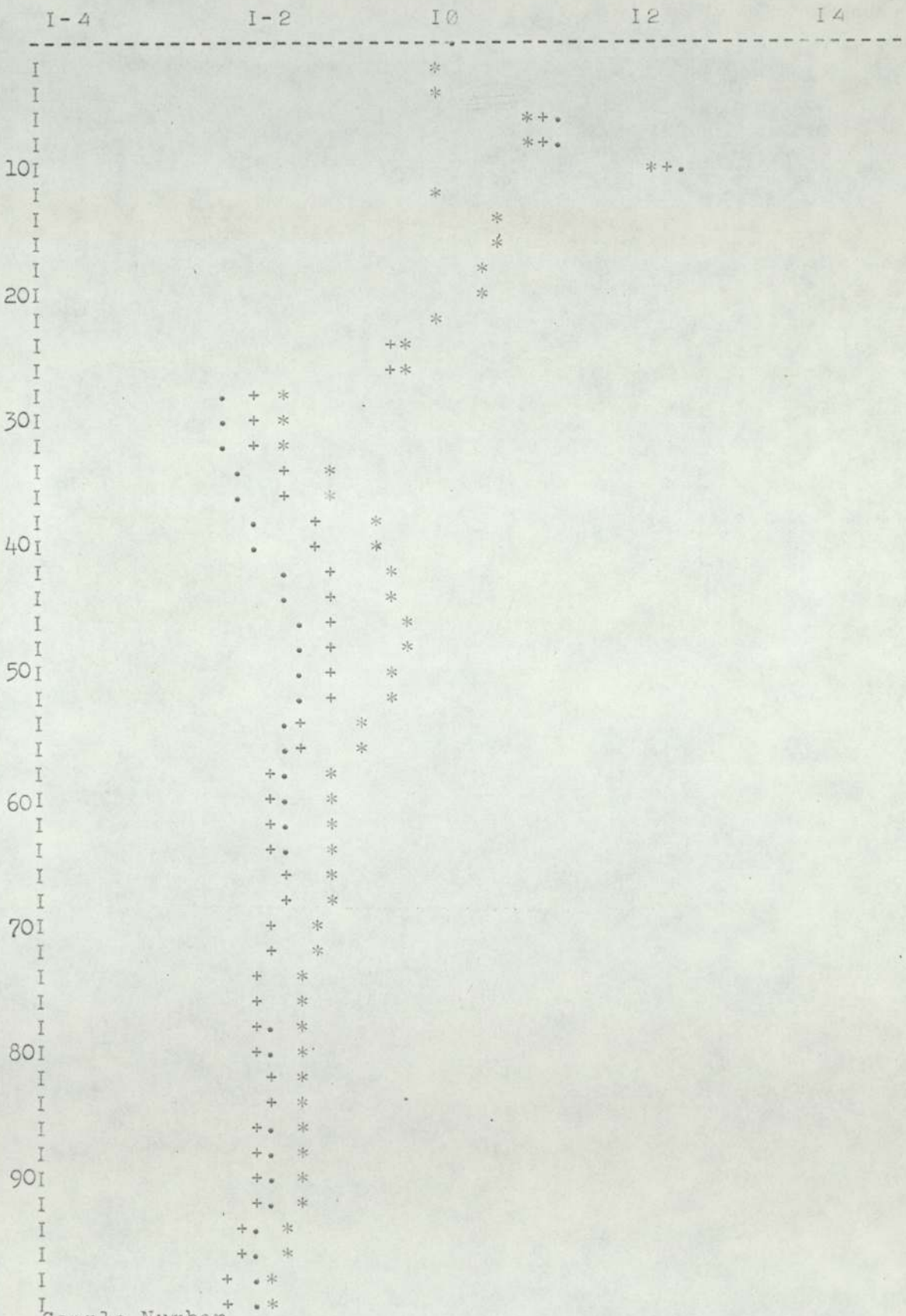


Figure (29).

Graph of The TRACE of The Residual Covariance Matrix ($\underline{\varepsilon}$).

(*)-β=0.2 : (+)-β=0.25 : (.)-β=0.3

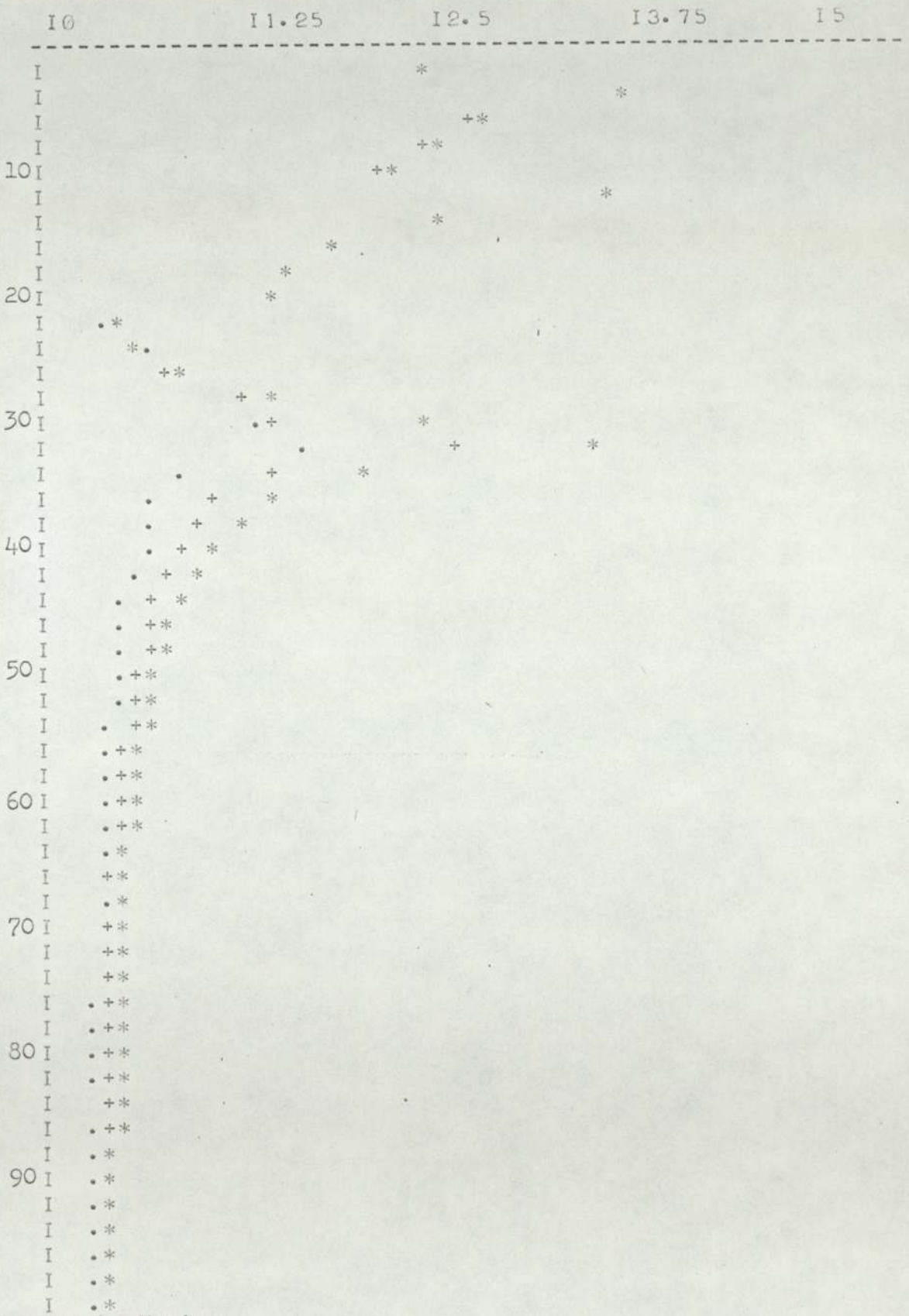


Figure (30).

Graph of Output Concentration (A) (Estimates).

(*)-β=0.4 : (+)-β=0.5 : (.)-β=0.6

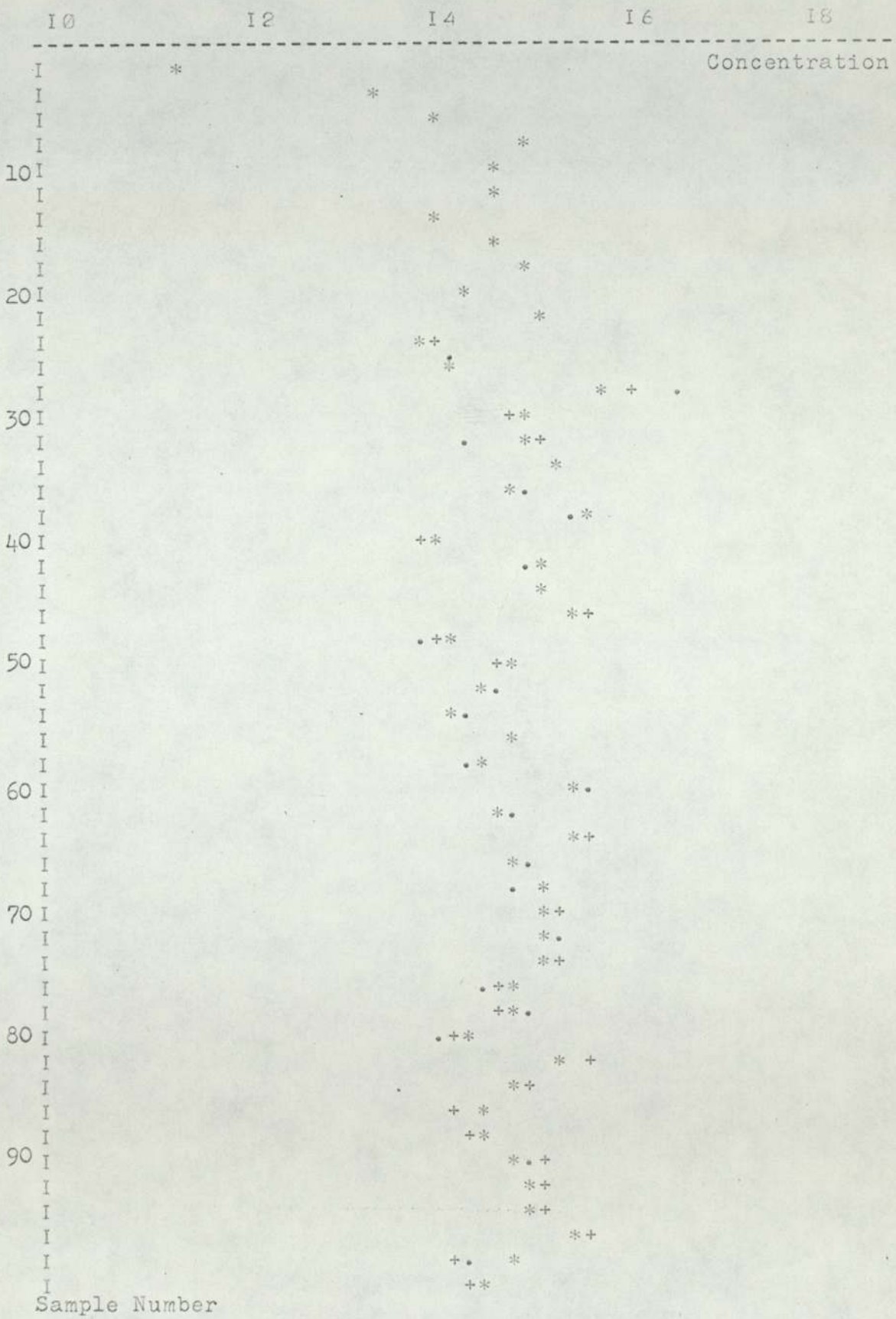


Figure (31).

Graph of Output Concentration (B) (Estimates).

(*)-β=0.4 : (+)-β=0.5 : (.)-β=0.6

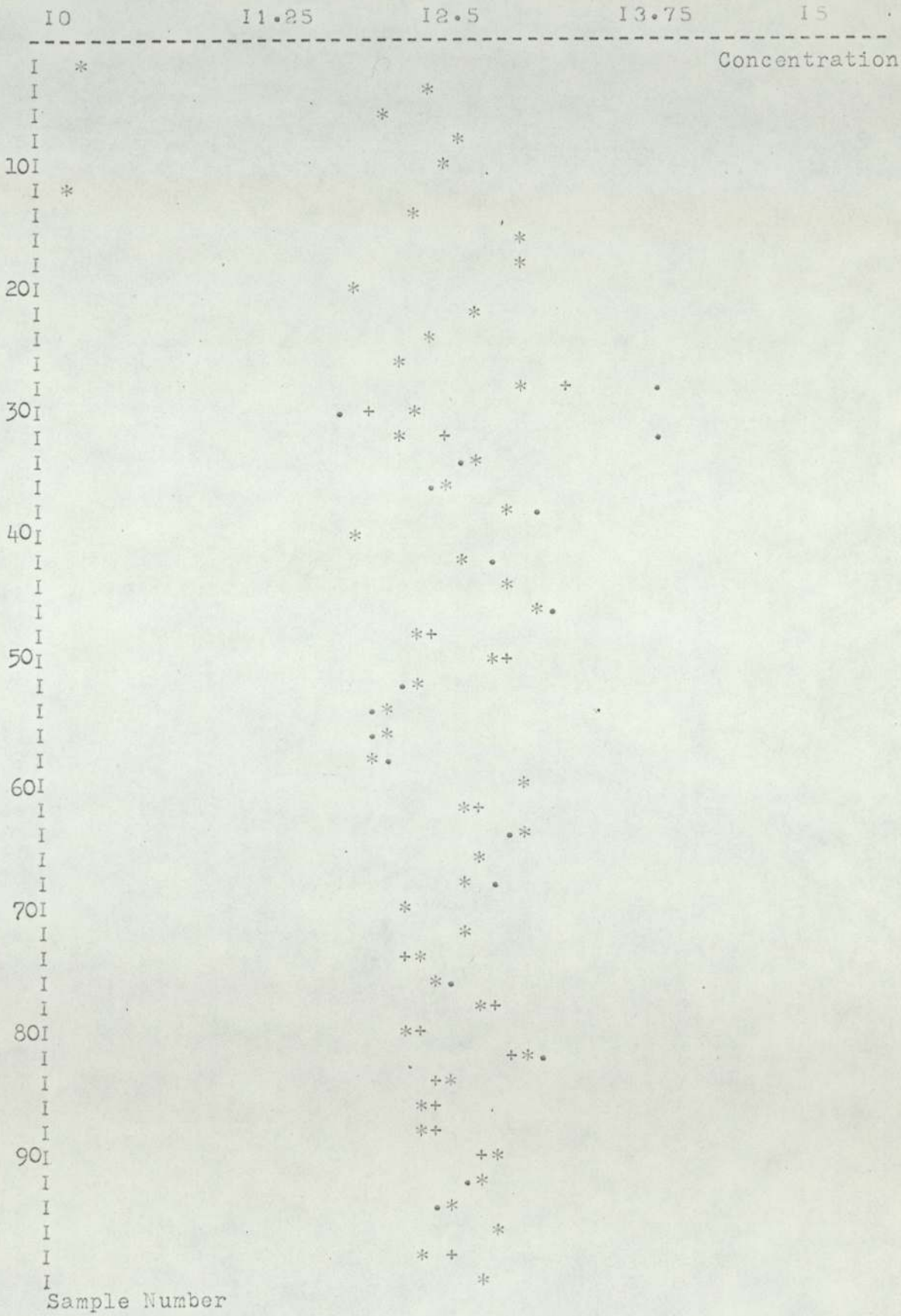


Figure (32).

Graph of Input Concentration (A_o) (Estimates).

(*)-β=0.4 : (+)-β=0.5 : (.)-β=0.6

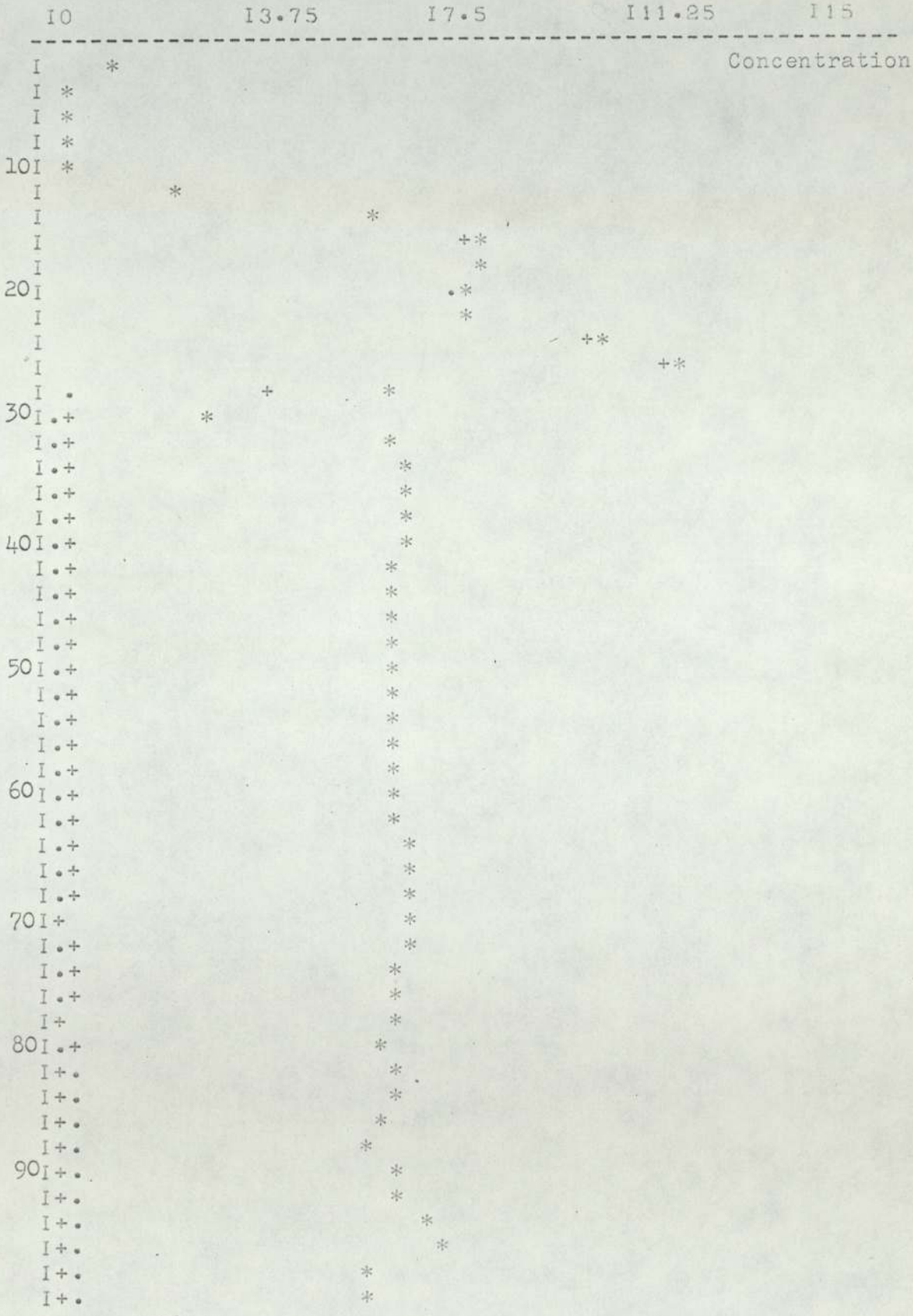


Figure (33).

Graph of Rate Constant (k_1) (Estimates).

(*)-β=0.4 : (+)-β=0.5 : (.)-β=0.6

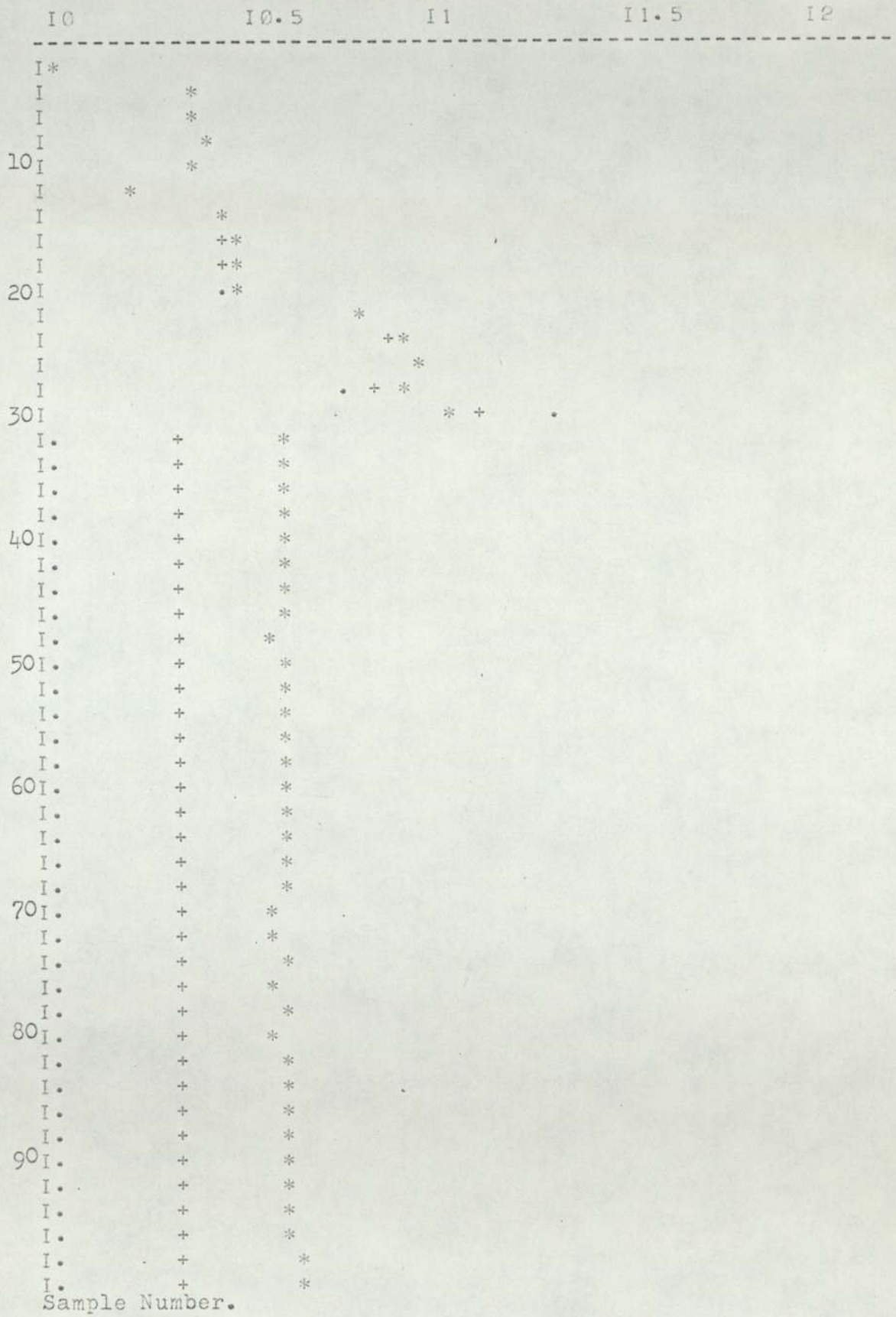
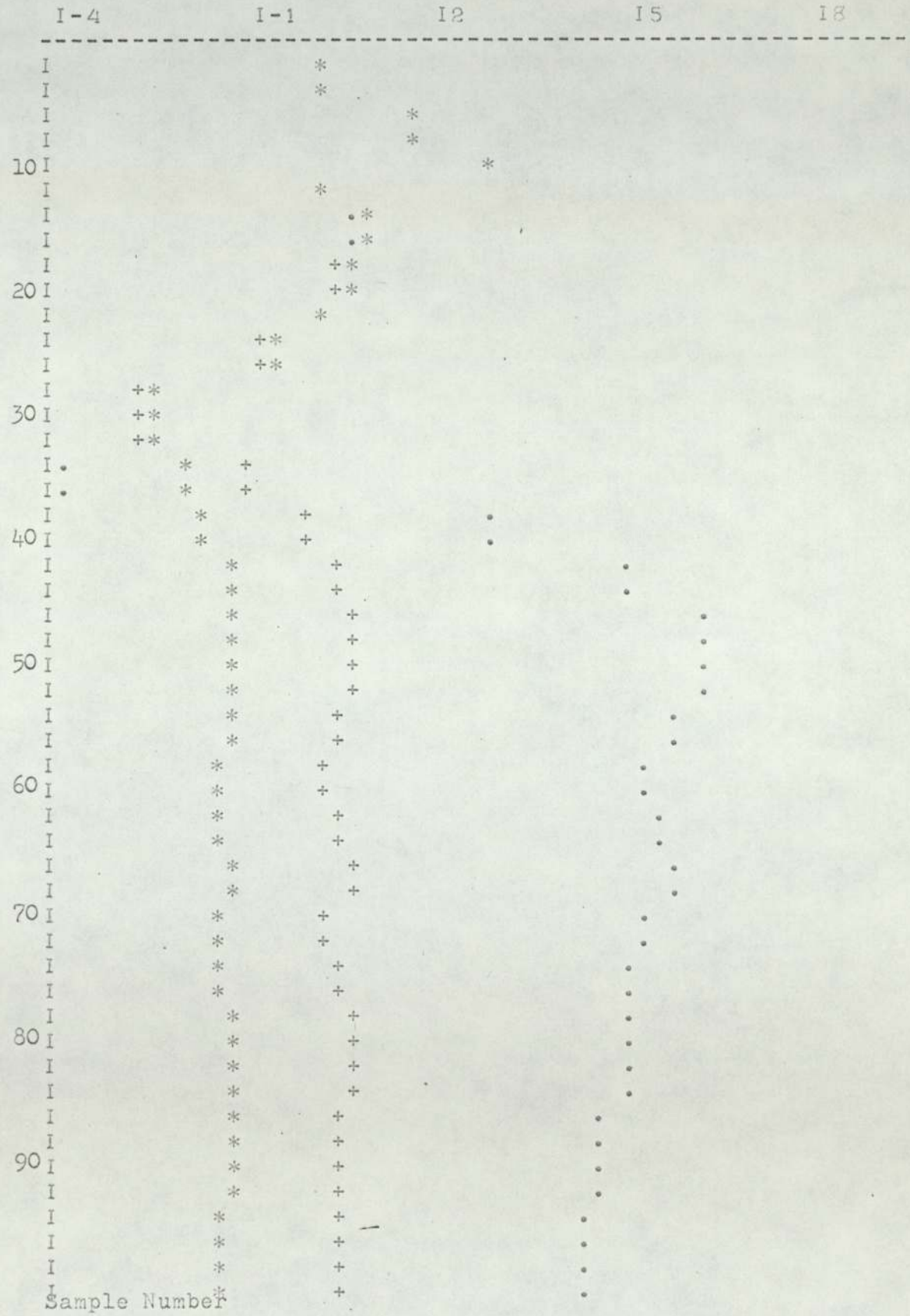


Figure (34).

Graph of Fictitious Input (w_1).

(*)-β=0.4 : (+)-β=0.5 : (•)-β=0.6



The matrix F_4 was found to be,

$$F_4 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

which matches the actual errors in the model.

The results obtained with β=0.5 and 0.6, however, were again very different, and again for similar reasons. In these cases F_4 was found to be,

$$F_4 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

With β=0.5 or 0.6 in equation (40) too much weight is given to the most recent measurement so that γ ceases to be a reliable estimate of the mean of the residuals.

The reason β=0.3 is better than β=0.2 is related to the fact that θ=4. As β decreases, less weight is given to recent measurements in equation (40). There comes a point where so little weight is given to new data that the affect of compensating is not noticed in θ time increments, when the next compensation calculation is performed. Again γ ceases to be a reliable estimate of the mean of the residuals.

The results discussed so far show that there is a lowest value for θ, below which the compensation equations fail, and also that there is a range of values of β outside which problems again arise. The values of these parameters that are indicated by the preceding results are θ=4 and β=0.3.

Figures (35, 36, 37, 38) show the results of the adaptive filter using θ=4 and β=0.3, the results of the standard filter,

Figure (35).

Graph of Output Concentration (A).

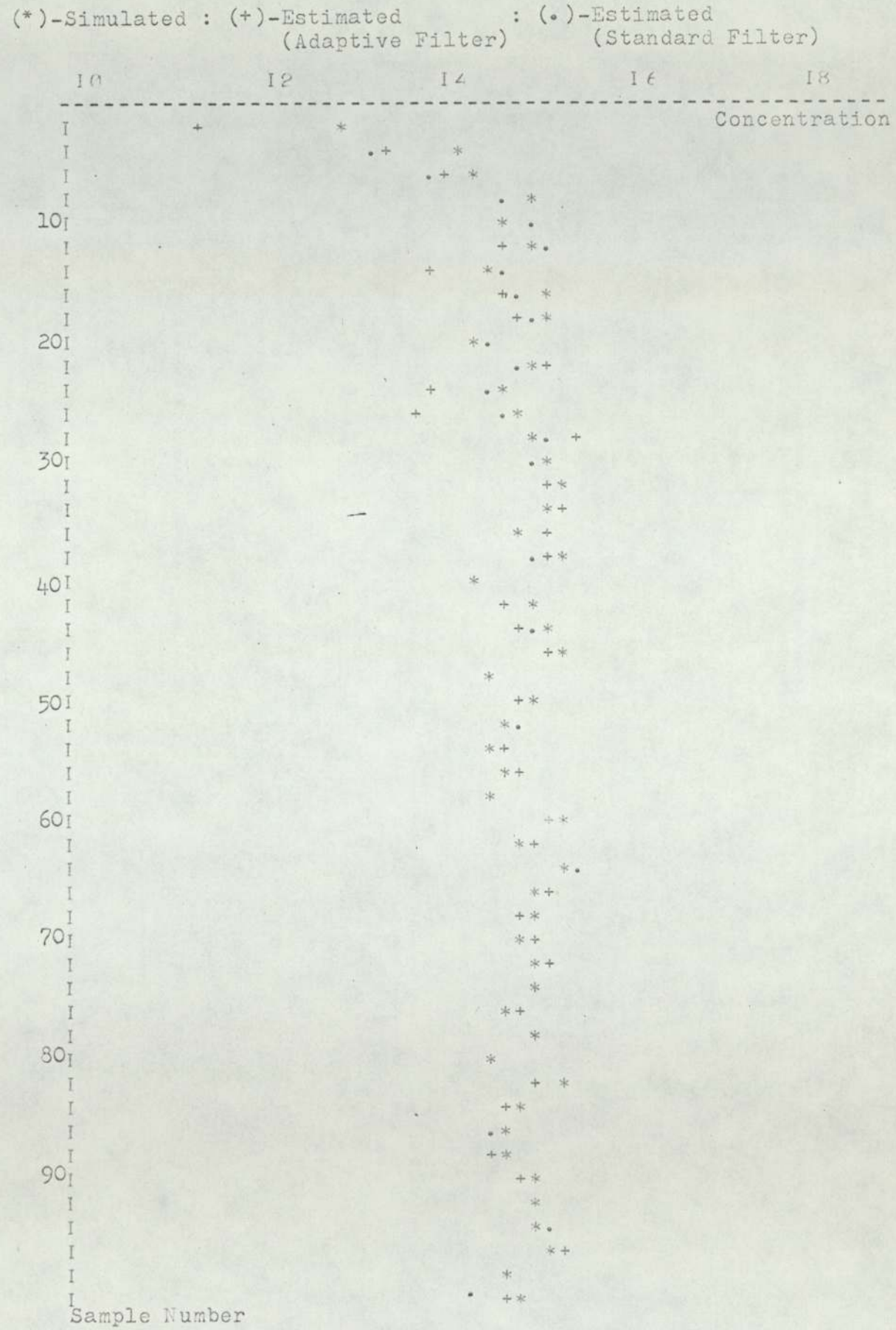


Figure (36).

Graph of Output Concentration (B).

(*)-Simulated : (+)-Estimated (Adaptive Filter) : (•)-Estimated (Standard Filter).

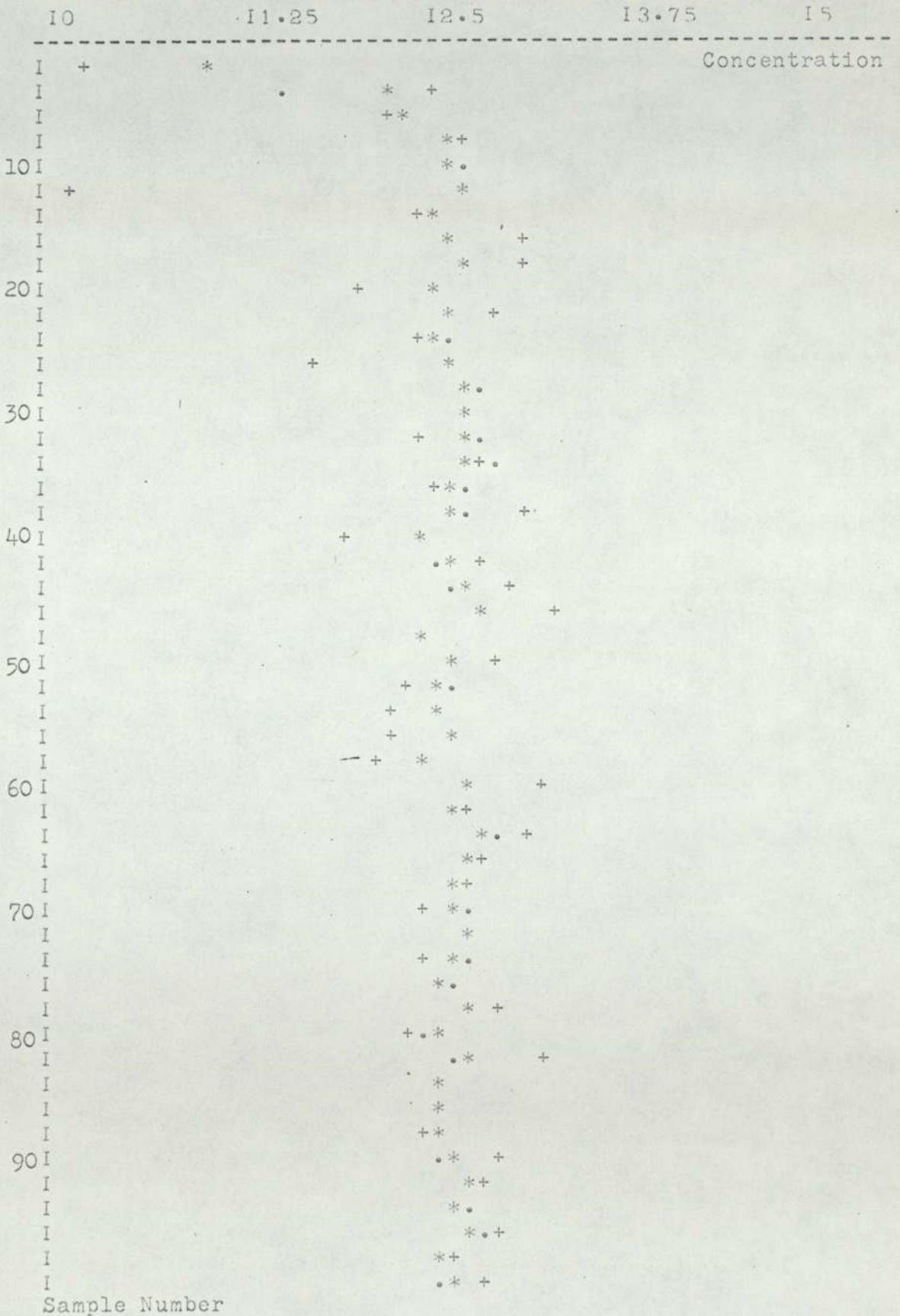


Figure (37).

Graph of Output Concentration (A_o).

(*)-Simulated : (+)-Estimated (Adaptive Filter) : (.)-Estimated (Standard Filter)

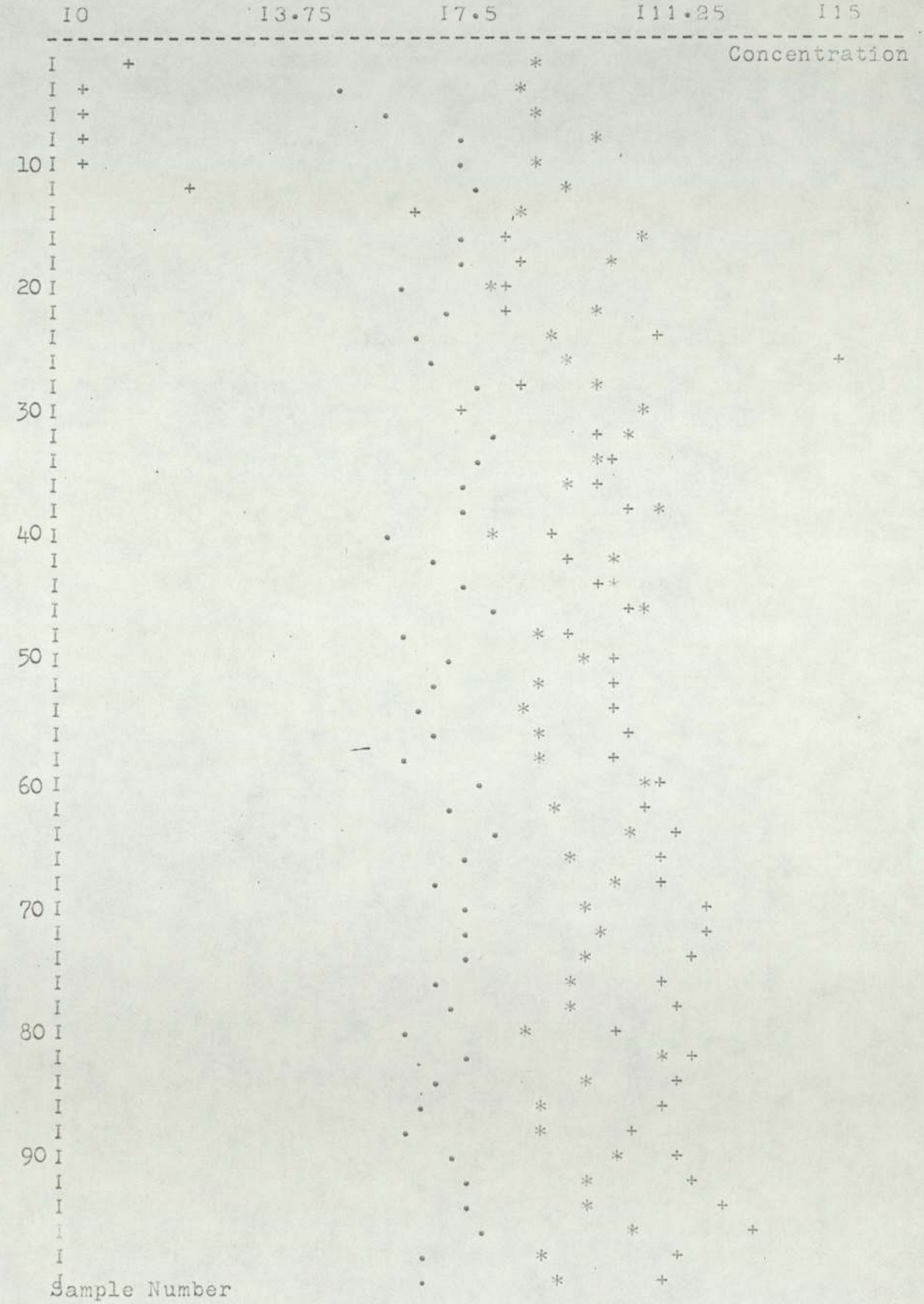
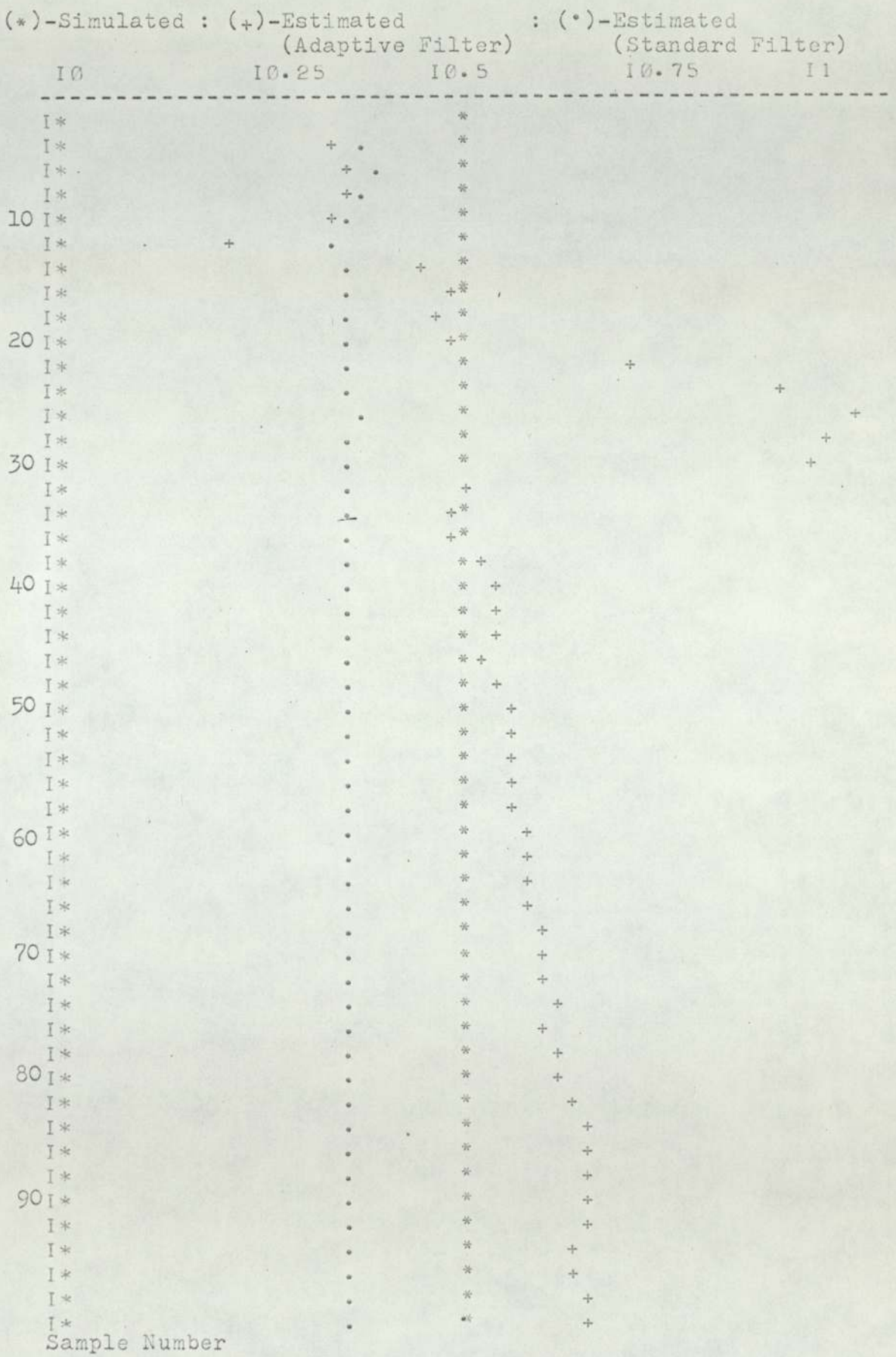


Figure (38).

Graph of Rate Constant (k_1).



and the true values. Figures (37, 38), in particular, show the improvement obtained by using the adaptive filter. Figures (35, 36) seem to indicate that there is little difference between the two, but this is to be expected because both A and B are measured so that most of the errors will be attributed to the estimates of A_o and k_3 by the standard filter.

5.4.3 Another Poor Model.

Figures (39, 40, 41, 42) show the results obtained by the adaptive filter; the standard filter; and the true values for the model.

$$\dot{A} = -0.846A + 0.385B$$

$$\dot{B} = 0.538A - B$$

with $H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $Ev_T = 0$ and $Evv^T = 0.833I$

The value of Q was taken to be 0.01 with $F_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

The true system was still the same (5.1.1) and the measurements were taken from the simulation Figure (6a).

These results show that the adaptive filter is superior to the standard filter, especially Figure (41) which shows the TRACE of the residual covariance matrix. Examination of Figure (41), however, shows a drawback to the adaptive filter as developed so far.

In this case F_4 was found to be, $F_4 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

This was chosen from the three following possibilities:

Figure (39).

Graph of Output Concentration (A).

(*)-Simulated : (+)-Estimated (Adaptive Filter) : (.)-Estimated (Standard Filter)

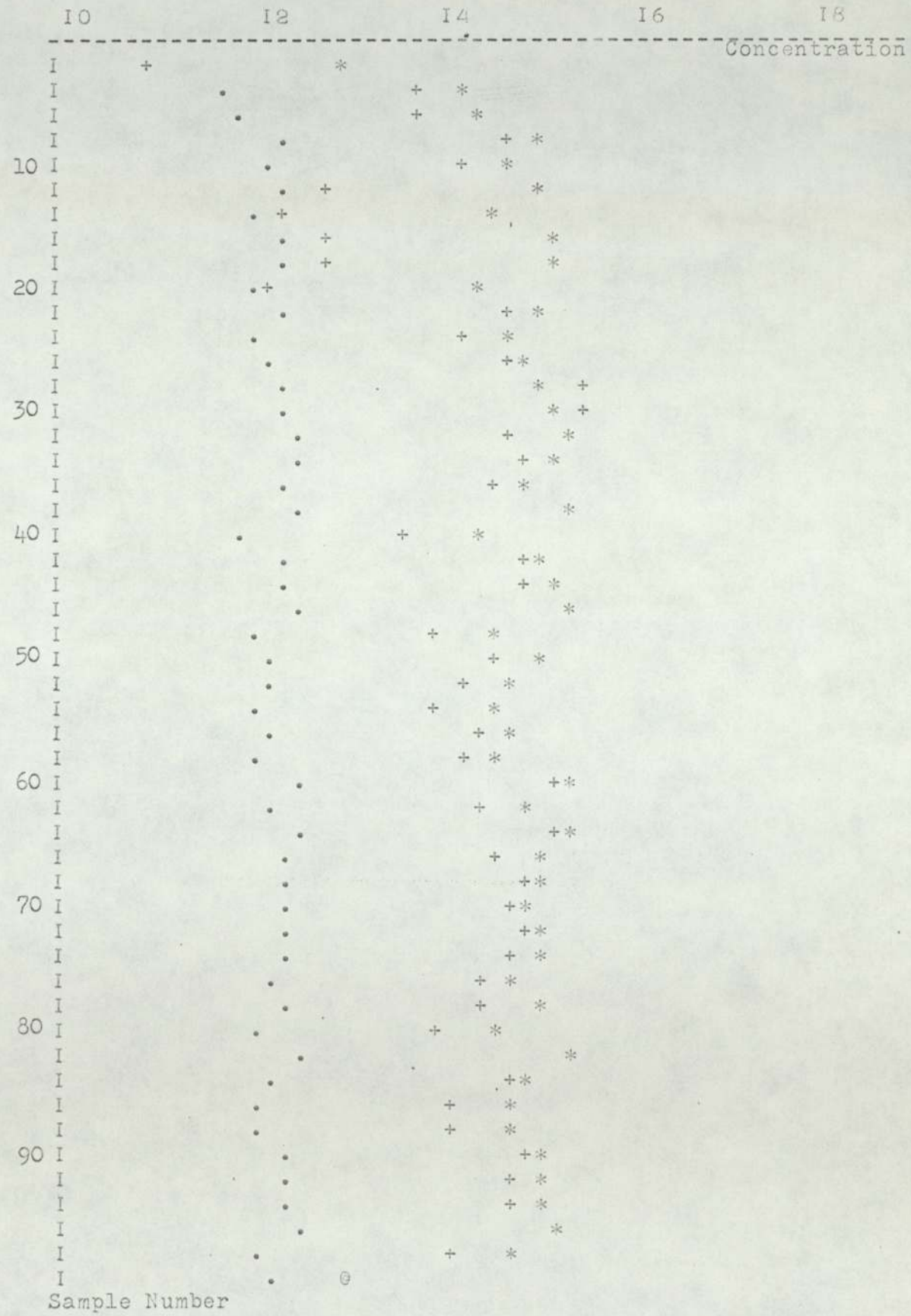


Figure (40).

Graph of Output Concentration (B).

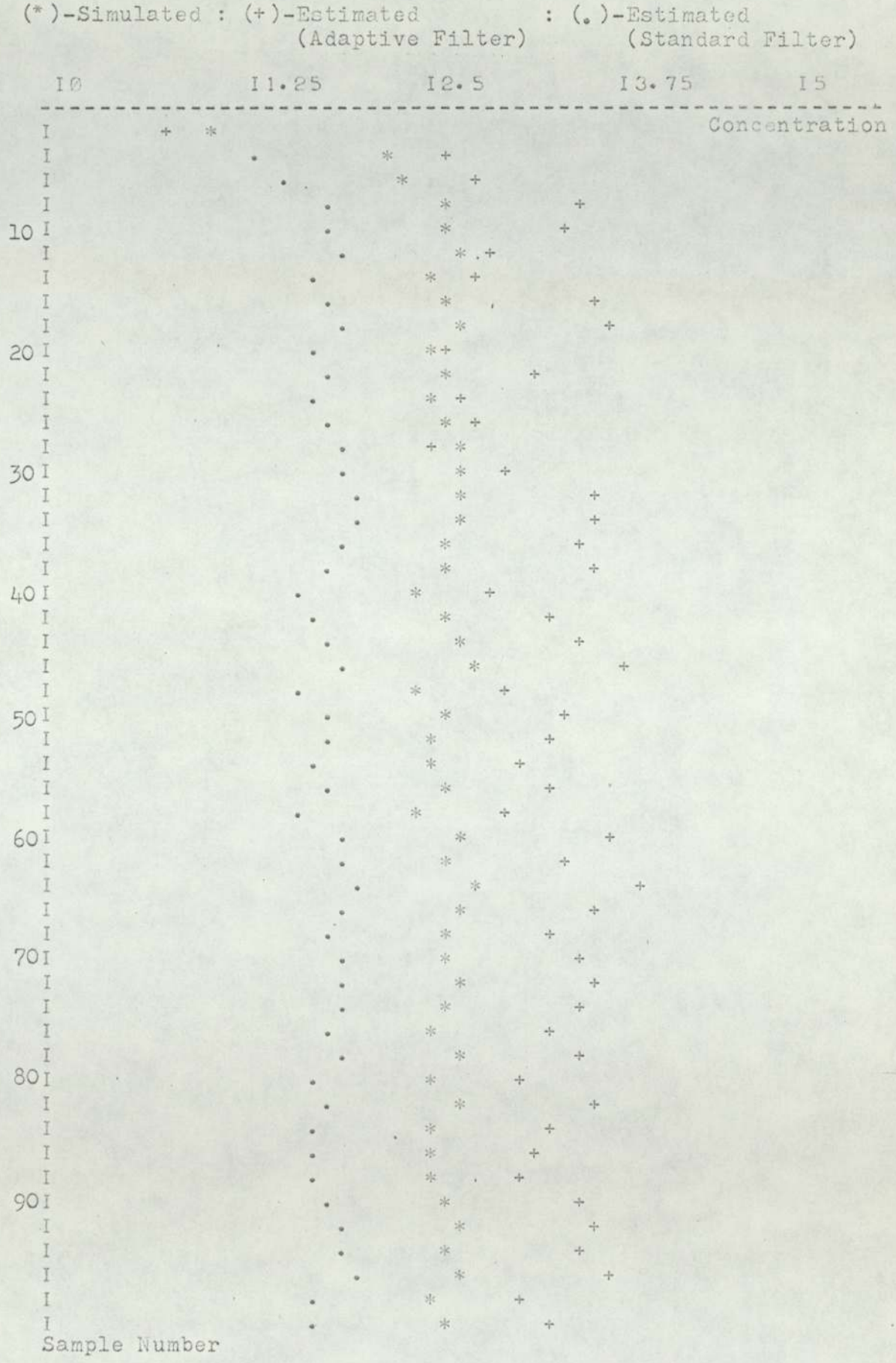


Figure (41).

Graph of the TRACE of the Residual Covariance Matrix (Σ).

(+)-Adaptive Filter : (.)-Standard Filter

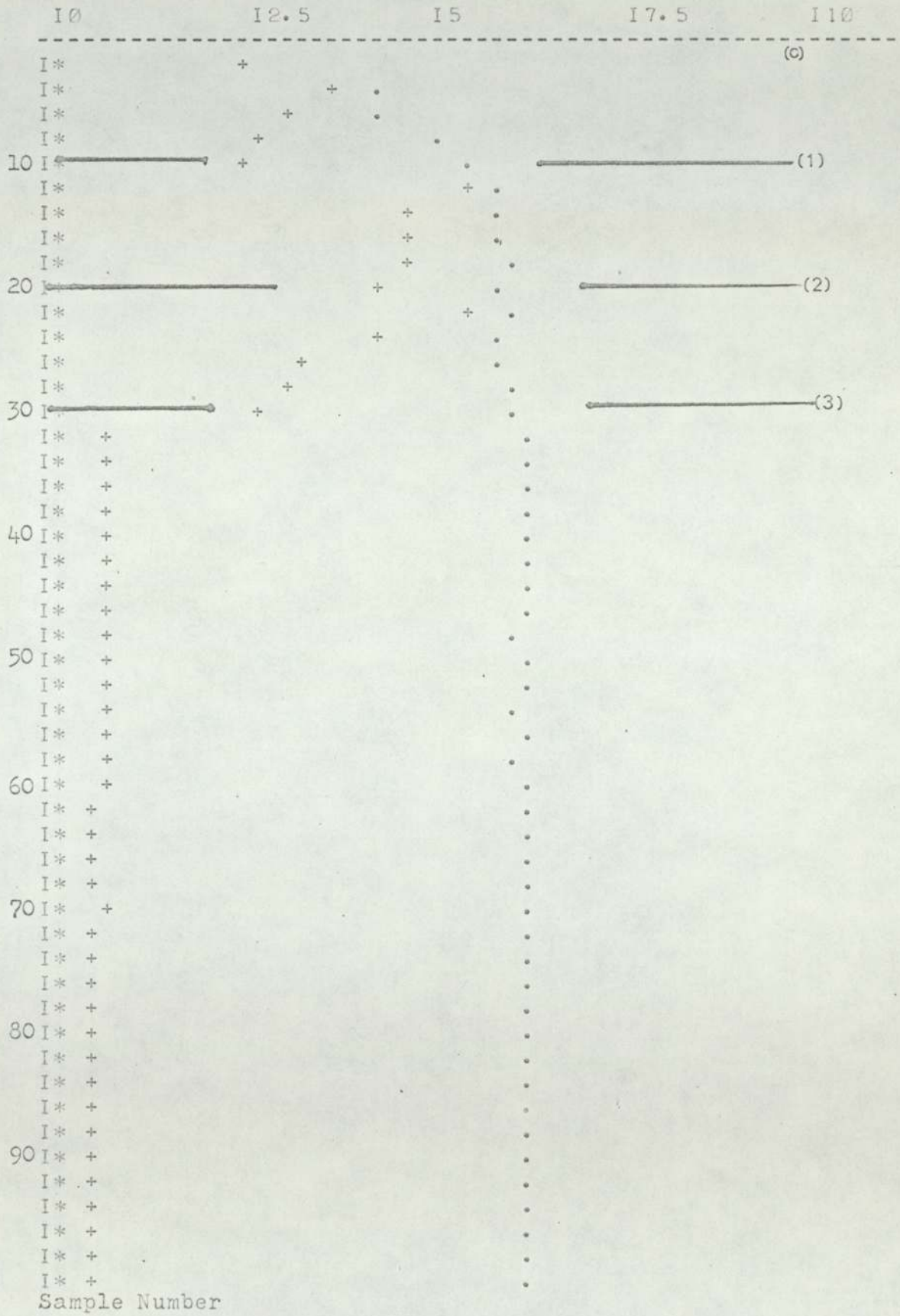
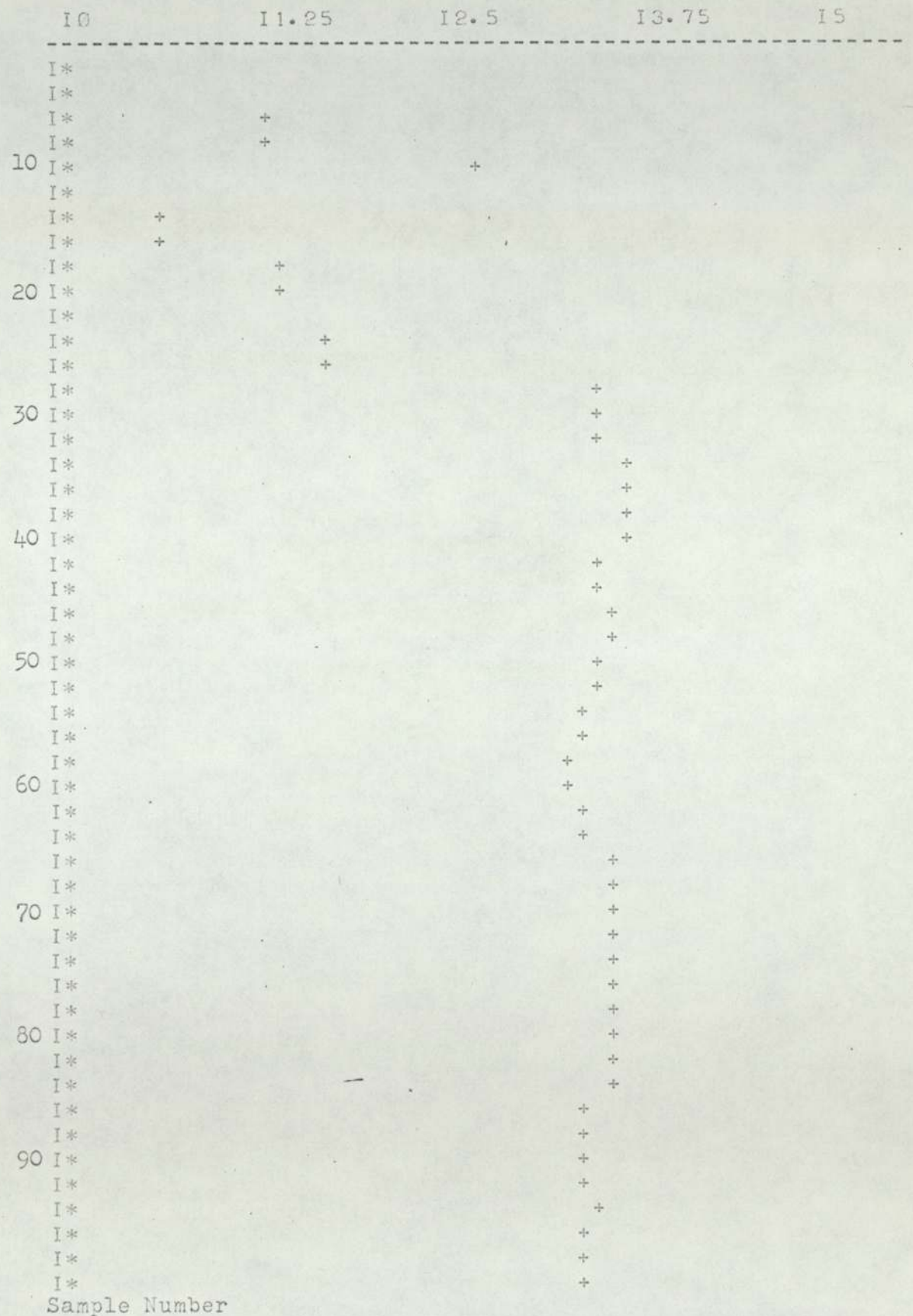


Figure (42).

Graph of Fictitious Input (w).

(+)-Adaptive Filter : (*)-Standard Filter



$$\begin{bmatrix}1\\0\end{bmatrix}, \quad \begin{bmatrix}0\\1\end{bmatrix}, \quad \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$$

by the fact that g at (1) (see Figure (41)) was less than g at (2) or (3). However, the rate of decrease of g between (2) and (3) was much greater than between (0) and (1) so that it would have been more reasonable to select

$$F_4 = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix} \quad \text{rather than } F_4 = \begin{bmatrix}1\\0\end{bmatrix}$$

This problem is caused by the fact that α is a moving weight, so that each measurement is treated equally. This results in the order of trial of the possible F_4's having an affect, because if the state of the filter is poor at the begining of a trial period then the convergence of g will be slow. To overcome this α is redefined as $\alpha = 1/i$ (where i is the number of samples taken) until $\alpha = 0.2$, where it remains constant. This places greater weight on recent measurements so that initial conditions are damped out more quickly. This is also a more practical situation as $\alpha = 1/i$ tends to zero as i tends to infinity, so that some lower limit to α would be necessary in any case.

Figures (43, 44, 45, 46, 47) show the results obtained with the adaptive filter with the modified value for α. In this case F_4 was found to be

$$F_4 = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$$

and the correction term

$$w = \begin{bmatrix}w_1\\w_2\end{bmatrix}$$

is shown in Figures (45, 46). These graphs show that w_1 is found to be positive to compensate for the omission of A_0 ; and that w_2 is found to be negative to compensate for the omission of C

Figure (43).

Graph of Output Concentration (A).

(*)-Adaptive Filter

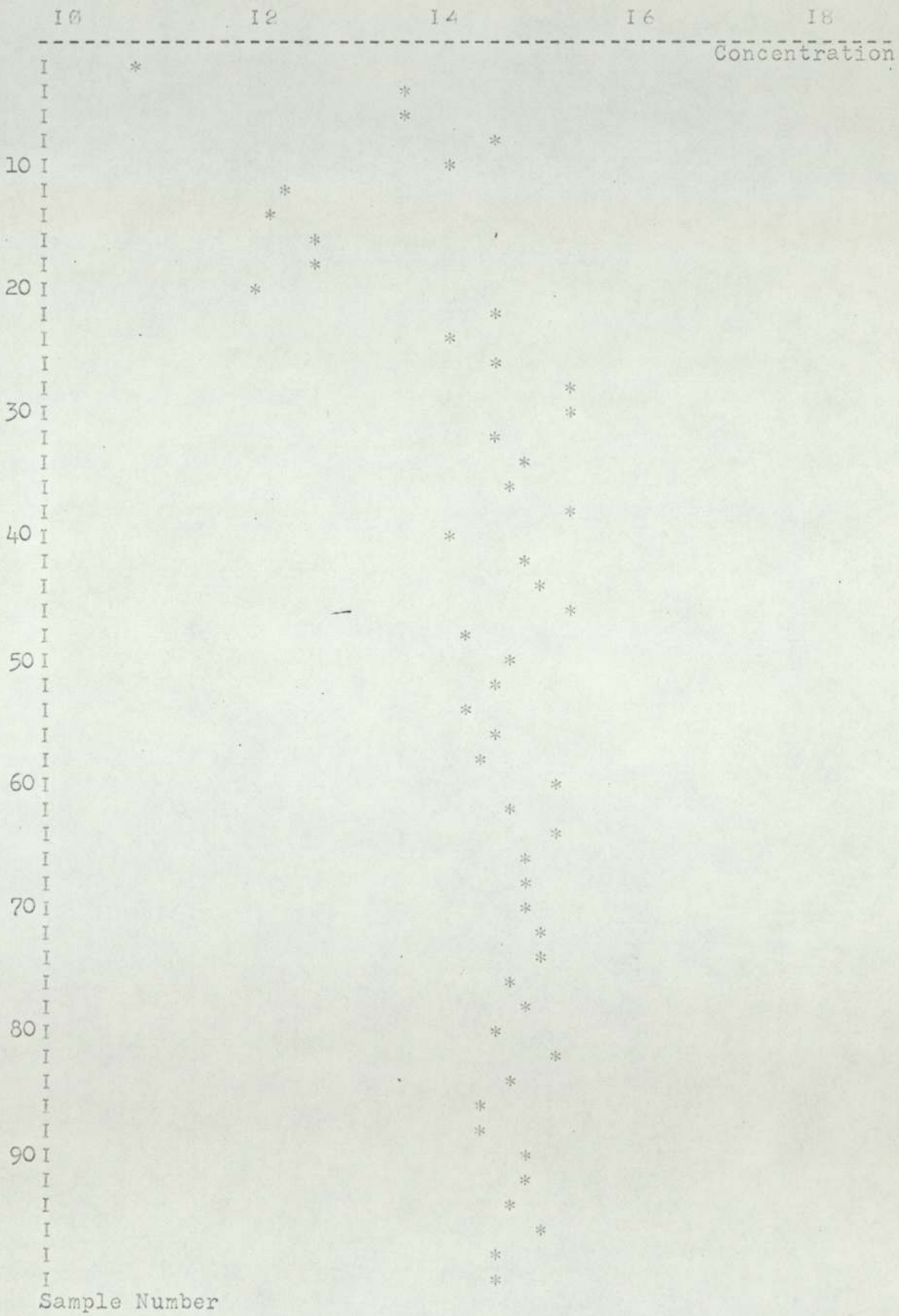


Figure (44).

Graph of Output Concentration (B).

(*)-Adaptive Filter.

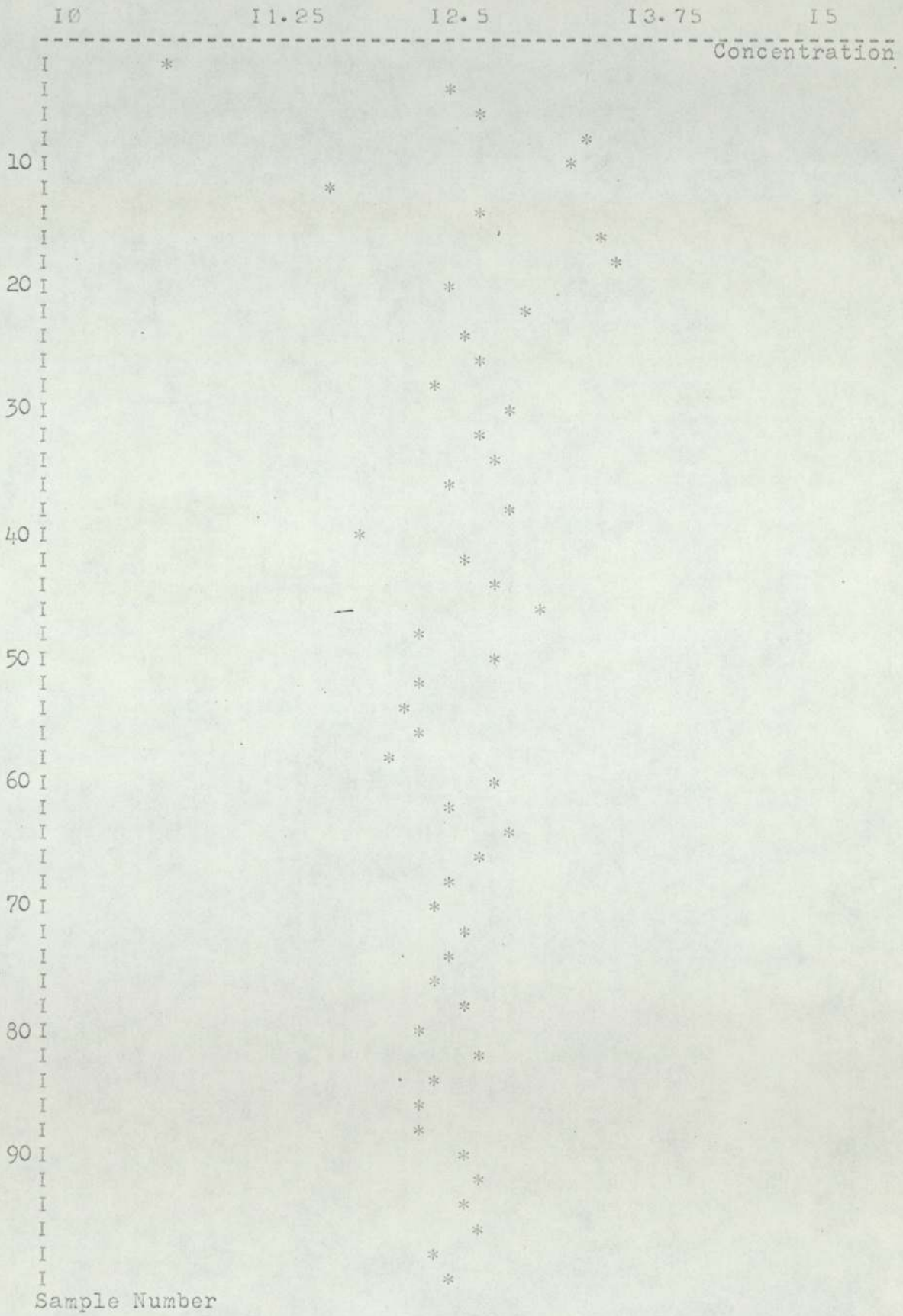


Figure (45).

Graph of Fictitious Input (w_1).

(*)-Adaptive Filter

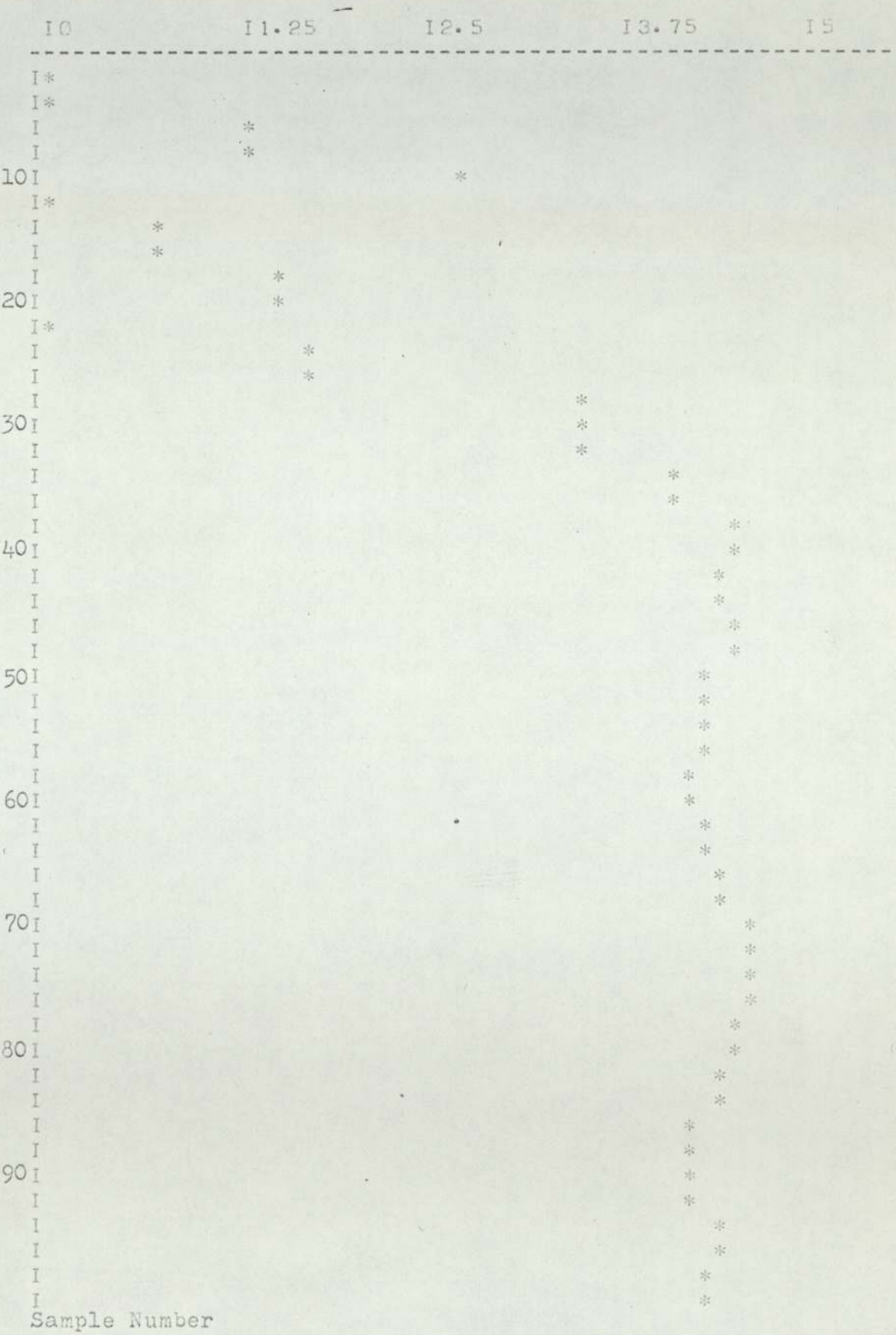


Figure (46).

Graph of Fictitious Input (w_2).

(*)-Adaptive Filter.

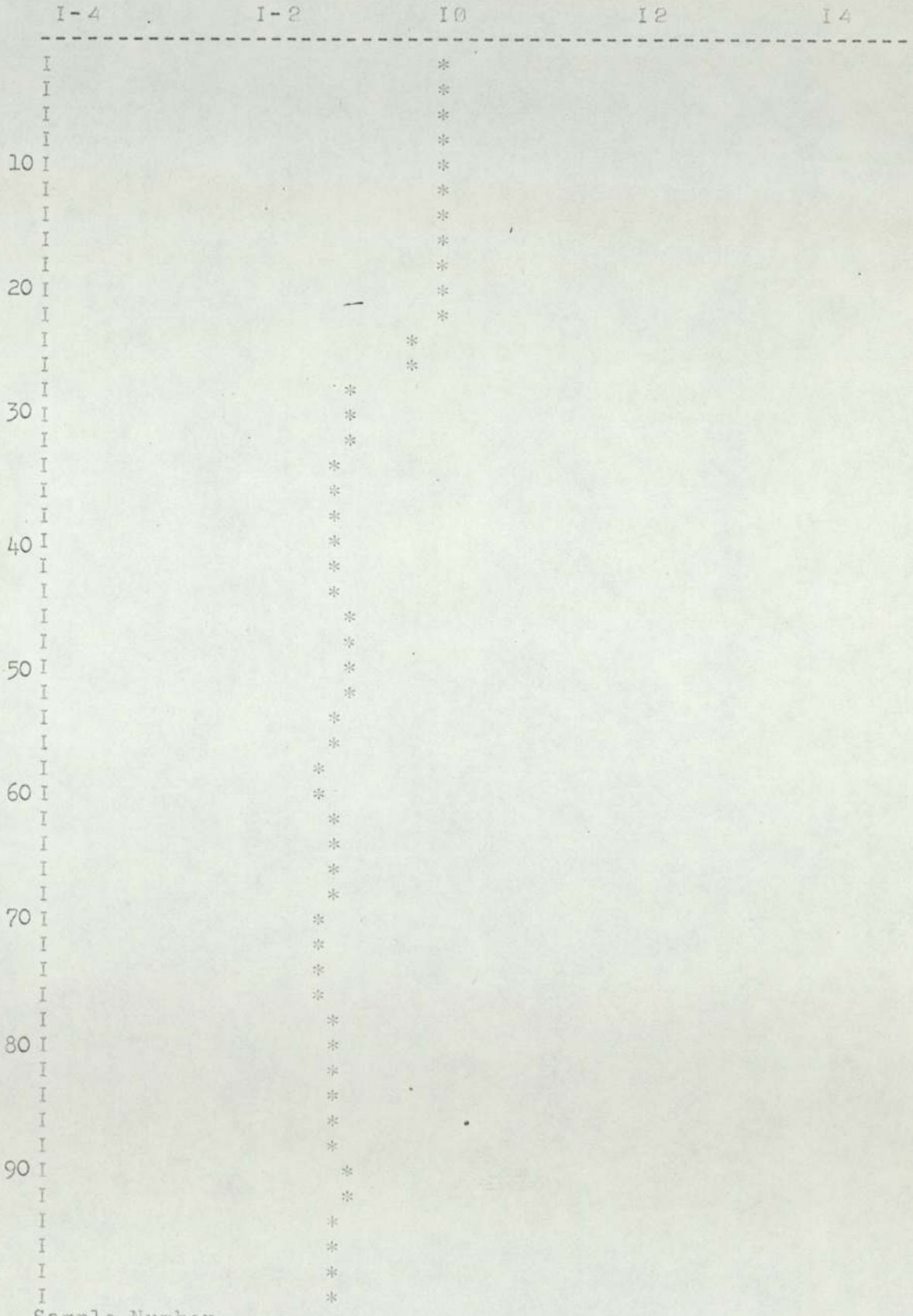
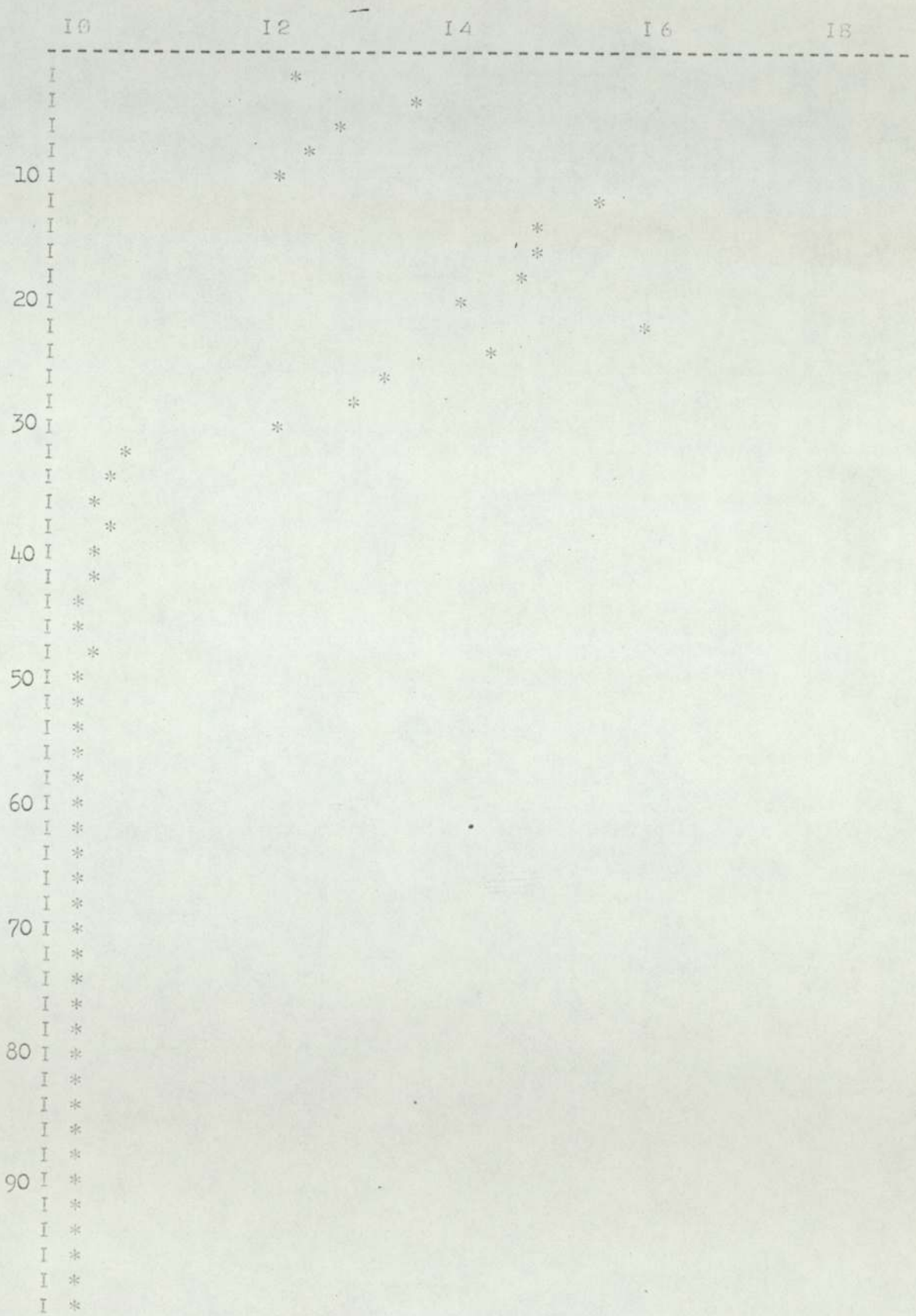


Figure (47).

Graph of the TRACE of the Residual Covariance Matrix (g).

(*)-Adaptive Filter.



and D from the model. Comparison with Figures (39, 40, 41, 42) show that these results are superior to the standard filter and are very good approximations to the true values.

5.4.4 Application To A Poor Non-linear Model.

The adaptive filter with $\theta=4$, $\beta=0.3$, and $\alpha=1/i$ until $\alpha=0.2$, where it remains constant, was then applied to the following system Figure (48).

$$\dot{x} = -(K+q/v)x + q/v\ x_o$$

$$\dot{y} = -q/v\ y - \frac{KH}{pC} + q/v\ y_o$$

with $K = EXP(36.49-(12100/y))$

$H = 10$

$q = 1$, $v = 4$, $p = 1$, $C = 1$, $x_o = 10$, $y_o = 350$

which is a single stage approximation to the system described previously (5.1.2). The measurements were obtained by adding random perturbations to the simulation results (Figures 9, 10) and were linked to the state variables by;

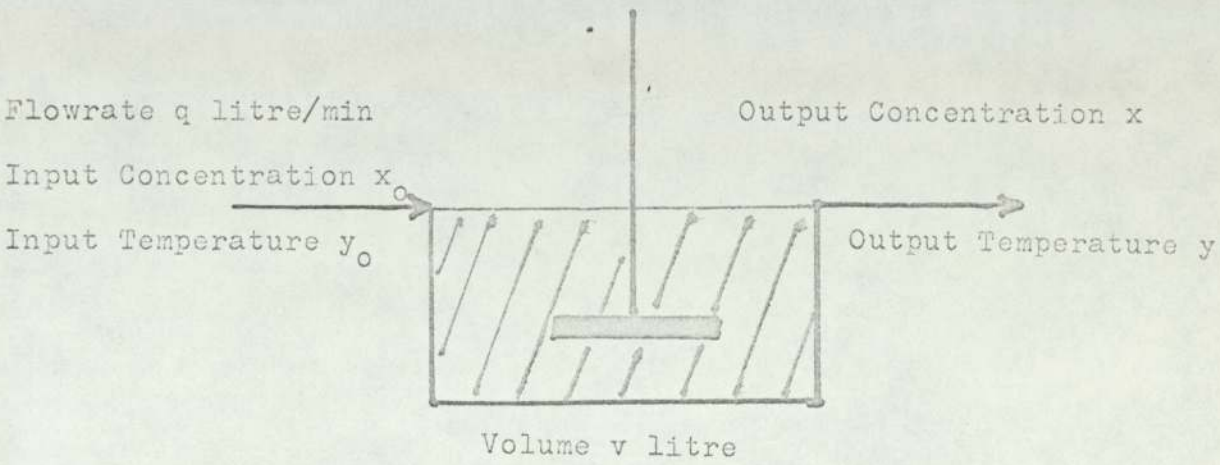
$$z = Hx+v$$

with $H = (0\ 1)$

Figures (49, 50, 51, 52) show the results obtained with both the adaptive filter and the standard filter. These results were obtained with the additional error $R_m=0.0833$ when it should have been 0.333 .

Figure (48).

Single Stage Stirred Tank Reactor With Non-linear Reaction.



Reaction In Tank.

$$\dot{x} = f_1(x, y, x_o)$$

$$\dot{y} = f_2(x, y, y_o)$$

Figure (49).

Graph of Output Concentration (x).

(•)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

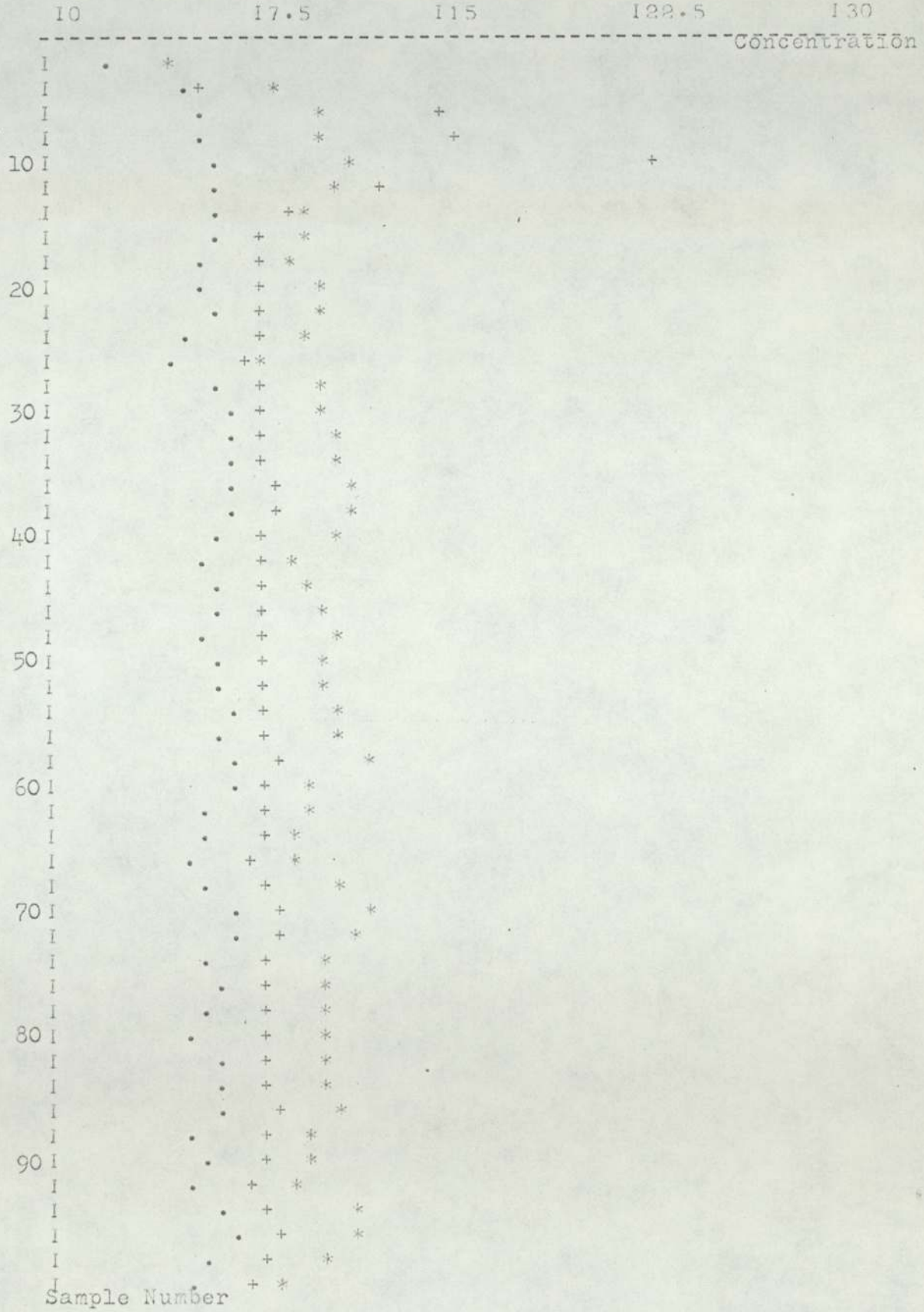


Figure (50).

Graph of Output Temperature (y).

(.)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

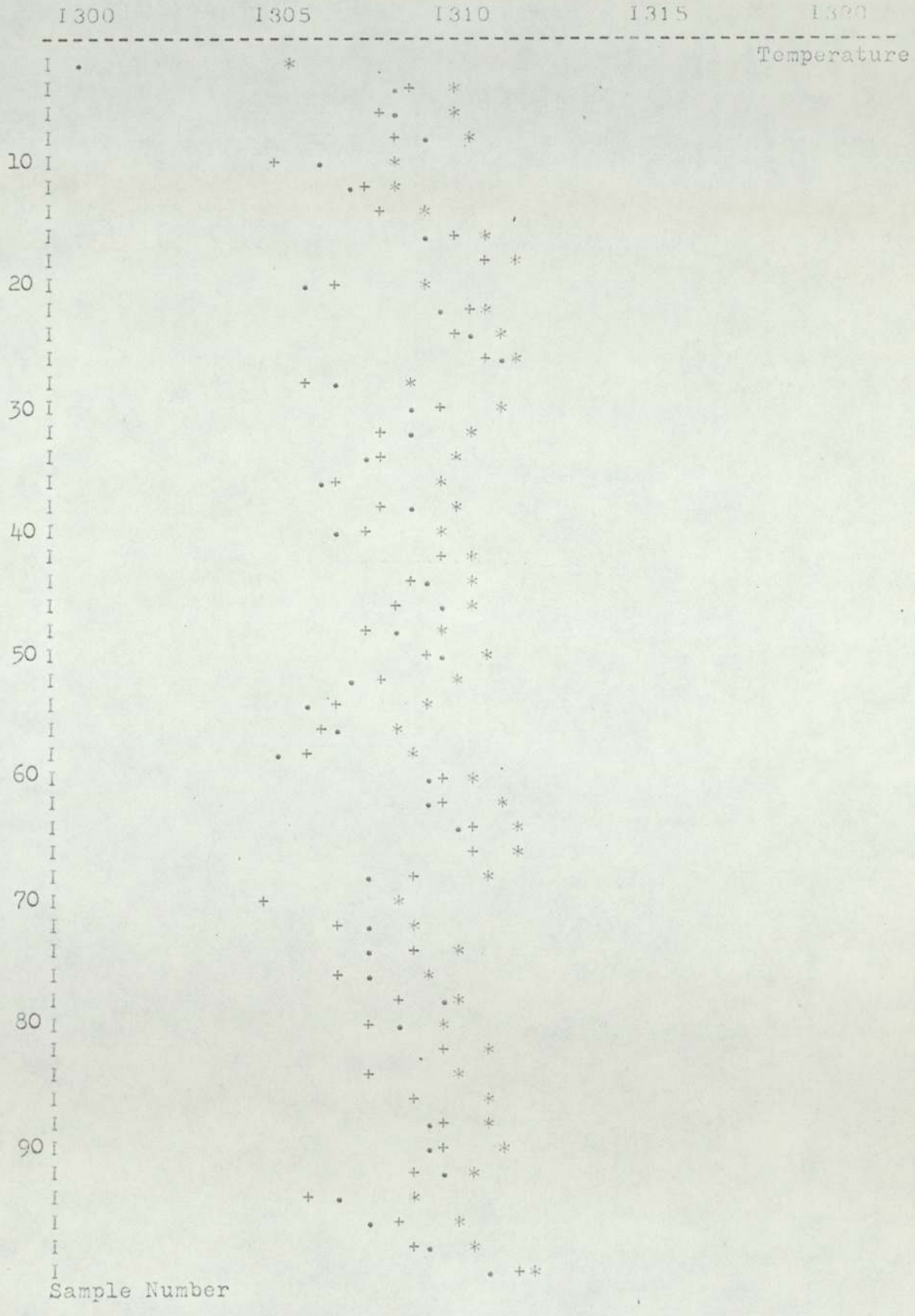


Figure (51).

Graph of Fictitious Input (w).

(+)-Adaptive Filter : (*)-Zero Line

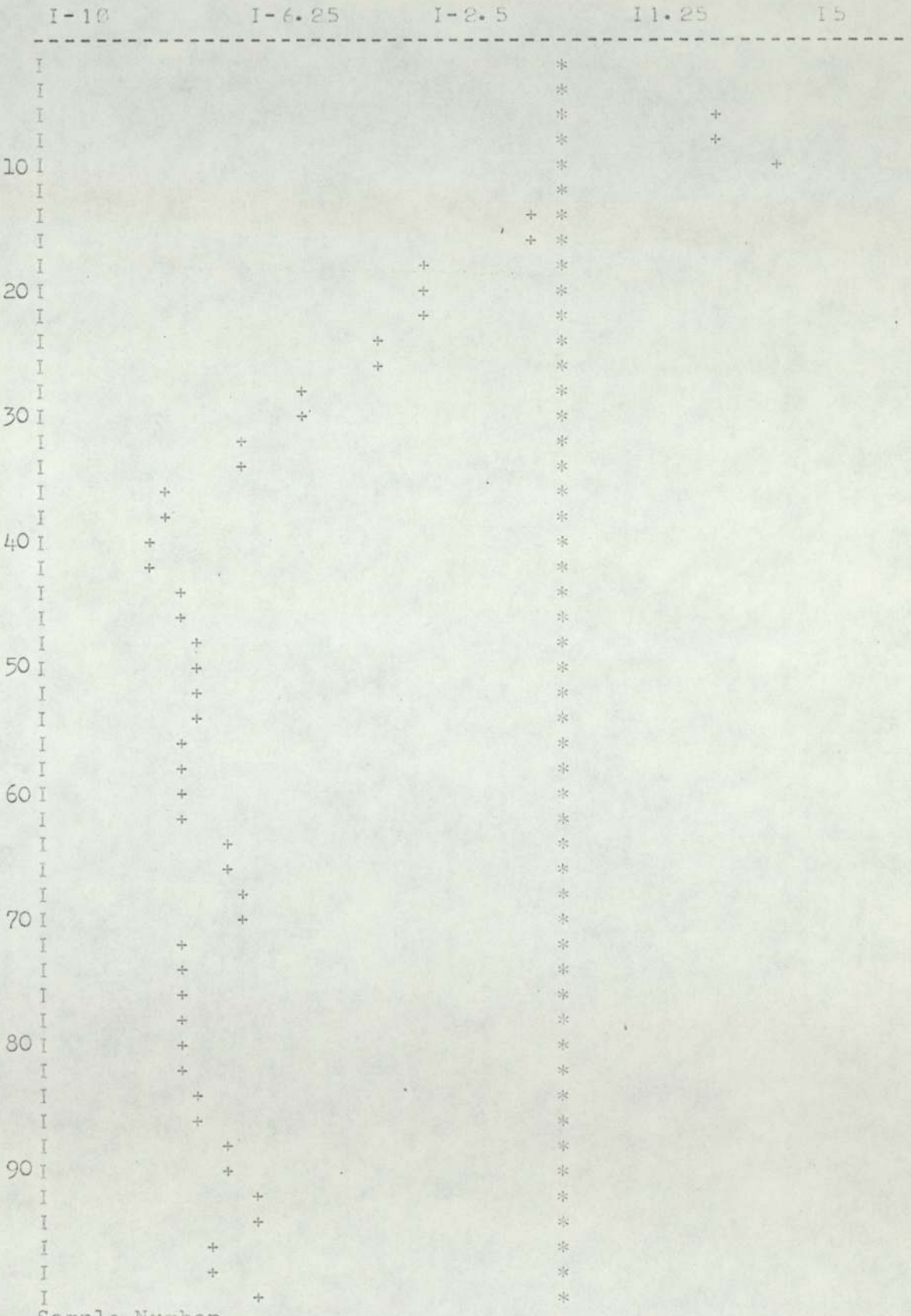
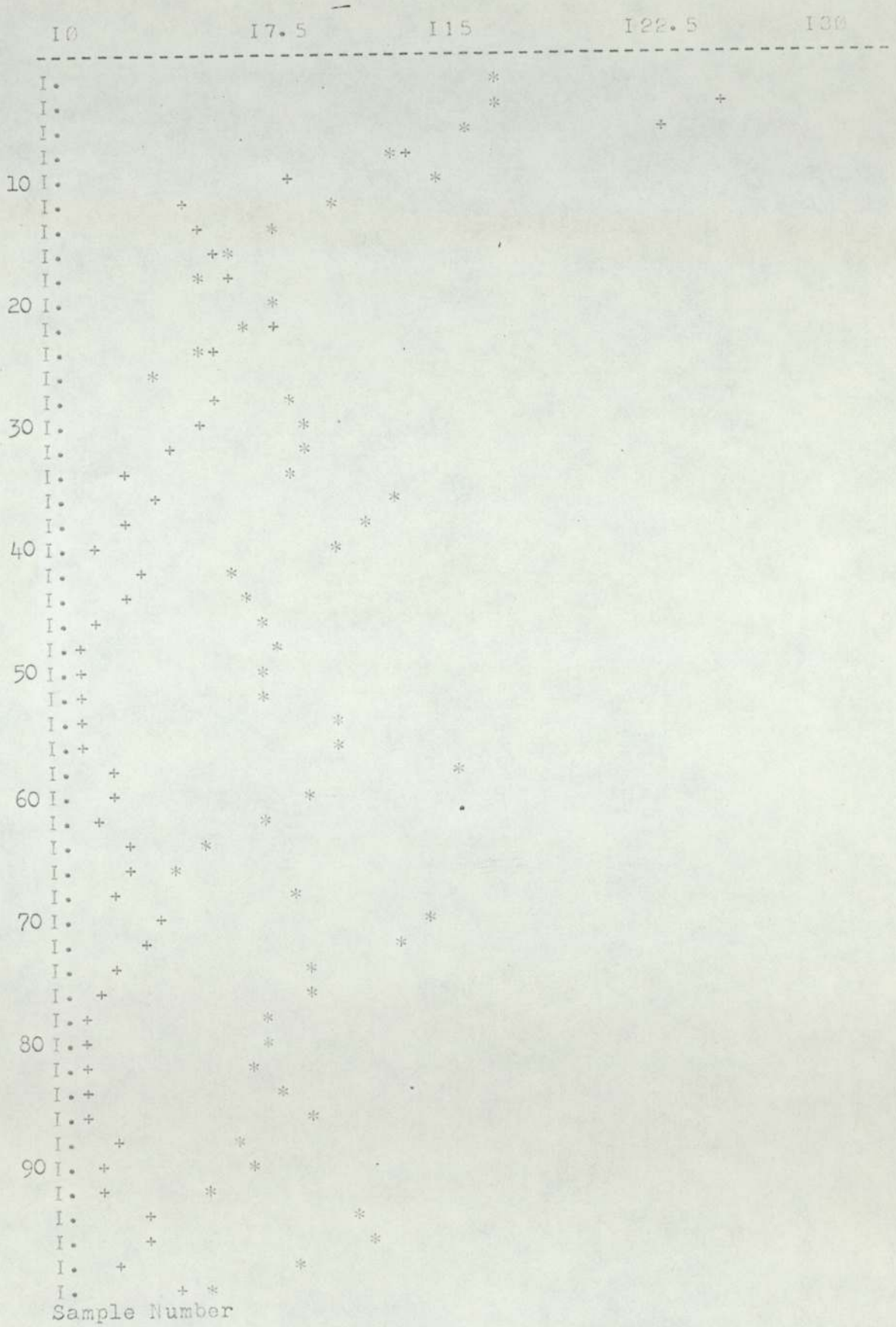


Figure (52).

Graph of the TRACE of the Residual Covariance Matrix (g).

(+)-Adaptive Filter : (*)-Standard Filter



Figures (53, 54, 55, 56) show the results for the same system but with R_m set correctly at 0.333 . The error committed by using the incorrect value of R_m alters the results obtained by the standard filter quite considerably, but has hardly any affect on the adaptive filter.

Figures (57,58,59,60) show the results obtained for the same system, but with R_m=8.33 .

These results demonstrate quite clearly the bias that results from using a standard filter in conjunction with an inaccurate model. This bias is much reduced by the adaptive filter. Furthermore, the adaptive filter provides an indication of the area of the model errors. This is provided by the compensation function (w) Figures (51, 55, 59). For this particular system the compensation function is difficult to analyse it merely tells us that the poor model is neglecting some heat loss from the system. In all these results (Figures 49 to 60) the matrix F_4 was found to be $F_4=\begin{bmatrix}0\\1\end{bmatrix}$. Clearly to improve this model, either some theoretical modelling is required, or more measurements should be taken.

5.5 General Discussion.

The results presented demonstrate a number of important points.

1. The standard filter provides useful estimates when the model and the statistical parameters are known (Figure 13).
2. The standard filter produces biassed, or even divergent, estimates when uncertainties are introduced into the model equation, or statistical parameters (Figures 37, 38, 39, 40).
3. The adaptive filter produces much improved estimates over

Figure (53).

Graph of Output Concentration (x).

(.)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

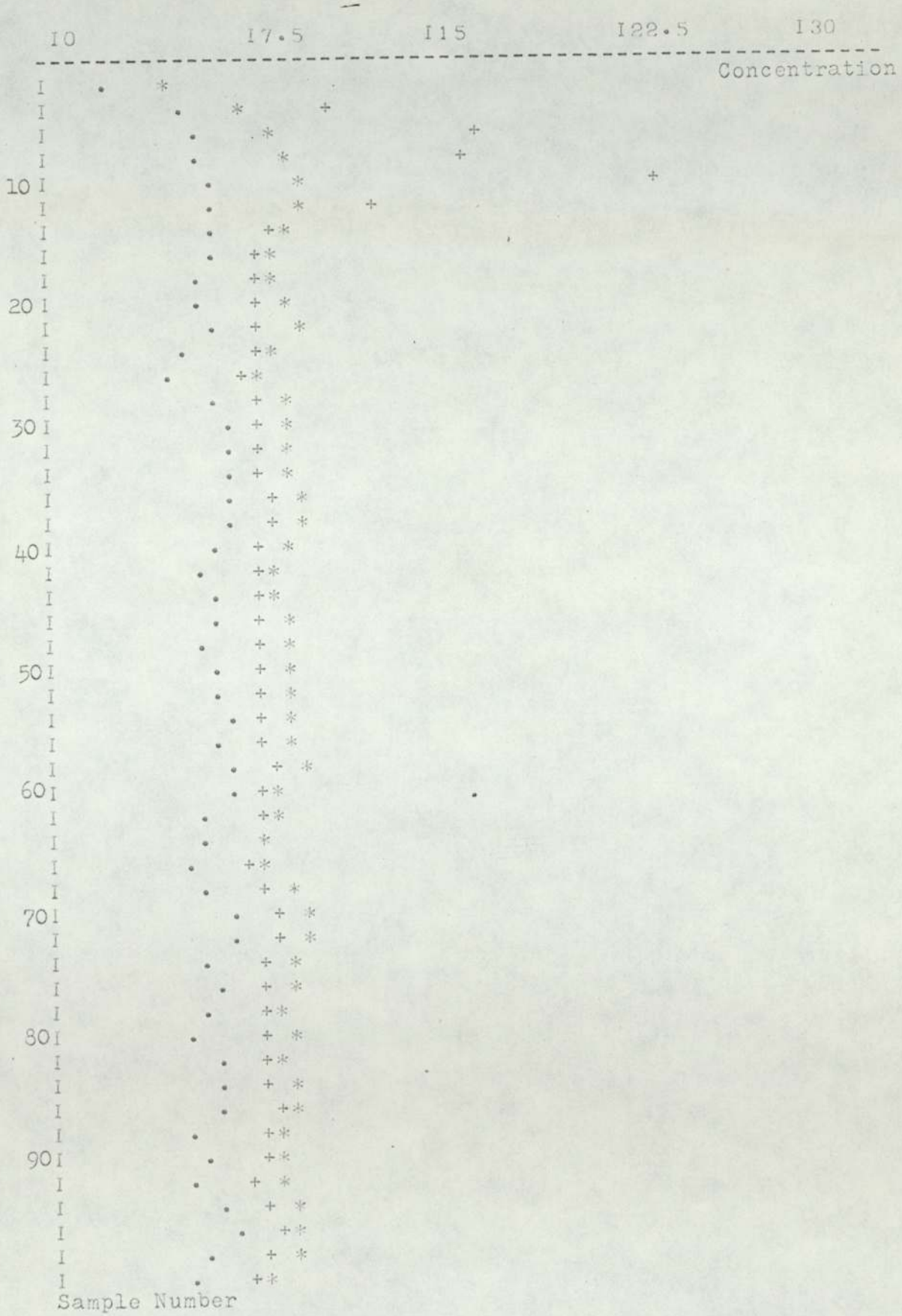


Figure (54).

Graph of Output Temperature (y).

(.)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

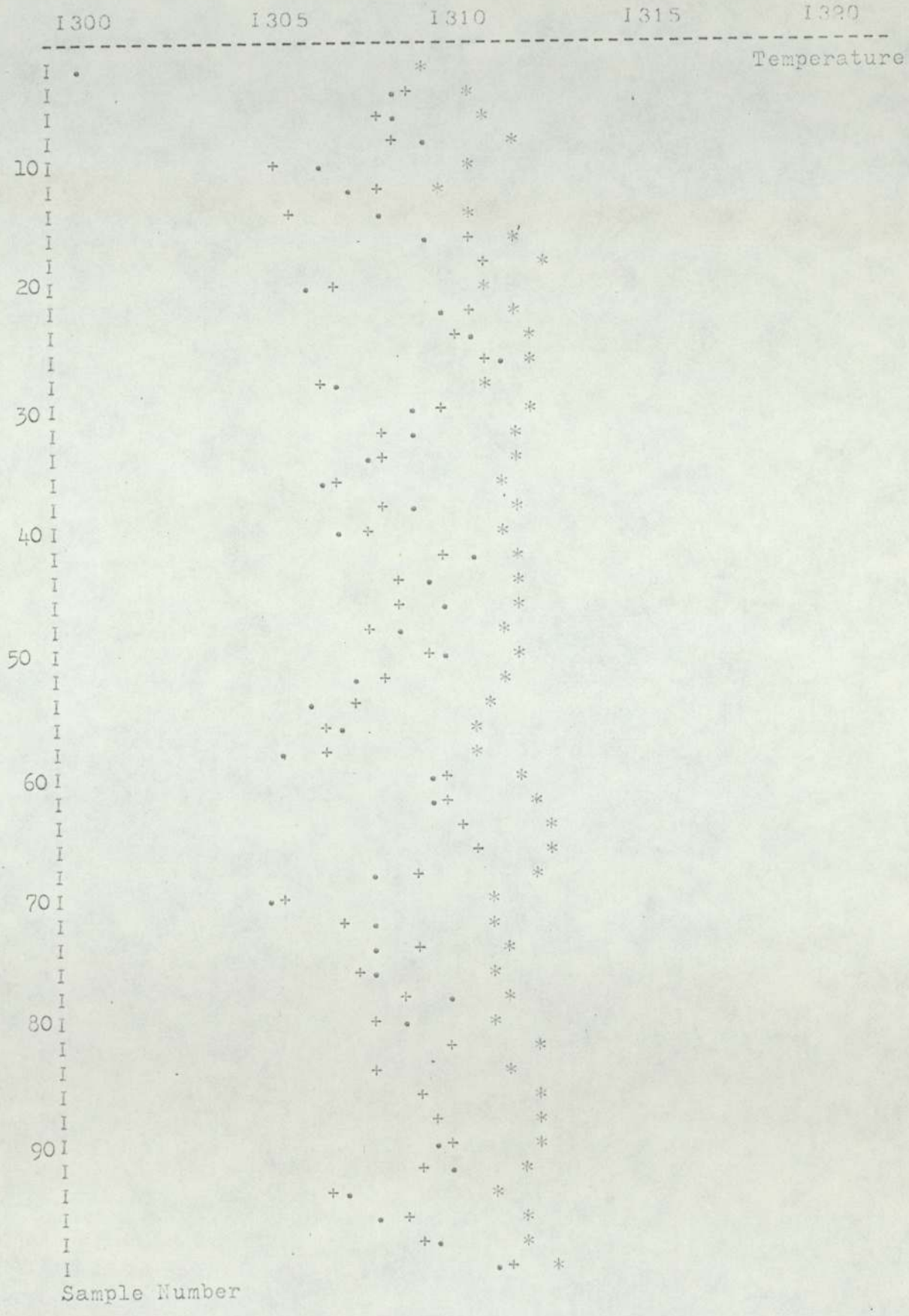


Figure (55).

Graph of Fictitious Input (w).

(+)-Adaptive Filter : (*)-Zero Line

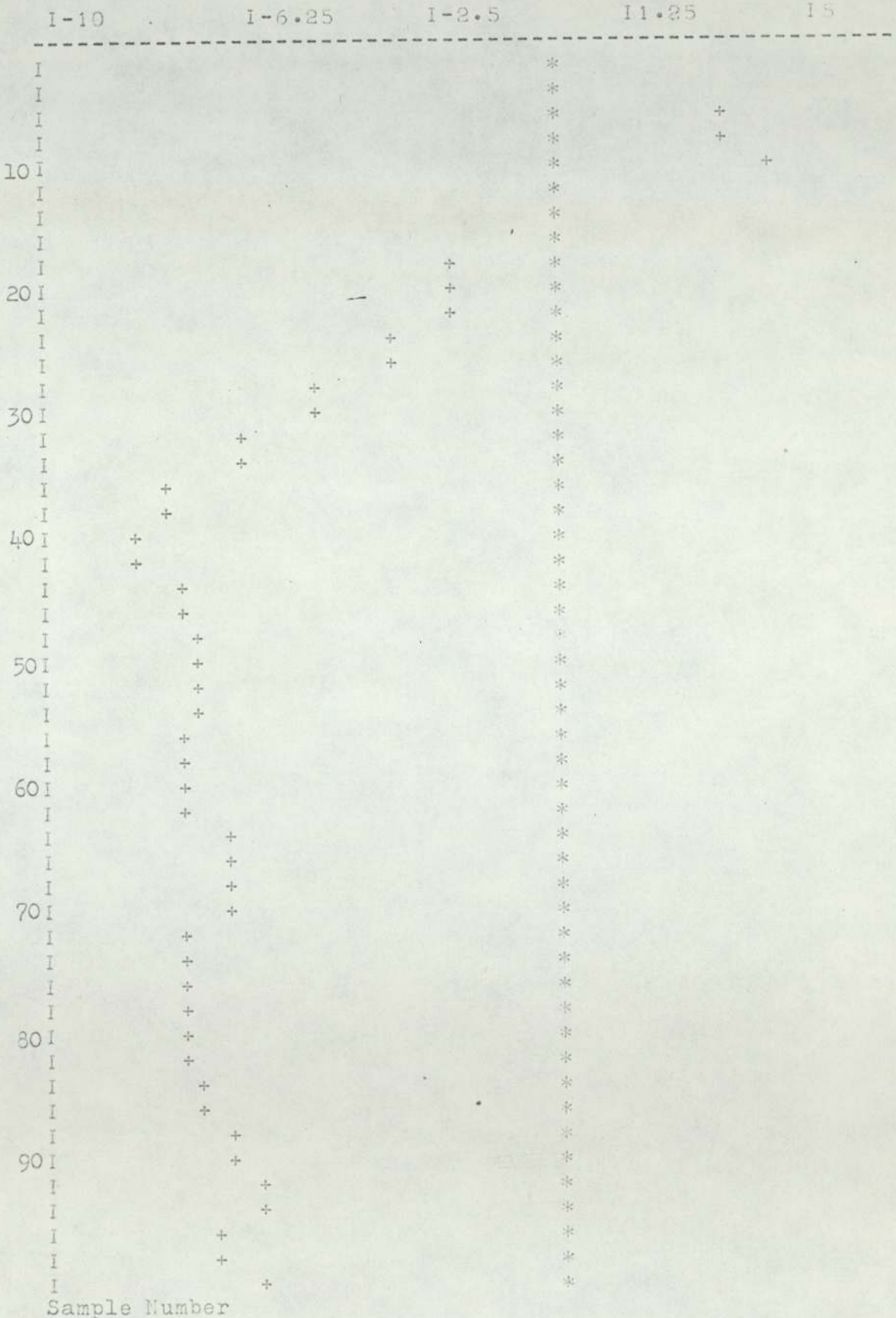


Figure (56).

Graph of the TRACE of the Residual Covariance Matrix (g).

(+)-Adaptive Filter : (*)-Standard Filter

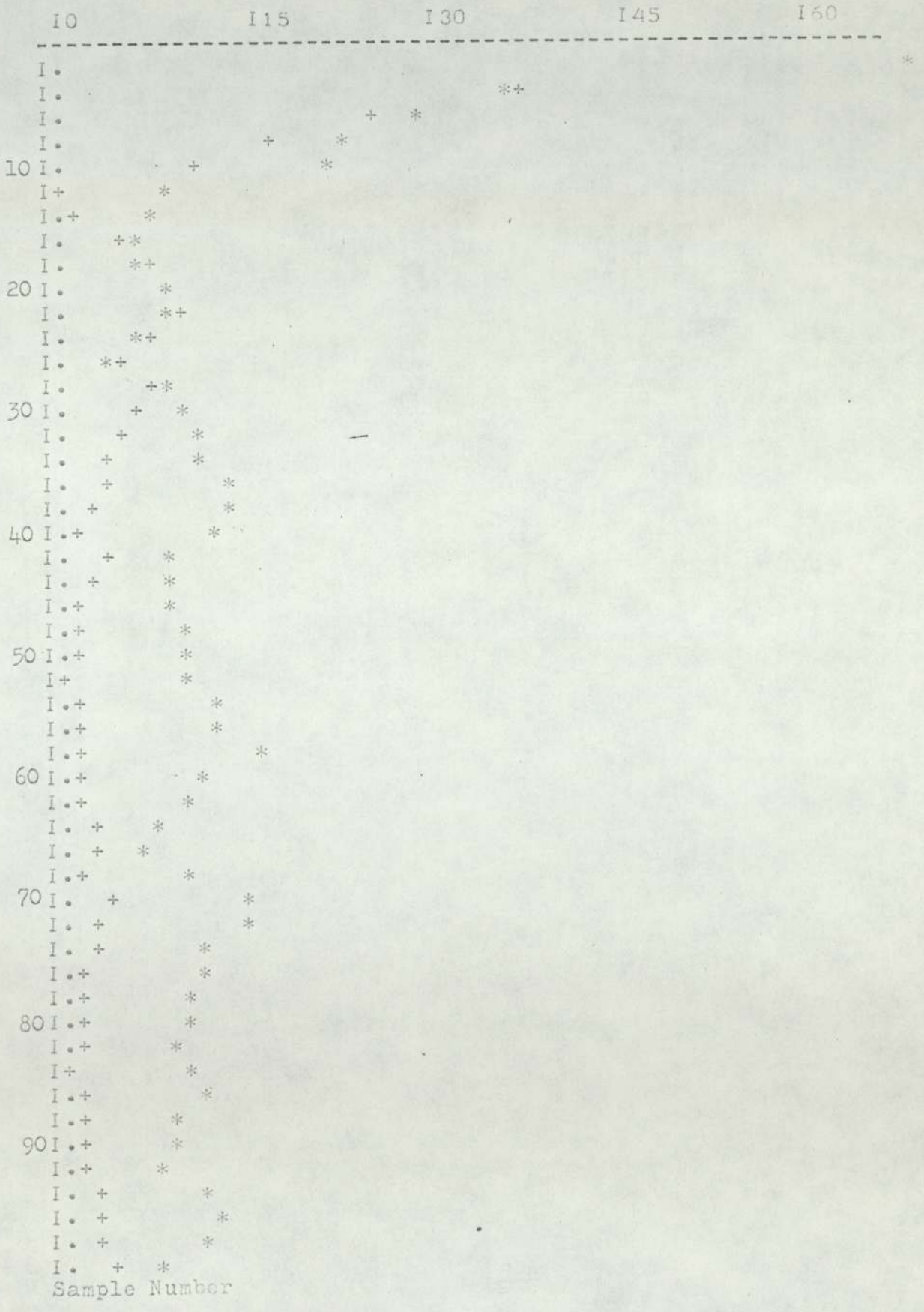


Figure (57).

Graph of Output Concentration (x).

(.)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

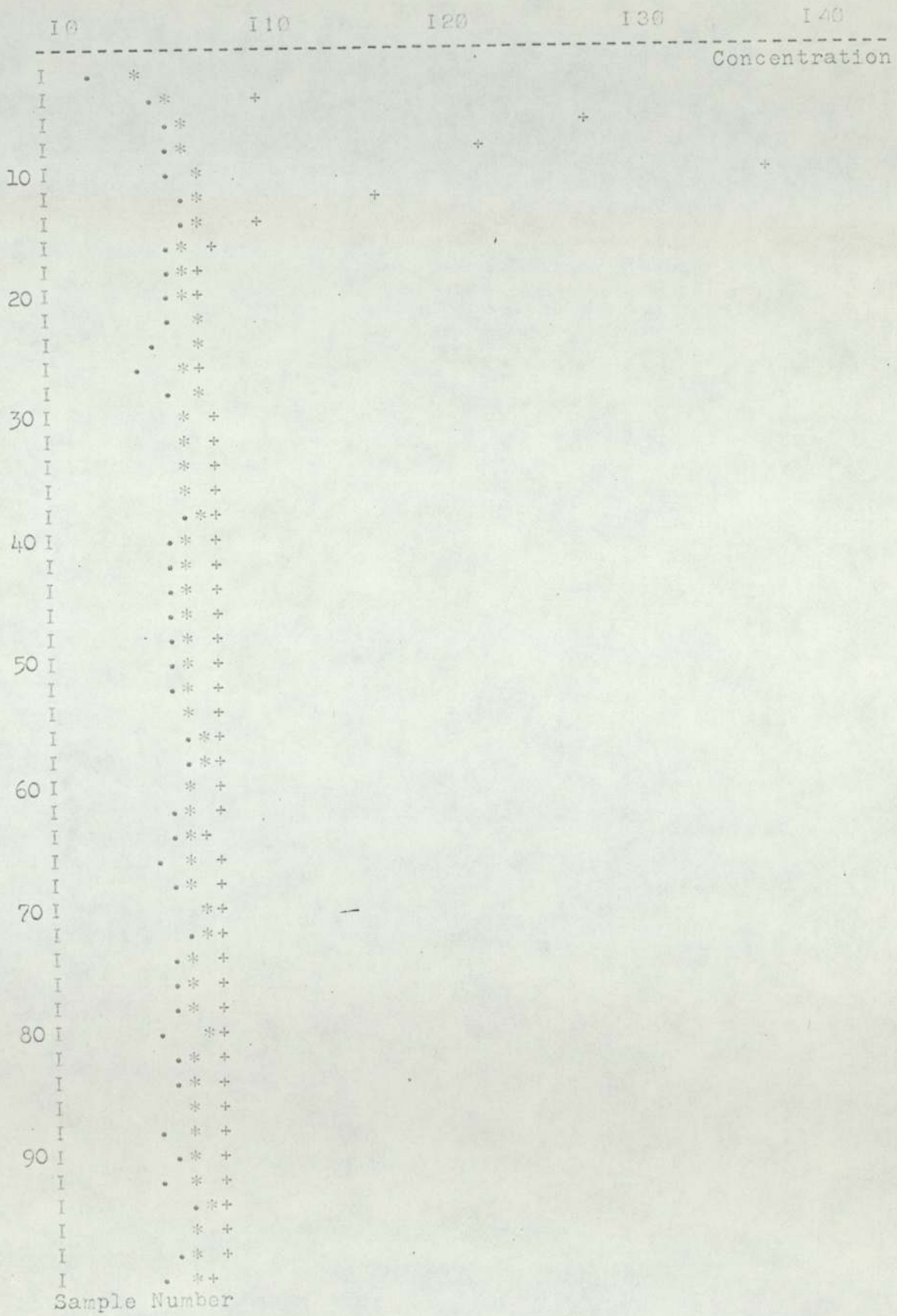


Figure (58).

Graph of Output Temperature (y).

(.)-Simulated : (+)-Estimated (Adaptive Filter) : (*)-Estimated (Standard Filter)

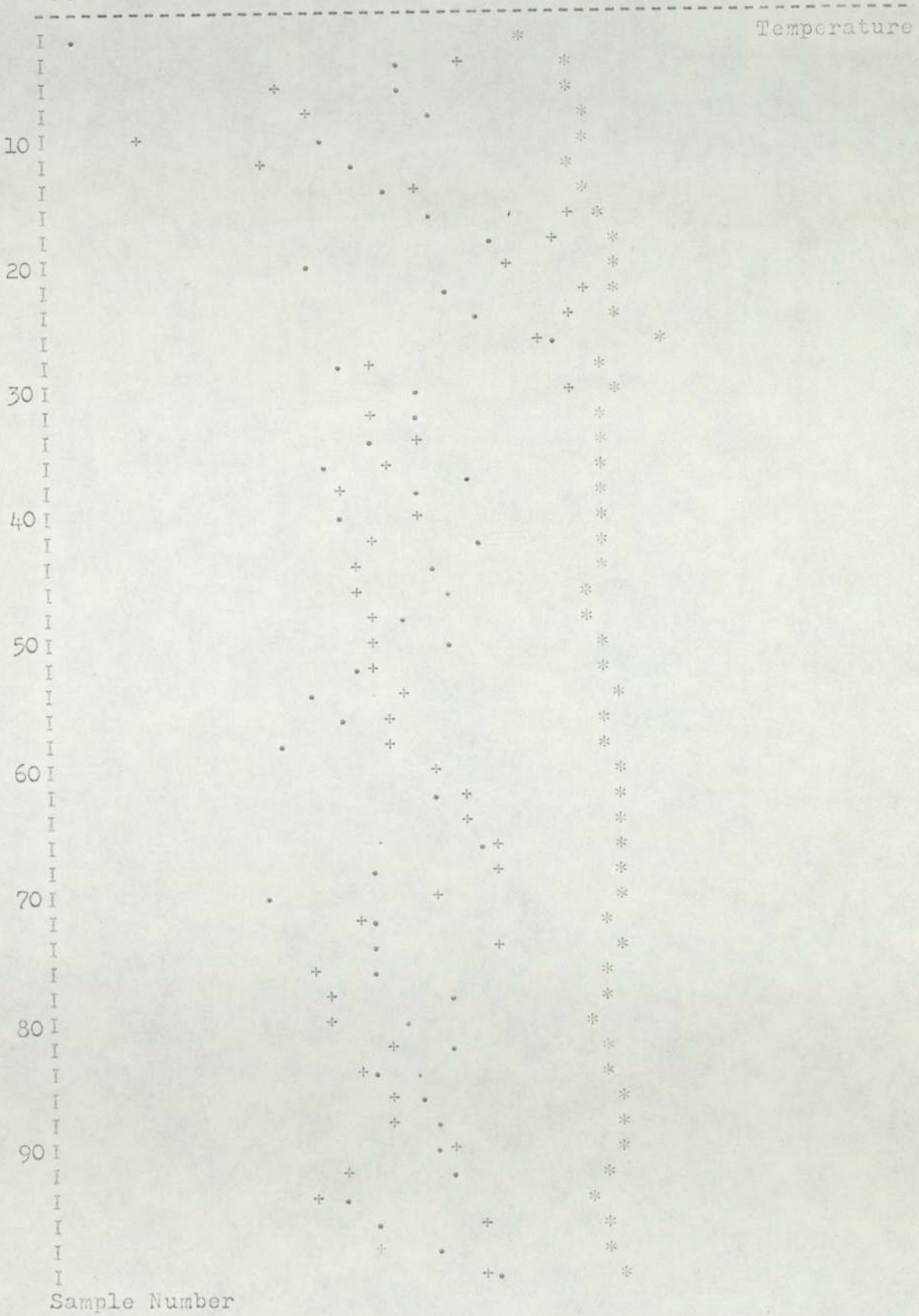


Figure (59).

Graph of Fictitious Input (w).

(+)-Adaptive Filter : (*)-Zero Line

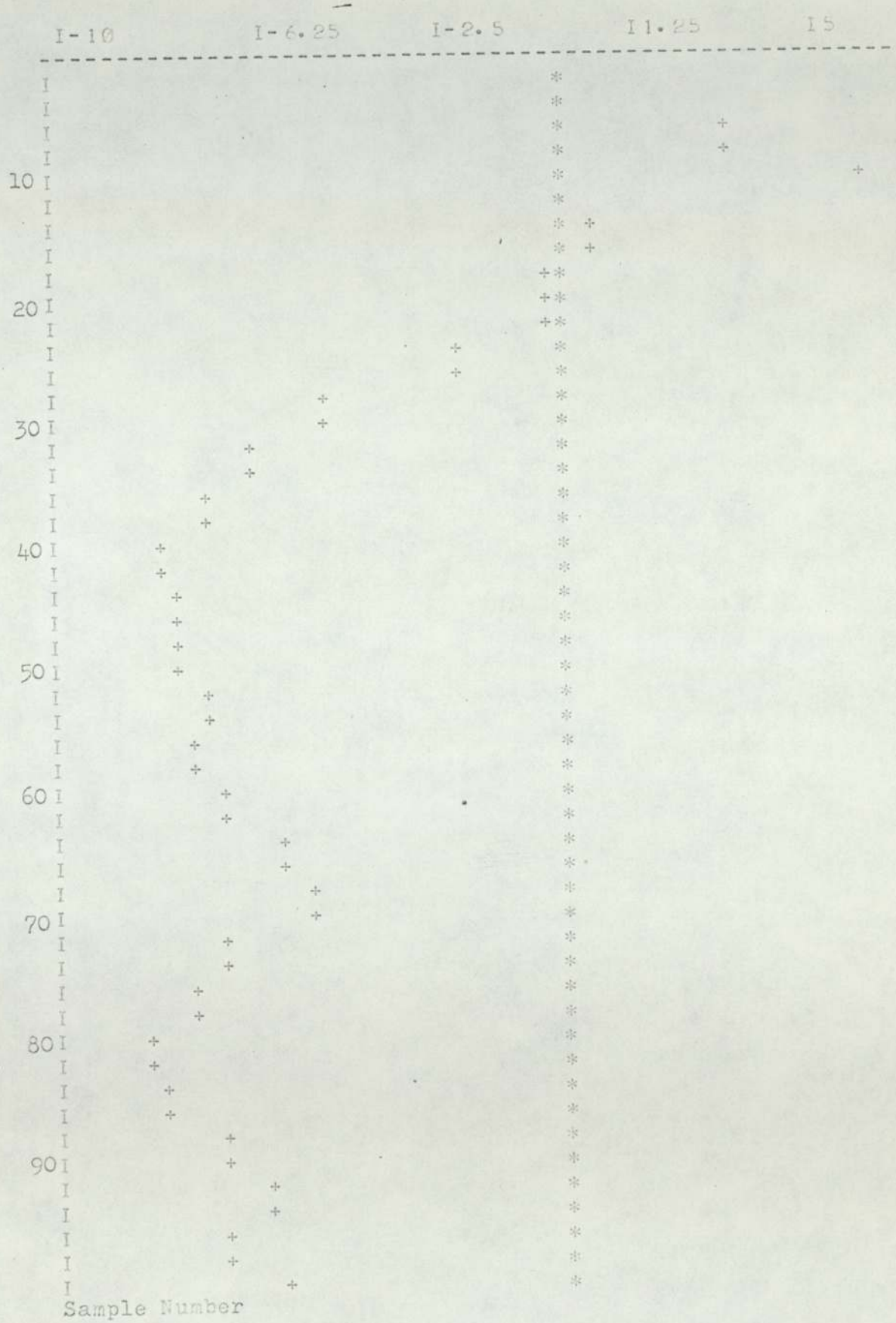
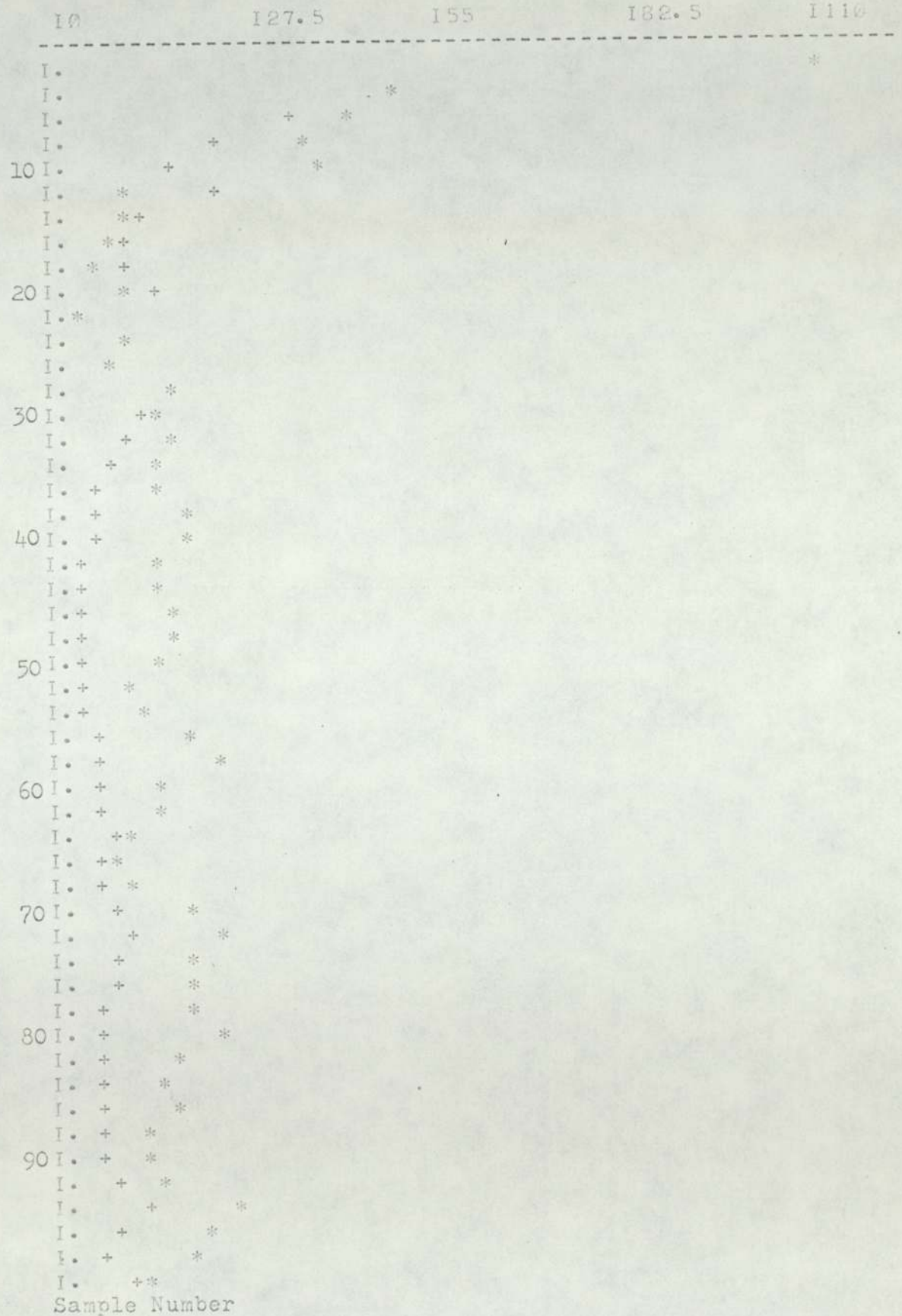


Figure (60).

Graph of the TRACE of the Residual Covariance Matrix (Σ).

(+)-Adaptive Filter : (*)-Standard Filter



the standard filter when dealing with uncertain models. (Figures 37, 38, 43, 44).

4. The adaptive filter provides an indication of the model errors, which can be very helpful to any attempt to improve the model. (Figures 45, 46, 51, 55, 59).

The results also show that the adaptive filter is robust with respect to its parameters (α, β, and θ). That is, the adaptive filter performs well over quite a large range of these parameters. However, problem areas have been indicated. (Figures 19, 20, 32, 33). When θ becomes too small or β becomes either too small or too large the estimates become 'noisy' and tend to oscillate about the true values. Another problem that has been demonstrated is that of unobservability. (appendix 13). This problem is caused by attempting to find too many fictitious inputs and is simply avoided by restricting the search for F_4 so that unobservable models are omitted.

The choice of ten sample steps for the testing of each possible F_4 is arbitrary, but it was felt that it was long enough for the filter to converge, without being too long, so that the optimum F_4 could be found quickly and efficiently.

It is clear that the optimum values of the adaptive filter parameters (α, β, and θ) are inter-related and in some way depend on the speed of response of the system. However, the fact that these parameters are based upon sample number and not on the independent variable (time), compensates for the system's speed of response, so that the results obtained should be fairly general.

CHAPTER (6).

CONCLUSIONS AND POINTS FOR FURTHER DEVELOPMENT.

6. Conclusions and Points for Further Development.

This thesis has discussed the importance of system identification, or, state and parameter estimation in modern control engineering. The relevant literature has been examined and the Kalman filter has been introduced and discussed. Problems arising in the application of the Kalman filter have been pointed out.

A number of theoretical developments have been made in this thesis. First a method of solving differential equations, providing piece-wise continuous functions, has been developed. The basic Kalman filter has then been reformulated by incorporating the new continuous solution programme, to provide piece-wise continuous estimates. The ability to produce continuous estimates has required the introduction of continuous measurement functions. The measurement functions described in the text are particularly simple to construct from the discrete data obtained by sampling the system, and ensure that the estimates are continuous at every point. However, there is a great deal of flexibility in the way the measurement functions are defined and this allows for freedom in the choice of sampling strategy (e.g. where in the range of solution is most convenient to sample; how many samples to include in the range of solution or whether to vary the length of the solution range, etc.). This is an area where further work might produce interesting results.

The continuous filter so developed is analogous to the original Kalman filter so the problems, already mentioned, which affect the Kalman filter are shared by this new filter. These problems, whether caused by uncertainties in the model, or by the filter statistics, or neglected non linearities, show themselves

by making the state estimates and the estimation error covariance matrix inconsistent statistics. That is, the estimation error covariance matrix calculated in the filter may become a very poor representation of the true (but unknown) estimation errors. A number of approaches to this problem, proposed in the literature, have been discussed and various short comings highlighted (e.g. the need for unreasonable a-priori information, or the inability to estimate parameters). These approaches, however, have served to introduce the concept of an adaptive filter i.e. a filter which is able to change its own operating parameters to improve its estimations. This concept of an adaptive filter has been taken and a new adaptive filter has been developed which, although utilising some of the ideas already presented in the literature has introduced new concepts, helpful in solving some of the problems of adaptive filtering.

The use of the TRACE of the residual covariance matrix, previously mentioned in the literature (10,43,68), has been used very effectively, not only as a means of adapting the input disturbance covariance matrix, but also as a useful performance index for the filter/model combination. It is in this area that some future research is possible. The TRACE of the residual covariance matrix can be used to select different types of model, or to select the correct order of model, where very little a-priori knowledge is available.

The calculation of a model error compensation strategy has been introduced as an important step in adaptive filtering. The model error compensation strategy effectively changes the model after every so many measurements and thus improves the performance of the filter while allowing parameter estimation to

be performed. Furthermore the compensation function , although not necessarily representing any real variable, does indicate the source and type of model error that exists. This information can be very helpful to any modelling work that seeks to improve the mathematical model used.

The concepts introduced in the adaptive filter have parameters associated with them. The results, while demonstrating the usefulness of the adaptive filter when applied to models containing serious errors and omissions, indicate values for these parameters which seem to be quite general. Further work is needed in using the adaptive filter on-line with some real process. In this connection it must be noted that the adaptive filter increased the time of estimation over the standard filter by approximately 8%. Bearing in mind that the programmes were written in BASIC for simulation work, this would indicate a very small time penalty to pay for the increased accuracy of the adaptive filter. Also the adaptive filter can be used with much simpler models than the standard filter and yet provide estimates of similar accuracy. From this point of view the adaptive filter could be faster to use by decreasing the time for the prediction steps, while maintaining the same general level of accuracy.

CHAPTER (7).

APPENDICES.

(1). Abstract Vector Spaces, Inner Products, and Orthogonality.

(2). The Method of Collocation.

(3). Minimum Variance Estimation.

(4). The Modelling of Uncertain Parameters.

(5). The Calculation of Orthonormal Polynomials.

(6). The Calculation of the Matrix Differential Operator.

(7). The Calculation of Matrix M and its Inverse.

(8). Programme to Solve Linear Time Invariant Differential Equations.

(9). The Calculation of Forcing Functions by Gaussian Quadrature.

(10). Computer Programme for Collocation.

(11). Programme for State Variable and Parameter Estimation.

(12). Programme for Adaptive Estimation.

(13). Observability.

(14). Subroutines of Matrix Operations.

Appendix (1).

Abstract Vector Spaces, Inner Products, and Orthogonality (71).

An abstract vector space (V) has the following properties.

1. If f and g are in V so is f+g.
2. $f + (g+h) = (f+g) + h$: $f,g,h \in V$.
3. There is an element $o \in V$ such that $h+o=h$ for all $h \in V$.
4. For each $h \in V$ there is an element $-h \in V$ such that
   $h + (-h) = o$.
5. $g+h = h+g$: $g,h \in V$.
6. For each real number a : $ah \in V$.
7. $a(g+h) = ag + ah$.
8. For each real number a,b : $(a+b)h = ah + bh$.
9. $a(bh) = (ab)h$.

Examples.

1. Euclidean n-dimensional real space (En). The elements consist of sequences of n real numbers.

   $f = (a_1,.....a_n)$: $g = (b_1,.....b_n)$

2. The space of polynomial functions of order less than or equal to n (Pn). Here,

   $$f = \sum_{i=0}^{n} a_i t^i : g = \sum_{i=0}^{n} b_i t^i$$

Inner Products.

An inner product on a vector space (V) is a mapping V x V into the real line. That is a real number is associated with every pair of vectors in V with the following properties.

1. $(af,g) = a(f,g)$.
2. $(f+g,h) = (f,h) + (g,h)$.
3. $(f,g) = (g,f)$.
4. $(f,f) > o$ unless $f=o$
5. (f,g) = some real number.

Examples.

1. For the space En

$$(f,g) = \sum_{i=1}^{n} a_i \, b_i$$

2. For the space Pn

$$(f,g) = \int_{-1}^{+1} f \, g \, dt.$$

If $(f,g) = o$ then the vectors f,g, are said to be orthogonal (in the case of E_3 this means that f and g are perpendicular).

A set of vectors f_i such that if $\sum_{i=1}^{n} a_i \, f_i = o$ implies $a_i = o$ for all i; are said to be linearly independent.

In an n-dimensional vector space every set of $m>n$ vectors are linearly dependent. Therefore any arbitrary vector g can be written as;

$g = \sum_{i=1}^{n} a_i \, f_i$ for some a_i if the f_i form a linearly independent set.

A set of linearly independent vectors is called orthogonal if,

$$(f_i, f_j) = \begin{cases} 0 & i \neq j \\ b_i & i = j \end{cases}$$

An orthogonal set is called orthonormal if,

$$(f_i, f_i) = 1 \quad \text{for all } i.$$

Any vector z, not in the space (V) can be approximated by a vector g which is an element of (V);

i.e. $$z \simeq g = \sum_{i=1}^{n} a_i f_i$$

The best choice of the a_i is when the error is orthogonal to the approximation.

i.e. $(z-g, f_i) = 0 \quad \text{for all } i.$

Therefore:

$$(z, f_i) = (g, f_i)$$

$$= \left(\sum_{j=1}^{n} a_j f_j, f_i\right)$$

$$= \sum_{j=1}^{n} a_j (f_j, f_i)$$

Therefore:

$(z, f_i) = a_i (f_i, f_i)$ if the f_i are orthogonal.

and,

$(z, f_i) = a_i$ if the f_i are orthonormal.

This can be generalised by considering a vector space consisting

of sets of m vectors. i.e.

$$G = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_m \end{bmatrix} = \sum_{i=1}^{n} \begin{bmatrix} a_{1,i} \\ a_{2,i} \\ \vdots \\ a_{m,i} \end{bmatrix} f_i$$

or, in matrix terminology,

$$G = AF \quad \text{where} \quad F = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_m \end{bmatrix}$$

and

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix}$$

Now any set of m-vectors, Z, not in the space can be approximated by a vector G

$$Z \simeq G = AF$$

and the best choice of the matrix A is when,

$$(Z-G,F) = 0$$

where (G,F) is a generalised inner product defined by;

$$(G,F) = \begin{bmatrix} (g_1,f_1) & (g_1,f_2) & \cdots \\ (g_2,f_1) & & \\ \vdots & & \\ (g_m,f_1 & \cdots & \end{bmatrix}$$

Clearly,

$$(G,F) = (F,G)^T$$

If the f_i are orthonormal then,

$$(F,F) = I$$

and the best choice of A reduces to,

$$A = (Z,F).$$

Appendix (2).

The Method of Collocation (69).

An arbitrary function of (t) f(t) can be represented by an infinite series of orthogonal polynomials:

$$f(t) = \sum_{i=1}^{\infty} a_i \, p_i(t) \quad \ldots\ldots\ldots\ldots\ldots (1)$$

an n^{th} order polynomial approximation to f(t) can be written;

$$y \simeq f(t) \simeq \sum_{i=o}^{n} b_i \, p_i(t) \quad \ldots\ldots\ldots (2)$$

and because,

$$(f(t)-y, p_j(t)) = o \quad \text{for } o \leq j \leq n \text{ (appendix 1)}$$

i.e.

$$\left(\sum_{i=o}^{n} (a_i - b_i) \, p_i(t), p_j(t)\right) + \left(\sum_{i=n+1}^{\infty} a_i \, p_i(t), \, p_j(t)\right) = o$$

Therefore:

$$b_i = a_i \quad \text{for } o \leq i \leq n.$$

The error f(t)-y = e can therefore be written as,

$$e = \sum_{i=n+1}^{\infty} a_i \, p_i(t) \quad \ldots\ldots\ldots\ldots (3)$$

and the (n+1)th order approximation e_n to this is,

$$\sum_{i=o}^{n+1} c_i \, p_i(t)$$

where c_i are found via,

$$(e-e_n, p_j) = o \quad \text{for } o \leq j \leq n+1$$

i.e.

$$c_i = o \quad \text{for } o \leq i \leq n$$

and,

$$c_{n+1} = a_{n+1}$$

Therefore the zeros of e are approximately the same as the zeros of $p_{n+1}(t)$. The approximation can therefore be found by setting $y=f(t)$ at the collocation points. (i.e. at the zeros of $p_{n+1}(t)$).

Clearly this approximation is not optimal in the sense discussed in appendix (1) but only nearly optimal. As n increases so this method approaches the optimal approximation defined by,

$$a_i = (f(t), p_i(t))/(p_i(t), p_i(t)).....\text{(appendix 1)}.$$

Appendix (3).

Minimum Variance Estimation (31, 38, 43).

Let $z = Hx+v$ be an h dimensional vector measurement of the n dimensional state vector x, where H is an h x n matrix and v is an h dimensional vector of random variables with,

$$Ev = o \text{ and } Evv^T = R$$

Let $\bar{x}$ be a prediction of x with,

$$E(x-\bar{x}) = o \text{ and } E(x-\bar{x})(x-\bar{x})^T = \bar{V}$$

Now assuming,

$$E(x-\bar{x})v^T = o$$

Then,

$\hat{x} = \bar{x} + K(z-H\bar{x})$ is a minimum variance estimation of x if K is a suitable weighting matrix. K must be calculated to minimise,

$$\text{TRACE } (E(x-\hat{x})(x-\hat{x})^T) = \text{TRACE } (\hat{V})$$

Now,

$$\hat{V} = \bar{V}-KH\bar{V}-\bar{V}H^TK^T + K(R+H\bar{V}H^T)K^T$$

and TRACE $(\hat{V})$ is minimised when,

$$\text{TRACE}\left(\frac{d\hat{V}}{dK(i,j)}\right) = o \quad \text{for all i and j.}$$

Now,

$$\frac{d\hat{V}}{dK(i,j)} = K(R+H\bar{V}H^T)\,\dot{K}^T(i,j) + \dot{K}(i,j)\,(R+H\bar{V}H^T)K^T$$

$$-\dot{K}(i,j)\,H\bar{V} - \bar{V}H^T\,\dot{K}^T(i,j)$$

where,

$$\dot{K}(i,j) = \frac{dK}{dK(i,j)}$$

i.e.

$$\frac{dK(m,k)}{dK(i,j)} = \begin{cases} 0 & m \neq i,\ k \neq j. \\ 1 & m = i,\ k = j. \end{cases}$$

Therefore K must satisfy,

$$\text{TRACE}\ (\dot{K}(i,j)\ ((R+H\bar{V}H^T)\,K^T - H\bar{V}) + (K(R+H\bar{V}H^T)-\bar{V}H^T)\,\dot{K}^T(i,j)\ ) = 0$$

for all i,j

This means that,

$$(K(R+H\bar{V}H^T)-\bar{V}H^T) = 0$$

Therefore,

$$K = \bar{V}H^T(R+H\bar{V}H^T)^{-1}$$

This important result can be obtained by a large variety of methods. For a discussion of the different approaches see references (43).

Appendix (4).

The Modelling of Uncertain Parameters (43, 50).

Consider the equation;

$$\dot{x} = F_1 x + F_4 w \qquad \ldots\ldots(1)$$

where x is an n dimensional state vector and w is an ℓ dimensional vector of unknown parameters.

F_1 is an nxn matrix, and,

F_4 is an nxℓ matrix.

Equation (1) can be rewritten,

$$\begin{bmatrix} \dot{x} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} F_1 & F_4 \\ o & o \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} \qquad \ldots\ldots(2)$$

This equation can be used in the filter described in the text to produce estimates of x and w. This procedure, however, can be time consuming because the dimensionality has been increased (from n to (n+ℓ)).

To overcome this the a-priori value $\bar{w}$ of w will not be improved by the filter, but the error committed in using $\bar{w}$ will be modelled. To do this consider the formal equations of the augmented system (2).

The prediction error covariance matrix will be,

$$\bar{V} = \begin{bmatrix} \bar{V}_{x'} & \bar{V}_{x,w} \\ \bar{V}^{T}_{x,w'} & \bar{V}_{w} \end{bmatrix}$$

Where $\bar{V}_x = E(x-\bar{x})(x-\bar{x})^T$

and $\bar{V}_{x,w} = E(x-\bar{x})(w-\bar{w})^T$

and $\bar{V}_w = E(w-\bar{w})(w-\bar{w})^T$

Now $\bar{w}$ is not to be improved by the filter so $\bar{V}_w$ remains constant. Therefore:

$$\begin{bmatrix} \bar{V}_x & \bar{V}_{x,w} \\ \bar{V}^{T}_{x,w} & \bar{V}w \end{bmatrix} = \begin{bmatrix} \emptyset & \Delta F_4 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{V}_x & \hat{V}_{x,w} \\ \hat{V}^{T}_{x,w} & \bar{V}_w \end{bmatrix} \begin{bmatrix} \emptyset^T & 0 \\ F_4{}^T\Delta^T & I \end{bmatrix} \quad .....(3)$$

where, $\hat{V}_x = E(x-\hat{x})(x-\hat{x})^T$

and, $\hat{V}_{x,u} = E(x-\hat{x})(w-\bar{w})^T$

and $\emptyset$ is the fundamental matrix derived from equation (1) and Δ is the integral of $\emptyset$ over the time increment.

Therefore, from (3),

$$\bar{V}_x = \phi \hat{V}_x \phi + \Delta F_4 \hat{V}^T_{x,w} \phi^T + \phi \hat{V}_{x,w} F_4^T \Delta^T + \Delta F_4 \bar{V}_w F_4^T \Delta^T \quad \ldots\ldots(4)$$

and, $$\bar{V}_{x,w} = \phi \hat{V}_{x,w} + \Delta F_4 \bar{V}_w \quad \ldots\ldots(5)$$

For the next time increment,

$$\hat{V} = (I-KH)\bar{V}$$

i.e.

$$\begin{bmatrix} \hat{V}_x & \hat{V}_{x,w} \\ \hat{V}^T_{x,w} & \bar{V}_w \end{bmatrix} = \begin{bmatrix} I-K_x H_x & 0 \\ -K_{x,w} H_x & I \end{bmatrix} \begin{bmatrix} \bar{V}_x & \bar{V}_{x,w} \\ \bar{V}^T_{x,w} & \bar{V}_w \end{bmatrix}$$

where, $$K_x = \bar{V}_x H^T_x (H_x \bar{V}_x H^T_x + R)^{-1}$$

and,

$$K_{x,w} = \bar{V}^T_{x,w} H^T_x (H_x \bar{V}_x H^T_x + R)^{-1}$$

and H_x is found from.

$$H = (H_x \quad 0)$$

Therefore: $$\hat{V}_x = (I-K_x H_x)\bar{V}_x \quad \ldots\ldots(6)$$

and, $$\hat{V}_{x,w} = (I-K_x H_x)\bar{V}_{x,w} \quad \ldots\ldots(7)$$

Now defining

$$\bar{C} = \bar{V}_{x.w} (\Delta \bar{F}_4)^T$$

$$\hat{C} = \hat{V}_{x,w} (\Delta F_4)^T$$

then the resulting modification to the filter can be written as follows:

Prediction.

For the range $t_i \leq t \leq t_{i+1}$ via

$$\dot{\bar{x}} = F_1 \bar{x} + F_4 \bar{w}$$

with $\bar{x}(t_i) = \hat{x}(t_i)$

$$\bar{V}_x(t_{i+1}) = \phi \hat{V}_x(t_i) \phi^T + \Delta F_4 \bar{V}_w F_4^T \Delta^T + \phi \hat{C}(t_i) + \hat{C}^T(t_i) \phi^T$$

$$\bar{C}(t_{i+1}) = \phi \hat{C}(t_i) + \Delta F_4 \bar{V}_w F_4^T \Delta^T$$

Estimation.

$$K_x = \bar{V}_x(t_{i+1}) H_x^T (H_x \bar{V}_x(t_{i+1}) H_x^T + R)^{-1}$$

$$\hat{x}(t) = \bar{x}(t) + K_x (z(t) - H_x \bar{x}(t)) \qquad t_i \leq t \leq t_{i+1}$$

$$\hat{V}_x(t_{i+1}) = (I - K_x H_x) \bar{V}_x(t_{i+1})$$

$$\hat{C}(t_{i+1}) = (I-K_x H_x)\,\bar{C}(t_{i+1})$$

These equations show how the filter is to be modified in order to account for the errors in $\bar{w}$ without having to improve on this a-priori prediction. Clearly this modified filter will no longer be optimal, in the sense that the maximum information contained in the measurements is being utilised, but on the other hand it will process the data a lot more quickly, without impairing the knowledge of the errors in $\hat{x}$.

Appendix (5).

The Calculation of Orthonormal Polynomials.

Orthonormal polynomials p_i are defined by the relationship,

$$\int_{-1}^{+1} p_i p_j dt = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

A set of orthogonal polynomials q_i are defined by,

$$\int_{-1}^{+1} q_i q_j dt = \begin{cases} 0 & i \neq j \\ b_i & i = j \end{cases}$$

Now defining $q_o = 1$ and $q_1 = t + a_{1,o}$; then $a_{1,o}$ can be calculated from

$$\int_{-1}^{+1} (t + a_{1,o}) dt = 0$$

i.e. $$a_{1,o} = 0$$

The coefficients of q_2 can then be calculated from,

$$\int_{-1}^{+1} q_o q_2 dt = 0$$

$$\int_{-1}^{+1} q_1 q_2 dt = 0$$

and so on up to any order (m) required.

The b_i can then be calculated from,

$$b_i = \int_{-1}^{+1} q_i^2 \, dt$$

and then the normalised orthogonal (orthonormal) polynomials p_i calculated from,

$$p_i = q_i / \sqrt{b_i}$$

The polynomials thus calculated are known as normalised Legendre polynomials and are listed below up to the 9^{th}-order.

$$p_o = 0.707$$

$$p_1 = 1.22t$$

$$p_2 = -0.791 + 2.372t^2$$

$$p_3 = -2.806t + 4.677t^3$$

$$p_4 = 0.795 - 7.955t^2 + 9.281t^4$$

$$p_5 = 4.397t - 20.521t^2 + 18.469t^5$$

$$p_6 = -0.797 + 16.732t^2 - 50.195t^4 + 36.81t^6$$

$$p_7 = -5.989t + 53.9t^3 - 118.6t^5 + 73.41t^7$$

$$p_8 = 0.799 - 28.745t^2 + 158.1t^4 - 274.0t^6 + 146.8t^8$$

$$p_9 = 7.589t - 111.3t^3 + 433.9t^5 - 619.8t^7 + 292.7t^9$$

Appendix (6).

The Calculation of the Matrix Differential Operator.

The matrix differential operator defined on the basis of the orthonormal Legendre polynomials p_i (appendix 5) is calculated from the relationship,

$$DP = \dot{P}$$

where,

$$P = \begin{bmatrix} p_o \\ p_1 \\ \vdots \\ p_m \end{bmatrix} \quad \text{and} \quad \dot{P} = \begin{bmatrix} dp_o/dt \\ \vdots \\ dp_m/dt \end{bmatrix}$$

This means:

$$(DP,P) = (\dot{P},P)$$

Therefore

$$D(P,P) = (\dot{P},P)$$

Therefore

$$D = (\dot{P},P) = \int_{-1}^{+1} \dot{P},P^T \, dt$$

Therefore

$$D = \int_{-1}^{+1} \begin{bmatrix} \dot{p}_o p_o & \dot{p}_o p_1 & \cdots \\ \vdots & \vdots & \\ \dot{p}_m p_o & \dot{p}_m p_1 & \cdots \end{bmatrix} dt$$

or, writing $P = LT$ where L is the coefficient matrix of the normalised Legendre polynomials and,

$$T = \begin{bmatrix} 1 \\ t \\ t^2 \\ t^3 \\ \vdots \\ t^m \end{bmatrix}$$

Then,

$$D = \int_{-1}^{+1} \dot{L}T\ T^T\ L^T\ dt$$

Therefore,

$$D = L \int_{-1}^{+1} \begin{bmatrix} o \\ 1 \\ 2t \\ 3t^2 \\ 4t^3 \\ \vdots \\ mt^{m-1} \end{bmatrix} (1\ t\ t^2\ \ldots t^m)\, dt\ L^T$$

Therefore,

$$D = L \int_{-1}^{+1} \begin{bmatrix} o & o & o & o \\ 1 & t & t^2 & t^3 \\ 2t & 2t^2 & 2t^3 & 2t^4 \\ 3t^2 & 3t^3 & 3t^4 & 3t^5 \\ & & \text{etc.} & \end{bmatrix} \text{etc.}\ dt\ L^T$$

Therefore,

$$D = L \begin{bmatrix} o & o & o & o \\ t & t^2/2 & t^3/3 & t^4/4 \\ 2t^2/2 & 2t^3/3 & 2t^4/4 & 2t^5/5 \\ 3t^3/3 & 3t^4/4 & 3t^5/5 & 3t^6/6 \\ & & \text{etc} & \end{bmatrix}_{-1}^{+1} \text{etc}\ L^T$$

Therefore,

$$D = L \begin{bmatrix} o & o & o & o \\ 2 & o & 2/3 & o \\ o & 4/3 & o & 4/5 \\ 2 & o & 6/5 & o \\ & & \text{etc} & \end{bmatrix} \text{etc}\ L^T$$

The matrix D defined above is easily computed and the result is:

$$\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1.732 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 3.873 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2.646 & 0 & 5.916 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 5.196 & 0 & 7.937 & 0 & 0 & 0 & 0 & 0 & 0 \\
3.317 & 0 & 7.416 & 0 & 9.950 & 0 & 0 & 0 & 0 & 0 \\
0 & 6.246 & 0 & 9.540 & 0 & 11.956 & 0 & 0 & 0 & 0 \\
3.872 & 0 & 8.659 & 0 & 11.616 & 0 & 13.959 & 0 & 0 & 0 \\
0 & 7.151 & 0 & 10.923 & 0 & 13.691 & 0 & 16.0 & 0 & 0 \\
4.357 & 0 & 9.741 & 0 & 13.065 & 0 & 15.703 & 0 & 17.991 & 0
\end{bmatrix}$$

Appendix (7).

The Calculation of Matrix M and Its Inverse.

The matrix M required for the solution of linear stationary differential equations is defined by:

$$M = (P(t_o)\ DP(t_o)\ D^2P(t_o)\ \text{etc})$$

and is quite simply calculated and inverted using Gaussian Elimination.

The result to four decimal places being:

1.414	0	0	0	0	0	0	0	0	0
1.414	0.816	0	0	0	0	0	0	0	0
0.943	0.816	0.211	0	0	0	0	0	0	0
0.471	0.490	0.211	0.0356	0	0	0	0	0	0
0.189	0.218	0.120	0.0356	0.0045	0	0	0	0	0
0.0623	0.0777	0.0502	0.0198	0.0045	0.0005	0	0	0	0
0.0179	0.0233	0.0167	0.0079	0.0024	0.0005	0	0	0	0
0.0045	0.0060	0.0047	0.0025	0.0010	0.0002	0	0	0	0
0.0010	0.0014	0.0011	0.0007	0.0003	0.0001	0	0	0	0
0.0002	0.0003	0.0002	0.0002	0.0001	0	0	0	0	0

Appendix (8).

Programme To Solve Linear Time Invariant Differential Equations.

```
1   DIM X(9,9),M(9,9),B(9,9),W(9,9),E(9,9),F(9,9),A(9),Z(9,9)
2   DIM Y(9),S(9,9)
3   A4=0:R1=0
5   FOR I=0,7: FOR J=0,7: READ W(J,I): NEXT J: NEXT I
10  FOR I=0,9: FOR J=0,9:E(I,J)=0:F(I,J)=0: NEXT J: FOR J=0,I
11  READ E(I,J): NEXT J: NEXT I
20  FOR I=0,9: FOR J=0,I: READ F(I,J): NEXT J: NEXT I
21  PRINT "INPUT ORDER": INPUT N
22  PRINT "INPUT MATRIX": FOR I=0,N-1: FOR J=0,N-1: INPUT M(I,J)
23  NEXT J: NEXT I
24  PRINT "INPUT INITIAL CONDS": FOR I=0,N-1: INPUT Y(I): NEXT I
25  PRINT "INPUT FORCING FNS": FOR I=0,7: FOR J=0,N-1
26  INPUT B(J,I): NEXT J: NEXT I
27  CALL (4,B(0,0),W(0,0),X(0,0),N,8,8)
30  FOR I=0,N-1
35  Z(I,0)=Y(I)
40  NEXT I
45  FOR I=1,9
50  CALL (4,X(0,0),F(0,I-1),A(0),N,10,1)
55  CALL (4,M(0,0),Z(0,I-1),Z(0,I),N,N,1)
60  CALL (3,Z(0,I),A(0),Z(0,I),N,1,1,1)
65  NEXT I
70  CALL (4,Z(0,0),E(0,0),S(0,0),N,10,10)
75  FOR I=0,N-1: FOR J=0,9: PRINT S(I,J);: NEXT J: PRINT : NEXT I
90  DATA .715806E-01,.157247,.221827,.256454,.256454,.221827,.157247
91  DATA .715806E-01,-.119057,-.216979,-.201916,-.814775E-01,.814775E-01
92  DATA .201916,.216979,.119057,.14137,.15894,-.425197E-01,-.257779
93  DATA -.257779,-.425197E-01,.15894,.14137,-.146471,-.287387E-01,.2499
94  DATA .17622,-.17622,-.24969,.287387E-01,.146471,.136853,-.114497
95  DATA -.217593,.195246,.195246,-.217593,-.114497,.136853
96  DATA -.114627,.210343,-.228428E-01,-.247992,.247992,.228428E-01
97  DATA -.21034,.114627,.82346E-01,-.219194,.241867,-.104888,-.104888
98  DATA .241867,-.219194,.82346E-01,-.429916E-01,.137932,-.230626
99  DATA .286505,-.286505,.230626,-.137933,.429907E-01
110 FOR I=0,9: FOR J=0,9: READ B(I,J): NEXT J: NEXT I
112 FOR I1=-1,1,.5
113 PRINT
115 FOR J=0,9:Y(J)=I1↑J
116 NEXT J
118 CALL (4,B(0,0),Y(0),A(0),10,10,1)
120 CALL (4,S(0,0),A(0),Y(0),N,10,1)
121 FOR I=0,N-1: PRINT Y(I);: NEXT I
122 NEXT I1
150 DATA 1.41421,1.4142,.816493,.942782,.816488,.210819
151 DATA .471373,.489883,.210817,.356346E-01,.188533,.217714
152 DATA .120464,.356343E-01,.448956E-02,.628334E-01
153 DATA .777455E-01,.501901E-01,.197963E-01,.448952E-02
154 DATA .451217E-03,.17946E-01,.23317E-01,.167275E-01,.791797E-02
155 DATA .244876E-02,.451213E-03,.377319E-04,.448326E-02,.60419E-02
156 DATA .468213E-02,.25189E-02,.95223E-03,.242955E-03
157 DATA .377317E-04,.270303E-05,.994838E-03,.137943E-02
158 DATA .113423E-02,.671454E-03,.292942E-03,.92548E-04
159 DATA .201232E-04,.270301E-05,.168949E-06
160 DATA .198406E-03,.281513E-03,.242688E-03,.154823E-03,.752985E-04
161 DATA .2776E-04,.754581E-05,.143103E-05,.168949E-06,.939055E-08
170 DATA .707109,-1.22474,1.22475,1.58115,-4.74338,4.74341
171 DATA -1.87083,11.2251,-28.0624,28.0626,2.12131,-21.2131
172 DATA 95.4601,-222.737,222.739,-2.34516,35.1781,-246.246
173 DATA 984.995,-2216.21,2216.23,2.54976,-53.5411,535.412
174 DATA -3212.44,.120468E05,-.265026E05,.265028E05,-2.73809,76.6612
175 DATA -1034.86,8623.79,-.474301E05,.17075E06,-.369953E06,.369955E06
176 DATA 2.91915,-105.107,1839.5,-.202347E05,.151765E06,-.78918E06
177 DATA .276218E07,-.59189E07,.59189E07,-3.08085,138.758,-3055.17
178 DATA .427964E05,-.417408E06,.292255E07,-.146147E08
179 DATA .501118E08,-.10649E09,.10649E09
180 DATA .707109,0,0,0,0,0,0,0,0,0,0,1.22474,0,0,0,0,0,0,0,0
181 DATA -.790561,0,2.37171,0,0,0,0,0,0,0,0,-2.80624,0,4.67707
182 DATA 0,0,0,0,0,0,.795492,0,-7.95492,0,9.28074,0,0,0,0,0
183 DATA 0,4.39726,0,-20.5206,0,18.4685,0,0,0,0,-.796741,0
184 DATA 16.7316,0,-50.1947,0,36.8096,0,0,0,0,-5.98871,0,53.899
185 DATA 0,-118.578,0,73.4058,0,0,.798552,0,-28.7454,0,158.091
186 DATA 0,-274.015,0,146.79,0,0,7.58885,0,-111.283,0,433.933
187 DATA 0,-619.82,0,292.662
```

Appendix (9).

The Calculation of Forcing Functions by Gaussian Quadrature.

The 8-point Gaussian weights and abscissae are given below.

γ_i	d_i
$\pm$ .96028	.10122
$\pm$ .79666	.22238
$\pm$ .52553	.31370
$\pm$ .18343	.36268

As described in the text (section 4.1) the coefficient matrix for the polynomial approximation to the following functions is:

$$(u,P) = (u(\gamma_1)\ u(\gamma_2)\ \ldots) \begin{bmatrix} d_1\ P^T(\gamma_1) \\ d_2\ P^T(\gamma_2) \\ \vdots \end{bmatrix}$$

where the $u(\gamma_i)$ are the values of the forcing functions at the Gaussian abscissa. Clearly the matrix,

$$\begin{bmatrix} d_1\ P^T(\gamma_1) \\ d_2\ P^T(\gamma_2) \\ \vdots \\ d_8\ P^T(\gamma_8) \end{bmatrix}$$

depends only on the polynomials P and the Gaussian weights and abscissae; d_i, γ_i.

This matrix is therefore independent of the differential equation being solved. Therefore this matrix can be precomputed and stored in the programme (appendix 8) for solving differential equations.

The result of the calculation of the value of $\begin{bmatrix} d_1 & P^T(\gamma_1) \\ \vdots & \vdots \\ d_8 & P^T(\gamma_8) \end{bmatrix}$ is:

$$\begin{bmatrix}
0.0716 & 0.157 & 0.222 & 0.256 & 0.256 & 0.222 & 0.157 & 0.0716 \\
-0.119 & -0.217 & -0.202 & -0.0815 & -0.0815 & -0.202 & -0.217 & -0.119 \\
0.141 & 0.159 & -0.0425 & -0.258 & -0.258 & -0.0425 & 0.159 & 0.141 \\
-0.146 & -0.0287 & 0.250 & 0.176 & -0.176 & -0.250 & -0.0287 & 0.146 \\
0.137 & -0.114 & -0.218 & 0.195 & 0.195 & -0.218 & -0.114 & 0.137 \\
-0.115 & 0.210 & -0.0228 & -0.248 & 0.248 & 0.0228 & -0.210 & 0.115 \\
0.0823 & -0.219 & 0.242 & -0.105 & -0.105 & 0.242 & -0.219 & 0.0823 \\
-0.0430 & 0.138 & -0.231 & 0.287 & -0.287 & 0.231 & -0.138 & 0.0430
\end{bmatrix}$$

Appendix (10).

Computer Programmes for Collocation.

```
10  DIM X(9),Y(9,9),Q(9),H(9),B(9),V(9,9),A(9,9),W(9,9)
15""FOR I=0,9:"FOR J=0,9:V(I,J)=0: NEXT J: NEXT I
20  INPUT A,B,N
25  FOR I=0,N-1: INPUT H(I),Q(I): NEXT I
30  O1=0
40  CALL (1,O1,A,B,X(0))
41  PRINT : PRINT "ORDER",O1: PRINT
45 CALL(4,X(0),O1,Y(0,0),V(0,0),Q(0),H(0),N1,0)
50  FOR K=0,N-1
60  FOR I=0,O1
90  CALL (2,Y(0,0),X(0),Q(0),A(0,0),H(0),B(0),N,O1,K,I)
100  NEXT I
110  CALL (3,O1,N,V(0,0),R1,K)
115 CALL(4,X(0),O1,Y(0,0),V(0,0),Q(0),H(0),K,K)
120  NEXT K
130  FOR I=0,O1
140  FOR J=0,N-1
150  IF ABS(V(I,J)-W(I,J))-.1E-03,160,160,200
160  NEXT J
170  NEXT I
180  O1=O1+1
190  GOTO 40
200  FOR J=0,N-1: PRINT : FOR I=0,O1
210  W(I,J)=V(I,J)
220  PRINT V(I,J);
230 NEXT I:NEXT J
231 PRINT
240 GOTO 50
```

```
      SUBROUTINE COLNA (RO1,A,B,X)
      DIMENSION T(10),X(10)
      COMMON/C1/P(10,10),F(10,10)
      NORD=INT(RO1+0.5)+1
      DO 1 I=1,NORD
      T(I)=(2.*FLOAT(I-1)+1.)*3.14159/(2.*FLOAT(NORD))
      X(I)=0.5*(A+B+(B-A)*COS(T(I)))
    1 CONTINUE
      DO 2 J=1,NORD
      P(1,J)=1
      F(1,J)=0
      P(2,J)=(2.*X(J)-B-A)/(B-A)
      F(2,J)=2./(B-A)
      IF(NORD-3)2,4,4
 4    DO 3 I=3,NORD
      P(I,J)=2.*P(2,J)*P(I-1,J)-P(I-2,J)
      F(I,J)=2.*P(2,J)*F(I-1,J)+2.*F(2,J)*P(I-1,J)-F(I-2,J)
 3    CONTINUE
 2    CONTINUE
      RETURN
      END
$0
```

```
      SUBROUTINE COLNB(Y,X,Q,A,H,B,RN,O1,RK,RI)
      DIMENSION X(10),Q(10),A(10,10),H(10),B(10),Y(10,10)
      COMMON/C1/P(10,10),F(10,10)
      COMMON/C2/RM(10,10),D(10)
      NORD=INT(O1+.5)+1
      K=INT(RK+.5)+1
      N=INT(RN+.5)
      I=INT(RI+.5)+1
      DO 1 J=1,NORD
      RM(I,J)=(X(I)-Q(K))*F(J,I)-(A(K,K)*(X(I)-Q(K))-1.)*P(J,I)
      D(I)=0.0
      DO 3 L=1,N
      IF(L-K)4,3,4
 4    D(I)=A(K,L)*Y(L,I)+D(I)
 3    CONTINUE
      D(I)=D(I)+B(K)+A(K,K)*H(K)
 1    CONTINUE
      RETURN
      END
$0
```

```
      SUBROUTINE COLNC(O1,RN,V,RR,RK)
      DIMENSION V(10,10),B(10,11)
      COMMON/C2/RM(10,10),D(10)
      N=INT(O1+.5)+1
      K7=INT(RK+.5)+1
```

Appendix (11).

Programme for State Variable and Parameter Estimation.

```
1   DIM Z(9,19),M(9,9),B(9,1),E(9,9),Y(9),S(9,9),R(9)
2   DIM L(9,9),H(9,9),P(9,9),Q(9,9),G(9,9),V(9,9),N(9,9)
3   DIM X(9,9)
4   INPUT Z1,K7:N5=RND(Z1)
5   FOR I=0,9: READ R(I): NEXT I
6   A8=1
10   FOR I=0,9: FOR J=0,9:E(I,J)=0: NEXT J: FOR J=0,I
11   READ E(I,J): NEXT J: NEXT I
20   INPUT N
24   PRINT "INPUT INITIAL CONDS": FOR I=0,N-1: INPUT Y(I): NEXT I
25   GOSUB 200
26   CALL (4,H(0,0),Y(0),B(0,0),N9,N,1)
35   CALL (3,Y(0),Y(0),Z(0,0),N,1,1,0)
41   GOSUB 45
42   PRINT : PRINT "PREDICTIONS"
43   GOSUB 112
44   IF A8,125,123,123
45   FOR I=1,9
55   CALL (4,M(0,0),Z(0,I-1),Z(0,I),N,N,1)
65   NEXT I
70   CALL (4,Z(0,0),E(0,0),S(0,0),N,10,10)
80   RETURN
90   DATA .707109,1.22474,1.58115,1.87083,2.12131,2.34516
91   DATA 2.54976,2.73809,2.91915,3.08085
103   J1=0
105   CALL (3,D(0,0),D(0,0),D(0,0),10,10,0,0)
106   FOR I=0,9: IF I=0 GOTO 108
107   FOR J=J1,I-1,2: READ D(I,J): NEXT J:J1=J1+(-1)↑(I+1)
108   NEXT I
111   GOTO 21
112   CALL (4,S(0,0),R(0),Y(0),N,10,1)
113   FOR I=0,N-1: PRINT Y(I);: NEXT I
114   PRINT : RETURN
123   PRINT "INPUT MEASUREMENTS":N4=N9:S9=-1: GOSUB 450
125   GOSUB 248
126   PRINT : PRINT "ESTIMATES"
127   GOSUB 112
128   GOSUB 212
129   GOTO 26
150   DATA 1.41421,1.4142,.816493,.942782,.816488,.210819
151   DATA .471373,.489883,.210817,.356346E-01,.188533,.217714
152   DATA .120464,.356343E-01,.448956E-02,.628334E-01
153   DATA .777455E-01,.501901E-01,.197963E-01,.448952E-02
154   DATA .451217E-03,.17946E-01,.23317E-01,.167275E-01,.791797E-02
155   DATA .244876E-02,.451213E-03,.377319E-04,.448326E-02,.60419E-02
156   DATA .468213E-02,.25189E-02,.95223E-03,.242955E-03
157   DATA .377317E-04,.270303E-05,.994838E-03,.137943E-02
158   DATA .113423E-02,.671454E-03,.292942E-03,.92548E-04
```

```
463   RETURN
500   END
```

```
202  FOR J=0,N-1: INPUT H(I,J): NEXT J: NEXT I
205  CALL (3,P(0,0),P(0,0),P(0,0),N9,N9,0,0)
206  PRINT "INPUT MEAS.ERROR VAR.": FOR I=0,N9-1: INPUT P(I,I): NEXT I
207  CALL (3,V(0,0),V(0,0),V(0,0),N,N,0,0)
208  PRINT "INPUT INITIAL ERROR VAR.": FOR I=0,N-1: INPUT V(I,I): NEXT I
209  CALL (3,X(0,0),X(0,0),X(0,0),N,N,0,0)
210  PRINT "INPUT FORCING VAR.": FOR I=0,N-1
211  INPUT X(I,I): NEXT I
212  CALL (3,M(0,0),M(0,0),M(0,0),N,N,0,0)
213  GOSUB 460
214  FOR J=0,N-1: FOR J1=0,N-1: IF M(J,J1)><0 GOTO 215: NEXT J1: NEXT J
215  CALL (3,Z(0,0),Z(0,0),Z(0,0),N,N,0,0)
216  FOR I=0,N-1:Z(I,I)=1: NEXT I
217  C7=1:C8=1:I=1:C6=9
218  CALL (3,G(0,0),G(0,0),G(0,0),N,N,0,0)
220  CALL (3,Z(0,0),G(0,0),G(0,0),N,N,1/C7,1)
222  IF ABS(C6-G(J,J1))-.1E-02,230,230,224
224  CALL (4,Z(0,0),M(0,0),Z(0,N),N,N,N)
225  CALL (3,Z(0,N),Z(0,N),Z(0,0),N,N,2,0)
226  I=I+1:C7=C8:C8=C8*I:C6=G(J,J1): GOTO 220
230  CALL (2,H(0,0),L(0,0),N9,N)
231  CALL (4,V(0,0),L(0,0),Q(0,0),N,N,N9)
232  CALL (4,H(0,0),Q(0,0),L(0,0),N9,N,N9)
233  CALL (3,P(0,0),L(0,0),Q(0,0),N9,N9,1/3,1/3)
234  FOR I=0,N9-1: FOR J=0,N9-1: PRINT Q(I,J);: NEXT J: PRINT : NEXT I
235  CALL (2,G(0,0),Z(0,0),N,N)
236  CALL (4,G(0,0),V(0,0),L(0,0),N,N,N)
237  CALL (4,L(0,0),Z(0,0),V(0,0),N,N,N)
238  CALL (3,V(0,0),X(0,0),V(0,0),N,N,1,1)
239  CALL (2,H(0,0),Z(0,0),N9,N)
240  CALL (4,V(0,0),Z(0,0),L(0,0),N,N,N9)
241  CALL (4,H(0,0),L(0,0),Z(0,0),N9,N,N9)
242  CALL (3,Z(0,0),Q(0,0),Z(0,0),N9,N9,1,1)
243  R1=-1:N8=N9: CALL (1,Z(0,0),Z(0,0),N8,R1)
244  IF R1>0 THEN  PRINT "NO INVERSE 243": GOTO 500
245  CALL (4,L(0,0),Z(0,N9),N(0,0),N,N9,N9)
247  RETURN
248  CALL (4,H(0,0),S(0,0),L(0,0),N9,N,10)
249  CALL (3,B(0,0),L(0,0),Z(0,0),N9,2,1,-1)
250  CALL (3,L(0,2),L(0,2),Z(0,2),N9,8,0,-1)
251  CALL (4,N(0,0),Z(0,0),L(0,0),N,N9,10)
252  CALL (3,S(0,0),L(0,0),S(0,0),N,10,1,1)
253  CALL (4,N(0,0),H(0,0),L(0,0),N,N9,N)
254  CALL (3,Z(0,0),Z(0,0),Z(0,0),N,N,0,0)
255  FOR I=0,N-1:Z(I,I)=1: NEXT I
256  CALL (3,L(0,0),Z(0,0),L(0,0),N,N,1,-1)
257  CALL (2,L(0,0),Z(0,0),N,N)
258  CALL (4,L(0,0),V(0,0),Z(0,N),N,N,N)
259  CALL (4,Z(0,N),Z(0,0),V(0,0),N,N,N)
260  CALL (2,N(0,0),L(0,0),N,N9)
261  CALL (4,N(0,0),Q(0,0),Z(0,0),N,N9,N9)
262  CALL (4,Z(0,0),L(0,0),Z(0,N),N,N9,N)
263  CALL (3,V(0,0),Z(0,N),V(0,0),N,N,1,1)
265  RETURN
400  DATA 1.73205,3.87297,2.64576,5.91611,5.19611,7.93723
401  DATA 3.31657,7.41618,9.94989,6.24551,9.53982,11.9585
402  DATA 3.87227,8.65817,11.6159,13.9591,7.15061,10.9234
403  DATA 13.6908,15.9991,4.35702,9.74061,13.0653,15.7034,17.9915
450  FOR I=0,N4-1: INPUT B(I,1):N5=RND(0):B(I,1)=B(I,1)+K7*(N5-.5)
452  PRINT B(I,1):B(I,0)=(B(I,0)+B(I,1))/1.41421
453  B(I,1)=(B(I,1)-B(I,0)/1.41421)/1.22474: NEXT I
454  RETURN
```

Appendix (12).

Programme for Adaptive Estimation.

```
1  DIM Z(9,19),M(9,9),B(9,1),E(9,9),A(9),Y(9),S(9,9),R(9)
2  DIM L(9,9),H(9,9),P(9,9),Q(9,9),G(9,9),V(9,9),N(9,9)
3  DIM X(9,9),C(9,9),D(9,9)
4  INPUT Z1,K7:N5=RND(Z1)
5  FOR I=0,9: READ R(I): NEXT I
6  A8=1:P1=1:P2=0:P3=10:I9=0:J9=0:V8=-1:V9=100
7  DIM K(9,12),F(9,9)
8  L8=0:L9=4
9  Z6=.25:V2=0:Z4=0
10  FOR I=0,9: FOR J=0,9:E(I,J)=0: NEXT J: FOR J=0,I
11  READ E(I,J): NEXT J: NEXT I
12  J7=0
13  L9=6
14 REM STANDARD FILTER
21  INPUT N
23  GOSUB 200
24  FOR I=0,N-1: INPUT Y(I): NEXT I
25  GOTO 330
26  CALL (4,H(0,0),Y(0),B(0,0),N9,N,1)
35  CALL (3,Z(0,0),Z(0,0),Z(0,0),N,1,0,0)
41  GOSUB 45
44  IF A8,125,123,123
45  FOR I=1,9
55  CALL (4,M(0,0),Z(0,I-1),Z(0,I),N,N,1)
60  CALL (3,Z(0,I),A(0),Z(0,I),N,1,1,Z3)
61  Z3=0
65  NEXT I
70  CALL (4,Z(0,0),E(0,0),S(0,0),N,10,10)
80  RETURN
90  DATA .707109,1.22474,1.58115,1.87083,2.12131,2.34516
91  DATA 2.54976,2.73809,2.91915,3.08085
112  CALL (4,S(0,0),R(0),A(0),N,10,1)
113  PRINT A(0): PRINT A(1): PRINT A(2): PRINT A(3)
114  PRINT K(0,1): PRINT K(1,1): PRINT V2
115  RETURN
123  N4=N9:S9=-1: GOSUB 450
124  CALL (3,S(0,0),Y(0),S(0,0),N,1,1,1.41421)
125  GOSUB 258
127  GOSUB 112
128  CALL (3,A(0),A(0),Y(0),N,1,1,0)
129  GOTO 330
130  GOTO 25
150  DATA 1.41421,1.4142,.816493,.942782,.816488,.210819
151  DATA .471373,.489883,.210817,.356346E-01,.188533,.217714
152  DATA .120464,.356343E-01,.448956E-02,.628334E-01
153  DATA .777455E-01,.501901E-01,.197963E-01,.448952E-02
154  DATA .451217E-03,.17946E-01,.23317E-01,.167275E-01,.791797E-02
155  DATA .244876E-02,.451213E-03,.377319E-04,.448326E-02,.60419E-02
156  DATA .468213E-02,.25189E-02,.95223E-03,.242955E-03
157  DATA .377317E-04,.270303E-05,.994838E-03,.137943E-02
158  DATA .113423E-02,.671454E-03,.292942E-03,.92548E-04
159  DATA .201232E-04,.270301E-05,.168949E-06
```

```
205  CALL (3,P(0,0),P(0,0),P(0,0),N9,N9,0,0)
206  FOR I=0,N9-1: INPUT P(I,I): NEXT I
207  CALL (3,V(0,0),V(0,0),V(0,0),N,N,0,0)
208  FOR I=0,N-1: INPUT V(I,I): NEXT I
209  N6=1: CALL (3,C(0,0),C(0,0),C(0,0),N,N,0,0)
210  N6=2:D(0,0)=1:D(1,1)=1:X(0,0)=.1E-01:X(1,1)=.1E-01
211  N7=1:K(0,3)=1:F1=0
212  CALL (3,V(0,0),V(0,0),V(0,0),N,N,.1E-05,0)
213  GOTO 24
214  FOR J=0,N-1: FOR J1=0,N-1: IF M(J,J1)><0 GOTO 215: NEXT J1: NEXT J
215  CALL (3,Z(0,0),Z(0,0),Z(0,0),N,N,0,0)
216  FOR I=0,N-1:Z(I,I)=1: NEXT I
217  C7=1:C8=2:I=2:C6=100
218  CALL (3,G(0,0),G(0,0),G(0,0),N,N,0,0)
220  CALL (3,Z(0,0),G(0,0),G(0,0),N,N,2/C7,1)
222  IF ABS(C6-G(J,J1))-.1E-02,230,230,224
224  CALL (4,Z(0,0),M(0,0),Z(0,N),N,N,N)
225  CALL (3,Z(0,N),Z(0,N),Z(0,0),N,N,2,0)
226  I=I+1:C7=C8:C8=C8*I:C6=G(J,J1): GOTO 220
230  CALL (2,H(0,0),L(0,0),N9,N)
231  CALL (4,V(0,0),L(0,0),Q(0,0),N,N,N9)
232  CALL (4,H(0,0),Q(0,0),L(0,0),N9,N,N9)
233  CALL (3,P(0,0),L(0,0),Q(0,0),N9,N9,1/3,1/3)
235  CALL (4,M(0,0),G(0,0),S(0,0),N,N,N)
236  FOR I=0,N-1:S(I,I)=S(I,I)+1: NEXT I
237  CALL (2,S(0,0),Z(0,0),N,N)
238  CALL (4,S(0,0),V(0,0),L(0,0),N,N,N)
239  CALL (4,L(0,0),Z(0,0),V(0,0),N,N,N)
240  CALL (4,S(0,0),C(0,0),Z(0,0),N,N,N)
241  CALL (2,Z(0,0),L(0,0),N,N)
242  CALL (3,V(0,0),L(0,0),V(0,0),N,N,1,1)
243  CALL (4,G(0,0),K(0,3),S(0,0),N,N,N7)
244  CALL (2,S(0,0),L(0,0),N,N7)
245  CALL (4,S(0,0),L(0,0),N(0,0),N,N7,N): GOTO 610
246  CALL (3,N(0,0),N(0,0),S(0,0),N,N,F1,0)
247  CALL (3,Z(0,0),S(0,0),C(0,0),N,N,1,1)
248  CALL (3,V(0,0),C(0,0),V(0,0),N,N,1,1): GOTO 505
249  CALL (2,H(0,0),Z(0,0),N9,N)
250  CALL (4,V(0,0),Z(0,0),L(0,0),N,N,N9)
251  CALL (4,H(0,0),L(0,0),Z(0,0),N9,N,N9)
252  CALL (3,Z(0,0),Q(0,0),Z(0,0),N9,N9,1,1): GOTO 600
253  R1=-1:N8=N9: CALL (1,Z(0,0),Z(0,0),N8,R1)
254  IF R1>0 THEN  PRINT "NO INVERSE 253": GOTO 500
255  CALL (4,L(0,0),Z(0,N9),N(0,0),N,N9,N9)
256  CALL (3,L(0,0),Z(0,0),L(0,0),N,N,1,-1)
257  GOTO 26
258  CALL (4,H(0,0),S(0,0),L(0,0),N9,N,10)
259  CALL (3,B(0,0),L(0,0),Z(0,0),N9,2,1,-1)
260  CALL (3,L(0,2),L(0,2),Z(0,2),N9,8,0,-1): GOTO 550
261  CALL (4,N(0,0),Z(0,0),L(0,0),N,N9,10): GOTO 515
262  CALL (3,S(0,0),L(0,0),S(0,0),N,10,1,1)
263  CALL (4,N(0,0),H(0,0),L(0,0),N,N9,N)
264  FOR I=0,N-1:L(I,I)=L(I,I)-1: NEXT I
265  CALL (4,L(0,0),V(0,0),Z(0,0),N,N,N)
266  CALL (2,L(0,0),Z(0,N),N,N)
267  CALL (4,Z(0,0),Z(0,N),V(0,0),N,N,N)
268  CALL (4,N(0,0),Q(0,0),Z(0,0),N,N9,N9)
269  CALL (2,N(0,0),Z(0,N),N,N9)
270  CALL (4,Z(0,0),Z(0,N),Q(0,0),N,N9,N)
271  CALL (3,V(0,0),Q(0,0),V(0,0),N,N,1,1)
```

```
331  Q6=EXP(36.49-(.121E05/Y(1)))
332  M(0,0)=-Q6-.25:M(0,1)=-.121E05/(Y(1)↑2)*Q6
333  M(1,0)=-Q6*10:M(1,1)=-.25-10*.121E05*Q6/(Y(1)↑2)
334  A(0)=(-Q6-.25)*Y(0)+10*.25
335  A(1)=(-.25*Y(1))-Y(0)*Q6*10+350*.25
337  Z3=.707109
338  GOTO 344
339  CALL (4,K(0,3),K(0,1),L(0,0),N,N7,1)
340  CALL (3,A(0),L(0,0),A(0),N,1,1.41421,1)
341  CALL (4,D(0,0),K(0,2),L(0,0),N,N6,1)
342  CALL (3,A(0),L(0,0),A(0),N,1,1,1.41421)
343  GOTO 214
344  IF P1<0 THEN  GOTO 339
345  P2=P2+1: IF P2<P3 THEN  GOTO 339
346 REM ADAPTIVE PART OF FILTER
350  IF V9=<V2 THEN  GOTO 372
351  V9=V2:H(9,J9+J7)=I9:V8=I9: GOTO 372
370  CALL (3,K(0,3+J9),K(0,3+J9),K(0,3+J9),N,1,0,0)
371  CALL (3,K(0,1),K(0,1),K(0,1),N,1,0,0):K(I9,3+J9)=1: GOTO 415
372  I9=I9+1: IF J9+J7=0 THEN  GOTO 375
373  FOR J8=0,J9+J7-1: IF I9=H(9,J8) THEN  GOTO 372
374  NEXT J8
375  P2=0:N7=J9+1
376  IF I9=N THEN  GOTO 400
377  IF J9=N9 THEN P1=-1:N7=N7-1: GOTO 415
378  GOTO 370
400  IF V8<0 THEN  GOTO 410
401  CALL (3,K(0,3+J9),K(0,3+J9),K(0,3+J9),N,1,0,0)
402  K(V8,3+J9)=1:J9=J9+1:V8=-1:I9=-1
403  GOTO 372
410  N7=N7-1:P1=-1
415  CALL (3,C(0,0),C(0,0),C(0,0),N,N,0,0)
416  L8=0:Z4=0
417  CALL (3,K(0,0),K(0,0),K(0,0),N9,1,0,0)
418  CALL (3,V(0,0),V(0,0),V(0,0),N,N,0,0)
419  FOR I=0,N-1:V(I,I)=1: NEXT I
420  GOTO 339
450  FOR I=0,N4-1: INPUT B(I,1):N5=RND(0):B(I,1)=B(I,1)+K7*(N5-.5)
452  B(I,0)=(B(I,0)+B(I,1))/1.41421
453  B(I,1)=(B(I,1)-B(I,0)/1.41421)/1.22474: NEXT I
454  RETURN
500  END
505  CALL (4,G(0,0),D(0,0),S(0,0),N,N,N6)
506  CALL (2,S(0,0),L(0,0),N,N6)
507  CALL (4,S(0,0),X(0,0),N(0,0),N,N6,N6)
508  CALL (4,N(0,0),L(0,0),S(0,0),N,N6,N)
509  CALL (3,V(0,0),S(0,0),V(0,0),N,N,1,1)
510  GOTO 249
515  N=N
516  CALL (3,K(0,0),Z(0,0),K(0,0),N9,1,1-Z6,Z6)
517  L8=L8+1: IF L8<L9 GOTO 532
518  CALL (4,G(0,0),K(0,3),Z(0,0),N,N,N7)
519  CALL (4,H(0,0),Z(0,0),Z(0,N),N9,N,N7)
520  CALL (2,Z(0,N),G(0,0),N9,N7)
522  CALL (4,G(0,0),Z(0,N),Z(0,0),N7,N9,N7)
523  R1=-1
524  CALL (1,Z(0,0),Z(0,0),N7,R1)
525  IF R1>0 THEN  PRINT "NO INVERSE 524": GOTO 500
526  CALL (4,G(0,0),K(0,0),Z(0,0),N7,N9,1)
528  CALL (4,Z(0,N7),Z(0,0),G(0,0),N7,N7,1)
530  CALL (3,K(0,1),G(0,0),K(0,1),N7,1,1)
```

```
553  V1=0
554  FOR I=0,N9-1:V1=V1+.5*L(I,I): NEXT I
555  Z4=Z4+1:Z5=1/Z4
556  V2=(1-Z5)*V2+Z5*V1
570  C3=(V2-C1)/C2
571  F1=F1+Z5*C3
572  IF F1<0 THEN F1=0
573  GOTO 261
600  C1=0
601  FOR I=0,N9-1:C1=C1+Z(I,I): NEXT I
602  GOTO 253
610  CALL (2,H(0,0),L(0,0),N9,N)
611  CALL (4,N(0,0),L(0,0),S(0,0),N,N,N9)
612  CALL (4,H(0,0),S(0,0),L(0,0),N9,N,N9)
613  C2=0
614  FOR I=0,N9-1:C2=C2+L(I,I): NEXT I
615  GOTO 246
```

Appendix (13).

Observability (43, 65).

The concept of observability can be introduced in an interesting way by considering, first, a related question. That is, given two n^{th} order linear time invariant systems forced by the same inputs,

$$\dot{x}_1 = M_1 x_1 + u$$

and $$\dot{x}_2 = M_2 x_2 + u$$

then what conditions must M_1 and M_2 satisfy so that a measurement taken from both systems will always be the same if the states x_1 and x_2 are initially the same; i.e. let the measurements be,

$$z_1(t) = Hx_1(t)$$

and $$z_2(t) = Hx_2(t)$$

and let $x_1(o)=x_2(o)=x(o)$

then what relationship exists between M_1 and M_2 so that,

$$z_1(t) = z_2(t) = z(t) \qquad \text{for all } t \geq o$$

First it is obvious that,

$$x_1(t) = e^{M_1(t)} x_1(o) + \int_o^t e^{M_1(t-\tau)} u(\tau)\, d\tau$$

and, $$x_2(t) = e^{M_2(t)} x_2(o) + \int_o^t e^{M_2(t-\tau)} u(\tau)\, d\tau$$

Now defining operators $\bar{H}$, $\bar{M}_1$ and $\bar{M}_2$ such that,

$$\bar{H}(x) = Hx$$

$$x_1 = \bar{M}_1(u)$$

and, $$x_2 = \bar{M}_2(u)$$

then, $$\bar{H}x_1 = \bar{H}\,\bar{M}_1(u) = \bar{H}x_2 = \bar{H}\,\bar{M}_2(u)$$

Therefore:

$$\bar{H}\,\bar{M}_1 = \bar{H}\,\bar{M}_2$$

Therefore:

$$\bar{H}\,\bar{M}_1\,\bar{M}_2^{-1} = \bar{H}$$

In particular,

$$\bar{H}\,\bar{M}_1\,\bar{M}_2^{-1}(x_1) = \bar{H}(x_1)$$

Now,

$$\bar{M}_1\,\bar{M}_2^{-1}(x_1) = e^{M_1(t)}\,\underline{x}_1(o) + \int_o^t e^{M_1(t-\tau)}\,(D-M_2)\,x_1(\tau)\,d\tau$$

where D is the differential operator

Therefore:

$$\bar{M}_1\,\bar{M}_2^{-1}(x_1) = x_1(t) + \int_o^t e^{M_1(t-\tau)}\,(M_1-M_2)\,x_1(\tau)\,d\tau$$

Therefore:

$$\bar{H}\,\bar{M}_1\,\bar{M}_2^{-1} = \bar{H} \qquad \text{if } He^{M_1(t-\tau)}(M_1-M_2)=0 \qquad \text{for } o\leq\tau\leq t.$$

Therefore:

$$HM_1 = HM_2$$

$$HM_1^2 = HM_1\,M_2$$

$$HM_1^3 = HM_1^2\,M_2$$

etc.

or, in a different form:

$$\begin{bmatrix} H \\ HM_1 \\ HM_1^2 \\ \text{etc.} \end{bmatrix} M_1 = \begin{bmatrix} H \\ HM_1 \\ HM_1^2 \\ \text{etc.} \end{bmatrix} M_2$$

Now the concept of observability can be introduced as the condition that M_1 must be unique. This means that the rank of

$$\begin{bmatrix} H \\ HM_1 \\ HM_1^2 \\ \text{etc.} \end{bmatrix}$$

must equal n where n is the order of the system.

If this rank is less than n, then M_1 need not be equal to M_2, and more than one system will be capable of producing the same measurements z(t).

Appendix (14).

Subroutines of Matrix Operations.

```
      SUBROUTINE MATINV(A,B,P,R)
      DIMENSION A(10,10),B(10,20)
      N=INT(P+0.5)
      N1=2*N
      DO 1 I=1,N
      DO 2 J=1,N1
      IF(J-N)3,3,4
3     B(I,J)=A(I,J)
      GOTO 2
4     B(I,J)=0.0
2     CONTINUE
      K=I+N
      B(I,K)=1.0
1     CONTINUE
      DO 10 I=1,N
      II=I+1
      IF(B(I,I))20,30,20
30    DO 11 J=II,N
      IF (B(J,I))15,11,15
11    CONTINUE
      R=1.0
      RETURN
15    DO  17 K1=I,N1
      Q=B(I,K1)
      B(I,K1)=B(J,K1)
      B(J,K1)=Q
17    CONTINUE
20    DO 40 J=I,N1
      J1=N1+I-J
      B(I,J1)=B(I,J1)/B(I,I)
      DO 50 K1=1,N
      IF(K1.EQ.I)GOTO 50
      B(K1,J1)=B(K1,J1)-B(I,J1)*B(K1,I)
 50   CONTINUE
 40   CONTINUE
 10   CONTINUE
      RETURN
      END
$0
-----------------------------------------------
      SUBROUTINE MATTRS(A,B,P,Q)
      DIMENSION A(10,10),B(10,10)
      N=INT(P+.5)
      M=INT(Q+.5)
      DO 1 I=1,N
      DO 2 J=1,M
      B(J,I)=A(I,J)
 2    CONTINUE
 1    CONTINUE
      RETURN
      END
$0
-----------------------------------------------
      SUBROUTINE MARITH(A,B,C,A1,A2,A3,A4)
      DIMENSION A(10,10),B(10,10),C(10,10)
      NA1=INT(A1+.5)
      NA2=INT(A2+.5)
      DO 1 I=1,NA1
      DO 2 J=1,NA2
      C(I,J)=A3*A(I,J)+A4*B(I,J)
 2     CONTINUE
 1    CONTINUE
      RETURN
      END
$0
-----------------------------------------------
      SUBROUTINE MATMUL(A,B,C,A1,A2,A3,A4)
      DIMENSION A(10,10),B(10,10),C(10,10)
      NA2=INT(A2+.5)
      NA3=INT(A3+.5)
      NA4=INT(A4+.5)
      DO 1 I=1,NA4
      DO 2 J=1,NA2
      C(J,I)=0.0
      DO 3 K=1,NA3
      C(J,I)=C(J,I)+A1*A(J,K)*B(K,I)
 3    CONTINUE
 2    CONTINUE
 1    CONTINUE
      RETURN
      END
$0
@,
```

CHAPTER (8).

BIBLIOGRAPHY.

8. Bibliography.

1. Astrom K.J. and Bohlin T. 'Numerical Identification of Linear Dynamic Systems from Normal Operating Records'.
   Proc. IFAC symposium (Teddington) (1965).

2. Box G.E.P. and Contie G.A. 'Application of Digital Computers in the Exploration of Functional Relationships'.
   Proc. Inst. Elect. Engrs. 103, B(1956).

3. Bolanger and Kennedy. 'Control of a Closed Circuit Cement Mill'.
   Proc. IFAC symposium (Paris (June 1972).

4. Bendat J.S. 'Exact Integral Equation Solution and Synthesis for a Large Class of Optimum Time Variable Linear Filters'.
   IRE. Trans, PGIT, IT-3 (March 1957), 71-80.

5. Bendat J.S. 'Joint Estimation of Delay, Doppler and Doppler rate.'
   J. Soc. Indust. Applied Math. 4(Sept 1956), 131-151.

6. Balakrishnan A.V. 'Report on Progress in Information Theory in the U.S.A.' 1960-1963 'Prediction and Filtering'.
   IEEE. Trans, PGIT, IT-9 (Oct.1963), 237-239.

7. Sage A.P. and Melsa J.L. 'System Identification.'
   Academic Press (1971).

8 Briggs P.A.N. et. al. 'Correlation Analysis of Process Dynamics Using Pseudo-random Binary Test Perturbations.'
   Proc. Inst. Mech. Engrs. 179, H(1965), 53-67.

9. Cowley P.E.A. 'The Application of an Analog Computer to the Measurement of Process Dynamics.'
   Trans. Am. Soc. Mech. Engrs. 79, (1957), 23-37.

10. Coggan S.C. and Wilson J.A. 'Approximate Models and the Elimination of Bias in On-line Estimation of Industrial Processes.'
Symposium on On-line Computer Methods Relevant to Chem. Eng. Nottingham (Sept. 1971), 151-167.

11. Davies W.D.T. and Fishwick. 'The Synthesis of Dynamical Models of Plants and Processes.'
Proc. Inst. Mech. Engrs. 179, H(1965).

12. Deutsch R. 'Orbital Dynamics of Space Vehicles.'
Prentice - Hall Inc. (1963).

13. Deutsch R. 'Estimation Theory'.
Prentice-Hall Inc. (1965).

14. Darlington S. 'Linear Least-squares Smoothing and Prediction.'
Bell System Tech. J. 37(Sept 1958), 1221-1293.

15. Darlington S. 'Non-stationary Smoothing and Prediction Using Network Theory Concepts.'
IRE. Trans, PGIT, IT-5(May 1959), 1-13.

16. Dwyer B. 'A Feedback Transfer Function Analyser.'
Control (Dec. 1965), 676-678.

17. Fisher R.A. 'On the Mathematical Foundations of Theoretical Statistics.'
Philos. Trans, Royal Soc. London (1922), 309.

18. Florentine J.J. et. al. 'Correlation Analysis of a Heat Exchanger.'
Proc. of Joint Symposium on Instrumentation and Computation in Process Development and Plant Design. SIT, A(1969), 7-17.

19. Fisher R.A. 'Theory of Statistical Estimation.'
Proc. Camb. Philos. Soc. 22(1925), 760.

20. Fisher R.A. 'On an Absolute Criterion for Fitting Frequency Curves.'
Mess. of Maths. 41(1912) 155.

21. Friedland B. 'Least-squares Filtering and Prediction of Non-stationary Sampled Data.'
Information and Control, 1(1958), 297-313.

22. Goodman N.R. et. al. 'Frequency Response from Stationary Noise: Two Case Histories.'
Technometrics, 3, No 2 (1961).

23. Goodman T.P. and Riswick J.B. 'Determination of System Characteristics from Normal Operating Records.'
Trans. Am. Soc. Mech. Engrs. 78,(1956), 259-271.

24. Gauss K.F. 'Theory of the Motion of the Heavenly Bodies Moving About the Sun in Conic Sections.'
New York: Dover Publications Inc. (1963).

25. Hughes M.T.G. and May R. 'Estimation of Frequency Response from Random Data.'
Proc. IFAC symposium (Praque) (1967).

26. Hughes M.T.G. and Noton A.R.M. 'The Measurement of Control System Characteristics by Means of a Cross Correlator.'
Proc. Instn. Elect. Engrs. 109, B(1962), 77-83.

27. Hughes M.T.G. 'Adaptive Model Approach to System Identification.'
Instn. Elect. Engrs. 'Colloquium on Random Signal Testing.' (London) (1965).

28. Ho Y.C. and Lee R.C.K. 'Identification of Linear Dynamic Systems.'
Information and Control, 8(1965), 93-110.

29. Kitamori T. 'Application of Orthogonal Functions to the Determination of Process Dynamic Characteristics and to the Construction of Self Optimising Control Systems.'
Proc. IFAC Congress (Moscow) (1960), 82.

30. Kopp R. and Oxford R. 'System Identification for Adaptive Control Systems.'
J. AIAA. 1, No. 10 (1963).

31. Kalman R.E. 'A New Approach to Linear Filtering and Prediction Problems.'
J. Basic Eng. Trans ASME. 82, D(1960), 33-45.

32. Kalman R.E. and Bertram 'Control System Analysis and Design Via the Second Method of Lyapunor.'
J.Basic Eng. Trans ASME. 82, D(1960), 371-399.

33. Kalman R.E. 'New Methods and Results in Linear Prediction and Filtering Theory.'
Tech. Report 61-1, Research Institute for Advanced Studies, Baltimore (Nov. 1961).

34. Kalman R.E. and Bucy R.S. 'New Results in Linear Filtering and Prediction Theory.'
J. Basic Eng. Trans ASME. 83, D(1961), 95-108.

35. Kolmogorov A.N. 'Interpolation and Extrapolation of Stationary Time Series.'
Bull. Acad. Sci. (USSR) ser. Math. 5(1941), 3-14

36. Legendre A.M. 'Nouvelles Methods pour la Determination des Orbites des Cometes.'
Paris (1806).

37. Lanning J.H. and Battin R.H. 'Random Processes in Automatic Control.'
New York: McGraw-Hill Co. (1956).

38. Leonides C.T. 'Theory and Applications of Kalman Filtering.'
NATO. Advisory Group for Aerospace Research and Development. (1970).

39. Levin M.J. 'Estimation of the Characteristics of Linear Systems in the Presence of Noise.'
Doctoral Thesis, Columbia Uni. USA, (1959).

40. Lubbock J.K. 'The Optimisation of a Class of Non-linear Filters.'
Proc. Inst. Elect. Engrs. 107, C(1960).

41. Jazwinski A.H. 'Limited Memory Optimal Filtering.'
Report 67-11, Analytical Mechanics Associates. (August 1967).

42. Jazwinski A.H. and Bailie. 'Adaptive Filtering.'
Report 67-6, Analytical Mechanics Associates, (March 1967).

43. Jazwinski A.H. 'Stochastic Processes and Filtering Theory.'
Academic Press (1970).

44. Munroe A.J. 'Digital Processes for Sampled Data Systems.'
New York: J.Wiley and Sons (1962).

45. Miller K.S. and Zadeh L.A. 'Solution of an Integral Equation Occurring in the Theories of Prediction and Estimation.'
IRE. Trans, PGIT, IT-2 (1956), 72-75.

46. Murphy and Sahara. 'A Mean Weighted Square-error Criterion for Optimum Filtering.'
IRE. Trans, PGAC, AC-6 (1961), 211-216.

47. Pearson K. 'Contributions to the Mathematical Theory of Evolution.'
Philos. Trans. Royal Soc. London 185 (1894), 71.

48. Pearson K. 'Contributions to the Mathematical Theory of Evolution IV.'
Philos. Trans. Royal Soc. London 191 (1898), 229.

49. Paul R.J.A. 'Orthogonal Functional Approximation to Linear Dynamical Systems.'
Proc. Inst. Elect. Engrs. 111, (1964), 1913-1922.

50. Schmidt S.F. 'Computational Techniques in Kalman Filtering.'
NATO Advisory Group for Aerospace Research and Development. (1970).

51. Schlee et. al. 'Divergence in the Kalman Filter.'
J. AIAA. 5(1967), 1114-1120.

52. Stumper F.C. 'A Bibliography of Information Theory: Communication - Cybernetics.'
IRE. Trans, PGIT, IT-1 (1955), 35-37.

53. Shinbrot M. 'A Generalisation of a Method for the Solution of the Integral Equation Arising in Optimisation of Linear Systems.'
IRE. Trans, PGIT, IT-3 (1957), 220-224.

54. Sabrison D.J. 'Iterative Design of Optimum Filters for Non-mean-square Error Performance Criteria.'
IEEE. Trans, PGIT, IT-9 (1963), 161-167.

55. Wainstein and Zubakov. 'Extraction of Signals from Noise.'
Prentice-Hall Inc. (1962).

56. Wiener N. 'The Extrapolation, Interpolation, and Smoothing of Time Series.'
New York: J.Wiley and Sons. (1949).

57. Yaglom A.M. 'An Introduction to the Theory of Stationary Random Functions.'
Prentice-Hall Inc. (1962).

58. Zadeh L.A. 'Progress in Information Theory in U.S.A. (1957-1960) Prediction and Filtering.'
IRE. Trans, PGIT, IT- (1961), 139-144.

59. Bell D.J. 'Recent Mathematical Developments in Control.'
Proc. of a Conference held at University of Bath, (Sept 1972).

60. Goldman S.F. and Sargeant R.W.H, 'Applications of Linear Estimation Theory to Chemical Processes: A Feasibility Study.'
Chem. Eng. Sci. 26, (1971), 1535-1553.

61. Joffe B.L. and Sargeant R.W.H. 'Design of an On-line Control Scheme for a Tubular Catalytic Reactor.'
Trans. Inst. Chem. Eng. 50, (1972), 270-281.

62. Eykhoff P. 'Identification and System Parameter Estimation.'
Proc. IFAC, Symposium (The Haque) (June 1973).

63. Jazwinsk A.H. 'Adaptive Sequential Estimation with Applications.'
Automatica, 10, 2(1974), 203-208.

64. Bell and Griffin. 'Modern Control Theory.'
McGraw-Hill (1970).

65. Graupe D. 'Identification of Systems.'
Litton Educational Publishing, Inc. (1972).

66. Hamilton J.C. et. al. 'An Experimental Evaluation of Kalman Filtering.'
J. AIChe. 19, 5(1973).

67. Belanger P.R. 'Estimation of Noise Covariance Matrices for a Linear Time-varying Stochastic Process.'
Automatica, 10, 3(1974), 267-275.

68. Mehra R.K. 'On the Identification of Variances and Adaptive Kalman Filtering.'
IEEE, Trans, Aut. Control, AC-15 (1970), 175-184.

69. Kilbride-Newman R.F. 'An Investigation into the Solutions of Ordinary Differential Equations on a Digital Computer.'
M.Sc. Process Analysis and Development, University of Aston in Birmingham. (1971).

70. Rosenbrock H.H. and Storey C. 'Mathematics of Dynamical Systems'.
Nelson (1970).

71. Bryson Jnr. A.E. and Ho Y.C. 'Applied Optimal Control'.
Waltham (1969).

THE USE OF ORTHOGONAL POLYNOMIALS IN SIMULATION AND STATE ESTIMATION

R. F. Kilbride-Newman and B. Gay

Presented at

THE EUROPEAN SYMPOSIUM 'COMPUTER APPLICATION IN PROCESS DEVELOPMENT'

D.E.C.H.E.M.A. ERLANGEN, GERMANY. (April 1974)

IN SUPPORT OF THE THESIS 'STATE VARIABLE AND PARAMETER ESTIMATION'

SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY BY

ROBERT F. KILBRIDE-NEWMAN.

The use of orthogonal polynomials in simulation and state estimation

R. F. Kilbride-Newman and B. Gay, Birmingham

Introduction

At present the standard methods of solving differential equations on digital computers yield solutions in discrete form. However, solutions in continuous form have considerable advantages, and this paper describes a method of obtaining continuous solutions to stationary differential equations by utilizing the properties of orthogonal polynomials.

A method of obtaining continuous estimates of state variables and parameters from 'noisy' measurements is also described. The resulting filter is similar to the Kalman filter (1) and consequently previous theorems concerning the convergence of the estimates can be applied directly to this new filter.

1. Simulation

The mathematical description of a large number of processes can be reduced to a set of ordinary differential equations. For the purpose of this paper simulation will be taken to mean the solution of such sets of differential equations.

A set of linearly independent polynomials forms a basis for the polynomial function space. Any function in this space can be written as a linear combination of the basis vectors.

An arbitrary function (w) not in this space can be approximated by a vector (y) :

$$w \simeq y = \sum_{i=0}^{n} \alpha_i p_i$$

where p_i are the basis vectors and α_i are the coordinates of the approximation.

The best choice of the α_i occurs when the error is orthogonal to the approximation i.e.

$$(w - y, p_i) = 0 \text{ for all } i$$

where the left hand side defines an inner product for the function space thus,

$$(x,y) = \int_{-1}^{+1} x(t)\, y(t)\, dt$$

If the basis vectors are orthonormal then we have the result

$$(w, p_i) = \alpha_i$$

This result can be generalised to consider sets of functions i.e. $w = (w_1 \ldots\ldots w_m)^T$

$$p = (p_0 \ldots\ldots p_n)^T$$

then the matrix of coefficients

$$A = (w,p) = \int_{-1}^{+1} w(t)\, p(\cdot)^T\, dt.$$

defines the set of best estimates such that

$$w \simeq Ap$$

and the error is minimised.

A set of linear stationary differential equations can be written

$$\dot{x} = Mx + Nu \qquad (1)$$

where x is an m-dimensional state-vector
u is a j-dimensional vector of forcing functions
and N is an m x j matrix.

Approximating x by polynomials gives

$$x \simeq Ap$$

Defining a matrix D such that

$$Dp = dp/dt$$

and substituting into (1) gives

$$ADp = MAp + Nu \qquad (2)$$

For the differential equation to be approximated optimally we must have

$$(ADp - MAp - Nu, p) = 0$$

i.e.
$$AD = MA + (Nu, p) \qquad (3)$$

Now the inital conditions give

$$A\, p(t_0) = x_0$$

then
$$AD\, p(t_0) = MAp(t_0) + (Nu, p)\, p(t_0)$$
$$AD^2 p(t_0) = MADp(t_0) + (Nu, p)\, Dp(t_0)$$
$$AD^3 p(t_0) = MAD^2 p(t_0) + (Nu, p)\, D^2 p(t_0)$$

Defining the matrix $E = (p(t_o) \vdots Dp(t_o) \vdots \quad \vdots D^n p(t_o))$

and the matrix $F = (x_o \vdots Mx_o + (Nu, p)\, p(t_o) \vdots$ etc.)

then $A = F E^{-1}$ from equation (3)

and $x = Ap$ gives a set of polynomials that satisfy the differential equation in a least squares sense over a chosen range of the independent variable.

Now the exact solution may be written as an infinite series thus:-

$x = \sum_{i=0}^{\infty} B_i p_i$ where the B_i are m-dimensional vectors of coefficients

and because of the orthogonality of the p_i the error of the polynomial approximation can be estimated as

$$e \simeq B_{n+1} p_{n+1}$$

therefore the total squared error of the approximation

$$e_{tot} = \int_{-1}^{+1} e^T e \, dt \simeq \| B_{n+1} \|^2$$

and if it is assumed that the series is monotonically convergent after some number of terms less than n we have

$$e_{tot} \simeq \| B_{n+1} \|^2 < \| An \|^2$$ where An is the nth column of A

If $\| An \| \neq (e_s)^{\frac{1}{2}}$

where e_s is some preset error bound, then define k_s such that

$$k_s \| An \| = (e_s)^{\frac{1}{2}}$$

$(k_s)^{1/n}$ then specifies a time scale factor which will define the range to ensure convergence to within the error bound. Note that this allows the time-scale factor to increase if a larger range is acceptable.

Computationally, the matrix A of coefficients for a 10^{th}-order fit is calculated over an initial range of the independent variable. The error of the approximation is checked for convergence and if necessary a suitable time scale factor k_s is calculated which redefines the range. This process is then continued until the required range of the independent variable has been covered.

2. Estimation

Consider the differential equation defined by equation (1) and consider that u is a set of stochastic disturbances such that

$$\varepsilon(u) = 0 \qquad \frac{1}{t_1 - t_o} \int_{t_o}^{t_1} \varepsilon(u\, u^T)\, dt = Q$$

and

where ε is the expected value operator.

Assume that an estimate exists over the time range $t_o \leq t \leq t_1$ such that

$\hat{x}(t)$ is an estimate of $x(t)$ $\quad t_o \leq t \leq t_1$

and $\tilde{x}(t) = x(t) - \hat{x}(t)$

and that

$$\hat{V}(t_1) = \frac{1}{t_1 - t_o} \int_{t_o}^{t_1} \varepsilon(\tilde{x}(t)\, \tilde{x}(t)^T)\, dt \text{ is known.}$$

Then a solution can be found by the method explained in Sec. 1 for the time range $t_1 \leq t \leq t_2$.

Let $\bar{x}(t)$ represent this prediction of $x(t)$ $t_1 \leq t \leq t_2$

and $\bar{V}(t_2) = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \varepsilon(x - \bar{x})\,(x - \bar{x})^T\, dt$

therefore $\bar{V}(t_2) = \phi(t_1, t_2)\, \hat{V}(t_1)\, \phi^T(t_1, t_2) + \Delta\, Q \Delta^T$ (4)

where $\phi(t_1, t)$ is the fundamental matrix defined by

$$\frac{d}{dt} \phi(t_1, t) = M \phi(t_1, t) \qquad \phi(t_1, t_1) = I$$

and $$\Delta = \int_{t_1}^{t_2} \phi(t_2, \tau) N \, d\tau .$$

Now assume that measurements are available as continuous functions $z(t)$ for the time range, $t_1 \leq t \leq t_2$ and are related to the state variables by

$$z(t) = Hx(t) + v$$

where H is a measurement matrix and v is additive noise such that

$$\varepsilon(v) = 0$$

and $$\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \varepsilon(v\, v^T)\, dt = R$$

Then the optimal estimate of $x(t)$ for $t_1 \leq t \leq t_2$ is given by

$$\hat{x}(t) = \bar{x}(t) + K(z(t) - H\, \bar{x}(t)) \qquad (5)$$

If K is chosen to minimise the convariance matrix of $\tilde{x}(t)$ the result is

$$K = \bar{V}(t_2) H^T (H\, \bar{V}(t_2) H^T + R)^{-1} \qquad (6)$$

and $$\hat{V}(t_2) = (I - K\, H)\, \bar{V}(t_2) \qquad (7)$$

The above equations define an estimator that is continuous over a chosen range. Estimates are obtained for all the state variables and the covariance matrix of the

estimation error is also computed.

These equations bear a close similarity to the standard equations of the Kalman filter (1,2,3). The main difference is that the continuous and differentiable matrix function $\varepsilon(\tilde{x}(t)\ \tilde{x}(t)^T)$ is replaced by the step function $\hat{V}(t)$. The result of this is that the awkward Riccati matrix differentia equation is removed from the algorithm.

3. Parameter estimation

If in equation (1) there exist some unknown parameters then the equation can be considered as a non-linear system by treating these unknown parameters as state-variables so that

$$\dot{x} = f(x, k) + Nu \qquad (8)$$

where k is a vector of unknown parameters.

Using initial guesses x*, k*, equation (8) can be linearised

$$f(x^* + \Delta x, k^* + \Delta k) \simeq f(x^*, k^*) + f_x'(x^*,k^*)\Delta x + f_k'(x^*, k^*)\,\Delta k$$

where $f_x'(x^*,k^*) = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \cdots \\ \cdots\cdots & \cdots\cdots & \cdots \\ \partial f_m/\partial x_1 & \partial f_m/\partial x_2 & \end{bmatrix}$ at x^*,k^*

and $f_k'(x^*,k^*) = \begin{bmatrix} \partial f_1/\partial k_1 & \partial f_1/\partial k_2 & \cdots \\ \cdots\cdots & \cdots\cdots & \cdots \\ \partial f_m/\partial k_1 & \partial f_m/\partial k_2 & \end{bmatrix}$ at x^*,k^*

then from equation (8)

$$\dot{\Delta x} = f_x'(x^*,k^*)\,\Delta x + f_k'(x^*,k^*)\Delta k + f(x^*,k^*) + Nu$$

therefore $\begin{bmatrix} \dot{\Delta x} \\ \dot{\Delta k} \end{bmatrix} = \left[\begin{array}{c|c} f_x'(x^*,k^*) & f_k(x^*,k^*) \\ \hline 0 & 0 \end{array}\right] \begin{bmatrix} \Delta x \\ \Delta k \end{bmatrix} + \begin{bmatrix} f(x^*,k^*) + Nu \\ 0 \end{bmatrix}$

and using the results of section (2) on the augmented system above provides continuous estimates of the state variables and the parameters.

4. Non-linear Estimation

Using the procedure explained in section (3) non-linear models can be linearised about some nominal state. The results of section (2) can then be applied to the linearised model to provide continuous estimates of state variables and parameters.

The simplest technique is to linearise the model about the current state estimates, but there are alternatives based upon an extrapolating of the current estimates into the next time range and linearising about this extrapolated state.

5. Results

The results show the method applied to a continuous stirred tank reactor in which the following reaction occurs:

$$A \underset{k_2}{\overset{k_1}{\rightleftharpoons}} B \xrightarrow{k_3} C$$
$$\qquad\qquad\quad \downarrow k_4$$
$$\qquad\qquad\quad D$$

The feed contains A only. The coefficients defining this system are

$$k_1 = 0.538,\ k_2 = 0.385,\ k_3 = 0.062,\ k_4 = 0.246$$

and the ratio of volumetric flowrate of feed to the capacity of the reactor = 0.308
then

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} -0.846 & 0.385 & 0 & 0 \\ 0.538 & -1 & 0 & 0 \\ 0 & 0.062 & -0.308 & 0 \\ 0 & 0.246 & 0 & -0.308 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0.308\ x_5 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where
x_1 = output concentration of A
x_2 = " " of B
x_3 = " " of C
x_4 = " " of D
x_5 = input concentration of A.

5.1 Simulation

This system was simulated for a series of step changes in the value of x_5. The values of $x_1 - x_5$ are plotted in Fig. (1) and the value of x_1 is compared with the analytical solution. The agreement is seen to be good.

Additional runs of the simulation will be presented which demonstrate the effect of a variable range of solution.

The polynomials used for the simulation were normalised Legendre polynomials.

5.2 State-variable estimation

The values of x_1 and x_2 obtained by the simulation were corrupted by additive noise (standard deviation = 0.2888) and used as measurements for an estimation of all the state-variables (Fig. (2)). The measurement function was chosen as the straight line from the previous estimates to the new measurements, the measurements being taken at the end of the range of solution. The method, however, allows a great deal of freedom as to when the measurements are taken within the range and in the number of measurements used to define the measurement function. (Appendix 1.)

The value of Q represents the covariance of the errors on the known value of x_5 and was in this case 0.0833. Poor initial estimates of the state variables were made (Fig.(2)) and the initial covariance matrix of the estimation errors $\hat{V}(t_1)$ was taken as the unit matrix.

5.3 Parameter estimation

The values of x_5 and k_3 were considered unknown and an augmented state vector set up.

Estimates were obtained for the state variables and for the unknown parameters using the same measurements as before (Figs. (3,4,5)). The initial estimates were taken to be zero for each component and the initial covariance of the estimation errors was again taken as the unit matrix.

The value of Q this time represents the covariance of the unknown value of x_5 and as such is a parameter that affects the performance of the filter. If the value of Q is too high the filtered estimates are 'noisy'. The value of Q used to obtain the results was 0.0833.

6. Discussion

The results show that the method described provides a solution to the problems of simulation and of obtaining good continuous estimates of state variables and unknown parameters. In particular good estimates are obtained of unknown process inputs (Fig.(4)) a problem that has currently been reported to cause difficulty (4).

The advantage of continuous solutions is that many standard mathematical tools (e.g. differentiation, integration, interpolation, extrapolation, etc.) become available, particularly in such problems as optimal control.

Further work in progress includes [illegible]ension [illegible] with inaccurate models through the prop[illegible]ice of covariance matrix Q that is to say, the problem of a[illegible] filtering. Attention will also be given to the stra[illegible] taking the measurements.

Appendix 1

Consider the situation where $\hat{x}(t_1)$ i.e. the estimate of x at $t = t_1$ and $\hat{V}(t_1)$ are known.

At $t = t_2$ measurements $z(t_2)$ become available, with a covariance matrix of errors = R_m.

The measurement function to be used is the straight line from $H\hat{x}(t_1)$ to $z(t_2)$ and is given by

$$z(t) = \hat{b}t + \hat{d} \qquad t_1 \leq t \leq t_2$$

where $\hat{b} = (z(t_2) - H\hat{x}(t_1)) / (t_2 - t_1)$,

and $\hat{d} = (t_2 H\hat{x}(t_1) - t_1 z(t_2)) / (t_2 - t_1)$

The linear approximation to the true state is

$$Hx(t) \simeq bt + d$$

where $b = (H x(t_2) - Hx(t_1) / (t_2 - t_1)$

and $d = (t_2 Hx(t_1) - t_1 Hx(t_2)) / (t_2 - t_1)$

The linear approximation to the measurement error

$$v(t) = z(t) - Hx(t) \simeq (\hat{b} - b)t + (\hat{d} - d)$$
$$= \tilde{b}t + \tilde{d}$$

The covariance matrix $R = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \varepsilon v(t)\, v(t)^T \, dt$

is calculated from

$$R \simeq \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} \varepsilon (\tilde{b}t + \tilde{d})(\tilde{b}t + \tilde{d})^T \, dt$$

If the range $t_1 \leq t \leq t_2$ is normalised to -1 to +1,

the above result reduces to

$$R \simeq 1/3 \left[H \hat{V}(t_1) H^T + R_m \right]$$

In this work $R_m = \begin{bmatrix} 0.0833 & 0 \\ 0 & 0.0833 \end{bmatrix}$

Notation

Capital letters, A, V, M, Q, etc, denote matrices. Lower case letters, x, y, z, etc. denote vectors and scalars.

The dot notation, $\dot{x}$, denotes differentiation with respect to the independent variable t.

The dash notation, $f_x'(x)$, denotes partial differentiation with respect to the vector x.

References

(1) R. E. KALMAN and R. S. BUCY
TRANS.ASME. Series D, Journal of Basic Eng. 83, (1961), 95-108

(2) R. E. KALMAN
TRANS.ASME. Series D, Journal of Basic Eng. 82, (1960), 35-45

(3) H. W. SORENSON
ADVANCES IN CONTROL SYSTEMS, Vol. 3 , 219-292
Academic Press, 1966

(4) J. C. HAMILTON, D. E. SEBORG and D. GRANT FISHER,
A.I.Ch.E. Journal, 19, (1973), 901-910

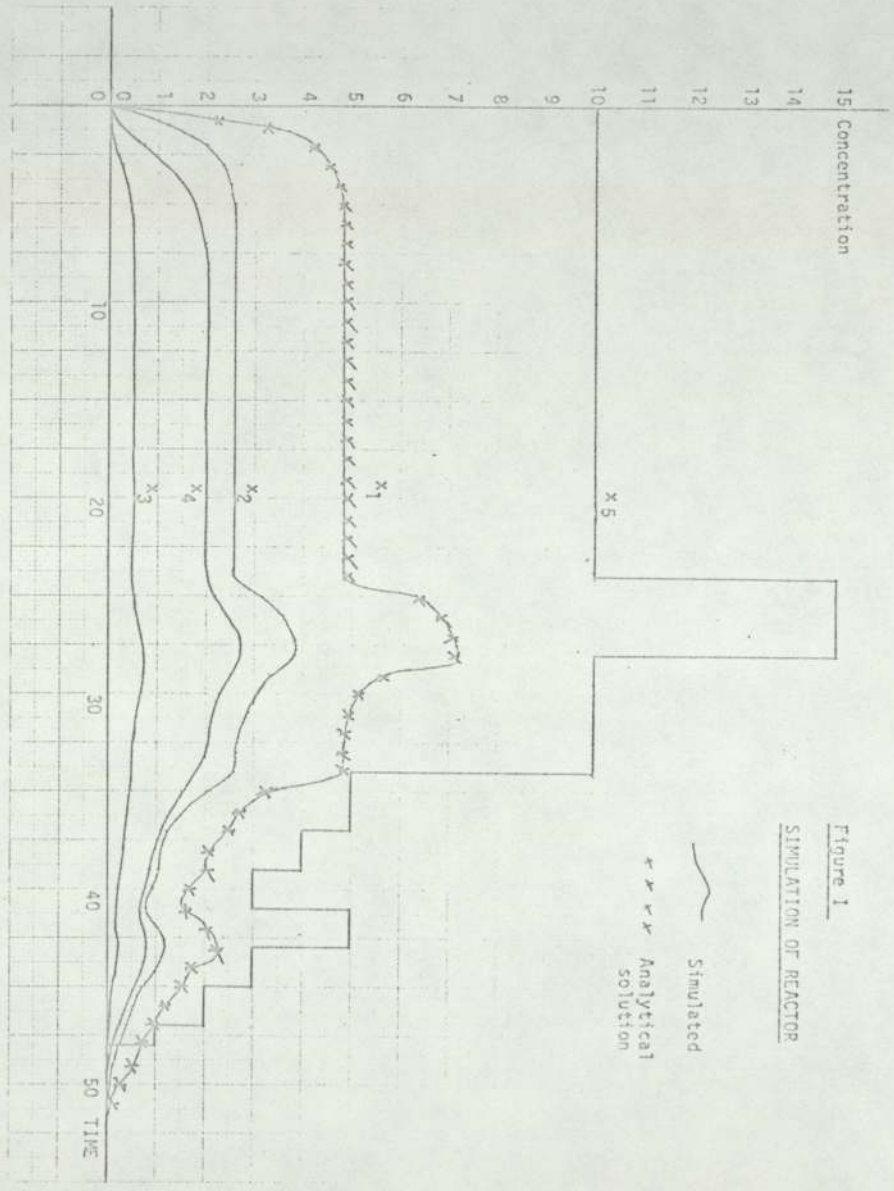


Figure 1
SIMULATION OF REACTOR

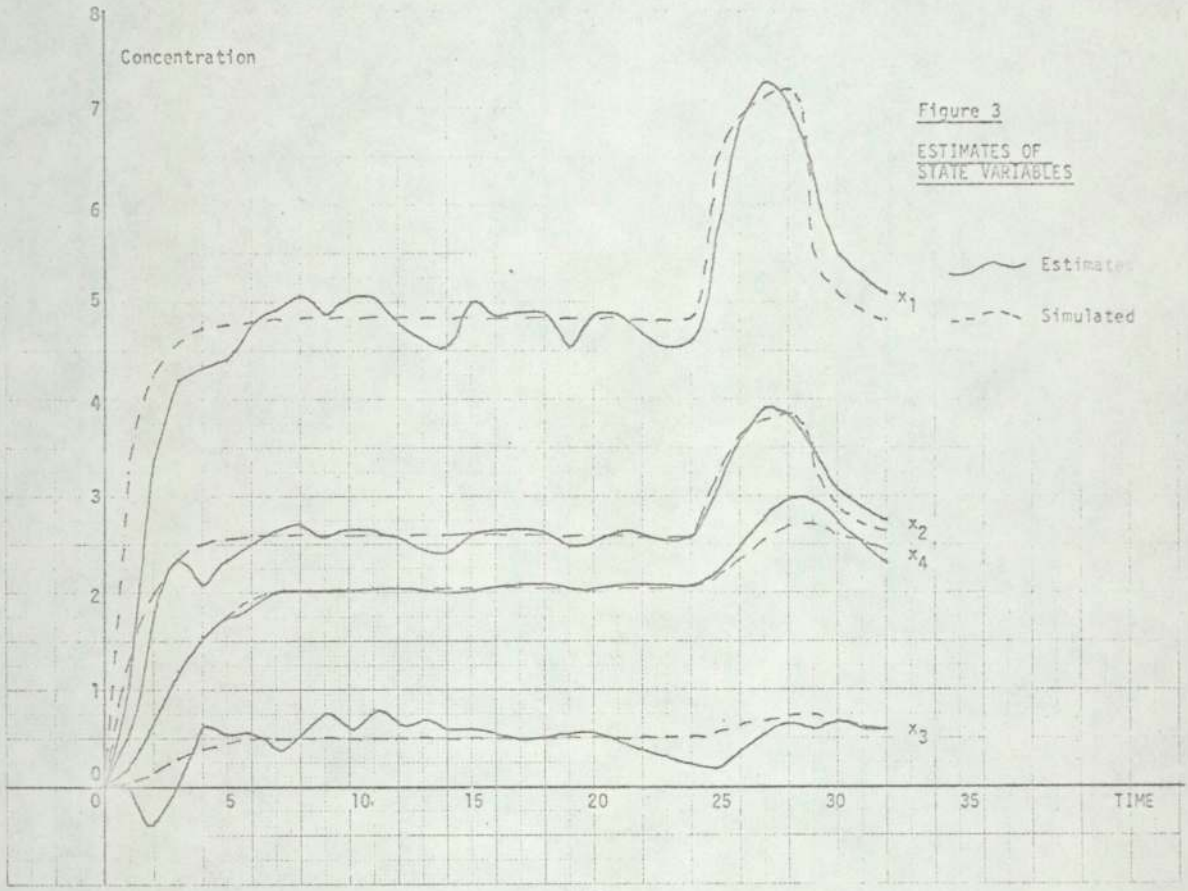


Figure 3

ESTIMATES OF STATE VARIABLES

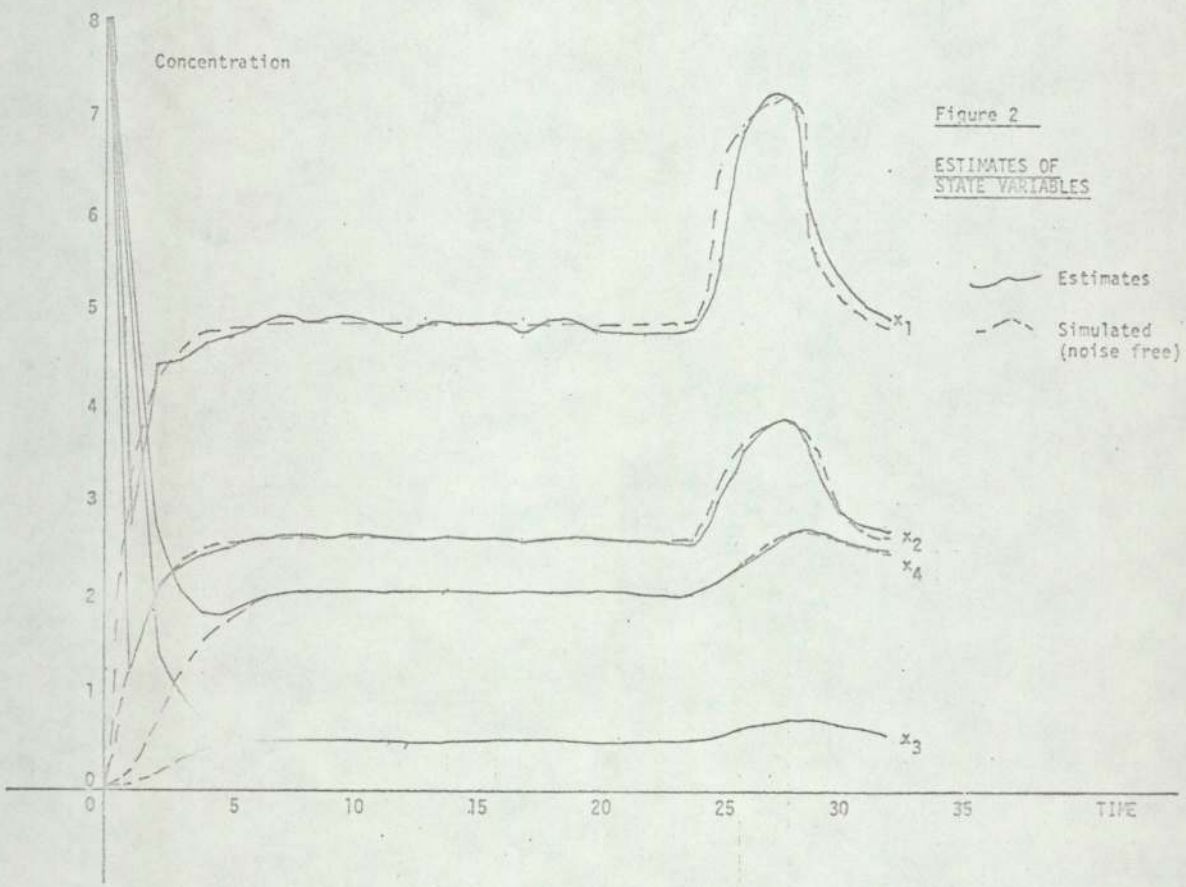


Figure 2

ESTIMATES OF STATE VARIABLES

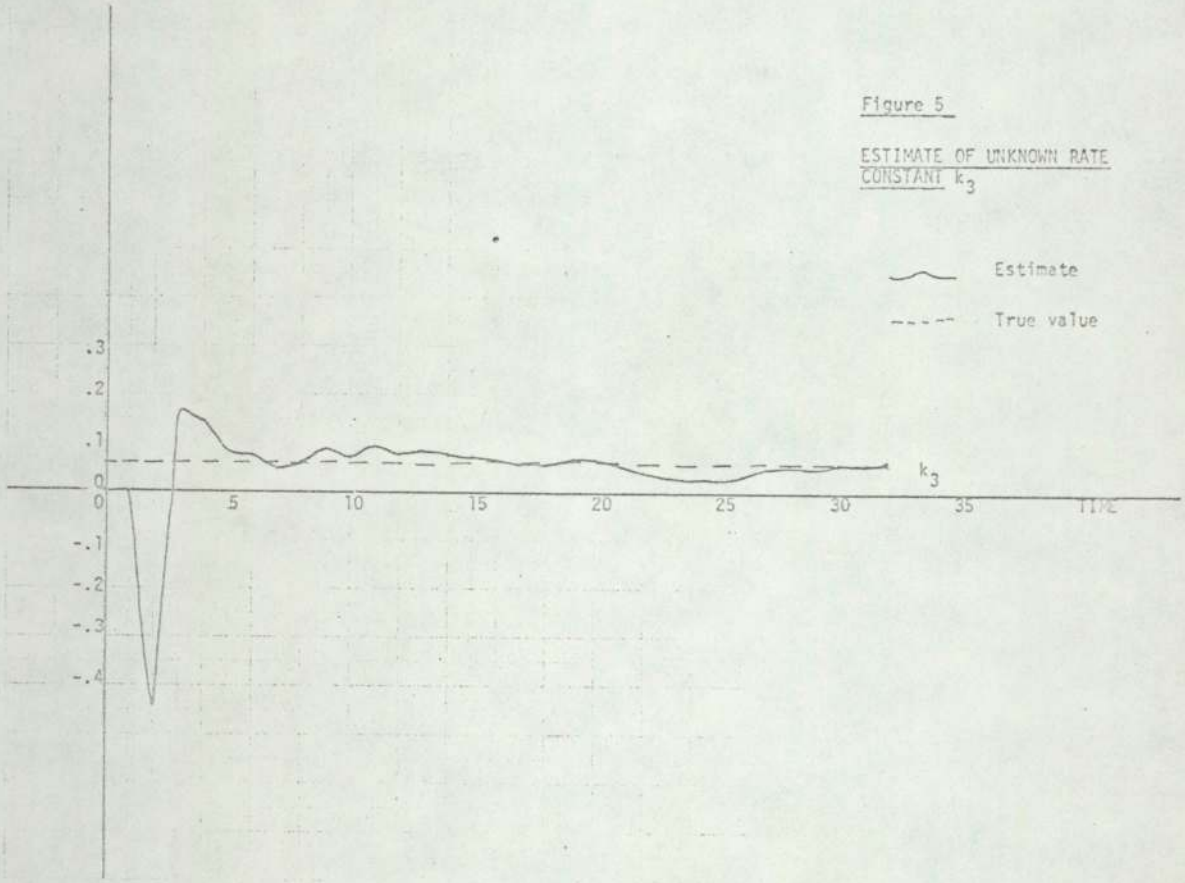


Figure 5

ESTIMATE OF UNKNOWN RATE CONSTANT k_3

Estimate

True value

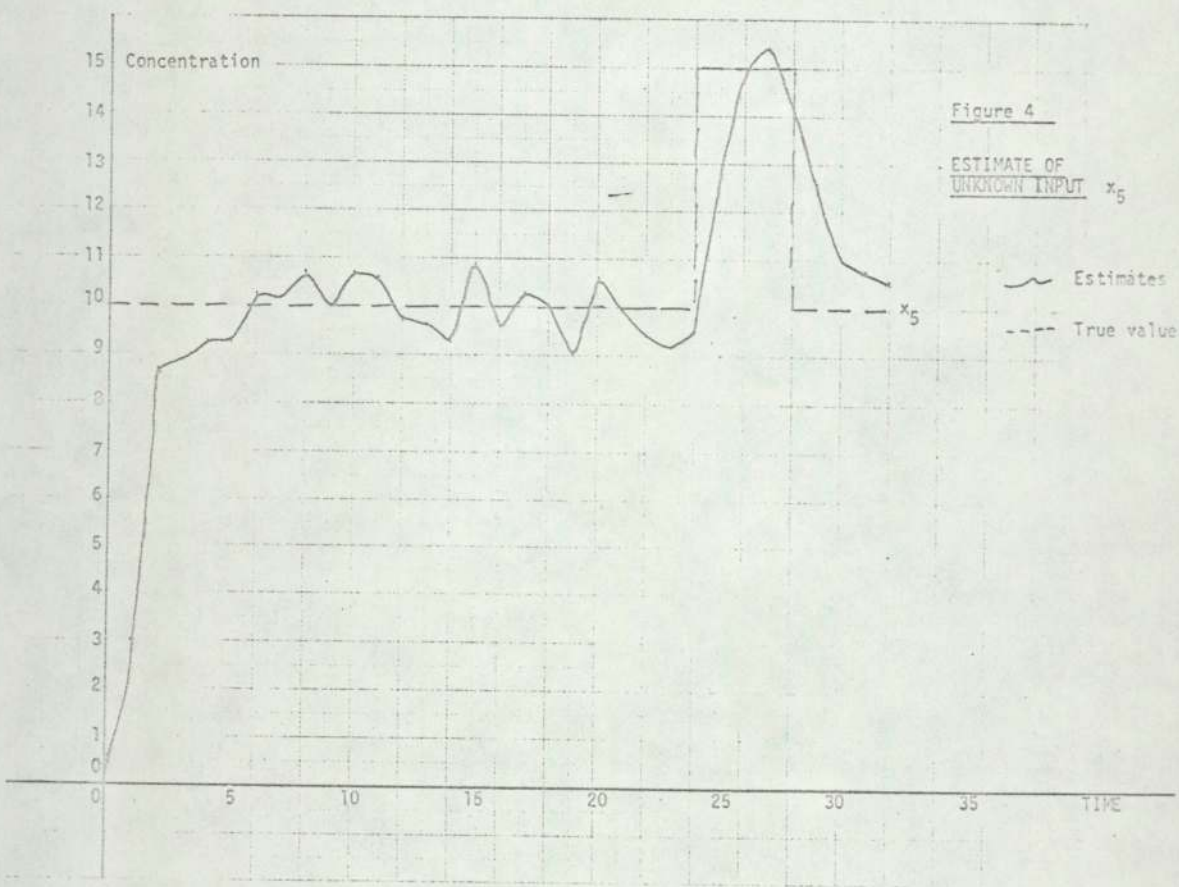


Figure 4

ESTIMATE OF UNKNOWN INPUT x_5

Estimates

True value