1977

Identification And Control Applied To Fermentation

Vuong Thanh Xuyen

Dedicated to my parents

Kính dâng tới ba má

IDENTIFICATION AND CONTROL APPLIED TO FERMENTATION

by

VƯƠNG THANH XUYÊN
Faculty of Engineering Science

Submitted in partial fulfillment
of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Faculty of Graduate Studies
The University of Western Ontario
London, Canada
August, 1976

ABSTRACT

A systematic approach was taken to compute the optimum operating conditions of a chemostat with and without recycle, using the Exhaustive Search and Newton-Raphson techniques. The process was assumed to follow the Haldane-Monod model, and the performance index of the chemostat considered was the production rate of biomass. Optimization variables included temperature, pH, influent substrate concentration, dilution rate and recycle constants.

The statement of a general system identification problem by least squares was mathematically formulated. The most effective optimization techniques used in solving the identification problem were then reviewed. They included the Gauss-Newton technique, its modifications and the Quasilinearization technique. A batch fermentation of Aureobasidium pullulans and a continuous fermentation of Morchella crassipes were then particularly identified.

Difficulty of estimating a large number of parameters in batch process identification was explained. A technique to overcome the difficulty was then derived. The technique requires the model to be linear in its state vector and converts a very high dimensional optimization problem to a low one with additional constraints by utilizing the Super-

position Principle and the Walling-Lawton-Sylvestre idea. Based on the technique, an on-line identification scheme was also derived to update the model as data from each new batch are being received.

Finally, a stochastic model for a continuous fermentation process was developed to take into account measurement errors and disturbances caused by imperfect mixing of bioreactor and substrate feeding tank. A problem of regulation effluent concentrations was also formulated and solved, using the Separation Theorem. Dilution rate was used as a control input.

ACKNOWLEDGEMENT

In this personal page, I would like to express my appreciation to the following people without whose help my Ph.D. program would have been more difficult or even impossible to be completed.

Dr. J.E. ZAJIC, who taught me the concept and methods in biochemical engineering and showed me the Art of learning, teaching and working with Science.

Dr. W.Y. SVRCEK, who selected the research topics and have been guided me with unfailing support and human understanding throughout the research.

Dr. A. LEMAY, my high school and college fellow, for providing and explaining his P. pullulans and M. crassipes data.

Finally but not the least EIK-KEE, my wife, for her warm and persistent encouragement and for spending many hours typing the manuscript.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE(*)

$\underline{a}$ = Process parameter vector.

$\underline{A}$ = Weighting matrix of state vector, defined from Equation (D.59).

$\underline{b}$ = Parameter vector in static model.

B = Total number of batch data used.

$\underline{B}$ = Weighting matrix of control vector, defined from Equation (D.59)

C = Biomass concentration in bioreactor.

$\overline{C}$ = Biomass regulating level.

d = Total number of measurements.

D = Dilution rate or total number of parameters to be estimated.

$\overline{D}$ = Dilution rate regulating level.

e = Subsript to denote effluent.

$\underline{e}$ = Error vector in dynamic model defined by either Equation (4:43) or Equation (C.7).

E = Expectation or statistical average operator.

f = Scalar function or subscript to denote at final time.

$\underline{f}$ = Vector function in plant model.

F = Flow rate or F-value in F-test defined by Equation (C.42)

(*) This table is not complete in that some symbols are not included or are used in other definitions than listed here. In these cases, the symbols are locally defined in the text and hence should not lead to confusion.

F_L = Lowest F-test value.

$\underline{F}$ = Matrix function in plant model.

$\underline{g}$ = Vector function in observation model.

$\underline{G}$ = Matrix function in observation model.

$\underline{h}$ = Vector function in static model.

H = pH or Hamiltonian function.

$\underline{H}$ = Approximate Hessian matrix of S.

i = Running index or subscript to denote at the $(i+1)^{th}$ measurement.

I = Number of measurements minus one.

$\underline{I}$ = Identity matrix.

j = Running index or superscript to denote from the j^{th} batch.

J = System performance index.

k = Weighting constant or iteration index.

k_c = Constant defined by Equation (5.7) describing degree of mixing and nature of biomass.

k_{s1} = Constant defined by Equation (5.8) describing degree of mixing in bioreactor and nature of substrate.

k_{s2} = Constant defined by Equation (5.9) describing degree of mixing in substrate feeding tank and nature of substrate.

K = Metabolism coefficient or specific death rate.

K_i = Inhibition coefficient.

K_s = Saturation coefficient.

$\underline{K}_1$ = Continuous Kalman control gain matrix.

$\underline{K}_2$ = Continuous Kalman filter gain matrix.

ℓ = Dimension of output vector $\underline{y}$.

M = Dimension of static parameter vector $\underline{b}$.

$\underline{M}$ = Joint weighting matrix of state vector and control vector, defined from Equation (D.59).

n = Dimension of state vector $\underline{x}$.

N = Dimension of dynamic parameter vector $\underline{a}$, or total number of measurements minus one, or number of stages.

$\underline{\bar{N}}$ = Joint weighting matrix of state vector and control vector, defined from Equation (5.15).

o = Subscript to denote at starting time.

$\underline{0}$ = Null vector or null matrix.

p = Probability density function.

$\underline{p}$ = New state vector defined by Equation (D.21).

P = Product concentration in bioreactor.

$\underline{P}_o$ = Covariance matrix of initial state vector $\underline{x}(t_o)$.

$\underline{P}_f$ = Weighting matrix of final state vector $\underline{x}(t_f)$, defined from Equation (5.15).

$\underline{q}$ = Gradient vector of S.

$\underline{Q}_1$ = Weighting matrix of state vector $\underline{x}$, defined from Equation (5.15).

$\underline{Q}_2$ = Spectral density matrix of process noise vector $\underline{w}$.

R = Gas constant or subscript to denote at recycle stream.

$\underline{R}_1$ = Weighting matrix of control vector $\underline{u}$, defined from

Equation (5.15).

$\underline{R}_2$ = Spectral density matrix of measurement noise vector $\underline{v}$.

S = Substrate concentration in bioreactor or sum of squares of errors or integration of squares of errors.

$\bar{S}$ = Substrate regulating level.

S_I = Influent substrate concentration.

$\bar{S}_I$ = Average influent substrate concentration.

t = Time.

T = Temperature or superscript to denote matrix transpose.

$\underline{T}$ = Cross spectral density matrix of process noise vector $\underline{w}$ and measurement noise vector $\underline{v}$.

$\underline{T}(s)$ = Wiener filter.

u = $D - \bar{D}$.

$\underline{u}$ = Input or control vector.

v_c = Measurement error in effluent biomass concentration.

v_s = Measurement error in effluent substrate concentration.

$\underline{v}$ = Measurement noise vector or new state vector consisting of old state vector and process parameter vector.

V = Volume of medium in bioreactor.

w_c = Disturbance in biomass concentration due to imperfect mixing.

w_{s1} = Disturbance in effluent substrate concentration due to imperfect mixing.

w_{s2} = Disturbance in influent substrate concentration due to imperfect mixing.

$\underline{w}$ = Process noise vector.

$\underline{x}$ = State vector.

$\underline{y}$ = Output or observation vector.

Y = Yield coefficient.

$\underline{z}$ = Data vector between batches or new observation vector defined by Equation (D.22).

ss = Subscript to denote at steady state.

tr = Trace operator.

$\dot{}$ = Superscript to denote derivative with respect to time.

* = Superscript to denote optimum value.

^{x} = Superscript to denote scaled vector or matrix.

$\tilde{}$ = Superscript to denote error.

$\hat{}$ = Superscript to denote estimated value.

$|(.)|$ = Absolute value of (.).

$|(\underline{.})|$ = Determinant of matrix $(\underline{.})$.

$\|(\underline{.})\|$ = Euclidean norm of vector $(\underline{.})$.

α = Significance level or recirculation ratio.

β = Stepsize factor used in Hartley's Modified Gauss-Newton technique.

β_C = Biomass concentration ratio.

β_S = Substrate concentration ratio.

$\underline{\Gamma}$ = Control transition matrix.

δ_D = Dirac delta function or Impulse function.

δ_K = Kronecker delta function.

$\delta(.)$ = Change in (.).

$\underline{\varepsilon}$ = Observation noise vector defined by Equation (D.61).

$\underline{\eta}$ = Error vector in static model or plant noise vector defined by Equation (D.60).

$\underline{\theta}$ = Vector to be identified, composed of process parameter $\underline{a}$ and initial state vector $\underline{x}(t_o)$.

${}_1\underline{K}$ = Discrete Kalman control gain matrix.

${}_2\underline{K}$ = Discrete Kalman filter gain matrix.

λ = Marquardt's constant.

Λ_C = Biomass recycle factor defined by Equation (2.57).

Λ_S = Substrate recycle factor defined by Equation (2.58).

$\underline{\Lambda}$ = Adjoint matrix or Lagrange multiplier matrix.

μ = Specific growth rate.

μ_m = Maximum specific growth rate.

ν = Specific substrate consumption rate.

ρ = Specific product formation rate.

σ = Standard deviation of error.

σ_1^2 = Power spectrum of v_c.

σ_2^2 = Power spectrum of v_s.

$\underline{\Sigma}$ = Disturbance transition matrix.

$\underline{\Phi}$ = State transition matrix.

χ_n^2 = Chi-square distribution with n-degrees of freedom.

$\underline{\Psi}$ = Particular solution of linear differential equation (4.9), computed from Equation (4.13).

$\underline{\Omega}$ = Covariance matrix of $\underline{\varepsilon}$.

${}_\eta\underline{\Omega}$ = Covariance matrix of $\underline{\eta}$.

${}_{\eta\varepsilon}\underline{\Omega}$ = Cross-covariance matrix of $\underline{\eta}$ and $\underline{\varepsilon}$.

CHAPTER I

INTRODUCTION

Classical control system design is generally a trial-and-error process in which various methods of analysis are used iteratively to determine the design parameters of an "acceptable" system. Acceptable performance is generally defined in terms of time and frequency domain criteria, such as rise time, settling time, peak overshoot, gain and phase margins, and bandwidth. Radically different performance criteria must be satisfied, however, by the complex multiple-input, multiple-output systems required to meet the demands of modern technology. For example, the design of a spacecraft attitude control system that minimizes fuel expenditure is not amendable to solution by classical methods. A new and direct approach to these complex systems, called modern control theory, has been made feasible by the development of the digital computer (Kirk 1970).

In this new and direct approach, the control problem must first be represented as a state space model with a suitable performance index (also called a cost function or a profit function). Then, the control problem is

solved by optimizing the performance index using optimization techniques such as Pontryagin's Maximum Principle (Pontryagin et al 1962, Fan 1966, Sage 1970) and Bellman's Dynamic Programming (Bellman 1957, Kirk 1970). Note that the state space model is a set of differential or difference equations which supposely describe the characteristics of the system to be controlled, and can be found by some appropriate system identification technique (Lee 1964, Sage and Melsa 1971b); and the performance index is a mathematical measure of the performance of the control system which is in turn a mathematical function of the model variables.

Although modern control theory is considered a rather new and fast developing field, its applications to fermentation have been recorded. D'Ans et al using dilution rate as a control input, solved the general problems of time optimal control (D'Ans et al 1971a) and of maximum harvest (D'Ans et al 1971b, 1972) utilizing the Monod and Haldane-Monod equations for the specific microbial growth rate. Takamatsu et al (1975) recently studied an amino acid fermentation and also with dilution rate as a control input solved the maximum harvest problem by modeling the specific microbial growth rate and the specific product

formation rate as first and second order algebraic equations in substrate concentration respectively. Shah (1972) with dilution rate as a control input solved the regulator problem using the Monod equation for the specific microbial growth rate.

Using temperature as a control input to solve the maximum harvest problem the papers of Constantinides et al (1970) for a batch penicillin fermentation and of Rai and Constantinides (1974) for a batch gluconic acid fermentation can be cited. King et al (1974) with also temperature as a control input solved the sub-optimal harvest problem for a batch penicillin fermentation. And finally with also the same batch penicillin fermentation King (1975) solved the sub-optimal energy conservation problem.

The above authors all utilized deterministic models. Svrcek et al (1974) formulated a stochastic model and derived a control policy to regulate effluent concentrations. The stochastic model is discrete and was formed by linearizing and discretizing the deterministic Monod model and then adding disturbance terms which were assumed to be white and Gaussian noise sequences with known variances. The control policy which they named "Kalman

Control" policy was given as a Kalman filter cascaded with a proportional controller.

The purpose of this research is to further apply modern control theory to fermentation. Firstly, a systematic approach is taken to compute the optimum operating conditions of a chemostat with and without recycle. The process is assumed to follow the Haldane-Monod model, and the performance index of the chemostat considered is the production rate of biomass. Optimization variables include temperature, pH, influent substrate concentration, dilution rate and recyle constants.

Secondly, a technique is derived to overcome the difficulty of estimating a large number of parameters which occurs in linear dynamic batch process identification by least squares. The technique, combining together the Superposition Principle (Distefano et al 1967, Sage and Melsa 1971b) and the Walling-Lawton-Sylvestre idea (Walling 1968, Lawton and Sylvestre 1971), converts a very high dimensional optimization problem to a low one with additional constraints. It produces faster convergence, a larger region of convergence and reduces computing time tremendously. The technique is useful in determining a

mathematical model for microbial growth in a batch fermentation process, because the specific microbial growth rate does not often depend on the biomass concentration but rather depends on factors such as substrate concentration, medium temperature, pH, etc.. The Monod model and the Verhulst-Pearl model are two examples. From the technique derived, the number of parameters to be estimated is reduced to a constant, independent of the number of batch runs from which the data are used. Based on the developed technique, an on-line identification scheme is also derived to update the model as data from each new batch are being received.

Finally, a stochastic model for a continuous fermentation process is developed to take into account measurement errors and disturbances due to imperfect mixing of bioreactor and of substrate feeding tank. A problem of regulating effluent concentrations is also formulated and solved using the Separation Theorem (Wonham 1968, Meditch 1969). Dilution rate is considered as a control input.

I THEORY OF GROWTH IN BATCH AND CONTINUOUS FERMENTATION PROCESSES

In this section, the theory of growth in batch and continuous fermentation processes on which the contents of later chapters will be based will be briefly reviewed. Other details can be found in the microbiological literature (Herbert 1972).

Continuous cultivation of micro-organisms was first introduced by Monod (1950) and Novick and Szilard (1950), and has become an important tool in biotechnology. It consists of growing a microbial culture in a bioreactor, Figure 1.1, with a flow of fresh substrate medium coming from a feeding tank at the rate F (volume/time). The culture with its micro-organisms and metabolites is removed at the rate F and stored in a collecting tank. Since the inflow and outflow rates are the same, the volume V occupied by the culture in the bioreactor is constant. The rate of growth of the micro-organisms is limited by various factors among which is the supply of nutrient substrate. This substrate is then called the growth-limiting substrate. When the continuous fermentation process is operated at constant conditions : constant flow rate, constant pH, constant temperature etc., it is called a chemostat.

A batch fermentation process is a limiting case of the continuous fermentation process where the flow rate F is zero. Thus, its operating time is limited because the original

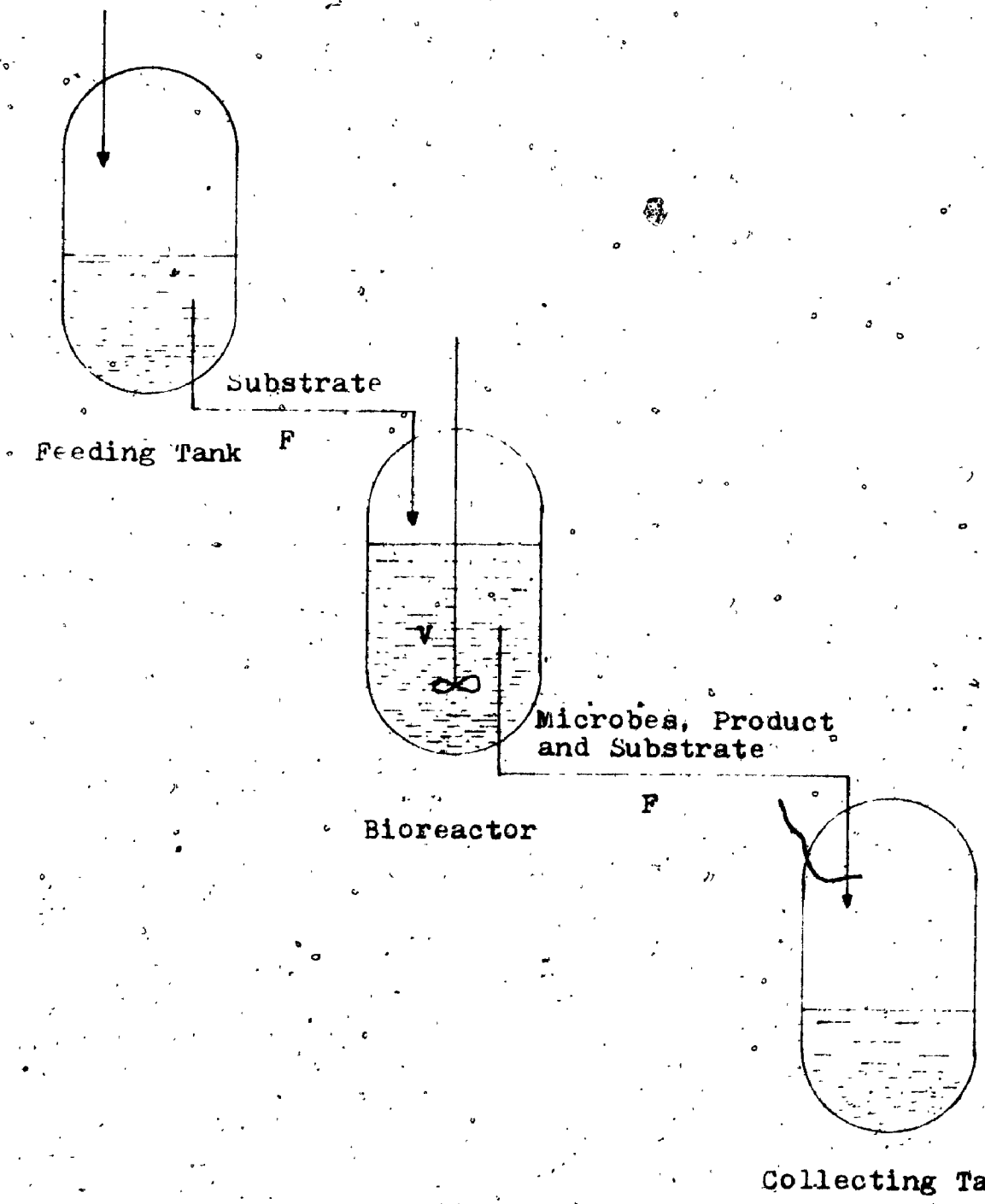


Figure 1.1 Schematic of a continuous fermentation process.

nutrient substrate supplied to the bioreactor at the beginning of the operation will become exhausted, microbial growth therefore will slow down and eventually stop.

On the other hand, in a continuous fermentation process, if mutation and contamination do not occur, the operation is very economical as the culture can be harvested continuously for a very long period of time.

For purposes of control or just of understanding, a mathematical description of a fermentation process is needed. It would be too complex and too difficult to take all the biochemical phenomena involved into consideration. Furthermore, a fully detailed model may result in a control problem, the solution of which can never be solved mathematically or can never be implemented. It is, therefore, often reasonable to attempt to apply in the first instance as simple a mathematical model as possible, and then to check experimentally whether it is suitable for the formulated control problem.

For a process which involves only one species of micro-organism, one kind of product and one type of substrate and where perfect mixing of the bioreactor is assumed, a mathematical model representing the process can be obtained by writing the material balance for the micro-organism (biomass), the growth-limiting substrate and the formed product. If C(t),

$C(t)$ and $P(t)$ represent the biomass(*), substrate and product concentrations, respectively, in the bioreactor or at the outflow at time t, then:

- For a biomass material balance:

Increase = Inflow - Outflow + Net growth

$$V\frac{dC(t)}{dt} = 0 - FC(t) + V\left[\mu(t) - K(t)\right]C(t)$$

where,

$\mu(t)$ = Specific growth rate at time t (1/time),

$K(t)$ = Metabolism coefficient or specific death rate at time t (1/time).

If $D = F/V$ = Dilution rate (1/time), t_o is time at which the operation starts and C_o is the biomass concentration at time t_o, then the differential equation representing the biomass at the outflow (or in the bioreactor) is:

$$\frac{dC(t)}{dt} = \left[\mu(t) - K(t) - D\right]C(t) \qquad (I.1)$$

Initial condition: $C(t_o) = C_o$

- For a substrate material balance:

Increase = Inflow - Outflow - Consumption

$$V\frac{dS(t)}{dt} = FS_1 - FS(t) - V\bar{V}(t)C(t)$$

or,

(*) Note that to represent biomass concentration in this text, C is used instead of the more commonly applied x in biochemical literature. However, x here will be designated to represent a state variable in later sections.

$$\frac{dS(t)}{dt} = D[S_I - S(t)] - \nu(t)C(t) \qquad (1.2)$$

Initial condition: $S(t_o) = S_o$

where,

S_I Substrate concentration in the feeding tank,

$\nu(t)$ Specific substrate consumption rate at time t (1/time),

S_o Substrate concentration in the bioreactor at time t_o.

- For a product material balance:

Increase = Inflow - Outflow + Formation

$$V\frac{dP(t)}{dt} = 0 - FP(t) + V\rho(t)C(t)$$

or,

$$\frac{dP(t)}{dt} = - DP(t) + \rho(t)C(t) \qquad (1.3)$$

Initial condition: $P(t_o) = P_o$

where,

$\rho(t)$ = Specific product formation rate at time t (1/time),

P_o = Product concentration in the bioreactor at time t_o.

Eqs. (1.1), (1.2) and (1.3) form a mathematical model representing the process. Note that as there is no biomass flowing into the bioreactor, C_o must be greater than zero for the bioreaction to exist.

In evaluating each new fermentation process, the metabolism coefficient K(t), the specific growth rate $\mu(t)$,

the specific substrate consumption rate $\nu(t)$ and the specific product formation rate $\rho(t)$ are not known in advance and depend on the nature of the microbes, the medium temperature, the medium pH, the rate of oxygen supply (in aerobic fermentation) etc.. However, they can be estimated by various developed estimation techniques (Sage and Melsa 1971a, 1971b).

From experience, when agitation, aeration, temperature and pH are held fixed, the metabolism coefficient $K(t)$ is often very small as compared to the specific growth rate $\mu(t)$ and therefore is either ignored or treated as a small constant. The specific growth rate is often considered to follow the Haldane-Monod equation, Eq. (1.4) (Haldane 1965, Andrews 1968, D'Ans 1972).

$$\mu(t) = \frac{\mu_m}{1 + \frac{K_s}{S(t)} + \frac{S(t)}{K_i}} \qquad (1.4)$$

where μ_m, K_s and K_i are positive numbers and are defined as the maximum specific growth rate, the saturation coefficient and the inhibition coefficient respectively.

If the microbial growth is not inhibited by the growth limiting substrate at high concentrations, then $K_i \to \infty$ and the Haldane-Monod equation is reduced to the Monod equation, Eq. (1.5).

$$\mu(t) = \frac{\mu_m S(t)}{S(t) + K_s} \qquad (1.5)$$

In a number of experimental studies with axenic cultures (Herbert et al 1956, Malek and Fencl 1966), the Monod equation was found to accurately predict the steady states from the values of the flow rate and the influent substrate concentration. This has also been observed for heterogenous cultures with microbial populations of sewage origin (Gaudy Jr. et al 1967, Ramanathan and Gaudy 1969).

Furthermore, Aiba et al (1967) performed research with continuous culture of Azotobacter vinelandii and found that the Monod equation accurately predicted the transient behaviour of the process, as did D'Ans (1972) with a culture of Escherichia coli K12. Both measured cell dry weight by optical densitometry.

On the other hand, Mateles et al (1965), working with Escherichia coli B, observed large discrepancies between Monod predictions and experimental data. For example, contrary to Monod prediction, there were overshoots in the variations of bacterial and nitrogen concentrations. Mateles et al (1965) measured biomass concentration based on the determination of Kjeldahl nitrogen removed from the medium, apparently due to absorption by E. coli B. The amount of nitrogen removed

was assumed to be a measure of microbial growth.

However, Gillay and Bungay (1967), experimenting with the culture Saccharomyces cerevisiae grown on a nitrogen-limited synthetic medium, found that Mateles' technique led to serious discrepancies even for steady state predictions with the Monod equation. For various constant dilution rates they evaluated the corresponding microbial concentrations at steady states, based on cell numbers, dry weights and Kjeldahl nitrogen contents (Figure 1.2). At steady state, results based on cell dry weights and cell numbers agreed roughly with the Monod prediction. On the other hand, results obtained from the Kjeldahl nitrogen showed an unexpected overshoot when approaching the wash-out dilution rate. Thus, it is likely that the validity of the Monod equation for continuous cultivation dynamics may depend on the nature of micro-organisms studied, the medium and the methods of determining the microbial concentrations.

Nevertheless, a survey of various experimental results with some pure cultures made by Edwards (1970) tend to establish the validity of the Haldane-Monod equation in predicting steady states.

Other common equations used in place of the Haldane-Monod equation for the specific growth rate are:

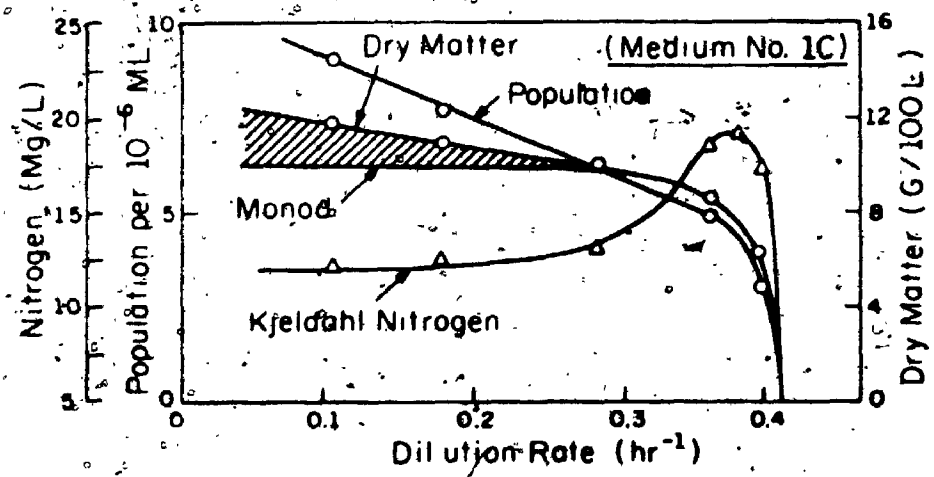


Figure 1:2 Steady-state properties at various dilution rate. From Gillay and Bungay (1967).

$$\mu(t) = \frac{\mu_m}{1 + aC(t)^{-b}} \qquad (1.6)$$

$$\mu(t) = aS(t) + b \qquad (1.7)$$

$$\mu(t) = aC(t) + b \qquad (1.8)$$

Eq. (1.6) was proposed by Moser (1958); Eq. (1.7) was referred to by Luedeking and Piret (1959), Humphrey (1963), and Takamatsu et al (1975); and Eq. (1.8) is the Verhulst-Pearl equation (Mohler 1973) which were used by Moraine and Rogovin (1966) and King et al (1974).

The specific substrate consumption rate $\nu(t)$ is often approximated as:

$$\nu(t) = \frac{\mu(t)}{Y_{C/S}} \qquad (1.9)$$

where $Y_{C/S}$ is defined as the yield coefficient of biomass based on the substrate.

The specific product formation rate $\rho(t)$ is often formulated as:

$$\rho(t) = Y_{P/C}\mu(t) \qquad (1.10)$$

where $Y_{P/C}$ is defined as the yield coefficient of the product formed based on the biomass.

There are many other expressions for $\rho(t)$. For

instance, one that was used recently in amino acid fermentation industry (Takamatsu et al 1975) is a quadratic equation in substrate concentration,

$$P(t) = aS(t)^2 + bS(t) + c \qquad (1.11)$$

It is often found that the product concentration $P(t)$ is more or less proportional to the biomass concentration $C(t)$. Therefore, for simplification in many cases, only Eqs. (1.1) and (1.2) are considered as the mathematical equations representing a continuous process.

II VECTOR AND MATRIX NOTATIONS

The mathematical notations used in the thesis are as follows:

A matrix is an underlined and capitalized letter, say $\underline{A}$.

A vector is a matrix with one column and is denoted by an underlined small letter, say $\underline{a}$.

A scalar is a matrix with one element and is denoted by either a small or capitalized letter, say a, A.

$\|\underline{x}\|^2_{\underline{A}}$ is a short notation for a quadratic $\underline{x}^T\underline{A}\underline{x}$ where the superscript T denotes transpose operator.

If $\underline{A}$ is a $(m \times n)$-matrix, a matrix with m rows and n columns, and $\underline{B}$ is a $(p \times q)$-matrix, then by the definition here, the derivative of $\underline{A}$ with respect to $\underline{B}$ is:

$$\frac{d\underline{A}}{d\underline{B}}_{(mp \times nq)} = \begin{bmatrix} \frac{d\underline{A}}{db_{11}} & \frac{d\underline{A}}{db_{12}} & \cdots & \frac{d\underline{A}}{db_{1q}} \\ \frac{d\underline{A}}{db_{21}} & & & \\ \vdots & & & \\ \frac{d\underline{A}}{db_{p1}} & & & \frac{d\underline{A}}{db_{pq}} \end{bmatrix}$$

where,

$$\frac{d\underline{A}}{db_{ij}}_{(m \times n)} = \begin{bmatrix} \frac{da_{11}}{db_{ij}} & \frac{da_{12}}{db_{ij}} & \cdots & \frac{da_{1n}}{db_{ij}} \\ \frac{da_{21}}{db_{ij}} & & & \\ \vdots & & & \\ \frac{da_{m1}}{db_{ij}} & & & \frac{da_{mn}}{db_{ij}} \end{bmatrix}$$

a_{ij} and b_{ij} are the (i,j)-elements of the matrices $\underline{A}$ and $\underline{B}$ respectively.

Note that $d\underline{a}/d\underline{b}$ has a special name - the Jacobian matrix of

of a with respect to b.

III CHAPTER SUMMARY

The thesis consists of six chapter and five appendicas as follows:

Chapter I is an introductory chapter that includes a brief literature survey of modern control applications to fermentation, a review of microbial growth theory, and vector and matrix notations used in the thesis.

In Chapter II, steady state effluent concentrations are computed for cases with and without recycle, based on the mathematical model derived in Chapter I. From these steady state concentrations, optimum operating conditions including temperature, pH, dilution rate, influent substrate concentration and recycle constants are determined.

Chapter III is a review of the most effective optimization techniques used in solving a general system identification problem by least squares. They include Hartley 's Modified Gauss-Newton, Marquardt's Algorithm and Bellman's Quasilinearization. A batch fermentation of Aureobasidium pullulans and a continuous fermentation of Morchella crassipes are then particularly identified.

In Chapter IV, a general dynamic batch process identification problem is formulated. Difficulty in solving the batch process identification problem is then stated. When the model is linear in its state vector, a technique to overcome the difficulty is also derived and tested with simulated data and experimental data from batch fermentation of A. pullulans. Based on this technique, an on-line identification scheme to update the model after receiving data from new batch runs is proposed.

A stochastic model for a continuous fermentation process is developed in Chapter V. It takes into consideration the measurement uncertainty and the disturbances due to imperfect mixing of both the bioreactor and the substrate feeding tank. Based on the model, a stochastic regulator problem of effluent concentrations is formulated and solved, using the Separation Theorem.

Chapter VI is the concluding chapter. It summarizes and comments on the present research work and suggests future research projects.

The proof of the equivalence of the two well-known system identification techniques: Gauss-Newton and Quasi-linearization, is developed in Appendix A.

Materials and methods required for the cultivation

of A. pullulans, the experimental data being used in Chapter III, are summarized in Appendix B.

In Appendix C, confidence intervals of parameters and the backward elimination technique used in Chapter II are reviewed.

Appendix D contains proofs of the optimum linear filter (Kalman filter) and of the Separation Theorem used in Chapter V.

Finally, Appendix E gives the Fortran listings of some important computer programs used in the thesis.

CHAPTER II

OPTIMUM OPERATING CONDITIONS FOR A CONTINUOUS FERMENTATION PROCESS - WITH AND WITHOUT RECYCLE

The performance of a continuous fermentation process depends on the operating conditions of the bioreactor. Based on the knowledge of the process, an engineer must decide what operating conditions will give the best performance. The purpose of this chapter is to show how such a decision can be made when the performance is defined as the production rate of biomass. A problem without recycle will be considered first, followed by another with recycle of both biomass and substrate.

I WITHOUT RECYCLE

I.1 Mathematical Model

From Chapter I, the mathematical equations representing a continuous fermentation process are:

$$\dot{C}(t) = -[\mu(t) - K - D]C_c(t) \qquad (2.1)$$

Initial condition: $C(t_o) = C_o$

$$\dot{S}(t) = D[S_I - S(t)] - \mu(t)C(t)/Y \qquad (2.2)$$

Initial condition: $S(t_o) = S_o$

$$\mu(t) = \frac{\mu_m}{1 + \frac{K_s}{S(t)} + \frac{S(t)}{K_i}} \qquad (2.3)$$

Note that for notational simplicity a dot operation "." has been used in place of "$\frac{d}{dt}$" and Y in place of $Y_{C/S}$.

Eqs. (2.1) to (2.3) form a set of nonlinear differential equations which cannot be solved analytically by classical methods, but can be solved graphically or numerically using either an analog computer or a digital computer with some appropriate numerical integration technique, viz., Euler, Runge-Kutta (Haggerty 1972). In most of modern digital computers used in industries or universities, there are "built-in" programs to solve sets of differential equations such as CSMP, MIMIC, IMAGE, etc.... For example, if CSMP (Continuous System Modeling Program) on an IBM-1130 computer is used, then Eqs. (2.1) to (2.3) are represented as in Figure 2.1. If the values established for a fermentation process are $D = .333\ hr^{-1}$, $S_I = 5.\ g/l$, $\mu_m = 1.\ hr^{-1}$, $K = 0.\ hr^{-1}$, $K_s = .03\ g/l$, $K_i = 2.\ g/l$, $Y = .5$, $C_o = .5\ g/l$ and $S_o = 0.\ g/l$, then the responses of C(t), S(t) and $\mu(t)$ are shown on Figures 2.2 to 2.4 (Note that these values are the same as those used by Andrews (1968)). Figure 2.2 shows that the biomass concentration S(t) increased from .5 g/l to "reach" its steady state level of 2.49 g/l in about 15 hours.

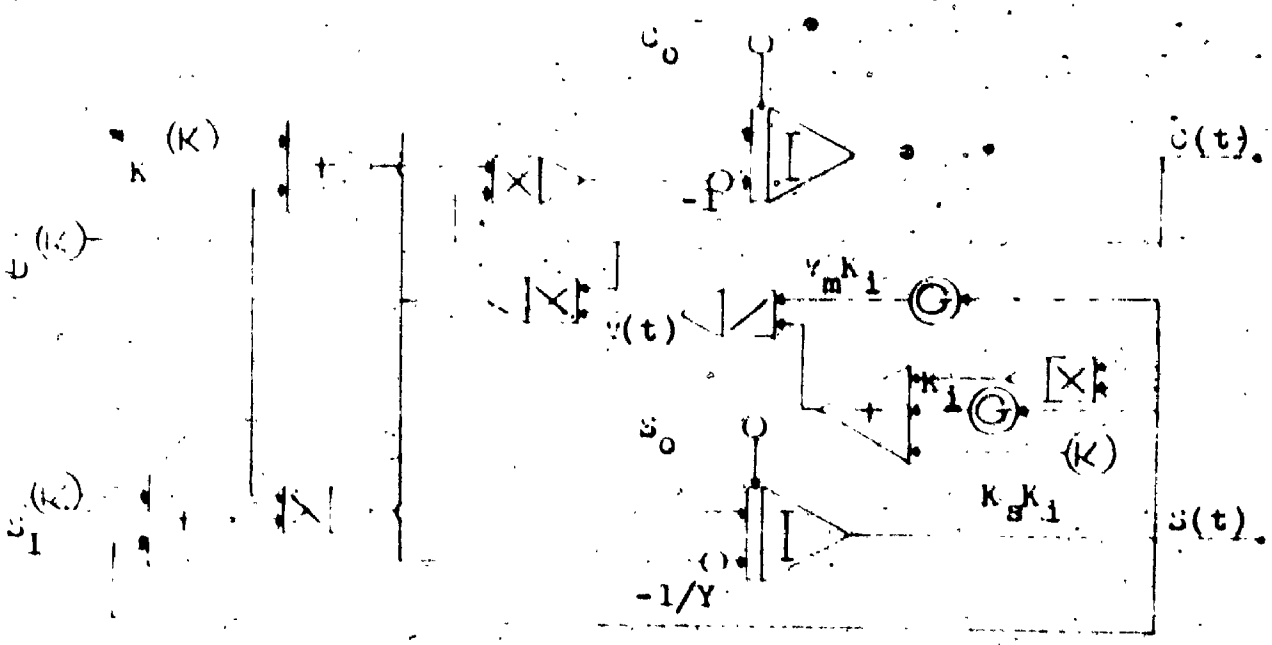
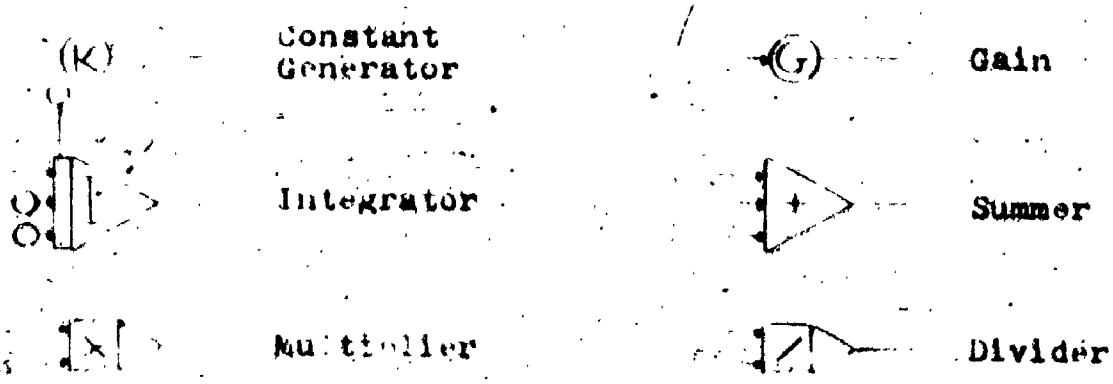


Figure 2.1 CSMP diagram representing the continuous fermentation process described by Equations (2.1) to (2.3). The symbolism used in Figure 2.1 is defined below:



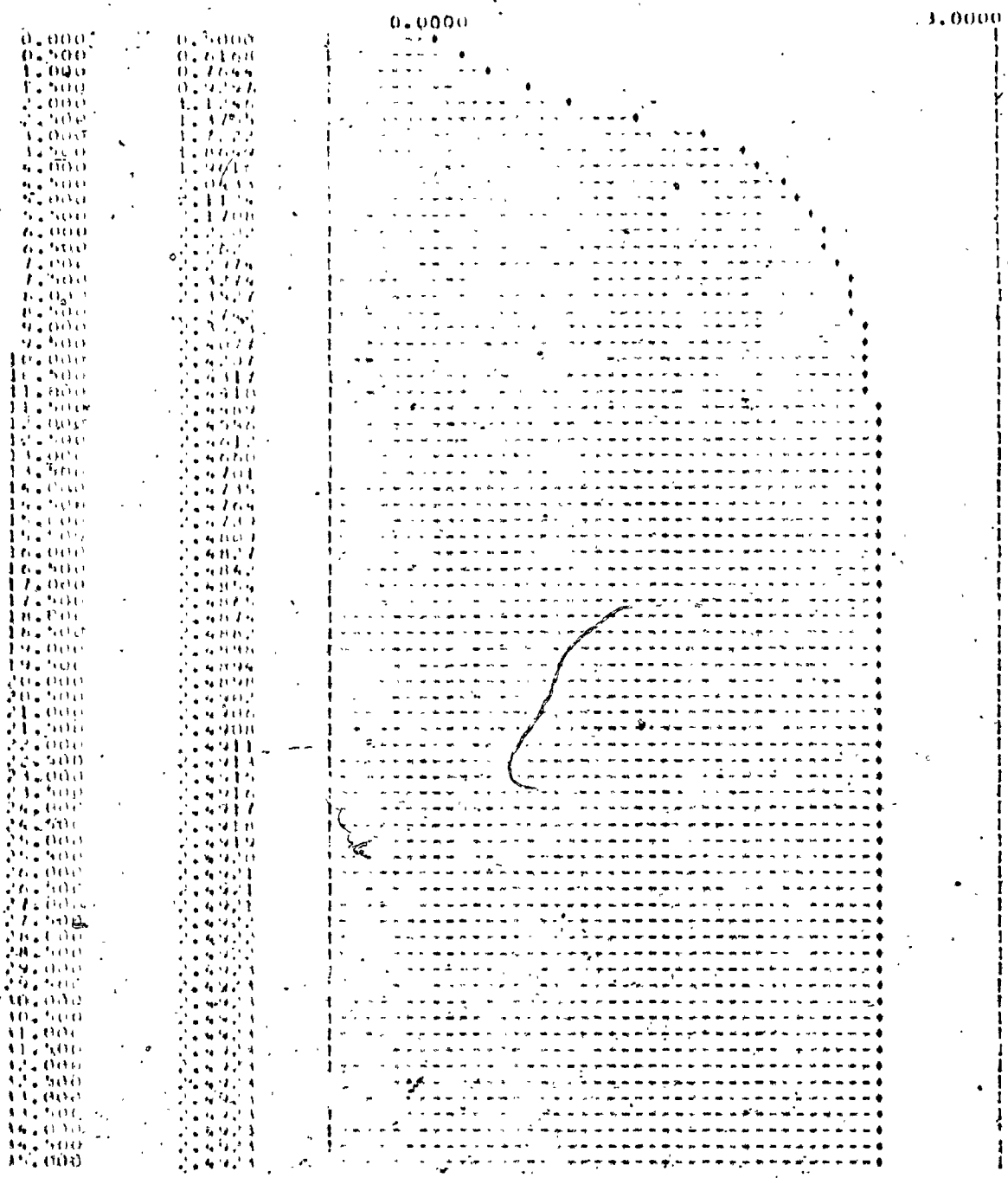


Figure 2.2 Biomass concentration C(t) versus time t for S_f = 5 g/l, μ_m = 1 hr^{-1}, K = 0 hr^{-1}, k_s = .03 g/l, K_i = 2 g/l, Y = .5, C_0 = .5 g/l and C_r = 0 g/l. - Without recycle.

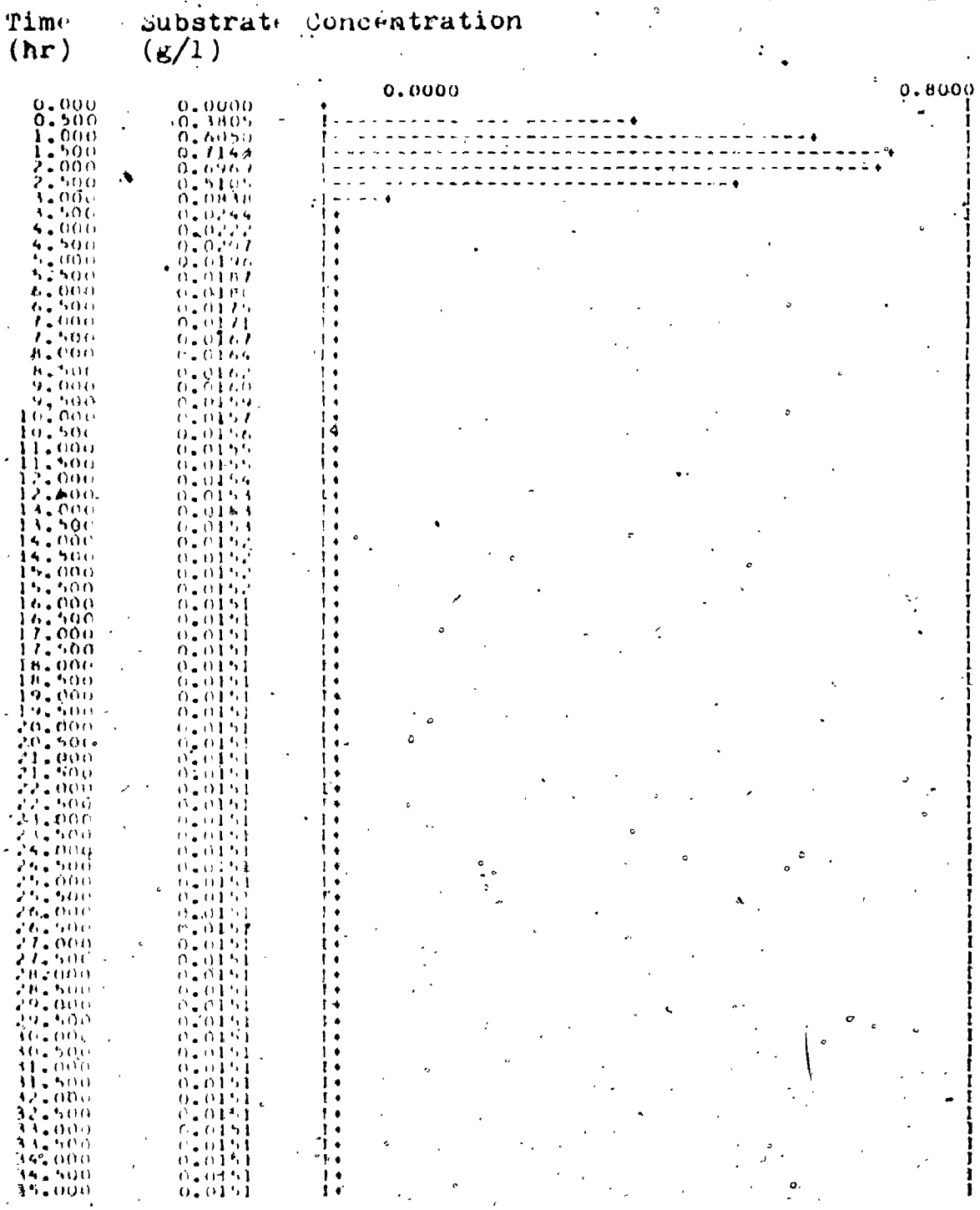


Figure 2.3 Substrate concentration S(t) versus time t for $S_1 = 5$ g/l, $\mu_m = 1$ hr^{-1}, $K_d = 0$ hr^{-1}, $K_s = .03$ g/l, $K_i = 2$ g/l, $Y = .5$, $C_o = .5$ g/l and $S_o = 0$ g/l. - Without recycle.

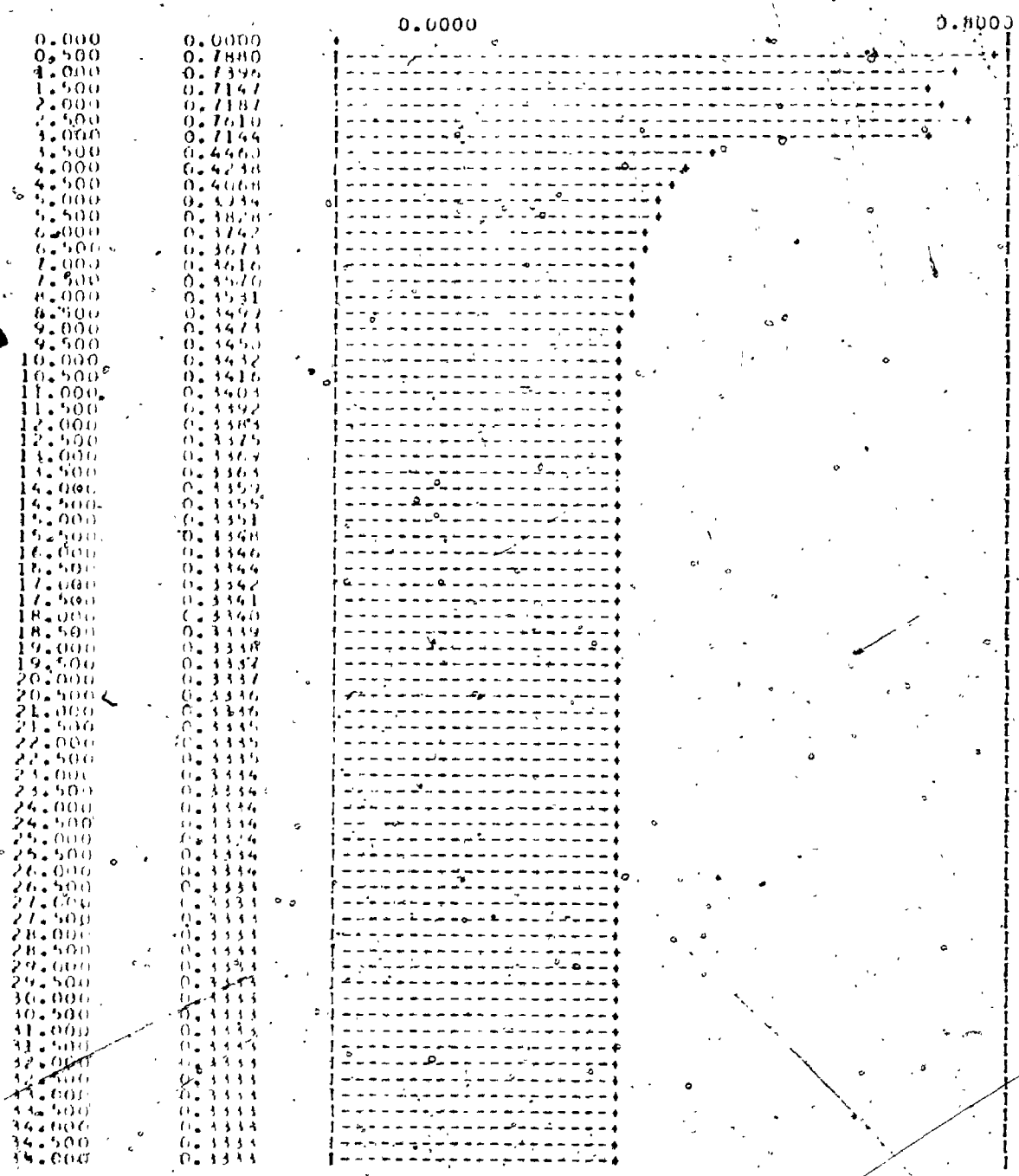


Figure 2.4 Specific growth rate $\mu(t)$ versus time t for S_I = 5 g/l, μ_m = 1 hr^{-1}, k = 0 hr^{-1}, K_s = .03 g/l, K_i = 2 g/l, Y = .5, C_o = .5 g/l and S_o = 0 g/l, - without recycle.

From Figure 2.3, initially due to low biomass in the bio-reactor, the substrate concentration was increasing then as microbes grew and consumed more substrate, the substrate concentration decreased and after 15 hours reached a steady state level of 1.51 g/l. Figure 2.4 shows that the specific growth rate was initially high and increasing due to low biomass concentration; then because microbe population increased but the rate of substrate supplied was still the same, the specific growth rate decreased and finally settled at the level of .33 hr^{-1}.

I.2 Steady State Values

The steady state value of a function A(t), if it exists, is by definition the limit of A(t) when t goes to infinty.

$$A_{ss} = \text{Steady state value of } A(t) = \lim_{t \to \infty} A(t)$$

At steady state, A(t) becomes A_{ss}, a constant, that is, its rate of change (its derivative with respect to t) is zero. In general, it is not easy to find A_{ss} directly from its definition because the analytical form of A(t) is either complicated or unavailable. Therefore, in practice the steady state value is often found by setting the derivative equal to zero, however, it must be done with care since local maxima,

local minima and points of inflection may be encountered when the derivative is zero.

Let the steady state values of C(t), S(t) and $\mu(t)$ be denoted by C_{ss}, S_{ss} and μ_{ss}, then from Eqs. (2.1), (2.2) and (2.3), if they exist, they must satisfy the following equations:

$$\left[\mu_{ss} - K - D\right]C_{ss} = 0 \qquad (2.4)$$

$$D(S_I - S_{ss}) - \mu_{ss}C_{ss}/Y = 0 \qquad (2.5)$$

$$\mu_{ss} = \frac{\mu_m}{1 + \frac{K_s}{S_{ss}} + \frac{S_{ss}}{K_i}} \qquad (2.6)$$

If Eq. (2.4) holds, then either of the following must also hold:

$$C_{ss} = 0 \qquad (2.7)$$

$$\mu_{ss} - K - D = 0 \qquad (2.8)$$

For $C_{ss} = 0$, then from Eq. (2.5), $S_{ss} = S_I$. These steady state values are undesirable, the situation for which they exist is called "wash-out" where all the biomass is being constantly washed away and will occur if the flow rate is set too high. Eqs (2.6) and (2.8) give:

$$\frac{\mu_m}{1 + \frac{K_s}{S_{ss}} + \frac{S_{ss}}{K_i}} - K - D = 0.$$

or,

$$S_{ss}^2 + K_i\left(1 - \frac{\mu_m}{D + K}\right)S_{ss} + K_i K_s = 0$$

or,

$$S_{ss} = \frac{K_i}{2}\left[\left(\frac{\mu_m}{D + K} - 1\right) \pm \sqrt{\left(\frac{\mu_m}{D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \qquad (2.9)$$

For a realizable steady state, S_{ss} must be meaningful, i.e., S_{ss} found in Eq. (2.9) must be both real and non-negative. To achieve this, the following constraints must be satisfied:

$$\left(\frac{\mu_m}{D + K} - 1\right)^2 - 4\frac{K_s}{K_i} \geq 0 \qquad (2.10)$$

$$\left(\frac{\mu_m}{D + K} - 1\right) > 0 \qquad (2.11)$$

Combining the two constraints together results in:

$$\left(\frac{\mu_m}{D + K} - 1\right) \geq 2\sqrt{K_s/K_i}$$

or,

$$D \leq \frac{\mu_m}{1 + 2\sqrt{K_s/K_i}} - K \qquad (2.12)$$

From Eqs. (2.5) and (2.8), the steady state value for biomass concentration is:

$$C_{ss} = \frac{YD}{D + K}(S_I - S_{ss}) \qquad (2.13)$$

Since $C_{ss} \geq 0$, from the above equation, the following constraint must also be satisfied:

$$S_I - S_{ss} \geq 0 \qquad (2.14)$$

In summary, there are three different steady state points:

- $S_{1ss} = S_I$ and $C_{1ss} = 0$,
- $S_{2ss} = \frac{K_i}{2}\left[\left(\frac{\mu_m}{D+K} - 1\right) + \sqrt{\left(\frac{\mu_m}{D+K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right]$ and

  $C_{2ss} = \frac{YD}{D+K}(S_I - S_{2ss})$,
- $S_{3ss} = \frac{K_i}{2}\left[\left(\frac{\mu_m}{D+K} - 1\right) - \sqrt{\left(\frac{\mu_m}{D+K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right]$ and

  $C_{3ss} = \frac{YD}{D+K}(S_I - S_{3ss})$.

The first one is the wash-out point, the second and third ones were found by D'Ans (1972), using Lyapunov's 2nd method, to be unstable and stable, respectively. Therefore, a chemostat should operate at the third steady state point, (S_{3ss}, C_{3ss}) and the following constraints must be satisfied:

$$D \leq \frac{\mu_m}{1 + 2K_s/K_i} - K \qquad (2.12)$$

$$D \geq 0 \qquad (2.15)$$

$$S_I - \frac{K_i}{2}\left[\left(\frac{\mu_m}{D+K} - 1\right) - \sqrt{\left(\frac{\mu_m}{D+K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \geq 0 \qquad (2.16)$$

If, in particular, $K_i \to \infty$, i.e., the specific

growth rate $\mu(t)$ follows the Monod equation, Eq. (2.17),

$$\mu(t) = \frac{\mu_m S(t)}{S(t) + K_s} \qquad (2.17)$$

then, it follows from Eqs. (2.1), (2.2) and (2.17) that there is only one steady state point besides the wash-out point:

$$S_{ss} = \frac{K_s(D + K)}{\mu_m - D - K} \qquad (2.18)$$

$$C_{ss} = \frac{YD(S_I - S_{ss})}{D + K} \qquad (2.19)$$

Since D, S_{ss} and C_{ss} must be non-negative, the following constraints are imposed:

$$D \geq 0 \qquad (2.15)$$

$$\mu_m - D - K > 0 \qquad (2.20)$$

$$S_I - S_{ss} \geq 0 \qquad (2.21)$$

Note that Eqs. (2.20) and (2.21) can be combined together via Eq. (2.18) to form just one constraint, shown in Eq. (2.22):

$$D < \frac{\mu_m S_I}{S_I + K_s} - K \qquad (2.22)$$

I.3 Optimum Operating Conditions

For economical reasons, it is usually preferable to produce as much product and/or biomass as possible. Since

product is often treated as a quantity which is more or less proportional to biomass, mathematically it is desirable to maximize

$$J_1 \quad VC(t_f) + \int_{t_o}^{t_f} FC(t)dt \qquad (2.23)$$

where (t_o, t_f) is the operating period starting at time t_o and ending at time t_f, and the first and second terms in the right hand side of the equation are the amount of biomass in the bioreactor at time t_f and the amount of biomass collected at the collecting tank respectively.

In general, the operating period is very long, so that the above optimal control problem (maximization of J_1 subject to Eqs. (2.1) to (2.3)) can be approximately reduced to the following nonlinear programming problem:

$$\text{Max. } J_2 = DC_{ss} \qquad (2.24)$$

subject to:

$$C_{ss} = \frac{YD}{D + K}(S_I - S_{ss}) \qquad (2.25)$$

$$S_{ss} = \frac{K_i}{2}\left[\left(\frac{\mu_m}{D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \qquad (2.26)$$

$$D + K - \frac{\mu_m}{1 + 2\sqrt{K_s/K_i}} \leq 0 \qquad (2.12)$$

$$D \geq 0 \qquad (2.15)$$

$$S_I - \frac{K_i}{2}\left[\left(\frac{\mu_m}{D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \geq 0 \qquad (2.16)$$

The maximization of J_2 is often done with respect to dilution rate D and/or influent substrate concentration S_I. However, if some other optimum operating conditions, say temperature T and pH H are also sought, further constraints must be needed to describe the relationship between the model coefficients (μ_m, K_s, K_i, K and Y) and these operating conditions (T and H) such as Eqs. (2.27) to (2.31),

$$\mu_m = f_1(T,H) \qquad (2.27)$$

$$K_s = f_2(T,H) \qquad (2.28)$$

$$K_i = f_3(T,H) \qquad (2.29)$$

$$K = f_4(T,H) \qquad (2.30)$$

$$Y = f_5(T,H) \qquad (2.31)$$

The stated nonlinear programming problem can be readily solved by various iterative search techniques available in optimization literature (Polak 1971, Himmelblau 1972, Cooper and Steinberg 1970).

It is noted that from Eqs. (2.24) and (2.25), to maximize J_2, S_I should be chosen as large as possible, that is infinitive.

However, for a bioreaction to exist and to follow Eqs. (2.1) to (2.3), the medium cannot be too concentrated, i.e., there is an upper limiting value that S_I cannot exceed.

For the case where the specific growth rate $\mu(t)$ follows the Monod equation, Eq. (2.17), then the nonlinear programming problem encountered is to minimize J_2 defined by Eq. (2.24) and subject to Eqs. (2.18), (2.19), (2.15), (2.22) and (2.27) to (2.31). After simplification, it becomes,

$$\text{Max. } J_2 = \frac{a_1(T,H)D^2\left[D - a_2(T,H)\right]}{\left[D - a_3(T,H)\right]\left[D - a_4(T,H)\right]} \qquad (2.32)$$

subject to:

$$a_1(T,H) \quad f_5(T,H)\left[S_I + f_2(T,H)\right] \qquad (2.33)$$

$$a_2(T,H) \quad \frac{f_1(T,H)S_I}{S_I + f_2(T,H)} - f_4(T,H) \qquad (2.34)$$

$$a_3(T,H) \quad - f_4(T,H) \qquad (2.35)$$

$$a_4(T,H) \quad f_1(T,H) - f_4(T,H) \qquad (2.36)$$

$$D \quad 0 \qquad (2.15)$$

$$D + f_4(T,H) - \frac{f_1(T,H)S_I}{S_I + f_2(T,H)} \leq 0 \qquad (2.22)$$

When T and H are held fixed, the plot of J_2 versus D is shown in Figure 2.5. where $D_1(T,H)$, $D_2(T,H)$, $D_3(T,H)$ and $D_4(T,H)$ are the roots of the following equation.

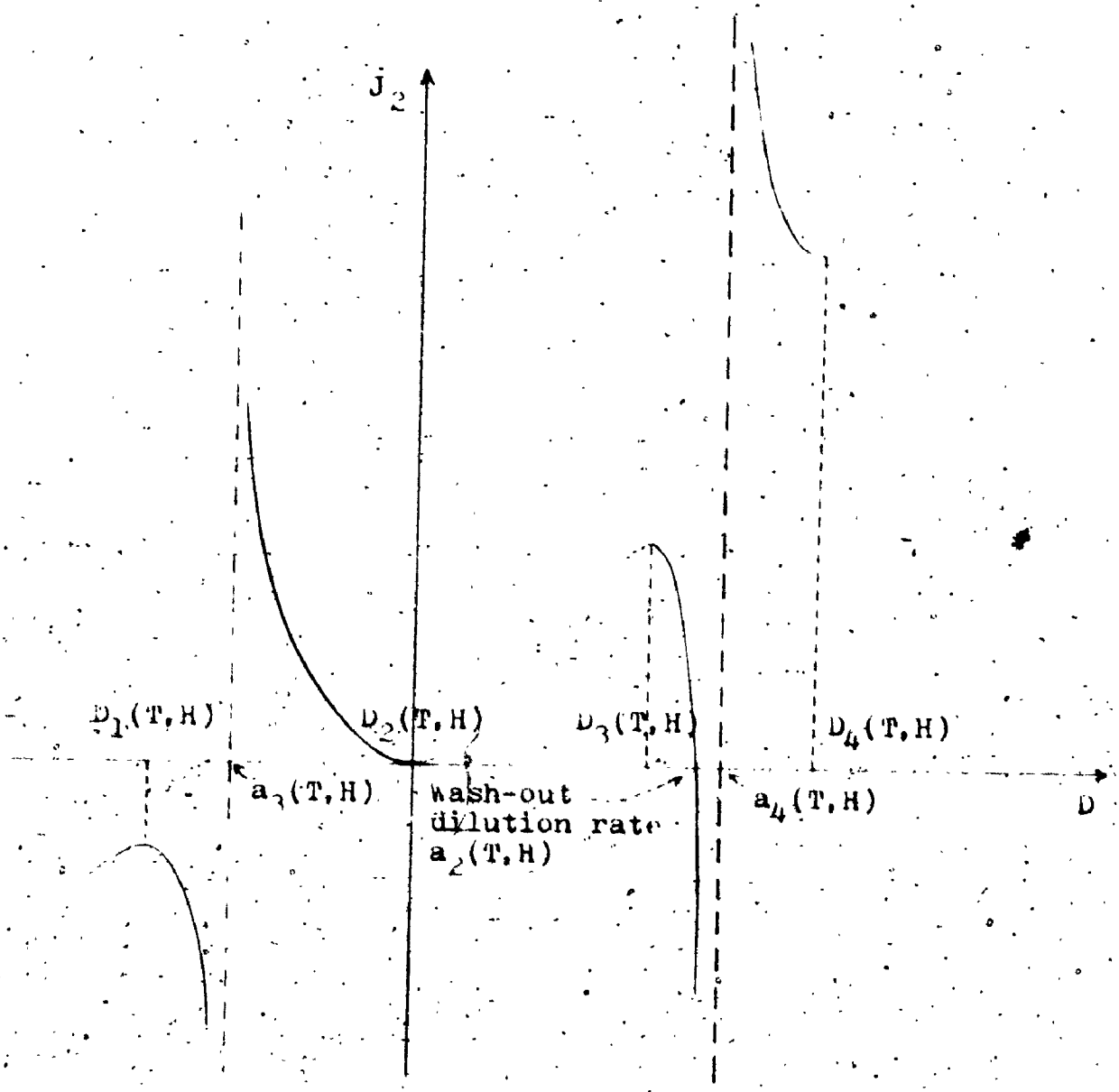


Figure 2.5 Plot of the performance index J_2 defined by Equation (2.32) versus dilution rate D when temperature T and pH H are held-fixed.

$$\frac{\partial J_3}{\partial D} = 0 \qquad (2.37)$$

or after taking the derivative and then simplifying:

$$D_3(T,H) = 0 \qquad (2.38)$$

$D_1(T,H)$, $D_3(T,H)$ and $D_4(T,H)$ are the roots of Eq. (2.39),

$$D^3 - 2\left[a_3(T,H) + a_4(T,H)\right]D^2 + \left\{3a_3(T,H)a_4(T,H) - a_2(T,H)\left[a_3(T,H) + a_4(T,H)\right]\right\}D - 2a_2(T,H)a_3(T,H)a_4(T,H) = 0 \qquad (2.39)$$

From Figure 2.5, it is clear that for fixed T and H, the optimum dilution rate should be chosen as $D^* = D_3(T,H)$. Thus, the non-linear programming is further reduced to:

$$\underset{T,H}{\text{Max.}}\ J_3 = \frac{a_1(T,H)D_3^2(T,H)\left[D_3(T,H) - a_2(T,H)\right]}{\left[D_3(T,H) - a_3(T,H)\right]\left[D_3(T,H) - a_4(T,H)\right]} \qquad (2.40)$$

$D_3(T,H)$ is a solution of Eq. (2.39) and generally cannot be computed analytically. Therefore, the optimum temperature T^* and the optimum pH H^* that maximize J_3 defined by Eq. (2.40) cannot be found analytically either. However, numerical search techniques on a digital computer can be used to find numerical solutions for the optimum temperature, pH and dilution rate, T^*, H^* and $D_3(T^*,H^*)$. The exhaustive search technique (Cooper and Steinberg 1970) is suggested for maximizing J_3 with respect to T and H, and the Newton-Raphson

search technique (Polak 1971) is suggested for computing $D_1(T,H)$ when T and H are given.

The Exhaustive Search Technique

Most of the search techniques which can be employed to maximize J_1 with respect to T and H such as Fletcher-Powell and Steepest Ascent (Cooper and Steinberg 1970) require computation of the first partial derivatives of J_1 with respect to T and H and produce only a local minimum of J_1. From Eq. (2.40), the partial derivatives of J_1 cannot be computed exactly or analytically owing to the fact that $D_1(T,H)$ cannot be computed analytically. Also the approximations of the partial derivatives by finite difference method consumes quite a bit of computer time. It is the reason why the exhaustive search technique was suggested. Moreover, since the dimension of the optimization problem is only two, and the optimum solution is often known to be lying in a certain small region(i.e., $T_{min} \leq T^* \leq T_{max}$, $H_{min} \leq H^* \leq H_{max}$, T_{min}, T_{max}, H_{min} and H_{max} are known; and $T_{max} - T_{min}$ and $H_{max} - H_{min}$ are not really large), the exhaustive search technique does not require many evaluations of J_1 at various values of T and H.

From the exhaustive search technique, the temperature-pH region is gridded as shown in Figure 2.6. For every grid point, there corresponds a temperature and a pH, and hence

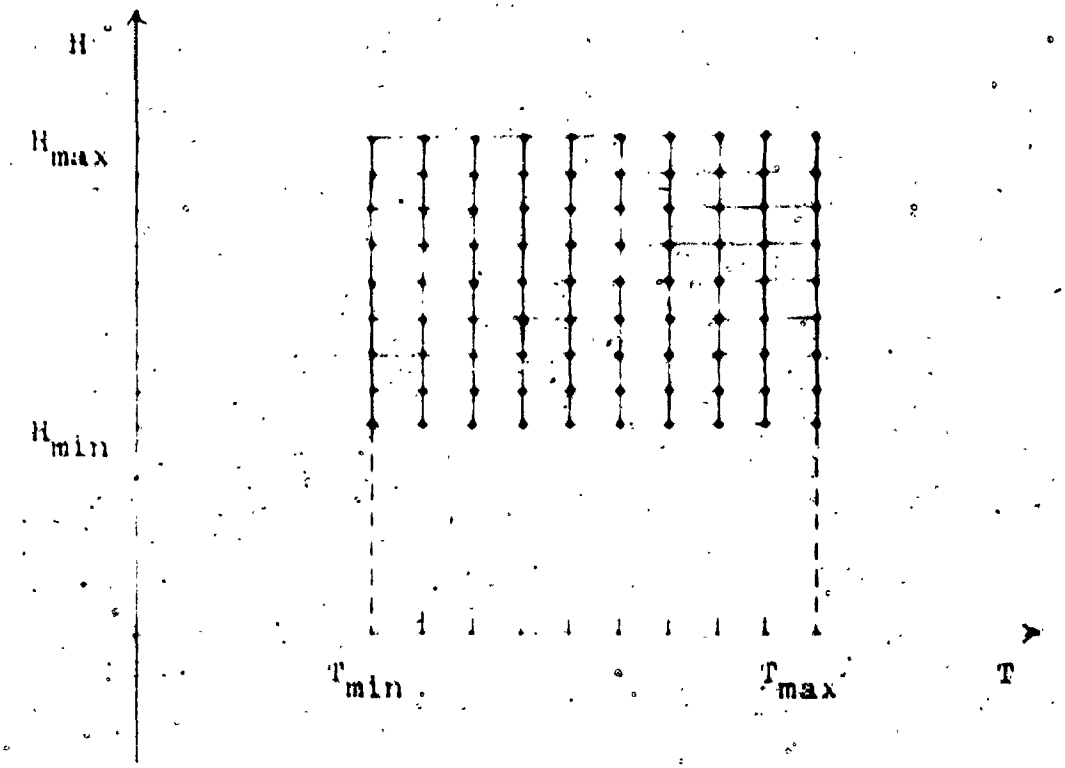


Figure 2.6 Temperature-pH grid in the exhaustive search technique.

a value of J_3. The optimum operating temperature T^* and pH H^* are the values of T and H which yield the highest value of J_3, and the optimum dilution rate D^* is calculated as $D_3(T^*,H^*)$.

Note that in order to determine accurately the optimum temperature and pH, the grid temperature interval and the grid pH interval should be significantly small. That is, one may have to compute J_3 many times. To reduce the number of computations, the grid intervals are first selected to be reasonably large. Rough T^* and H^* are then computed. A new grid is formed around the point (T^*,H^*), but now with smaller grid intervals. Better T^* and H^* are then computed. This procedure is repeated until values of T^* and H^* are within the acceptable accuracy.

The Newton-Raphson Technique

There are many good search techniques which can be used to compute $D_3(T,H)$ and hence J_3 for given T and H such as the half interval search technique and Reguli Falsi (chord) search technique (Cooper and Steinberg 1970). The Newton-Raphson was suggested due to its fast convergence (quadratic convergence) and the ease of derivative computation when dealing with this problem.

As $D_1(T,H)$ and $D_4(T,H)$ are also solutions of Eq.

(2.39), to guarantee that $D_3(T,H)$ is found instead of $D_1(T,H)$ and $D_4(T,H)$, the initial approximation of $D_3(T,H)$ (required by the Newton-Raphson technique) must be close to $D_3(T,H)$. From Figure 2.5, $a_4(T,H)/2$ should be significantly close to $D_3(T,H)$ and should be chosen as the initial approximation. From the Newton-Raphson technique, the value of $D_3(T,H)$ is found as follows:

$$D_{3N}(T,H) = D_{30}(T,H) - f(T,H,D)/\frac{\partial f(T,H,D)}{\partial D}\Bigg|_{\text{at } D = D_{30}(T,H)} \qquad (2.41)$$

where $f(T,H,D)$ is the left hand side of Eq. (2.39),

$$f(T,H,D) = D^3 - 2\left[a_3(T,H) + a_4(T,H)\right]D^2 + \left\{3a_3(T,H)a_4(T,H) - a_2(T,H)\left[a_3(T,H) + a_4(T,H)\right]\right\}D - 2a_2(T,H)a_3(T,H)a_4(T,H) \qquad (2.42)$$

and $D_{30}(T,H)$ is initially chosen as $a_4(T,H)/2$.

The value $D_{3N}(T,H)$ found in Eq. (2.41) is then used as the new $D_{30}(T,H)$ for the next iteration. This procedure is repeated until there is no significant difference in $D_{3N}(T,H)$ and $D_{30}(T,H)$, that is until some test of convergence is met. The convergence value is then considered as $D_3(T,H)$.

There are many convergence tests that can be used (see Chapter III). Eq. (2.43) represents a very popular test,

$$\left|D_{30}(T,H) - D_{3N}(T,H)\right| \leq \alpha \left|D_{30}(T,H)\right| \qquad (2.43)$$

where α is a small positive number, say 10^{-4}. The smaller α is, the more accurate $D_3(T,H)$ becomes, however a longer computing time is required.

The Newton-Raphson technique is illustrated in Figure 2.7.

In literature, the coefficient $K = f_4(T,H)$ is very often neglected. When this is acceptable, the problem can be further simplified. J_2 defined in Eq. (2.32) is reduced to J_4,

$$J_4 = \frac{a_1(T,H)D[D - a_2(T,H)]}{D - a_4(T,H)} \qquad (2.44)$$

When T and H are held fixed, the plot of J_4 versus D is shown in Figure 2.8. Where $D_5(T,H)$ and $D_6(T,H)$ are the roots of Eq. (4.45),

$$\frac{\partial J_4}{\partial D} = 0 \qquad (2.45)$$

or,

$$D_5(T,H) = a_4(T,H)\left\{1 - \sqrt{[a_4(T,H) - a_2(T,H)]/a_4(T,H)}\right\} \qquad (2.46)$$

$$D_6(T,H) = a_4(T,H)\left\{1 + \sqrt{[a_4(T,H) - a_2(T,H)]/a_4(T,H)}\right\} \qquad (2.47)$$

From Figure 2.8, it is clear that for certain fixed T and D, the optimum dilution rate should be chosen as $D^* = D_5(T,H)$. The nonlinear programming defined by Eq. (2.40) is reduced to Eq. (2.48),

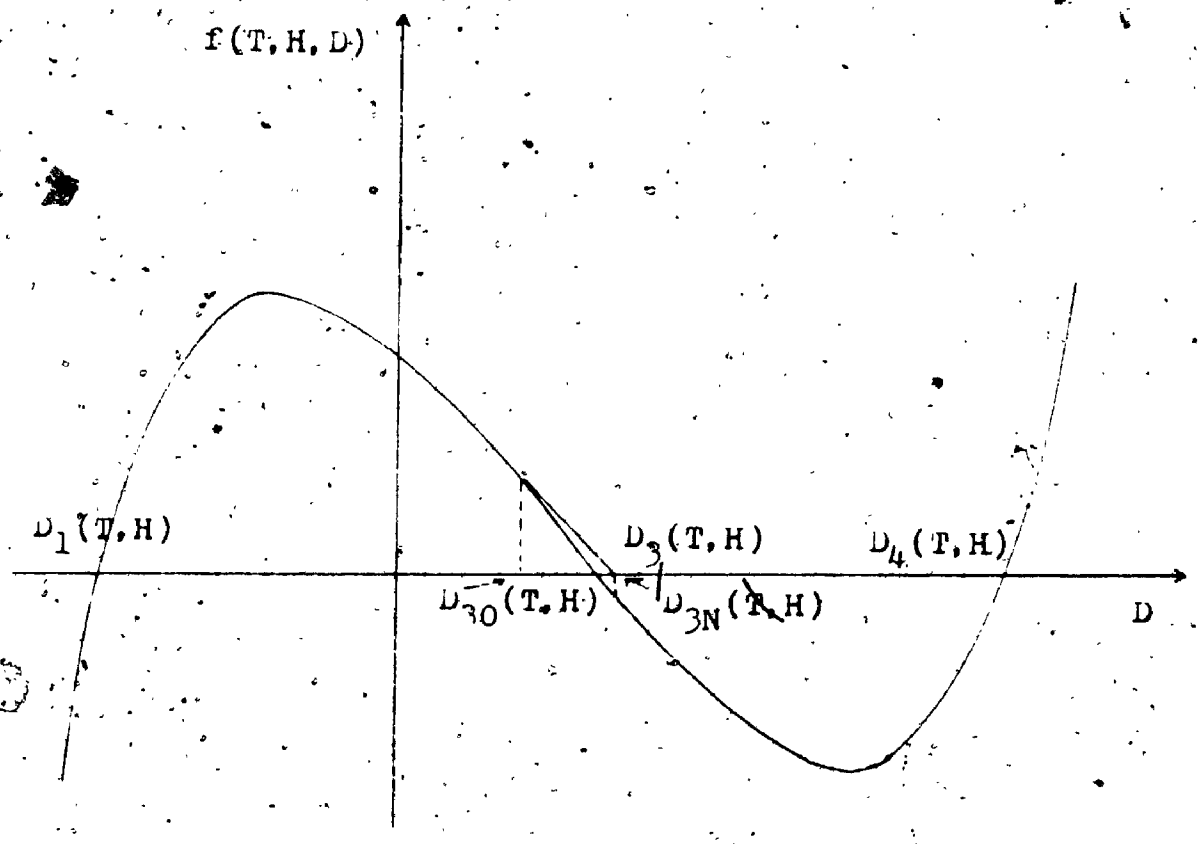


Figure 2.7 Illustration of the Newton-Raphson search technique.

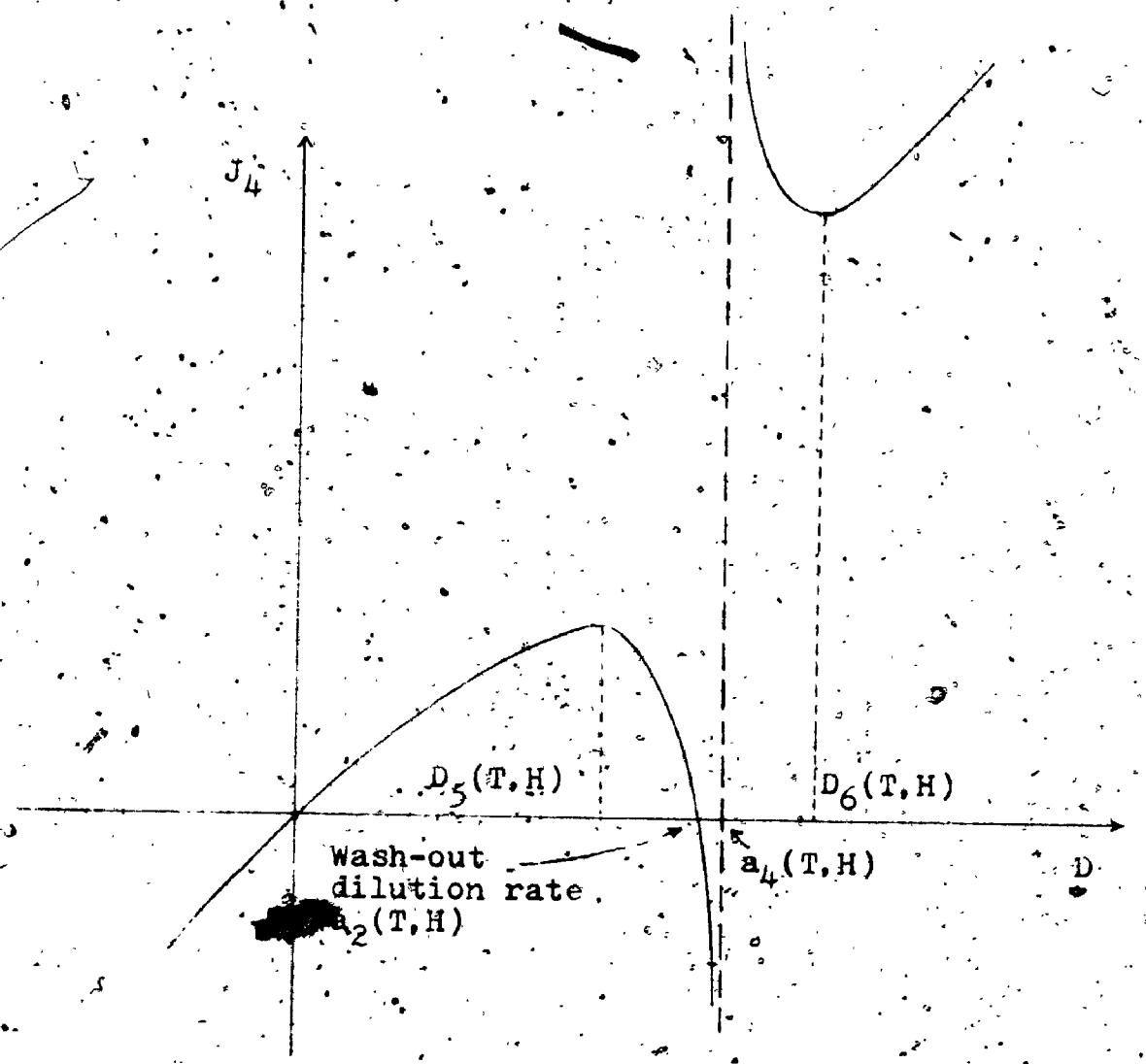


Figure 2.8 Plot of the performance index J_4 defined by equation (2.44) versus dilution rate D when temperature T and pH H are held fixed.

$$\underset{T,H}{\text{Max.}}\ J_5 = \frac{a_1(T,H)D_5(T,H)\left[D_5(T,H) - a_2(T,H)\right]}{D_5(T,H) - a_4(T,H)} \qquad (2.48)$$

where $D_5(T,H)$ is defined by Eq. (2.46).

I.4 Example

Topiwala and Sinclair (1971) presented a temperature relationship for a continuous fermentation process of Aerobacter aerogenes and stated that the relationship can be used to optimize production of biomass. However they did not indicate just how the relationship could be used. As an example, an optimum temperature and an optimum dilution rate were determined for the fermentation of A. aerogenes using the techniques mentioned in Section I.3 and the temperature relationship presented by Topiwala and Sinclair.

Topiwala and Sinclair found that the model coefficients do not really depend on the operating pH and are related to the operating temperature by the following equations:

$$\gamma_m = f_1(T) = 2.45 \times 10^{10}\, e^{-14230/RT} \quad (hr^{-1}) \qquad (2.49)$$

$$K_s = f_2(T) = \frac{1}{2.96} \times 10^{-10}\, e^{-11800/RT} \quad (g/l) \qquad (2.50)$$

$$K_i = f_3(T) = 00 \quad (g/l) \qquad (2.51)$$

$$K = f_4(T) = 2.70 \times 10^{5}\, e^{-9000/RT} \quad (hr^{-1}) \qquad (2.52)$$

$$Y = f_5(T) = .5 \text{ (unitless)} \qquad (2.53)$$

where,

R = The gas constant = 1.987 (cal/mole ^{o}K)

T = Temperature in Kelvin degree ^{o}K)

For various influent substrate concentration, S_I, a summary of results is shown on Table 2.1. From the table, it is noted that as the influent concentration S_I increases, the optimum temperature should be decreased and the optimum dilution rate should be increased to maintain the biomass production at maximum. Also, the changes in the optimum temperature are very small and for practical purpose, the optimum temperature can be taken as 36^{o}C. Furthermore, it is more economical to operate at higher influent substrate concentration S_I (if S_I is doubled, $J_2^* = D^* C_{ss}$ is more than doubled).

For a given S_I, say S_I = 3 g/l, the plot of J_3 versus T is shown as Figure 2.9 and the plot of J_2 versus D for $T = T^*$ = 36.07^{o}C is shown as Figure 2.10.

II <u>WITH RECYCLE OF BIOMASS AND SUBSTRATE</u>

Theoretical and experimental studies of the problems associated with biomass recycle have appeared in the literature (Herbert 1961, Ramathan and Gaudy 1969 and 1971). In these papers, however, little has been said about the optimum opera-

Table 2.1 Optimum operating temperature T^*, optimum dilution rate D^*, steady state substrate concentration S_{ss}, steady state biomass concentration C_{ss} and maximum rate of biomass production per volume of the bioreactor $J_2^* = D^* C_{ss}$ at various levels of the influent substrate concentration S_I for a continuous fermentation of A. aerogenes, using the Topiwala-Sinclair model.

S_I (g/l)	T^* (^oC)	$D^* = D_3(T^*)$ (hr^{-1})	S_{ss} (g/l)	C_{ss} (g/l)	$J_2^* = D^* C_{ss}$ (g/l/hr)
.1	36.80	.708	.024	.033	.023
1.0	36.22	.877	.089	.402	.352
2.0	36.12	.901	.129	.828	.746
3.0	36.07	.911	.160	1.258	1.146
4.0	36.05	.918	.186	1.692	1.553
5.0	36.03	.922	.208	2.126	1.961
10.0	35.99	.933	.298	4.312	4.023

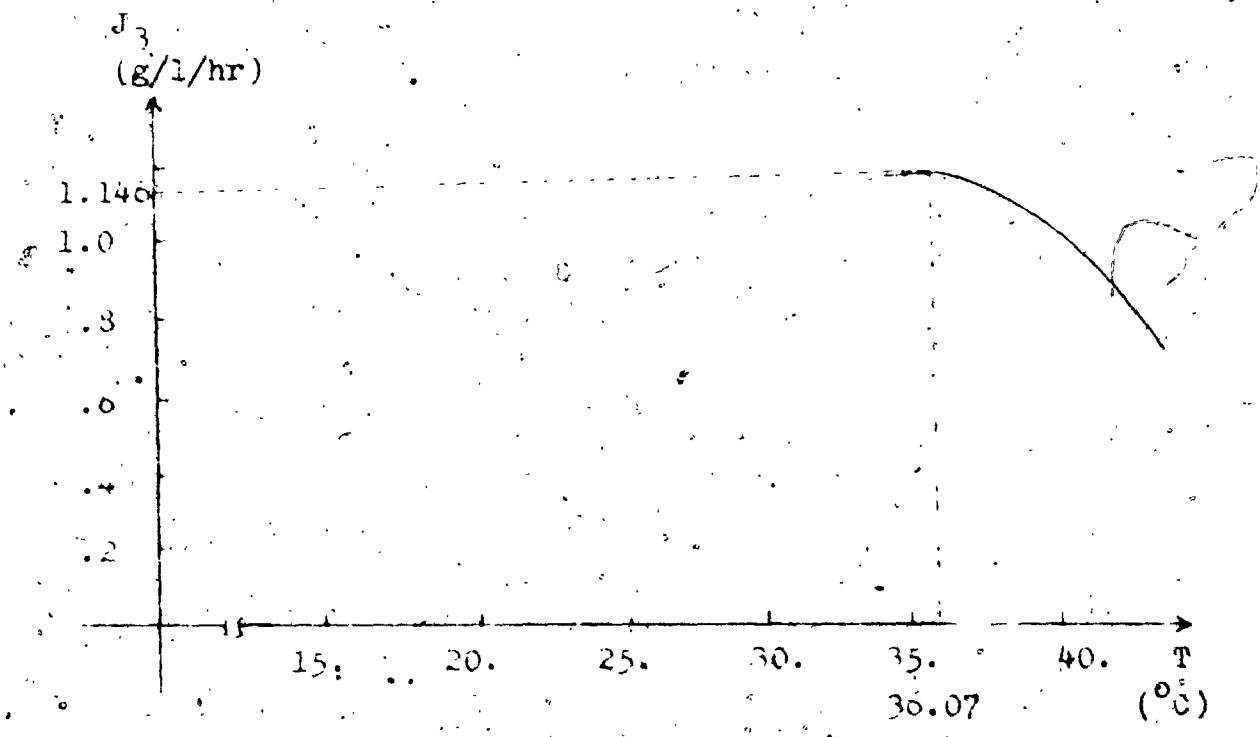


Figure 2.9 Plot of the performance index J_3 defined by equation (2.40) versus temperature T for a continuous fermentation of A. aerogenes, using the Topiwala-Sinclair model.

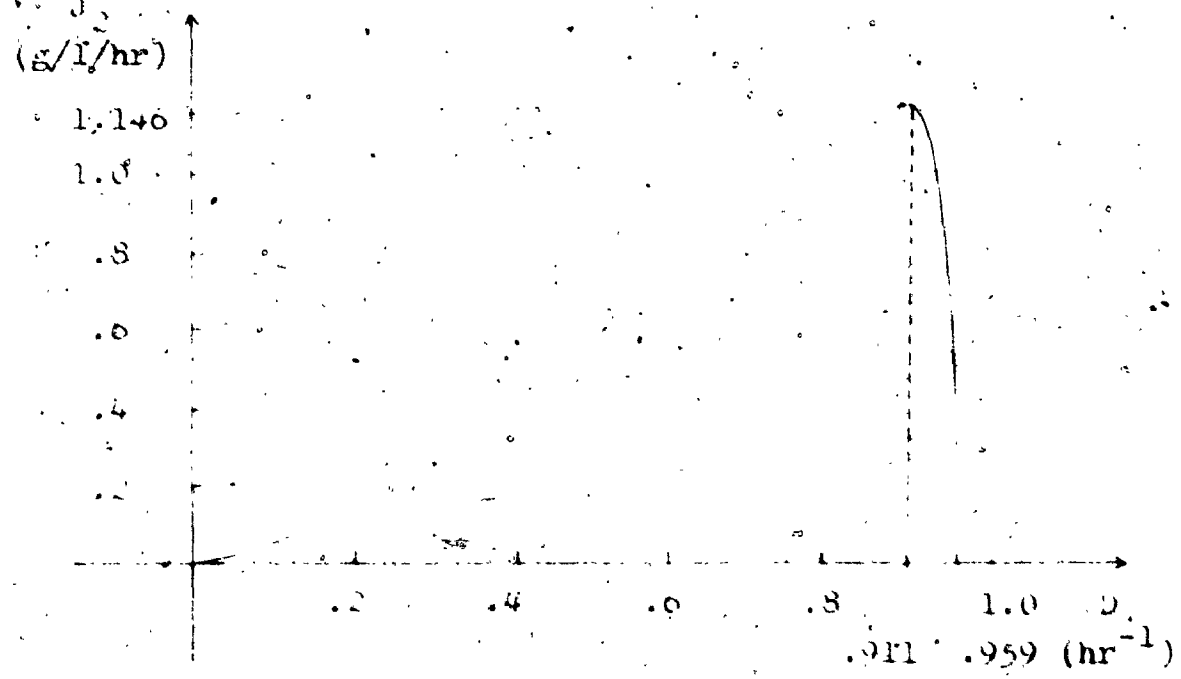


Figure 2.10 Plot of the performance index J_3 defined by equation (2.32) versus dilution rate D when temperature T = T^* = 30.07 °C for a continuous fermentation of A. aerogenes, using the Topiwala-Sinclair model.

ting conditions. In this section, a more complete theoretical investigation concerning the optimum operating conditions of a continuous fermentation process with recycle of both biomass and substrate as shown in Figure 2.11 will be considered.

II.1 Mathematical Model

Let

$$\alpha = \frac{F_R}{F} \tag{2.54}$$

$$\beta_C = \frac{C_R(t)}{C_T(t)} \tag{2.55}$$

$$\beta_S = \frac{S_R(t)}{S_T(t)} \tag{2.56}$$

$$A_C = 1 + \alpha - \alpha\beta_C \tag{2.57}$$

$$A_S = 1 + \alpha - \alpha\beta_S \tag{2.58}$$

α, β_C, and β_S are recycle constants and are labelled the recirculation ratio, the biomass concentration ratio and the substrate concentration ratio, respectively. Then:

- For a biomass material balance around the bioreactor:

Increase = Inflows - Outflow + Net growth

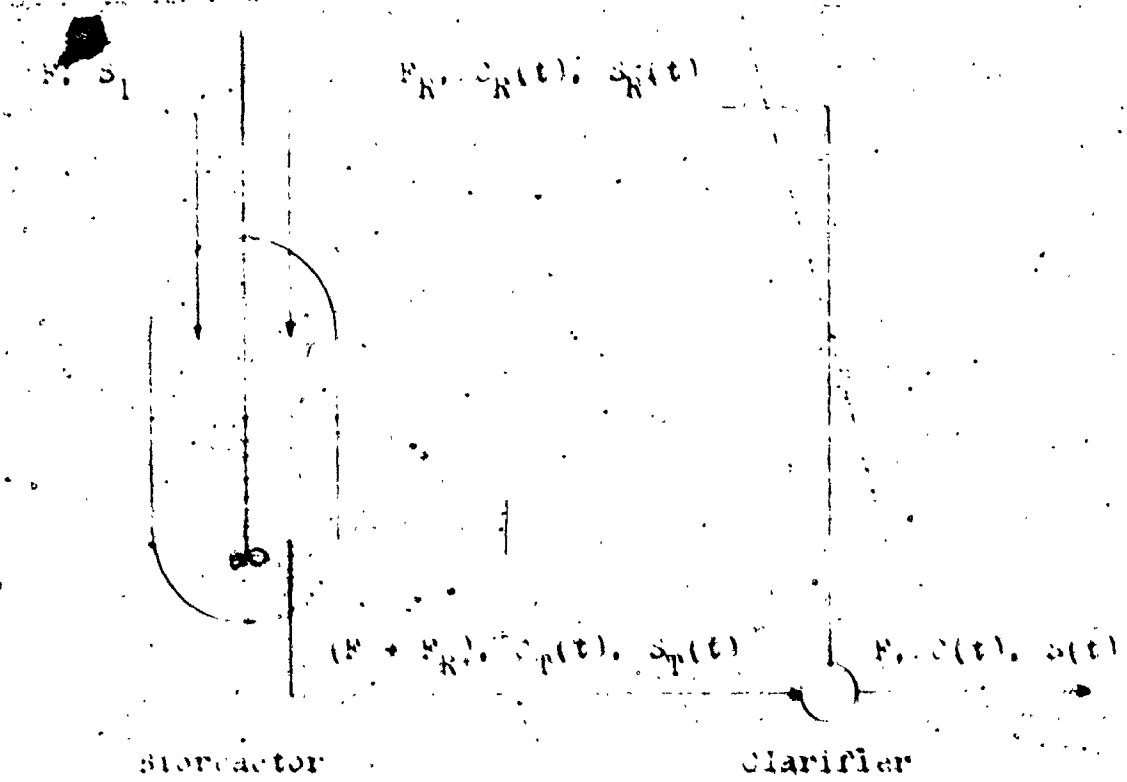


Figure 2.11 Schematic of a continuous fermentation process with recycle of both biomass and substrate.

$$V\frac{dc_p(t)}{dt} = [(F - F_R)c_f(t) + F_R c_R(t)] - (F + F_R)c_p(t) + V\mu(t)c_p(t) - k_d c_p(t)$$

or after simplification,

$$\frac{dc_p(t)}{dt} = \mu(t)c_p(t) - k_d c_p(t) \qquad (2.59)$$

Initial condition: $c_p(t_0) = c_0$

- For a substrate material balance around the bioreactor:

Increase = Inflows - Outflow - Consumption

$$V\frac{ds_p(t)}{dt} = Fs_f + F_R s_R(t) - (F + F_R)s_p(t) - V\mu(t)c_p(t)/Y$$

or after simplification,

$$\frac{ds_p(t)}{dt} = \frac{1}{\tau}(s_f - s_p(t)) - \mu(t)c_p(t)/Y \qquad (2.60)$$

Initial condition: $s_p(t_0) = s_0$

- For a biomass material balance around the clarifier:

Inflow = Outflows

$$(F + F_R)c_p(t) = Fc_e(t) + F_R c_R(t)$$

or after simplification,

$$c_R(t) = \alpha c_p(t) \qquad (2.61)$$

- For a substrate material balance around the clarifier:

Inflow = Outflows

$$(F + F_R)S_T(t) = FS(t) + F_R S_R(t)$$

or after simplification,

$$S(t) = \Lambda_S S_T(t) \qquad (2.62)$$

The specific growth rate is calculated from the Haldane-Monod equation as:

$$\mu(t) = \frac{\mu_m}{1 + \frac{k_s}{S_T(t)} + \frac{S_T(t)}{k_i}} \qquad (2.63)$$

Thus, Eqs. (2.59) to (2.63) represent a continuous fermentation process with recycle of both biomass and substrate. Note that the equations do not depend on α, β_C and β_S separately, but rather depend on $\Lambda_C = 1 + \alpha - \alpha\beta_C$ and $\Lambda_S = 1 + \alpha - \alpha\beta_S$, and there are many different combinations of α, β_C and β_S yielding the same values of Λ_C and Λ_S. When $\Lambda_C = \Lambda_S = 1$, the equations are the same as those representing the process without recycle, Eqs. (2.1) to (2.3). That is, recycle does not have any effect when $\Lambda_C = \Lambda_S = 1$. The CSMP diagram for the set of equations is shown as Figure 2.12. For the same coefficient values as in the previous section and with $\Lambda_C = \Lambda_S = .5$ (say $\alpha = 1$, $\beta_C = 1.5$ and $\beta_S = 1.5$), then the responses of $C(t)$, $S(t)$ and $\mu(t)$ are shown in Figures 2.13, 2.14 and 2.15, respectively. Comparing these figures with Figures 2.2, 2.3 and 2.4, one notes that recycle at $\Lambda_C = \Lambda_S = .5$ caused longer time to reach steady state and, however, increased the biomass concentration.

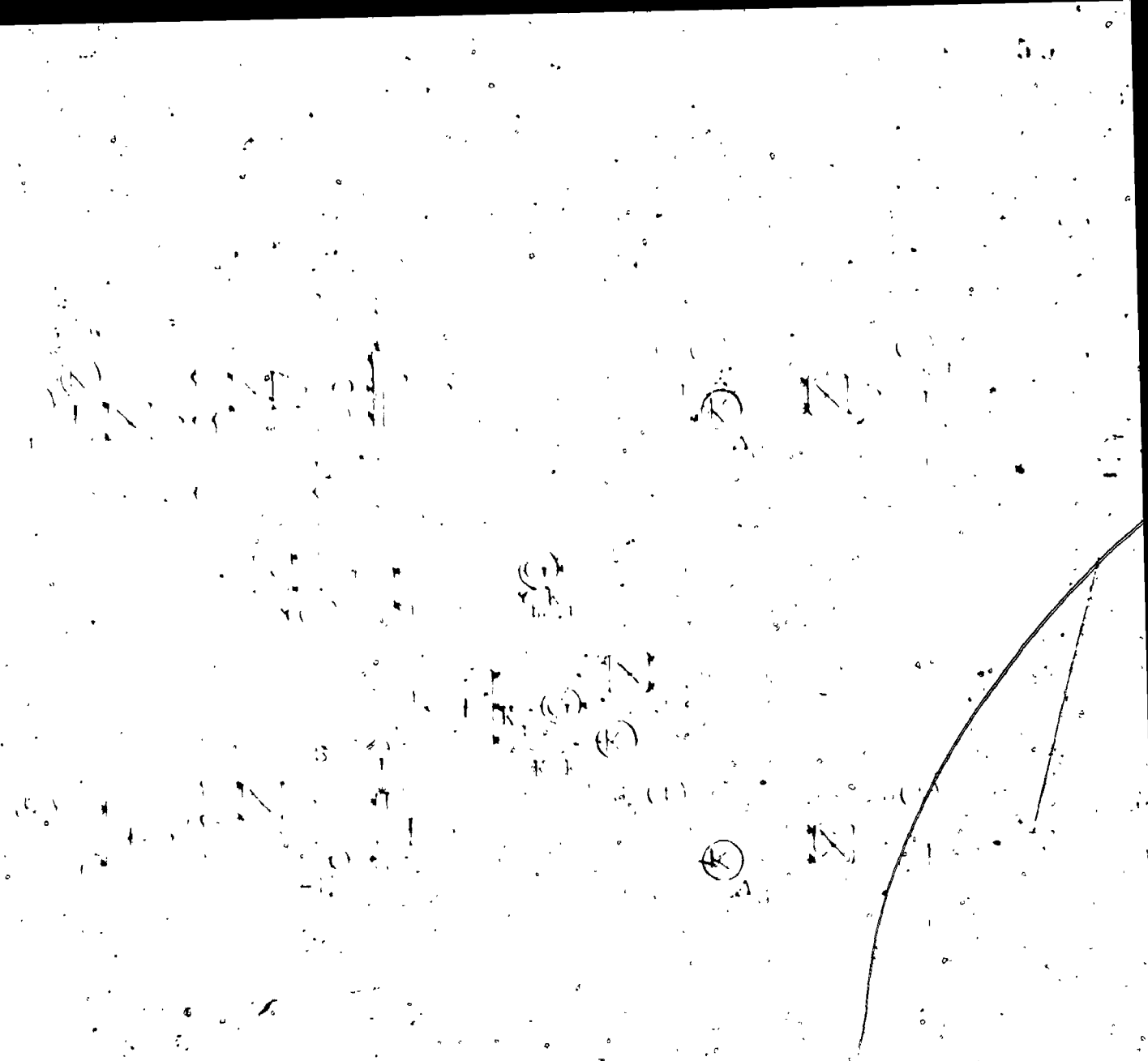


Figure 2.12 CSMP diagram representing the continuous fermentation process with recycle of both biomass and substrate as described by Equations (2.58) to (2.62).

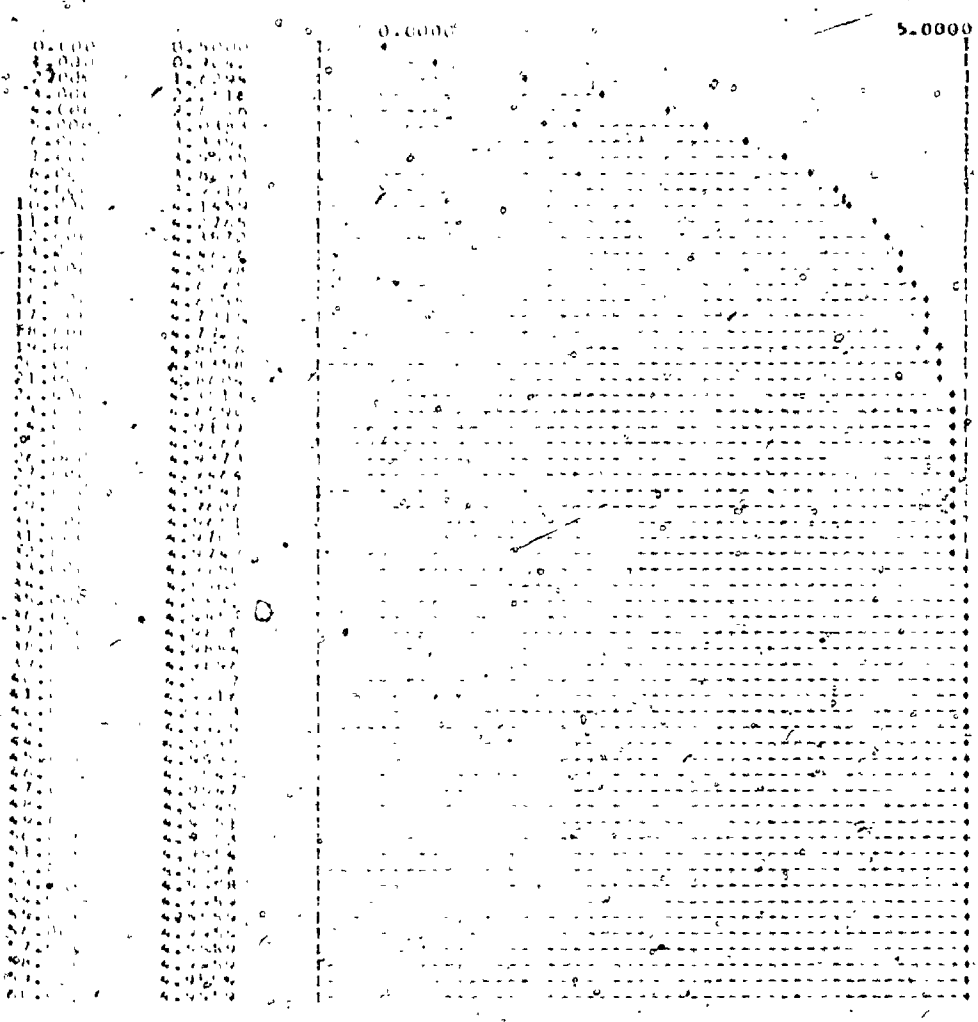


Figure 3.13 Biomass concentration C(t) versus time t for S_I = 5 g/l, γ_m = 1 hr^{-1}, K = 0 hr^{-1}, K_s = .03 g/l, K_i = 2 g/l, Y = .5, C_0 = .5 g/l and S_0 = 0 g/l. - With recycle of both biomass and substrate: Λ_C = .5 and Λ_S = .5.

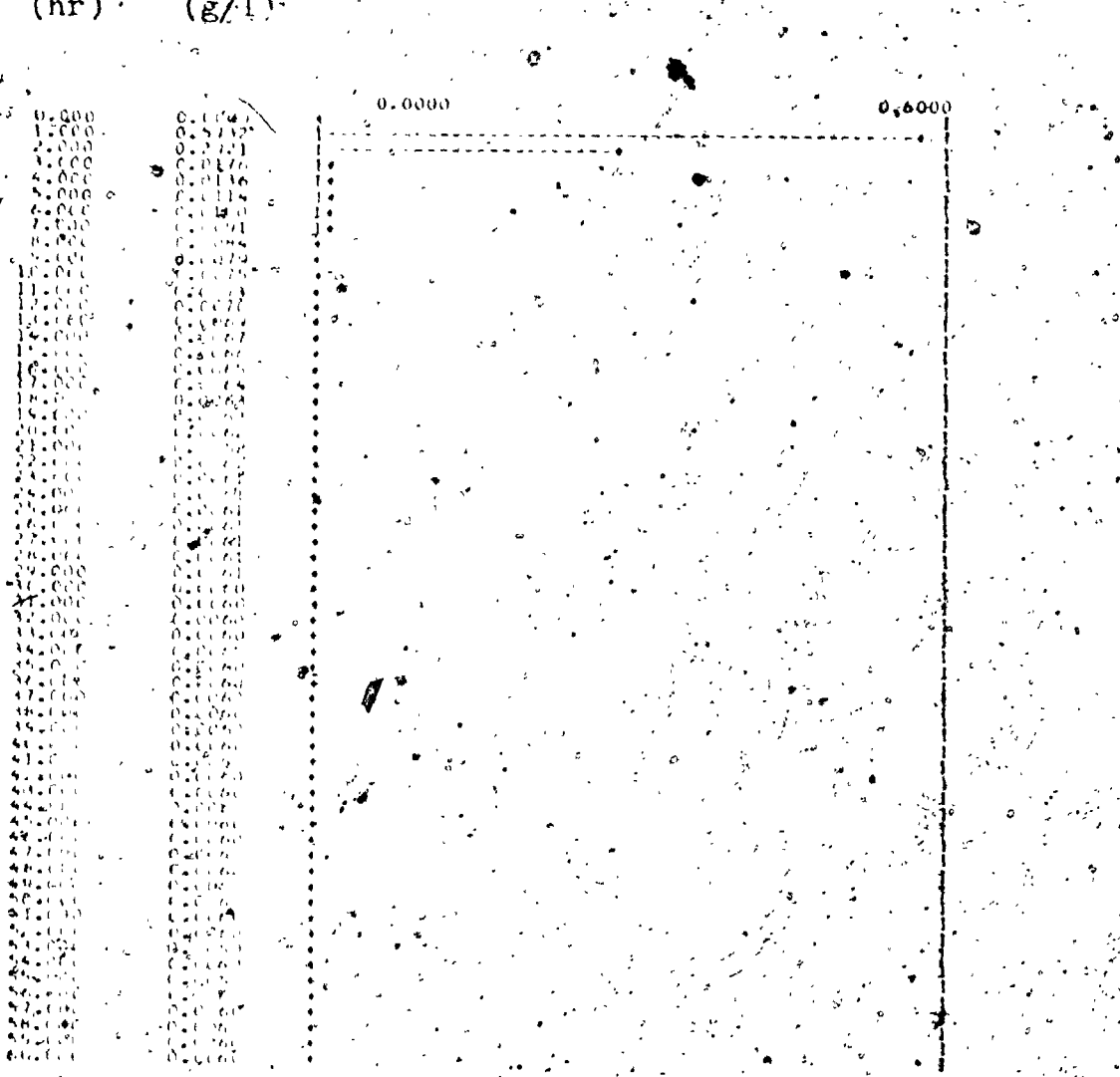


Figure 2.14 Substrate concentration $S(t)$ versus time t for $S_1 = 5$ g/l, $\mu_m = 1$ hr^{-1}, $k = 0$ hr^{-1}, $K_s = .03$ g/l, $K_i = 2$ g/l, $Y = .5$, $C_o = .5$ g/l and $S_o = 0$ g/l. - With recycle of both biomass and substrate: $\Lambda_C = .5$ and $\Lambda_S = .5$.

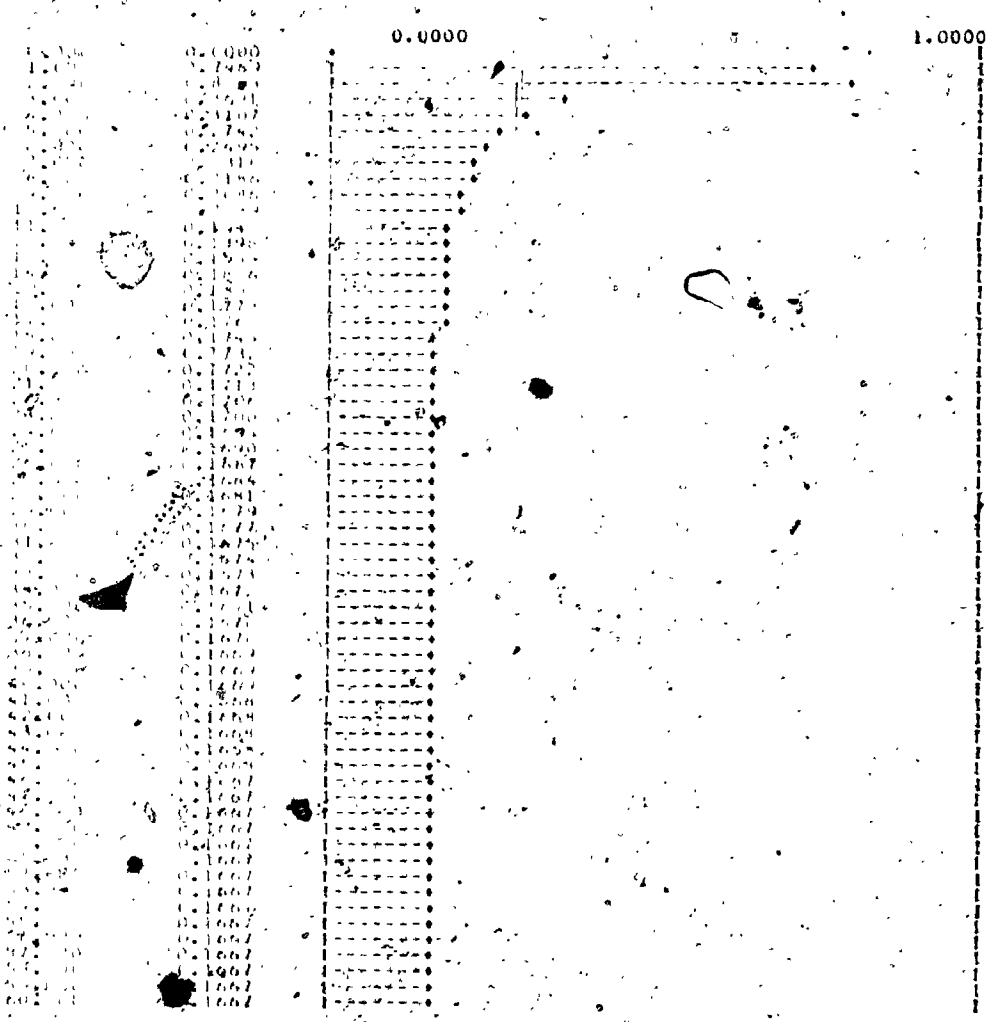


Figure 2.15 Specific growth rate $\mu(t)$ versus time t for S_I = 5 g/l, μ_m = 1 hr^{-1}, K = 0 hr^{-1}, K_s = .03 g/l, K_i = 2 g/l, Y = .5, C_o = .5 g/l and S_o = 0 g/l. - With recycle of both biomass and substrate Δ_C = .5 and Δ_S = .5.

II.2 Steady State Values

From Eqs. (2.59) to (2.63), the steady state equations are:

$$(\mu_{ss} - K - \Lambda_C D)C_{Tss} = 0 \qquad (2.64)$$

$$D(S_I - \Lambda_S S_{Tss}) - \mu_{ss} C_{Tss}/Y = 0 \qquad (2.65)$$

$$C_{ss} = \Lambda_C C_{Tss} \qquad (2.66)$$

$$S_{ss} = \Lambda_S S_{Tss} \qquad (2.67)$$

$$Y_{ss} = \frac{\mu_m}{1 + \frac{K_s}{S_{Tss}} + \frac{S_{Tss}}{K_i}} \qquad (2.68)$$

Following the same approach as in the previous section, for a given chemostat, there are three different steady state points:

$S_{1ss} = S_I$ and $C_{1ss} = 0$,

$$S_{2ss} = \frac{\Lambda_S K_i}{2}\left[\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right) + \sqrt{\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right]$$

and $C_{2ss} = \frac{Y\Lambda_C \nu}{\Lambda_C D + K}(S_I - S_{2ss})$,

$$S_{3ss} = \frac{\Lambda_S K_i}{2}\left[\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right]$$

and $C_{3ss} = \frac{Y\Lambda_C D}{\Lambda_C D + K}(S_I - S_{3ss})$

Only the third steady state point (S_{3ss}, C_{3ss}) is the desirable one, and the following constraints must be satisfied:

$$\Lambda_C D \leq \frac{\mu_m}{1 + 2\sqrt{K_s/K_i}} - K \qquad (2.69)$$

$$D \geq 0 \qquad (2.70)$$

$$S_I - \frac{\Lambda_S K_i}{2}\left[\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \geq 0 \qquad (2.71)$$

$$\Lambda_S \geq 0 \qquad (2.72)$$

$$\Lambda_C \geq 0 \qquad (2.73)$$

If in particular $K_i \to \infty$, that is the specific growth rate follows the Monod equation, then besides the wash-out point there is only one steady state point:

$$S_{ss} = \frac{\Lambda_S K_s (\Lambda_C D + K)}{\mu_m - \Lambda_C D - K} \qquad (2.74)$$

$$C_{ss} = \frac{Y\Lambda_C D}{\Lambda_C D + K}(S_I - S_{ss}) \qquad (2.75)$$

The following constraints are imposed:

$$\Lambda_C D \leq \frac{\mu_m S_I}{S_I + \Lambda_S K_s} - K \qquad (2.76)$$

$$D \geq 0 \qquad (2.70)$$

$$\Lambda_S \geq 0 \qquad (2.72)$$

$$\Lambda_C \geq 0 \qquad (2.73)$$

II.3 Optimum Operating Conditions

For a fermentation process with recycle of biomass and substrate, the nonlinear programming encountered is:

$$\max_{S_I, D, \Lambda_S, \Lambda_C, T, H} J_6 = D C_{ss} \qquad (2.77)$$

subject to:

$$C_{ss} = \frac{Y \Lambda_C D}{\Lambda_C D + K}(S_I - S_{ss}) \qquad (2.78)$$

$$S_{ss} = \frac{\Lambda_S K_i}{2}\left[\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \qquad (2.79)$$

$$\Lambda_C D = \frac{\mu_m}{1 + 2K_s/K_i} - K \qquad (2.69)$$

$$D \geq 0 \qquad (2.70)$$

$$S_I - \frac{\Lambda_S K_i}{2}\left[\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right) - \sqrt{\left(\frac{\mu_m}{\Lambda_C D + K} - 1\right)^2 - 4\frac{K_s}{K_i}}\right] \geq 0 \qquad (2.71)$$

$$\Lambda_S \geq 0 \qquad (2.72)$$

$$\Lambda_C \geq 0 \qquad (2.73)$$

$$\mu_m = f_1(T, H) \qquad (2.27)$$

$$K_s = f_2(T, H) \qquad (2.28)$$

$$K_1 = f_3(T,H) \tag{2.29}$$

$$K = f_4(T,H) \tag{2.30}$$

$$Y = f_5(T,H) \tag{2.31}$$

From Eqs. (2.77), (2.78), (2.79) and (2.71) it is clear that the optimum value for Λ_S should be zero and the influent concentration S_I should be from a theoretical point of view as large as possible. However, as mentioned before, for the bioreaction to operate well and to follow Eqs. (2.59) to (2.63), the medium cannot be too concentrated, that is there is an upper limiting value that S_I cannot exceed. Let it be S_I^+, then

$$\Lambda_S^* = 0 \tag{2.80}$$

$$S_I^* = S_I^+ \tag{2.81}$$

Hence the problem is reduced to:

$$\max_{D,\Lambda_C,T,H} J_7 = \frac{Y\Lambda_C D^2}{\Lambda_C D + K} S_I^* \tag{2.82}$$

subject to Eqs. (2.69), (2.70), (2.73), (2.27) to (2.31).

For certain given T, H and Λ_C, the plot of J_7 versus D, including the constraints in Eqs. (2.69) and (2.70), is shown in Figure 2.16. From Figure 2.16, it is clear that the dilution rate should be chosen as $\left(\frac{\mu_m}{1+2\sqrt{K_s/K_i}} - K\right)/\Lambda_C$ and the

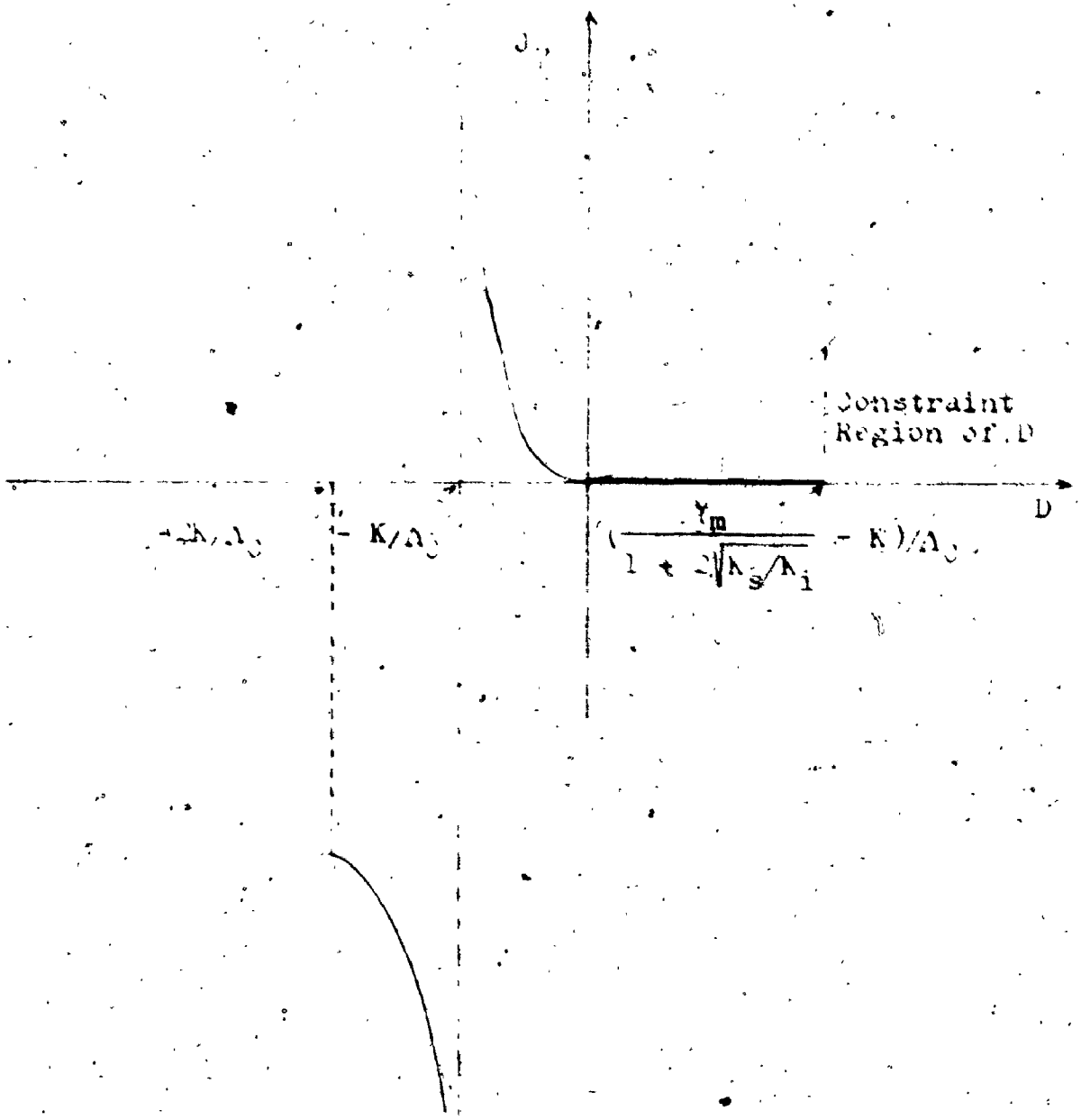


Figure 3.10 Plot of the performance index J_3 defined by Equation (3.82) versus dilution rate D when temperature T, pH H and biomass recycle factor Λ_3 are held fixed.

nonlinear programming problem is further reduced to:

$$\text{Max. } J_s \quad \frac{Y\left(\frac{\gamma_m}{1+2\sqrt{k_s/k_i}} - \lambda\right)^2 \lambda_1}{\left(\frac{\gamma_m}{1+2\sqrt{k_s/k_i}} - \lambda\right) + \lambda} S_1^*$$

$$\frac{Y\gamma_m - \lambda(1+2\sqrt{k_s/k_i})}{\lambda_1 \gamma_m (1+2\sqrt{k_s/k_i})} \qquad (2.83)$$

subject to Eqs. (2.73), (2.77) to (2.81).

From Eq. (2.83), it is obvious that for certain given T and H, λ_1 should be chosen as a small positive number as possible. Note that λ_1 cannot be zero because as seen from Eqs. (2.60) and (2.77) if $\lambda_1 = 0$ then $S_{ss} = 0$ and hence $J_s = 0$. Let the smallest acceptable value of λ_1 be λ_1^* then,

$$\lambda_1^* = \lambda_1^* \qquad (2.84)$$

Thus, the problem reduces to the following two-dimensional unconstrained optimization problem:

$$\text{Max. } J_s \quad \frac{f_5(T,H)\left[f_1(T,H) - f_4(T,H)\left[1+2\sqrt{f_3(T,H)/f_4(T,H)}\right]\right]^2 S_1^*}{\lambda_1^* f_1(T,H)\left[1 + 2\sqrt{f_2(T,H)/f_4(T,H)}\right]}$$

(2.85)

The solutions of this problem, T^* and H^* can be found by the exhaustive search technique that was used earlier. The optimum

dilution rate D^* is then:

$$D^* = \left[\frac{f_1(F^*,H^*)}{1 + \lambda_1 f_2(F^*,H^*)/f_3(F^*,H^*)} - f_4(F^*,H^*) \right] / \lambda_5 \qquad (2.86)$$

In the case where the specific growth rate follows the Monod equation, $\lambda_1 > 0$. Then, the optimum values for λ_5, S_1 and λ_4 are still the same as shown in Eqs. (2.80), (2.81) and (2.82). However, the optimum value for F and H are now found by solving a simpler problem, described by Eq. (2.87) instead of Eq. (2.80).

$$\underset{F,H}{\text{Max.}} \; J_{10} = \frac{f_2(F,H)[f_1(F,H) - f_4(F,H)]S_1^*}{\lambda_5^* f_1(F,H)} \qquad (2.87)$$

and the optimum dilution rate is computed from Eq. (2.88) instead of Eq. (2.80).

$$D^* = [f_1(F^*,H^*) - f_4(F^*,H^*)]/\lambda_5^* \qquad (2.88)$$

From Eqs. (2.87) and (2.88), it should be noted that, when the specific growth rate follows the Monod equation, the saturation coefficient λ_3 or $f_3(F,H)$ has no effect on the optimum performance index.

2.4 Example

As an example, the continuous fermentation process of A. aerogenes mentioned in Section 1.3 is considered. Suppose

the largest acceptable influent substrate concentration and the smallest acceptable biomass recycle factor are 10. g l^{-1} and 0.1, respectively. Then,

$$S_i^* = 10. \qquad (2.89)$$

$$X^* = 0. \qquad (2.90)$$

$$\lambda^* = 0.1 \qquad (2.91)$$

$$P_{10}^* = \frac{[\text{[illegible]} \times 10^{10} e^{-14230/RT} - \text{[illegible]} \times 10^{5} e^{-9000/RT}]\, 10.}{0.1 \times \text{[illegible]} \times 10^{10} e^{-14230/RT}}$$

(2.92)

The plot of P_{10} versus T is shown as Figure 2.1[illegible]. From the plot, the temperature T should be chosen as large as possible. However, Topiwala and Sinclair formulated the temperature relationship by using only data collected in the temperature range (20°C - 40°C). Therefore, the relationship may not be valid for temperatures lying outside this range. If the operating temperature is constrained to this range, then,

$$T^* = 40°C \qquad (2.93)$$

$$P_{10}^* = 128.02 \text{ g l}^{-1}\text{ hr} \qquad (2.94)$$

and the optimum dilution rate is

$$D^* = [\text{[illegible]} \times 10^{10} e^{-14230/R(40 + 273)} - \text{[illegible]} \times 10^{5} e^{-9000/R(40 + 273)}]$$

$$= \text{[illegible]} \text{ hr}^{-1} \qquad (2.95)$$

Comparing the values found here with the last line

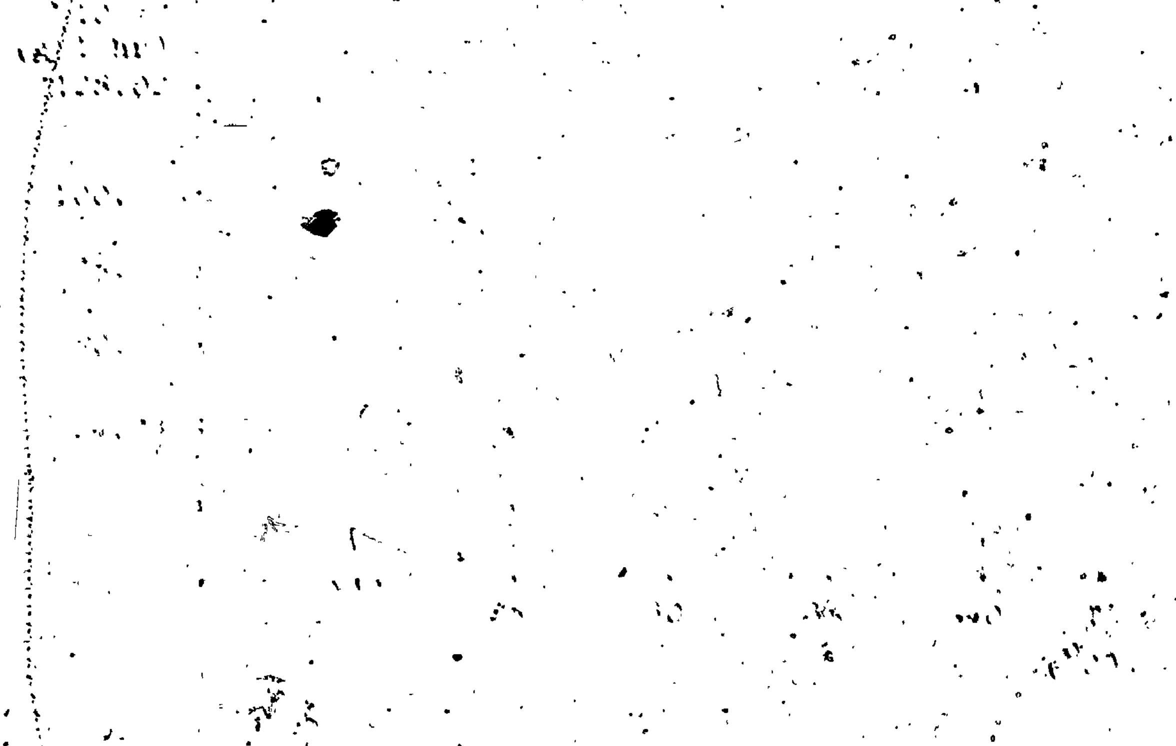


Figure [illegible]. Plot of the performance index [illegible] defined by Equation [illegible] versus temperature T for a continuous fermentation of *A. awamori*, using the [illegible] model.

of Table 3.1, one will realize that:

- with recycle the rate of production of biomass has been increased almost 32 times (128.02 versus 4.02),

- with recycle one can operate the dilution rate at the value far greater than the maximum specific growth rate (D^* 20.00 hr^{-1} versus μ_m 2.33 hr^{-1}); whereas without recycle the dilution rate must always be smaller than the maximum specific growth rate as seen from Eq. (3.1.2) or (3.1.3).

To conclude this chapter, it should be noted that all of the above derivations were done with the assumptions that the bioreactor is perfectly mixed, the clarifier is ideal and the model as well as the model coefficients are known exactly. However, in practice, the bioreactor may not be perfectly mixed, the clarifier cannot be ideal, there is time delay in the recycle loop, there is loss of materials due to clarification, and it is extremely difficult to maintain X_r and S_r at constant levels particularly during the transient period (Ramanthan and Gaddy 1971). Furthermore, it is, in general, impossible to find a model including its coefficients that can describe a process perfectly. Therefore, the experimental optimum operating conditions are not necessary the same as, but should be close to the derived optimum operating conditions as long as the assumptions are not severely violated.

CHAPTER III

SYSTEM IDENTIFICATION BY LEAST SQUARES

In the last several years, much interest has been devoted to the study of modern system theory. An essential ingredient of many modern system problems is the requirement to accomplish system identification or modeling. As defined here, system identification or modeling is the process of determining a difference or differential equation (or the coefficient parameters of such an equation) such that it describes a physical process in accordance with some predetermined criterion (Sage and Melsa 1971b).

There are many criteria which one frequently uses such as least squares, maximum likelihood, maximum a posteriori, and minimum variance. However, the most well-known and simplest criterion is least squares (Sage and Melsa 1971a, 1971b). Least squares was first used by Gauss (1821) and is the subject of this chapter. For other criteria one can consult text books and survey papers such as those written by Sage and Melsa (1971a, 1971b), Meirdel (1973), Eykhoff (1968) and Astrom and Eykhoff (1971).

Firstly, the statement of a general system identification problem by least squares will be mathematically

formulated. Then, the most effective optimization techniques used in solving the identification problem will be reviewed. And finally, as illustrative examples, a batch fermentation of Aureobasidium pullulans and a continuous fermentation of Morchella crassipes will be identified.

I STATEMENT OF THE PROBLEM

In general a dynamic system can be represented by the following state space model:

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}(t), \underline{a}, t) \qquad (3.1)$$

$$\underline{y}(t) = \underline{g}(\underline{x}(t), \underline{u}(t), \underline{a}, t) \qquad (3.2)$$

where,

$\underline{u}(t)$ = Input vector at time t,

$\underline{x}(t)$ = State vector at time t,

$\underline{y}(t)$ = Output vector at time t,

$\underline{a}$ = Unknown constant parameter vector to be identified,

$\underline{f}$ = known vector function of $\underline{x}(t)$, $\underline{u}(t)$, $\underline{a}$ and t,

$\underline{g}$ = known vector function of $\underline{x}(t)$, $\underline{u}(t)$, $\underline{a}$ and t.

Suppose the input vector and the output vector can be measured during the time interval (t_o, t_f). Let the measured quantities be denoted by $\underline{u}_m(t)$ and $\underline{y}_m(t)$ respectively. Then, from the criterion of least squares, the identification

problem is to find

$$\underline{\theta} = \frac{\underline{a}}{\underline{x}(t_o)} \tag{3.3}$$

so that

$$\rho = \frac{1}{2}\int_{t_o}^{t_f} \left\| \underline{y}_m(t) - \underline{y}(t) \right\|_{\underline{V}(t)} dt \tag{3.4}$$

is a minimum, subject to the following equations:

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t) \tag{3.5}$$

$$\underline{y}(t) = \underline{h}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t) \tag{3.6}$$

where $\underline{V}(t)$ is a known symmetric, positive definite weighting matrix function.

If data can only be measured at discrete instants t_i for $i = 0, 1, \ldots, l$ in the interval (t_o, t_f), which is often the case, then Eq. (3.4) is replaced by Eq. (3.7),

$$\rho = \frac{1}{2}\sum_{i=0}^{l} \left\| \underline{y}_m(t_i) - \underline{y}(t_i) \right\|^2_{\underline{V}_i} \tag{3.7}$$

where $\underline{V}_i$ is a known, symmetric, positive definite weighting matrix for all i. Eqs. (3.5) and (3.6) remain the same.

For a given $\underline{\theta}$, $\underline{y}(t_i)$ for all i can be found from Eqs. (3.5) and (3.6) by some numerical integration technique

(Haggerty 1972, Stark 1970) which requires values of $\underline{u}_m$ at various times in the interval (t_o, t_i). For the values of $\underline{u}_m$ which do not correspond to time $t_o, t_1, \ldots, t_i$ one must find them by using some proper interpolation technique (Haggerty 1972, Stark 1970). Linear interpolation is often found sufficiently accurate and is based on the following equation:

$$\underline{u}_m(t) = \underline{u}_m(t_i) + \frac{t - t_i}{t_{i+1} - t_i} \left[\underline{u}_m(t_{i+1}) - \underline{u}_m(t_i)\right] \quad (3.8)$$

$$\text{for } t_i \leq t \leq t_{i+1}$$

Note that if the differences between $\underline{y}_m(t_i)$ and $\underline{y}(t_i)$ for all i are assumed to be due to white Gaussian noise with $\underline{0}$-mean and $(\sigma^2 \underline{v}_i^{-1})$-variance and the rest of variables are assumed to be deterministic, then the least squares criterion is equivalent to the maximum likelihood criterion (Fisher 1922, 1925; Sage and Melsa 1971a) and of each element of $\underline{\theta}$, an approximate confidence interval (an interval estimate) can be computed and an approximate F-test for significance can be carried out (Appendix B).

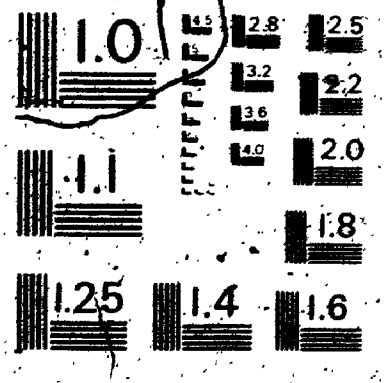
By classical calculus, to minimize S, one must solve the following equation:

$$\frac{dS}{d\underline{\theta}} = \underline{0} \quad (3.9)$$

However, from Eqs. (3.5) and (3.6), $\underline{y}(t)$ cannot in general

2 4

OF/DE



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963

be solved analytically in terms of $\underline{\theta}$. Therefore $dS/d\underline{\theta}$ cannot in general be represented analytically in terms of $\underline{\theta}$. Moreover, if it is possible then the solution of Eq. (3.9) rarely can be computed using classical approaches.

Thus, a modern optimization technique must be employed to minimize S defined in Eq. (3.4) or (3.7), subject to Eqs. (3.5) and (3.6).

Bard (1970) has reported that the Modified Gauss-Newton techniques (Booth and Peterson 1958, Hartley 1961, Astrom et al 1965) including Marquardt's modification (Marquardt 1963) are the best among the existing iterative gradient search techniques for solving nonlinear least squares problems. The following is a review of these techniques as well as Bellman's Quasilinearization technique, a very popular technique which is different in approach but can be shown to be equivalent to the Gauss-Newton technique in Appendix A.

II THE GAUSS-NEWTON TECHNIQUE

The Gauss-Newton technique is an iterative search technique, in which a sequence $\{\underline{\theta}_k\}$ is generated. The first element of the sequence $\underline{\theta}_o$ must be supplied. If the sequence converges then the converged value is a local minimum point and the value of S at the converged value is a local minimum.

To ensure that a global minimum is being found, various starting points for $\underline{\theta}_o$ must be tried. If the starting point $\underline{\theta}_o$ is close to a minimum point, the Gauss-Newton technique should give very fast convergence (quadratic convergence), otherwise it will oscillate or even diverge.

From the Gauss-Newton technique, at the beginning of the $(k+1)^{\underline{th}}$ iteration, $\underline{\theta}_k$ is known and $\underline{\theta}_{k+1}$ is computed as follows. S defined in Eq. (3.7) is expanded around $\underline{\theta}_k$ to give:

$$S = \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{y}_m(t_i) - \underline{y}(t_i)\right\|^2_{\underline{V}_i}$$

$$- \delta\underline{\theta}_k^T \sum_{i=0}^{I}\left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i\left[\underline{y}_m(t_i) - \underline{y}(t_i)\right]$$

$$+ \tfrac{1}{2}\delta\underline{\theta}_k^T\left\{\sum_{i=0}^{I}\left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i\left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]\right\}\delta\underline{\theta}_k \qquad (3.10)$$

or,

$$S = \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{y}_m(t_i) - \underline{y}(t_i) - \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]\delta\underline{\theta}_k\right\|^2_{\underline{V}_i} \qquad (3.11)$$

where $\underline{y}(t_i)$ and $d\underline{y}(t_i)/d\underline{\theta}^T$ $i = 0,1,\ldots,I$ from both Eqs. (3.10) and (3.11) are computed at $\underline{\theta} = \underline{\theta}_k$, and

$$\underline{\theta} = \underline{\theta}_k + \delta\underline{\theta}_k \qquad (3.12)$$

Note that the third term in the right hand side of Eq. (3.10) is only a part of the second order terms in the Taylor series expansion of S. The Steepest Descent technique is approximated by only the first two terms (linear terms), whereas the Newton-Raphson technique would include all second order terms. The Gauss-Newton technique requires only the terms containing the first derivatives of $\underline{y}(t_i)$ with respect to $\underline{\theta}$ at $\underline{\theta} = \underline{\theta}_k$. This is equivalent to the linearization of $\underline{y}(t_i)$ around $\underline{\theta} = \underline{\theta}_k$ as seen from Eq. (3.11).

Set the derivative of S defined by Eq. (3.10) with respect to $\underline{\theta}$ equal to zero, then from Eq. (3.12),

$$\frac{dS}{d\delta\underline{\theta}_k} = \underline{0} \tag{3.13}$$

or from Eq. (3.10),

$$\left\{ -\sum_{i=0}^{I} \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i \left[\underline{y}_m(t_i) - \underline{y}(t_i)\right] + \sum_{i=0}^{I} \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right] \delta\underline{\theta}_k \right\} \Bigg|_{\underline{\theta} = \underline{\theta}_k} = \underline{0} \tag{3.14}$$

Let

$$\underline{H}_k = \sum_{i=0}^{I} \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right] \Bigg|_{\underline{\theta} = \underline{\theta}_k} \tag{3.15}$$

$$\underline{g}_k = -\sum_{i=0}^{I}\left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]^T \underline{V}_i\left[\underline{y}_m(t_i) - \underline{y}(t_i)\right]\Bigg|_{\underline{\theta} = \underline{\theta}_k} \qquad (3.16)$$

then, if $\underline{H}_k$ is nonsingular, Eq. (3.14) gives:

$$\delta\underline{\theta}_k = -\underline{H}_k^{-1}\underline{g}_k \qquad (3.17)$$

and the value of $\underline{\theta}_{k+1}$ is computed as:

$$\underline{\theta}_{k+1} = \underline{\theta}_k + \delta\underline{\theta}_k \qquad (3.18)$$

Note that $\underline{H}_k$ and $\underline{g}_k$ are the Hessian matrix of S with terms containing second derivatives being ignored and the gradient vector of S both with respect to $\underline{\theta}$ and at $\underline{\theta} = \underline{\theta}_k$. If continuous data are available, that is Eq. (3.4) is used in place of Eq. (3.7), then instead of Eqs. (3.15) and (3.16), $\underline{H}_k$ and $\underline{g}_k$ will be defined as:

$$\underline{H}_k = \int_{t_o}^{t_f}\left[\frac{d\underline{y}(t)}{d\underline{\theta}^T}\right]^T \underline{V}(t)\left[\frac{d\underline{y}(t)}{d\underline{\theta}^T}\right] dt\Bigg|_{\underline{\theta} = \underline{\theta}_k} \qquad (3.19)$$

$$\underline{g}_k = -\int_{t_o}^{t_f}\left[\frac{d\underline{y}(t)}{d\underline{\theta}^T}\right]^T \underline{V}(t)\left[\underline{y}_m(t) - \underline{y}(t)\right] dt\Bigg|_{\underline{\theta} = \underline{\theta}_k} \qquad (3.20)$$

II.1 Algorithm

The following is an algorithm using the Gauss-Newton technique:

1. An initial approximation $\underline{\theta}_o$ is supplied either by guessing or by some approximate technique. Let k = 0.

2. Compute $\underline{y}(t_i)$ for all i at $\underline{\theta} = \underline{\theta}_k$ from Eqs. (3.5) and (3.6) using a numerical integration technique such as Euler or Runge-Kutta (Haggerty 1972). If the integration technique used requires values of $\underline{u}_m$ at times between measurements, then an interpolation technique can be employed.

3. Compute $d\underline{y}(t_i)/d\underline{\theta}^T$ for all i at $\underline{\theta} = \underline{\theta}_k$. This can be done by using a finite difference approximation as described by Eq. (3.21),

$$\left.\frac{d\underline{y}(t_i)}{d(\underline{\theta})_j}\right|_{\underline{\theta}=\underline{\theta}_k} = \frac{1}{\alpha(\underline{\theta}_k)_j}\left[\underline{y}(t_i)\Big|_{\underline{\theta}=\underline{\theta}_k+\Delta_j\underline{\theta}_k} - \underline{y}(t_i)\Big|_{\underline{\theta}=\underline{\theta}_k}\right] \qquad (3.21)$$

where,

$(\underline{\theta})_j$ = The $j^{\underline{th}}$ element of vector $\underline{\theta}$,

$(\underline{\theta}_k)_j$ = The $j^{\underline{th}}$ element of vector $\underline{\theta}_k$,

α = A small positive constant, say 10^{-5},

$\Delta_j\underline{\theta}_k$ = A vector with its $j^{\underline{th}}$ element equal to $\alpha(\underline{\theta}_k)_j$ while the rest of its elements are equal to zero.

4. Compute $\underline{H}_k$ and $\underline{g}_k$ from Eqs. (3.15) and (3.16).

5. Compute the Gauss-Newton step $\delta\underline{\theta}_k$ from Eq. (3.17).

6. Compute $\underline{\theta}_{k+1}$ from Eq. (3.18).

7. If a test of convergence mentioned in the next section, Section III, is met then go to step 8. Otherwise, let $k \longleftarrow (k + 1)$ then go to step 2.

8. Stop. The converged value $\underline{\theta}_{k+1}$ is then a local minimum point (not necessarily a global minimum point).

For a given starting point $\underline{\theta}_0$, the Gauss-Newton technique does not necessarily produce a converging sequence $\{\underline{\theta}_k\}$. Thus for a practical computing purpose, another stopping test is needed. Say, in step 7, add this sentence:

- Stop after 15 iterations.

III CONVERGENCE TESTS

If convergence occurs, the sequence $\{\underline{\theta}_k\}$ will get closer and closer to the converged value. In general, it will take an infinite amount of time to reach the converged value. Therefore, from a practical computation point of view, a convergence test is needed. The two most frequently used tests are: test for convergence of the sequence of the parameter vector $\{\underline{\theta}_k\}$ and test for convergence of the sequence of the sum of squares $\{S(\underline{\theta}_k)\}$.

III.1 Test For Convergence Of $\{\underline{\theta}_k\}$

At the end of the $(k+1)^{\underline{th}}$ iteration, $\underline{\theta}_{k+1}$ is accepted as a local minimum point only when:

$$\left|(\underline{\theta}_{k+1})_i - (\underline{\theta}_k)_i\right| \leq \alpha_1 \left|(\underline{\theta}_k)_i + \alpha_2\right| \quad \text{for all } i \tag{3.22}$$

where,

α_1 and α_2 = Prespecified positive small constants, say 10^{-5} and 10^{-10}.

$(\underline{\theta}_k)_i$ and $(\underline{\theta}_{k+1})_i$ = The $i^{\underline{th}}$ elements of vectors $\underline{\theta}_k$ and $\underline{\theta}_{k+1}$ respectively.

It should be noted that when S is very sensitive to $\underline{\theta}$, it is necessary to set α_1 and α_2 sufficiently small to ensure a local minimum, the converged value of the sequence $\{S(\underline{\theta}_k)\}$, is really reached.

III.2 Test For Convergence Of $\{S(\underline{\theta}_k)\}$

For the test of convergence of the sequence $\{S(\underline{\theta}_k)\}$, iterations cease and $\underline{\theta}_{k+1}$ is accepted as a local minimum point only when:

$$\left|S(\underline{\theta}_{k+1}) - S(\underline{\theta}_k)\right| \leq \alpha_1 \left[S(\underline{\theta}_k) + \alpha_2\right] \tag{3.23}$$

where α_1 and α_2 are defined as before (Section III.1).

It should be noted that when S is not sensitive to $\underline{\theta}$, it is necessary to set α_1 and α_2 sufficiently small to ensure that a local minimum point, the converged value of the sequence $\{\underline{\theta}_k\}$, is also reached.

Referred to Eqs. (3.22) and (3.23), this test requires only one comparison as compared to many comparisons required by the previous test. Therefore this test takes less computer time. However, in some instances, it may produce a false minimum as shown in Figure 3.1. In the figure, $S(\underline{\theta}_k)$ and $S(\underline{\theta}_{k+1})$ are approximately the same, that is they satisfy the convergence test described by Eq. (3.22), however $S(\underline{\theta}_{k+1})$ is a false minimum, not a true minimum. Thus, care must be exercised when the test for convergence of the sequence $\{S(\underline{\theta}_k)\}$ is being used.

Another convergence test which one sometimes employs is shown as Eq. (3.24),

$$\left| \|\underline{\theta}_{k+1}\| - \|\underline{\theta}_k\| \right| \leq \alpha_1 (\|\underline{\theta}_k\| + \alpha_2) \qquad (3.24)$$

where α_1 and α_2 are defined as before, $\|\underline{\theta}_k\|$ and $\|\underline{\theta}_{k+1}\|$ are Euclidean norms of $\underline{\theta}_k$ and $\underline{\theta}_{k+1}$ respectively. This test may also produce a false minimum as two different vectors may have the same norms.

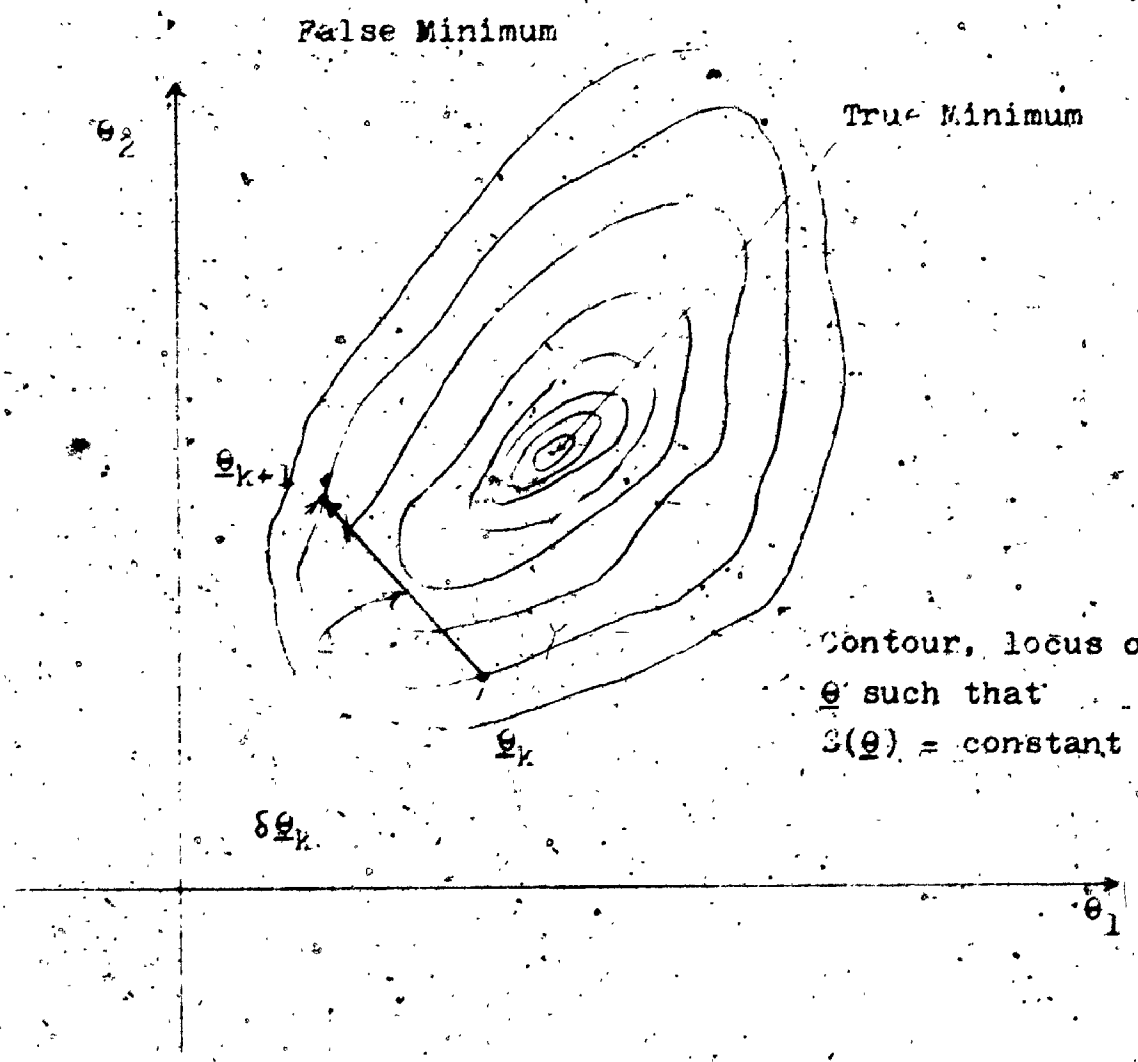


Figure 3.1 Graphical illustration of a false minimum produced by the test of convergence of $\{S(\underline{\theta}_k)\}$ in a two dimensional space ($\underline{\theta}^T = [\theta_1 \quad \theta_2]$).

IV. MODIFIED GAUSS-NEWTON TECHNIQUES

When the initial approximation $\underline{\theta}_0$ is far away from a minimum point, the Gauss-Newton sequence $\{\underline{\theta}_k\}$ may diverge. To modify the Gauss-Newton technique and to enlarge the region of convergence, there exist two well-known ideas in optimization literature: Hartley's reduction of step-sizes (Hartley 1961) and Marquardt's incorporation with the steepest descent technique (Marquardt 1963, Smith Jr. and Shanno 1971). Note that the region of convergence is the locus of all values of the initial approximation $\underline{\theta}_0$ such that the sequence $\{\underline{\theta}_k\}$ will converge.

IV.1 Hartley's Reduction of Step-Sizes

Hartley found that nonconvergence of the Gauss-Newton-sequence $\{\underline{\theta}_k\}$ is often due to step-sizes being too large, retaining the same step-directions and reducing step-sizes in a proper manner would enlarge the region of convergence. According to Hartley, at the $(k+1)^{\text{th}}$ iteration, instead of choosing $\underline{\theta}_{k+1}$ as in Eq. (3.18), let

$$\underline{\theta}_{k+1} = \underline{\theta}_k + \beta^* \delta\underline{\theta}_k \tag{3.25}$$

where β^* is a real constant lying between zero and unity and such that it minimizes $S(\underline{\theta}_k + \beta\delta\underline{\theta}_k)$. That is,

$$S(\underline{\theta}_k + \beta^*\delta\underline{\theta}_k) \leq S(\underline{\theta}_k + \beta\delta\underline{\theta}_k) \qquad (3.26)$$
$$0 \leq \beta \leq 1$$

An illustration of Hartley's modification versus the Gauss-Newton technique in a two dimensional space is shown on Figure 3.2.

Minimizing $S(\underline{\theta}_k + \beta\delta\underline{\theta}_k)$ over β for each iteration is time consuming, thus Hartley also suggested a method to approximate β^* using a quadratic interpolation: passing a parabola through three points corresponding to $\beta = 0$, $\beta = \frac{1}{2}$ and $\beta = 1$ and then selecting an optimum point. This results in Eq. (3.27),

$$\beta^* = \tfrac{1}{2} + \tfrac{1}{4}\left[S(\underline{\theta}_k) - S(\underline{\theta}_k + \delta\underline{\theta}_k)\right] / \left[S(\underline{\theta}_k) - 2S(\underline{\theta}_k + \tfrac{1}{2}\delta\underline{\theta}_k) + S(\underline{\theta}_k + \delta\underline{\theta}_k)\right] \qquad (3.27)$$

However, it is sometimes found that, as shown in Figure 3.3, β^* computed from Eq. (3.27) is a maximum point rather a minimum point (with respect to the parabola). In general, to simplify calculation, one may just let $\beta^* = 1, \frac{1}{2}, \frac{1}{4}, \ldots$ until $S(\underline{\theta}_k + \beta^*\delta\underline{\theta}_k) < S(\underline{\theta}_k)$.

Note that Hartley's modification is similar to that of Booth and Peterson (1958), that is, retaining the step-directions and adjusting step-sizes. Nevertheless, these workers allowed a step-size to be greater than that of $\delta\underline{\theta}$ if possible.

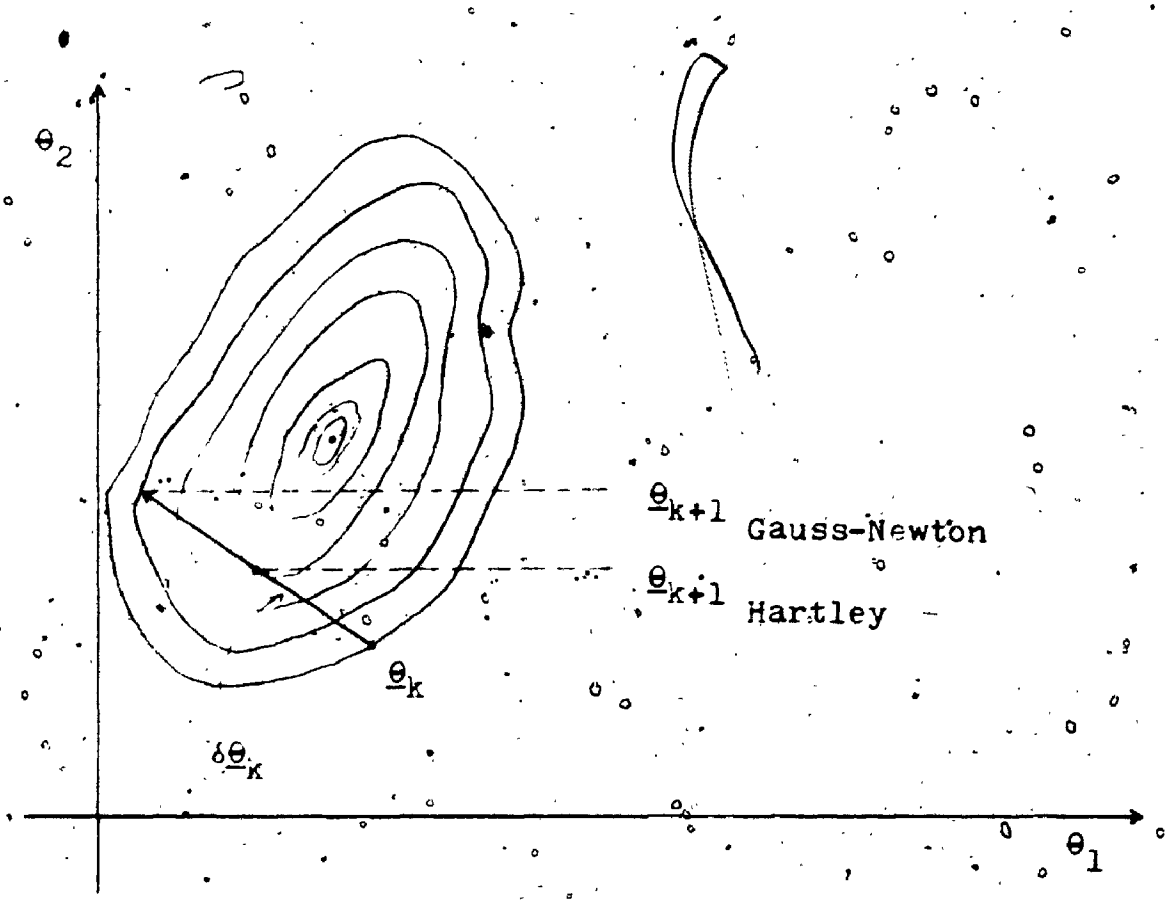


Figure 3.2 Graphical illustration of Hartley's modification versus the Gauss-Newton technique in a two dimensional space.

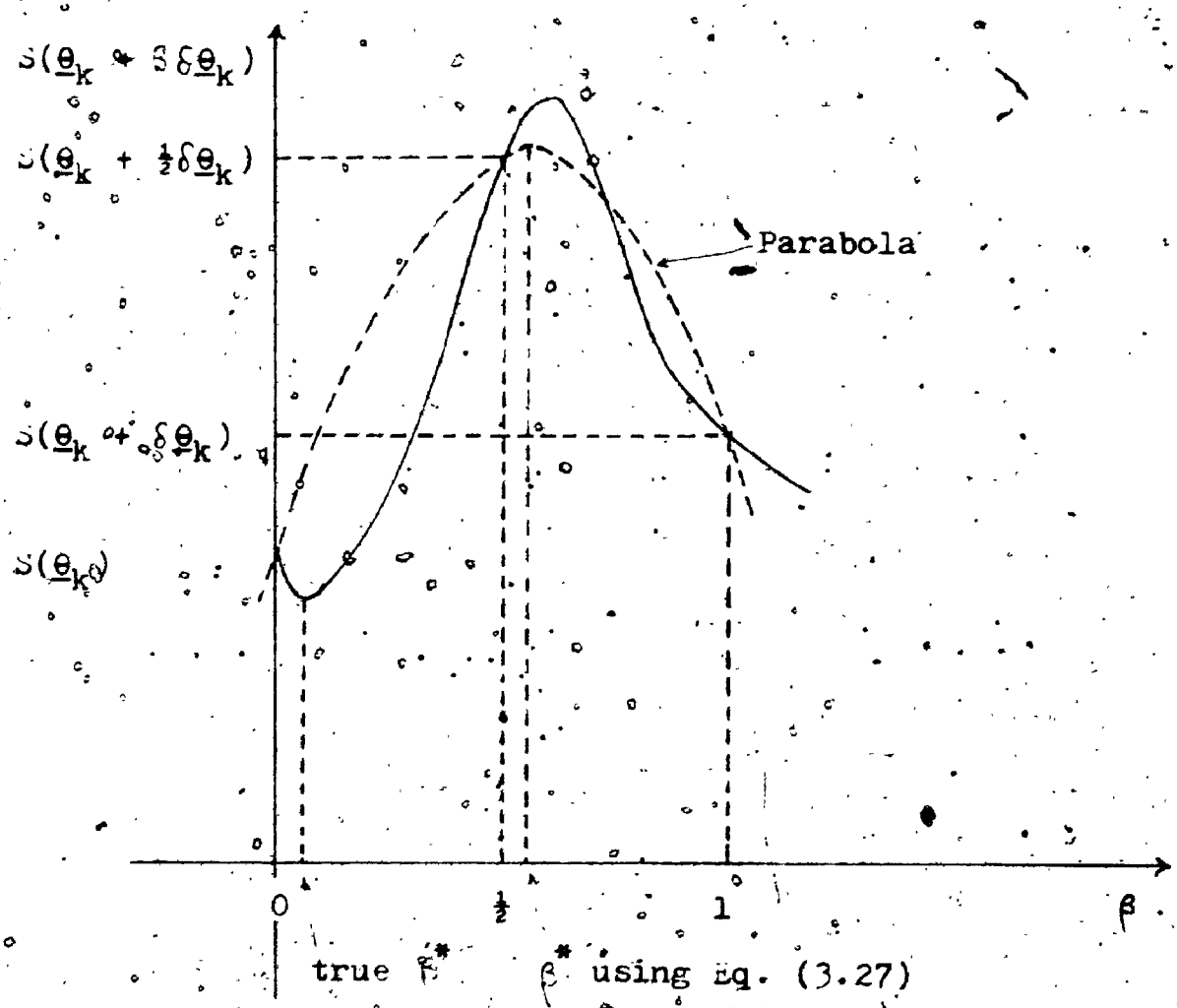


Figure 3.3 Graphical illustration of a poor quadratic interpolation to approximate β^* as suggested by Hartley.

IV.2 Marquardt's Incorporation with the Steepest Descent Technique

Marquardt (1963) found out that nonconvergence of the Gauss-Newton sequence $\{\underline{\theta}_k\}$ is frequently due to step-directions. In many practical problems, he observed that the Gauss-Newton steps are almost perpendicular to the steps derived from the Steepest Descent technique (Cooper and Steinberg 1970), an iterative minimization technique which provides a large region of convergence, but has a slow rate of convergence. Marquardt's idea is to obtain the best of the two techniques by interpolating between the steps derived by the two techniques.

From the Steepest Descent technique, at the $(k+1)^{th}$ iteration, the Steepest Descent step $\delta\underline{\theta}_{k_{SD}}$ is chosen as:

$$\delta\underline{\theta}_{k_{SD}} = -\frac{1}{\lambda}\underline{g}_k$$

or,

$$\delta\underline{\theta}_{k_{SD}} = -(\lambda\underline{I})^{-1}\underline{g}_k \qquad (3.28)$$

where $\underline{I}$ is an identity matrix, $\underline{g}_k$ is the gradient vector of S as defined by Eq. (3.16) or Eq. (3.20), and λ is some positive constant chosen with respect to factors such as convergence and rate of descent.

An interpolation between the Steepest Descent step defined by Eq. (3.28) and the Gauss-Newton step defined by Eq. (3.17) is:

$$\delta\underline{\theta}_{\text{Interpolation}} = -(\underline{H}_k + \lambda\underline{I})^{-1}\underline{g}_k \qquad (3.29)$$

which approaches Eq. (3.17) as $\lambda \longrightarrow 0$ and Eq. (3.28) as $\lambda \longrightarrow \infty$

Marquardt did not use the exact form of Eq. (3.29) Instead of adding a positive constant right away to each diagonal element of the matrix $\underline{H}_k$ in Eq. (3.17) to form Eq. (3.29), Eq. (3.17) was scaled first and the adding was undertaken. Note that in numerical computations, scaling a matrix or a vector is a frequent practice to avoid ill condition which may happen due to computer round-off. Scaling Eq. (3.17) gives:

$$\delta\underline{\theta}_k^* = -\underline{H}_k^*\underline{g}_k^* \qquad (3.30)$$

where $\underline{H}_k^*$, $\underline{g}_k^*$ and $\delta\underline{\theta}_k^*$ are related to $\underline{H}_k$, $\underline{g}_k$ and $\delta\underline{\theta}_k$ through Eqs. (3.31), (3.32) and (3.33),

$$(\underline{H}_k^*)_{ij} = (\underline{H}_k)_{ij}/\left[(\underline{H}_k)_{ii}(\underline{H}_k)_{jj}\right]^{\frac{1}{2}} \qquad (3.31)$$

$$(\underline{g}_k^*)_i = (\underline{g}_k)_i/\left[(\underline{H}_k)_{ii}\right]^{\frac{1}{2}} \qquad (3.32)$$

$$(\delta\underline{\theta}_k^*)_i = (\delta\underline{\theta}_k)_i\left[(\underline{H}_k)_{ii}\right]^{\frac{1}{2}} \qquad (3.33)$$

the scaled Marquardt step then is:

$$\delta\underline{\theta}_k^* = -(\underline{H}_k^* + \lambda\underline{I})^{-1}\underline{g}_k^* \qquad (3.34)$$

Finally, $\underline{\theta}_{k+1}$ is formed by adding the unscaled Marquardt step to $\underline{\theta}_k$:

$$\underline{\theta}_{k+1} = \underline{\theta}_k + \delta\underline{\theta}_k \qquad (3.35)$$

Marquardt (1963) also proposed a method to compute λ. It is as follows:

1. Let γ be a constant greater than unity.

2. Let λ_k denote the value of λ from the previous iteration (the k^{th} iteration). Initially let λ_0 be a small constant, say 10^{-1}.

3. Compute the values of S at $\underline{\theta} = \underline{\theta}_{k+1}\Big|_{\lambda = \lambda_k}$ and at $\underline{\theta} = \underline{\theta}_{k+1}\Big|_{\lambda = \lambda_k/\gamma}$. Let them be denoted by $S(\lambda_k)$ and $S(\lambda_k/\gamma)$.

4. Let s_k be the value of S corresponding to the k^{th} iteration. Compare $S(\lambda_k)$ and $S(\lambda_k/\gamma)$ with s_k. If

   (a) $S(\lambda_k/\gamma) \leq s_k$, let $\lambda_{k+1} = \lambda_k/\gamma$ and $s_{k+1} = S(\lambda_k/\gamma)$.

   (b) $S(\lambda_k/\gamma) > s_k$ and $S(\lambda_k) \leq s_k$, let $\lambda_{k+1} = \lambda_k$ and $s_{k+1} = S(\lambda_k)$.

   (c) $S(\lambda_k/\gamma) > s_k$ and $S(\lambda_k) > s_k$, increase λ by successive multiplication by γ until for some smallest integer n such that $S(\lambda_k\gamma^n) \leq s_k$, then let $\lambda_{k+1} = \lambda_k\gamma^n$ and $s_{k+1} = S(\lambda_k\gamma^n)$.

An illustration of Marquardt's modification in a two dimensional space is shown on Figure 3.4.

Note that Marquardt's idea is the same as Levenberg's (1944) but somewhat different in reasoning.

Marquardt let λ's take only positive values. Smith Jr. and Shanno have improved Marquardt's modification by allowing λ's to take negative values whenever possible. For more details, see Smith Jr. and Shanno (1971).

V THE QUASILINEARIZATION TECHNIQUE

The Quasilinearization technique was first developed by Bellman (Bellman et al 1963), the father of Dynamic Programming. Besides its use in solving a nonlinear boundary value problem (Bellman and Kalaba 1965, Lee 1968), the Quasilinearization technique can also be used in solving a system identification problem (Bellman et al 1963, Zupp and Childs 1969, Sage and Melsa 1971b) and is the subject of this section.

To use the Quasilinearization technique, the state space model defined by Eqs. (3.1) and (3.2) must be put into another form. Let

$$\underline{v}(t) = \begin{bmatrix} \underline{a} \\ \hline \underline{x}(t) \end{bmatrix} \qquad (3.36)$$

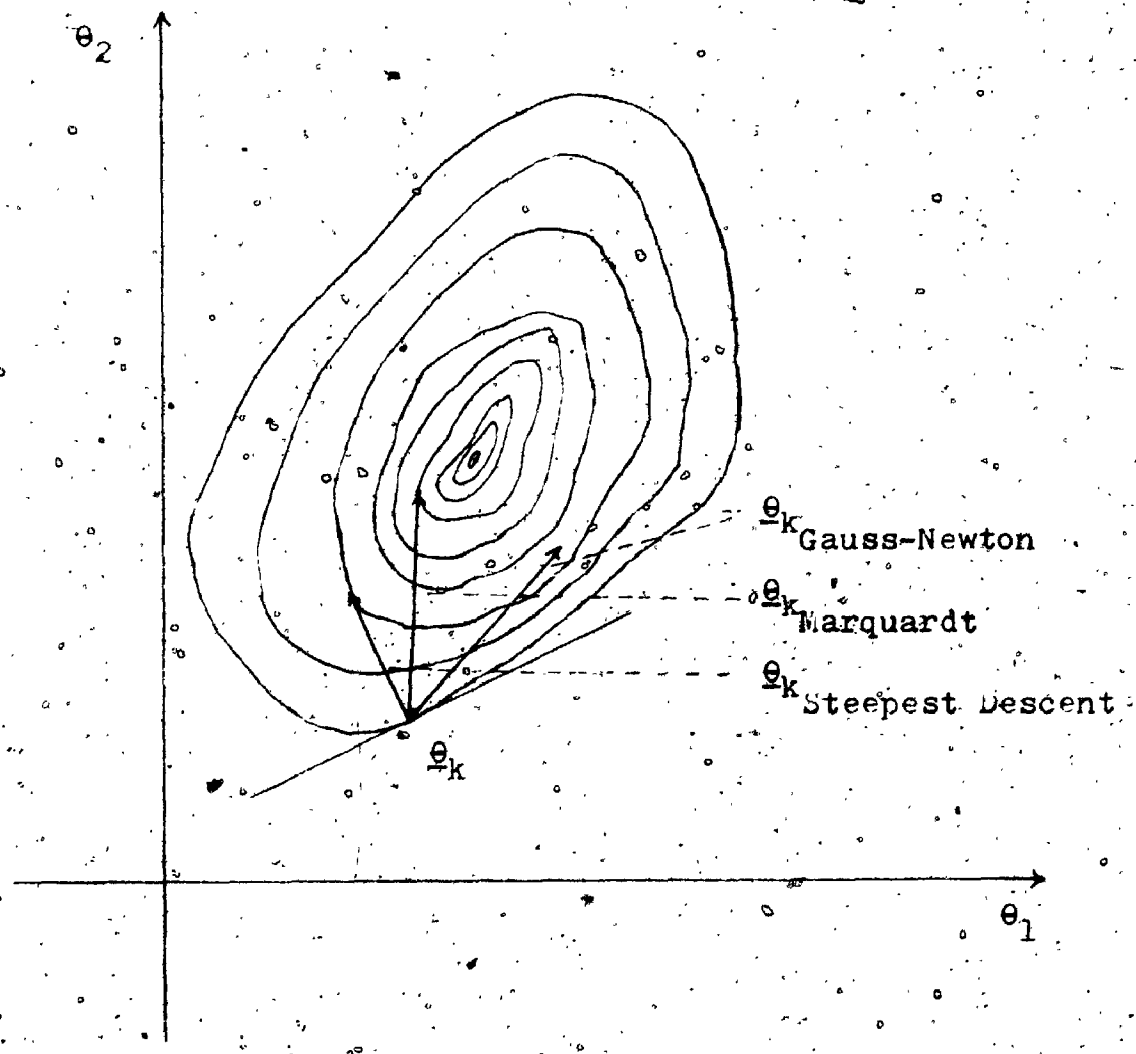


Figure 3.4 Graphical illustration of Marquardt's modification versus the Steepest Descent technique and the Gauss-Newton technique in a two dimensional space.

then Eqs. (3.5) and (3.6) become:

$$\dot{\underline{v}}(t) = \underline{f}^{\#}(\underline{v}(t), \underline{u}_m(t), t) \tag{3.37}$$

$$\underline{y}(t) = \underline{g}^{\#}(\underline{v}(t), \underline{u}_m(t), t) \tag{3.38}$$

From Eqs. (3.3) and (3.36), it should be noted that:

$$\underline{v}(t_o) = \underline{\theta} \tag{3.39}$$

From the quasilinearization technique, at the beginning of the $(k+1)^{th}$ iteration, an approximate trajectory $\underline{v}_k(t)$ for all $t_o \leq t \leq t_f$ is available (Initially the whole trajectory $\underline{v}_o(t)$ $t_o \leq t \leq t_f$ or just $\underline{v}_o(t_o)$ must be supplied). Then if $\underline{f}^{\#}$ and $\underline{g}^{\#}$ are replaced by their linear parts when expanded around $\underline{v}_k(t)$, Eqs. (3.37) and (3.38) become:

$$\dot{\underline{v}}(t) = \underline{f}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t) + \left[\frac{\partial \underline{f}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right]\left[\underline{v}(t) - \underline{v}_k(t)\right] \tag{3.40}$$

$$\underline{y}(t) = \underline{g}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t) + \left[\frac{\partial \underline{g}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right]\left[\underline{v}(t) - \underline{v}_k(t)\right] \tag{3.41}$$

Combining Eqs. (3.37) and (3.40) results in Eq. (3.42):

$$\delta \dot{\underline{v}}_k(t) = \left[\frac{\partial \underline{f}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right]\delta \underline{v}_k(t) \tag{3.42}$$

provided,

$$\delta \underline{v}_k(t) = \underline{v}(t) - \underline{v}_k(t) \tag{3.43}$$

The general solution of the linear differential equation, Eq. (3.42), can be put into the following form:

$$\delta \underline{v}_k(t) = \underline{\Phi}_k(t)\delta \underline{v}_k(t_o) \tag{3.44}$$

where $\underline{\Phi}_k(t)$ is known as the state transition matrix of Eq. (3.42) and can be found from Eq. (3.45),

$$\dot{\underline{\Phi}}_k(t) = \left[\frac{\partial \underline{f}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right]\underline{\Phi}_k(t)$$
$$\underline{\Phi}_k(t_o) = \underline{I} = \text{Identity matrix} \tag{3.45}$$

Combining Eqs. (3.41), (3.43) and (3.44) results in:

$$\underline{y}(t) = \underline{g}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t) + \left[\frac{\partial \underline{g}^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T}\right]\underline{\Phi}_k(t)\delta \underline{v}_k(t_o) \tag{3.46}$$

From Eqs. (3.46) and (3.7), the value of S will then be:

$$S = \frac{1}{2}\sum_{i=0}^{I} \left\| \underline{y}_m(t_i) - \underline{g}^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i) - \left[\frac{\partial \underline{g}^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)}\right]\underline{\Phi}_k(t_i)\delta \underline{v}_k(t_o) \right\|^2_{\underline{V}_i} \tag{3.47}$$

Setting the derivative of S with respect to $\delta \underline{v}(t_o)$ to zero gives:

$$- \sum_{i=0}^{I} \Phi_k^T(t_i) \left[\frac{\partial g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)} \right]^T \underline{V}_i$$

$$\left\{ \underline{y}_m(t_i) - g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i) \right.$$

$$\left. - \left[\frac{\partial g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)} \right] \Phi_k(t_i) \delta \underline{v}_k(t_o) \right\} = 0 \qquad (3.48)$$

or after rearranging,

$$\delta \underline{v}_k(t_o) = - (\underline{H}_k^{\#})^{-1} \underline{g}_k^{\#} \qquad (3.49)$$

where,

$$\underline{H}_k^{\#} = \sum_{i=0}^{I} \Phi_k^T(t_i) \left[\frac{\partial g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)} \right]^T \underline{V}_i$$

$$\left[\frac{\partial g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)} \right] \Phi_k(t_i) \qquad (3.50)$$

and,

$$\underline{g}_k^{\#} = - \sum_{i=0}^{I} \Phi_k^T(t_i) \left[\frac{\partial g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i)}{\partial \underline{v}_k^T(t_i)} \right]^T \underline{V}_i$$

$$\left[\underline{y}_m(t_i) - g^{\#}(\underline{v}_k(t_i), \underline{u}_m(t_i), t_i) \right] \qquad (3.51)$$

For the case where continuous data are available and used, then instead of Eqs. (3.50) and (3.51), $\underline{H}_k^{\#}$ and $\underline{g}_k^{\#}$ will be defined as in Eqs. (3.52) and (3.53),

$$\underline{H}_k^{\#} = \int_{t_o}^{t_f} \underline{\Phi}_k^T(t)\left[\frac{\partial \underline{g}^{\#}(\underline{y}_k(t), \underline{u}_m(t), t)}{\partial \underline{y}_k^T(t)}\right]^T \underline{v}(t)$$

$$\left[\frac{\partial \underline{g}^{\#}(\underline{y}_k(t), \underline{u}_m(t), t)}{\partial \underline{y}_k^T(t)}\right] \underline{\Phi}_k(t)\,dt \qquad (3.52)$$

$$\underline{g}_k^{\#} = -\int_{t_o}^{t_f} \underline{\Phi}_k^T(t)\left[\frac{\partial \underline{g}^{\#}(\underline{y}_k(t), \underline{u}_m(t), t)}{\partial \underline{y}_k^T(t)}\right]^T \underline{v}(t)$$

$$\left[\underline{y}_m(t) - \underline{g}^{\#}(\underline{y}_k(t), \underline{u}_m(t), t)\right] dt \qquad (3.53)$$

$\delta \underline{y}_k(t_o)$ found in Eq. (3.49) is a step in $\underline{y}(t_o)$, the result of which will become the new approximate value of $\underline{y}(t_o)$ at the $(k+1)^{\text{th}}$ iteration,

$$\underline{y}_{k+1}(t_o) = \underline{y}_k(t_o) + \delta \underline{y}_k(t_o) \qquad (3.54)$$

V.1 Algorithm

The following is an algorithm for the Quasilinearization technique:

1. An initial approximation $\underline{y}_o(t_o)$ is supplied either

by guessing or by some approximate technique. Let k = 0.

2. Compute $\underline{y}(t_i)$ for all i at $\underline{y}(t_o) = \underline{v}_k(t_o)$ from Eqs. (3.37) and (3.38) using a numerical integration technique such as Euler or Runge-Kutta (Haggerty 1972). If the integration technique used requires values of $\underline{u}_m$ at times between measurements, then an interpolation technique, say, the linear interpolation (Stark 1970) can be employed.

3. Compute $\underline{\Phi}_k(t_i)$ for all i from Eq. (3.45) by a numerical integration technique as mentioned in step 2.

4. Compute $\underline{H}_k^{\#}$ and $\underline{c}_k^{\#}$ from Eqs. (3.50) and (3.51).

5. Compute $\underline{v}_{k+1}(t_o)$ from Eqs. (3.51) and (3.49).

6. If one of the tests for convergence mentioned in Section III is met, then go to step 7. Otherwise let k ⟵ (k+1), then go to step 2.

7. Stop. The converged value $\underline{v}_{k+1}(t_o)$ is then a minimum point (not necessarily a global minimum point).

To enlarge the region of convergence, a modification of the quasilinearization technique was proposed by Ramaker *et al* (1967). However, this modification is just a rewritten form of Hartley's modified Gauss-Newton technique, because, as shown in Appendix A, the Quasilinearization technique and the Gauss-Newton technique are mathematically equivalent.

VI. APPLICATIONS

As illustrative examples, the simplified form of Hartley's modification mentioned at the end of Section III.1 (the stepsize is cut into half until there is a reduction in value of the sum of squares S) is used to find mathematical models describing microbial growth in a batch fermentation of Aureobasidium pullulans 2552 and a continuous fermentation of Morchella crassipes.

VI.1 Example 1 (with batch fermentation data)

By definition, the specific growth rate $\mu(t)$ is the ratio of the rate of change of biomass concentration $C(t)$ and the biomass concentration $C(t)$.

$$\mu(t) = \frac{1}{C(t)} \frac{dC(t)}{dt} \qquad (3.55)$$

The specific growth rate is a function of the total microbial environment. The environment can usually be considered as the medium temperature, the oxygen available for microbial metabolism (in aerobic fermentation), the degree of mixing, the medium pH, the biomass concentration, the product concentration and the substrate concentration. If the medium temperature, the aeration and the agitation are held fixed, then the specific growth rate is reduced to a function of the last four environmental factors.

$$\mu(t) = f\left[C(t), S(t), P(t), H(t)\right] \qquad (3.56)$$

where $S(t)$, $P(t)$ and $H(t)$ are defined as substrate concen-

tration, product concentration and pH of the culture at time t respectively.

Note that Eq. (3.56) can also be used when the operating conditions (medium temperature, aeration and agitation) are varied within certain ranges which do not really affect the dynamic behaviour of the process.

Eq. (3.56) is expanded into a power series to give;

$$\begin{aligned}\mu(t) = {} & \alpha_o + \alpha_1 C(t) + \alpha_2 [C(t)]^2 + \ldots + \alpha_n [C(t)]^n + \ldots \\ & + \beta_o + \beta_1 S(t) + \beta_2 [S(t)]^2 + \ldots + \beta_n [S(t)]^n + \ldots \\ & + \gamma_o + \gamma_1 P(t) + \gamma_2 [P(t)]^2 + \ldots + \gamma_n [P(t)]^n + \ldots \\ & + \delta_o + \delta_1 H(t) + \delta_2 [H(t)]^2 + \ldots + \delta_n [H(t)]^n + \ldots \end{aligned} \tag{3.57}$$

To simplify the model, it is reasonable to take into account only all linear terms of the power series, and perhaps some higher order terms which are suspected to have significant effect on microbial growth. These terms will then be fitted and tested for their significance through an F-test (Appendix C). If only the linear terms are considered, then,

$$\mu(t) = a_1 + a_2 C(t) + a_3(t) + a_4(t) + a_5(t) \tag{3.58}$$

or combining Eqs. (3.58) and (3.55),

$$\frac{dC(t)}{dt} = C(t)\left[a_1 + a_2 C(t) + a_3 S(t) + a_4 P(t) + a_5 H(t)\right] \tag{3.59}$$

where a's are parameters depending on the operating conditions.

Seven batches of Aureobasidium pullulans 2552 have been grown and data have been collected as described in Appendix B. These data will now be used to identify the parameters $a_1, a_2, \ldots, a_5$. Since there are seven batch runs involved, the sum of squares to be minimized is:

$$S = \frac{1}{2} \sum_{j=1}^{7} \sum_{i=0}^{I^j} \left[C_m^j(t_i^j) - C^j(t_i^j)\right]^2 \qquad (3.60)$$

where

i = Subscript to denote from the $(i+1)^{th}$ measurement,

j = Superscript to denote from the j^{th} batch,

$I^j + 1$ = Number of measurements in the j^{th} batch,

t_i^j = Time at the $(i+1)^{th}$ measurement in the j^{th} batch,

C_m^j = Measured biomass concentration in the j^{th} batch,

C^j = Model biomass concentration in the j^{th} batch which is found from the following equation:

$$\frac{dC^j(t)}{dt} = C^j(t)\left[a_1 + a_2 C^j(t) + a_3 S_m^j(t) + a_4 P_m^j(t) + a_5 H_m^j(t)\right] \qquad (3.61)$$

with S_m, P_m and H_m being measured substrate concentration, measured product concentration and measured pH respectively.

Since initial conditions for each batch must also

be identified (Chapter IV), the sum of squares S will be minimized with respect to $\underline{\theta}$ where

$$\underline{\theta} = \left[a_1 \ldots a_5\ C^1(t_o^1) \ldots C^7(t_o^7)\right]^T \qquad (3.62)$$

Based on the simplified form of Hartley's modification and the backward elimination technique (Appendix C), a computer program, listed as Program I in Appendix E, was written in the FORTRAN language and run on the university's time-sharing computer DEC-PDP-10. With a 95%-confidence level, the mathematical model describing the microbial growth in the batch fermentation of A. pullulans was found to be as:

$$\frac{dC(t)}{dt} = C(t)\left[(.0243 \pm .0045) + (.0014 \pm .0005)S(t) - (.0058 \pm .0015)P(t)\right] \qquad (3.63)$$

with the minimum sum of squares $\hat{S} = .289$, the estimated standard deviation $\hat{\sigma} = .094$, and the estimated initial conditions $\hat{C}^1(t_o^1) = (.0757 \pm .0236)$, $\hat{C}^2(t_o^2) = (.1820 \pm .0421)$, $\hat{C}^3(t_o^3) = (.0847 \pm .0283)$, $\hat{C}^4(t_o^4) = (.1268 \pm .0302)$, $\hat{C}^5(t_o^5) = (.1918 \pm .0436)$, $\hat{C}^6(t_o^6) = (.1086 \pm .0367)$, and $\hat{C}^7(t_o^7) = (.1363 \pm .0330)$. Note that C(t), S(t) and P(t) are all in g/100ml.

The model biomass concentrations obtained from Eq. (3.63) together with the measured biomass concentrations for a batch, say batch #4, are shown in Figure 3.5. From the result in Eq. (3.63), one can draw a conclusion that under the set operating conditions:

BATCH # 4

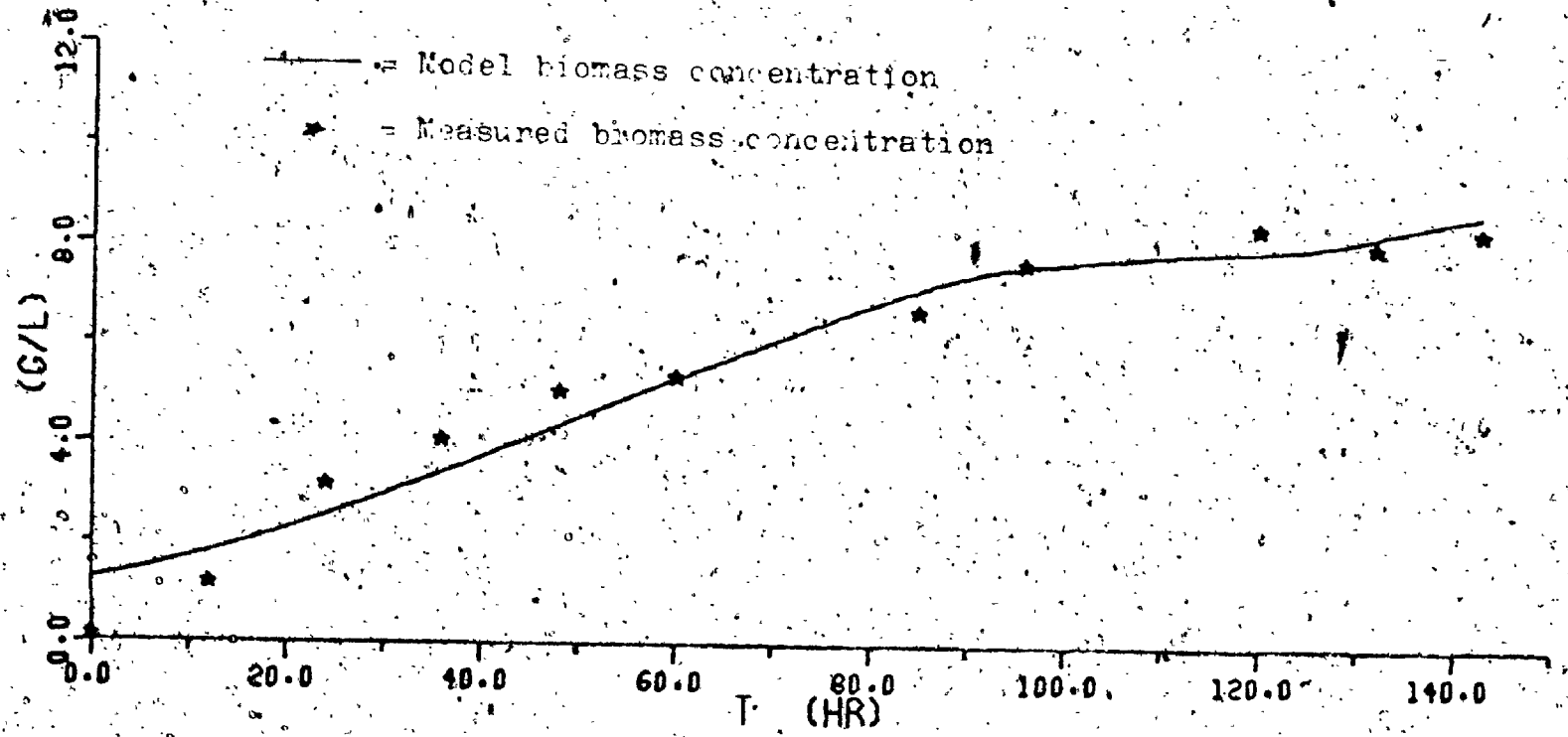


Figure 3.5 Model biomass concentration and measured biomass concentration of A. pullulans 2552 in batch # 4.

- The medium pH does not have any significant effect on the specific growth rate,

- The biomass concentration does not have any significant effect on the specific growth rate,

- The product concentration has a negative effect on the specific growth rate. That is, the higher the product concentration, the lower is the specific growth rate ($\hat{a}_4$ = - .0058),

- The substrate concentration has a positive effect on the specific growth rate. That is, the higher the substrate concentration, the higher is the specific growth rate ($\hat{a}_3$ = .0014).

Table 4.1 shows the iterative results produced by the algorithm based on a modification of the Gauss-Newton technique when a_2 and a_5 were set to zero (were not fitted). The parameters a_1, a_3 and a_4 initially took the values of .0200, .0016 and -.0066 and converged to .0243, .0014 and -.0058 in four iterations. The test for convergence was based on Eq. (3.23) with $\alpha_1 = 10^{-3}$ and $\alpha_2 = 10^{-15}$.

The Monod model and the Verhulst-Pearl model were also tried. However, the results using these models were found to be inferior to that obtained by Eq. (3.63). The Monod model is:

$$\mu(t) = \frac{\mu_m S(t)}{S(t) + K_s} \qquad (3.64)$$

Table 3.1 Results obtained from a simplified form of Hartley's modified Gauss-Newton technique when a_2 and a_5 are set to zero.

Iteration	$\hat{a}_1$	$\hat{a}_3$	$\hat{a}_4$	S
0	.0200	.0016	-.0066	2.318
1	.0264	.0011	-.0054	.693
2	.0237	.0013	-.0055	.291
3	.0244	.0014	-.0058	.289
4	.0243	.0014	-.0058	.289

or,

$$\frac{dC(t)}{dt} = C(t)\left[\frac{\mu_m S(t)}{S(t) + K_s}\right] \qquad (3.65)$$

By the least squares technique, the result found is:

$$\hat{\mu}_m = .0070 \pm .0011$$

$$\hat{K}_s = -.1555 \pm .0014$$

$$\hat{S} = 2.011$$

and $\hat{\Gamma} = .247$

The Monod model is not favorable because the estimated saturation coefficient $\hat{K}_s$ takes a negative value which is unacceptable, also it should be noted that the minimum sum of squares is far greater (2.011 versus .289).

The Verhulst-Pearl model is:

$$\mu(t) = a_1 + a_2 C(t) \qquad (3.66)$$

or,

$$\frac{dC(t)}{dt} = C(t)\left[a_1 + a_2 C(t)\right] \qquad (3.67)$$

By the least squares technique, the result found is:

$$\hat{a}_1 = .0404 \pm .0109$$

$$\hat{a}_2 = .0413 \pm .0144$$

$$\hat{S} = .526$$

and $\hat{\Gamma} = .126$

The Verhulst-Pearl model is better than the Monod model

($\hat{S}$ = .525 versus $\hat{S}$ = 2.011). However, it still produces higher minimum sum of squares when compared to the model presented by Eq. (3.63) and therefore would not be preferred.

VI.2 Example 2 (with continuous fermentation data)

From Chapter II, for a continuous fermentation process governed by the Monod logistic law, the mathematical equations representing the process are:

$$\dot{C}(t) = \left[\frac{\mu_m S(t)}{S(t) + K_s} - K - D\right]C(t) \qquad (3.68)$$

$$\dot{S}(t) = D\left[S_I - S(t)\right] - \frac{\mu_m S(t) C(t)}{Y\left[S(t) + K_s\right]} \qquad (3.69)$$

If measurements on the process are taken at discrete times t_i for $i = 0, 1, \ldots, I$, and C_m and S_m denote measured effluent biomass concentration and substrate concentration, then from the least squares criterion the process is identified by minimizing

$$S = \frac{1}{2}\sum_{i=0}^{I}\left\{V_1\left[C_m(t_i) - C(t_i)\right]^2 + V_2\left[S_m(t_i) - S(t_i)\right]^2\right\} \qquad (3.70)$$

over $\underline{\theta}$. Where V_1 and V_2 are some appropriate positive weighting constants and

$$\underline{\theta} = \left[\mu_m \quad K_s \quad K \quad Y \quad C(t_o) \quad S(t_o)\right]^T \qquad (3.71)$$

Based on the simplified form of Harley's modifica-

tion, a computer program, listed as Program VIII in Appendix E, was written to compute $\underline{\theta}$ and their 95%-confidence intervals (Appendix C) for a given set of continuous data.

An experiment on a continuous fermentation of Morchella crassipes was carried out and data has been collected (LeDuy 1975). From these data where the influent substrate (total carbohydrate) concentration $S_I = 5$ g/ℓ and the dilution rate $D = .1\ hr^{-1}$, the result found is:

$$\hat{\theta}_1 = \hat{\mu}_m = .100 \pm .861$$
$$\hat{\theta}_2 = \hat{K}_s = .142 \pm 1.456$$
$$\hat{\theta}_3 = \hat{K} = .003 \pm .868$$
$$\hat{\theta}_4 = \hat{Y} = .478 \pm 4.416$$
$$\hat{\theta}_5 = \hat{C}(t_o) = 1.537 \pm .249$$
$$\hat{\theta}_6 = \hat{S}(t_o) = 1.803 \pm .233$$

and $\hat{S} = 13.42$

The values chosen for V_1 and V_2 were the dispersions (the inverses of the variances) of measured biomass and substrate concentrations which were estimated by Leduy (1975) as 17.54 and 47.62, respectively. The model and measured effluent concentrations are shown in Figures 3.6 and 3.7.

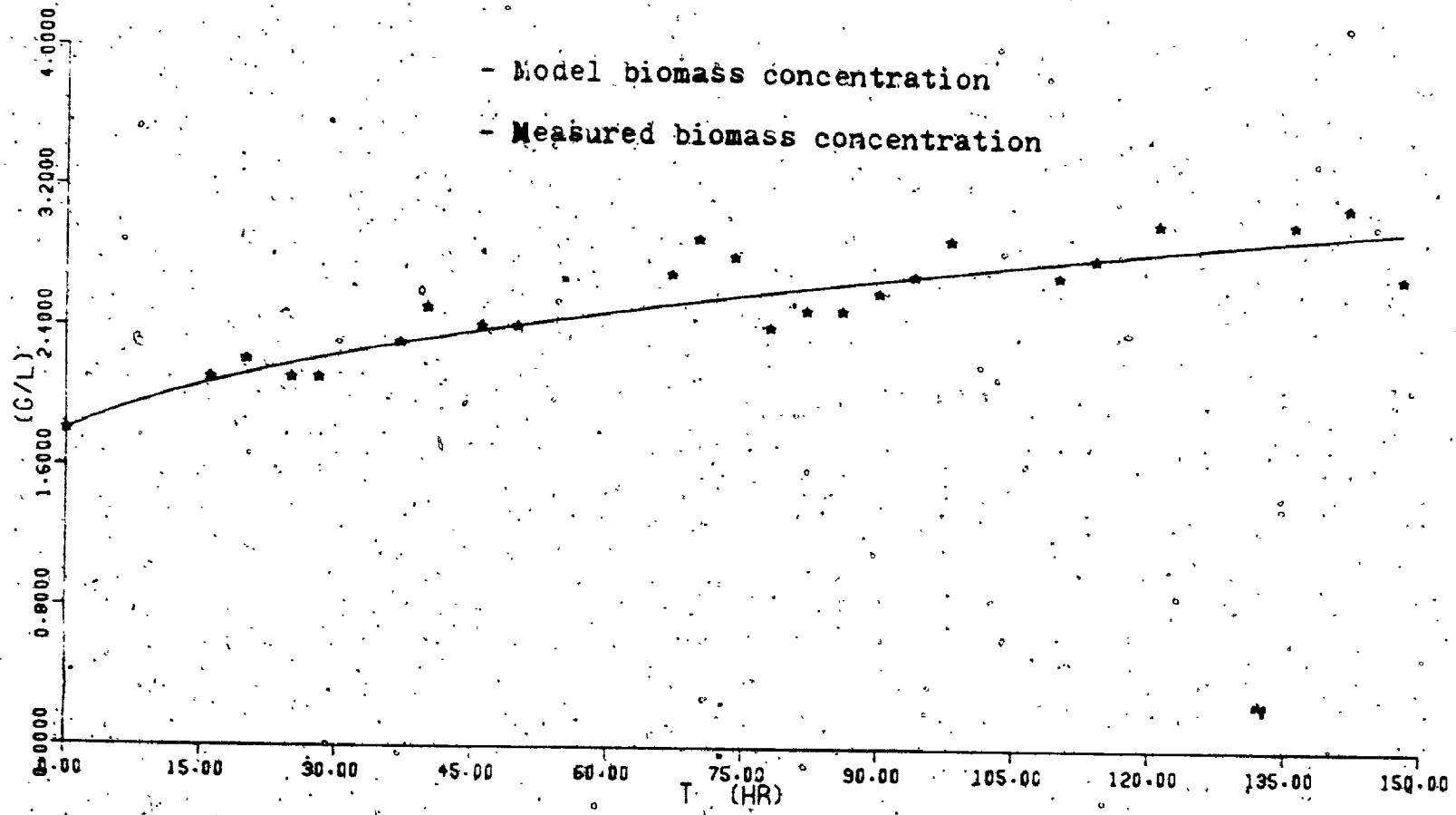


Figure 3.6 Model biomass concentration and measured biomass concentration in a continuous fermentation of M. crassipes.

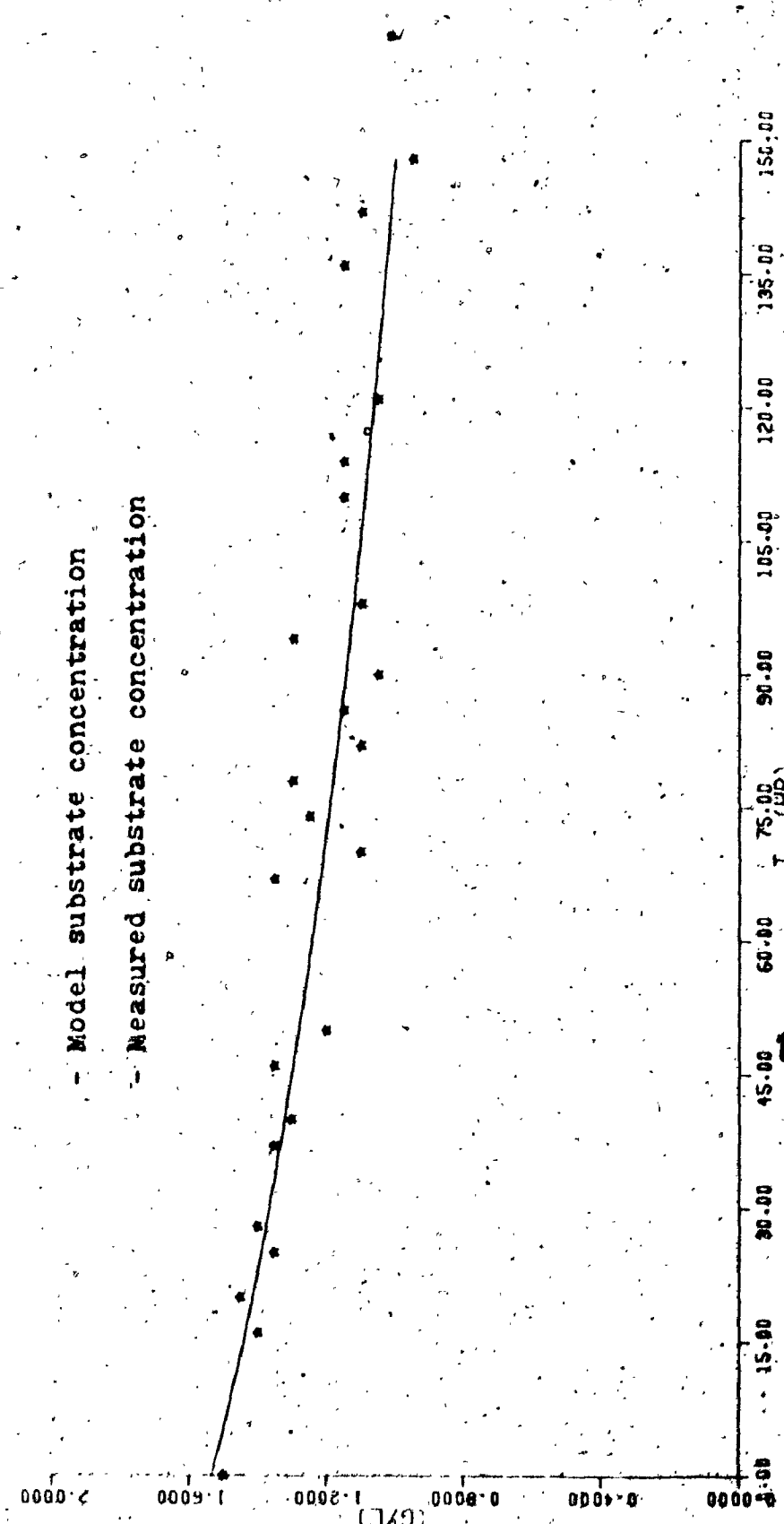


Figure 3.7 Model substrate concentration and measured substrate concentration in a continuous fermentation of *M. crassipes*.

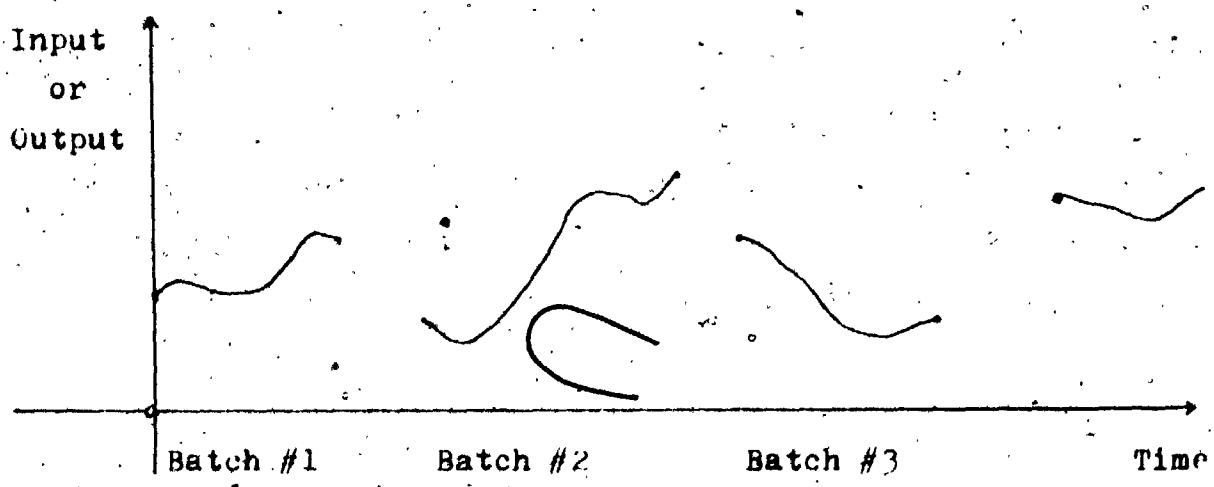
CHAPTER IV

LINEAR DYNAMIC BATCH PROCESS IDENTIFICATION

I. INTRODUCTION

There has been a tendency at the industry level to replace large-scale batch operations by continuous processes primarily for cost reduction reasons. However due to the nature of many processes and the materials involved, many are still carried out in batches. In batch processing, the operating time for each batch is short, finished products are available for collection only at the end of each batch operation and a variety of products are often produced depending on operating conditions for that particular batch. Whereas in continuous processing, the operating time is long, finished products are produced continuously and are often controlled to be as uniformly as possible. In fermentations, as mentioned in Chapter I, a process is considered to be run in batch if the inflow rate and the outflow rate are zero, otherwise it is a continuous process. The difference between a batch process and a continuous process is shown on Figure 4.1.

The duration of operation of a single batch is short, giving either an inadequate number of measurements or an inadequate variety of data for purposes of process identification. Therefore it is necessary to use data from many



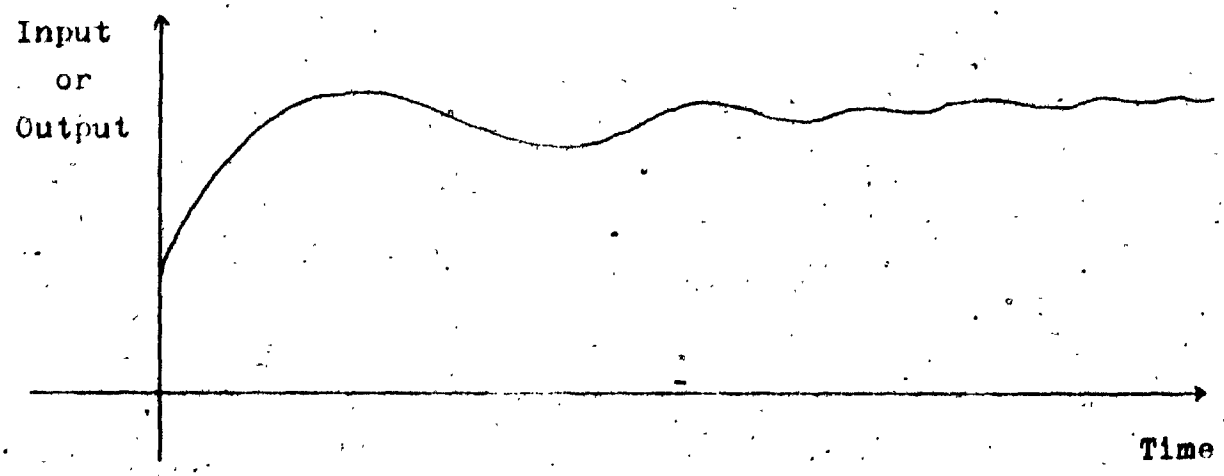


Figure 4.1 Difference between a batch process and a continuous process.

batch runs and this results in the estimation of a very large number of parameters as estimation of the initial conditions for each batch are also required.

Mathematically, consider a dynamic batch process with a general $n^{\underline{th}}$ order state space model of the form as described by Equations (3.1) and (3.2). Suppose measurements on the input vector $\underline{u}$ and on the output vector $\underline{y}$, denoted by $\underline{u}_m$ and $\underline{y}_m$, are available at time t_i^j for the $(i+1)^{\underline{th}}$ measurement in the $j^{\underline{th}}$ batch for $i = 0,1,\ldots,I^j$ and $j = 1,2,\ldots,B$, then the least squares criterion is:

$$\underset{\underline{\theta}}{\text{Min.}}\ S = \frac{1}{2}\sum_{j=1}^{B}\sum_{i=0}^{I^j}\left\|\underline{y}_m^j(t_i^j) - \underline{y}^j(t_i^j)\right\|^2_{\underline{V}_i^j} \qquad (4.1)$$

subject to:

$$\dot{\underline{x}}^j(t) = \underline{f}(\underline{x}^j(t), \underline{u}_m^j(t), \underline{a}, t) \qquad (4.2)$$

$$\underline{y}^j(t) = \underline{g}(\underline{x}^j(t), \underline{u}_m^j(t), \underline{a}, t) \qquad (4.3)$$

$$j = 1,2,\ldots,B$$

where

i = Subscript to denote from the $(i+1)^{\underline{th}}$ measurement,

j = Superscript to denote from the $j^{\underline{th}}$ batch,

I^j = Number of measurements in the $j^{\underline{th}}$ batch minus one,

B = Total number of batch data used,

$\underline{V}_i^j$ = Known symmetric, positive definite weighting matrix,

$$\underline{\theta} = \left[\underline{a}^T \mid \{\underline{x}^1(t_o^1)\}^T \mid \ldots \mid \{\underline{x}^B(t_o^B)\}^T\right]^T \qquad (4.4)$$

Let N and n be the dimension of the process parameter vector $\underline{a}$ and that of the state vector $\underline{x}(t)$ respectively, then $\underline{\theta}$ is a vector of $(N + nB)$ elements. Thus to minimize S over $\underline{\theta}$ is very time-consuming or even prohibitive if B is large.

To reduce the number of parameters to be estimated, Woodside and Xuyen (1973), while attempting to develop a bath-temperature model in a Stelco-Lurgi-Republic National (SL-RN) steelmaking process, have tried to estimate the initial conditions from a static model, using data available between batches.

$$\underline{x}^j(t_o^j) = \underline{h}(\underline{z}^j, \underline{b}) + \underline{\eta}^j \qquad (4.5)$$

where

$\underline{h}$ = Known vector function,

$\underline{z}^j$ = Data vector available between the $(j-1)^{\text{th}}$ and the j^{th} batch,

$\underline{\eta}^j$ = Random error vector in the static model,

$\underline{b}$ = $(M \times 1)$ parameter vector in the static model, unknown and to be estimated.

By assuming that $\underline{e}^j(t_i^j) = \underline{y}_m^j(t_i^j) - \underline{y}^j(t_i^j)$ and $\underline{\eta}^j$ for all j

and i are independent random vectors which have Gaussian distributions with mean vectors $\underline{0}$ and covariance matrices $\sigma^2\underline{R}_\epsilon$ and $\sigma^2\underline{R}_\eta$ respectively, and that the rest of the variables are deterministic, they derived an approximate maximum likelihood estimator which results in the following nonlinear programming problem:

$$\underset{\underline{a},\underline{b},\sigma}{\text{Min.}}\ S_{MLF} = \frac{1}{2}\sum_{j=1}^{B}\sum_{i=0}^{I^j}\left\{\log|\underline{R}_i^j| + \left\|\underline{y}_m^j(t_i^j) - \underline{g}(\underline{x}^j(t_i^j),\underline{u}_m^j(t_i^j),\underline{a},t)\right|_{\underline{\eta}^j=\underline{0}}\right\|^2_{(\sigma^2\underline{R}_i^j)^{-1}}\Bigg\} \tag{4.6}$$

subject to Eqs. (4.2), 4.3) and (4.5), and where the matrix $\underline{R}_i^j$ is defined by Eq. (4.7):

$$\underline{R}_i^j = \left[\frac{d\underline{g}(\underline{x}^j(t_i^j),\underline{u}_m^j(t_i^j),\underline{a},t_i^j)}{d\underline{x}^j(t_i^j)}\right]\underline{R}_\eta \left[\frac{d\underline{g}(\underline{x}^j(t_i^j),\underline{u}_m^j(t_i^j),\underline{a},t_i^j)}{d\underline{x}^j(t_i^j)}\right]^T + \underline{R}_e \tag{4.7}$$

and $|\underline{R}_i^j|$ is the determinant of the matrix $\underline{R}_i^j$.

The dimension (M + N + 1) of the new optimization problem is far smaller than that of the original optimization problem, (N + nB), if the number of batches B is large. However there

are some drawbacks in their approach. Firstly, it requires the collection of extra data, data between batches, which may be costly. Secondly, the maximum likelihood estimator is only approximate; it is based on the assumption that the static model $\underline{h}$ fits the initial condition vectors $\underline{x}^j(t_o^j)$ for all j quite well so that only the linear terms of the McLaurin series of g with respect to $\underline{\eta}$ is good enough to approximate g, which may also be costly as it takes time and extra study to find a suitable vector function $\underline{h}$. And finally, minimizing S_{MLE} defined by Eq. (4.6) is still very time-consuming particularly when the dimension ℓ of the output vector $\underline{y}$ is high. As can be seen from Eq. (4.5), in order to calculate the value of S_{MLE} at certain nominal values of $\underline{a}$, $\underline{b}$ and Γ one must find the determinant and the inverse of the $(\ell \times \ell)$ matrix $\underline{R}_1^j$, $\sum_{j=1}^{B}(I^j + 1)$ times.

In this chapter, a better approach to overcome the difficulty in estimating a large number of parameters is presented. This approach is exact, does not require extra data or extra modeling as in the Woodside-Xuyen approach. The only requirement is that the state space model representing a batch process is linear in its state vector. It eliminates the initial conditions by replacing them by their estimates which are functions of the process parameters by utilizing the Superposition Principle (Distefano et al

1967, Sage and Melsa 1971b) and the idea "Elimination of Linear Parameters in Nonlinear Regression" which was initiated by Walling (1968) and refined by Lawton and Sylvestre (1971).

The number of parameters to be estimated from this new approach is a constant equal to the number of the process parameters and independent of the number of batches B from which data are used. Thus it also suggests the possibility of an approximate on-line identification scheme to update the model as data from each new batch are being received. One such scheme is developed here.

It should be noted that the model describing the bath-temperature in the SL-RN steelmaking process is linear and so is the model describing the microbial growth in a batch fermentation process because the specific growth rate $\mu(t)$, which is defined as the relative rate of change of biomass concentration C(t),

$$\mu(t) = \frac{1}{C(t)} \frac{dC(t)}{dt} \qquad (4.8)$$

in general does not depend on the biomass concentration but rather depends on factors such as substrate concentration, medium temperature and pH. The Monod and Haldane-Monod equations are two typical examples. Hence the approach is directly applicable for use in fermentor model identification.

It should be noted that from the view point of identification, a continuous process where data at some periods are missing or are not recorded can be treated as a batch process as shown in Figure 4.2 and therefore the approach can be employed if its state space model is linear.

II ELIMINATION OF INITIAL CONDITIONS

When the model representing a batch process is linear in its state vector, Equations (4.2) and (4.3) can be put into the following linear form:

$$\dot{\underline{x}}^j(t) = \underline{F}(\underline{u}_m^j(t), \underline{a}, t)\underline{x}^j(t) + \underline{f}(\underline{u}_m^j(t), \underline{a}, t) \qquad (4.9)$$

$$\underline{y}^j(t) = \underline{G}(\underline{u}_m^j(t), \underline{a}, t)\underline{x}^j(t) + \underline{g}(\underline{u}_m^j(t), \underline{a}, t) \qquad (4.10)$$

$$j = 1, 2 \ldots, B$$

where $\underline{F}$, $\underline{G}$, $\underline{f}$ and $\underline{g}$ are known matrix and vector functions respectively. Thus to identify a linear dynamic batch process is equivalent to to minimize S defined by Eq. (4.1) with respect to $\underline{\theta}$ defined by Eq. (4.4) subject to Eqs. (4.9) and (4.10).

Since Eq. (4.9) is a linear differential equation in the state vector $\underline{x}^j(t)$, the Superposition Principle can be applied and the solution can be put into the following form:

$$\underline{x}^j(t) = \underline{\Phi}^j(\underline{a}, t)\underline{x}^j(t_o^j) + \underline{\Psi}^j(\underline{a}, t) \qquad (4.11)$$

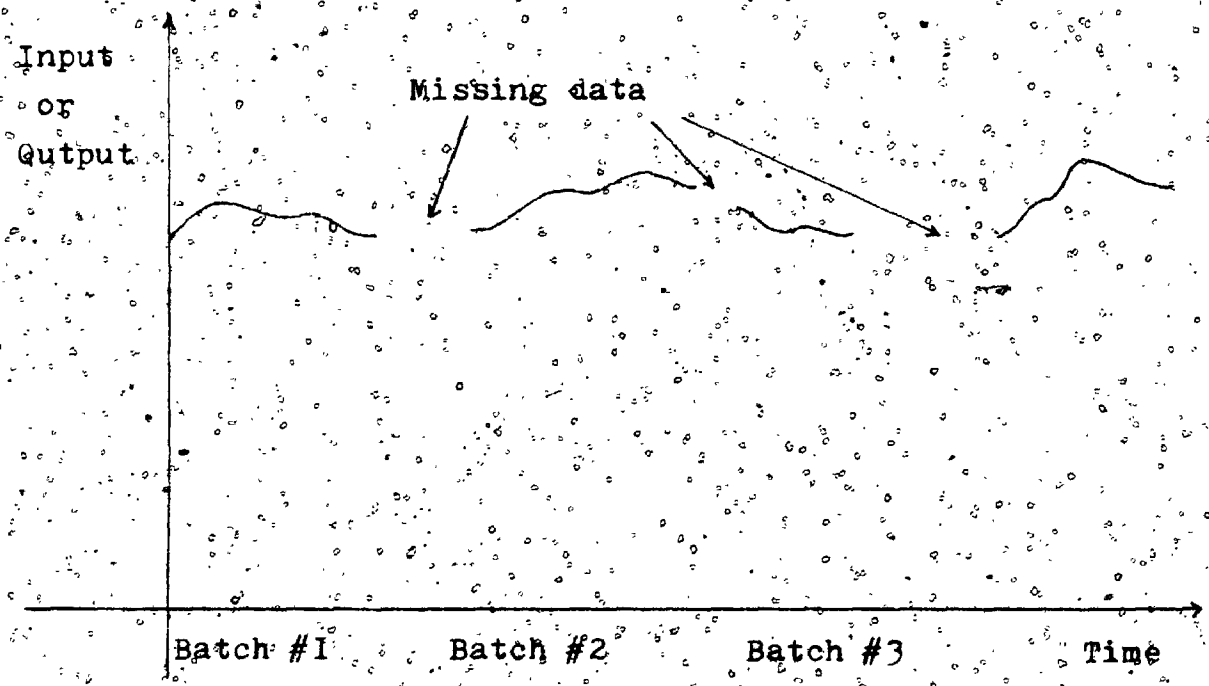


Figure 4.2 Continuous process with data at some periods missing or not recorded. For purposes of identification, it can be treated as a batch process.

where $\underline{\Phi}^j(\underline{a}, t)$ is the state transition matrix of the linear differential equation, Eq. (4.10), and can be computed from Eq. (4.12),

$$\dot{\underline{\Phi}}^j(\underline{a}, t) = \underline{F}(\underline{u}_m^j(t), \underline{a}, t)\underline{\Phi}^j(\underline{a}, t)$$
$$\underline{\Phi}^j(\underline{a}, t_o^j) = \underline{I} = \text{Identity matrix} \qquad (4.12)$$

and $\underline{\Psi}^j(\underline{a}, t)$ is the particular solution of the linear differential equation and can be computed from Eq. (4.13),

$$\dot{\underline{\Psi}}^j(\underline{a}, t) = \underline{F}(\underline{u}_m^j(t), \underline{a}, t)\underline{\Psi}^j(\underline{a}, t) + \underline{f}(\underline{u}_m^j(t), \underline{a}, t)$$
$$\underline{\Psi}^j(\underline{a}, t_o^j) = \underline{0} = \text{Null vector} \qquad (4.13)$$

Substitution of $\underline{x}^j(t)$ found in Eq. (4.11) into Eq. (4.10) results in:

$$\dot{\underline{y}}^j(t) = \underline{G}(\underline{u}_m^j(t), \underline{a}, t)\underline{\Phi}^j(\underline{a}, t)\underline{x}^j(t_o^j) + \underline{G}(\underline{u}_m^j(t), \underline{a}, t)\underline{\Psi}^j(\underline{a}, t) + \underline{g}(\underline{u}_m^j(t), \underline{a}, t) \qquad (4.14)$$

In Eq. (4.14), $\underline{y}^j(t)$ is linear in the initial condition vector $\underline{x}^j(t_o^j)$, therefore without losing any degree of accuracy, $\underline{x}^j(t_o^j)$ can be replaced by its estimate $\hat{\underline{x}}^j(\underline{a}, t_o^j)$ by using the Walling-Lawton-Sylvestre idea (Walling 1968, Lawton and Sylvestre 1971). $\hat{\underline{x}}^j(\underline{a}, t_o^j)$ is the value of $\underline{x}^j(t_o^j)$ that minimizes S when $\underline{a}$ is held fixed and can be found directly (not iteratively) because S is quadratic in $\underline{x}^j(t_o^j)$ as seen from Eqs. (4.1) and (4.14). Setting the derivative of S with respect to $\underline{x}^j(t_o^j)$ to zero gives:

$$\frac{d}{d\underline{x}^j(t_o^j)}\left\{\frac{1}{2}\sum_{k=1}^{B}\sum_{i=0}^{I^k}\left\|\underline{y}_m^k(t_i^k) - \underline{y}^k(t_i^k)\right\|^2_{\underline{V}_i^k}\right\} = \underline{0}$$

or since $\underline{y}^k(t_i^k)$ is independent of $\underline{x}^j(t_o^j)$ for $k \neq j$,

$$\frac{d}{d\underline{x}^j(t_o^j)}\left\{\sum_{i=0}^{I^j}\left\|\underline{y}_m^j(t_i^j) - \underline{y}^j(t_i^j)\right\|^2_{\underline{V}_i^j}\right\} = \underline{0} \qquad (4.15)$$

or from Eq. (4.14),

$$\sum_{i=0}^{I^j}\left[\underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\Phi}^j(\underline{a},t_i^j)\right]^T\underline{V}_i^j\left[\underline{y}_m^j(t_i^j) - \underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\psi}^j(\underline{a},t_i^j) - \underline{g}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j) - \underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\Phi}^j(\underline{a},t_i^j)\hat{\underline{x}}^j(\underline{a},t_o^j)\right] = \underline{0} \qquad (4.16)$$

or,

$$\hat{\underline{x}}^j(\underline{a},t_o^j) = -\left[\underline{H}^j(\underline{a})\right]^{-1}\underline{q}^j(\underline{a}) \qquad (4.17)$$

where,

$$\underline{H}^j(\underline{a}) = \sum_{i=0}^{I^j}\left[\underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\Phi}^j(\underline{a},t_i^j)\right]^T\underline{V}_i^j\left[\underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\Phi}^j(\underline{a},t_i^j)\right] \qquad (4.18)$$

$$\underline{q}^j(\underline{a}) = -\sum_{i=0}^{I^j}\left[\underline{G}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\underline{\Phi}^j(\underline{a},t_i^j)\right]^T\underline{V}_i^j\left[\underline{y}_m^j(t_i^j) - \underline{G}(\underline{u}_m(t_i^j),\underline{a},t_i^j)\underline{\psi}^j(\underline{a},t_i^j) - \underline{g}(\underline{u}_m^j(t_i^j),\underline{a},t_i^j)\right] \qquad (4.19)$$

In summary, the problem is reduced to:

$$\underset{\underline{a}}{\text{Min.}}\ S = \tfrac{1}{2} \sum_{j=1}^{B} \sum_{i=0}^{I^j} \left\| \underline{y}_m^j(t_i^j) - \underline{y}^j(\underline{a}, t_i^j) \right\|^2_{\underline{y}_i^j} \qquad (4.20)$$

where,

$$\underline{y}^j(\underline{a}, t_i^j) = \underline{G}(\underline{u}_m^j(t_i^j), \underline{a}, t_i^j)\underline{\Phi}^j(\underline{a}, t_i^j)\hat{\underline{x}}^j(\underline{a}, t_i^j) + \underline{g}(\underline{u}_m^j(t_i^j), \underline{a}, t_i^j) + \underline{G}(\underline{u}_m^j(t_i^j), \underline{a}, t_i^j)\underline{\Psi}^j(\underline{a}, t_i^j) \qquad (4.21)$$

$\underline{\Phi}^j(\underline{a}, t_i^j)$ is defined in Eq. (4.12),

$\underline{\Psi}^j(\underline{a}, t_i^j)$ is defined in Eq. (4.13), and

$\hat{\underline{x}}^j(\underline{a}, t_i^j)$ is defined in Eqs. (4.17), (4.18) and (4.19).

II.1 Algorithm

Since S defined in Eq. (4.20) is a minimum sum of squares, to minimize S with respect to $\underline{a}$, a modification of the Gauss-Newton technique should be used as stated by Bard (1970). The following is an algorithm to minimize S based on the simplified form of Hartley's modification which was reviewed in Chapter III.

1. An initial approximation $\underline{a}_o$ is supplied either by guessing or by some approximate technique. Let k = 0.

2. Compute $\underline{\Phi}^j(\underline{a}_k, t_i^j)$ and $\underline{\Psi}^j(\underline{a}_k, t_i^j)$ for all j and i from Eqs. (4.12) and (4.13).

3. Compute $\hat{\underline{x}}^j(\underline{a}_k, t_i^j)$ for all j from Eqs. (4.17), (4.18) and (4.19).

4. Compute $\underline{y}^j(\underline{a}_k, t_i^j)$ for all j and i from Eq. (4.21).

5. Compute $\partial \underline{y}^j(\underline{a}_k, t_i^j)/\partial \underline{a}_k^T$ for all j and i. It is often found more convenient to compute $\partial \underline{y}^j(\underline{a}_k, t_i^j)/\partial \underline{a}_k^T$ by using a finite difference approximation as described in Section II.1 of Chapter III.

6. Compute the Gauss-Newton step $\delta \underline{a}_k$ from the following equation:

$$\delta \underline{a}_k = - \underline{H}_k^{-1} \underline{g}_k \tag{4.22}$$

where,

$$\underline{H}_k = \sum_{j=1}^{B} \sum_{i=0}^{I^j} \left[\frac{\partial \underline{y}^j(\underline{a}_k, t_i^j)}{\partial \underline{a}_k^T}\right]^T \underline{V}_i^j \left[\frac{\partial \underline{y}^j(\underline{a}_k, t_i^j)}{\partial \underline{a}_k^T}\right] \tag{4.23}$$

$$\underline{g}_k = -\sum_{j=1}^{B} \sum_{i=0}^{I^j} \left[\frac{\partial \underline{y}^j(\underline{a}_k, t_i^j)}{\partial \underline{a}_k^T}\right]^T \underline{V}_i^j \left[\underline{y}_m^j(t_i^j) - \underline{y}^j(\underline{a}_k, t_i^j)\right] \tag{4.24}$$

7. Let

$$\underline{a}_{k+1} = \underline{a}_k + \beta^* \delta \underline{a}_k \tag{4.25}$$

where $\beta^* = 1, \frac{1}{2}, \frac{1}{4}, \ldots$ until $S(\underline{a}_{k+1}) \leq S(\underline{a}_k)$.

8. If

$$S(\underline{a}_{k+1}) - S(\underline{a}_k) \leq \alpha_1 \left[S(\underline{a}_k) + \alpha_2\right] \tag{4.26}$$

where α_1 and α_2 are some small positive constants, say, 10^{-4} and 10^{-10}, then go to step 9. Otherwise let $k \leftarrow (k + 1)$ and

then go to step 2.

9. Stop. The converged value $\underline{a}_{k+1}$ is then a minimum point.

Note that the converged value is only a local minimum point, to ensure that a global minimum point is found it is necessary to try several different values for the initial approximation $\underline{a}_o$. Also, since the technique does not guarantee the convergence of the sequence $\{\underline{a}_k\}$ if $\underline{a}_o$ is far away from a minimum point, two other stopping tests are needed: stop after, say, 15 iterations and stop after cutting half the Gauss-Newton step $\delta\underline{a}_{k_o}$, say, 15 times and still fail to produce a smaller sum of squares S.

II.2 Applications

The following are two examples, one with simulated data and the other with experimental data.

Example 1

The model is arbitrarily chosen as:

$$\dot{x}^j(t) = a_1 u_1^j(t) x^j(t) + a_2 u_2^j(t) \qquad (4.27)$$

$$y^j(t) = x^j(t) \qquad (4.28)$$

There are 5 batches of data used and there are 10 measurements in each batch. Therefore the sum of squares to be minimized is:

$$S = \frac{1}{2} \sum_{j=1}^{5} \sum_{i=0}^{9} \left[y_m^j(t_i^j) - y^j(t_i^j) \right]^2 \qquad (4.29)$$

The following values are chosen:

- $\underline{a} = [a_1 \quad a_2]^T = [-.5 \quad 2.]^T$
- $t_i^j = (.01)i$
- $\underline{u}_m^j(t_i^j) = \left[\bar{u}_{1m}^j(t_i^j) \quad \bar{u}_{2m}^j(t_i^j)\right]^T$
  $= \left[(-10. + 2q_{i1}^j) \quad (5. + q_{i2}^j)\right]^T$
  = Value of $\underline{u}_m^j(t)$ over the time interval (t_i^j, t_{i+1}^j).
- $x^j(t_o^j) = 5. + q^j$
- $y_m^j(t_i^j) = y^j(t_i^j) + e^j(t_i^j)$
- q^j, q_{i1}^j and q_{i2}^j = Randomly chosen integers from 0 to 10.
- $e^j(t_i^j)$ = Independent random variable having Gaussian distribution with zero-mean and σ^2-variance. For this example $\sigma^2 = 0$.

The results with and without the use of the Superposition Principle and the Walling-Lawton-Sylvestre idea (with and without elimination of the initial conditions) are given in Table 4.1. Note that to have a fair start for purposes of comparison, the initial approximations of the initial conditions $x^j(t_o^j)$ $j = 1,2,\ldots,5$ required when identification is done without eliminating the initial conditions are taken as $\hat{x}^j(a_o, t_o^j)$ $j = 1,2\ldots,5$ where the initial approximation of $\underline{a}$ is $\underline{a}_o = [-.4 \quad 1.6]^T$.

From Table 4.1, as expected, the case with elimi-

Table 4.1 Effect of elimination of the initial conditions - Using simulated data.

	With elimination of the initial conditions		Without elimination of the initial conditions	
	a_1	a_2	a_1	a_2
Initial values	-0.4000	1.6000	-0.4000	1.6000
Converged values	-0.5000	2.0000	-0.5000	2.0000
True values	-0.5000	2.0000	-0.5000	2.0000
Number of iterations	4		4	
CPU time on a PDP-10 computer	9.7 sec		24.1 sec	

mation of the initial conditions required less iterations (3 versus 4) and took less computer time (9.7 sec versus 24.1 sec). The difference in computer time would be far greater if much more batch data were used.

Two computer programs were written and listed in Appendix E as Program II and Program III. The former was used to simulate data for purposes of identification. The latter was used to identify the model from the simulated data and by utilizing the Superposition Principle and the Walling-Lawton-Sylvestre idea. In these two programs, to do the integration, a fourth order Runge-Kutta technique with an integration interval of .01 unit is employed.

Example 2

From Chapter III, it was found that the specific growth rate $\mu(t)$ of A. pullulans 2552 in the batch process described in Appendix B is independent of the biomass concentration $c(t)$. Hence, the state space model representing the microbial growth is linear:

$$\dot{x}^j(t) = \left[a_1 + a_2 u^j_{1m}(t) + a_3 u^j_{2m}(t)\right] x^j(t) \qquad (4.30)$$

$$y^j(t) = x^j(t) \qquad (4.31)$$

where $u^j_{1m}(t)$, $u^j_{2m}(t)$ and $y^j(t)$ are, respectively, measured subtrate concentration, measured product concentration and model biomass concentration at time t in the $j^{\underline{th}}$ batch.

Based on the Superposition Principle and the Walling-Lawton-Sylvestre idea, a computer program, listed as Program IV in Appendix E, was written to identify the batch fermentation of A. pullulans. To do the integration, a fourth order Runge-Kutta technique is first used to generate the first 4 points of biomass concentrations, then the Adams-Bashforth predictor-corrector algorithm is used. The integration interval is one hour and the linear interpolation is employed to compute the values of substrate and product concentrations at times between measurements.

From the results, shown on Table 4.2, again, the case with elimination of the initial conditions required less iterations (3 versus 4) and took less computer time (56.6 sec versus 268.8 sec).

III ON-LINE IDENTIFICATION

By using the Superposition Principle and the Walling-Lawton-Sylvestre idea, the initial conditions are eliminated, and therefore the number of parameters to be estimated is reduced to a constant, independent of the number of batches from which the data are used. This facilitates the development of an approximate on-line identification algorithm to update the model as data from each new batch are being received. According to Astrom and Eyhoff (1971), on-line

Table 4.2 Effect of elimination of the initial conditions - Using experimental data.

	With elimination of the initial conditions			Without elimination of the initial conditions		
	a_1	a_2	a_3	a_1	a_2	a_3
Initial values	.0200	.0016	-.0066	.0200	.0016	-.0066
Converged values	.0243	.0014	-.0058	.0243	.0014	-.0058
Minimum sum of squares S	.289			.289		
Number of iterations	3			4		
CPU time on a PDP-10 computer	56.6 sec			268.8 sec		

identification algorithm, as opposed to off-line identification algorithm, is recursive and does not require the whole string of output-input data to be brought in at each step. It provides a convenient way of arranging the computation which often results in reducing computer storage and computing time.

Suppose data from the first B batches are available and an estimate of the process parameter vector $\underline{a}$ is $\hat{\underline{a}}_B$. To derive an on-line identification algorithm as data from the $L^{\underline{th}}$ batch are being received ($L \rightarrow B$), it is necessary to approximate $\underline{y}^j(\underline{a}, t_i^j)$ as its linearization around $\hat{\underline{a}}_B$. That is:

$$\underline{y}^j(\hat{\underline{a}}_B+\delta\underline{a}, t_i^j) = \underline{y}^j(\hat{\underline{a}}_B, t_i^j) + \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right]\delta\underline{a} \qquad (4.32)$$

Thus, the sum of squares of squares S becomes,

$$S = \frac{1}{2}\sum_{j=1}^{L}\sum_{i=0}^{I^j}\left\|\underline{y}_m^j(t_i^j) - \underline{y}^j(\hat{\underline{a}}_B, t_i^j) - \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right]\delta\underline{a}\right\|_{\underline{V}_i^j}^2 \qquad (4.33)$$

where,

$$\underline{a} = \hat{\underline{a}}_B + \delta\underline{a} \qquad (4.34)$$

Setting derivative of S defined in Eq. (4.33) with respect to $\underline{a}$ to zero gives:

$$\frac{\partial S}{\partial \delta \underline{a}^T} = \underline{0} \tag{4.35}$$

or,

$$\sum_{j=1}^{L} \sum_{i=0}^{I^j} \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right]^T \underline{V}_i^j \left\{ y_m^j(t_i^j) - \underline{y}^j(\hat{\underline{a}}_B, t_i^j) - \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right] \delta \hat{\underline{a}}_L \right\} = \underline{0} \tag{4.36}$$

or,

$$\delta \hat{\underline{a}}_L = - \underline{H}_L^{-1} \underline{g}_L \tag{4.37}$$

where,

$$\underline{H}_L = \sum_{j=1}^{L} \sum_{i=0}^{I^j} \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right]^T \underline{V}_i^j \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right] \tag{4.38}$$

$$\underline{g}_L = - \sum_{j=1}^{L} \sum_{i=0}^{I^j} \left[\frac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_i^j)}{\partial \hat{\underline{a}}_B^T}\right]^T \underline{V}_i^j \left[\underline{y}_m^j(t_i^j) - \underline{y}^j(\hat{\underline{a}}_B, t_i^j)\right] \tag{4.39}$$

Note that the subscript "L" and the cap "$\wedge$" were added to $\delta \underline{a}$ to identify that it is the optimum value from data of L batches.

Thus, the optimum estimates of $\underline{a}$ when (B+1), (B+2), ... L, (L+1), ... batches are used are as follows:

$$\hat{\underline{a}}_{B+1} = \hat{\underline{a}}_B + \delta\hat{\underline{a}}_{B+1}$$

where $\delta\hat{\underline{a}}_{B+1} = - \underline{H}_{B+1}^{-1}\underline{g}_{B-1}$

$$\hat{\underline{a}}_{B+2} = \hat{\underline{a}}_B + \delta\hat{\underline{a}}_{B+2}$$

where $\delta\hat{\underline{a}}_{B+2} = - \underline{H}_{B+2}^{-1}\underline{g}_{B+2}$

$$\hat{\underline{a}}_L = \hat{\underline{a}}_B + \delta\hat{\underline{a}}_L$$

where $\delta\hat{\underline{a}}_L = - \underline{H}_L^{-1}\underline{g}_L$ (4.40)

$$\hat{\underline{a}}_{L+1} = \hat{\underline{a}}_B + \delta\hat{\underline{a}}_{L+1}$$

where $\delta\hat{\underline{a}}_{L+1} = - \underline{H}_{L+1}^{-1}\underline{g}_{L+1}$ (4.41)

Let

$$\underline{Q}_j = \begin{bmatrix} \dfrac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_o^j)}{\partial \hat{\underline{a}}_B^T} \\ \hline \dfrac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_1^j)}{\partial \hat{\underline{a}}_B^T} \\ \hline \\ \hline \dfrac{\partial \underline{y}^j(\hat{\underline{a}}_B, t_I^j)}{\partial \hat{\underline{a}}_B^T} \end{bmatrix} \qquad (4.42)$$

$$\underline{e}_j = \begin{bmatrix} \underline{y}_m^j(t_o^j) - \underline{y}^j(\hat{\underline{a}}_B, t_o^j) \\ \hline \underline{y}_m^j(t_1^j) - \underline{y}^j(\hat{\underline{a}}_B, t_1^j) \\ \hline \\ \hline \underline{y}_m^j(t_I^j) - \underline{y}^j(\hat{\underline{a}}_B, t_I^j) \end{bmatrix} \qquad (4.43)$$

and

$$\underline{W}_j = \begin{bmatrix} \underline{V}_o^j & & & \\ & \underline{V}_1^j & & \\ & & \ddots & \\ & & & \underline{V}_{I_j}^j \end{bmatrix} \tag{4.44}$$

then $\underline{H}_L$ and $\underline{g}_L$ defined by Eqs. (4.38) and (4.39) can be put into the following forms:

$$\underline{H}_L = \sum_{j=1}^{L} \underline{Q}_j^T \underline{W}_j \underline{Q}_j \tag{4.45}$$

$$\underline{g}_L = - \sum_{j=1}^{L} \underline{Q}_j^T \underline{W}_j \underline{e}_j \tag{4.46}$$

Let

$$\underline{P}_{L+1} = \underline{H}_{L+1}^{-1} \tag{4.47}$$

then, from Eq. (4.45),

$$\underline{H}_{L+1} = \underline{H}_L + \underline{Q}_{L+1}^T \underline{W}_{L+1} \underline{Q}_{L+1}$$

or,

$$\underline{P}_{L+1} = (\underline{P}_L^{-1} + \underline{Q}_{L+1}^T \underline{W}_{L+1} \underline{Q}_{L+1})^{-1} \tag{4.48}$$

From the Matrix Inversion Lemma (Sage and Melsa 1971a), if $\underline{A}$, $(\underline{A} + \underline{B}\underline{C})$ and $(\underline{I} + \underline{C}\underline{A}^{-1}\underline{B})$ are nonsingular, then,

$$(\underline{A} + \underline{B}\underline{C})^{-1} = \underline{A}^{-1} - \underline{A}^{-1}\underline{B}(\underline{I} + \underline{C}\underline{A}^{-1}\underline{B})^{-1}\underline{C}\underline{A}^{-1} \qquad (4.49)$$

Applying this lemma to Eq. (4.48) with $\underline{A} = \underline{P}_L^{-1}$, $\underline{B} = \underline{Q}_{L+1}^T$ and $\underline{C} = \underline{W}_{L+1}\underline{Q}_{L+1}$ results in:

$$\underline{P}_{L+1} = \underline{P}_L - \underline{P}_L\underline{Q}_{L+1}^T(\underline{I} + \underline{W}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)^{-1}\underline{W}_{L+1}\underline{Q}_{L+1}\underline{P}_L \qquad (4.50)$$

or,

$$\underline{P}_{L+1} = \underline{P}_L - \underline{P}_L\underline{Q}_{L+1}^T(\underline{W}_{L+1}^{-1} + \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)^{-1}\underline{Q}_{L+1}\underline{P}_L \qquad (4.51)$$

or,

$$\underline{P}_{L+1} = (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\underline{P}_L \qquad (4.52)$$

where,

$$\underline{K}_{L+1} = \underline{P}_L\underline{Q}_{L+1}^T(\underline{W}_{L+1}^{-1} + \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)^{-1} \qquad (4.53)$$

From Eq. (4.46), one has:

$$\underline{g}_{L+1} = \underline{g}_L - \underline{Q}_{L+1}^T\underline{W}_{L+1}\underline{e}_{L+1} \qquad (4.54)$$

Substitution of Eqs. (4.54), (4.47), (4.52) and (4.53) into Eq. (4.41) results in:

$$\delta\hat{\underline{a}}_{L+1} = -\underline{P}_{L+1}\underline{g}_{L+1}$$

$$= -(\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\underline{P}_L(\underline{g}_L - \underline{Q}_{L+1}^T\underline{W}_{L+1}\underline{e}_{L+1})$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})(-\underline{P}_L\underline{q}_L + \underline{P}_L\underline{Q}_{L+1}^T\underline{W}_{L+1}\underline{e}_{L+1})$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})(\delta\hat{\underline{a}}_L + \underline{P}_L\underline{Q}_{L+1}^T\underline{W}_{L+1}\underline{e}_{L+1})$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\delta\hat{\underline{a}}_L + (\underline{P}_L\underline{Q}_{L+1}^T - \underline{K}_{L+1}\underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)$$

$$\underline{W}_{L+1}\underline{e}_{L+1}$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\delta\hat{\underline{a}}_L + [\underline{K}_{L+1}(\underline{W}_{L+1}^{-1} + \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)$$

$$- \underline{K}_{L+1}\underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T]\underline{W}_{L+1}\underline{e}_{L+1}$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\delta\hat{\underline{a}}_L + \underline{K}_{L+1}(\underline{W}_{L+1}^{-1} + \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T$$

$$- \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)\underline{W}_{L+1}\underline{e}_{L+1}$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\delta\hat{\underline{a}}_L + \underline{K}_{L+1}\underline{W}_{L+1}^{-1}\underline{W}_{L+1}\underline{e}_{L+1}$$

$$= (\underline{I} - \underline{K}_{L+1}\underline{Q}_{L+1})\delta\hat{\underline{a}}_L + \underline{K}_{K+1}\underline{e}_{L+1} \qquad (4.55)$$

or,

$$\delta\hat{\underline{a}}_{L+1} = \delta\hat{\underline{a}}_L + \underline{K}_{L+1}(\underline{e}_{L+1} - \underline{Q}_{L+1}\delta\hat{\underline{a}}_L) \qquad (4.56)$$

Eq. (4.56) coupled with Eqs. (4.52) and (4.53) result in an on-line identification algorithm to compute $\delta\hat{\underline{a}}_{L+1}$. This algorithm is only approximate since only the linear parts of of $\underline{y}^j(\underline{a}, t_i^j)$ for all j and i around $\hat{\underline{a}}_B$ were taken into account and therefore may not be valid when $\hat{\underline{a}}_B$ is a poor estimate or when L is much greater than B. Thus, modification was needed and done here by linearizing $\underline{y}^j(\underline{a}, t_i^j)$ around the most recent

estimate of $\underline{a}$. In Eqs. (4.50), (4.52) and (4.53), $\underline{a}_1$ was set to $\underline{0}$ and $\underline{e}_{L+1}$ and $\underline{Q}_{L+1}$ were evaluated at $\hat{\underline{a}}_1$,

$$\delta\hat{\underline{a}}_{L+1} = \underline{h}_{L+1}\underline{e}_{L+1} \tag{4.57}$$

$$\underline{h}_{L+1} = \underline{P}_L\underline{Q}_{L+1}^T(\underline{W}_{L+1}^{-1} + \underline{Q}_{L+1}\underline{P}_L\underline{Q}_{L+1}^T)^{-1} \tag{4.58}$$

$$\underline{P}_{L+1} = (\underline{I} - \underline{h}_{L+1}\underline{Q}_{L+1})\underline{P}_L \tag{4.59}$$

$$\hat{\underline{a}}_{L+1} = \grave{\underline{a}}_L + \delta\hat{\underline{a}}_{L+1} \tag{4.60}$$

$\underline{e}_{L+1}$ and $\underline{Q}_{L+1}$ are still defined from Eqs. (4.42) and (4.43) but with $\grave{\underline{a}}_L$ in place of $\hat{\underline{a}}_B$.

To compute $\hat{\underline{a}}_{L+1}$, the on-line identification algorithm needs to store only $\grave{\underline{a}}_L$, $\underline{P}_L$ and data from the new batch (the $(L+1)^{th}$ batch). Figure 4.3 is a flow diagram showing how the model is updated after data from each new batch has been received.

III.1 On-Line Identification Algorithm

The following is an on-line identification algorithm to compute $\hat{\underline{a}}_{L+1}$ and $\underline{P}_{L+1}$ when $\hat{\underline{a}}_L$, $\underline{P}_L$ and data from the $(L+1)^{th}$ batch are given.

1. Compute $\Phi^{L+1}(\hat{\underline{a}}_L, t_i^{L+1})$ and $\Psi^{L+1}(\hat{\underline{a}}_L, t_i^{L+1})$ for $i = 0, 1, \ldots, I$ by solving the differential equations, Eqs. (4.12) and (4.13). A numerical integration technique, say Runge-Kutta, can be used. Linear interpolation can be employed to calculate

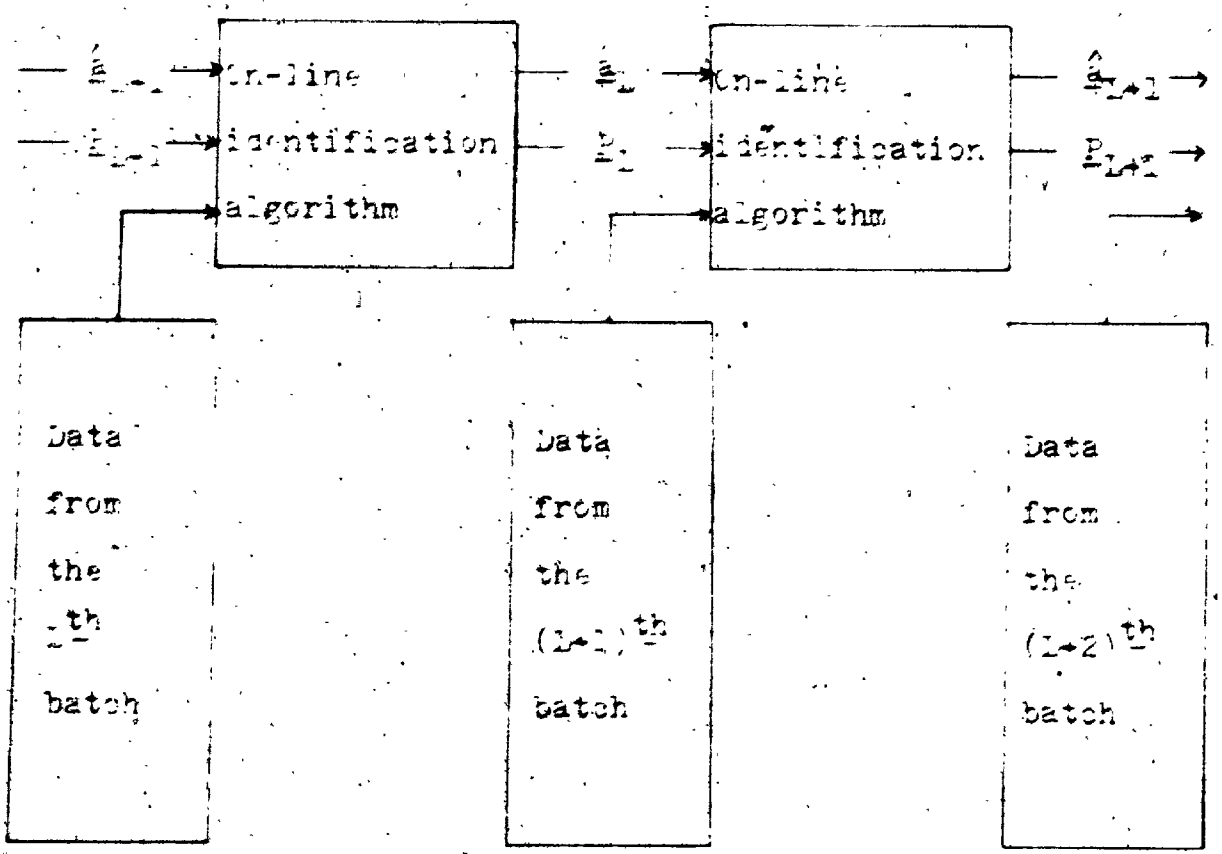


Figure 4.3 Flow diagram showing how the model is updated as data from each new batch are being received.

the values of $\underline{u}_m^{L+1}$ at times between measurements.

2. Compute $\underline{x}^{L+1}(\hat{\underline{a}}_L, t_o^{L+1})$ from Eqs. (4.17), (4.18) and (4.19).

3. Compute $\underline{y}^{L+1}(\hat{\underline{a}}_L, t_i^{L+1})$ for $i = 0,1,\ldots,I^{L+1}$ from Eq. (4.21).

4. Compute $\partial \underline{y}^{L+1}(\hat{\underline{a}}_L, t_i^{L+1})/\partial \underline{a}_L^T$ for $i = 0,1,\ldots,I^{L+1}$. The finite difference approximation mentioned in Section II.1 of Chapter III can be used.

5. Form $\underline{g}_{L+1}$, $\underline{c}_{L+1}$ and $\underline{W}_{L+1}$ based on Eqs. (4.42), (4.43) and (4.44) with $\hat{\underline{a}}_B$ replaced by $\hat{\underline{a}}_L$.

6. Compute $\hat{\underline{a}}_{L+1}$ from Eqs. (4.60), (4.57) and (4.58).

7. Compute $\underline{P}_{L+1}$ from Eq. (4.59), required for computing $\hat{\underline{a}}_{L+2}$.

It should be noted that new data required for the model to be updated do not have to be from a whole batch as shown in Figure 4.3, they can just be from a part of a batch as shown in either Figure 4.4 or Figure 4.5 where the subscripts "old" and "new" stand for before and after updating respectively. Indeed, by following the same reasoning and derivation, it can be easily shown that:

$$\hat{\underline{a}}_{new} = \hat{\underline{a}}_{old} + \underline{h}\underline{e} \tag{4.61}$$

$$\underline{h} = \underline{P}_{old}\underline{S}^T(\underline{W}^{-1} + \underline{S}\underline{P}_{old}\underline{S}^T)^{-1} \tag{4.62}$$

$$\underline{P}_{new} = (\underline{I} - \underline{h}\underline{S})\underline{P}_{old} \tag{4.63}$$

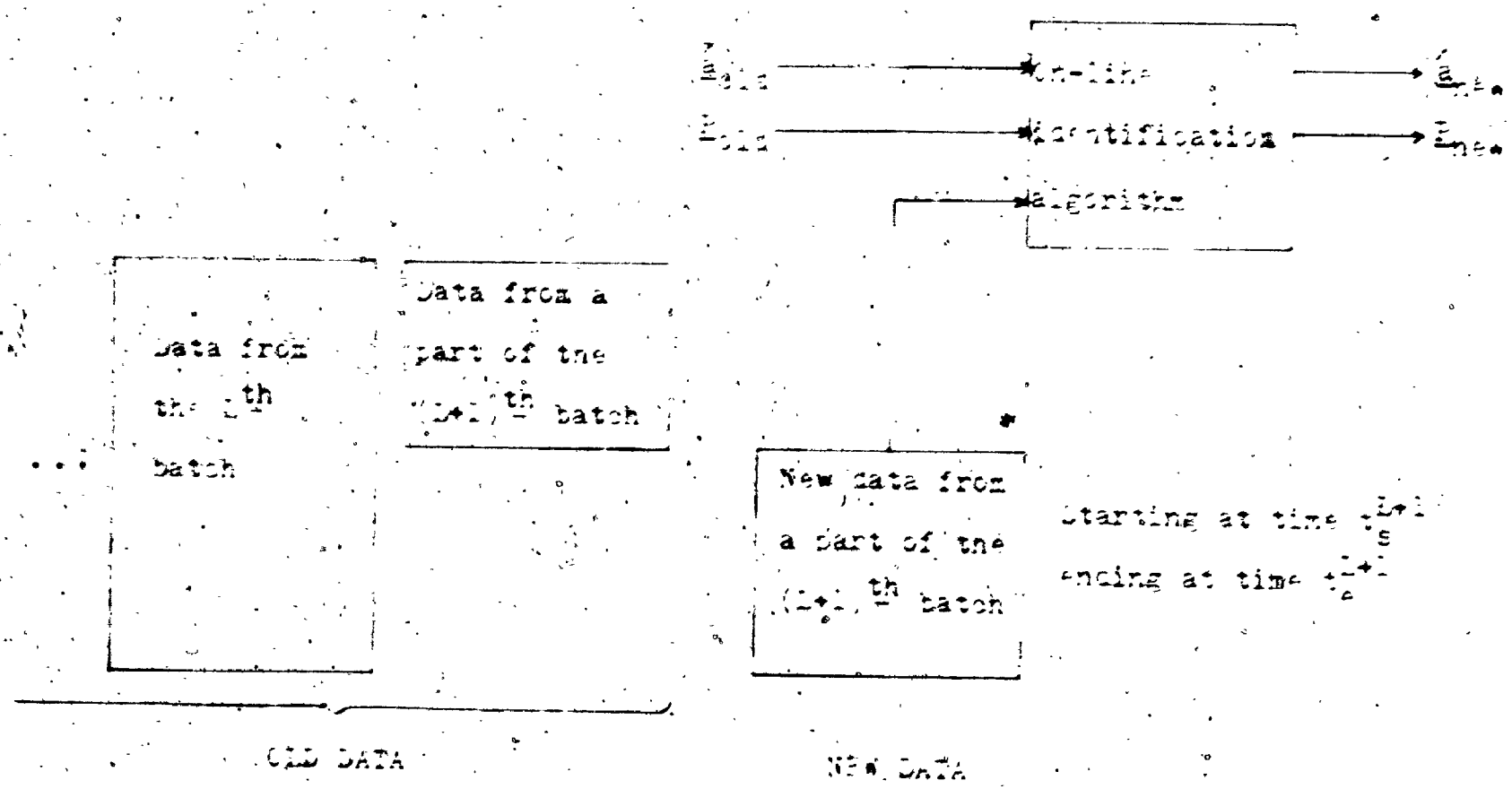


Figure 4.4 Diagram showing how the model is updated as new data from a part of the $(i+1)^{th}$ batch are being received.

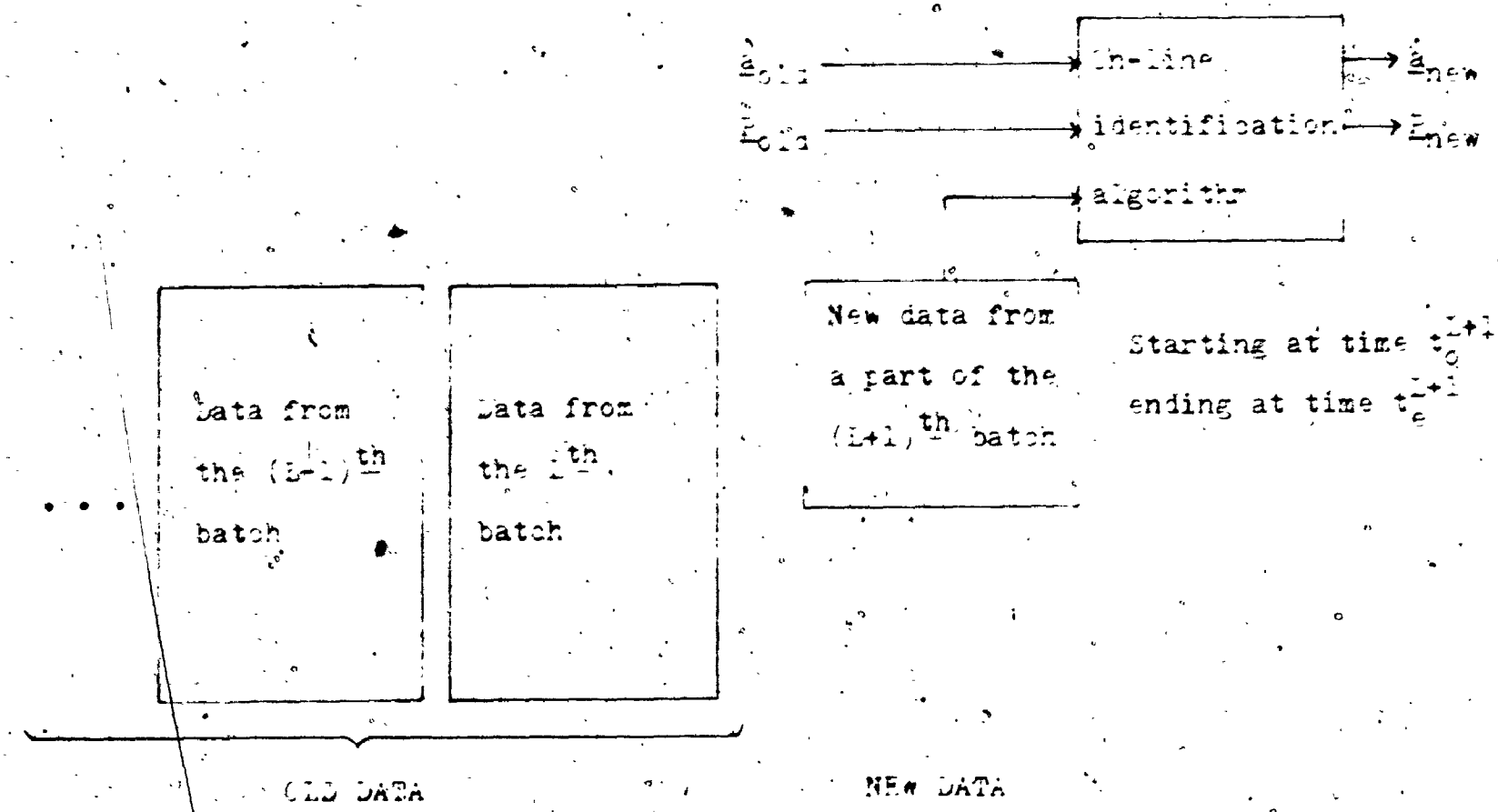


Figure 4.5 Diagram showing how the model is updated as new data from a part of a new batch are being received.

where $\underline{U}$, $\underline{e}$ and $\underline{W}$ are defined below, depending on whether new data from a part of the old batch or of the new batch.

New data from a part of the old batch

When the new data are from a part of the old batch, the $(L+1)^{th}$ batch, starting at time t_s^{L+1} and ending at time t_e^{L+1} then,

$$\underline{U} = \begin{bmatrix} \dfrac{\partial \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_s^{L+1})}{\partial \hat{\underline{a}}_{old}^T} \\ \hline \dfrac{\partial \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_{s+1}^{L+1})}{\partial \hat{\underline{a}}_{old}^T} \\ \hline \vdots \\ \hline \dfrac{\partial \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_e^{L+1})}{\partial \hat{\underline{a}}_{old}^T} \end{bmatrix} \qquad (4.64)$$

$$\underline{e} = \begin{bmatrix} \underline{y}_m^{L+1}(t_s^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_s^{L+1}) \\ \hline \underline{y}_m^{L+1}(t_{s+1}^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_{s+1}^{L+1}) \\ \hline \vdots \\ \hline \underline{y}_m^{L+1}(t_e^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_e^{L+1}) \end{bmatrix} \qquad (4.65)$$

and

$$\underline{W} = \begin{bmatrix} \underline{v}_s^{l+1} & & & \\ & \underline{v}_{s+1}^{l+1} & & \\ & & \ddots & \\ & & & \underline{v}_e^{l+1} \end{bmatrix} \qquad (4.66)$$

Now data from a part of a new batch

When the new data are from a part of a new batch, the $(l+1)^{th}$ batch, starting at time t_o^{l+1} and ending at time t_1^{l+1} then,

$$\underline{\alpha} = \begin{bmatrix} \dfrac{\partial \underline{Y}^{l+1}(\hat{\underline{a}}_{old}, t_o^{l+1})}{\partial \underline{a}_{old}^T} \\ \hline \dfrac{\partial \underline{Y}^{l+1}(\hat{\underline{a}}_{old}, t_1^{l+1})}{\partial \hat{\underline{a}}_{old}^T} \\ \hline \vdots \\ \hline \dfrac{\partial \underline{Y}^{l+1}(\hat{\underline{a}}_{old}, t_e^{l+1})}{\partial \hat{\underline{a}}_{old}^T} \end{bmatrix} \qquad (4.67)$$

$$\begin{bmatrix} \underline{y}_m^{L+1}(t_o^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_o^{L+1}) \\ \hline \underline{y}_m^{L+1}(t_1^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_1^{L+1}) \\ \hline \\ \\ \hline \underline{y}_m^{L+1}(t_e^{L+1}) - \underline{y}^{L+1}(\hat{\underline{a}}_{old}, t_e^{L+1}) \end{bmatrix} \qquad (4.68)$$

and

$$\underline{W} = \begin{bmatrix} \underline{v}_o^{L+1} & & & \\ & \underline{v}_1^{L+1} & & \\ & & \ddots & \\ & & & \underline{v}_e^{L+1} \end{bmatrix} \qquad (4.69)$$

III.2 Applications

To test the on-line identification algorithm, artificial data were generated from the model previously described in Example 1 of Section II.2 and with $\sigma = .2$ and used in the following two examples:

Example 1

The data from the first five batches were used to compute $\hat{\underline{a}}_5$ and $\underline{P}_5$. These were the initial conditions required for the on-line identification algorithm which started as data from the 6th batch were received. Table 4.3 are results from the on-line identification algorithm as compared to those from the off-line identification algorithm described in Section II.1. True values are used as the starting values for the off-line identification algorithm, that is $\underline{a}_o = [-.5 \quad 2.0]^T$.

From Table 4.3, the on-line identification algorithm took much less computer time (30 sec versus 426 sec), and produced estimates which are very close to the off-line estimates and would become even closer as more and more batch data are used.

Example 2

In the previous example, the initial conditions $\hat{\underline{a}}_5$ and $\underline{P}_5$ for the on-line identification algorithm were computed from the off-line identification algorithm and based on data from the first five batches.

$$\hat{\underline{a}}_5 = [-.50543 \quad 2.1444]^T$$

Table 4.3 On-line results as compared to off-line results from artificial data with standard deviation σ = .2 and process parameter vector $\underline{a} = [-.5 \quad 2.]^T$.

Number of batches	On-line results		Off-line results	
	a_1	a_2	a_1	a_2
5	-0.52204	2.5824	-0.52204	2.5824
6	-0.51345	2.3066	-0.51388	2.3055
7	-0.51303	2.2881	-0.51296	2.2872
8	-0.50936	2.2000	-0.50928	2.1991
9	-0.50002	2.0634	-0.49994	2.0628
10	-0.50159	2.1125	-0.50152	2.1120
11	-0.50034	2.0963	-0.50027	2.0958
12	-0.50113	2.0080	-0.50107	2.0076
13	-0.50183	2.0285	-0.50178	2.0281
14	-0.50103	2.0753	-0.50098	2.0750
15	-0.50053	2.0799	-0.50048	2.0797
16	-0.50029	2.1015	-0.50025	2.1011
17	-0.50030	2.1328	-0.50026	2.1325
18	-0.50135	2.1470	-0.50132	2.1466
19	-0.50476	2.1904	-0.50473	2.1901
20	-0.50510	2.1168	-0.50507	2.1166
21	-0.50296	2.1434	-0.50293	2.1431
22	-0.50275	2.1310	-0.50271	2.1307
23	-0.50267	2.1046	-0.50264	2.1043
24	-0.50253	2.1053	-0.50250	2.1050
25	-0.50230	2.1062	-0.50227	2.1059
26	-0.50261	2.1497	-0.50258	2.1495
27	-0.50307	2.1374	-0.50304	2.1372
28	-0.50239	2.2262	-0.50237	2.2260
29	-0.50301	2.2379	-0.50299	2.2377
30	-0.50312	2.2743	-0.50309	2.2741
CPU time on a PDP-10	30 sec		426 sec	

$$\underline{P}_5 = \begin{bmatrix} 3.8102 \times 10^{-3} & -5.0327 \times 10^{-2} \\ -5.0327 \times 10^{-2} & 3.5259 \times 10^{0} \end{bmatrix}$$

In this example, the off-line identification algorithm is supposed to be either unavailable or undesirable to be used. The initial conditions therefore must be supplied through guessing. Figures 4.6 and 4.7 are results from the on-line identification algorithm which started as data from the first batch were received for the following three cases corresponding to three different initial conditions.

Case 1

Starting value for $\underline{a} = \begin{bmatrix} -.75 & 3. \end{bmatrix}^T$

Starting value for $\underline{P} = \begin{bmatrix} 1. & 0. \\ 0. & 1. \end{bmatrix}$

Case 2

Starting value for $\underline{a} = \begin{bmatrix} .5 & -2. \end{bmatrix}^T$

Starting value for $\underline{P} = \begin{bmatrix} 1. & 0. \\ 0. & 1. \end{bmatrix}$

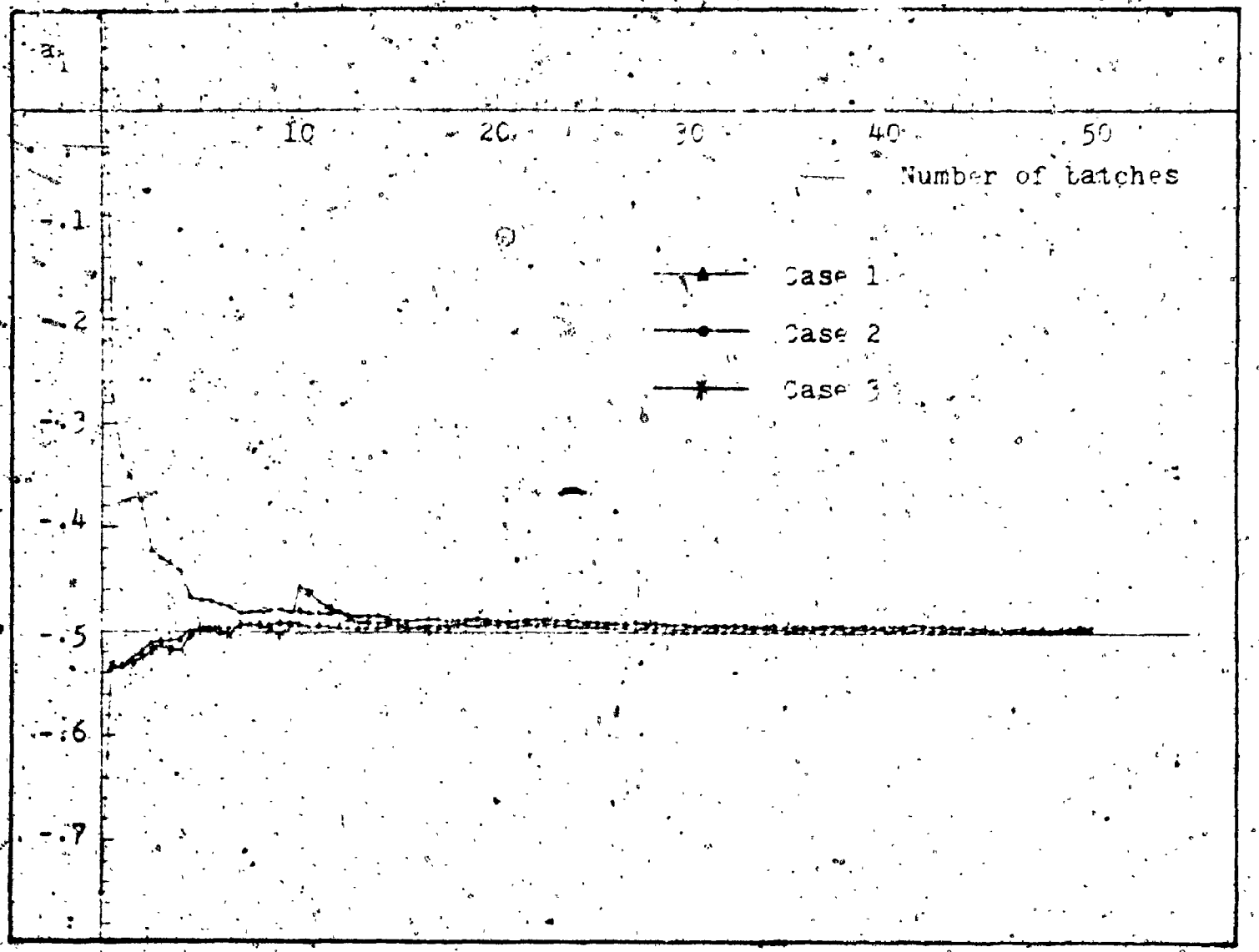


Figure 4.6 On-line results for a_1 from artificial data with standard deviation $\sigma = .2$ and process parameter vector $\underline{a} = [-.5 \quad 2.]^T$ for the three different cases.

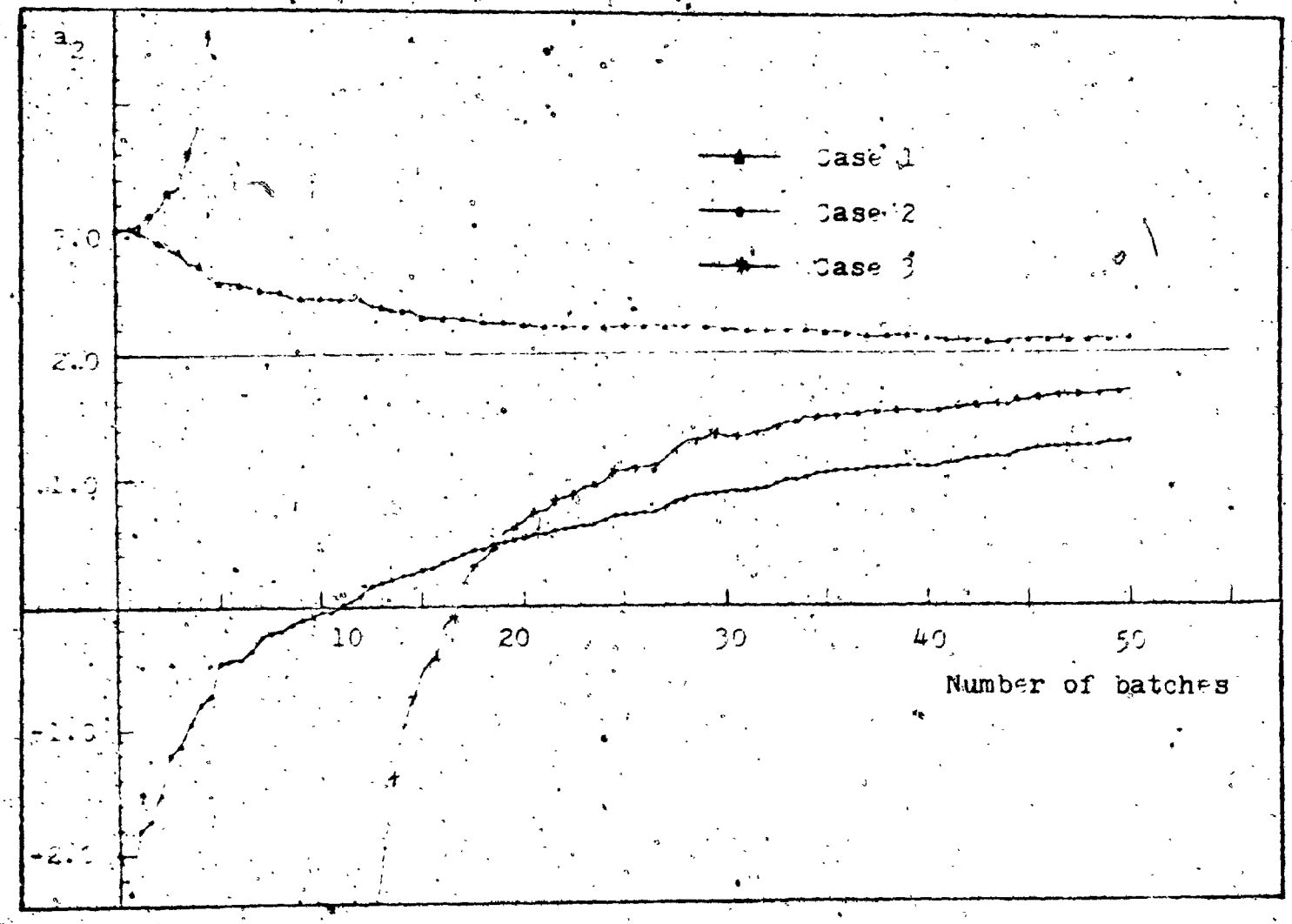


Figure 7.7 On-line results for a_2 from artificial data with standard deviation $\sigma = .2$ and process parameter vector $\underline{a} = [-.5 \quad 2.]^T$ for the three different cases.

Case 3

Starting value for $\underline{a}$ = $[-.75 \quad 3.]^T$

Starting value for $\underline{P}$ = $\begin{bmatrix} -1. & 0. \\ 0. & -1. \end{bmatrix}$

The starting value for $\underline{a}$ in both Case 1 and Case 3 is 50% above the true value (in absolute value) and that in Case 2 is exactly opposite to the true value which is a very poor guess. The starting value for $\underline{P}$ in both Case 1 and Case 2 is an identity matrix (positive definite) and that in Case 3 is an identity matrix with a negative sign (negative definite) which is a very poor and very unreasonable guess as directly from the definition in Eqs. (4.47) and (4.45), the matrix $\underline{P}$ is known to be positive definite.

As seen from the figures, for all three cases, the sequence of the on-line estimate of the parameter a_1 and that of the parameter a_2 both converged to their true values, though for Case 3 they oscillated at the beginning. The former converged faster than the latter.

The program used to compute the on-line estimates in both examples is listed in Appendix E as Program-V.

CHAPTER V

DEVELOPMENT OF A STOCHASTIC MODEL AND A REGULATING CONTROL POLICY FOR A CONTINUOUS FERMENTATION PROCESS

I INTRODUCTION

Disturbances and measurement errors exist in operating bioreactor. Therefore using a deterministic model to represent a continuous fermentation process may not be realistic and hence control policies based on the the deterministic model may not work satisfactorily.

Based on this fact, Svroek et al (1974) formulated a stochastic model and derived a control policy to regulate effluent concentrations. The stochastic model is discrete and was formed by linearising and discretising the deterministic Monod model and then adding disturbance-terms which were assumed to be white and Gaussian noise sequences with zero-means and known variances. The control policy which they named "Kalman control" policy was composed of a Kalman filter cascaded with a proportional controller.

In this chapter, a continuous stochastic model is proposed to take into account measurement errors and disturbances which are due to imperfect mixing of substrate feeding tank and of bioreactor. Other disturbances such as those

due to imperfect control of pH and temperature are assumed to be negligible. Based on the model and the Separation Theorem, a control policy to regulate effluent concentrations is derived, using dilution rate as a control input.

II MATHEMATICAL MODEL

Figure 5.1 is a schematic of a continuous fermentation process where V, F, c, s, c_e, s_e and s_I are volume of medium in the bioreactor, flowrate, biomass and substrate concentrations in the bioreactor, effluent biomass and substrate concentrations, and influent substrate concentration, respectively. Mass balances across the bioreactor result in the following equations for biomass concentration and substrate concentration:

$$\dot{c}(t) = -D(t)c_e(t) + \left[\mu(t) - k\right]c(t) \qquad (5.1)$$

$$\dot{s}(t) = D(t)\left[s_I(t) - s_e(t)\right] - \mu(t)c(t)/Y \qquad (5.2)$$

where the supercript "." stands for derivative with respect to time t, $D(t) = F(t)/V$ is dilution rate, k is metabolism coefficient or specific death rate, Y is the yield coefficient of biomass based on substrate and $\mu(t)$ is specific growth rate which is assumed to follow the Monod equation, Equation (5.3),

$$\mu(t) = \frac{\mu_m s(t)}{s(t) + k_s} \qquad (5.3)$$

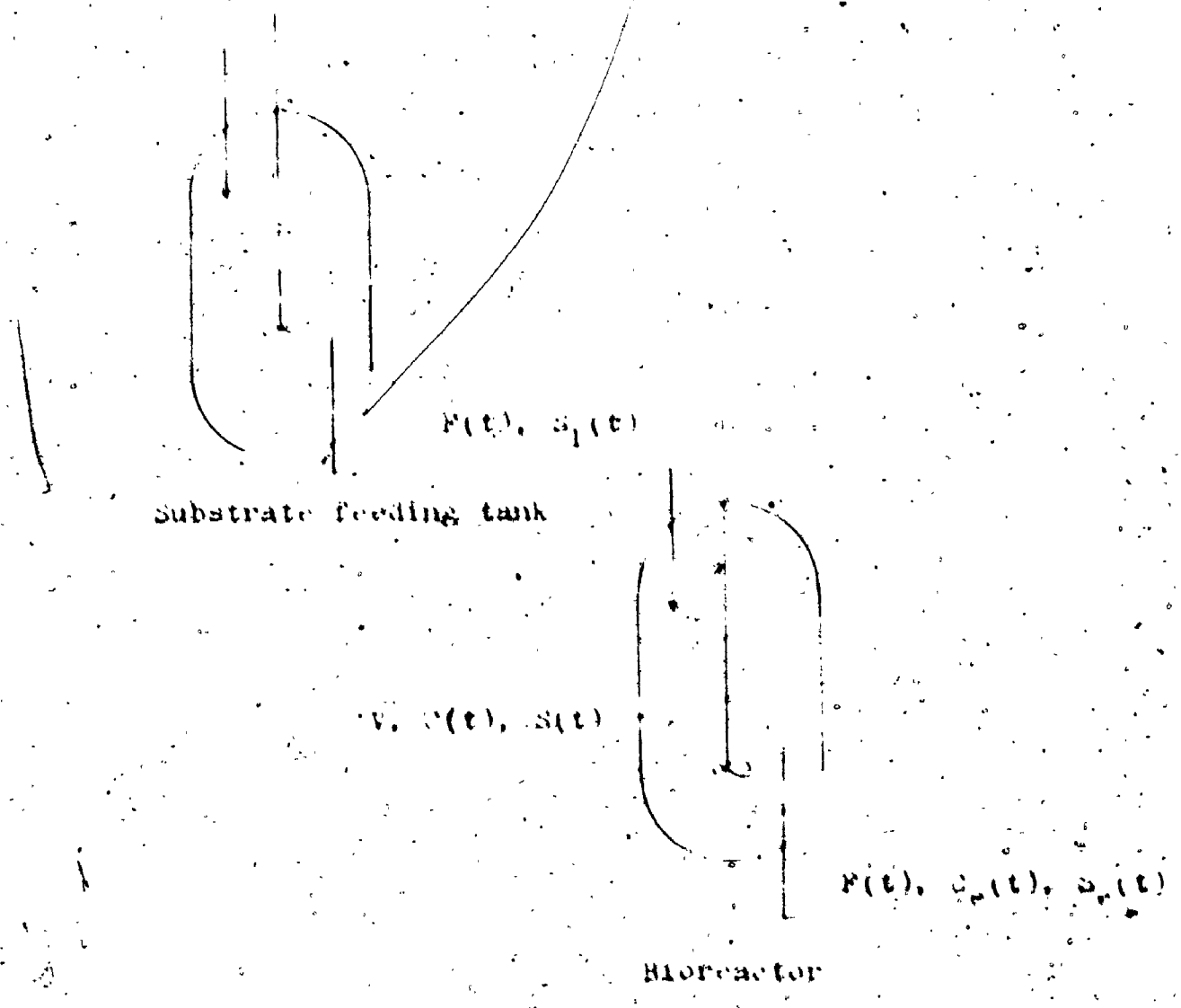


Figure 2.1 Schematic of a continuous fermentation process.

values of the parameters k_s, Y, γ_m and K_s often depend on the operating medium temperature, pH and rate of oxygen supplied (in aerobic fermentation).

due to imperfect mixing of the substrate feeding tank and of the bioreactor, C_e and S_e are not the same as C and S, and S_1 is not a constant $\bar{S}_1$, but rather they are related to each other by the following equations:

$$C_e(t) = C(t) + w_c(t) \tag{5.4}$$

$$S_e(t) = S(t) + w_{s1}(t) \tag{5.5}$$

$$S_1(t) = \bar{S}_1 + w_{s2}(t) \tag{5.6}$$

where w_c, w_{s1} and w_{s2} represent disturbances due to imperfect mixing. These disturbances are zero in statistical average. The degrees of fluctuation of these disturbances depend on the degree of mixing and on the nature of biomass and substrate involved, are proportional to concentrations C, S and $\bar{S}_1$. Mathematically they can be approximated as:

$$w_c(t) = k_c C(t) w_c'(t) \tag{5.7}$$

$$w_{s1}(t) = k_{s1} S(t) w_{s1}'(t) \tag{5.8}$$

$$w_{s2}(t) = k_{s2} \bar{S}_1 w_{s2}'(t) \tag{5.9}$$

where k_c, k_{s1} and k_{s2} are some appropriate constants depending on the degree of mixing and the nature of biomass and substrate; w_c', w_{s1}' and w_{s2}' are white, Gaussian noises with

zero-means and unity-power spectra and are independent of each other. Note that perfect mixing is a special case of imperfect mixing where the values for k_c, k_{s1} and k_{s2} are all zero, and that higher values for k_c, k_{s1} and k_{s2} correspond to poorer mixing.

Measurements on the effluent concentrations, c_m and s_m are assumed available and related to c_e and s_e as:

$$c_m(t) = c_e(t) + v_c(t) \qquad (5.10)$$

$$s_m(t) = s_e(t) + v_s(t) \qquad (5.11)$$

where v_c and v_s are measurement noise (errors), and as normal practice, can be assumed to be white and to have Gaussian distributions with zero-means and constant power spectra of Γ_1 and Γ_2.

Thus Eqs. (5.1) to (5.11) represent a stochastic model for a continuous fermentation process with measurement errors and disturbances due to imperfect mixing.

III REGULATING CONTROL PROBLEM

For a continuous process, the purpose of control is often to design a regulator (a controller which maintains the products as uniform as possible). In a continuous fermentation process, this means keeping the effluent concentrations as close to some desired values as possible. If the dilution

rate is used as a control input, then mathematically the following stochastic optimum control problem must be solved:

$$\underset{D(t)}{\text{Min.}}\ J' = E\left\{\frac{1}{2}\int_{t_o}^{t_f}\left\{k_1\left[C_e(t) - \bar{C}\right]^2 + k_2\left[S_e(t) - \bar{S}\right]^2 + k_3\left[D(t) - \bar{D}\right]^2\right\}dt\right\} \qquad (5.12)$$

$$t_o \le t \le t_f$$

subject to Equations (5.1) to (5.11).

where E is the expectation (statistical average) operator; t_o and t_f are the starting time and the ending time of the operation; k_1, k_2 and k_3 are some positive weighting constants; $\bar{C}$ and $\bar{S}$ are the regulating levels of the effluent biomass and substrate concentrations; and $\bar{D}$ is some average level at which D(t) should operate. A reasonable value that can be chosen for $\bar{D}$ is the dilution rate which gives the highest rate of biomass production in a chemostat. The values chosen for $\bar{C}$ and $\bar{S}$ are the corresponding steady state biomass and substrate concentrations (see Chapter II).

The solution to the problem can be formulated using the Stochastic Maximum Principle (Sage and Melsa 1971b) or the Dynamic Programming (Bellman 1957). However due to complexity of the differential equations formed by Equations (5.1) to (5.11), the solution can not be solved exactly. An approximate solution of the problem will be computed here

using the Separation Theorem.

IV THE SEPARATION THEOREM

The Separation Theorem or Separation Principle originated from the Certainty Equivalence Principle which first appeared in the field of econometrics (Simon 1956, Theil 1957). For the discrete case, it was first introduced by Joseph and Tou (1961) and Gunckel and Franklin (1963). And for the continuous case, a rigorous proof was first given by Wonham (1968). A more complete history of the Separation Theorem and its variations can be found in a paper written by Speyer and Gustafson (1974). A proof of the Separation Theorem, at engineering standard (not rigorous), is given in Appendix D.

The Separation Theorem, verbally, states that, for Linear-Quadratic-Gaussian problems, that is for linear systems with quadratic cost functions and subjected to additive white Gaussian noise inputs, the optimum stochastic controller (stochastic regulator) is realized by cascading an optimum state estimator (Kalman filter) with a deterministic optimum controller (deterministic regulator). Note that the Kalman filter is an algorithm to estimate the states of the linear systems (see Appendix D for more details) and the deterministic optimum controller is an algorithm to

compute the optimum control policy when the linear systems are not subjected to noise inputs. Graphically, the Separation Theorem can be described as in Figure 5.2. Mathematically, the theorem states that, for a given linear plant system,

$$\underline{\dot{x}}(t) = \underline{F}_1(t)\underline{x}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{F}_3(t)\underline{w}(t) \qquad (5.13)$$

with a linear observation system,

$$\underline{y}(t) = \underline{G}(t)\underline{x}(t) + \underline{v}(t) \qquad (5.14)$$

where $\underline{u}(t)$, $\underline{x}(t)$, $\underline{y}(t)$, $\underline{w}(t)$ and $\underline{v}(t)$ are respectively, control vector, state vector, measurement vector, process noise vector and measurement noise vector, having the following properties:

$\underline{w}(t)$ and $\underline{v}(t)$ $t_o \leq t \leq t_f$ are white Gaussian stochastic processes and $\underline{x}(t_o)$ is a Gaussian random vector.

$$E\{\underline{w}(t)\} = \underline{0}$$

$$E\{\underline{v}(t)\} = \underline{0}$$

$$E\{\underline{x}(t_o)\} = \underline{\mu}_{x_o}$$

$$E\left\{\begin{bmatrix}\underline{w}(t)\\ \hline \underline{v}(t)\end{bmatrix}\left[\underline{w}^T(\tau) \;\vdots\; \underline{v}^T(\tau)\right]\right\} = \left[\begin{array}{c:c}\underline{R}_2(t) & \underline{T}(t)\\ \hline \underline{T}^T(t) & \underline{R}_2(t)\end{array}\right]\delta_D(t-\tau)$$

$$E\left\{\left[\underline{x}(t_o) - \underline{\mu}_{x_o}\right]\left[\underline{x}(t_o) - \underline{\mu}_{x_o}\right]^T\right\} = \underline{P}_o$$

$$E\left\{\underline{x}(t_o)\underline{w}^T(t)\right\} = \underline{0}$$

$$E\left\{\underline{x}(t_o)\underline{v}^T(t)\right\} = \underline{0}$$

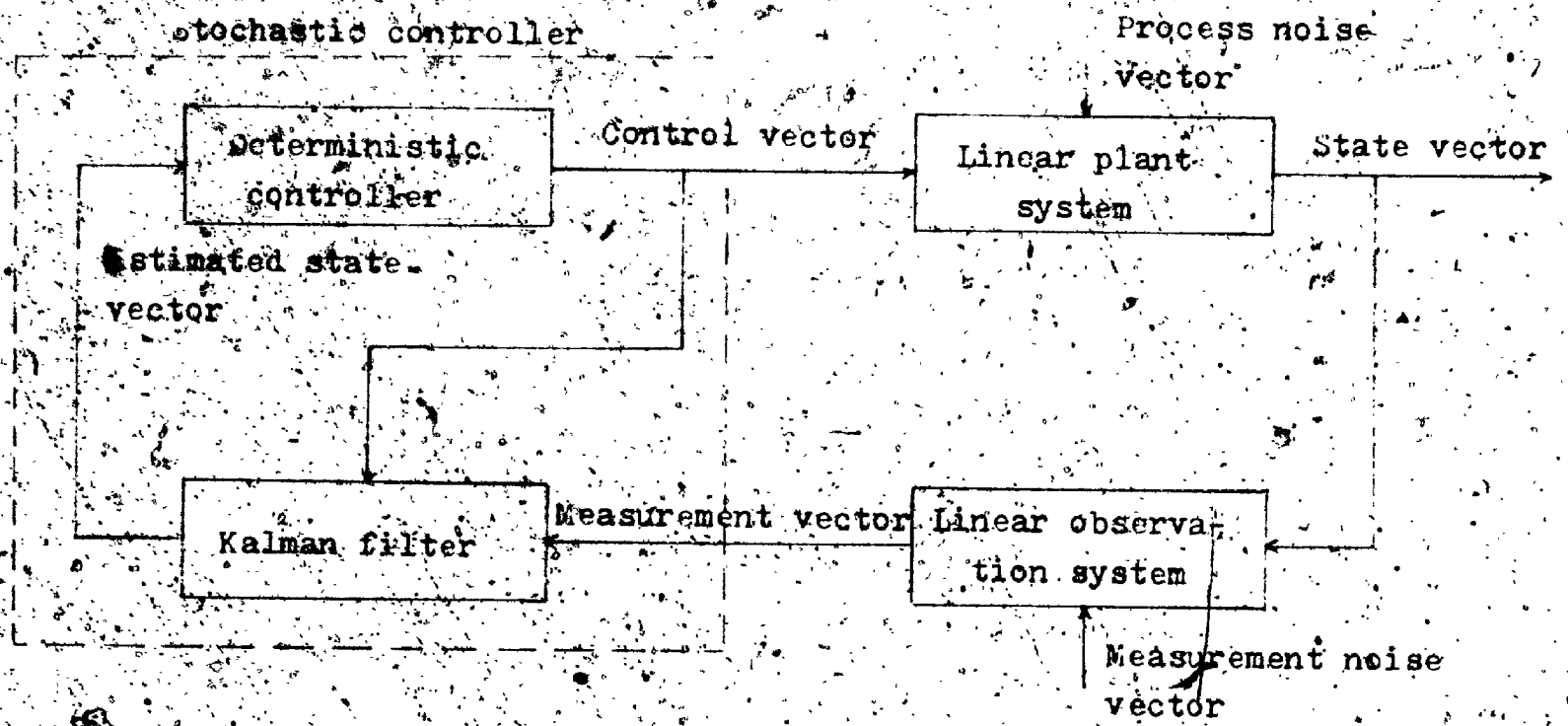


Figure 5.2 Graphical representation of the Separation Theorem.

where $\underline{\delta}_D(t)$ denotes the unit impulse function (Dirac delta function), $\underline{F}_1(t)$, $\underline{F}_2(t)$, $\underline{F}_3(t)$, $\underline{G}(t)$, $\underline{Q}_2(t)$, $\underline{R}_2(t)$ and $\underline{T}(t)$ are known matrix functions, and $\underline{P}_o$ and $\underline{\gamma}_{x_o}$ are known matrix and vector.

Then the optimum feedback control vector $\underline{u}(t)$ as a functional of $\underline{y}(\tau)$ $t_o \leq \tau \leq t$, which minimizes the quadratic function J defined by Equation (5.15), is given by Equations (5.16) to (5.21),

$$J = \tfrac{1}{2}E\left\{\underline{x}^T(t_f)\underline{P}_f\underline{x}(t_f) + \int_{t_o}^{t_f}\left[\underline{x}^T(t) \;\vdots\; \underline{u}^T(t)\right]\left[\begin{array}{c|c}\underline{Q}_1(t) & \underline{N}(t)\\ \hline \underline{N}^T(t) & \underline{R}_1(t)\end{array}\right]\left[\begin{array}{c}\underline{x}(t)\\ \hline \underline{u}(t)\end{array}\right]dt\right\} \qquad (5.15)$$

where $\underline{P}_f$, $\underline{Q}_1(t)$, $\underline{R}_1(t)$ and $\left[\begin{array}{c|c}\underline{Q}_1(t) & \underline{N}(t)\\ \hline \underline{N}^T(t) & \underline{R}_1(t)\end{array}\right]$ are known symmetric, positive definite matrices.

$$\underline{u}(t) = -\underline{K}_1(t)\hat{\underline{x}}(t) \qquad (5.16)$$

$$K_1(t) = -\underline{R}_1^{-1}(t)\left[\underline{F}_2^T(t)\underline{P}_1(t) + \underline{N}^T(t)\right] \qquad (5.17)$$

= Kalman control gain matrix

$$\underline{\dot{P}}_1(t) = -\underline{P}_1(t)\underline{F}_1(t) - \underline{F}_1^T(t)\underline{P}_1(t) + \underline{K}_1^T(t)\underline{R}_1(t)\underline{K}_1(t) - \underline{Q}_1(t) \qquad (5.18)$$

Final condition: $\underline{P}_1(t_f) = \underline{P}_f$

$$\dot{\hat{\underline{x}}}(t) = \underline{F}_1(t)\hat{\underline{x}}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{K}_2(t)\left[\underline{y}(t) - \underline{G}(t)\hat{\underline{x}}(t)\right] \tag{5.19}$$

Initial condition: $\hat{\underline{x}}(t_o) = \underline{\mu}_{x_o}$

$$\underline{K}_2(t) = \left[\underline{F}_2(t)\underline{G}^T(t) + \underline{F}_3(t)\underline{T}(t)\right]\underline{R}_2^{-1}(t) \tag{5.20}$$

= Kalman filter gain matrix

$$\dot{\underline{P}}_2(t) = \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) - \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \tag{5.21}$$

Initial condition: $\underline{P}_2(t_o) = \underline{P}_o$

Equations (5.16) to (5.18) form the deterministic optimum controller and Equations (5.19) to (5.21) the optimum state estimator (Kalman filter).

V APPROXIMATION

The system equations, Equations (5.1) to (5.11), and the performance index described by Equation (5.12) do not really fit into the Linear-Quadratic-Gaussian form represented by Equations (5.13) to (5.15), therefore an approximation is needed. Let

$$\underline{x} = \begin{bmatrix} C & S \end{bmatrix}^T$$

$$\underline{w} = \begin{bmatrix} w_c & w_{s1} & w_{s2} \end{bmatrix}^T$$

then Equations (5.1) to (5.9) can be put into the following compact form:

$$\dot{\underline{x}} = \underline{F}(\underline{x}, D) + \underline{F}(\underline{x}, U)\underline{w} \tag{5.22}$$

Suppose $\bar{D}$ is chosen as the dilution rate which gives the highest rate of biomass production in a chemostat at steady state, and $\underline{\bar{x}}' = [\bar{C} \quad \bar{S}]^T$ is chosen as the corresponding steady state effluent concentration vector, that is,

$$\underline{f}'(\underline{\bar{x}}', \bar{D}) = \underline{0} \tag{5.23}$$

Then if $\underline{f}'(\underline{x}', D)$ is replaced by its linear terms around $\underline{\bar{x}}'$, and $\bar{D}$, and $\underline{F}'(\underline{x}', D)$ by $\underline{F}'(\underline{\bar{x}}', \bar{D})$, Equation (5.22) becomes the following linear differential equation:

$$\dot{\underline{x}}(t) = \underline{F}_1 \underline{x}(t) + \underline{f}_2 u(t) + \underline{w}(t) \tag{5.24}$$

where

$$\underline{x}(t) = \underline{x}'(t) - \underline{\bar{x}}'$$

$$u(t) = D(t) - \bar{D}$$

$$\underline{w}(t) = \underline{F}'(\underline{\bar{x}}', \bar{D})\underline{w}$$

$$E\{\underline{w}(t)\} = \underline{0}$$

$$E\{\underline{w}(t)\underline{w}^T(\tau)\} = \underline{Q}_2(t)\delta_D(t-\tau)$$

$$= \begin{bmatrix} (k_c\bar{C})^2 & 0 \\ 0 & (k_{s1}\bar{S})^2 + (k_{s2}\bar{S}_f)^2 \end{bmatrix} \delta_D(t-\tau) \tag{5.25}$$

$$\underline{F}_1 = \begin{bmatrix} 0 & \dfrac{\mu_m K_s \bar{C}}{(\bar{S}+K_s)^2} \\ \dfrac{\mu_m \bar{S}}{Y(\bar{S}+K_s)} & -\bar{D} - \dfrac{\mu_m k_c \bar{C}}{Y(\bar{S}+K_s)^2} \end{bmatrix} \tag{5.26}$$

$$\underline{f}_2 = \left[-\bar{c} \quad (\bar{s}_I - \bar{s})\right]^T \tag{5.27}$$

Similarly, Equations (5.10) and (5.11) can be approximated as:

$$\underline{y}(t) = G\underline{x}(t) + \underline{v}(t) \tag{5.28}$$

where,

$$\underline{y}(t) = \left[\, c_m(t) - \bar{c} \quad s_m(t) - \bar{s} \,\right]^T$$

$$\underline{v}(t) = \begin{bmatrix} k_c\bar{c} & 0 \\ 0 & k_{s1}\bar{s} \end{bmatrix} \begin{bmatrix} w_c'(t) \\ w_{s1}'(t) \end{bmatrix} + \begin{bmatrix} v_c(t) \\ v_s(t) \end{bmatrix}$$

$$E\left\{\underline{v}(t)\right\} = 0$$

$$E\left\{\underline{v}(t)\underline{v}^T(\tau)\right\} = \underline{R}_2(t)\delta_D(t-\tau)$$

$$= \begin{bmatrix} (k_c\bar{c})^2 + \sigma_1^2 & 0 \\ 0 & (k_{s1}\bar{s})^2 + \sigma_2^2 \end{bmatrix} \delta_D(t-\tau) \tag{5.29}$$

$$E\left\{\underline{w}(t)\underline{v}^T(\tau)\right\} = \underline{T}(t)\delta_D(t-\tau)$$

$$= \bar{D} \begin{bmatrix} -(k_c\bar{c})^2 & 0 \\ 0 & -(k_{s1}\bar{s})^2 \end{bmatrix} \delta_D(t-\tau) \tag{5.30}$$

$$\underline{Q} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix} \qquad (5.31)$$

Finally, if C_e and S_e are approximated by $C(t) + k_c \bar{w}_c$ and $S + k_{s1} \bar{w}_{s1}$ respectively, then minimizing J' defined by equation (5.13) with respect to $D(t)$ is equivalent to minimizing J with respect to $u(t)$, where,

$$J = E\left\{\frac{1}{2}\int_{t_o}^{t_f} (k_1 x_1^2 + k_2 x_2^2 + k_3 u^2)\,dt\right\} \qquad (5.32)$$

Comparing Equation (5.32) to Equation (5.13),

$$\underline{F}_f = \underline{0} \qquad (5.33)$$

$$\underline{N}(t) = \underline{0} \qquad (5.34)$$

$$\underline{M}_1(t) = \begin{bmatrix} k_1 & & 0 \\ & \ddots & \\ 0 & & k_2 \end{bmatrix} \qquad (5.34),$$

$$\underline{R}_1(t) = k_3 \qquad (5.36)$$

VI. ALGORITHM

The following is an algorithm to regulate the effluent concentrations using the Separation Theorem.

1. Compute the dilution rate $\bar{D}$ which gives the highest rate of biomass production at steady state and when distur-

bances are ignored by the method outlined in chapter II.

. . . compute the steady state values $\bar{S}$ and $\bar{S}$ for the effluent concentrations when disturbances are ignored and at the dilution rate $\bar{D}$ by solving

$$\underline{f}'(\bar{\underline{x}}; \bar{D}) = \underline{0}$$

or,

$$\bar{X} = \frac{k_s(\bar{D} + k_d)}{\mu_m - (\bar{D} + k_d)}$$

$$\bar{S} = \frac{Y\bar{D}(S_1 - \bar{S})}{\bar{D} + k_d}$$

. Read in $\underline{x}_{A_0}$, $\underline{P}_0$, $\underline{V}_1$ and $\underline{V}_2$.

. Compute the matrices $\underline{F}_1$, $\underline{f}_c$, $\underline{H}$, $\underline{M}_1$, $\underline{M}_2$, $\underline{A}_1$, $\underline{K}_2$, $\underline{\Gamma}$, $\underline{N}$ and $\underline{i}_p$ from Equations (5.26), (5.27), (5.31), (5.35), (5.25), (5.26), (5.29), (5.30), (5.34) and (5.33) respectively.

. Compute and store the kalman control gain matrix $\underline{k}_1(t)$ and the kalman filter gain matrix $\underline{k}_2(t)$ for $t_0 \leq t \leq t_f$ from Eqs. (5.17) and (5.20) by solving the two matrix-Riccatti differential equations (5.18) and (5.21) with the former backward in time and the latter forward in time. As a digital computer is often used as a controller, only values of $\underline{k}_1(t)$ and $\underline{k}_2(t)$ at appropriate discrete times in the interval (t_0, t_f) are computed and stored. When the operating time is long, that is t_f is large, it is impractical to compute and to store all values of the gain matrices. However, kalman (1960a) has

shown that if

- The system is completely controllable and observable.
- $\underline{F}_0 \neq \underline{0}$ and $\underline{E}_0 = \underline{0}$.
- All the coefficients in the matrix-Riccati differential equations are time-invariant, that is $\underline{E}_1$, $\underline{E}_2$, $\underline{F}_1$, $\underline{F}_2$, $\underline{E}_3$, $\underline{F}_3$, $\underline{E}_4$, $\underline{F}_4$ and $\underline{I}$ are constant-matrices,

then,

$$\underline{E}_1(t) = \underline{E}_1^{\infty} = \text{constant matrix}$$
$$\underline{F}_1(t) = \underline{F}_1^{\infty} = \text{constant matrix} \qquad \text{as } t_f \to \infty$$

Thus $\underline{E}_1^{\infty}$ and $\underline{F}_1^{\infty}$ can be used in place of $\underline{E}_1(t)$ and $\underline{F}_1(t)$ for all t.

5. Compute the Kalman filter estimate $\hat{\underline{x}}(t)$ and the control policy $\underline{u}(t)$ from equations (5.18) and (5.19). Note that the dilution rate required is $D(t) = u(t) + \bar{D}$. When a digital computer is used as a controller, data and results are given only at discrete times t_i for $i = 0, 1, \ldots$ and one needs a continuous control input $u(t)$. Therefore the control input $u(t)$ for $t_i \leq t < t_{i+1}$ must be supplied and clearly that the optimum choice is:

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

For some fermentation processes, effluent concentrations cannot be measured instantly by sensors. There may be time delays before their measured values are known. If this was the case, then the best way to estimate the state

vector $x(t)$ would be to use the Kalman predictor (Gage and Kelsey [6-7]) in place of the Kalman filter; that is, equation (5.10) is replaced by equation (5.17),

$$y(t) = h_1(t)\hat{x}(t, t-\Delta t) \tag{5.18}$$

where $\hat{x}(t, t-\Delta t)$ is the Kalman predictor of $x(t)$ or the conditional expectation of $x(t)$ given data up to time $(t-\Delta t)$ and is computed as (Gage and Kelsey [6-7])

$$\hat{x}(t, t-\Delta t) = \Phi(t, t-\Delta t)\hat{x}(t-\Delta t) + \int_{t-\Delta t}^{t} \Phi(t, \tau) F(\tau) u(\tau)\, d\tau \tag{5.19}$$

where Δt is the delay time and $\Phi(t, \tau)$ is the state transition matrix associated with equation (5.1) and can be computed from the following equation:

$$\dot{\Phi}(t, \tau) = F_1(t)\Phi(t, \tau)$$
$$\Phi(\tau, \tau) = I = \text{identity matrix} \tag{5.20}$$

Note that solving $\hat{x}(t, t-\Delta t)$ from equations (5.19) and (5.20) is equivalent to solving equation (5.1) with the noise term ignored, that is,

$$\dot{x}(t) = F_1(t)x(t) + F_2(t)u(t) \tag{5.21}$$

for $x(t)$, where,

$x(t) = \hat{x}(t, t-\Delta t)$

$x(t-\Delta t)$ is given as $\hat{x}(t-\Delta t)$ and

$u(t)$ is known for $t-\Delta t \le t \le t$

VII [illegible]

To study the effect and limitation of applying the separation theorem to regulate effluent concentrations, a continuous fermentation process with the following parameters is simulated.

S_i = [illegible]

μ_m = [illegible] hr^{-1}

k_s = [illegible] g/l

k_i = [illegible] hr^{-1}

Y = [illegible] unitless.

These parameter values correspond to those computed from the Monod-Andrews model for a continuous fermentation of A. aerogenes at the temperature [illegible] °K. The dilution rate D, which gives the highest rate of biomass production at steady state when disturbances are ignored, has been computed in chapter VI as:

D = [illegible] hr^{-1}

and the corresponding steady state effluent concentrations are:

S_e = [illegible] g/l

x_e = [illegible] g/l

Suppose the substrate feed tank and the bioreactor are imperfectly mixed so that

k_1 = [illegible]

k_2 = [illegible]

k_3 = [illegible]

Then the results averaged from 20 simulated runs are shown on Figures 5.3 to 5.11. It should be noted that due to imperfect mixing, the process is stochastic and the results from one runs differ from another run even under the same operating conditions. This is why average results are shown.

In general, it is desirable to have the control weighting factor $k_3|x(t)-\bar{x}|^2$ equal to zero (by setting $k_3 = 0$) or as small as possible when compared to the state weighting factors $k_1|c_e(t)-\bar{c}|^2$ and $k_2|s_e(t)-\bar{s}|^2$ so that c_e and s_e will be as close to $\bar{c}$ and $\bar{s}$ as possible. However when the control weighting factor is too small (or is zero), the control dilution rate varies widely (or becomes unbounded) as can be seen from Eqs. (5.16), (5.17) and (5.36) and therefore the linearized model described by Eqs. (5.36) and (5.37) is no longer valid, that is, poor results occur. Figures 5.3 and 5.4 are plots of controlled effluent concentrations for various values of k_3. Better regulation corresponds to lower value of k_3. Nevertheless, the control breaks down for any value of k_3 lower than 5 for the above reason.

Figures 5.5 and 5.6 are plots of controlled effluent concentrations for several different levels of measurement noise, when $k_3 = 1.5$. The Kalman filter estimates the state vector $x(t)$ poorly and therefore the control breaks down.

Figure 5.7 is an optimum dilution rate profile. Figures 5.8 and 5.9 show plots of the kalman control gain matrix $\underline{K}_1(t)$ and the kalman filter gain matrix $\underline{K}_2(t)$ respectively. These matrices "reach" their steady state values quite quickly as would be expected.

Figures 5.10 and 5.11 are plots of controlled effluent concentrations for the case where there is a time delay of 0.5 hour in measurement. For a delay time of one hour or more the state vector $\underline{x}(t)$ is estimated (predicted) poorly and therefore the control breaks down.

In summary, the simulation study shows that, though the Separation Theorem requires a linear system with a quadratic cost function, results from the linearized system with approximate quadratic cost function are quite acceptable as long as:

- the control weighting factor is not set too small as compared to the state weighting factors,
- The variances of measurement noise are not too large,
- The delay time in measurement is not too long (For the case with measurement delay).

Programs VI and VII listed in Appendix E have been used to simulate all the results. The former employs the kalman filter and the latter employs the kalman predictor for the case with measurement delay.

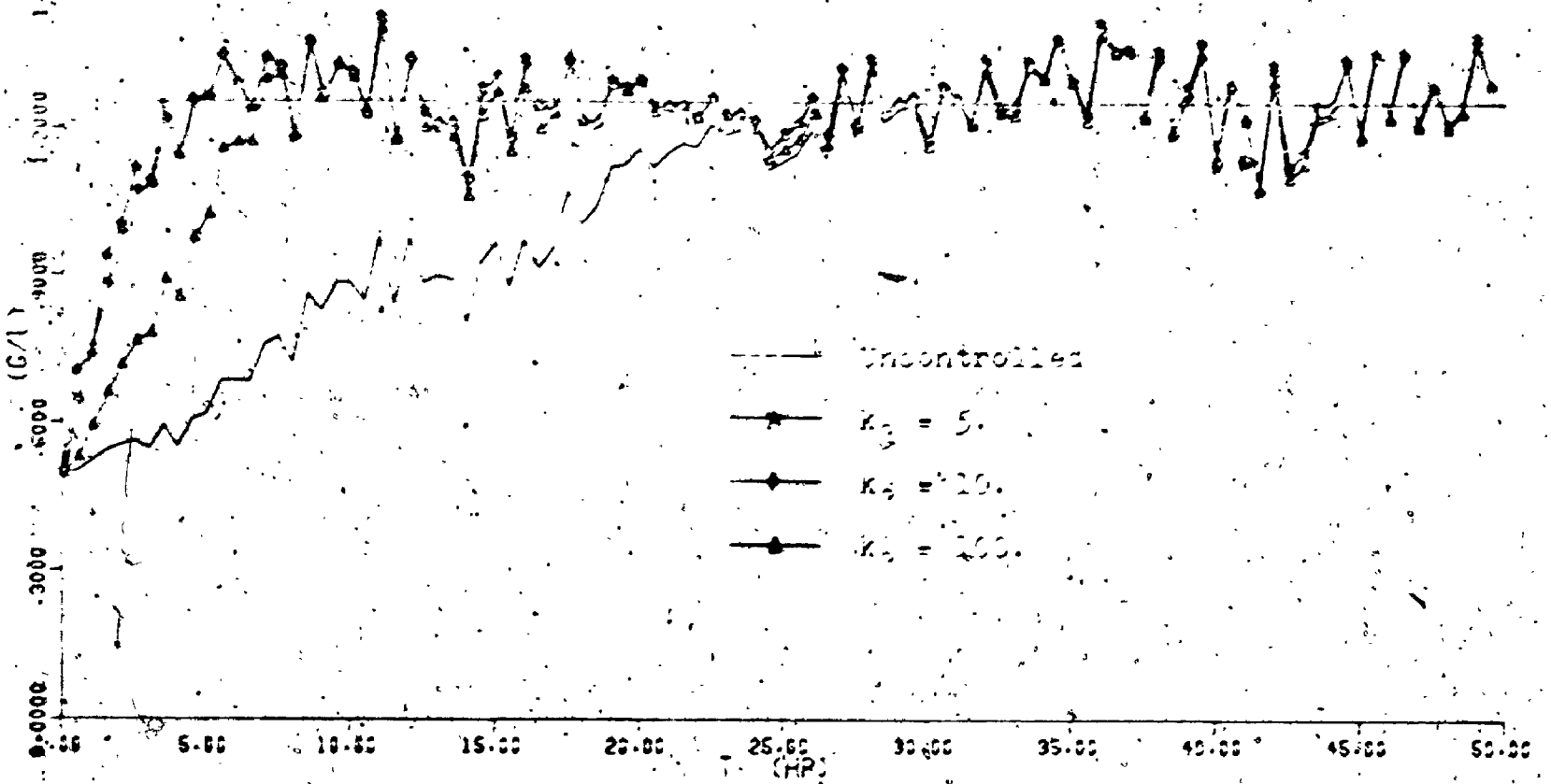


Figure 5.2. Controlled effluent biomass concentrations for various values of control weighting constant K_2, K_1 = 1., K_2 = 10., T_1 = T_2 = .02.

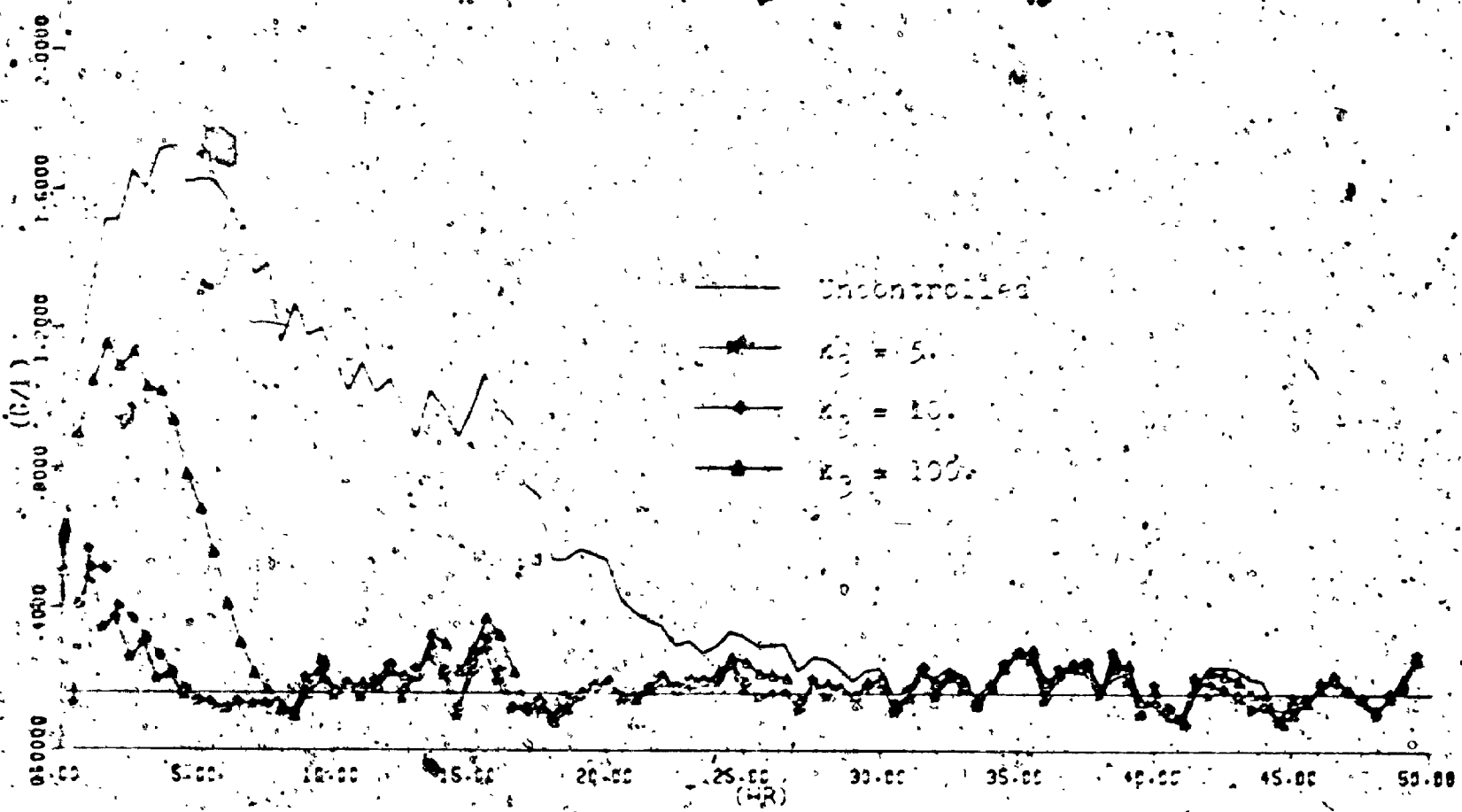


Figure 5.4 Controlled effluent substrate concentrations for various values of control weighting constant k_2, $k_1 = 1.$, $k_2 = 10.$, $\tau_1 = \tau_2 = .02$.

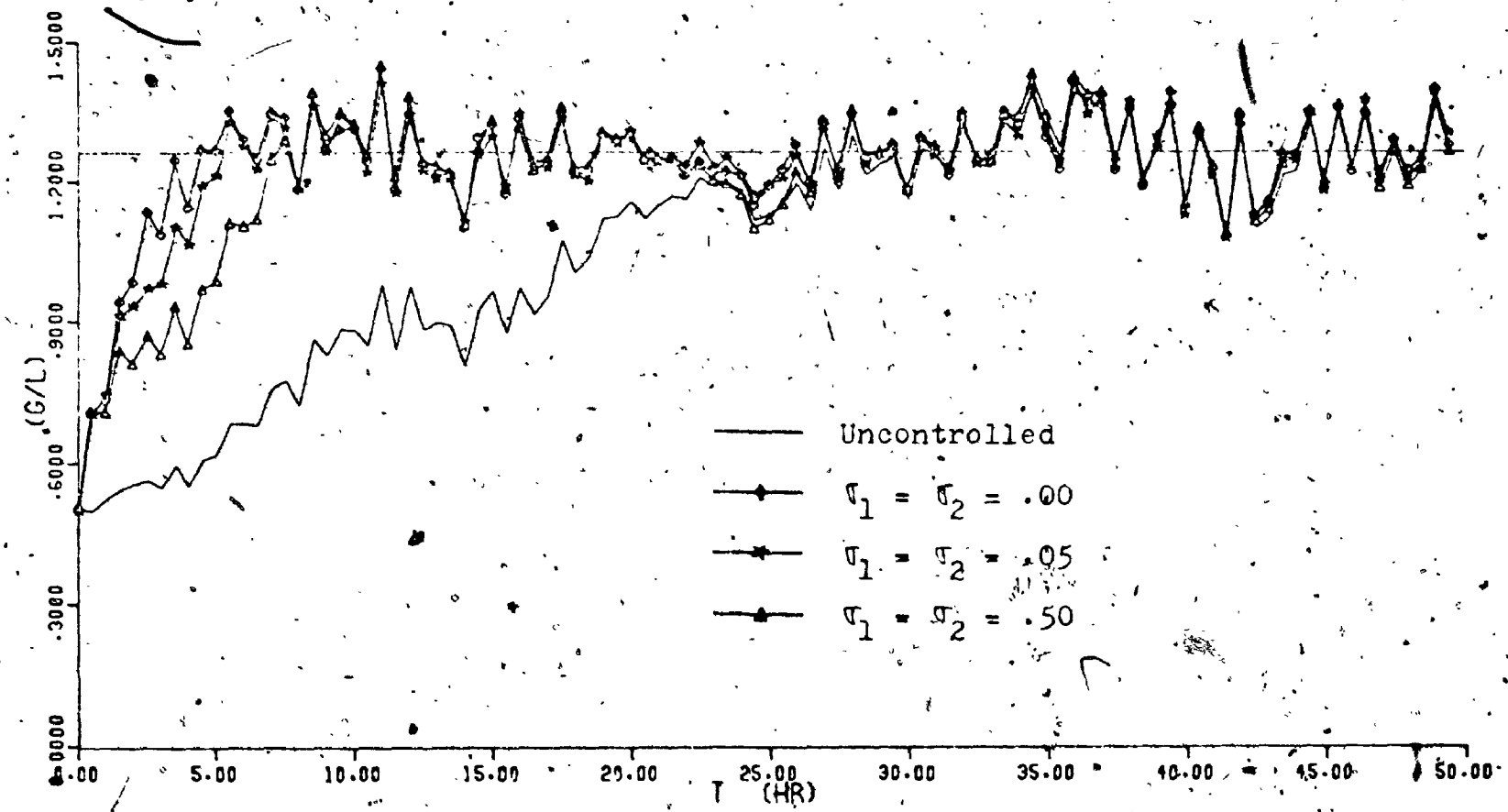


Figure 5.5 Controlled effluent biomass concentrations for various levels of measurement noise, $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$.

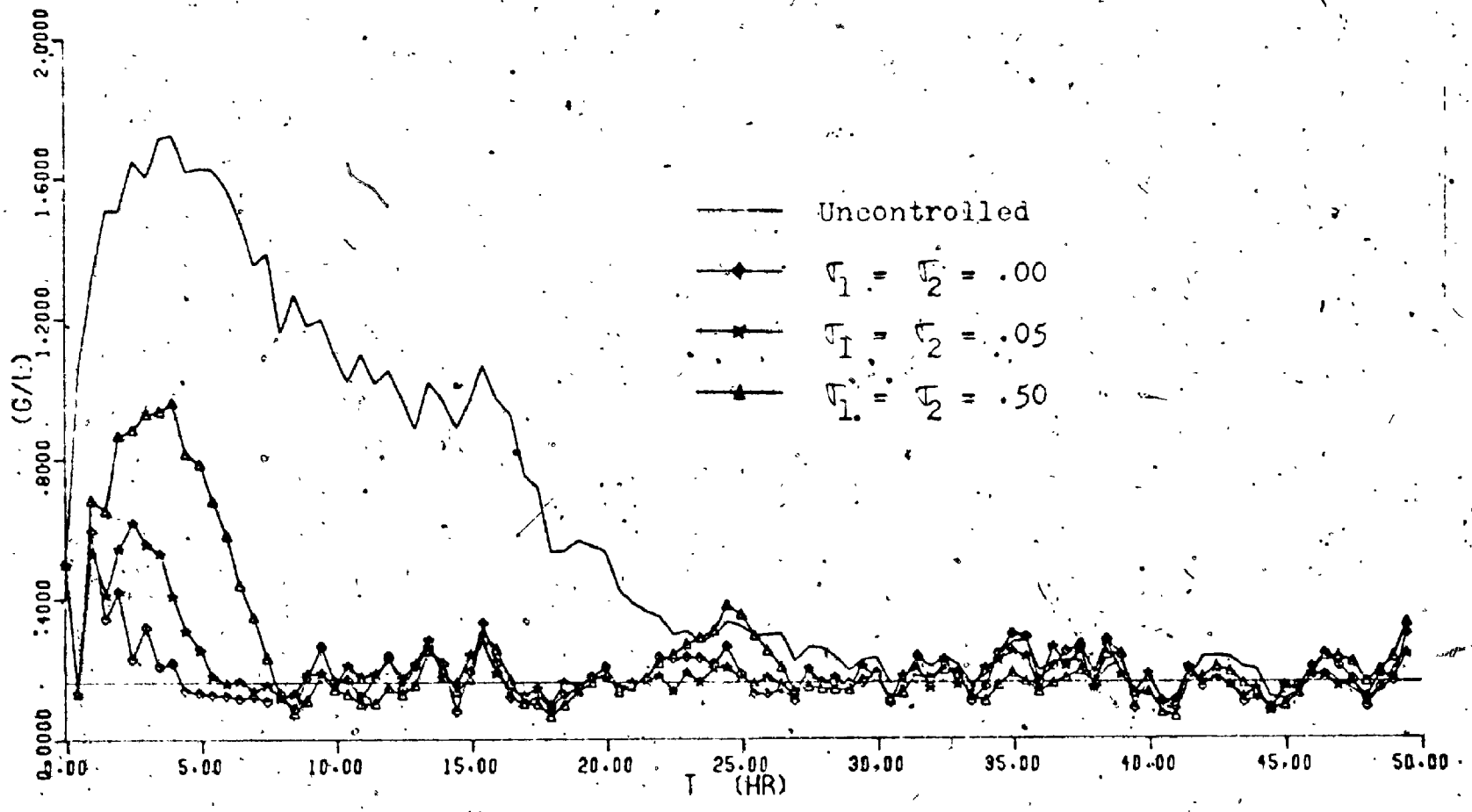


Figure 5.6 Controlled effluent substrate concentrations for various levels of measurement noise, $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$.

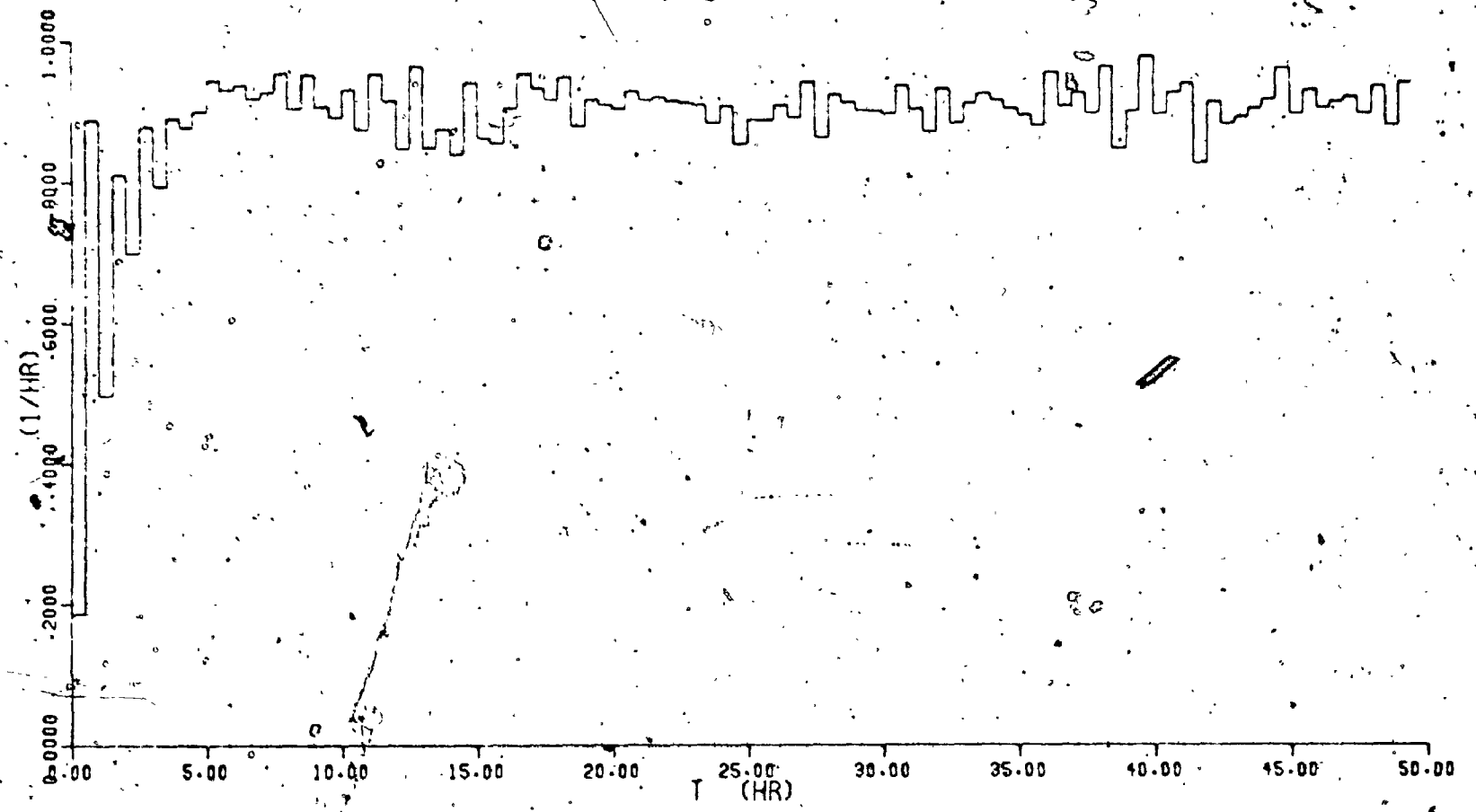


Figure 5.7 Optimum dilution rate profile for $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$, $\sigma_1 = \sigma_2 = .02$.

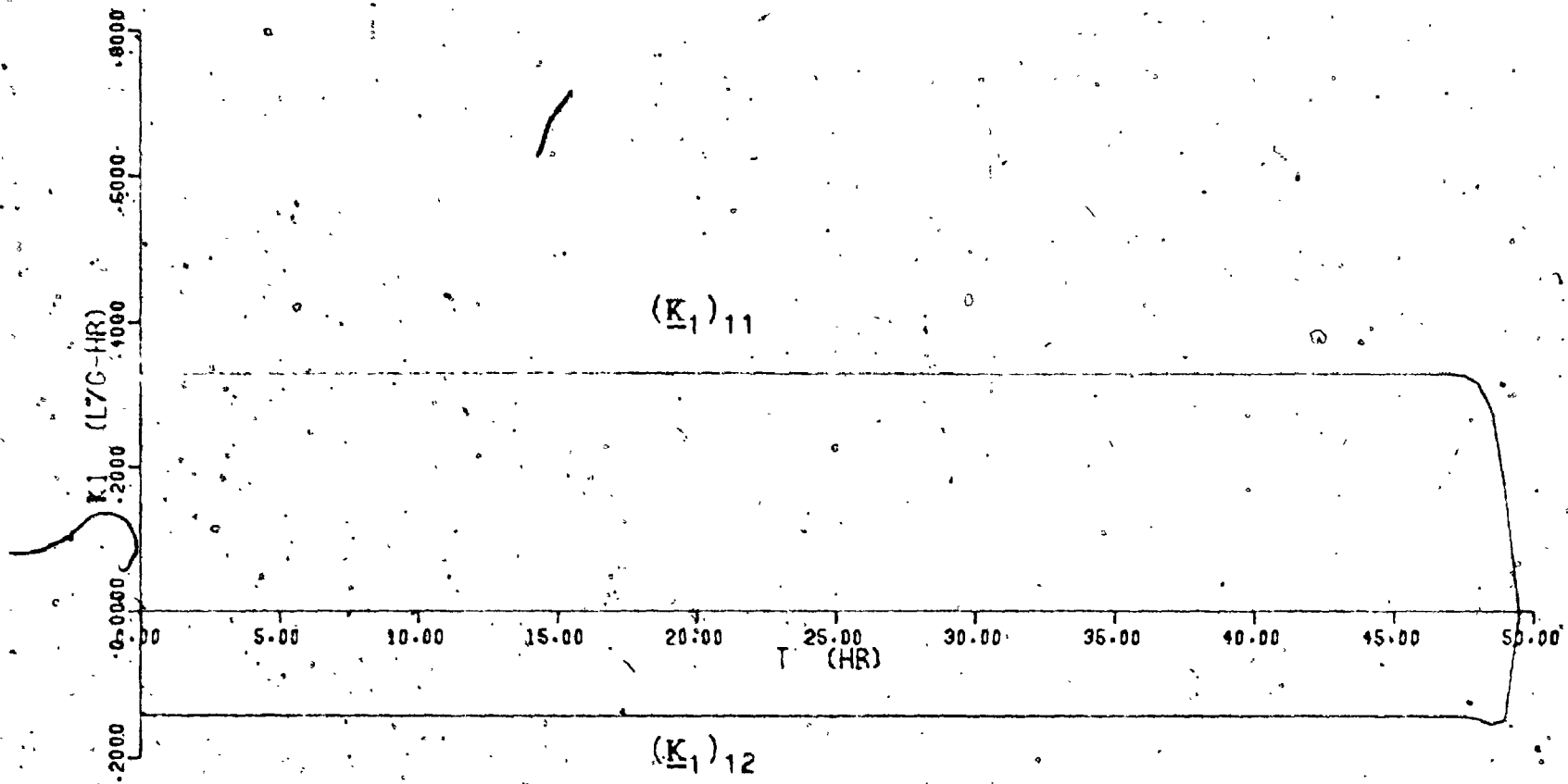


Figure 5.8 Kalman control gain matrix $\underline{K}_1$ versus time for $k_1^* = 1.$, $k_2^* = 10.$, $k_3^* = .5$, $\sigma_1 = \sigma_2 = .02$.

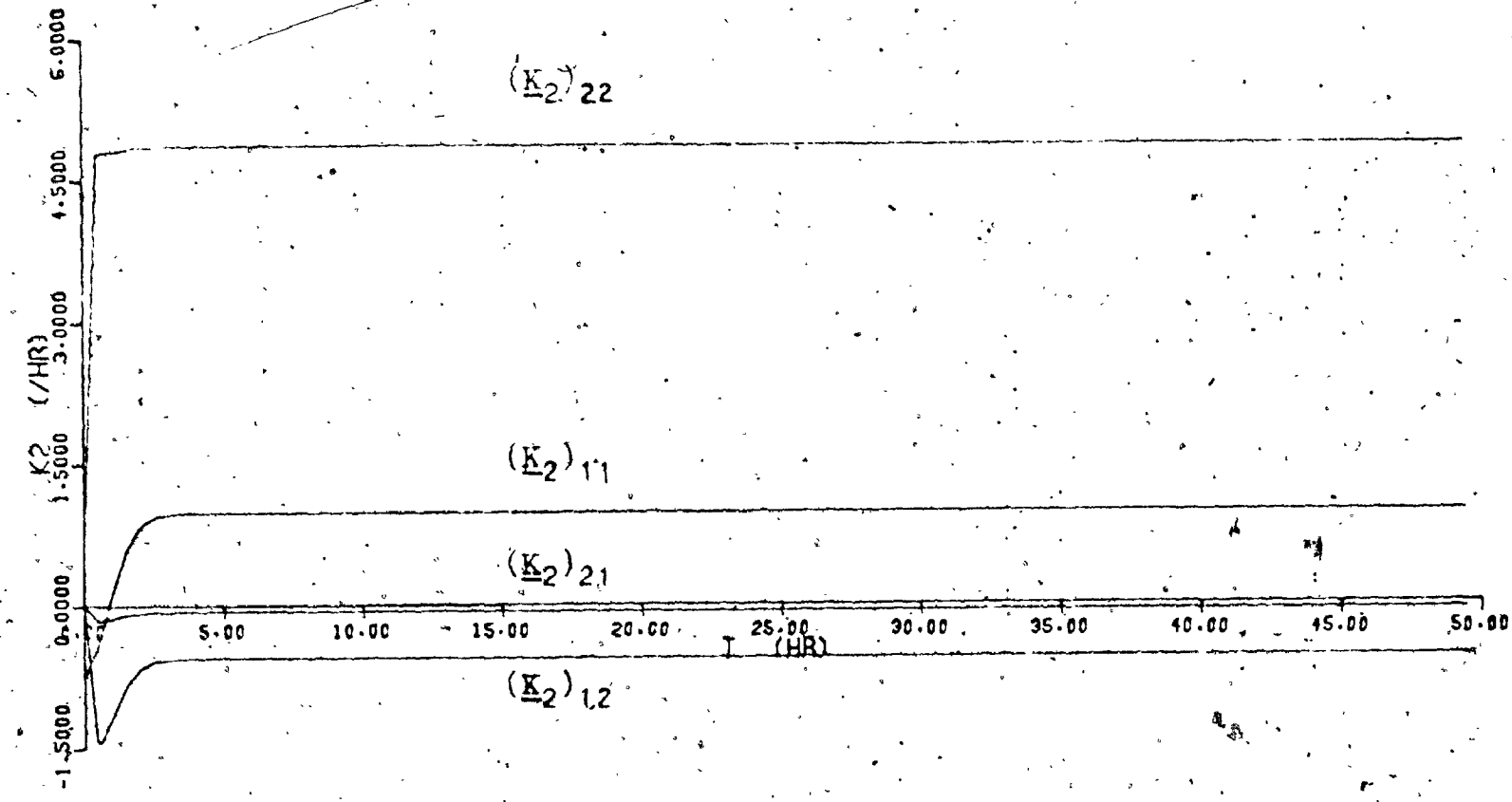


Figure 5.9 Kalman filter gain matrix $\underline{K}_2$ versus time for $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$, $\tau_1 = \tau_2 = .02$.

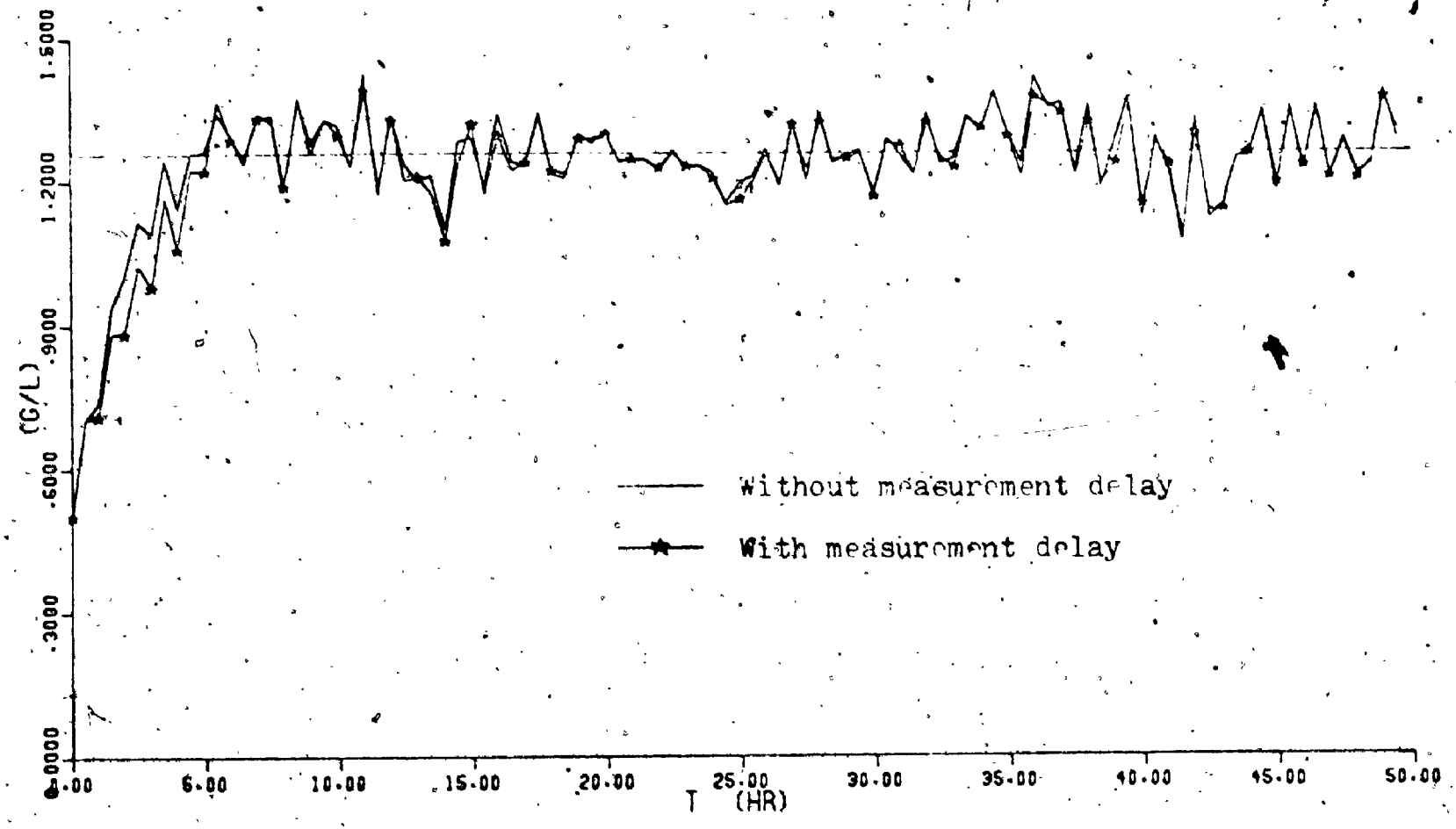


Figure 5.10 Controlled effluent biomass concentrations for cases without measurement delay and with measurement delay of .5 hour, $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$, $\tau_1 = \tau_2 = .02$.

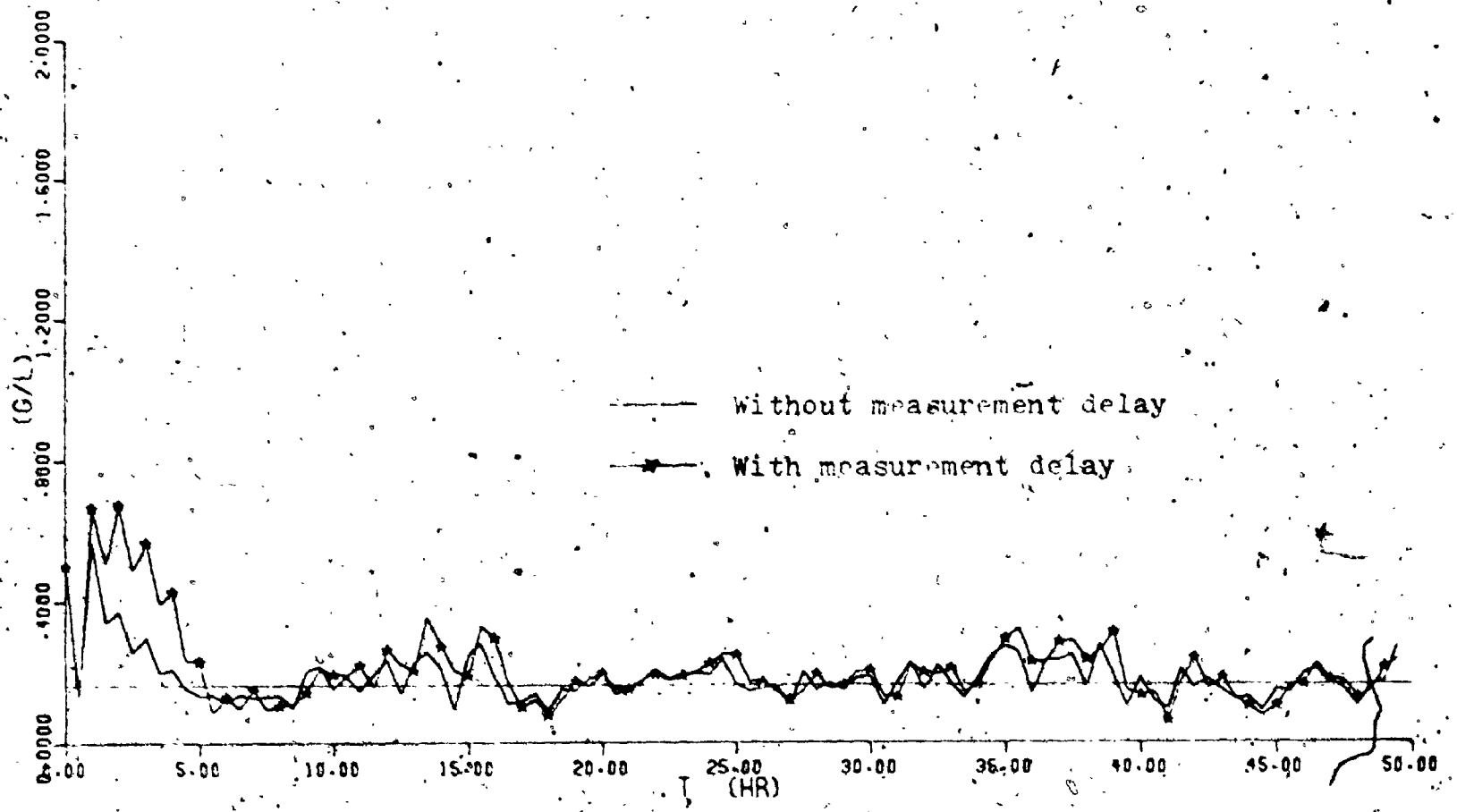


Figure 5.11 Controlled effluent substrate concentrations for cases without measurement delay and with measurement delay of .5 hour, $k_1 = 1.$, $k_2 = 10.$, $k_3 = 5.$, $\tau_1 = \tau_2 = .02$.

CHAPTER VI

CONCLUSION AND RECOMMENDATIONS

A mathematical model of a chemostat with recycle of both biomass and substrate concentrations was derived in Chapter II. From the model, the effluent concentrations do not depend on the recirculation ratio α, the biomass concentration ratio β_C and the substrate concentration ratio β_S separately but rather depend on the recycle factors $\Lambda_C = 1 + \alpha(1 - \beta_C)$ and $\Lambda_S = 1 + \alpha(1 - \beta_S)$. There are many different combinations of α, β_C and β_S yielding the same values of Λ_C and Λ_S. Recycle does not have any effect when $\Lambda_C = \Lambda_S = 1$.

Also in Chapter II, a systematic approach was outlined to compute the optimum operating conditions for a chemostat governed by the general Haldane-Monod model for cases with and without recycle. The performance index is defined as the rate of biomass production. It was found that for the Monod model and with recycle, the optimum performance index is independent of the saturation coefficient K_s. As examples, optimum operating conditions were found for a continuous fermentation of Aerobacter aerogenes, using the model already derived and identified by Topiwala and Sinclair. It was noticed that at the optimum operating conditions, the rate of A. aerogenes production for the case with recycle is more

than thirty times that for the case without recycle. Recommendation for further research is to develop an equipment closest possible to an ideal clarifier so that recycle can be fully utilized. This development is important since it increases biomass production tremendously, more than thirty times as in the fermentation of A. aerogenes. However, as was observed by Ramanthan and Gaudy (1971) it is not an easy task.

In Chapter III, the statement of a general deterministic system identification problem by least squares was mathematically formulated. The most effective optimization techniques used in solving the identification problem were then reviewed. They included the Gauss-Newton technique, its modifications and the quasilinearization technique. Statistical techniques used to determine the best model structure and confidence intervals of parameters were given in Appendix B. As illustrative examples, mathematical models for a batch fermentation of Aureobasidium pullulans 2552 and a continuous fermentation of Morchella crassipes were then particularly identified.

The most significant contribution of the thesis is Chapter IV. In this chapter, difficulty in estimating a large number of parameters in batch process identification was explained. When the model is linear in its state vector,

a technique was derived to surmount this difficulty. This technique is based on the Superposition Principle of linear system theory and the Walling-Lawton-Sylvestre idea of "Elimination of Linear Parameters in Nonlinear Regression". The approach basically replaces the initial conditions by their estimates which are functions of the process parameters that are being identified. Examples were carried out with artificial data simulated from a known model and with experimental data from the batch fermentation of A. pullulans. For both examples, tremendous reduction in computer time and faster convergence were observed. Also, based on the technique, an on-line identification algorithm was derived to update the model parameters as new batch data are being received. The algorithm was tested using simulated data and the results are follows:

- A large amount of computer time can be saved if the algorithm is employed, as expected.

- The on-line identification algorithm is approximately the same as the off-line results when the supplied initial conditions of the algorithm are well known.

- The on-line identification algorithm works very satisfactory even when the supplied initial conditions of the algorithm are unreasonably poor.

It should be noted that the technique is applicable not only to batch fermentation processes but also to other batch pro-

cesses as well. Identification of a linear dynamic continuous process where data at some periods are either missing or not recorded, from the identification viewpoint, can also be treated with the same technique. Recommendation for further research is to find a technique to overcome the difficulty of estimating a large number of parameters when the model is nonlinear in its state vector. The elimination technique derived in Chapter IV works only for linear models, and the technique derived by Woodside and Nuyen (1973) requires extra data, reduces model accuracy and yet takes a lot of computer time. One possibility that can be seen is to combine the Quasilinearization technique and the elimination technique. The Quasilinearization technique transforms a nonlinear model to a linear one and the elimination technique then eliminates the initial conditions.

In Chapter V, a stochastic model was proposed for microbial growth in a continuous fermentation process. The model takes into account measurement errors and disturbances due to imperfect mixing of the bioreactor and of the substrate feed tank. A problem of regulating effluent concentrations was also formulated and solved, using the Separation Theorem. Although the Separation Theorem requires a linear system with a quadratic cost function, a simulation study shows that results from the linearized system with the approximate quadratic cost

function are quite acceptable as long as:

- The control weighting factor is not set too small as compared to the state weighting factors,
- The variances of measurement noise are not too large,
- The delay time in measurement is not too long (For the case with measurement delay).

It is recommended that further research is required in representing a continuous fermentation process as a linear stochastic model around its steady state values and that the developed algorithm based on the Separation Theorem be implemented on a mini-computer to regulate the effluent concentrations.

APPENDIX A

ON THE EQUIVALENCE OF THE GAUSS-NEWTON TECHNIQUE AND THE QUASILINEARIZATION TECHNIQUE IN SYSTEM IDENTIFICATION BY LEAST SQUARES

As mentioned in Chapter III, in system identification by least squares, the Gauss-Newton technique and the Quasilinearization technique, although different in name and in their approach, are actually equivalent. The proof of the equivalence is given here.

It should be noted that for a special case where Equation (3.2) can be reduced to Equation (A.1),

$$\underline{y}(t) = \underline{G}\underline{x}(t) \qquad \cdots (A.1)$$

from which $\underline{G}$ is a known constant matrix, Allison (1958) has proved the equivalence of the Quasilinearization technique and the Parameter-Influence-Coefficient (Sensitivity) technique, a technique which can be easily shown to be the same as the Gauss-Newton technique but with a different name.

The Quasilinearization technique is based on the linearization of the model around an approximate trajectory of $\underline{y}(t)$ as described by Equations (3.40) and (3.41) or equivalently just by Equation (3.46) which is rewritten here as Equation (A.2),

$$\underline{y}(t) = \underline{g}^{\#}(\underline{x}_k(t), \underline{u}_m(t), t)$$

$$+ \left[\frac{\partial g^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right] \underline{\Phi}_k(t) \delta \underline{v}_k(t_o) \qquad (A.2)$$

The Gauss-Newton technique is based on the linearization of $\underline{y}(t)$ around an approximate value of $\underline{\theta}$ as described by Equation (3.11) which is rewritten here as Equation (A.3),

$$\underline{y}(t) = \underline{g}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t)\Big|_{\underline{\theta} = \underline{\theta}_k}$$

$$+ \left[\frac{d\underline{g}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t)}{d\underline{\theta}^T}\right]\Bigg|_{\underline{\theta} = \underline{\theta}_k} \delta \underline{\theta}_k \qquad (A.3)$$

To prove the two techniques are equivalent to each other, it must be shown that Equations (A.2) and (A.3) are equivalent. Directly from the definitions of $\underline{v}$ and $g^{\#}$, it can be seen that:

$$\underline{v}_k(t_o) = \underline{\theta}_k \qquad (A.4)$$

$$g^{\#}(\underline{v}(t), \underline{u}_m(t), t) = \underline{g}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t) \qquad (A.5)$$

and

$$g^{\#}(\underline{v}_k(t), \underline{u}_m(t), t) = \underline{g}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t)\Big|_{\underline{\theta} = \underline{\theta}_k} \qquad (A.6)$$

Thus, from Equations (A.2) and (A.3), the remainder of the proof is to show the validity of the following equation:

$$\left[\frac{\partial g^{\#}(\underline{v}_k(t),\underline{u}_m(t),t)}{\partial \underline{v}_k^T(t)}\right]\underline{\Phi}_k(t) = \left.\frac{dg(x(t),\underline{u}_m(t),\underline{a},t)}{d\underline{\theta}^T}\right|_{\underline{\theta}=\underline{\theta}_k} \tag{A.7}$$

Let RHS be the right hand side and LHS be the left hand side of Equation (A.7), then the proof is as follows:

$$\text{LHS} = \left.\left[\frac{dg(\underline{x}(t),\underline{u}_m(t),\underline{a},t)}{d\underline{\theta}^T}\right]\right|_{\underline{\theta}=\underline{\theta}_k}$$

$$= \left.\left[\frac{dg^{\#}(\underline{v}(t),\underline{u}_m(t),t)}{d\underline{\theta}^T}\right]\right|_{\underline{\theta}=\underline{\theta}_k}$$

(From Equation A.5))

$$= \left.\left[\frac{dg^{\#}(\underline{v}(t),\underline{u}_m(t),t)}{d\underline{v}^T(t)}\right]\left[\frac{d\underline{v}(t)}{d\underline{\theta}^T}\right]\right|_{\underline{\theta}=\underline{\theta}_k}$$

(From chain rule)

$$= \left.\left[\frac{dg^{\#}(\underline{v}(t),\underline{u}_m(t),t)}{d\underline{v}^T(t)}\right]\left[\frac{d\underline{v}(t)}{d\underline{v}^T(t_o)}\right]\right|_{\underline{v}(t_o)=\underline{v}_k(t_o)}$$

(From Equation A.4))

$$= \left.\left[\frac{dg^{\#}(\underline{v}(t),\underline{u}_m(t),t)}{d\underline{v}^T(t)}\right]\left[\frac{d\,\underline{v}_k(t)}{d\,\underline{v}_k^T(t_o)}\right]\right|_{\underline{v}(t_o)=\underline{v}_k(t_o)}$$

(From Equation (3.43))

$$= \left[\frac{dg^{\#}(\underline{v}(t), \underline{u}_m(t), t)}{d\underline{v}^T(t)}\right] \underline{\Phi}_k(t) \Bigg|_{\underline{v}(t_o) = \underline{v}_k(t_o)}$$

(From Equation (3.44))

$$= \left[\frac{\partial g^{\#}(\underline{v}(t), \underline{u}_m(t), t)}{\partial \underline{v}^T(t)}\right] \underline{\Phi}_k(t) \Bigg|_{\underline{v}(t_o) = \underline{v}_k(t_o)}$$

(Because $\underline{u}_m(t)$ does not depend on $\underline{v}^T(t)$)

$$= \left[\frac{\partial g^{\#}(\underline{v}(t), \underline{u}_m(t), t)}{\partial \underline{v}^T(t)}\right] \underline{\Phi}_k(t) \Bigg|_{\underline{v}(t) = \underline{v}_k(t)}$$

(Because $\underline{v}(t_o) = \underline{v}_k(t_o)$ and $\underline{v}(t) = \underline{v}_k(t)$ provide the same information)

$$= \left[\frac{\partial g^{\#}(\underline{v}_k(t), \underline{u}_m(t), t)}{\partial \underline{v}_k^T(t)}\right] \underline{\Phi}_k(t)$$

= RHS

APPENDIX B

BATCH FERMENTATION OF AUREOBASIDIUM PULLULANS 2552

The following is a brief presentation of materials and methods used to cultivate Aureobasidium pullulans 2552, the data of which were employed in Chapters III and IV. For a more complete presentation, one can consult LeDuy (1972) and LeDuy and Zajic (1973).

Seven batches of A. pullulans 2552 have been grown in a GF 0007 CHEMAP glass fermentor containing five liters of culture medium. The composition of the culture medium before sterilization was: 5% or 10% sucrose, 0.5% K_2HPO_4, 0.1% NaCl, 0.02% $MgSO_4$, 0.06% $(NH_4)_2SO_4$, 0.04% yeast extract (Difco), and pH adjusted to 5.5 or 6.5. The aeration, agitation and temperature were controlled at roughly 2.5 liters/min, 750 rpm and 23.5°C respectively. Immediately after inoculation, culture medium was removed through a modified sampling tube (LeDuy and Zajic 1973), analyzed and these data were used as the measured initial conditions for that batch. Subsequent analyses were completed at a time-interval of 12 hours or 24 hours for about six days.

The evaluations of biomass concentration and polysaccharide concentration and the determination of sucrose

concentration were as follows.. The sample was diluted to lower the viscosity and centrifuged at 15000 rpm for 45 minutes to remove the cells. The settled cells were washed with distilled water, centrifuged, washed again with 2N HCl solution, centrifuged and finally washed with distilled water and recentrifuged. The cells were dried under vacuum at 70°C to a constant weight. The polysaccharide was precipitated with an equal volume of methanol from the supernatant obtained after centrifugation and the removal of cells. The precipitated polysaccharide fibers were centrifuged at 15000 rpm for 20 minutes. The settled polysaccharide was dried under vacuum to a constant weight. The residual sucrose was determined by orcinol reagent (Snell and Snell 1957) from the methanol-stripped supernatant obtained from the centrifugation of polysaccharide.

The collected data which consist of measured pH, biomass concentration, polysaccharide concentration and sucrose conoentration, for seven batch runs, are shown on Tables B.1 to B.7 respectively.

Table B.1 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #1

Time (hr)	H_m (unitless)	C_m (g/100 ml)	P_m (g/100 ml)	S_m (g/100 ml)
0.0	5.375	0.0099	0.050	4.95
12.0	3.042	0.0827	0.067	4.73
24.0	2.660	0.3201	0.426	3.62
36:0	2.662	0.4279	1.061	2.53
48.0	2.640	0.5309	1.624	1.87
61.0	2.645	0.6256	2.296	1.12
72.0	2.642	0.7375	2.427	0.41
83.0	2.635	0.7764	2.854	0.08
94.0	2.630	0.8015	3.338	0.08
106.0	2.638	0.8293	3.173	0.05
118.0	2.644	0.8565	3.403	0.02
130.0	2.645	0.8739	3.679	0.03
143.0	2.643	0.9467	3.551	0.02

Table B.2 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #2

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	5.320	0.0108	0.056	8.50
12.0	3.185	0.0658	0.080	8.36
24.0	2.630	0.2439	0.129	8.10
36.0	2.622	0.4025	0.181	7.39
48.0	2.607	0.4876	1.858	6.37
60.0	2.604	0.6478	2.780	6.87
84.0	2.596	0.8156	4.043	6.68
106.0	2.565	1.0220	4.928	7.19
119.0	2.535	1.1263	6.625	7.14

Table B.3 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #3

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	6.445	0.0093	0.060	4.22
13.0	6.050	0.3757	0.105	4.48
24.0	4.160	0.2852	0.263	3.93
36.00	4.070	0.4863	0.840	3.12
48.0	4.035	0.6258	0.616	2.48
61.0	3.965	0.8095	2.383	2.03
84.00	3.930	1.0057	3.361	0.76
96.0	3.885	1.1064	4.157	0.26
109.0	3.974	1.2499	4.266	0.22

Table B.4 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #4

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	6.425	0.0134	0.085	4.45
12.0	5.525	0.1191	0.100	4.66
24.0	4.003	0.3170	0.479	3.57
36.0	3.905	0.4068	1.053	2.81
48.0	3.861	0.5026	1.595	2.35
60.0	3.965	0.5324	2.165	1.77
85.0	4.140	0.6677	2.734	0.62
96.0	4.152	0.7641	3.795	0.22
120.0	4.312	0.8390	4.088	0.21
132.0	4.313	0.8026	3.292	0.23
143.0	4.310	0.8334	3.604	0.20

Table B.5 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #5

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	5.510	0.0250	0.056	8.13
12.0	3.470	0.0707	0.069	7.65
24.0	2.740	0.2878	0.313	7.70
36.0	2.752	0.4139	0.987	6.65
48.0	2.733	0.4674	1.624	6.21
61.0	2.745	0.5990	2.146	5.50
84.0	2.737	0.7134	3.099	4.53
96.0	2.730	0.7629	3.732	4.25
108.0	2.744	0.7939	3.945	3.86
120.0	2.737	0.9046	5.057	3.95
132.0	2.740	0.9632	4.501	3.31
148.0	2.735	1.0451	4.968	2.93

Table B.6 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #6

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	5.305	0.0166	0.080	4.56
12.0	2.805	0.0546	0.119	4.81
24.0	2.695	0.3547	0.422	3.45
36.0	2.700	0.5274	1.188	2.75
48.0	2.660	0.6817	1.851	2.07
60.0	2.660	0.7502	1.994	1.57
84.0	2.657	0.9182	3.024	0.66
99.0	2.665	1.1061	3.473	0.25
108.0	2.660	1.1312	4.013	0.09

Table B.7 Measured pH H_m, biomass concentration C_m, polysaccharide concentration P_m and sucrose concentration S_m from batch fermentation of Aureobasidium pullulans 2552.

- BATCH #7

Time (hr)	H_m (unitless)	C_m (g/100ml)	P_m (g/100ml)	S_m (g/100ml)
0.0	6.410	0.0274	0.090	8.12
13.0	5.980	0.0752	0.126	8.13
26.0	4.280	0.2634	0.475	6.70
36.0	3.998	0.3416	0.958	5.78
47.0	3.870	0.6985	1.446	5.61
72.0	3.870	0.4995	3.125	4.85
84.0	3.870	0.5228	2.993	4.75
96.0	3.865	0.5560	4.258	4.13
109.0	3.860	0.5895	4.197	3.77
121.0	3.862	0.6662	5.844	3.45
133.0	3.830	0.7325	6.004	3.17
144.0	3.825	0.7720	6.456	2.64

APPENDIX C

CONFIDENCE INTERVALS OF PARAMETERS AND THE BACKWARD ELIMINATION TECHNIQUE IN SYSTEM IDENTIFICATION BY LEAST SQUARES

I CONFIDENCE INTERVALS OF PARAMETERS

I.1 Definition

A confidence interval of a random parameter θ_i with a significance level α is an interval in which a sample of θ_i will lie with a probability $(1 - \alpha)$.

$(1 - \alpha)$ is called a confidence level, and the interval is then called a $(1 - \alpha)$-confidence interval.

In normal practice α is often chosen as .05 or 5% so that $(1 - \alpha) = .95$ or 95%.

In general, a confidence interval is chosen in such a way that the expected value (the statistical mean) of the random parameter θ_i divides the interval into two subintervals with equal probability. If (θ_i', θ_i'') is a $(1 - \alpha)$-confidence interval of the parameter θ_i with the expected value $E\{\theta_i\}$ and the probability density function $p(\theta_i)$. Then as shown in Figure C.1 that

$$\theta_i' < E\{\theta_i\} < \theta_i'' \qquad (C.1)$$

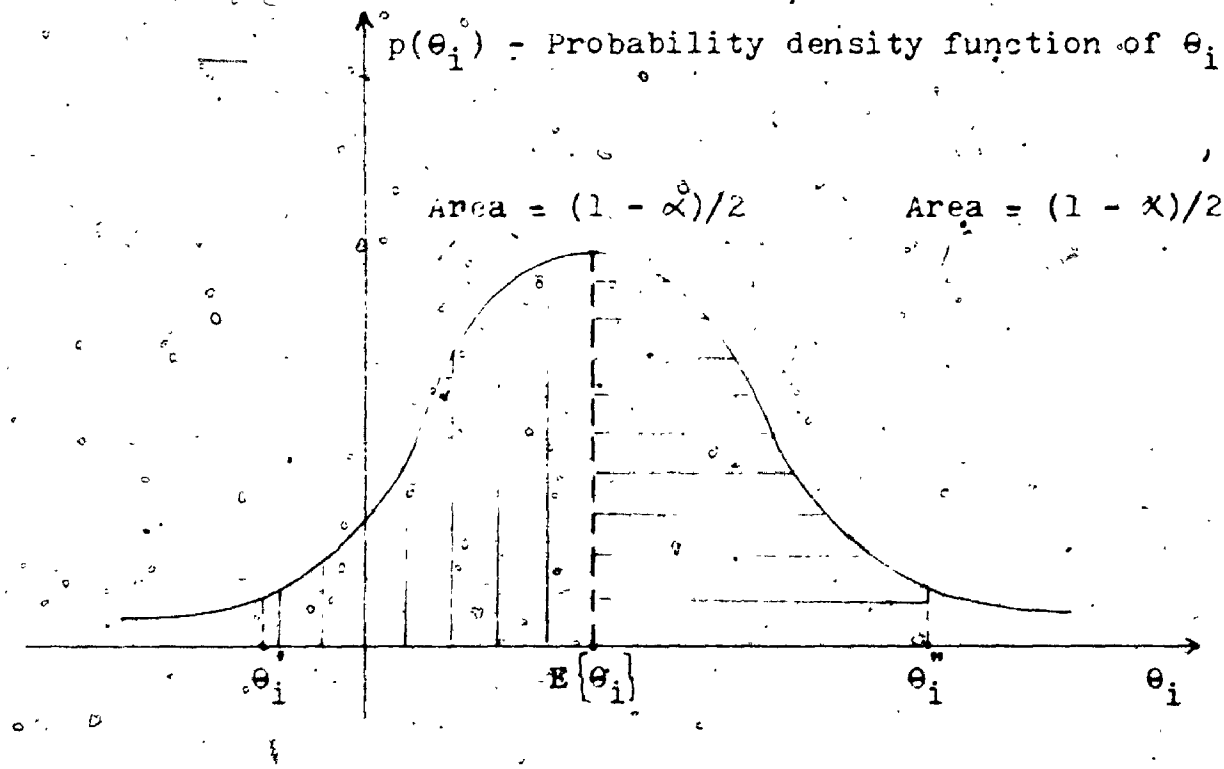


Figure 3.1 $(1 - \alpha)$-confidence interval (θ_i', θ_i'') of the random parameter θ_i with the expected value $E\{\theta_i\}$ and the probability density function $p(\theta_i)$.

$$\int_{\theta_i'}^{E\{\theta_i\}} p(\theta_i)d\theta_i = \int_{E\{\theta_i\}}^{\theta_i''} p(\theta_i)d\theta_i = (1 - \alpha)/2 \qquad (C.2)$$

I.2 Confidence Intervals of Parameters in System Identification by Least Squares

In system identification by least squares as described in Chapter III, one wishes to find $\underline{\theta}$ so that

$$S = \frac{1}{2}\sum_{i=0}^{I} \left\| \underline{y}_m(t_i) - \underline{y}(t_i) \right\|^2_{\underline{y}_i} \qquad (C.3)$$

is minimized over $\underline{\theta}$ and subject to:

$$\underline{y}(t) = \underline{g}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t) \qquad (C.4)$$

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}_m(t), \underline{a}, t) \qquad (C.5)$$

and

$$\underline{\theta} = \begin{bmatrix} \underline{a} \\ \hline \underline{x}(t_o) \end{bmatrix} \qquad (C.6)$$

To find the confidence intervals of the parameters, it is necessary to assume that

$$\underline{e}(t_i) = \underline{y}_m(t_i) - \underline{y}(t_i) \qquad (C.7)$$
$$i = 0, 1, \ldots, I$$

is a white, Gaussian noise sequence with zero-mean and $(\sigma^2 \underline{y}_i^{-1})$-variance ($\sigma$ is unknown, real and positive).

Suppose the Gauss-Newton technique described in Chapter III is used and after $(k+1)^{th}$ iterations the sequence $\{\underline{\theta}_k\}$ converges to $\hat{\underline{\theta}}$. Then at the last iteration, the $(k+1)^{th}$ iteration, $\underline{y}(t_i)$ is essentially linear in $\underline{\theta}_k$. That is,

$$\underline{y}(t_i) = y(t_i)\Big|_{\underline{\theta} = \underline{\theta}_k} + \underline{Q}_i(\underline{\theta} - \underline{\theta}_k) \qquad (C.8)$$

where

$$\underline{Q}_i = \left[\frac{d\underline{y}(t_i)}{d\underline{\theta}^T}\right]\Bigg|_{\underline{\theta} = \underline{\theta}_k} \qquad (C.9)$$

Thus, S defined in Equation (C.3) becomes:

$$S = \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{Y}_m(t_i) - \underline{y}(t_i)\Big|_{\underline{\theta} = \underline{\theta}_k} - \underline{Q}_i(\underline{\theta} - \underline{\theta}_k)\right\|^2_{\underline{V}_i} \qquad (C.10)$$

At $\underline{\theta} = \hat{\underline{\theta}}$, the derivative of S with respect to $\underline{\theta}$ is equal to zero. That is,

$$\frac{dS}{d\underline{\theta}} = \underline{0} \quad \text{at } \underline{\theta} = \hat{\underline{\theta}}$$

or,

$$-\sum_{i=0}^{I}\underline{Q}_i^T\underline{V}_i\left[\underline{Y}_m(t_i) - \underline{y}(t_i)\Big|_{\underline{\theta} = \underline{\theta}_k} - \underline{Q}_i(\underline{\theta} - \underline{\theta}_k)\right] = \underline{0} \qquad (C.11)$$

or,

$$\hat{\underline{\theta}} - \underline{\theta}_k = \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{Q}_i\right\}^{-1} \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \left[\underline{Y}_m(t_i) - \underline{Y}(t_i)\right]\right|_{\underline{\theta} = \underline{\theta}_k} \tag{C.12}$$

From Equations (C.7) and (C.8), one has:

$$\underline{Y}_m(t_i) - \underline{Y}(t_i)\Big|_{\underline{\theta} = \underline{\theta}_k} = \underline{e}(t_i) + \underline{Q}_i(\underline{\theta} - \underline{\theta}_k) \tag{C.13}$$

Combining Equations (C.12) and (C.13) gives:

$$\hat{\underline{\theta}} - \underline{\theta}_k = \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{Q}_i\right\}^{-1} \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \left[\underline{e}(t_i) + \underline{Q}_i(\underline{\theta} - \underline{\theta}_k)\right]\right\} \tag{C.14}$$

or,

$$\hat{\underline{\theta}} - \underline{\theta}_k = \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{Q}_i\right\}^{-1} \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{e}(t_i)\right\} + (\underline{\theta} - \underline{\theta}_k) \tag{C.15}$$

or,

$$\hat{\underline{\theta}} - \underline{\theta} = \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{Q}_i\right\}^{-1} \left\{\sum_{i=0}^{I} \underline{Q}_i^T \underline{V}_i \underline{e}(t_i)\right\} \tag{C.16}$$

Since $\underline{V}_i$ is a positive definite matrix, a square matrix $\underline{W}_i$ can be found such that (Bellman 1960, Johnston 1963):

$$\underline{W}_i^T \underline{W}_i = \underline{V}_i \tag{C.17}$$

Combining Equations (C.16) and (C.17) results in:

$$\hat{\underline{\theta}} - \underline{\theta} = \left\{ \sum_{i=0}^{I} (\underline{W}_i \underline{Q}_i)^T (\underline{W}_i \underline{Q}_i) \right\}^{-1} \left\{ \sum_{i=0}^{I} (\underline{W}_i \underline{Q}_i)^T \underline{W}_i \underline{e}(t_i) \right\} \quad (C.18)$$

or,

$$\hat{\underline{\theta}} - \underline{\theta} = (\underline{Q}^T \underline{Q})^{-1} \underline{Q}^T \underline{\xi} \quad (C.19)$$

where,

$$\underline{Q} = \begin{bmatrix} \underline{W}_0 \underline{Q}_0 \\ \hline \underline{W}_1 \underline{Q}_1 \\ \hline \vdots \\ \hline \underline{W}_I \underline{Q}_I \end{bmatrix} \quad (C.20)$$

and

$$\underline{\xi} = \begin{bmatrix} \underline{W}_0 \underline{e}(t_0) \\ \hline \underline{W}_1 \underline{e}(t_1) \\ \hline \vdots \\ \hline \underline{W}_I \underline{e}(t_I) \end{bmatrix} \quad (C.21)$$

One has:

- $\underline{e}(t_i)$ is Gaussian,
- $E\left\{ \underline{W}_i \underline{e}(t_i) \right\} = \underline{W}_i E\left\{ \underline{e}(t_i) \right\} = \underline{0}$ (C.22)
- $\mathrm{COV}\left\{ \underline{W}_i \underline{e}(t_i) \right\} = E\left\{ \left[\underline{W}_i \underline{e}(t_i) - E\left\{ \underline{W}_i \underline{e}(t_i) \right\} \right] \left[\underline{W}_i \underline{e}(t_i) \right]^T \right\}$

$$= E\left\{ \underline{W}_i \underline{e}(t_i) \right\}^T \Big\}$$

$$= E\left\{\underline{W}_i \underline{e}(t_i)\underline{e}^T(t_i)\underline{W}_i^T\right\}$$

$$= \underline{W}_i \sigma^2 \underline{V}_i^{-1} \underline{W}_i^T$$

$$= \sigma^2 \underline{W}_i (\underline{W}_i^T \underline{W}_i)^{-1} \underline{W}_i^T$$

$$= \sigma^2 \underline{I}_\ell \qquad (C.23)$$

where $\underline{I}_\ell$ is a $(\ell \times \ell)$-identity matrix and ℓ is the dimension of the residual vector $\underline{e}(t_i)$ or equivalently the dimension of the output vector $\underline{y}(t_i)$.

Therefore $\underline{W}_i \underline{e}(t_i)$ has a Gaussian distribution with zero-mean and $(\sigma^2 \underline{I}_\ell)$-variance,

$$\underline{W}_i \underline{e}(t_i) \underset{c}{\sim} N(\underline{0}, \sigma^2 \underline{I}_\ell) \qquad (C.24)$$

Also from Equations (C.24) and (C.21), $\underline{\xi}$ has a Gaussian distribution with zero-mean and $(\sigma^2 \underline{I}_d)$-variance,

$$\underline{\xi} \sim N(\underline{0}, \sigma^2 \underline{I}_d) \qquad (C.25)$$

where,

$$d = (I+1)\ell \qquad (C.26)$$

Thus from Equations (C.25) and (C.19), $(\hat{\underline{\theta}} - \underline{\theta})$ has a Gaussian distribution with its mean and its covariance calculated as:

$$E\left\{\hat{\underline{\theta}} - \underline{\theta}\right\} = E\left\{(\underline{\Omega}^T\underline{\Omega})^{-1}\underline{\Omega}^T\underline{\xi}\right\}$$

$$= (\underline{\Omega}^T\underline{\Omega})^{-1}\underline{\Omega}^T E\left\{\underline{\xi}\right\}$$

$$= \underline{0} \qquad (C.27)$$

$$\begin{aligned}\mathrm{COV}\left\{\hat{\underline{\theta}} - \underline{\theta}\right\} &= E\left\{\left[(\hat{\underline{\theta}} - \underline{\theta}) - E\left\{\hat{\underline{\theta}} - \underline{\theta}\right\}\right]\left[(\hat{\underline{\theta}} - \underline{\theta}) - E\left\{\hat{\underline{\theta}} - \underline{\theta}\right\}\right]^T\right\} \\ &= E\left\{(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\underline{\xi}\underline{\xi}^T\underline{Q}(\underline{Q}^T\underline{Q})^{-1}\right\} \\ &= (\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T E\left\{\underline{\xi}\underline{\xi}^T\right\}\underline{Q}(\underline{Q}^T\underline{Q})^{-1} \\ &= (\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\sigma^2\underline{I}_d\underline{Q}(\underline{Q}^T\underline{Q})^{-1} \\ &= \sigma^2(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\underline{Q}(\underline{Q}^T\underline{Q})^{-1} \\ &= \sigma^2(\underline{Q}^T\underline{Q})^{-1} \qquad (C.28)\end{aligned}$$

therefore,

$$(\hat{\underline{\theta}} - \underline{\theta}) \sim N\left\{\underline{0}, \sigma^2(\underline{Q}^T\underline{Q})^{-1}\right\} \qquad (C.29)$$

From Equations (C10) and (C.17), the minimum sum of squares is:

$$\hat{S} = \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{W}_i\left[\underline{y}_m(t_i) - \underline{y}(t_i)\Big|_{\underline{\theta} = \underline{\theta}_k} - \underline{Q}_i(\hat{\underline{\theta}} - \underline{\theta}_k)\right]\right\|^2 \qquad (C.30)$$

or from Equation (C.13),

$$\begin{aligned}\hat{S} &= \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{W}_i\left[\underline{e}(t_i) + \underline{Q}_i(\underline{\theta} - \underline{\theta}_k) - \underline{Q}_i(\hat{\underline{\theta}} - \underline{\theta}_k)\right]\right\|^2 \\ &= \tfrac{1}{2}\sum_{i=0}^{I}\left\|\underline{W}_i\left[\underline{e}(t_i) - \underline{Q}_i(\hat{\underline{\theta}} - \underline{\theta})\right]\right\|^2 \qquad (C.31)\end{aligned}$$

or from Equations (C.20) and (C.21),

$$\hat{S} = \tfrac{1}{2}\left\|\underline{\varepsilon} - \underline{\varphi}(\hat{\underline{\theta}} - \underline{\theta})\right\|^2 \qquad (C.32)$$

or from Equation (C.19),

$$\begin{aligned} S &= \tfrac{1}{2}\left\|\underline{\varepsilon} - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\underline{\varepsilon}\right\|^2 \\ &= \tfrac{1}{2}\left\|\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]\underline{\varepsilon}\right\|^2 \\ &= \tfrac{1}{2}\,\underline{\varepsilon}^T\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]^T\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]\underline{\varepsilon} \\ &= \tfrac{1}{2}\,\underline{\varepsilon}^T\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right. \\ &\quad \left. + \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]\underline{\varepsilon} \end{aligned}$$

or,

$$S = \tfrac{1}{2}\,\underline{\varepsilon}^T\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]\underline{\varepsilon} \qquad (C.33)$$

The matrix $\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]$ is symmetric, idempotent and has its trace computed as:

$$\begin{aligned} \mathrm{tr}\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right] &= \mathrm{tr}\left[\underline{I}_d\right] - \mathrm{tr}\left[\underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right] \\ &= d - \mathrm{tr}\left[(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\underline{\varphi}\right] \\ &= d - \mathrm{tr}\left[\underline{I}_b\right] \\ &= d - b \end{aligned} \qquad (C.34)$$

therefore $\left[\underline{I}_d - \underline{\varphi}(\underline{\varphi}^T\underline{\varphi})^{-1}\underline{\varphi}^T\right]$ has $(d - b)$ as its rank (Press 1972). Note that b is the order of the square matrix $(\underline{\varphi}^T\underline{\varphi})$ or equivalently the dimension of the parameter vector $\underline{\theta}$.

From Equation (C.33), $2\hat{S}$ is a quadratic form in d independent Gaussian variables having zero-mean and common

variance σ^2, and $\left[\underline{I}_d - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\right]$ is idempotent and of rank $(d - D)$. Thus it can be said that (Johnston 1963) $2\hat{S}/\sigma^2$ has a chi-square distribution with $(d - D)$ degrees of freedom,

$$\frac{2\hat{S}}{\sigma^2} \sim \chi^2_{d-D} \qquad (C.35)$$

From Equations (C.29) and (C.35), if one lets q_{ii} be the (i,i)-element of the square matrix $(\underline{Q}^T\underline{Q})^{-1}$, then directly from the definition (Wonnacott and Wonnacott 1970),

$$\frac{\hat{\theta}_i - \theta_i}{\sqrt{\dfrac{\sigma^2 q_{ii}}{\dfrac{2\hat{S}}{\sigma^2(d - D)}}}} = (\hat{\theta}_i - \theta_i)\sqrt{(d - D)/2q_{ii}\hat{S}}$$

has a t-distribution with $(d - D)$ degrees of freedom if $(\hat{\theta}_i - \theta_i)$ and $\hat{S}$ are statistically independent. They are in fact independent because if let

$$2\hat{S} = \underline{s}^T\underline{s} \qquad (C.36)$$

then,

$$\underline{s} = \left[\underline{I}_d - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\right]\underline{\varepsilon} \qquad (C.37)$$

and

$$\begin{aligned} E\left\{\underline{s}(\hat{\underline{\theta}} - \underline{\theta})^T\right\} &= E\left\{\left[\underline{I}_d - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\right]\underline{\varepsilon}\left[(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\underline{\varepsilon}\right]^T\right\} \\ &= E\left\{\underline{\varepsilon}\underline{\varepsilon}^T\underline{Q}(\underline{Q}^T\underline{Q})^{-1} - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\underline{\varepsilon}\underline{\varepsilon}^T\underline{Q}(\underline{Q}^T\underline{Q})^{-1}\right\} \\ &= E\left\{\underline{\varepsilon}\underline{\varepsilon}^T\right\}\underline{Q}(\underline{Q}^T\underline{Q})^{-1} - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T E\left\{\underline{\varepsilon}\underline{\varepsilon}^T\right\}\underline{Q}(\underline{Q}^T\underline{Q})^{-1} \\ &= \sigma^2\underline{I}_d\underline{Q}(\underline{Q}^T\underline{Q})^{-1} - \underline{Q}(\underline{Q}^T\underline{Q})^{-1}\underline{Q}^T\sigma^2\underline{I}_d\underline{Q}(\underline{Q}^T\underline{Q})^{-1} \end{aligned}$$

$$= \sigma^2 \underline{Q}(\underline{Q}^T\underline{Q})^{-1} - \sigma^2 \underline{Q}(\underline{Q}^T\underline{Q})^{-1}$$

$$= \underline{0} \qquad (C.38)$$

Thus, for a significance level α, the confidence interval of parameter θ_i is:

$$- t_{d-D,(1-\alpha)/2} < (\hat{\theta}_i - \theta_i)\sqrt{\frac{d-D}{2q_{ii}S}} < t_{d-D,(1-\alpha)/2}$$

or

$$\hat{\theta}_i - \hat{\sigma}\sqrt{q_{ii}}\, t_{d-D,(1-\alpha)/2} < \theta_i < \hat{\theta}_i - \hat{\sigma}\sqrt{q_{ii}}\, t_{d-D,(1-\alpha)/2} \qquad (C.39)$$

where $\hat{\sigma}$ is an unbiased estimate of σ and is computed as,

$$\hat{\sigma} = \sqrt{\frac{2S}{d-D}} \qquad (C.40)$$

and $t_{d-D,(1-\alpha)/2}$, as shown in Figure C:2, is a value such that if $p(t_{d-D})$ is the probability density function of a random variable t_{d-D} having t-distribution with $(d - D)$-degrees of freedom, then,

$$\int_{-t_{d-D,(1-\alpha)/2}}^{0} p(t_{d-D})dt_{d-D} = \int_{0}^{t_{d-D,(1-\alpha)/2}} p(t_{d-D})dt_{d-D} = (1-\alpha)/2 \qquad (C.41)$$

II. THE BACKWARD ELIMINATION TECHNIQUE

II.1 Introduction

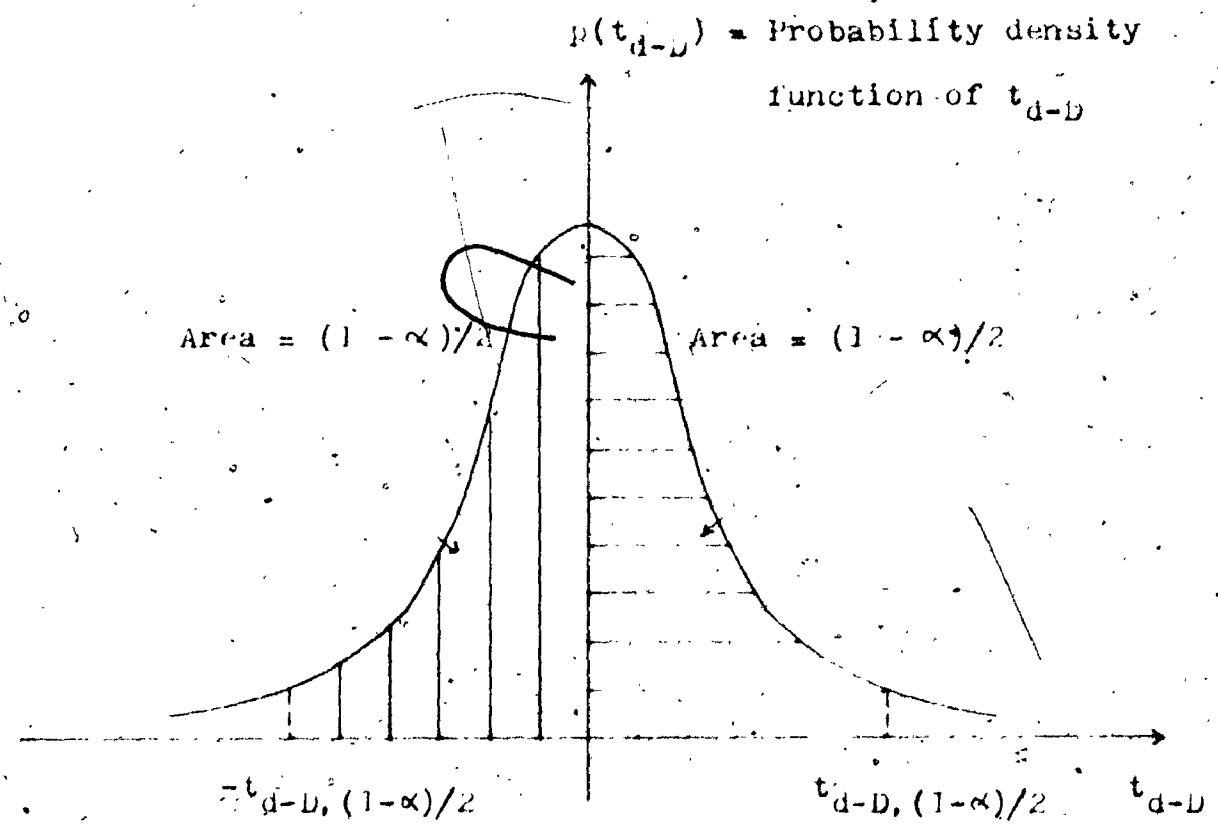


Figure C.2 $(1-\alpha)$-confidence interval of a random variable t_{d-D} having t-distribution with $(d-D)$-degrees of freedom.

In general, the structure of a model that represents the system is not confidently known. It must be chosen to meet the following two opposed criteria:

- It should be as simple as possible due to the costs and difficulty involved in obtaining information on a large number of terms in the model, and consequently in controlling the system using the derived model.

- The model should describe all characteristics of the system accurately. That is the structure should have high order and should consist of many terms.

A good model structure must be a compromise between the above two criteria. In control literature, very recently, there appeared several tests to estimate the order of a system (Woodside 1971, Chow 1972, Unbehaven and Gohring 1974, Van Den Boom and Van Den Enden 1974). To select terms in a linear model, Draper and Smith (1966) have given six different procedures: all possible regression, backward elimination, forward selection, stepwise regression, two variations of the four previous procedures, and stagewise regression.

From the Gauss-Newton technique, when the approximations of the parameters are sufficiently close to their estimates, that is at the last iteration, a nonlinear dynamic state-space model as shown in Chapter III can be treated

as a linear model. And therefore, if the order of the model is confidently known, the best model structure can be found from any one of the Draper-Smith procedures.

II.2 The Backward Elimination Technique

The backward elimination technique was used in Chapter III to determine the best model structure for the microbial growth in the batch fermentation process of A. pullulans 2552. It is based on the F-test for significance of the parameters and is as follows:

1. Initially, a model with all possible parameters is fitted (estimated).

2. A partial F-test value for each parameter appearing in the present model is calculated. For parameter θ_i, the F-test value is:

$$F = \frac{\hat{S}_2 - \hat{S}_1}{\hat{S}_1}(d - D) \qquad (C.42)$$

where

$\hat{S}_1$ = Minimum sum of squares S from the present model of D parameters including θ_i,

$\hat{S}_2$ = Minimum sum of squares S from the present model with the term containing θ_i ignored,

d = Number of data points.

3. The lowest partial F-test value, say F_L, is then com-

pared with tabulated value $F_{1-\alpha}$ where α is a prespecified significance level value, say 5%, and $F_{1-\alpha}$, as shown on Figure 3.3, is a value such that if p(F) is the probability density function of a random variable F having F-distribution with (1, d - b)-degrees of freedom, then,

$$\int_0^{F_{1-\alpha}} p(F)dF = 1 - \alpha \qquad (3.43)$$

If

. $F_L < F_{1-\alpha}$, set the parameter which gave rise to F_L to zero and then go back to step 2.

. $F_L \geq F_{1-\alpha}$, accept the present model as the best one.

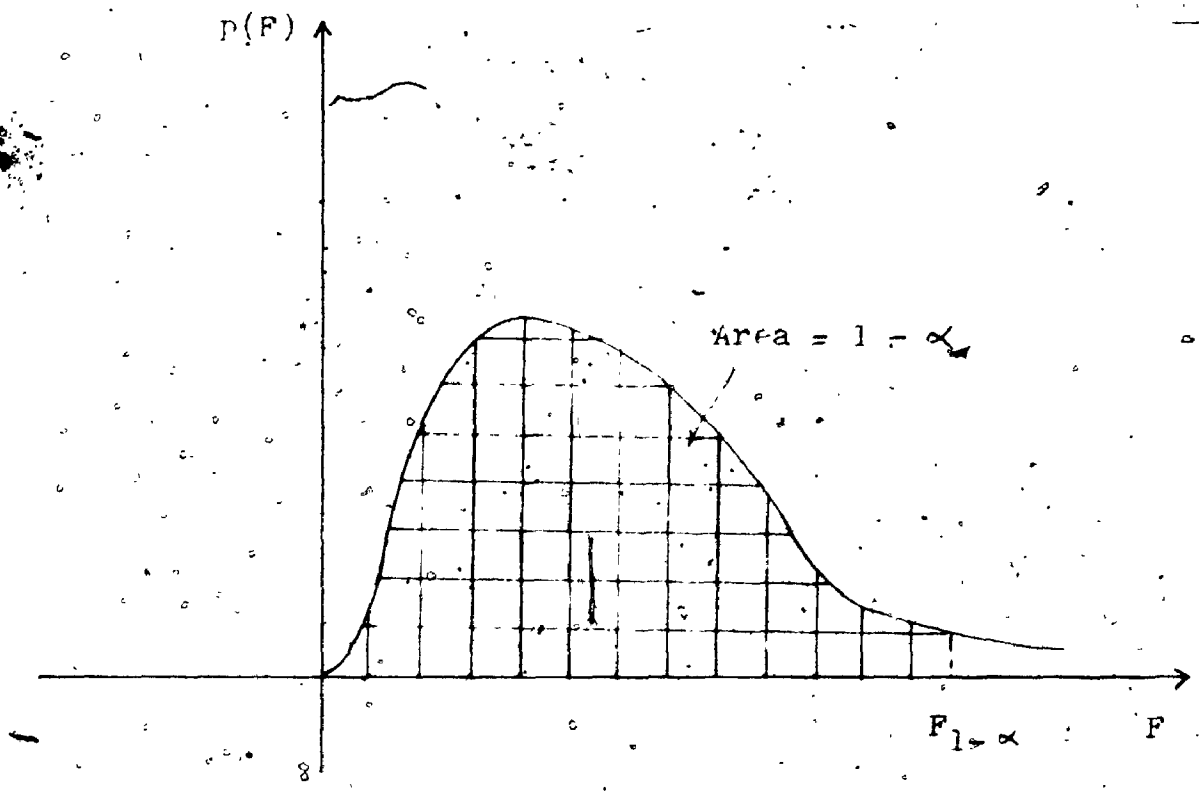


Figure 2.3 Graphical representation of the value $F_{1-\alpha}$. p(F) is the probability density function of a random variable F having F-distribution with (1, d-D)-degrees of freedom.

APPENDIX D

PROOFS OF THE KALMAN FILTER AND THE SEPARATION THEOREM

I INTRODUCTION

Physical systems are designed and built to perform certain defined functions. 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. In other words, the engineer must know the state of the 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 his system, the engineer builds a measuring device and takes measurements or observations on his system. The measurements in general are contaminated with noise. The problem of determining the state of system from noisy measurements is called filtering. It is of central importance in engineering since state estimates are required in the monitoring and for the control of systems. Furthermore a large class of system identification problems can be considered as of filtering (Jazwinski 1971).

More generally, a filtering problem is defined as that of estimating a signal (message) $\underline{x}$ at time t^*, $\underline{x}(t^*)$,

from a set of data collected from starting time t_o to time t, $\{y(\tau),\ t_o \leq \tau \leq t\}$. Depending on the values of t^* and t, the filtering problem is divided into three classes:

- Smoothing or interpolation, if $t^* < t$.
- Filtering, if $t^* = t$.
- Prediction or forecasting or extrapolation, if $t^* > t$.

Kolmogorov (1942) and the father of Cybernetics, Wiener (1949) were credited as the first pioneers in the filtering field for their work known as the optimum linear filter, also called as the Wiener-Kolmogorov filter, or just the Wiener filter. Their work marked the beginning of the engineering awareness of the problem of state estimation and formed an important milestone because not only the results were new and of current interest, but more importantly they cast the problem into a framework which was of practical use at that time, namely, the frequency domain. Unfortunately, because the results were expressed in the frequency domain, they are difficult to be computed (due to spectral factorization described by Equation (D.3)) and they could not be extended to nonstationary problems.

Kalman (1960b) brought the filtering theory into practice by attacking the problem directly in the time domain of which the results are easily computed or implemented. His work, partially earning him the 1974-IEEE's Medal of

Honor, is known as the Wiener-Kalman filter, Kalman-Bucy filter or just Kalman filter. The problem involved is not restricted to be stationary, therefore the Kalman filter is more general. As a matter of fact, it can be proved (Sage and Melsa 1971a) that the Wiener filter is only a particular case of the Kalman filter. Thus from 1960, the optimum linear filter has been referred to as the Kalman filter. The filter, since being developed, has been successfully applied to various fields in engineering, especially in space engineering: satellite orbit determination, aircraft navigation and space flight including the Ranger, Mariner, Apollo missions (Jazwinski 1971). Since early 1970's the Kalman filter has also been applied to social sciences, particularly to economics to estimate the economy of a country and to regulate it using the Separation Theorem (Vishwakarma 1974, Athans and Kendrick 1974).

Filtering theory for nonlinear models, that is, nonlinear filtering theory, was also developed by Stratonovich (1960), Wonham (1963) and Kushner (1964). However, it has not yet been seen fruitfully applied to any significant practical problem.

In this appendix, the Kalman filter will be stated and proved. Also the Separation Theorem, a theorem based on which a linear stochastic optimum control problem can be

treated separately as a Kalman filtering problem and a linear deterministic optimum control problem, will be developed.

II THE OPTIMUM LINEAR FILTER

II.1 The Wiener Filter

Wiener problem

As seen from Figure D.1, the observation vector $\underline{y}(t)$, which is the combination of a signal (message) $\underline{x}(t)$ and a noise $\underline{v}(t)$,

$$\underline{y}(t) = \underline{x}(t) + \underline{v}(t) \qquad (D.1)$$

is passed through a linear time-invariant system (so-called Wiener filter) with matrix transfer function $\underline{T}(s)$. The output of the system is denoted by $\underline{z}(t)$. The Wiener problem is that of selecting the matrix transfer function $\underline{T}(s)$ in such a way that the output $\underline{z}(t)$ is the best estimate of the signal $\underline{x}(t)$ (best in the minimum error-variance sense).

Assumptions

The processes $\{\underline{y}(t)\}$ and $\{\underline{x}(t)\}$ are stationary with known spectral and cross spectral density matrices: $\underline{R}_{xx}(s)$, $\underline{R}_{vv}(s)$, $\underline{R}_{xv}(s)$ and $\underline{R}_{vx}(s)$.

Solution

$$\underline{T}(s) = \left\{\underline{R}_{yx}(s)\left[\underline{D}^{-1}(-s)\right]^T\right\}_{PR} \underline{D}^{-1}(s) \qquad (D.2)$$

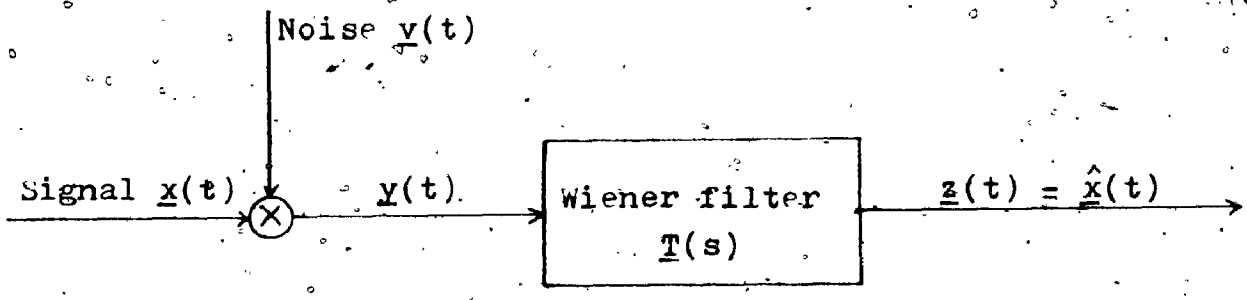


Figure D.1 Schematic of the Wiener problem.

where,

$$\underline{D}(s)\underline{D}^T(s) = \underline{R}_{yy}(s) = \underline{R}_{xx}(s) + \underline{R}_{vv}(s) + \underline{R}_{xv}(s) + \underline{R}_{vx}(s) \qquad (D.3)$$

and such that the determinant $|\underline{D}(s)|$ has all of its poles and zeros in the left half of the s-plane.

$\left\{ \quad \right\}_{PR}$ is the physically realizable portion of $\left\{ \quad \right\}$ and is defined as the Laplace transformation of the positive time portion of the time response of the transformed quantity $\left\{ \quad \right\}$. Note that $\left\{ \quad \right\}$ contains all terms with poles in the left half of the s-plane.

$$\underline{R}_{yx}(s) = \underline{R}_{xx}(s) + \underline{R}_{vx}(s) \qquad (D.4)$$

Proof

For proof one can consult the literature (Wiener 1949, Sage and Melsa 1971a).

II.2 The Kalman Filter

Kalman problem

Figure D.2 is a state space representation of a linear stochastic system. The Kalman problem is that of determining an estimator for the state vector $\underline{x}(t)$, given the observation data up to time t, $\left\{\underline{y}(\tau) \quad t_o \leq \tau \leq t\right\}$. The estimator must acquire the following characteristics:

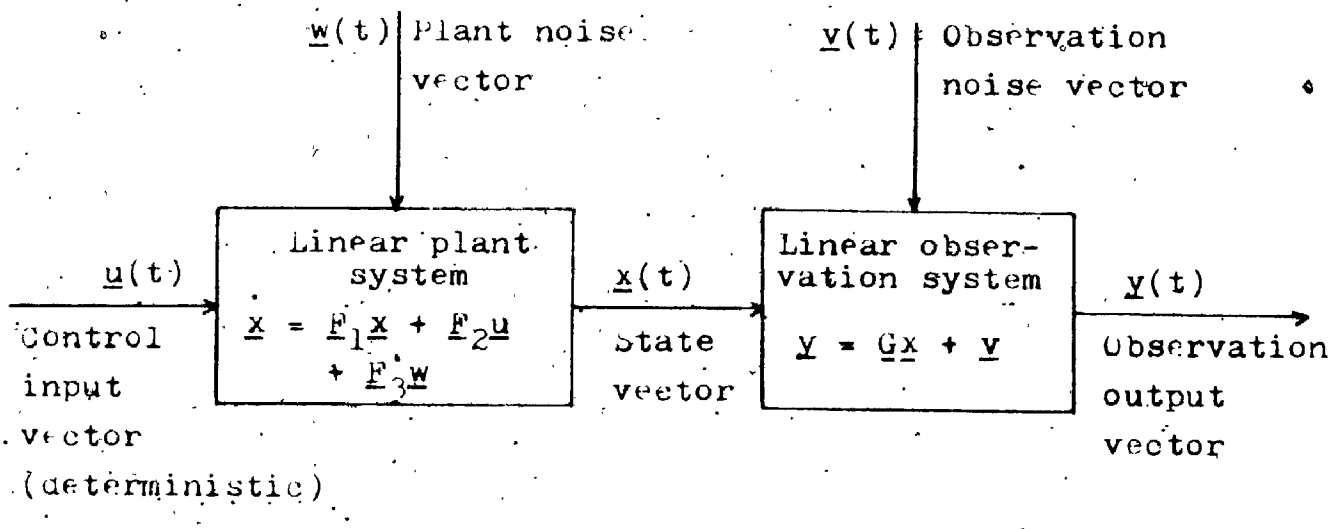


Figure 5.2 State space representation of a linear stochastic system.

(a) Linearity (with respect to the observation data).

(b) Unbiasedness (conditionally and unconditionally).

(c) Minimum error-variance (amongst the class of linear estimator).

Assumptions:

The plant system and the observation system are both linear and of the following form:

$$\dot{\underline{x}}(t) = \underline{F}_1(t)\underline{x}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{F}_3(t)\underline{w}(t) \qquad (D.5)$$

$$\underline{y}(t) = \underline{G}(t)\underline{x}(t) + \underline{v}(t) \qquad (D.6)$$

where $\underline{u}(t)$ is a known deterministic control input vector; $\underline{F}_1(t)$, $\underline{F}_2(t)$, $\underline{F}_3(t)$ and $\underline{G}(t)$ are known deterministic matrix functions, the initial condition $\underline{x}(t_o)$ is a random vector with the first two moments known,

$$E\left\{\underline{x}(t_o)\right\} = \underline{\mu}_{x_o} \qquad (D.7)$$

$$E\left\{\left[\underline{x}(t_o) - \underline{\mu}_{x_o}\right]\left[\underline{x}(t_o) - \underline{\mu}_{x_o}\right]^T\right\} = \underline{P}_o \qquad (D.8)$$

and the noise sources $\left\{\underline{w}(t)\right\}$ and $\left\{\underline{v}(t)\right\}$ are stochastic processes which are white with the following known statistics:

$$E\left\{\underline{w}(t)\right\} = \underline{0}. \qquad (D.9)$$

$$E\left\{\underline{v}(t)\right\} = \underline{0} \qquad (D.10)$$

$$E\left\{\underline{w}(t)\underline{w}^T(\tau)\right\} = \underline{Q}_2(t)\delta_D(t-\tau) \qquad (D.11)$$

$$E\left\{\underline{v}(t)\underline{v}^T(\tau)\right\} = \underline{R}_2(t)\delta_D(t-\tau) \qquad (D.12)$$

$$E\left\{\underline{w}(t)\underline{v}^T(\tau)\right\} = \underline{T}(t)\delta_D(t-\tau) \qquad (D.13)$$

$$E\left\{\underline{x}(t_o)\underline{w}^T(t)\right\} = \underline{0} \qquad (D.14)$$

$$E\left\{\underline{x}(t_o)\underline{v}^T(t)\right\} = \underline{0} \qquad (D.15)$$

Solution

The solution of the Kalman problem was given in Chapter V by Equations (5.19) to (5.21), or rewritten here as Equations (D.16) to (D.18),

$$\dot{\hat{\underline{x}}}(t) = \underline{F}_1(t)\hat{\underline{x}}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{K}_2(t)\left[\underline{y}(t) - \underline{G}(t)\hat{\underline{x}}(t)\right] \qquad (D.16)$$

Initial condition: $\hat{\underline{x}}(t_o) = \underline{\gamma}_{x_o}$

$$\underline{K}_2(t) = \left[\underline{P}_2(t)\underline{G}^T(t) + \underline{F}_3(t)\underline{T}(t)\right]\underline{R}_2^{-1}(t) \qquad (D.17)$$

= Kalman filter gain matrix

$$\dot{\underline{P}}_2(t) = \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) - \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \qquad (D.18)$$

Initial condition: $\underline{P}_2(t_o) = \underline{P}_o$

Proof

Kalman himself first solved the problem by using the Orthogonal Projection Lemma in the Hilbert space for the discrete case (Kalman 1960b), and the Wiener-Hopf matrix integral equation, a version of the Orthogonal Projection Lemma, for the continuous case (Kalman and Bucy 1961). The following is the proof of the Kalman filter using Calculus of Variations. It is based upon, but is more general than the proof given by Athans and Tse (1967).

Taking the expectations of Equations (D.5) and

(D.6) results in:

$$E\{\dot{\underline{x}}(t)\} = \underline{F}_1(t)E\{\underline{x}(t)\} + \underline{F}_2(t)\underline{u}(t) \qquad (D.19)$$

$$E\{\underline{y}(t)\} = \underline{G}(t)E\{\underline{x}(t)\} \qquad (D.20)$$

Let

$$\underline{p}(t) = \underline{x}(t) - E\{\underline{x}(t)\} \qquad (D.21)$$

$$\underline{z}(t) = \underline{y}(t) - E\{\underline{y}(t)\} \qquad (D.22)$$

then from equations (D.5), (D.6), (D.19) and (D.20) the new equations representing the system are:

$$\dot{\underline{p}}(t) = \underline{F}_1(t)\underline{p}(t) + \underline{F}_3(t)\underline{w}(t) \qquad (D.23)$$

$$\underline{z}(t) = \underline{G}(t)\underline{p}(t) + \underline{v}(t) \qquad (D.24)$$

Due to the linearity characteristics (a), the estimate $\hat{\underline{x}}(t)$ of the state vector $\underline{x}(t)$ must be in the form of a linear differential equation which is also linear in the observation vector $\underline{y}(t)$. From Equations (D.21) and (D.22), the linearity characteristics (a) is equivalent to the following equation:

$$\dot{\hat{\underline{p}}}(t) = \underline{A}(t)\hat{\underline{p}}(t) + \underline{K}_2(t)\underline{z}(t) \qquad (D.25)$$

where

$$\hat{\underline{p}}(t) = \dot{\underline{x}}(t) - E\{\underline{x}(t)\} \qquad (D.26)$$

$\underline{A}(t)$ and $\underline{K}_2(t)$ are unknown matrix functions yet to be found.

Let the error of the estimate $\hat{\underline{p}}(t)$ be $\tilde{\underline{p}}(t)$,

$$\tilde{\underline{p}}(t) = \underline{p}(t) - \hat{\underline{p}}(t) \qquad (D.27)$$

then from Equations (D.23), (D.25) and (D.24),

$$\dot{\hat{\underline{p}}}(t) = \underline{F}_1(t)\underline{p}(t) + \underline{F}_3(t)\underline{w}(t) - \underline{A}(t)\hat{\underline{p}}(t) - \underline{K}_2(t)\underline{z}(t)$$
$$= \underline{F}_1(t)\underline{p}(t) + \underline{F}_3(t)\underline{w}(t) - \underline{A}(t)\left[\underline{p}(t) - \tilde{\underline{p}}(t)\right]$$
$$- \underline{K}_2(t)\left[\underline{G}(t)\underline{p}(t) + \underline{v}(t)\right]$$

or,

$$\dot{\tilde{\underline{p}}}(t) = \left[\underline{F}_1(t) - \underline{A}(t) - \underline{K}_2(t)\underline{G}(t)\right]\underline{p}(t) + \underline{A}(t)\tilde{\underline{p}}(t)$$
$$+ \underline{F}_3(t)\underline{w}(t) - \underline{K}_2(t)\underline{v}(t) \qquad (D.28)$$

Due to the unbiasedness characteristics (b), and from the definition of $\underline{p}(t)$ in Equation (D.21), the expected values of both $\dot{\tilde{\underline{p}}}(t)$ and $\tilde{\underline{p}}(t)$, conditioned to $\left\{\underline{y}(\tau) \quad t_o \leq \tau \leq t\right\}$, must be zero. Thus taking the conditional expectation of Equation (D.28) gives:

$$\underline{0} = \left[\underline{F}_1(t) - \underline{A}(t) - \underline{K}_2(t)\underline{G}(t)\right] E\left\{\underline{p}(t)/\underline{y}(\tau) \quad t_o \leq \tau \leq t\right\} \qquad (D.29)$$

because the expectations of $\underline{w}(t)$ and $\underline{v}(t)$ are also zero. Since one cannot assume $E\left\{\underline{p}(t)/\underline{y}(\tau) \quad t_o \leq \tau \leq t\right\}$ is zero for all time t, Equation (D.29) can be satisfied only if

$$\underline{F}_1(t) - \underline{A}(t) - \underline{K}_2(t)\underline{G}(t) = \underline{0}$$

or,

$$\underline{A}(t) = \underline{F}_1(t) - \underline{K}_2(t)\underline{G}(t) \qquad (D.30)$$

so that the filter equation (D.25) now takes the following form:

$$\dot{\hat{\underline{p}}}(t) = \left[\underline{F}_1(t) - \underline{K}_2(t)\underline{G}(t)\right]\hat{\underline{p}}(t) + \underline{K}_2(t)\underline{z}(t) \qquad (D.31)$$

Also due to the unbiasedness characteristics (b), $E\{\hat{\underline{x}}(t)\}$ must be equal to $E\{\underline{x}(t)\}$ for all $t \geq 0$. They must surely be equal for $t = t_o$. Therefore,

$$\hat{\underline{x}}(t_o) = E\{\underline{x}(t_o)\} = \underline{\mu}_{x_o} \tag{D.32}$$

or from Equation (D.26),

$$\hat{\underline{p}}(t_o) = \underline{0} \tag{D.33}$$

The remainder of the proof, as seen from Equation (D.31), is to determine the matrix function $\underline{h}_2(t)$ from the last required characteristics (c). That is, one wishes to select the matrix function $\underline{h}_2(t)$ on the observation interval $t_o \leq t \leq \tau$ such that the following scalar function,

$$J' = E\left\{\left[\underline{x}(\tau) - \hat{\underline{x}}(\tau)\right]^T\left[\underline{x}(\tau) - \hat{\underline{x}}(\tau)\right]\right\} \tag{D.34}$$

is minimized.

Directly from the definition of $\underline{p}(t)$ in Equation (D.21), minimizing J' is the same as minimizing J, where,

$$J = E\left\{\tilde{\underline{p}}^T(\tau)\tilde{\underline{p}}(\tau)\right\}$$

or,

$$J = E\left\{\mathrm{tr}\left[\tilde{\underline{p}}(\tau)\tilde{\underline{p}}^T(\tau)\right]\right\} \tag{D.35}$$

Since the trace and expectation operators are both linear, they can be interchanged to give,

$$J = \mathrm{tr}\left\{E\left\{\tilde{\underline{p}}(\tau)\tilde{\underline{p}}^T(\tau)\right\}\right\} = \mathrm{tr}\left\{\underline{P}_2(\tau)\right\} \tag{D.36}$$

where,

$$\underline{P}_2(\tau) = E\left\{\tilde{\underline{p}}(\tau)\tilde{\underline{p}}^T(\tau)\right\} \qquad (D.37)$$

Before minimizing J defined by Equation (D.36), one must first find out how $\underline{k}_2(t)$ and $\underline{P}_2(t)$ are related. This can be done as follows. Substituting Equation (D.30) to Equation (D.28) results in Equation (D.38),

$$\dot{\tilde{\underline{p}}}(t) = \left[\underline{F}_1(t) - \underline{k}_2(t)\underline{G}(t)\right]\tilde{\underline{p}}(t) + \underline{F}_3(t)\underline{w}(t) - \underline{k}_2(t)\underline{v}(t) \qquad (D.38)$$

the solution of which is:

$$\tilde{\underline{p}}(t) = \underline{\Phi}(t,t_o)\tilde{\underline{p}}(t_o) + \int_{t_o}^{t}\underline{\Phi}(t,\gamma)\left[\underline{F}_3(\gamma)\underline{w}(\gamma) - \underline{k}_2(\gamma)\underline{v}(\gamma)\right]d\gamma \qquad (D.39)$$

where $\underline{\Phi}(t,t_o)$ is the state transition matrix associated with the linear differential equation (D.38) and is defined by equation (D.40),

$$\dot{\underline{\Phi}}(t,t_o) = \left[\underline{F}_1(t) - \underline{k}_2(t)\underline{G}(t)\right]\underline{\Phi}(t,t_o) \qquad (D.40)$$

Initial condition: $\underline{\Phi}(t_o,t_o) = \underline{I}$ = Identity matrix

Thus,

$$\underline{P}_2(t) = E\left\{\tilde{\underline{p}}(t)\tilde{\underline{p}}^T(t)\right\}$$

$$= \underline{\Phi}(t,t_o)E\left\{\tilde{\underline{p}}(t_o)\tilde{\underline{p}}^T(t_o)\right\}\underline{\Phi}^T(t,t_o) + \int_{t_o}^{t}\int_{t_o}^{t}\underline{\Phi}(t,\gamma_1)$$

$$\left\{\underline{F}_3(\gamma_1)E\left\{\underline{w}(\gamma_1)\underline{w}^T(\gamma_2)\right\}\underline{F}_3^T(\gamma_2) + \underline{k}_2(\gamma_1)E\left\{\underline{v}(\gamma_1)\underline{v}^T(\gamma_2)\right\}\underline{k}_2^T(\gamma_2) - \underline{F}_3(\gamma_1)E\left\{\underline{w}(\gamma_1)\underline{v}^T(\gamma_2)\right\}\underline{k}_2^T(\gamma_2) - \underline{k}_2(\gamma_1)E\left\{\underline{v}(\gamma_1)\underline{w}^T(\gamma_2)\right\}\underline{F}_3^T(\gamma_2)\right\}\underline{\Phi}^T(t,\gamma_2)\,d\gamma_1\,d\gamma_2 \qquad (D.41)$$

or, using the statistics defined in Equations (D.11) to (D.13),

$$\underline{P}_2(t) = \underline{\Phi}(t,t_o)\underline{P}_2(t_o)\underline{\Phi}^T(t,t_o) + \int_{t_o}^{t} \underline{\Phi}(t,\gamma)\underline{M}(\gamma)\underline{\Phi}^T(t,\gamma)d\gamma \qquad (D.42)$$

where $\underline{M}(\gamma)$ is defined by Equation (D.43),

$$\underline{M}(t) = \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) + \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) - \underline{F}_3(t)\underline{T}(t)\underline{K}_2^T(t) - \underline{K}_2(t)\underline{T}^T(t)\underline{F}_3(t) \qquad (D.43)$$

Taking the derivative of Equation (D.42) with respect to time and using Leibnitz's Rule (Greenberg 1971) results in:

$$\dot{\underline{P}}_2(t) = \dot{\underline{\Phi}}(t,t_o)\underline{P}_2(t_o)\underline{\Phi}^T(t,t_o) + \underline{\Phi}(t,t_o)\underline{P}_2(t_o)\dot{\underline{\Phi}}^T(t,t_o) + \int_{t_o}^{t} \dot{\underline{\Phi}}(t,\gamma)\underline{M}(\gamma)\underline{\Phi}^T(t,\gamma)d\gamma + \int_{t_o}^{t} \underline{\Phi}(t,\gamma)\underline{M}(\gamma)\dot{\underline{\Phi}}^T(t,\gamma)d\gamma + \underline{M}(t) - \underline{Q} \qquad (D.44)$$

Substitution of Equation (D.40) to Equation (D.44) gives:

$$\dot{\underline{P}}_2(t) = [\underline{F}_1(t) - \underline{K}_2(t)\underline{G}(t)]\left\{\underline{\Phi}(t,t_o)\underline{P}_2(t_o)\underline{\Phi}^T(t,t_o) + \int_{t_o}^{t} \underline{\Phi}(t,\gamma)\underline{M}(\gamma)\underline{\Phi}^T(t,\gamma)d\gamma\right\} + \left\{\underline{\Phi}(t,t_o)\underline{P}_2(t_o)\underline{\Phi}^T(t,t_o) + \int_{t_o}^{t} \underline{\Phi}(t,\gamma)\underline{M}(\gamma)\underline{\Phi}^T(t,\gamma)d\gamma\right\}[\underline{F}_1(t)$$

$$- \underline{k}_2(t)\underline{G}(t)\Big]^T + \underline{M}(t) \qquad (D.45)$$

The quantity in the bracket $\{\quad\}$ is $\underline{P}_2(t)$ as seen from Equation (D.42). Thus,

$$\underline{\dot{P}}_2(t) = \Big[\underline{F}_1(t) - \underline{k}_2(t)\underline{G}(t)\Big]\underline{P}_2(t) + \underline{P}_2(t)\Big[\underline{F}_1(t) - \underline{k}_2(t)\underline{G}(t)\Big]^T + \underline{M}(t) \qquad (D.46)$$

and the initial condition of Equation (D.46) is found as:

$$\begin{aligned}\underline{P}_2(t_o) &= E\Big\{\underline{\tilde{p}}(t_o)\underline{\tilde{p}}^T(t_o)\Big\} \\ &= E\Big\{\Big[\underline{p}(t_o) - \underline{\hat{p}}(t_o)\Big]\Big[\underline{p}(t_o) - \underline{\hat{p}}(t_o)\Big]^T\Big\} \\ &= E\Big\{\Big[\underline{x}(t_o) - \underline{\hat{x}}(t_o)\Big]\Big[\underline{x}(t_o) - \underline{\hat{x}}(t_o)\Big]^T\Big\} \\ &= E\Big\{\Big[\underline{x}(t_o) - \underline{M}_{x_o}\Big]\Big[\underline{x}(t_o) - \underline{M}_{x_o}\Big]^T\Big\}\end{aligned}$$

or,

$$\underline{P}_2(t_o) = \underline{P}_o \qquad (D.47)$$

Hence the problem yet to be solved is to minimize the functional $J = \mathrm{tr}\{\underline{P}_2(\tau)\}$ with respect to $\underline{k}_2(t)$ $t_o \leq t \leq \tau$ and subject to Equations (D.46) and (D.43). It is a linear optimum control problem with specified initial value (Equation (D.47)), free terminal value and fixed terminal time, that is, a Mayer-type variational problem. From the matrix form of Pontryagin's Maximum Principle which was developed by Athans (1968), let a scalar function H, called the Hamiltonian, be defined as:

$$H = \mathrm{tr}\{\underline{\dot{P}}_2(t)\underline{\Lambda}^T(t)\} \qquad (D.48)$$

where $\underline{\Lambda}(t)$ is called as an adjoint matrix or Lagrange multi-

plier matrix. Then, the necessary conditions which the matrix $\underline{K}_2(t)$ must satisfy are:

$$\dot{\underline{\Lambda}}(t) = -\frac{\partial H}{\partial \underline{P}_2(t)} \tag{D.49}$$

$$\dot{\underline{P}}_2(t) = \frac{\partial H}{\partial \underline{\Lambda}(t)} \tag{D.50}$$

$$\frac{\partial H}{\partial \underline{K}_2(t)} = \underline{0} \tag{D.51}$$

and the transversality (boundary) conditions are:

$$\underline{\Lambda}(\tau) = \frac{\partial \mathrm{tr}\{\underline{P}_2(\tau)\}}{\partial \underline{P}_2(\tau)} = \underline{I} \tag{D.52}$$

$$\underline{P}_2(t_o) = \underline{P}_o \tag{D.53}$$

Note that Equation (D.50) is a rewritten form of Equation (D.46). Carrying the derivatives, Equations (D.49) and (D.51) become:

$$\dot{\underline{\Lambda}}(t) = -[\underline{F}_1(t) - \underline{K}_2(t)\underline{G}(t)]^T\underline{\Lambda}(t) - \underline{\Lambda}(t)[\underline{F}_1(t) - \underline{K}_2(t)\underline{G}(t)] \tag{D.54}$$

$$-\underline{\Lambda}(t)\underline{P}_2(t)\underline{G}^T(t) - \underline{\Lambda}^T(t)\underline{P}_2(t)\underline{G}^T(t) + \underline{\Lambda}^T(t)\underline{K}_2(t)\underline{R}_2(t) + \underline{\Lambda}(t)\underline{K}_2(t)\underline{R}_2(t) - \underline{\Lambda}^T(t)\underline{F}_3(t)\underline{T}(t) - \underline{\Lambda}(t)\underline{F}_3(t)\underline{T}(t) = \underline{0} \tag{D.55}$$

From Equation (D.54), one notices that if $\underline{\Lambda}(t)$ is symmetric for some value of t in the interval (t_o,τ), then $\underline{\Lambda}(t)$ is symmetric for all t in the interval (t_o,τ). Since $\underline{\Lambda}(t)$ is symmetric at $t = \tau$ as seen from Equation (D.52), $\underline{\Lambda}(t)$

is symmetric for all t in the interval (t_o,τ). Also from the linear differential equation (D.54) and from the boundary condition, Equation (D.53), $\underline{\Delta}(t)$ is also nonsingular for all t in the interval (t_o,τ). Using these two properties of $\underline{\Delta}(t)$ to Equation (D.55), the desired matrix $\underline{K}_2(t)$ is found as:

$$\underline{K}_2(t) = \left[\underline{P}_2(t)\underline{G}^T(t) + \underline{F}_3(t)\underline{T}(t)\right]\underline{R}_2^{-1}(t) \qquad (D.56)$$

Thus Equation (D.46) can be reduced to Equation (D.57),

$$\begin{aligned}
\dot{\underline{P}}_2(t) &= \underline{F}_1(t)\underline{P}_2(t) - \underline{K}_2(t)\underline{G}(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) \\
&\quad - \underline{P}_2(t)\underline{G}^T(t)\underline{K}_2^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) + \underline{K}_2(t) \\
&\quad \underline{R}_2(t)\underline{K}_2^T(t) + \underline{F}_3(t)\underline{T}(t)\underline{K}_2^T(t) - \underline{K}_2(t)\underline{T}^T(t)\underline{F}_3^T(t) \\
&= \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \\
&\quad + \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) - \underline{K}_2(t)\left[\underline{G}(t)\underline{P}_2(t) + \underline{T}^T(t)\underline{F}_3^T(t)\right] \\
&\quad - \left[\underline{P}_2(t)\underline{G}^T(t) + \underline{F}_3(t)\underline{T}(t)\right]\underline{K}_2^T(t) \\
&= \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \\
&\quad + \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) - \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) \\
&\quad - \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t)
\end{aligned}$$

or,

$$\begin{aligned}
\dot{\underline{P}}_2(t) &= \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \\
&\quad - \underline{K}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) \qquad (D.57)
\end{aligned}$$

Equation (D.56) is the same as Equation (D.17) and Equation (D.57) coupled with Equation (D.47) is the same as Equation (D.18). Thus the proof is completed if the validity

of Equation (D.16) is justified. Adding Equations (D.31) and (D.19) together gives:

$$\dot{\hat{\underline{p}}}(t) + E\{\dot{\underline{x}}(t)\} = \underline{F}_1(t)\left[\hat{\underline{p}}(t) + E\{\underline{x}(t)\}\right] + \underline{F}_2(t)\underline{u}(t) - \underline{K}_2(t)\underline{G}(t)\hat{\underline{p}}(t) + \underline{K}_2(t)\underline{z}(t)$$

or from Equations (D.26), (D.22) and (D.20),

$$\dot{\hat{\underline{x}}}(t) = \underline{F}_1(t)\hat{\underline{x}}(t) + \underline{F}_2(t)\underline{u}(t) - \underline{K}_2(t)\underline{G}(t)\left[\hat{\underline{x}}(t) - E\{\underline{x}(t)\}\right] + \underline{K}_2(t)\left[\underline{y}(t) - \underline{G}(t)E\{\underline{x}(t)\}\right]$$

or,

$$\dot{\hat{\underline{x}}}(t) = \underline{F}_1(t)\hat{\underline{x}}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{K}_2(t)\left[\underline{y}(t) - \underline{G}(t)\hat{\underline{x}}(t)\right]$$

which is Equation (D.16).

Note that if $\underline{x}(t_o)$, $\underline{w}(t)$ and $\underline{v}(t)$ are further assumed to Gaussian, then the filter has the minimum error-variance not only amongst the class of linear estimators, but also amongst the class of nonlinear estimators as well, and then the minimum error-variance estimate $\hat{\underline{x}}(t)$ is equivalent to the conditional expectation of $\underline{x}(t)$,

$$\hat{\underline{x}}(t) = E\left\{\underline{x}(t)/\underline{y}(\tau) \quad t_o \leq \tau \leq t\right\} = \hat{\underline{x}}(t/t) \qquad (D.58)$$

III THE SEPARATION THEOREM

The history and statement of the Separation Theorem were given in Chapter V for continuous problems. The proof of the theorem is given here. To ease mathematical

complication, the proof for discrete problems is derived first by using Bellman's Principle of Optimality in Dynamic Programming, then followed by that for continuous problems by letting the sampling-interval approach zero.

III.1 Discrete Problem

For the discrete case, a Linear-Quadratic-Gaussian problem is to select a sequence of physically realizable control vectors $\underline{u}_0, \underline{u}_1, \ldots, \underline{u}_{N-1}$ so that the quadratic performance index (cost function)

$$J = \tfrac{1}{2}E\left\{\underline{x}_N^T\underline{A}_N\underline{x}_N + \sum_{i=0}^{N-1}\left[\underline{x}_i^T \;\vdots\; \underline{u}_i^T\right]\left[\begin{array}{c|c}\underline{A}_i & \underline{M}_i \\ \hline \underline{M}_i^T & \underline{B}_i\end{array}\right]\left[\begin{array}{c}\underline{x}_i \\ \hline \underline{u}_i\end{array}\right]\right\} \qquad (D.59)$$

of the linear system,

$$\underline{x}_{k+1} = \underline{\Phi}_k\underline{x}_k + \underline{\Gamma}_k\underline{u}_k + \underline{\Sigma}_k\underline{\eta}_k \qquad (D.60)$$

$$\underline{y}_k = \underline{G}_k\underline{x}_k + \underline{\varepsilon}_k \qquad (D.61)$$

is a minimum.

The state transition matrix $\underline{\Phi}_k$, the control transition matrix $\underline{\Gamma}_k$, the disturbance transition matrix $\underline{\Sigma}_k$ and the measurement matrix $\underline{G}_k$ are assumed known for all k. The weighting matrices in Equation (D.59) are also known and such that the following matrices for all k are symmetric and positive definite:

$$\underline{A}_k \cdot \underline{B}_k \cdot \left[\begin{array}{c|c} \underline{A}_k & \underline{M}_k \\ \hline \underline{M}_k^T & \underline{B}_k \end{array}\right]$$

Further assumptions are that the initial state vector is a Gaussian variable with known statistics,

$$E\{\underline{x}_o\} = \underline{\mu}_{x_o} \tag{D.62}$$

$$E\left\{\left[\underline{x}_o - \underline{\mu}_{x_o}\right]\left[\underline{x}_o - \underline{\mu}_{x_o}\right]^T\right\} = \underline{P}_o \tag{D.63}$$

and the noise sequences $\{\underline{\eta}_k\}$ and $\{\underline{\varepsilon}_k\}$ are white and Gaussian with known statistics and are independent of $\underline{x}_o$.

$$E\{\underline{\eta}_k\} = \underline{0} \tag{D.64}$$

$$E\{\underline{\varepsilon}_k\} = \underline{0} \tag{D.65}$$

$$E\left\{\left[\begin{array}{c} \underline{\eta}_k \\ \hline \underline{\varepsilon}_k \end{array}\right]\left[\underline{\eta}_i^T \;\middle|\; \underline{\varepsilon}_i^T\right]\right\} = \left[\begin{array}{c|c} \eta\underline{\Omega}_k & \eta\varepsilon\underline{\Omega}_k \\ \hline \eta\varepsilon\underline{\Omega}_k^T & \varepsilon\underline{\Omega}_k \end{array}\right]\delta_K(i-k) \tag{D.66}$$

$$E\{\underline{\eta}_k \underline{x}_o^T\} = \underline{0} \tag{D.67}$$

$$E\{\underline{\varepsilon}_k \underline{x}_o^T\} = \underline{0} \tag{D.68}$$

where $\delta_k(t)$ is the Kronecker delta function,

$$\delta_k(t) = \begin{cases} 1 & \text{for } t = 0 \\ 0 & \text{otherwise} \end{cases} \tag{D.69}$$

Note that a control vector $\underline{u}_k$ is called physically realizable if it is a function of only the given parameters in the problem and of data up to the k-measurement, $\underline{Y}_k$ (Meditch 1969),

$$\underline{Y}_k = \left[\underline{y}_o \;\middle|\; \underline{y}_1 \;\middle|\; \cdots \;\middle|\; \underline{y}_k\right] \tag{D.70}$$

The solution of the Linear-Quadratic-Gaussian problem, according to the Separation Theorem, can be obtained by solving separately the following two problems:

- Optimum state estimation problem or the Kalman filtering problem.

- Deterministic optimum control problem which is the same as the Linear-Quadratic-Gaussian problem except that in Equation (D.60) and (D.61) the noise vectors $\underline{\eta}_k$ and $\underline{\xi}_k$ are dropped and the state vector $\underline{x}_k$ is replaced by its estimate, the solution of the optimum state estimation problem.

The result is,

$$\underline{u}_i = {}_1\underline{K}_i \hat{\underline{x}}_i \tag{D.71}$$

where,

$${}_1\underline{K}_i = -(\underline{\Gamma}_i^T \underline{S}_{i+1} \underline{\Gamma}_i + \underline{B}_i)^{-1}(\underline{\Gamma}_i^T \underline{S}_{i+1} \underline{\Phi}_i + \underline{M}_i^T) \tag{D.72}$$

= Kalman control gain matrix.

$$\underline{S}_i = \underline{\Phi}_i^T \underline{S}_{i+1} \underline{\Phi}_i - {}_1\underline{K}_i^T (\underline{\Gamma}_i^T \underline{S}_{i+1} \underline{\Gamma}_i + \underline{B}_i) {}_1\underline{K}_i + \underline{A}_i \tag{D.73}$$

Final condition: $\underline{S}_N = \underline{A}_N$

$$\hat{\underline{x}}_i = E\{\underline{x}_i / \underline{Y}_i\} \tag{D.74}$$

= Conditional expectation of $\underline{x}_i$

$\hat{\underline{x}}_i$ is also called the discrete Kalman filter estimate of $\underline{x}_i$ and can be found by either discretizing the continuous Kalman filter shown earlier or following the same derivation of

the continuous Kalman filter for discrete-time case,

$$\hat{\underline{x}}_{i+1} = \hat{\underline{x}}_{i+1/i} - \underline{K}_{i+1}(\underline{y}_{i+1} - \underline{G}_{i+1}\hat{\underline{x}}_{i+1/i}) \qquad (D.75)$$

Initial condition: $\hat{\underline{x}}_0 = \underline{\mu}_{x_0}$

$$\hat{\underline{x}}_{i+1/i} = \underline{\Phi}_i\hat{\underline{x}}_i + \underline{\Gamma}_i\underline{u}_i + \underline{\Sigma}_{i\ \eta\varepsilon}\underline{\Omega}_{i\ \varepsilon}\underline{\Omega}_i^{-1}(\underline{y}_i - \underline{G}_i\hat{\underline{x}}_i) \qquad (D.76)$$

$$\underline{K}_i = \underline{C}_i\underline{G}_i^T(\underline{G}_i\underline{C}_i\underline{G}_i^T + {}_\varepsilon\underline{\Omega}_i)^{-1} \qquad (D.77)$$

= Kalman filter gain matrix

$$\underline{C}_{i+1} = (\underline{\Phi}_i - \underline{\Sigma}_{i\ \eta\varepsilon}\underline{\Omega}_{i\ \varepsilon}\underline{\Omega}_i^{-1}\underline{G}_i)\underline{\Lambda}_i(\underline{\Phi}_i - \underline{\Sigma}_{i\ \eta\varepsilon}\underline{\Omega}_{i\ \varepsilon}\underline{\Omega}_i^{-1}\underline{G}_i)^T + \underline{\Sigma}_{i\ \eta}\underline{\Omega}_i\underline{\Sigma}_i^T - \underline{\Sigma}_{i\ \eta\varepsilon}\underline{\Omega}_{i\ \varepsilon}\underline{\Omega}_i^{-1}{}_{\eta\varepsilon}\underline{\Omega}_i^T\underline{\Sigma}_i^T \qquad (D.78)$$

$$\underline{\Lambda}_{i+1} = (\underline{I} - \underline{K}_{i+1}\underline{G}_{i+1})\underline{C}_{i+1} \qquad (D.79)$$

Initial condition: $\underline{\Lambda}_0 = \underline{P}_0$

The validity of Equations (D.70) to (D.72) is shown as follows. Let

$$V_j = \min_{\underline{u}_{N-j}} \cdots \min_{\underline{u}_{N-1}} \tfrac{1}{2}E\left\{\underline{x}_N^T\underline{A}_N\underline{x}_N + \sum_{i=N-j}^{N-1}\left[\underline{x}_i^T \mid \underline{u}_i^T\right]\left[\begin{array}{c|c}\underline{A}_i & \underline{M}_i \\ \hline \underline{M}_i^T & \underline{B}_i\end{array}\right]\left[\begin{array}{c}\underline{x}_i \\ \hline \underline{u}_i\end{array}\right]\right\} \qquad (D.80)$$

then V_N is the minimum of the performance index J. From the Principle of Optimality in Dynamic Programming (Bellman 1957, Meditch 1969), which states that "An optimum control has the property that whatever the initial state and initial control are, the remaining control must constitute an optimum one with regard to the state which results from initial control" the optimum sequence $\underline{u}_0, \underline{u}_1, \ldots, \underline{u}_{N-1}$ can be obtained by

solving the following problems for $\underline{u}_{N-1}$, then $\underline{u}_{N-2}$, ..., $\underline{u}_0$ respectively:

$$V_1 = \min_{\underline{u}_{N-1}} E\left\{\tfrac{1}{2}\underline{x}_N^T \underline{A}_N \underline{x}_N + \tfrac{1}{2}\begin{bmatrix}\underline{x}_{N-1}^T & \underline{u}_{N-1}^T\end{bmatrix}\begin{bmatrix}\underline{A}_{N-1} & \underline{N}_{N-1} \\ \underline{M}_{N-1}^T & \underline{B}_{N-1}\end{bmatrix}\begin{bmatrix}\underline{x}_{N-1} \\ \underline{u}_{N-1}\end{bmatrix}\right\} \tag{D.81}$$

$$V_2 = \min_{\underline{u}_{N-2}} E\left\{V_1 + \tfrac{1}{2}\begin{bmatrix}\underline{x}_{N-2}^T & \underline{u}_{N-2}^T\end{bmatrix}\begin{bmatrix}\underline{A}_{N-2} & \underline{N}_{N-2} \\ \underline{M}_{N-2}^T & \underline{B}_{N-2}\end{bmatrix}\begin{bmatrix}\underline{x}_{N-2} \\ \underline{u}_{N-2}\end{bmatrix}\right\} \tag{D.82}$$

$$V_k = \min_{\underline{u}_{N-k}} E\left\{V_{k-1} + \tfrac{1}{2}\begin{bmatrix}\underline{x}_{N-k}^T & \underline{u}_{N-k}^T\end{bmatrix}\begin{bmatrix}\underline{A}_{N-k} & \underline{N}_{N-k} \\ \underline{M}_{N-k}^T & \underline{B}_{N-k}\end{bmatrix}\begin{bmatrix}\underline{x}_{N-k} \\ \underline{u}_{N-k}\end{bmatrix}\right\} \tag{D.83}$$

$$V_N = \min_{\underline{u}_0} E\left\{V_{N-1} + \tfrac{1}{2}\begin{bmatrix}\underline{x}_0^T & \underline{u}_0^T\end{bmatrix}\begin{bmatrix}\underline{A}_0 & \underline{N}_0 \\ \underline{M}_0^T & \underline{B}_0\end{bmatrix}\begin{bmatrix}\underline{x}_0 \\ \underline{u}_0\end{bmatrix}\right\} \tag{D.84}$$

Now it is attempted to find $\underline{u}_{N-1}$ from Equation (D.81). Combining Equations (D.60) and (D.81) gives:

$$V_1 = \min_{\underline{u}_{N-1}} \tfrac{1}{2}E\left\{\underline{x}_{N-1}^T(\underline{\Phi}_{N-1}^T \underline{A}_N \underline{\Phi}_{N-1} + \underline{A}_{N-1})\underline{x}_{N-1} + \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T \underline{A}_N \underline{\Gamma}_{N-1}\right.$$

$$+ \underline{B}_{N-1})\underline{u}_{N-1} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{x}_{N-1}^T\underline{\Phi}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}$$
$$\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Phi}_{N-1}$$
$$+ \underline{M}_{N-1}^T)\underline{x}_{N-1}\Big\} \qquad (D.85)$$

By definition $\underline{x}_0$ is statistically independent of $\underline{\eta}_i$ for all i, thus $\underline{x}_i$ is also statistically independent of $\underline{\eta}_i$ for all i as can be seen from equation (D.80). Therefore the 4th term in equation (D.85) is zero because $E\{\underline{\eta}_{N-1}\} = \underline{0}$ and equation (D.84) reduces to:

$$V_1 = \underset{\underline{u}_{N-1}}{\text{Min.}} \tfrac{1}{2}E\Big\{\underline{x}_{N-1}^T(\underline{\Phi}_{N-1}^T\underline{\Delta}_N\underline{\Phi}_{N-1} + \underline{\Delta}_{N-1})\underline{x}_{N-1} + \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Gamma}_{N-1}$$
$$+ \underline{B}_{N-1})\underline{u}_{N-1} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{\Delta}_N$$
$$\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\underline{x}_{N-1}\Big\} \qquad (D.86)$$

Recall from one of the properties of the conditional expectation is that $E\{\underline{z}\} = E\{E(\underline{z}/\underline{\xi})\}$ where the outer expectation is over $\underline{\xi}$. Utilizing this property, equation (D.86) can be written as:

$$V_1 = \underset{\underline{u}_{N-1}}{\text{Min.}} \tfrac{1}{2}E\Big\{E\Big\{\underline{x}_{N-1}^T(\underline{\Phi}_{N-1}^T\underline{\Delta}_N\underline{\Phi}_{N-1} + \underline{\Delta}_{N-1})\underline{x}_{N-1} + \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\Delta}_N$$
$$\underline{\Gamma}_{N-1} + \underline{B}_{N-1})\underline{u}_{N-1} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T$$
$$\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + 2\underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\Delta}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\underline{x}_{N-1}$$
$$/\underline{x}_{N-1}\Big\}\Big\} \qquad (D.87)$$

Thus $\underline{u}_{N-1}$ can be found by just minimizing the inner expected value in equation (D.87). Setting the derivative of the inner expected value to zero and noting the following:

$$\cdot\ E\left\{\underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\underline{x}_{N-1}/\underline{Y}_{N-1}\right\}$$

$$= \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)E\left\{\underline{x}_{N-1}/\underline{Y}_{N-1}\right\}$$

$$- \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\hat{\underline{x}}_{N-1} \qquad (D.88)$$

$$\cdot\ E\left\{\underline{u}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1}/\underline{Y}_{N-1}\right\}$$

$$= \underline{u}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Sigma}_{N-1}E\left\{\underline{\eta}_{N-1}/\underline{Y}_{N-1}\right\} = \underline{0} \qquad (D.89)$$

(because $E\{\underline{\eta}_{N-1}\} = \underline{0}$)

$$\cdot\ E\left\{\underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Gamma}_{N-1} + \underline{B}_{N-1})\underline{u}_{N-1}/\underline{Y}_{N-1}\right\}$$

$$= \underline{u}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Gamma}_{N-1} + \underline{B}_{N-1})\underline{u}_{N-1} \qquad (D.90)$$

results in:

$$(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Gamma}_{N-1} + \underline{B}_{N-1})\underline{u}_{N-1} + (\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\hat{\underline{x}}_{N-1} = \underline{0}$$

or,

$$\underline{u}_{N-1} = -(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Gamma}_{N-1} + \underline{B}_{N-1})^{-1}(\underline{\Gamma}_{N-1}^T\underline{A}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T)\hat{\underline{x}}_{N-1} \qquad (D.91)$$

From Equation (D.91), one can say that Equations (D.71), (D.72) and (D.73) have been proved for i = N-1. That is,

$$\underline{u}_{N-1} = {}_1\underline{K}_{N-1}\hat{\underline{x}}_{N-1} \qquad (D.92)$$

$${}_1\underline{K}_{N-1} = -(\underline{\Gamma}_{N-1}^T\underline{\Sigma}_N\underline{\Gamma}_{N-1} + \underline{B}_{N-1})^{-1}(\underline{\Gamma}_{N-1}^T\underline{\Sigma}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T) \qquad (D.93)$$

$$\underline{\Sigma}_N = \underline{A}_N \qquad (D.94)$$

V_1 is evaluated by substituting the value of $\underline{u}_{N-1}$ in Equa-

tion (D.91) to Equation (D.86),

$$V_1 = \tfrac{1}{2}E\Big\{\underline{x}_{N-1}^T\underline{\pi}_{N-1}\underline{x}_{N-1} + \hat{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\hat{\underline{x}}_{N-1} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T\underline{\nu}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1}$$
$$- 2\hat{\underline{x}}_{N-1}^T\underline{K}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{\nu}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} - 2\hat{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\underline{x}_{N-1}\Big\} \quad (D.95)$$

where,

$$\underline{\pi}_{N-1} = \underline{\Phi}_{N-1}^T\underline{\nu}_N\underline{\Phi}_{N-1} + \underline{\Delta}_{N-1} \quad (D.96)$$

$$\underline{\Delta}_{N-1} = \underline{K}_{N-1}^T(\underline{\Gamma}_{N-1}^T\underline{\nu}_N\underline{\Phi}_{N-1} + \underline{M}_{N-1}^T) \quad (D.97)$$

let

$$\tilde{\underline{x}}_{N-1}^T = \underline{x}_{N-1} - \hat{\underline{x}}_{N-1} \quad (D.98)$$

then,

$$\hat{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\hat{\underline{x}}_{N-1} - 2\hat{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\underline{x}_{N-1}$$
$$= -\underline{x}_{N-1}^T\underline{\Delta}_{N-1}\underline{x}_{N-1} + \tilde{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\tilde{\underline{x}}_{N-1} \quad (D.99)$$

and Equation (D.95) becomes,

$$V_1 = \tfrac{1}{2}E\Big\{\underline{x}_{N-1}^T(\underline{\pi}_{N-1} - \underline{\Delta}_{N-1})\underline{x}_{N-1} + \tilde{\underline{x}}_{N-1}^T\underline{\Delta}_{N-1}\tilde{\underline{x}}_{N-1} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T$$
$$\underline{\nu}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} - 2\hat{\underline{x}}_{N-1}^T\underline{K}_{N-1}^T\underline{\Gamma}_{N-1}^T\underline{\nu}_N\underline{\Sigma}_{N-1}\underline{\eta}_{N-1}\Big\} \quad (D.100)$$

$\underline{u}_{N-1}$ has been computed, now it is attempted to find $\underline{u}_{N-2}$ from Equation (D.87). Combining Equations (D.82), (D.60) and (D.100) gives,

$$V_2 = \min_{\underline{u}_{N-2}} \tfrac{1}{2}E\Big\{\underline{x}_{N-2}^T\Big[\underline{\Phi}_{N-2}^T(\underline{\pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Phi}_{N-2} + \underline{\Delta}_{N-2}\Big]\underline{x}_{N-2}$$
$$+ \underline{u}_{N-2}^T\Big[\underline{\Gamma}_{N-2}^T(\underline{\pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Gamma}_{N-2} + \underline{R}_{N-2}\Big]\underline{u}_{N-2}$$
$$+ \underline{\eta}_{N-2}^T\underline{\Sigma}_{N-2}^T(\underline{\pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Sigma}_{N-2}\underline{\eta}_{N-2} + \underline{\eta}_{N-1}^T\underline{\Sigma}_{N-1}^T\underline{\nu}_N$$

$$\underline{\Sigma}_{N-1}\underline{\eta}_{N-1} + \underline{\tilde{x}}_{N-1}^{T}\underline{\Delta}_{N-1}\underline{\tilde{x}}_{N-1} + 2\underline{u}_{N-2}^{T}\Big[\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Phi}_{N-2} + \underline{M}_{N-2}^{T}\Big]\underline{x}_{N-2} + 2\underline{x}_{N-2}^{T}\underline{\Phi}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Sigma}_{N-2}\underline{\eta}_{N-2} - 2\underline{u}_{N-2}^{T}\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Sigma}_{N-2}\underline{\eta}_{N-2} - 2\underline{\hat{x}}_{N-1}^{T}\underline{x}_{N-1}^{T}\underline{\Gamma}_{N-1}\underline{S}_{N}\underline{\Sigma}_{N-1}\underline{\eta}_{N-1}\Big\} \qquad (D.101)$$

Equation (D.101) is very similar to Equation (D.85), therefore $\underline{u}_{N-2}$ can be found in the same way as $\underline{u}_{N-1}$ was found. Replace $E\{\underline{\alpha}_{N-2}\}$ by $E\{E(\underline{\alpha}_{N-2}/\underline{Y}_{N-2})\}$ where $\underline{\alpha}_{N-2}$ is the quantity in the bracket of Equation (D.101) and note the following facts:

. The $7^{\underline{th}}$ term in Equation (D.101) is zero because $\underline{x}_{N-2}$ and $\underline{\eta}_{N-2}$ are statistically independent of each other and $E\{\underline{\eta}_{N-2}\} = \underline{0}$.

. The $1^{\underline{st}}$, $3^{\underline{rd}}$ and $4^{\underline{th}}$ terms in Equation (D.101) are independent of $\underline{u}_{N-2}$.

. The $5^{\underline{th}}$ term is independent of $\underline{u}_{N-2}$ because $E\{\underline{\tilde{x}}_{N-1}^{T}\underline{\Delta}_{N-1}\underline{\tilde{x}}_{N-1}\} = E\{tr(\underline{\Delta}_{N-1}\underline{\tilde{x}}_{N-1}\underline{\tilde{x}}_{N-1}^{T})\} = tr(\underline{\Delta}_{N-1}E\{\underline{\tilde{x}}_{N-1}\underline{\tilde{x}}_{N-1}^{T}\})$ and $E\{\underline{\tilde{x}}_{N-1}\underline{\tilde{x}}_{N-1}^{T}\}$, from the discrete Kalman filter, is the matrix $\underline{\Delta}_{N-1}$ defined by Equation (D.79) which is independent of $\underline{u}_{N-2}$.

. $E\Big\{\underline{u}_{N-2}^{T}\Big[\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Gamma}_{N-2} + \underline{B}_{N-2}\Big]\underline{u}_{N-2}/\underline{Y}_{N-2}\Big\}$

$$= \underline{u}_{N-2}^{T}\Big[\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Gamma}_{N-2} + \underline{B}_{N-2}\Big]\underline{u}_{N-2} \qquad (D.102)$$

. $E\Big\{\underline{u}_{N-2}^{T}\Big[\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Phi}_{N-2} + \underline{M}_{N-2}^{T}\Big]\underline{x}_{N-2}/\underline{Y}_{N-2}\Big\}$

$$= \underline{u}_{N-2}^{T}\Big[\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Phi}_{N-2} + \underline{M}_{N-2}^{T}\Big]\underline{\hat{x}}_{N-2} \qquad (D.103)$$

. $E\Big\{\underline{u}_{N-2}^{T}\underline{\Gamma}_{N-2}^{T}(\underline{\Pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Sigma}_{N-2}\underline{\eta}_{N-2}/\underline{Y}_{N-2}\Big\}$

$$= \underline{u}_{N-2}^T \underline{\Gamma}_{N-2}^T (\underline{\pi}_{N-1} - \underline{\Delta}_{N-1}) \underline{\Sigma}_{N-2} E\left\{\underline{\eta}_{N-2} / \underline{Y}_{N-2}\right\} = \underline{0} \quad \text{(D.104)}$$

$E\left\{\hat{\underline{x}}_{N-1}^T \; {}_1\underline{K}_{N-1}^T \underline{\Gamma}_{N-1} \underline{S}_N \underline{\Sigma}_{N-1} \underline{\eta}_{N-1} / \underline{Y}_{N-2}\right\}$ is independent of $\underline{u}_{N-2}$ as can be seen from Equations (D.75) to (D.79), (D.60) and (D.61) and by realizing that $E\left\{\underline{u}_{N-2}^T \underline{Z} \underline{\eta}_{N-1} / \underline{Y}_{N-2}\right\} = 0$ for any deterministic matrix $\underline{Z}$.

Then setting the derivative of $E\left\{\underline{\alpha}_{N-2} / \underline{Y}_{N-2}\right\}$ with respect to $\underline{u}_{N-2}$ to zero will result in Equation (D.105),

$$\underline{u}_{N-2} = -\left[\underline{\Gamma}_{N-2}^T (\underline{\pi}_{N-1} - \underline{\Delta}_{N-1}) \underline{\Gamma}_{N-2} + \underline{B}_{N-2}\right]^{-1} \left[\underline{\Gamma}_{N-2}^T (\underline{\pi}_{N-1} - \underline{\Delta}_{N-1}) \underline{\Phi}_{N-2} + \underline{M}_{N-2}^T\right] \hat{\underline{x}}_{N-2} \quad \text{(D.105)}$$

From Equations (105), (D.96) and (D.97), one can say that Equations (D.71), (D.72) and (D.73) have been justified for $i = N-2$. That is,

$$\underline{u}_{N-2} = {}_1\underline{K}_{N-2} \hat{\underline{x}}_{N-2} \quad \text{(D.106)}$$

$${}_1\underline{K}_{N-2} = -\left(\underline{\Gamma}_{N-2}^T \underline{S}_{N-1} \underline{\Gamma}_{N-2} + \underline{B}_{N-2}\right)^{-1} \left(\underline{\Gamma}_{N-2}^T \underline{S}_{N-1} \underline{\Phi}_{N-2} + \underline{M}_{N-2}^T\right) \quad \text{(D.107)}$$

$$\underline{S}_{N-1} = \underline{\Phi}_{N-1}^T \underline{S}_N \underline{\Phi}_{N-1} - {}_1\underline{K}_{N-1} \left(\underline{\Gamma}_{N-1}^T \underline{S}_N \underline{\Gamma}_{N-1} + \underline{B}_{N-1}\right) {}_1\underline{K}_{N-1} + \underline{\Delta}_{N-1} \quad \text{(D.108)}$$

V_2 is evaluated in the same manner as V_1. After some mathematical simplification, the value of V_2 is:

$$V_2 = \tfrac{1}{2} E\left\{\underline{x}_{N-2}^T \underline{S}_{N-2} \underline{x}_{N-2} + \sum_{i=1}^{2} \left(\tilde{\underline{x}}_{N-i}^T \underline{\Delta}_{N-i} \tilde{\underline{x}}_{N-i} + \underline{\eta}_{N-i} \underline{\Sigma}_{N-i}^T \underline{S}_{N-i+1} \underline{\Sigma}_{N-i} \underline{\eta}_{N-i} - 2 \hat{\underline{x}}_{N-i}^T \; {}_1\underline{K}_{N-i}^T \underline{\Gamma}_{N-i}^T \underline{S}_{N-i+1} \underline{\Sigma}_{N-i} \underline{\eta}_{N-i}\right)\right\} \quad \text{(D.109)}$$

where,

$$\underline{\pi}_{N-2} = \underline{\Phi}_{N-2}^T(\underline{\pi}_{N-1} - \underline{\Delta}_{N-1})\underline{\Phi}_{N-2} + \underline{A}_{N-2} \tag{D.110}$$

$$\underline{\Delta}_{N-2} = {}_1\underline{K}_{N-2}^T(\underline{\Gamma}_{N-2}^T\underline{S}_{N-1}\underline{\Phi}_{N-2} + \underline{M}_{N-2}^T) \tag{D.111}$$

$$\underline{S}_{N-i} = \underline{\pi}_{N-i} - \underline{\Delta}_{N-i} \tag{D.112}$$

$$= \underline{\Phi}_{N-i}^T\underline{S}_{N-i+1}\underline{\Phi}_{N-i} - {}_1\underline{K}_{N-i}^T(\underline{\Gamma}_{N-i}^T\underline{S}_{N-i+1}\underline{\Gamma}_{N-i} + \underline{B}_{N-i}){}_1\underline{K}_{N-i}$$
$$+ \underline{A}_{N-i} \qquad \text{for } i = 1, 2$$

Equations (D.71) to (D.73) have been justified for for i = N-1 and for i = N-2. To show that they are valid for all possible i, proof by induction can be used. That is, suppose that they are valid for i = j+1 , then they must also be valid for i = j. If they are valid for i = j+1, then,

$$\underline{u}_{j+1} = {}_1\underline{K}_{j+1}\hat{\underline{x}}_{j+1} \tag{D.113}$$

$${}_1\underline{K}_{j+1} = -(\underline{\Gamma}_{j+1}^T\underline{S}_{j+2}\underline{\Gamma}_{j+1} + \underline{B}_{j+1})^{-1}(\underline{\Gamma}_{j+1}^T\underline{S}_{j+2}\underline{\Gamma}_{j+1} + \underline{M}_{j+1}^T) \tag{D.114}$$

$$\underline{S}_{j+1} = \underline{\Phi}_{j+1}^T\underline{S}_{j+2}\underline{\Phi}_{j+1} - {}_1\underline{K}_{j+1}^T(\underline{\Gamma}_{j+1}^T\underline{S}_{j+2}\underline{\Gamma}_{j+1} + \underline{B}_{j+1})$$
$$- + \underline{A}_{j+1} \tag{D.115}$$

$$\hat{\underline{x}}_{j+1} = E\left\{\underline{x}_{j+1}/\underline{Y}_{j+1}\right\} \tag{D.116}$$

$$V_{N-j-1} = \tfrac{1}{2}E\Big\{\underline{x}_{j+1}^T\underline{S}_{j+1}\underline{x}_{j+1} + \sum_{k=1}^{N-j-1}(\underline{\xi}_{N-k}^T\underline{\Delta}_{N-k}\underline{\xi}_{N-k} + \underline{\eta}_{N-k}^T\underline{\Sigma}_{N-k}$$
$$\underline{S}_{N-k+1}\underline{\Sigma}_{N-k}\underline{\eta}_{N-k} - 2\hat{\underline{x}}_{N-k}^T{}_1\underline{K}_{N-k}^T\underline{\Gamma}_{N-k}^T\underline{S}_{N-k+1}$$
$$\underline{\Sigma}_{N-k}\underline{\eta}_{N-k})\Big\} \tag{D.117}$$

$$\underline{\Delta}_{N-j-1} = {}_1\underline{K}_{N-j-1}^T(\underline{\Gamma}_{N-j-1}^T\underline{S}_{N-j}\underline{\Gamma}_{N-j-1} + \underline{M}_{N-j-1}^T) \tag{D.118}$$

From Equation (D.83), for $k = j$,

$$V_{N-j} = \underset{\underline{u}_j}{\text{Min.}}\ E\left\{V_{N-j-1} - \tfrac{1}{2}\begin{bmatrix}\underline{x}_j^T & \underline{u}_j^T\end{bmatrix}\begin{bmatrix}\underline{A}_j & \underline{M}_j \\ \underline{M}_j^T & \underline{B}_j\end{bmatrix}\begin{bmatrix}\underline{x}_j \\ \underline{u}_j\end{bmatrix}\right\} \quad \text{(D.119)}$$

or combining Equations (D.119), (D.117) and (D.60) and after some simplification, the value of V_{N-j} is:

$$\begin{aligned} V_{N-j} = \underset{\underline{u}_j}{\text{Min.}}\ \tfrac{1}{2}E\Big\{ & \underline{x}_j^T(\underline{\Phi}_j^T\underline{S}_{j+1}\underline{\Phi}_j + \underline{A}_j)\underline{x}_j + \underline{u}_j^T(\underline{\Gamma}_j^T\underline{S}_{j+1}\underline{\Gamma}_j + \underline{B}_j)\underline{u}_j \\ & + \underline{\eta}_j^T\underline{\Sigma}_j^T\underline{S}_{j+1}\underline{\Sigma}_j\underline{\eta}_j + \sum_{k=1}^{N-j-1}(\underline{\eta}_{N-k}^T\underline{\Sigma}_{N-k}^T\underline{S}_{N-k+1}\underline{\Sigma}_{N-k}\underline{\eta}_{N-k} \\ & + \tilde{\underline{x}}_{N-k}^T\underline{A}_{N-k}\tilde{\underline{x}}_{N-k} - 2\hat{\underline{x}}_{N-k}^T\underline{K}_{N-k}^T\underline{\Gamma}_{N-k}^T\underline{S}_{N-k+1}\underline{\Sigma}_{N-k} \\ & \underline{\eta}_{N-k}) + 2\underline{u}_j^T\underline{\Gamma}_j^T\underline{S}_{j+1}\underline{\Sigma}_j\underline{\eta}_j + 2\underline{u}_j^T(\underline{\Gamma}_j^T\underline{S}_{j+1}\underline{\Phi}_j + \underline{M}_j^T) \\ & \underline{x}_j\Big\} \end{aligned} \quad \text{(D.120)}$$

Equation (D.120) is very similar to Equation (D.101), therefore $\underline{u}_j$ can be found in the same way as $\underline{u}_{N-2}$ and the result is:

$$\underline{u}_j = -(\underline{\Gamma}_j^T\underline{S}_{j+1}\underline{\Gamma}_j + \underline{B}_j)^{-1}(\underline{\Gamma}_j^T\underline{S}_{j+1}\underline{\Phi}_j + \underline{M}_j^T)\hat{\underline{x}}_j \quad \text{(D.121)}$$

From Equation (D.121), one can say that Equations (D.71) and (D.72) are valid for $i = j$. Substituting the value of $\underline{u}_j$ from Equation (D.121) into Equation (D.120), the value of V_{N-j} is found to follow the form of Equation (D.117) with $j+1$ replaced by j and where $\underline{S}_j$ is computed from Equation (D.73) for $i = j$. Thus the proof is completed.

III.2 Continuous Problem

For the continuous case, a Linear-Quadratic-Gaussian problem, as stated in Chapter V, is to determine a physically realizable control vector function $\underline{u}(t)$ in the time interval (t_o, t_f) so that the quadratic performance index (cost function)

$$J = \tfrac{1}{2}E\left\{\underline{x}^T(t_f)\underline{P}_f\underline{x}(t_f) + \int_{t_o}^{t_f}\left[\underline{x}^T(t) \;\vdots\; \underline{u}^T(t)\right]\left[\begin{array}{c|c}\underline{Q}_1(t) & \underline{N}(t)\\ \hline \underline{N}^T(t) & \underline{R}_1(t)\end{array}\right]\left[\begin{array}{c}\underline{x}(t)\\ \hline \underline{u}(t)\end{array}\right] dt\right\} \qquad (D.122)$$

of the linear stochastic system mentioned earlier in Section II.2 is a minimum.

The solution was given in Chapter V as Equations (5.19) to (5.21) which are rewritten here as Equations (D.123) to (D.128).

$$\underline{u}(t) = \underline{K}_1(t)\hat{\underline{x}}(t) \qquad (D.123)$$

where $\underline{K}_1(t)$ is the Kalman control gain matrix,

$$\underline{K}_1(t) = -\underline{R}_1^{-1}(t)\left[\underline{F}_2^T(t)\underline{P}_1(t) + \underline{N}^T(t)\right] \qquad (D.124)$$

$$\dot{\underline{P}}(t) = -\underline{P}_1(t)\underline{F}_1(t) - \underline{F}_1^T(t)\underline{P}_1(t) + \underline{K}_1^T(t)\underline{R}_1(t)\underline{K}_1(t) - \underline{Q}_1(t) \qquad (D.125)$$

Final condition: $\underline{P}_1(t_f) = \underline{P}_f$

and $\hat{\underline{x}}(t)$ is the Kalman filter estimate which is also the

conditional estimate $E\{\underline{x}(t)/\underline{Y}(\tau) \quad t_o \leq \tau \leq t\}$ of $\underline{x}(t)$ and can be found from Equations (D.126) to (D.128),

$$\dot{\hat{\underline{x}}}(t) = \underline{F}_1(t)\hat{\underline{x}}(t) + \underline{F}_2(t)\underline{u}(t) + \underline{h}_2(t)\left[\underline{y}(t) - \underline{G}(t)\hat{\underline{x}}(t)\right] \tag{D.126}$$

Initial condition; $\hat{\underline{x}}(t_o) = \underline{m}_{x_o}$

$$\underline{h}_2(t) = \left[\underline{P}_2(t)\underline{G}^T(t) + \underline{F}_3(t)\underline{T}(t)\right]\underline{R}_2^{-1}(t) \tag{D.127}$$

= Kalman filter gain matrix

$$\dot{\underline{P}}_2(t) = \underline{F}_1(t)\underline{P}_2(t) + \underline{P}_2(t)\underline{F}_1^T(t) - \underline{h}_2(t)\underline{R}_2(t)\underline{K}_2^T(t) + \underline{F}_3(t)\underline{Q}_2(t)\underline{F}_3^T(t) \tag{D.128}$$

Initial condition: $\underline{P}_2(t_o) = \underline{P}_o$

The validity of Equations (D.123) to (D.128) can be confirmed by first finding an equivalent discrete problem and then letting the samples become dense, that is, letting the sampling interval approach zero. The limit of the solution of the discrete problem is that of the continuous one.

Consider the discrete problem mentioned earlier in Section III.1. Let the subscript k be denoted at time $t = k\Delta$ where Δ is a sampling interval; then Equations (D.60) and (D.61) become,

$$\underline{x}(k\Delta+\Delta) = \underline{\Phi}(k\Delta)\underline{x}(k\Delta) + \underline{\Gamma}(k\Delta)\underline{u}(k\Delta) + \underline{\Sigma}(k\Delta)\underline{n}(k\Delta) \tag{D.129}$$

$$\underline{y}(k\Delta) = \underline{G}(k\Delta)\underline{x}(k\Delta) + \underline{\varepsilon}(k\Delta) \tag{D.130}$$

Also let

$$\underline{\Phi}(k\Delta) = \underline{I} + \Delta\underline{F}_1(k\Delta) \tag{D.131}$$

$$\underline{P}(k\Delta) = \Delta \underline{F}_2(k\Delta) \tag{D.132}$$

$$\underline{\Sigma}(k\Delta) = \Delta \underline{F}_3(k\Delta) \tag{D.133}$$

$$\underline{\eta}(k\Delta) = \underline{w}(k\Delta) \tag{D.134}$$

$$\underline{\xi}(k\Delta) = \underline{v}(k\Delta) \tag{D.135}$$

$$E\left\{\begin{bmatrix}\underline{\eta}(k\Delta)\\ \hline \underline{\xi}(k\Delta)\end{bmatrix}\left[\underline{\eta}^T(i\Delta) \;\vdots\; \underline{\xi}^T(i\Delta)\right]\right\}$$

$$= \begin{bmatrix}\eta\underline{\Omega}(k\Delta) & \vdots & \eta\xi\underline{\Omega}(k\Delta)\\ \hline \eta\xi\underline{\Omega}^T(k\Delta) & \vdots & \xi\underline{\Omega}(k\Delta)\end{bmatrix}\delta_h(k-i)$$

$$= \frac{1}{\Delta}\begin{bmatrix}\underline{Q}_2(k\Delta) & \vdots & \underline{T}(k\Delta)\\ \hline \underline{T}^T(k\Delta) & \vdots & \underline{R}_2(k\Delta)\end{bmatrix}\delta_h(k-i) \tag{D.136}$$

then the discrete equations (D.129) and (D.130) will be the same as the continuous equations (D.5) and (D.6) when the following limits are taken:

$$\Delta \longrightarrow 0$$

$$k \longrightarrow \infty$$

$$k\Delta = t$$

The discrete performance index, defined in Equation (D.59), is rewritten here as Equation (D.137),

$$J = \tfrac{1}{2}E\left\{\underline{x}^T(N\Delta)\underline{A}(N\Delta)\underline{x}(N\Delta) + \sum_{i=0}^{N-1}\left[\underline{x}^T(i\Delta) \;\vdots\; \underline{u}^T(i\Delta)\right]\right.$$

$$\left.\begin{bmatrix}\underline{A}(i\Delta) & \vdots & \underline{M}(i\Delta)\\ \hline \underline{M}^T(i\Delta) & \vdots & \underline{B}(i\Delta)\end{bmatrix}\begin{bmatrix}\underline{x}(i\Delta)\\ \hline \underline{u}(i\Delta)\end{bmatrix}\right\} \tag{D.137}$$

Let

$$t_o = 0\Delta = 0 \tag{D.138}$$

$$t_f = N\Delta \tag{D.139}$$

$$\underline{A}(N\Delta) = \underline{P}_f \tag{D.140}$$

$$\begin{bmatrix} \underline{A}(i\Delta) & \underline{M}(i\Delta) \\ \underline{M}^T(i\Delta) & \underline{B}(i\Delta) \end{bmatrix} = \Delta \begin{bmatrix} \underline{Q}_1(i\Delta) & \underline{N}(i\Delta) \\ \underline{N}^T(i\Delta) & \underline{R}_1(i\Delta) \end{bmatrix} \tag{D.141}$$

$$\text{for } i = 0,1,\ldots,N-1$$

then if the following limits are taken,

$$\Delta \longrightarrow 0$$

$$N \longrightarrow \infty$$

$$\Delta N = t_f$$

the discrete performance index described by Equation (D.137) will be the same as the continuous performance index described by Equation (D.122).

The two Linear-Quadratic-Gaussian problems, one is discrete and another is continuous, are now equivalent. Therefore their solutions are also equivalent,

$$\hat{\underline{x}}(t) = \underset{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}}{\text{Lim.}} \hat{\underline{x}}(i\Delta) = \underset{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}}{\text{Lim.}} E\left\{\underline{x}(i\Delta)/\underline{Y}(i\Delta)\right\}$$

$$= E\left\{\underline{x}(t)/\underline{Y}(\tau) \quad t_o = \tau = t\right\}$$

which is the continuous Kalman filter estimate of $\underline{x}(t)$ and

was found in Section II.1 as the solution of equations (D.126) to (D.128).

Let the equivalence of ${}_1\underline{K}_i$ and $\underline{S}_i$ in the continuous problem be $\underline{k}_1(t)$ and $\underline{F}_1(t)$ then,

$$\underline{k}_1(t) = \lim_{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}} {}_1\underline{K}(i\Delta) \qquad (D.142)$$

or from Equation (D.72),

$$\underline{k}_1(t) = - \lim_{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}} \left[\underline{\Gamma}^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{\Gamma}(i\Delta) + \underline{B}(i\Delta)\right]^{-1} \left[\underline{\Gamma}^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{\Phi}(i\Delta) + \underline{M}^T(i\Delta)\right]$$

or from Equations (D.131), (D.132) and (D.141),

$$\underline{k}_1(t) = - \lim_{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}} \left[\Delta^2\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{F}_2(i\Delta) + \Delta\underline{R}_1(i\Delta)\right]^{-1} \left[\Delta\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\left\{\underline{I} + \Delta\underline{F}_1(i\Delta)\right\} + \Delta\underline{N}^T(i\Delta)\right]$$

or,

$$\underline{k}_1(t) = - \lim_{\substack{\Delta \to 0 \\ i \to \infty \\ i\Delta = t}} \left[\Delta\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{F}_2(i\Delta) + \underline{R}_1(i\Delta)\right]^{-1} \left[\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta) + \Delta\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{F}_1(i\Delta) + \underline{N}^T(i\Delta)\right] .$$

or,

$$\underline{k}_1(t) = - \underline{R}_1^{-1}(t)\left[\underline{F}_2^T(t)\underline{F}_1(t) + \underline{N}^T(t)\right] \qquad (D.143)$$

Equation (D.143) is the same as Equation (D.124). Thus the remainder of the proof is to show the validity of Equation (D.125). By definition $\underline{F}_1(t)$ is the equivalence of $\underline{F}_i$ in the continuous problem. Hence,

$$\underline{P}_1(t_f) = \underline{S}(N\Delta) = \underline{A}(N\Delta) = \underline{P}_f \qquad (D.144)$$

Combining equations (D.73), (D.131), (D.132) and (D.141) gives:

$$\underline{S}(i\Delta) = \left[\underline{I} + \Delta\underline{F}_1(i\Delta)\right]^T \underline{S}(i\Delta+\Delta)\left[\underline{I} + \Delta\underline{F}_1(i\Delta)\right] - {}_1\underline{K}^T(i\Delta)$$
$$\left[\Delta\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{F}_2(i\Delta) + \Delta\underline{R}_1(i\Delta)\right]{}_1\underline{K}(i\Delta) + \Delta\underline{Q}_1(i\Delta)$$

or,

$$\frac{\underline{S}(i\Delta+\Delta) - \underline{S}(i\Delta)}{\Delta} = - \Delta\underline{F}_1^T(i\Delta)\underline{S}(i\Delta+\Delta)\underline{F}_1(i\Delta) - \underline{F}_1^T(i\Delta)\underline{S}(i\Delta+\Delta)$$
$$- \underline{S}(i\Delta+\Delta)\underline{F}_1(i\Delta) + {}_1\underline{K}_1^T(i\Delta)\left[\Delta\underline{F}_2^T(i\Delta)\underline{S}(i\Delta+\Delta)\right.$$
$$\left.\underline{F}_2(i\Delta) + \underline{R}_1(i\Delta)\right]{}_1\underline{K}(i\Delta) - \underline{Q}_1(i\Delta) \qquad (D.145)$$

Taking the limit of Equation (D.145) as $\Delta \longrightarrow 0$, $i \longrightarrow \infty$ and $i\Delta = t$, results in Equation (D.146),

$$\dot{\underline{S}}(t) = - \underline{F}_1^T(t)\underline{F}_1(t) - \underline{F}_1(t)\underline{F}_1(t) + \underline{K}_1^T(t)\underline{R}_1(t)\underline{K}_1(t) - \underline{Q}_1(t)$$
$$(D.146)$$

Equation (D.146) combined with Equation (D.144) forms Equation (D.136). The proof is completed.

APPENDIX E

FORTRAN LISTINGS

This appendix contains the Fortran listings of eight important programs used in the thesis. They were run on teletypes connected to the DEC-PDP-10 time-sharing digital computer of the University of Western Ontario.

1 PROGRAM 1

Based on the simplified form of the Hartley's modification, Program 1 was written to compute least squares estimates of the elements of $\underline{\theta}$ and their 95%-confidence intervals from the seven sets of batch data of A. pullulans (Chapter III). The vector $\underline{\theta}$ consists of process parameters and initial conditions of the microbial growth model represented by Eq. (3.01). The program is in double precision and contains four subroutine-subprograms: TRAJ, DER, SUMSQ and MINV.

The subroutine TRAJ computes $x^j(t_i^j)$ for all j and i from Eq. (3.01) when $\underline{\theta}$ is given. The Adams-Bashforth predictor-corrector with the first four points generated from a fourth order Runge-Kutta integration technique is employed. The integration interval is one hour and the linear-interpolation is used to compute values substrate concentrations.

product concentrations and pH's at times between measurements.

The subroutine JAK calculates $dc^j(t_i^j)/d\underline{\theta}$ for all j and i when $c^j(t_i^j)$ for all j and i, and $\underline{\theta}$ are provided. The finite difference approximation technique described by Eq. (3.21) is employed.

The subroutine OBJEQ computes the sum of squares s defined by Eq. (3.06) when both $c_m^j(t_i^j)$ and $c^j(t_i^j)$ for all j and i are supplied.

Finally, the subroutine MINFV inverts a square matrix using the Gauss-Jordan technique (Haggerty 1977). Note that the matrices involved in this program, as well as in the remaining programs are symmetric and positive definite; to save computer time a subroutine which inverts a symmetric positive definite matrix if available should be used instead.

Input to Program 1

Input to Program 1 are:

- NB (format (I2)) specifying the number of batches from which data are used. Maximum allowable value for NB is 5.

- THETA(I), I = 1,...,12 (format (12G)) representing the initial approximation of vector $\underline{\theta}$.

- KA(I) I = 1,...,5 (format (5I1)) specifying which parameters are not fitted (are set to zero). If KA(I) = 1 then a_I is not fitted.

- The fermentation data stored on a disc in a file named DATA.DAT. NI(I) specifies the number of data points in batch I and is read in first under format (I2). Then DELT(I,J), XM(I,J), S(I,J), P(I,J) and R(I,J) J = 1,...,NI(I) are read in together using format (5F10.4) with one line for each value of J. The whole cycle of reading is repeated for batch #2,...,#NB. Note that DELT(I,J) is time between the j^{th} and the $(j-1)^{th}$ measurements in the i^{th} batch with the unit of hour; XM(I,J), S(I,J) and P(I,J) are the measured biomass concentration, substrate concentration and product concentration at time of the j^{th} measurement in the i^{th} batch respectively, all in the same unit of g/100ml; and R(I,J) is the pH at time of the j^{th} measurement in the i^{th} batch.

Output from Program 1

Output from program 1 are estimated values of the parameters a's, their 95%-confidence intervals, sum of squares S, standard deviation σ and determinant of the inverted matrix for every iteration; at the end, values of $t_{n-1,\alpha}$, of measured biomass concentrations, of model biomass concentrations and of their differences are printed out. If,

. ITEST = 0 then, normal convergence has occurred,

. ITEST = 1 then, sum of squares S was not improved even after the stepsize had been cut in half 15 times,

. ITEST = 2 then, the test of convergence of S had not been satisfied within 15 iterations.

It should be noted that Program 1 can also be used when the microbial growth model is not described by Eq. (3.81). That is, if the Monod model is used,

$$\frac{dc(t)}{dt} = c(t)\left[\frac{\mu_m S(t)}{S(t) + k_s}\right]$$

then the function FDERIV, defined by lines 7 and 8 of the subroutine TRAJ, would have to be changed to the following:

FDERIV(CCX, SX, PX, HX) = CCX*(THET(1)*SX/(SX+THET(2))

and the values for kA(1) I = 1,...,5 should be read into the program as 00111.

A listing of Program 1 follows:

```
      IF (I-K) 120,120,108
108   JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=A(JI)
110   A(JI)=HOLD
120   J=M(K)
      IF (J-K) 100,100,125
125   K1=K-N
      DO 130 I=1,N
      K1=K1+N
      HOLD=A(K1)
      JI=K1-K+J
      A(K1)=A(JI)
130   A(JI)=HOLD
      GO TO 100
150   RETURN
      END
```

```
      TYPE 12,0
      NN=0
      DO 56 J=1,NPAR
      DO 56 I=1,NPAR
      NN=NN+1
56    R(I,J) CR(NN)
      DO 59 I=1,NPAR
      DTHET(I)=0.D+00
      DO 59 J=1,NPAR
      R(I,J) R(I,J)/DSQRT(DI(I)*DI(J))
59    DTHET(I)=DTHET(I)+R(I,J)*Q(J)
      VBK=1.
      DO 100 NCH=1,15
      NN=0
      DO 90 I=1,NPAR
      IF (I.GT.5) GO TO 99
      IF (KA(I).EQ.0) GO TO 101
99    NN=NN+1
      THETN(I) THET(I)+VBK*DTHET(NN)
      GO TO 90
101   THETN(I)=THET(I)
90    CONTINUE
      CALL TKAJ(THETN,CC)
      CALL SUMSQ(CC,SUM2)
      ESTSIG=DSQRT(SUM2/XND)
      TYPE 13,NCH,SUM2,ESTSIG
      IF (SUM1.GE.SUM2) GO TO 92
      VBK=VBK*.50+00
100   CONTINUE
      ITEST=1
      GO TO 150
92    NN=0
      DO 93 I=1,NPAR
      IF (I.GT.5) GO TO 120
      IF (KA(I).EQ.0) GO TO 121
120   NN=NN+1
      DELTH(I)=2.0*00*DSQRT(DABS(R(NN,NN)))*ESTSIG
      GO TO 93
121   DELTH(I)=0.D+00
93    CONTINUE
      TYPE 14
      TYPE 15,(THETN(I),DELTH(I),I=1,NPAR)
      DIF=SUM1-SUM2
      TCON=1.D-03*(SUM1+1.D-15)
      IF (DIF.LT.TCON) GO TO 140
      DO 96 I=1,NPAR
96    THET(I)=THETN(I)
      SUM1=SUM2
200   CONTINUE
      ITEST=2
      GO TO 150
140   ITEST=0
150   TYPE 16,ITEST
      TYPE 9,(THETN(I),I=1,NPAR)
      DO 300 I=1,NB
      MN1=N1(I)
      TYPE 17,I
      DO 300 J=1,MN1
300   ERROR(M(I,J))=CC(I,J)
      TYPE 18,(M(I,J),CC(I,J),ERROR
1     FORMAT(I2)
2     FORMAT(5E20.4)
3     FORMAT(I26)
4     FORMAT(5I11)
5     FORMAT(//////////,18H TERMS NOT FITTED )
```

```
6       FORMAT(/H IPRM   ,I3)
7       FORMAT(///,24H NUMBER OF BATCHES USED ,I4)
8       FORMAT(//,26H INITIAL GUESS OF THET(I) )
9       FORMAT(5D14.5)
10      FORMAT(6HSSUM =,D14.7,5X,8HESTSIG =,D14.7)
11      FORMAT(///,12H ITERATION  ,I4)
12      FORMAT(5H DET  ,D17.7)
13      FORMAT(3H N=  ,I3,5X,5HSUM =,D14.7,5X,8HESTSIG =,D14.7)
14      FORMAT(38H CONFIDENCE INTERVALS OF THET(I) )
15      FORMAT(D18.7,5H +OR-,D14.7)
16      FORMAT(///,8H ITEST =,I2,7,25H FINAL VALUES OF THET(I) )
17      FORMAT(2///,9H BATCH   ,I4,/,15X,3H CM,15X,3H CC,12X,
     1  6H ERROR,/)
18      FORMAT(3D18.5)
        STOP
        END

C
C
        SUBROUTINE TRAJ (TIME,CC)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DIMENSION THET(12),CC(7,13),S(7,13),P(7,13),
     1  H(7,13),DELT(7,13),NI(7),STORE(4)
        COMMON /CBZ/ S,P,H,DELT
        COMMON /CC/ NI,NB
        FDERIV(CCX,SX,PX,HX)=CCX*(THET(1)+THET(2)*CCX
     1  +THET(3)*SX+THET(4)*PX+THET(5)*HX)
        DO 1 I=1,NB
        GRS=(S(I,2)-S(I,1))/DELT(I,1)
        GRP=(P(I,2)-P(I,1))/DELT(I,1)
        GRH=(H(I,2)-H(I,1))/DELT(I,1)
        NII=NI(I)-1
        YS=S(I,1)
        YP=P(I,1)
        YH=H(I,1)
        CC(I,1)=THET(I+5)
        ZCC=CC(I,1)
        DO 2 M=1,3
        XM=FLOAT(M-1)
        XK1=FDERIV(ZCC,YS,YP,YH)
        STORE(M)=XK1
        YCC=ZCC+.5D+00*XK1
        YS=S(I,1)+GRS*(XM+.5D+00)
        YP=P(I,1)+GRP*(XM+.5D+00)
        YH=H(I,1)+GRH*(XM+.5D+00)
        XK2=FDERIV(YCC,YS,YP,YH)
        YCC=ZCC+.5D+00*XK2
        XK3=FDERIV(YCC,YS,YP,YH)
        YCC=ZCC+XK3
        YS=S(I,1)+GRS*(XM+1.D+00)
        YP=P(I,1)+GRP*(XM+1.D+00)
        YH=H(I,1)+GRH*(XM+1.D+00)
        XK4=FDERIV(YCC,YS,YP,YH)
2       ZCC=ZCC+(XK1+2.D+00*(XK2+XK3)+XK4)/6.D+00
        STORE(4)=FDERIV(ZCC,YS,YP,YH)
        XI=XM+1.D+00
        DO 4 J=1,NII
        YS=S(I,J+1)-S(I,J))/DELT(I,J)
        YP=(P(I,J+1)-P(I,J))/DELT(I,J)
        YH=(H(I,J+1)-H(I,J))/DELT(I,J)
        ZCC=ZCC+.5D+00*STORE(4)-5*D+00*STORE(3)+37.D+00*
     1  STORE(2)-2.D+00*STORE(1))/24.D+00
        XI=XI+1.D+00
        YS=S(I,J)+GRS*ACCUNT
        YP=P(I,J)+GRP*ACCUNT
        YH=H(I,J)+GRH*ACCUNT
```

```
      ZCS=FDERIV(ZCS,YS,YP,YH)
      ZCC=ZCC+(9.D+00*ZCS+19.D+00*STORE(4)-5.D+00*STORE(3)+
     1 STORE(2))/24.D+00
      DO 3 N=1,3
    3 STORE(N)=STORE(N+1)
      STORE(4)=FDERIV(ZCC,YS,YP,YH)
      IF (XCOUNT.NE.DELT(I,J)) GO TO 4
      XCOUNT=0.D+00
      JJ=J+1
    1 CC(I,JJ)=ZCC
      RETURN
      END
C
C
      SUBROUTINE DER (THET,CC,DERCC)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION DELT(12),THET(12),CC(7,13),CCC(7,13),
     1 DERCC(7,13,12),NI(7),KA(5)
      COMMON /CC/ NI,NB
      COMMON /CD/ KA
      FACT=1.D-04
      K=1
      NPAR=5+NB
      DO 20 L=1,NPAR
      IF (L.GT.5) GO TO 5
      IF (KA(L).EQ.0) GO TO 20
    5 DO 2 J=1,NPAR
      IF (L.EQ.J) GO TO 3
      THET(J)=THET(J)
      GO TO 2
    3 THET(J)=(1.D+00+FACT)*(THET(J)+1.D-15)
    2 CONTINUE
      CALL TRAJ (THET,CCC)
      DO 4 M=1,NB
      NIM=NI(M)
      DO 4 N=1,NIM
      DERCC(M,N,K)=(CCC(M,N)-CC(M,N))/(FACT*(THET(L)+1.D-15))
    4 CONTINUE
      K=K+1
   20 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE SUMSQ (CC,SUM)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION CC(7,13),CM(7,13),NI(7)
      COMMON /CA/ CM
      COMMON /CC/ NI,NB
      SUM=0.D+00
      DO 1 I=1,NB
      NII=NI(I)
      DO 1 J=1,NII
    1 SUM=SUM+(CM(I,J)-CC(I,J))**2
      RETURN
      END
C
C
      SUBROUTINE MINV (A,N,D,L,M)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(1),L(1),M(1)
      D=1.D+00
      NK=-N
      DO 80 K=1,N
      NK=NK+N
```

```
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=A(KK)
      DO 20 J=K,N
      IZ=N*(J-1)
      DO 20 I=K,N
      IJ=IZ+I
      AAA=DABS(BIGA)
      BBB=DABS(A(IJ))
      IF (AAA-BBB) 15,20,20
   15 BIGA=A(IJ)
      L(K)=I
      M(K)=J
   20 CONTINUE
      J=L(K)
      IF (J-K) 35,35,25
   25 KI=K-N
      DO 30 I=1,N
      KI=KI+N
      HOLD=-A(KI)
      JI=KI-K+J
      A(KI)=A(JI)
   30 A(JI)=HOLD
   35 I=M(K)
      IF (I-K) 45,45,38
   38 JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-A(JK)
      A(JK)=A(JI)
   40 A(JI)=HOLD
   45 IF (DABS(BIGA)-1.D-38)46,46,48
   46 D=0.D+00
      RETURN
   48 DO 55 I=1,N
      IF (I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
      DO 65 I=1,N
      IK=NK+I
      HOLD=A(IK)
      IJ=I-N
      DO 65 J=1,N
      IJ=IJ+N
      IF (I-K) 60,65,60
   60 IF (J-K) 62,65,62
   62 KJ=IJ-I+K
      A(IJ)=HOLD*A(KJ)+A(IJ)
   65 CONTINUE
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF (J-K) 70,75,70
   70 A(KJ)=A(KJ)/BIGA
   75 CONTINUE
      D=D*BIGA
      A(KK)=1.D+00/BIGA
   80 CONTINUE
      K=N
  100 K=K-1
      IF (K) 150,150,105
  105 I=L(K)
```

```
      IF (I-K) 120,120,108
108   JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=A(JI)
110   A(JI)=HOLD
120   J=M(K)
      IF (J-K) 100,100,125
125   K1=K-N
      DO 130 I=1,N
      K1=K1+N
      HOLD=A(K1)
      JI=K1-K+J
      A(K1)=A(JI)
130   A(JI)=HOLD
      GO TO 100
150   RETURN
      END
```

II PROGRAM II

For purposes of checking identification procedures, Program II was written to simulate artificial data from the model described in Example 1 of Chapter IV. The program is in double precision and contains a function-subprogram RANDOM and three subroutine-subprograms TRAJ, QUAN and GAUSS.

RANDOM is a library function-subprogram which generates random numbers distributed uniformly between zero and unity.

The subroutine TRAJ is not the same as that used in Program I, though they both have the subroutine integrates eq. (4.27) when $\underline{\theta} = [a_1 \quad a_2 \quad x^1(t_o^1) \quad x^2(t_o^2) \quad \ldots]^T$ is given using a fourth order Runge-Kutta technique. The integration interval is .01 unit.

The subroutine QUAN provides a random number of either 0. or 1. or . . . or 10..

Finally the subroutine GAUSS provides a random number having Gaussian distribution with 0-mean and $(SIG)^2$-variance. The following approximation is made: If x_i's for $i = 1, 2, \ldots, 12$ are independent random numbers distributed uniformly between zero and unity then,

$$y = \left(\sum_{i=1}^{12} x_i - 6\right)\sigma + \bar{y}$$

is a random number which approximately follows a Gaussian distribution with $\bar{y}$-mean and σ^2-variance.

Input to Program 11

Input to Program 11 are the number of batches NB and the standard deviation SIG (format (I3,G)). NB can have a maximum value of 300.

Output from Program 11

Output from Program 11 are stored in a disc file named DATA.DAT. Data for each batch are stored in sequence with 11 lines/batch. The first line specifies the batch number under format (I3) and each line thereafter represents a measurement and contains y, $\bar{u}_1$ and $\bar{u}_2$ (format (1H , 3F20.6)).

A listing of Program 11 follows.

FORTRAN LISTING OF PROGRAM II

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),U1(300,10),U2(300,10),Y(300,10),YIN(300)
      COMMON NB,U1,U2,YIN
      DATA A/-.50D00,2.0D00/
      ACCEPT,1,NB,SIG
      DO 11 I=1,NB
      DO 12 J=1,10
      CALL QUAN(Q)
      U1(I,J)=5.*Q
      CALL QUAN(Q)
12    U2(I,J)=10.*Q-5.
      CALL QUAN(Q)
11    YIN(I)=5.*Q
      CALL TRAJ(A,Y)
      DO 13 I=1,NB
      DO 14 J=1,10
      CALL GAUSS(SIG,ERR)
14    Y(I,J)=Y(I,J)+ERR
      CALL OPEN(2,'DATA')
      DO 13 I=1,NB
      WRITE(2,2) I
13    WRITE(2,3)(Y(I,J),U1(I,J),U2(I,J),J=1,10)
1     FORMAT(I3,)
2     FORMAT(I3)
3     FORMAT(10F20.5)
      STOP
      END
C
      SUBROUTINE TRAJ(A,Y)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),U1(300,10),U2(300,10),Y(300,10),YIN(300)
      COMMON NB,U1,U2,YIN
      FUNC(XY,XU1,XU2)=A(1)*XU1+XY+A(2)*XU2
      H=.01
      DO 11 I=1,NB
      Y(I,1)=YIN(I)
      DO 11 J=2,10
      ZY=Y(I,J-1)
      ZU1=U1(I,J-1)
      ZU2=U2(I,J-1)
      F1=H*FUNC(ZY,ZU1,ZU2)
      ZYA=ZY+F1/2.
      F2=H*FUNC(ZYA,ZU1,ZU2)
      ZYB=ZY+F1/2.+F2
      F3=H*FUNC(ZYB,ZU1,ZU2)
      ZYC=ZY+F1+F2*4+F3
      F4=H*FUNC(ZYC,ZU1,ZU2)
11    Y(I,J)=Y(I,J-1)+(F1+4.*F2+3.*F3+F4)/8.
      RETURN
      END
C
      SUBROUTINE QUAN(Q)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      YY=RANDOM(0)
      Q=0.
      P=1./11.
11    IF(YY.LT.P) GO TO 12
      Q=Q+1.
      P=P+2./11.
      GO TO 11
12    RETURN
      END
C
      SUBROUTINE GAUSS(SIG,ERR)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      SUM=0.
      DO 11 I=1,12
11    SUM=SUM+RANDOM(0)
      ERR=(SUM-6.)*SIG
      RETURN
      END
```

III PROGRAM III

Utilizing the superposition Principle and the Walling-Lawton Sylvestre idea, Program was written to identify parameters a_1 and a_2 of the model described in Example 1, section II.2 of Chapter IV from the five batch data simulated by Program II. The program is in double precision and contains four subroutine-subprograms: FKAI, DERIV, SUMSQ and AINTV.

The subroutine FKAI is not the same as that used in Programs I and II. The subroutine computes the five trajectories $y^j(\underline{a}, t_i^j)$ $j = 1,\ldots,5$ and $i = 0,\ldots,9$ when $\underline{a} = (a_1, a_2)^T$ is given. The computation is based on Eqs. (4.12), (4.13), (4.17), (4.18), (4.19) and (4.21), and on the use of a fourth order Runge-Kutta integration technique with an integration interval of .01 unit.

The subroutine DERIV calculates $\partial y^j(\underline{a}, t_i^j)/\partial \underline{a}^T$ for all j and i when $y^j(\underline{a}, t_i^j)$ for all j and i, and $\underline{a}$ are provided. The finite difference approximation technique described by Eq. (4.21) is employed.

The subroutine SUMSQ computes the sum of squares defined by Eq. (4.29) when both $y_m^j(t_i^j)$ and $y^j(\underline{a}, t_i^j)$ for all j and i are supplied.

Finally the subroutine AINTV is the same as that

used and listed in Program 1. The matrix to be inverted in the program is only (2 x 2), thus the subroutine MINV is not really needed. However it is included just for the sake of generality.

Input to Program III

Input to Program III are:

- A(1) and A(2) (format (2G)) representing the initial approximation of the process parameter vector $\underline{a}$.

The artificial data simulated from Program II and stored on a disc in file data.dat. Data for each batch are read in sequentially and in the following order: the first line is the batch number N (format (I3)) and the next ten lines represent the ten consecutive measurements in that batch with each line containing the values of y_m, $\bar{u}_{1m}$ and $\bar{u}_{2m}$ (format (2X, 3F10.5)).

Output from Program III

Output from Program III are the estimated values of a_1 and a_2, their 95%-confidence intervals, sum of squares S_1, standard deviation $\hat{\sigma}$ and determinant of the inverted matrix for every iteration. At the end, values of I_{max}, of measured outputs, of model outputs and of their differences are printed out. The interpretation of the value of I_{max}

is the same as that used in Program 1.

Program III without the subroutine MINV is listed on the following pages.

FORTRAN LISTING OF PROGRAM III (Subroutine MINTV not included)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),ANR(2),DIA(10),B(5,10),X(5,10),Y(5,10),
     1  DXDA(5,10,2),R(2,2),RR(2),R1(4),II(2),IM(2),DA(2),DI(A(2)
      COMMON G(2,2)
      ACCEPT 1,A(1),A(2)
      CALL TITLE (2,'DATA')
      DO 20 I=1,9
      READ (5,1) N
      DO 10 J=1,10
   10 READ (5,1) Y(I,J),B(I,J),B2(I,J)
      TYPE *,A(1),A(2)
      CALL INV(A,N)
      CALL NORM(A,N,SUM1)
      ISING=SQRT(SUM1/99.)
      TYPE *,SUM1,ISING
      DO 100 NITER=1,15
      CALL DERIV(A,N,DX,A)
      TYPE 2,NITER
      DO 21 I=1,2
      DO 21 J=1,2
      R(I,J)=0.
      DO 22 M=1,9
      DO 22 N=1,10
   22 R(I,J)=R(I,J)+DXDA(M,N,I)*DXDA(M,N,J)
      RR(I)=0.
      DO 21 M=1,9
      DO 21 N=1,10
   21 RR(I)=RR(I)+DXDA(M,N,I)*(Y(M,N)-X(M,N))
      NN=0
      DO 24 I=1,2
      DO 24 J=1,2
      NN=NN+1
   24 R1(NN)=R(I,J)
      CALL MINV(R1,2,D,II,IM)
      TYPE 2,D
      NN=0
      DO 25 I=1,2
      DO 25 J=1,2
      NN=NN+1
   25 R(I,J)=R1(NN)
      DO 26 I=1,2
      DA(I)=0.
      DO 26 J=1,2
   26 DA(I)=DA(I)+R(I,J)*RR(J)
      ANR=0.
      DO 100 N=1,10
      DO 100 I=1,2
      ANR=ANR+A(I)*B(I,N)*DA(I)
      CALL TRA(ANR,X)
      CALL NORM(X,N,SUM2)
      ISING=SQRT(SUM2/99.)
      TYPE *,NITER,SUM2,ISING
      IF(SUM2.GT.SUM1) GO TO 22
      SUM1=SUM2
  100 CONTINUE
      TYPE 1
      DO 30 I=1,2
   30 A(I)=A(I)+DA(I)
      GO TO 23
   22 DO 31 I=1,2
   31 A(I)=A(I)+DA(I)*(SUM1/(ABS(SUM1-SUM2)))*(ISING/
     1  ISING)
      TYPE 2,(ANR(I),II=1,2)
      TYPE 2,(A(I),I=1,2)
      GO TO 23
      END
```

```
      DO 29 I=1,2
29    A(I)=ANEW(I)
      SUM1=SUM2
200   CONTINUE
      ITEST=2
      GO TO 150
140   ITEST=0
150   TYPE 12,ITEST
      TYPE 11,ANEW(1),ANEW(2),(X(I,1),I=1,5)
      DO 300 I=1,5
      TYPE 13,I
      DO 300 J=1,10
      ERROR=Y(I,J)-X(I,J)
300   TYPE 14,Y(I,J),X(I,J),ERROR
1     FORMAT (2G)
2     FORMAT (I3)
3     FORMAT (1H ,D20.5)
4     FORMAT (//,38H INITIAL GUESSES OF A(1) AND A(2) ARE ,
     1/,D15.5,/,D15.5)
5     FORMAT (6H SUM =,D15.5,5X,9HESTSIG =,D15.5)
6     FORMAT (//,11H ITERATION ,I4)
7     FORMAT (5H DET=,D15.5)
8     FORMAT (3H ,I3,5X,5HSUM =,D15.5,5X,9HESTSIG =,D15.5)
9     FORMAT (30H CONFIDENCE INTERVALS OF A(I) )
10    FORMAT (D15.5,5H  OR ,D15.5)
11    FORMAT (D18.5)
12    FORMAT (//,8H ITEST ,I2,/,22H FINAL VALUES OF A(I) )
13    FORMAT (//,8H BATCH ,I3,/,15X,3H Y,15X,3H X,
     1 [illegible],6H ERROR,/)
14    FORMAT (3D18.5)
      STOP
      END
C
C
      SUBROUTINE TRAI (A,X)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),X(5,10),U1(5,10),U2(5,10),
     1 Y(5,10),P1(5,10),P2(5,10)
      COMMON U1,U2,Y
      FUN1(XP1,XU1)=A(1)*XU1*XP1
      FUN2(XP2,XU1,XU2)=A(1)*XU1*XP2+A(2)*XU2
      H=.01
      DO 1 I=1,5
      P1(I,1)=1.
      P2(I,1)=0.
      DO 1 J=2,10
      ZP1=P1(I,J-1)
      ZP2=P2(I,J-1)
      ZU1=U1(I,J-1)
      ZU2=U2(I,J-1)
      FA1=H*FUN1(ZP1,ZU1)
      FA2=H*FUN2(ZP2,ZU1,ZU2)
      ZPA1=ZP1+FA1/2.
      ZPA2=ZP2+FA2/2.
      FB1=H*FUN1(ZPA1,ZU1)
      FB2=H*FUN2(ZPA2,ZU1,ZU2)
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      [illegible]
      P1(I,J)=P1(I,J-1)+(FA1+2.*FB1+2.*FC1+FD1)/6.
```

```
     1      P2(I,J) P2(I,J-1)*((A2*3.*B2*3.*C2*D2)/8.
            DO 2 I=1,5
            DUM1=0.
            DUM2=0.
            DO 3 J=1,10
            DUM1=DUM1+P1(I,J)*(Y(I,J)-P2(I,J))
     3      DUM2=DUM2+P1(I,J)*P1(I,J)
     2      X(I,1)=DUM1/DUM2
            DO 4 I=1,5
            DO 4 J=2,10
     4      X(I,J)=P1(I,J)*X(I,1)+P2(I,J)
            RETURN
            END
C
C
            SUBROUTINE DERIV (A,X,DXDA)
            IMPLICIT DOUBLE PRECISION (A-H,O-Z)
            DIMENSION A(5),AA(5),X(5,10),XX(5,10),DXDA(5,10,2)
            FACT=1.D-03
            DO 1 I=1,5
            DO 1 J=1,2
            IF(A(I).EQ.0) GO TO 2
            AA(I)=A(I)
            GO TO 3
     2      AA(I)=(1.D+00+FACT)*(A(I)+1.D-15)
     3      CONTINUE
            CALL DRIV (AA,XX)
            DO 4 M=1,5
            DO 4 N=1,10
     4      DXDA(M,N,I)=(XX(M,N)-X(M,N))/((FACT*(A(I)+1.D-15))
     1      CONTINUE
            RETURN
            END
C
C
            SUBROUTINE SUMSQ (Y,X,S)
            IMPLICIT DOUBLE PRECISION (A-H,O-Z)
            DIMENSION X(5,10),Y(5,10)
            S=0.
            DO 1 I=1,5
            DO 1 J=1,10
     1      S=S+(Y(I,J)-X(I,J))**2
            RETURN
            END
```

IV PROGRAM IV

Utilizing the superposition principle and the Walling-Lawton-Sylvestre idea, Program IV was written to identify the process parameter vector $\underline{a}$ of the microbial growth model from the seven batch data of A. pullulans as described in Example 2 in Section II.2 of Chapter IV. The program is in double precision and contains four subroutine-subprograms: TRAJ, DER, SUMSQ and MINTV.

The subroutine TRAJ computes $y^j(\underline{a}, t_i^j)$ for all j and i when $\underline{a} = [a_1 \quad a_2 \quad \cdots \quad a_4]^T$ is provided. The computation is based on eqs. (4.12), (4.13), (4.17), (4.18), (4.19) and (4.21), and on the use of the Adams-Bashforth predictor-corrector with the first four points generated from a fourth order Runge-Kutta integration technique. The integration interval is one hour and the linear interpolation is used to compute values of substrate and product concentrations at times between measurements.

The subroutine DER calculates $\partial y^j(\underline{a}, t_i^j)/\partial \underline{a}^T$ for all j and i when $y^j(\underline{a}, t_i^j)$ for all j and i, and $\underline{a}$ are supplied. The finite-difference approximation technique described by eq. (4.21) is employed.

The subroutine SUMSQ computes the sum of squares S when both $y_m^j(t_i^j)$ and $y^j(\underline{a}, t_i^j)$ for all j and i are supplied.

And the subroutine MINTV is the same as that used and listed in Program I.

Input to Program IV

Input to Program IV are:

- The experimental data of A. pullulans stored in a disc file DAT.DAT. Data for each batch are read in sequentially and in the following order: the first line is the number of measuements NI(I) in that batch (format (I2)) and each line for the next NI(I) lines specifies time between this measurement and next measurement DELT(I,J), measured biomass concentration CM(I,J), measured substrate concentration S(I,J) and measured product concentration P(I,J) (format 4F10.4)). These concentrations are all in the same unit of g/100ml.

- A(1), A(2) and A(3) (format (3G)) representing the initial approximation of the process parameter vector $\underline{a}$.

Output from Program IV

Output from Program II are estimated values of $\hat{a}_1$, a_2 and a_3; their 95%-confidence intervals, sum of squares $\hat{S}$, standard deviation $\hat{\sigma}$ and determinant of the inverted matrix for every iteration. At the end, values of ITEST, of

measured biomass concentrations, of model biomass concentrations, and of their differences are printed out. The interpretation of the value of ITEST is the same as that used in Program I.

Program IV without subroutine MINTV is listed on the following three pages.

FORTRAN LISTING OF PROGRAM IV (Subroutine MINTV not included)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(3),ANEW(3),CC(7,13),CM(7,13),S(7,13),
     1 P(7,13),DELT(7,13),NI(7),IL(3),IM(3),
     2 DCCDA(7,13,3),R(3,3),DA(3),Q(3),DELA(3)
      COMMON /CA/CM
      COMMON /CB/ S,P,DELT
      COMMON /CC/NI
      CALL IFILE (1,'DAT')
      DO 30 I=1,7
      READ(1,1) NI(I)
      MNI=NI(I)
      DO 30 J=1,MNI
   30 READ(1,2) DELT(I,J),CM(I,J),S(I,J),P(I,J)
      ACCEPT 4,(A(I),I=1,3)
      TYPE 5
      CALL TRAJ (A,CC)
      TYPE 6,(A(I),I=1,3),(CC(I,1),I=1,7)
      CALL SUMSQ (CC,SUM1)
      ESTSIG=DSQRT(SUM1/72.)
      TYPE 7,SUM1,ESTSIG
      DO 200 NITER=1,15
      TYPE 8,NITER
      CALL DER (A,CC,DCCDA)
      DO 20 I=1,3
      DO 21 J=1,3
      R(I,J)=0.
      DO 21 M=1,7
      MNI=NI(M)
      DO 21 N=1,MNI
   21 R(I,J)=R(I,J)+DCCDA(M,N,I)*DCCDA(M,N,J)
      Q(I)=0.
      DO 22 M=1,7
      MNI=NI(M)
      DO 22 N=1,MNI
   22 Q(I)=Q(I)+DCCDA(M,N,I)*(CM(M,N)-CC(M,N))
   20 CONTINUE
      CALL MINTV (R,3,D,IL,IM)
      TYPE 9,D
      DO 23 I=1,3
      DA(I)=0.
      DO 23 J=1,3
   23 DA(I)=DA(I)+R(I,J)*Q(J)
      VBK=1.
      DO 100 NCH=1,15
      DO 24 I=1,3
      ANEW(I)=A(I)+VBK*DA(I)
   24 CONTINUE
      CALL TRAJ (ANEW,CC)
      CALL SUMSQ (CC,SUM2)
      ESTSIG=DSQRT(SUM2/72.)
      TYPE 10,NCH,SUM2,ESTSIG
      IF (SUM1.GE.SUM2) GO TO 25
      VBK=VBK*.5D+00
  100 CONTINUE
      ITEST=1
      GO TO 150
   25 DO 26 I=1,3
      DELA(I)=2.D+00*DSQRT(DABS(R(I,I)))*ESTSIG
   26 CONTINUE
      TYPE 11
      TYPE 12,(ANEW(I),DELA(I),I=1,3)
      TYPE 6,(CC(I,1),I=1,7)
      DIF=SUM1-SUM2
      TCON=1.D-03*(SUM1+1.D-15)
      IF (DIF.LE.TCON) GO TO 140
      DO 27 I=1,3
```

```
   27       A(I)=ANEW(I)
            SUM1=SUM2
  200       CONTINUE
            ITEST=2
            GO TO 150
  140       ITEST=0
  150       TYPE 13,ITEST
            TYPE 6,(ANEW(I),I=1,3)
            DO 300 I=1,7
            MNI=NI(I)
            TYPE 14,I
            DO 300 J=1,MNI
            ERROR=CM(I,J)-CC(I,J)
  300       TYPE 15,CM(I,J),CC(I,J),ERROR
    1       FORMAT (I2)
    2       FORMAT (4F10.4)
    4       FORMAT (3G)
    5       FORMAT (//,24H INITIAL GUESS OF A(I) =)
    6       FORMAT (D14.5)
    7       FORMAT (6H SUM =,D14.7,5X,8HESTSIG =,D14.7)
    8       FORMAT (///,12H ITERATION  ,I4)
    9       FORMAT (6H DET =,D14.7)
   10       FORMAT (3H   ,I3,5X,5HSUM =,D14.7,5X,8HESTSIG =,D14.7)
   11       FORMAT (31H CONFIDENCE INTERVALS OF A(I) =)
   12       FORMAT (D18.7,5H +OR-,D14.7)
   13       FORMAT (///,8H ITEST =,I2,/,23H FINAL VALUES OF A(I) =)
   14       FORMAT (///,8H BATCH  ,I3,/,15X,3H CM,15X,3H CC,12X,
     1      6H ERROR,/)
   15       FORMAT (3D18.5)
            STOP
            END
C
C
            SUBROUTINE TRAJ (A,CC)
            IMPLICIT DOUBLE PRECISION (A-H,O-Z)
            DIMENSION A(3),CC(7,13),S(7,13),P(7,13),CM(7,13),
     1      DELT(7,13),NI(7),STORE(4),H(7,13)
            COMMON /CA/ CM
            COMMON /CB/S,P,DELT
            COMMON /CC/ NI
            FDERIV (CCX,SX,PX)=CCX*(A(1)+A(2)*SX+A(3)*PX)
            DO 1 I=1,7
            GRS=(S(I,2)-S(I,1))/DELT(I,1)
            GRP=(P(I,2)-P(I,1))/DELT(I,1)
            NI11=NI(I)-1
            YS=S(I,1)
            YP=P(I,1)
            H(I,1)=1.
            ZCC=H(I,1)
            DO 2 M=1,3
            XM=FLOAT(M-1)
            XK1=FDERIV(ZCC,YS,YP)
            STORE(M)=XK1
            YCC=ZCC+.5D+00*XK1
            YS=S(I,1)+GRS*(XM+.5D+00)
            YP=P(I,1)+GRP*(XM+.5D+00)
            XK2=FDERIV(YCC,YS,YP)
            YCC=ZCC+.5D+00*XK2
            XK3=FDERIV(YCC,YS,YP)
            YCC=ZCC+XK3
            YS=S(I,1)+GRS*(XM+1.D+00)
            YP=P(I,1)+GRP*(XM+1.D+00)
            XK4=FDERIV(ZCC,YS,YP)
    2       ZCC=ZCC+(XK1+2.D+00*XK2+2.D+00*XK3+XK4)/6.D+00
            STORE(4)=FDERIV(ZCC,YS,YP)
            XCOUNT=3.D+00
```

```
      DO 1 J=1,NIII
      GRS=(S(I,J+1)-S(I,J))/DELT(I,J)
      GRP=(P(I,J+1)-P(I,J))/DELT(I,J)
4     ZC5=ZCC+(55.D+00*STORE(4)-59.D+00*STORE(3)+37.D+00*
     1  STORE(2)-9.D+00*STORE(1))/24.D+00
      XCOUNT=XCOUNT+1.D+00
      YS=S(I,J)+GRS*XCOUNT
      YP=P(I,J)+GRP*XCOUNT
      ZC5=FDERIV(ZC5,YS,YP)
      ZCC=ZCC+(9.D+00*ZC5+19.D+00*STORE(4)-5.D+00*STORE(3)+
     1  STORE(2))/24.D+00
      DO 3 N=1,3
3     STORE(N)=STORE(N+1)
      STORE(4)=FDERIV(ZCC,YS,YP)
      IF (XCOUNT.NE.DELT(I,J)) GO TO 4
      XCOUNT=0.D+00
      JJ=J+1
1     H(I,JJ)=ZCC
      DO 5 I=1,7
      NNI=NI(I)
      D1=0.
      D2=0.
      DO 6 J=1,NNI
      D1=D1+H(I,J)*CM(I,J)
6     D2=D2+H(I,J)*H(I,J)
5     CC(I,1)=D1/D2
      DO 7 I=1,7
      NNI=NI(I)
      DO 7 J=2,NNI
7     CC(I,J)=H(I,J)*CC(I,1)
      RETURN
      END
C
C
      SUBROUTINE DER (A,CC,DCCDA)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(3),AA(3),CC(7,13),CCC(7,13),DCCDA(7,13,3),NI(7)
      COMMON /CC/ NI
      FACT=1.D-04
      DO 4 I=1,3
      DO 2 J=1,3
      IF (I.EQ.J) GO TO 3
      AA(J)=A(J)
      GO TO 2
3     AA(J)=(1.D+00+FACT)*(A(J)+1.D-15)
2     CONTINUE
      CALL TRAJ(AA,CCC)
      DO 4 M=1,7
      MNI=NI(M)
      DO 4 N=1,MNI
4     DCCDA(M,N,I)=(CCC(M,N)-CC(M,N))/(FACT*(A(I)+1.D-15))
      RETURN
      END
C
C
      SUBROUTINE SUMSQ (CC,SUM)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION CC(7,13),CM(7,13),NI(7)
      COMMON /CA/ CM
      COMMON /CC/ NI
      SUM=0.D+00
      DO 1 I=1,7
      MNI=NI(I)
      DO 1 J=1,MNI
1     SUM=SUM+(CM(I,J)-CC(I,J))**2
      RETURN
      END
```

V PROGRAM V

Program V was written to compute an on-line estimate of the process parameter vector $\underline{a} = [a_1 \quad a_2]^T$ from the model described in section III.2 of Chapter IV. The program is in double precision and contains four subroutine-subprograms: TRAJ1, DER1, MAMUL and MINTV.

The subroutine TRAJ1 calculates the whole new trajectory $y^{L+1}(\underline{a}, t_i^{L+1})$ $i = 0, 1, \ldots, 9$ when $\underline{a}$ is given. The computation is based on Eqs. (4.12), (4.13), (4.17), (4.18), (4.19) and (4.21), and on the use of a fourth order Runge-Kutta integration technique with an integration interval of .01 unit.

The subroutine DER1 computes $\partial y^{L+1}(\underline{a}, t_i^{L+1})/\partial \underline{a}^T$ for all i when $y^{L+1}(\underline{a}, t_i^{L+1})$ for all i, and $\underline{a}$ are provided. The finite difference approximation technique described by Eq. (3.21) is employed.

The subroutine MAMUL multiplies two matrices $\underline{A}$ and $\underline{B}$ and gives the result in matrix $\underline{C}$.

And finally the subroutine MINTV inverts a square matrix. It is the same as the subroutine MINTV used and listed in Program I.

Input to Program V

Input to Program V are:

- NTB (format (I3)) specifying the total number of batch from which data are read in from the file DATA1.DAT.

- Starting values for $\underline{a}$ and $\underline{P}$, i.e., A(1), A(2), P(1,1), P(1,2), P(2,1) and P(2,2) (format (2G)).

- The artificial data simulated from Program II and stored in a disc file DATA1.DAT. Data for each batch are read in one at a time and in the following sequence: the first line is the batch number NBN (format (I3)) and the next ten lines represent the ten consecutive measurements in that batch with each line containing the values of y_m, $\bar{u}_1$ and $\bar{u}_2$ (format (1H , 3F20.5)).

Output from Program V

Output from Program V are batch number, on-line estimates of a_1 and a_2 after data from that batch are received. All these three values are typed in one line and there are NTB lines altogether.

Program V without the subroutine MINTV is listed on the following two pages.

FORTRAN LISTING OF PROGRAM V (Subroutine MINTV not included)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),P(2,2),X(10),U1(10),U2(10),Y(10),TW1(2,1),
     1  QQ(10,2),QQT(2,10),TWTEN(2,10),TETEN(10,10),CR(100),
     2  IL(10),IM(10),GAINK(2,10),TETW(10,2),TWTW(2,2),TEN1(10,1)
      COMMON U1,U2,Y
      ACCEPT 3,NTB
      ACCEPT 5,A(1),A(2),P(1,1),P(1,2),P(2,1),P(2,2)
      NBN=0
      TYPE 1
      TYPE 2,NBN,A(1),A(2)
      CALL IFILE (1,'DATA1')
      DO 20 NB=1,NTB
      READ (1,3) NBN
      DO 21 I=1,10
21    READ (1,4) Y(I),U1(I),U2(I)
      CALL TRAJ1(A,X)
      CALL DER1(A,X,QQ)
      DO 22 I=1,2
      DO 22 J=1,10
22    QQT(I,J)=QQ(J,I)
      CALL MAMUL (P,QQT,TWTEN,2,2,10)
      CALL MAMUL (QQ,TWTEN,TETEN,10,2,10)
      NN=0
      DO 23 J=1,10
      DO 23 I=1,10
      NN=NN+1
      CR(NN)=TETEN(I,J)
      IF (I.NE.J) GO TO 23
      CR(NN)=CR(NN)+1.
23    CONTINUE
      CALL MINTV (CR,10,D,IL,IM)
      NN=0
      DO 24 J=1,10
      DO 24 I=1,10
      NN=NN+1
24    TETEN(I,J)=CR(NN)
      CALL MAMUL (TWTEN,TETEN,GAINK,2,10,10)
      CALL MAMUL (QQ,P,TETW,10,2,2)
      CALL MAMUL (GAINK,TETW,TWTW,2,10,2)
      DO 25 I=1,2
      DO 25 J=1,2
25    P(I,J)=P(I,J)-TWTW(I,J)
      DO 26 I=1,10
26    TEN1(I,1)=Y(I)-X(I)
      CALL MAMUL (GAINK,TEN1,TW1,2,10,1)
      A(1)=A(1)+TW1(1,1)
      A(2)=A(2)+TW1(2,1)
20    WRITE (5,2) NBN,A(1),A(2)
1     FORMAT (////,7X,'BATCH',14X,4HA(1),14X,4HA(2),/)
2     FORMAT (I7,2D18.5)
3     FORMAT (I3)
4     FORMAT (1H ,3F20.5)
5     FORMAT (2G)
      STOP
      END
C
C
      SUBROUTINE TRAJ1 (A,X)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),X(10),U1(10),U2(10),Y(10),P1(10),P2(10)
      COMMON U1,U2,Y
      FUN1(XP1,XU1)=A(1)*XU1+XP1
      FUN2(XP2,XU1,XU2)=A(1)*XU1*XP2+A(2)*XU2
      H=.01
      P1(1)=1.
```

```
      P2(1)=0.
      DO 1. J=2,10
      ZP1=P1(J-1)
      ZP2=P2(J-1)
      ZU1=U1(J-1)
      ZU2=U2(J-1)
      FA1=H*FUN1(ZP1,ZU1)
      FA2=H*FUN2(ZP2,ZU1,ZU2)
      ZPA1=ZP1+FA1/3.
      ZPA2=ZP2+FA2/3.
      FB1=H*FUN1(ZPA1,ZU1)
      FB2=H*FUN2(ZPA2,ZU1,ZU2)
      ZPA1=ZP1-FA1/3.+FB1
      ZPA2=ZP2-FA2/3.+FB2
      FC1=H*FUN1(ZPA1,ZU1)
      FC2=H*FUN2(ZPA2,ZU1,ZU2)
      ZPA1=ZP1+FA1-FB1+FC1
      ZPA2=ZP2+FA2-FB2+FC2
      FD1=H*FUN1(ZPA1,ZU1)
      FD2=H*FUN2(ZPA2,ZU1,ZU2)
      P1(J)=P1(J-1)+(FA1+3.*FB1+3.*FC1+FD1)/8.
1     P2(J)=P2(J-1)+(FA2+3.*FB2+3.*FC2+FD2)/8.
      DUM1=0.
      DUM2=0.
      DO 3 J=1,10
      DUM1=DUM1+P1(J)*(X(J)-P2(J))
2     DUM2=DUM2+P1(J)*P1(J)
      X(1)=DUM1/DUM2
      DO 4 J=2,10
4     X(J)=P1(J)*X(1)+P2(J)
      RETURN
      END
C
C
      SUBROUTINE DER1 (A,X,QQ)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(2),AA(2),X(10),XX(10),QQ(10,2)
      FACT=1.D-03
      DO 1 I=1,2
      DO 3 J=1,2
      IF (I.EQ.J) GO TO 2
      AA(J)=A(J)
      GO TO 3
2     AA(J)=(1.D+00+FACT)*(A(J)+1.D-15)
3     CONTINUE
      CALL TRAJ1 (AA,XX)
      DO 4 N=1,10
4     QQ(N,I)=(XX(N)-X(N))/(FACT*(A(I)+1.D-15))
1     CONTINUE
      RETURN
      END
C
C
      SUBROUTINE MAMUL(A,B,C,NX,NY,NZ)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(NX,NY),B(NY,NZ),C(NX,NZ)
      DO 1 I=1,NX
      DO 1 J=1,NZ
      C(I,J)=0.
      DO 1 K=1,NY
1     C(I,J)=C(I,J)+A(I,K)*B(K,J)
      RETURN
      END
```

VI PROGRAM VI

As described in Chapter V, Program VI was written to compute the optimum dilution rate profile that regulates the effluent concentrations in a continuous fermentation process of Aerobacter aerogenes. In the program data and results are available only at discrete time $t_i = .51$ hr for $i = 0,1,\ldots,99$. The program is in double precision and contains five subroutine-subprograms: P1, P2, PARA, ANEW and DAFIL.

The subroutine P1 computes the values of the matrix function $\underline{P}_1(t)$ at times t_i $i = 0,1,\ldots,99$ by solving the matrix Ricatti differential equation (5.18) backward in time. The Euler integration technique with an integration interval of .05 hr is used.

The subroutine P2 computes the values of the matrix function $\underline{P}_2(t)$ at times t_i $i = 0,1,\ldots,99$ by solving the matrix Ricatti differential equation (5.21) forward in time. The Euler integration technique with an integration interval of .05 hr is used.

The subroutine PARA calculates $\underline{F}_1$, $\underline{f}_2$, $\underline{R}_2$, $\underline{s}_2$ and $\underline{T}$ as defined from Eqs. (5.26), (5.27), (5.29), (5.25) and (5.30), respectively. The matrix $\underline{F}_1$ is represented in the computer program by A(I) I = 1,2,3 and the vector $\underline{f}_2$ by

B(1) and B(2).

The subroutine ANEW computes $\underline{x}(t_{i+1})$ from Eq. (5.13) when $\underline{x}(t_i)$, $u(t_i)$, $\underline{w}(t_i)$ and $\underline{w}(t_{i+1})$ are given. The Euler integration technique with an integration interval of .05 hour is used. The linear interpolation technique is employed to compute $\underline{w}(t)$ for $t_i < t < t_{i+1}$.

And finally the subroutine DXFIL computes $\hat{\underline{x}}(t_{i+1})$ from Eq. (5.19) when $\underline{x}(t_i)$, $u(t_i)$, $\underline{y}(t_i)$ and $\underline{y}(t_{i+1})$ are given. The Euler integration technique with an integration interval of .05 hour is used. The linear interpolation technique is employed to compute $\underline{y}(t)$ for $t_i < t < t_{i+1}$.

Input to Program VI

Besides the values k_3, k_{31}, k_{32}, M_m, K_s, K, Y, $(t_{i+1} - t_i)$, $\bar{D}$ and $\bar{I}_I$ supplied to the program through the three DATA statements, input to Program VI are:

- SIG1 and SIG2 (format (2G)) representing τ_1 and τ_2.
- R1, A(1) and A(2) (format (3G)) representing k_3, k_1 and k_2.
- X(1,1) and X(2,1) (format (2G)) representing $D(t_o)$ and $\dot{D}(t_o)$.
- Noise data stored on a disc in a file named ADAT1.DAT: W(1,I), W(2,I), W(3,I), V(1,I) and V(2,I) I = 1,...,100

(format (5F14.6)) representing $w'_c(t_i)$, $w'_{S1}(t_i)$, $w'_{S2}(t_i)$, $v_c(t_i)$ and $v_c(t_i)$ $i = 0,\ldots,99$.

Output from Program VI

Output from Program VI are A(1,I), A(2,I), AFIL(1,I), AFIL(2,I) and B(I) I = 1,...,100 (format (5F14.6)) representing $C(t_i)$, $S(t_i)$, $C(t_i)$, $S(t_i)$ and $D(t_i)$ $i = 0,\ldots,99$. The output are stored on a disc in a file ADAT2.DAT. Note that $C(t_i)$ and $S(t_i)$ can be combined with $w'_c(t_i)$ and $w'_{S1}(t_i)$ to give the control effluent concentrations $C_e(t_i)$ and $S_e(t_i)$ as seen from Eqs. (5.4), (5.5), (5.7) and (5.8).

A listing of Program VI follows:

FORTRAN LISTING OF PROGRAM VI.

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(3),B(2),P(3,100),PP(3,100),XFIL(2,100),X(2,100),
     1 R(2),RO(2),R(2),T(2),XSS(2),Y(2,100),U(100),V(2,100),W(3,100),
     2 WOLD(3),WNEW(3),XOLD(2),XN(2),DXFN(2),AL(3)
      COMMON A,B,R,R1,Q,QQ,T,H
      COMMON /COM/ ZMU,ZKS,ZK,YLD,AL,SIG1,SIG2,XSS
      COMMON /COM1/XFIL,U
      DATA AL/.50-01,.50-01,.50-01/
      DATA ZMU,ZKS,ZK,YLD/1.0760+00,.7480-02,.1170+00,.50+00/
      DATA H,D,PI/.50+00,.9110+00,3.0+00/
      ACCEPT 2,SIG1,SIG2
      ACCEPT 2,R1,Q(1),Q(2)
      ACCEPT 2,X(1,1),X(2,1)
      P(1,100)=0.
      P(2,100)=0.
      P(3,100)=0.
      PP(1,1)=SIG1*SIG1
      PP(2,1)=0.
      PP(3,1)=SIG2*SIG2
      CALL PARA
      CALL IFILE (1,'ADAT1')
      READ (1,1)(W(1,I),W(3,I),V(1,I),V(2,I),I=1,100)
      DO 11 I=1,2
      XFIL(I,1)=X(I,1)
      DXFN(I)=X(I,1)-XSS(I)
   11 Y(I,1)=X(I,1)+V(I,1)
      CALL P1(P)
      CALL P2(PP)
      DO 20 I=1,99
      DU=-(B(1)*(P(1,I)*DXFN(1)+P(2,I)*DXFN(2))
     1 +B(2)*(P(2,I)*DXFN(1)+P(3,I)*DXFN(2)))/R1
      DO 12 J=1,2
      XOLD(J)=X(J,I)
      WOLD(J)=W(J,I)
   12 WNEW(J)=W(J,I+1)
      WOLD(3)=W(3,I)
      WNEW(3)=W(3,I+1)
      U(I)=DU+D
      CALL XNEW(H,U(I),WOLD,WNEW,XOLD,XN)
      DO 13 J=1,2
      X(J,I+1)=XN(J)
   13 Y(J,I+1)=XN(J)+AL(J)*XN(J)*W(J,I+1)+V(J,I+1)
      CALL DXFIL(XSS,D,PP,Y,I)
      DXFN(1)=XFIL(1,I+1)-XSS(1)
      DXFN(2)=XFIL(2,I+1)-XSS(2)
   20 CONTINUE
      U(100)=0
      CALL OFILE(2,'ADAT2')
      WRITE (2,1)(X(1,I),X(2,I),XFIL(1,I),XFIL(2,I),U(I),I=1,100)
    1 FORMAT(5F14.6)
    2 FORMAT(3G)
      STOP
      END
C
C
      SUBROUTINE P1(P)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      DIMENSION P(3,100),A(3),B(2),R(2),RO(2),QQ(2),T(2)
      COMMON A,B,R,R1,Q,QQ,T,H
      DT=H/10.
      P1=P(1,100)
      P2=P(2,100)
      P3=P(3,100)
      DO 20 I=2,100
      DO 15 J=1,10
```

```
      PN1=P1-DT*(-2.*P2*A(2)+((P1*B(1)+P2*
     1  B(2))**2)/R1-Q(1))
      PN2=P2-DT*(-P1*A(1)-P2*A(3)-P3*A(2)+
     1  (P1*B(1)+P2*B(2))*(P2*B(1)+P3*B(2))/R1)
      PN3=P3-DT*(-2.*(P2*A(1)+P3*A(3))+((P2*
     1  B(1)+P3*B(2))**2)/R1-Q(2))
      P1=PN1
      P2=PN2
      P3=PN3
      CONTINUE
      M=101-I
      P(1,M)=P1
      P(2,M)=P2
      P(3,M)=P3
      CONTINUE
      RETURN
      END

      SUBROUTINE P2(PP)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION PP(3,100),A(3),B(2),R(2),Q(2),QQ(2),T(2)
      COMMON A,B,R,R1,Q,QQ,T,H
      DT=H/10.
      P1=PP(1,1)
      P2=PP(2,1)
      P3=PP(3,1)
      DO 20 I=2,100
      DO 15 J=1,10
      PN1=P1+DT*(2.*P2*A(2)-((P1+T(1))**2)/R(1)-P2*P2/R(2)+QQ(1))
      PN2=P2+DT*(P1*A(1)+P2*A(3)+P3*A(2)-(P1+T(1))*P2/R(1)
     1  -P2*(P3+T(2))/R(2))
      PN3=P3+DT*(2.*(P2*A(1)+P3*A(3))-P2*P2/R(1)-((P3+T(2))**2)/R(2)
     1  +QQ(2))
      P1=PN1
      P2=PN2
      P3=PN3
      CONTINUE
      PP(1,I)=P1
      PP(2,I)=P2
      PP(3,I)=P3
      CONTINUE
      RETURN
      END

      SUBROUTINE PARA
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(3),B(2),R(2),Q(2),QQ(2),T(2),XSS(2),AL(3)
      COMMON A,B,R,R1,Q,QQ,T,H
      COMMON /COM/ D,SI,ZMU,ZKS,ZK,YLD,AL,SIG1,SIG2,XSS
      XSS(2)=ZKS*(D+ZK)/(ZMU-D-ZK)
      XSS(1)=YLD*D*(SI-XSS(2))/(D+ZK)
      A(1)=ZMU*ZKS*XSS(1)/((XSS(2)+ZKS)**2)
      A(2)=-ZMU*XSS(2)/YLD/(XSS(2)+ZKS)
      A(3)=-D-A(1)/YLD
      B(1)=-XSS(1)
      B(2)=SI-XSS(2)
      R(1)=(AL(1)*XSS(1))**2+SIG1*SIG1
      R(2)=(AL(2)*XSS(2))**2+SIG2*SIG2
      QQ(1)=(D*AL(1)*XSS(1))**2
      QQ(2)=(D*AL(2)*XSS(2))**2+(D*AL(3)*SI)**2
      T(1)=-D*(AL(1)*XSS(1))**2
      T(2)=-D*(AL(2)*XSS(2))**2
      RETURN
      END
```

```
      SUBROUTINE XNEW(H,U,WOLD,WNEW,XOLD,XNE)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION WOLD(3),WNEW(3),XOLD(2),XNE(2),AL(3),XSS(2)
      COMMON /COM/ O,SI,ZMU,ZKS,ZK,YLD,AL,SIG1,SIG2,XSS
      DT=H/10.
      XO1=XOLD(1)
      XO2=XOLD(2)
      DO 15 I=1,10
      ZI=FLOAT(I-1)/10.
      W1=WOLD(1)+(WNEW(1)-WOLD(1))*ZI
      W2=WOLD(2)+(WNEW(2)-WOLD(2))*ZI
      W3=WOLD(3)+(WNEW(3)-WOLD(3))*ZI
      XN1=XO1+DT*((-U-ZK+ZMU*XO2/(XO2+ZKS))*XO1-AL(1)*U*XO1*W1)
      XN2=XO2+DT*(U*(SI-XO2)-ZMU*XO1*XO2/YLD/(XO2+ZKS)
     1  -AL(2)*U*XO2*W2+AL(3)*U*SI*W3)
      XO1=XN1
      XO2=XN2
   15 CONTINUE
      XNE(1)=XO1
      XNE(2)=XO2
      RETURN
      END
C
C
      SUBROUTINE DXFIL(XSS,O,PP,Y,M)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION PP(3,100),XSS(2),XFIL(2,100),Y(2,100),U(100)
      DIMENSION A(3),B(2),R(2),Q(2),QQ(2),T(2)
      COMMON A,B,R,RL,Q,QQ,T,H
      COMMON /COM1/XFIL,U
      DT=H/10.
      DXO1=XFIL(1,M)-XSS(1)
      DXO2=XFIL(2,M)-XSS(2)
      DU=U(M)-O
      DO 15 I=1,10
      ZI=FLOAT(I-1)/10.
      PP1=PP(1,M)+(PP(1,M+1)-PP(1,M))*ZI
      PP2=PP(2,M)+(PP(2,M+1)-PP(2,M))*ZI
      PP3=PP(3,M)+(PP(3,M+1)-PP(3,M))*ZI
      DY1=Y(1,M)-XSS(1)+(Y(1,M+1)-Y(1,M))*ZI
      DY2=Y(2,M)-XSS(2)+(Y(2,M+1)-Y(2,M))*ZI
      DXN1=DXO1+DT*(A(1)*DXO2+B(1)*DU+(PP1+T(1))*(DY1-DXO1)/R(1)
     1  +PP2*(DY2-DXO2)/R(2))
      DXN2=DXO2+DT*(A(2)*DXO1+A(3)*DXO2+B(2)*DU+PP2*(DY1-DXO1)
     1  /R(1)+(PP3+T(2))*(DY2-DXO2)/R(2))
      DXO1=DXN1
   15 DXO2=DXN2
      XFIL(1,M+1)=DXO1+XSS(1)
      XFIL(2,M+1)=DXO2+XSS(2)
      RETURN
      END
```

VII PROGRAM VII

As described in Chapter V, Program VII was also written to compute the optimum dilution rate profile that regulates the effluent concentrations in a continuous fermentation process of Aerobacter aerogenes. The difference between Program VI and Program VII is that the latter is used for the case where there is measurement delay. Program VII is in double precision and contains six subroutine-subprograms: P1, P2, PARA, XNEW, DXFIL and DXPRE.

The first five subroutines P1, P2, PARA, XNEW and DXFIL are the same as those used in Program VI. And the subroutine DXPRE computes the state predictor $\underline{x}(t/t - \Delta t)$ from Eq. (5.40). The Euler integration technique with an integration interval of .05 hour is used.

Input to Program VII

Input to Program VII are the same as those to Program VI except that in the third paragraph one more number is read in:

- X(1,I), X(2,I) and N (format (3G)) where N specifies the delay time Δt,

$$N = \Delta t/(t_{i+1} - t_i) = \Delta t/.5$$

Output from Program VII

Output from Program VII are X(1,I), X(2,I), XPRE(1,I), XPRE(2,I) and U(I) I = 1,...,100 (format (5F14.6)) representing $C(t_i)$, $S(t_i)$, $C(t_i/t_i - \Delta t)$, $S(t_i - \Delta t)$ and $D(t_i)$ i = 0,...,99. The output are stored on a disc in a file aDAT2.DAT.

The main program and the subroutine DXPRE of Program VII are listed on the last two pages.

```
      SUBROUTINE DXPRE(XSS,D,XPRE,M,NN)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION XPRE(2,100),XFIL(2,100),XSS(2),U(100)
      DIMENSION A(3),B(2),R(2),Q(2),QQ(2),T(2)
      COMMON A,B,R,RI,Q,QQ,T,H
      COMMON /COM1/ XFIL,U
      DT=H/10.
      MN=M+1-NN
      DX01=XFIL(1,MN)-XSS(1)
      DX02=XFIL(2,MN)-XSS(2)
      DO 15 J=1,NN
      MJ=M+J-1
      DU=U(MJ)-D
      DO 15 I=1,10
      DXN1=DX01+DT*(A(1)*DX02+B(1)*DU)
      DXN2=DX02+DT*(A(2)*DX01+A(3)*DX02+B(2)*DU)
      DX01=DXN1
      DX02=DXN2
   15 CONTINUE
      XPRE(1,M+1)=DX01+XSS(1)
      XPRE(2,M+1)=DX02+XSS(2)
      RETURN
      END
```

FORTRAN LISTING OF PROGRAM VII (Subroutines P1, P2, PARA, XNEW and DXFIL not included)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION A(3),B(2),P(3,100),PP(3,100),XFIL(2,100),X(2,100),
     1 Q(2),QQ(2),R(2),T(2),XSS(2),Y(2,100),U(100),V(2,100),W(3,100),
     2 WOLD(3),WNEW(3),XOLD(2),XN(2),XPRE(2,100),DXP(2),AL(3)
      COMMON A,B,R,R1,Q,QQ,T,H
      COMMON /COM/ D,SI,ZMU,ZKS,ZK,YLD,AL,SIG1,SIG2,XSS
      COMMON /COM1/XFIL,U
      DATA AL/.5D-01,.5D-01,.5D-01/
      DATA ZMU,ZKS,ZK,YLD/1.076D+00,.748D-02,.117D+00,.5D+00/
      DATA H,D,SI/.5D+00,.911D+00,3.D+00/
      ACCEPT 2,SIG1,SIG2
      ACCEPT 2,R1,Q(1),Q(2)
      ACCEPT 2,X(1,1),X(2,1),N
      P(1,100)=0.
      P(2,100)=0.
      P(3,100)=0.
      PP(1,1)=SIG1*SIG1
      PP(2,1)=0.
      PP(3,1)=SIG2*SIG2
      CALL PARA
      CALL IFILE (1,'ADAT1')
      READ (1,1)(W(1,I),W(2,I),W(3,I),V(1,I),V(2,I),I=1,100)
      DO 11 I=1,2
      XFIL(I,1)=X(I,1)
      XPRE(I,1)=X(I,1)
      DXP(I)=X(I,1)-XSS(I)
11    Y(I,1)=X(I,1)+V(I,1)
      CALL P1(P)
      CALL P2(PP)
      DO 20 I=1,99
      NN=I
      IF (I.LT.N) GO TO 50
      NN=N
50    DU=-(B(1)*(P(1,I)*DXP(1)+P(2,I)*DXP(2))
     1 +B(2)*(P(2,I)*DXP(1)+P(3,I)*DXP(2)))/R1
      DO 12 J=1,2
      XOLD(J)=X(J,I)
      WOLD(J)=W(J,I)
12    WNEW(J)=W(J,I+1)
      WOLD(3)=W(3,I)
      WNEW(3)=W(3,I+1)
      U(I)=DU+D
      CALL XNEW(H,U(I),WOLD,WNEW,XOLD,XN)
      DO 13 J=1,2
      X(J,I+1)=XN(J)
13    Y(J,I+1)=XN(J)+AL(J)*XN(J)*W(J,I+1)+V(J,I+1)
      CALL DXPRE(XSS,D,XPRE,I,NN)
      CALL DXFIL(XSS,D,PP,Y,I)
      DXP(1)=XPRE(1,I+1)-XSS(1)
      DXP(2)=XPRE(2,I+1)-XSS(2)
20    CONTINUE
      U(100)=D
      CALL OFILE(2,'ADAT2')
      WRITE (2,1)(X(1,I),X(2,I),XPRE(1,I),XPRE(2,I),U(I),I=1,100)
1     FORMAT(5F14.6)
2     FORMAT(3G)
      STOP
      END
C
C
```

VIII. PROGRAM VIII

Based on the simplified form of Hartley's modification, Program VIII was written to compute least squares estimates of the elements of $\underline{\theta}$ and their 95%-confidence intervals for a given set of continuous data. The vector $\underline{\theta}$, as defined by Eq. (3.71), consists of maximum specific growth rate μ_m, saturation coefficient K_s, metabolism coefficient K, yield coefficient Y, initial biomass concentration $C(t_o)$ and initial substrate concentration $S(t_o)$. The program is in double precision and contains four subroutine-subprograms: TRAJ, DERIV, SUMSQ and MINTV.

The subroutine TRAJ computes $C(t_i)$ and $S(t_i)$ for all i from Eqs. (3.68) and (3.69) when $\underline{\theta}$ is given. The Euler integration technique is employed.

The subroutine DERIV calculates $dC(t_i)/d\underline{\theta}$ and $dS(t_i)/d\underline{\theta}$ for all i when $\underline{\theta}$, $C(t_i)$ and $S(t_i)$ for all i are provided. The finite difference approximation technique, described by Eq. (3.21), is used.

The subroutine SUMSQ computes the sum of squares ϕ defined by Eq. (3.70) when $C(t_i)$ and $S(t_i)$ for all i are supplied.

And finally the subroutine MINTV is the same as that used in Program I.

Input to Program VIII

Input to Program VIII are:

- STEP (format (G)) representing the Euler integration interval. For the continuoys fermentation of M. crassipes, STEP is chosen as .5 (hr^{-1}).

- V(1), V(2), D, SI, TVAL (format (5G)) representing biomass weighting constant, substrate weighting constant, dilution rate, influent substrate concentration and the t-value $t_{d-D,(1-\alpha)/2}$ (Appendix C). For the continuous fermentation of M. crassipes, these values are 17.54, 47.62, .1 (hr^{-1}), 5. (g/ℓ) and 2.02 ($t_{42,.475}$) respectively.

- THET(I) I = 1, 6 (format (6G)) representing the initial approximation of vector $\underline{\theta}$.

- The experimental data stored in a disc file FER.DAT. The first line of the file is the number of measurements NDAT (format (I3)). Each line thereafter for the next NDAT lines contains time between two consecutive measurements DELT(I), measured biomass concentration XM(1,I) and measured substrate concentration XM(2,I).

Output from Program VIII

Output from Program VIII are estimated values of the elements of $\underline{\theta}$, their 95%-confidence intervals, $\hat{S}$, $\hat{\Gamma}$ and determinant of the inverted matrix for every iteration. At the end, values of ITEST, measured effluent concentrations

and model effluent concentrations are printed out. The interpretation of the value of ITEST is the same as that used in Program I.

Program VIII without subroutine MINTV.is listed on the following three pages.

FORTRAN LISTING OF PROGRAM VIII (Subroutine MINTV not included)

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION THET(6),THETN(6),X(2,100),XM(2,100),DXDTH(2,100,6),
     1  V(2),R(6,6),RR(6),IL(6),IM(6),DTHET(6),DETHET(6),DELT(100)
      COMMON /COM1/ NDAT
      COMMON /COM2/ D,SI,DELT,STEP
      COMMON /COM3/ V,XM
      ACCEPT 1,STEP
      ACCEPT 1,V(1),V(2),D,SI,TVAL
      ACCEPT 1,(THET(I),I=1,6)
      CALL IFILE(1,'FER')
      READ(1,2) NDAT
      DO 111 I=1,NDAT
111   READ(1,1) DELT(I),XM(1,I),XM(2,I)
      TYPE 3
      TYPE 4,(THET(I),I=1,6)
      CALL TRAJ(THET,X)
      CALL SUMSQ(X,SUM1)
      XNDAT=FLOAT(2*NDAT-6)
      ESTSIG=DSQRT(2.*SUM1/XNDAT)
      TYPE 5,SUM1,ESTSIG
      DO 200 ITER=1,15
      CALL DERIV(THET,X,DXDTH)
      TYPE 6,ITER
      DO 21 I=1,6
      DO 22 J=1,6
      R(I,J)=0.
      DO 22 M=1,2
      DO 22 N=1,NDAT
22    R(I,J)=R(I,J)+V(M)*DXDTH(M,N,I)*DXDTH(M,N,J)
      DO 21 M=1,2
      DO 21 N=1,NDAT
      RR(I)=0.
21    RR(I)=RR(I)+V(M)*DXDTH(M,N,I)*(XM(M,N)-X(M,N))
      CALL MINTV(R,6,DET,IL,IM)
      TYPE 7,DET
      DO 25 I=1,6
      DTHET(I)=0.
      DO 25 J=1,6
25    DTHET(I)=DTHET(I)+R(I,J)*RR(J)
      VBK=1.
      DO 100 NCH=1,10
      DO 26 I=1,6
26    THETN(I)=THET(I)+VBK*DTHET(I)
      CALL TRAJ(THETN,X)
      CALL SUMSQ(X,SUM2)
      ESTSIG=DSQRT(2.*SUM2/XNDAT)
      TYPE 8,NCH,SUM2,ESTSIG
      IF(SUM1.GE.SUM2) GO TO 27
      VBK=VBK*.5
100   CONTINUE
      ITEST=1
      GO TO 150
27    DO 28 I=1,6
28    DETHET(I)=TVAL*ESTSIG*DSQRT(R(I,I))
      TYPE 9
      TYPE 10,(THETN(I),DETHET(I),I=1,6)
      DIF=SUM1-SUM2
      TCON=1.D-03*(SUM1+1.D-15)
      IF(DIF.LE.TCON) GO TO 140
      DO 29 I=1,6
29    THET(I)=THETN(I)
      SUM1=SUM2
200   CONTINUE
      ITEST=2
```

```
      GO TO 150
140   ITEST=2
150   TYPE 11,ITEST
      DO 300 I=1,NDAT
300   TYPE 12,XM(1,I),X(1,I),XM(2,I),X(2,I)
1     FORMAT(6D)
2     FORMAT(I3)
3     FORMAT(26H INITIAL GUESS OF THET(I) ,/)
4     FORMAT(5X,D15.5)
5     FORMAT(/,6H SUM =,D15.5,5X,8HESTSIG =,D15.5)
6     FORMAT(///,12H ITERATION  ,I3)
7     FORMAT(5H DET=,D15.5)
8     FORMAT(3H    ,I3,5X,5HSUM =,D15.5,5X,8HESTSIG =,D15.5)
9     FORMAT(32H CONFIDENCE INTERVALS OF THET(I))
10    FORMAT(D15.5,5H +OR-,D12.5)
11    FORMAT(///,8H ITEST =,I2)
12    FORMAT(4D15.5)
      STOP
      END
C
C
      SUBROUTINE TRAJ(THET,X)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION THET(6),X(2,100),DELT(100)
      COMMON /COM1/ NDAT
      COMMON /COM2/ D,SI,DELT,STEP
      X(1,1)=THET(5)
      X(2,1)=THET(6)
      XOLD1=X(1,1)
      XOLD2=X(2,1)
      DO 10 I=2,NDAT
      XSTEP=STEP
2     XNEW1=XOLD1+STEP*(-D-THET(3)*THET(1)*XOLD2/(XOLD2+THET(2)))
     1*XOLD1
      XNEW2=XOLD2+STEP*(D*(SI-XOLD2)-THET(1)*XOLD2*XOLD1/THET(4)
     1/(XOLD2+THET(2)))
      XOLD1=XNEW1
      XOLD2=XNEW2
      IF (XSTEP.EQ.DELT(I-1)) GO TO 3
      XSTEP=XSTEP+STEP
      GO TO 2
3     X(1,I)=XNEW1
      X(2,I)=XNEW2
10    CONTINUE
      RETURN
      END
C
C
      SUBROUTINE DERIV(THET,X,DXDTH)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION THET(6),THETT(6),X(2,100),XX(2,100),DXDTH(2,100,6)
      COMMON /COM1/ NDAT
      FACT=1.D-04
      DO 4 I=1,6
      DO 2 J=1,6
      IF(I.EQ.J) GO TO 3
      THETT(J)=THET(J)
      GO TO 2
3     THETT(J)=(1.+FACT)*(THET(J)+1.D-15)
2     CONTINUE
      CALL TRAJ(THETT,XX)
      DO 4 M=1,NDAT
      DXDTH(1,M,I)=(XX(1,M)-X(1,M))/(FACT*(THET(I)+1.D-15))
4     DXDTH(2,M,I)=(XX(2,M)-X(2,M))/(FACT*(THET(I)+1.D-15))
      RETURN
      END
```

```
C
C
      SUBROUTINE SUMSQ(A,SUM)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION V(2),X(2,100),XM(2,100)
      COMMON /COM1/ NDAT
      COMMON /COM3/ V,XM
      SUM=0.
      DO 1 I=1,NDAT
    1 SUM=SUM+.5*V(1)*(XM(1,I)-X(1,I))**2+.5*V(2)*(XM(2,I)-X(2,I))**2
      RETURN
      END
```

REFERENCES

AIBA, S., HUMPHREY, A.E. and MILLIS, N.F. (1965). "Biochemical Engineering". Academic Press, New York and Toronto.

AIBA, S., NAGAI, S., NISHIZAWA, S. and ONODERA, M. (1967). Nucleic Approach to some Response of Chemostatic Culture of Azotobacter vinelandi. Journal of General and Applied Microbiology, v. 13, p. 85.

ALLISON, T.S. (1958). On the Comparison of Two Methods of Off-Line Parameter Identification. Journal of Mathematical Analysis and Applications, v. 18, p. 229.

ANDREWS, J.F. (1968). A Mathematical Model for the Continuous Culture of Microorganisms Utilizing Inhibiting Substrates. Biotechnology and Bioengineering, v. X, p. 707.

ASTROM, K.J., BOHLIN, T. and WENSMARK, S. (1965). Automatic Construction of Linear Stochastic Dynamic Models for Stationary Industrial Processes with Random Disturbances Using Operating Records. I.B.M. Nordic Laboratory, TP 18.150.

ASTROM, K.J. (1970a). Application of Linear Stochastic Control Theory to Paper Machine Control. The 1970 Case Studies in Systems Control, Georgia Institute of Technology, Atlanta, Georgia.

ASTROM, K.J. (1970b). " Introduction to Stochastic Control Theory ". Academic Press, New York and Toronto.

ASTROM, K.J. and EYKHOFF, P. (1971). System Identification - A Survey. Automatica, v. 7, p. 123.

ATHANS, M. and TSE, E. (1967). A Direct Derivation of the Optimum Linear Filter Using the Maximum Principle. I.E.E.E. Transactions on Automatic Control, v. AC-12, p. 690.

ATHANS, M. (1968). The Matrix Minimum Principle. Information and Control, v. 11, p. 592.

ATHANS, M. and KENDRICK, D. (1974). Control Theory and Economics: A Survey, Forecast and Speculations. I.E.E.E. Transactions on Automatic Control, v. AC-19, p. 518.

BALAKRISHAN, A.V. and PETERKA, V. (1969). Identification in Automatic Control Systems. Automatica, v. 5, p. 817.

BARD, Y. (1970). Comparison of Gradient Methods for the Solutions of Nonlinear Parameter Estimation Problem. S.I.A.M. Journal on Numerical Analysis, v. 7, p. 157.

BELLMAN, R.E. (1957). " Dynamic Programming ". Princeton University Press, Princeton, New Jersey.

BELLMAN, R.E. (1960). " Introduction to Matrix Analysis ". McGraw Hill, New York and Toronto.

BELLMAN, R.E., KAGIWADA, H.H. and KALABA, R. (1963). Quasi-linearization, System Identification and Prediction. Report

RM-3821-PR, Rand Corporation, Santa Monica, California.

BELLMAN, R.E. and KALABA, R.E. (1965). " Quasilinearization and Nonlinear Boundary Value Problems ". Elsevier, Amsterdam.

BODE, H.W. and SHANNON, C.E. (1950). A Simplified Derivation of Linear Least Squares Smoothing and Prediction Theory. Proceedings of I.R.E., v. 38, p. 417.

BOOTH, G.W. and PETERSON, T.L. (1958). Nonlinear Estimation. I.B.M. Share Program, # 687-WL-NL1.

BROYDEN, C.G. (1967). Quasi-Newton Methods and Their Application to Function Minimization. Mathematics of Computation, v. 21, p. 368.

BRYSON, A.E. Jr. and HO, Y.C. (1969). " Applied Optimal Control ". Blaisdell Publishing Company, Waltham, Massachusetts.

CHIU, S.Y., ERICKSON, L.E., FAN, L.T. and KAO, I.C. (1972). Kinetic Model Identification of Mixed Populations Using Continuous Culture Data. Biotechnology and Bioengineering, v. XIV, p. 207.

CHOW, J.C. (1972). On Estimating the Orders of an Autoregressive-Moving Average Process with Uncertain Observations. I.E.E.E. Transactions on Automatic Control, v.

AC-17, p. 707.

CONSTANTINDES, A., SPENCER, J.L. and GADEN, E.L. Jr. (1970). Optimization of Batch Process - II. Optimum Temperature Profiles for Batch Penicillin Fermentations. Biotechnology and Bioengineering, v. XII, p. 1081.

COOPER, L. and STEINBERG, D. (1970). " Introduction to Methods of Optimization ". Academic Press, New York and Toronto.

CUENOD, M. and SAGE, A.P. (1967). Comparison of some Methods Used for Process Identification, Survey Paper. I.F.A.C. Symposium on Problems of Identification in Automatic Control, Prague.

D'ANS, G., KOKOTOVIC, P. and GOTTLIEB, D. (1971a). Time Optimal Control for a Model of Bacterial Growth. Journal of Optimization Theory and Applications, v. 7, p. 61.

D'ANS, G., KOKOTOVIC, P. and GOTTLIEB, D. (1971b). A Non-linear Regulator Problem for a Model of Biological Waste Treatment. I.E.E.E. Transactions on Automatic Control, v. AC-16, p. 341.

D'ANS, G., KOKOTOVIC, P. and GOTTLIEB, D. (1972). Optimal Control of Bacterial Growth. Automatica, v. 8, p. 729.

D'ANS, G. (1972). A Theoretical and Experimental Study of Bacterial Growth. Technical Report # T2, Coordinated Science Laboratory, University of Illinois, Urbana.

DAVIES, W.D. (1970). "System Identification for Self Adaptive Control". Wiley-Interscience, New York and Toronto.

DRAPER, N.R. and SMITH, H. (1966). "Applied Regression Analysis". John Wiley and Sons, New York and Toronto.

DUONG, N. (1973). "Modern Control Concept in Hydrology". Ph. D. Dissertation, Colorado State University, Fort Collins.

EDWARDS, V.H. (1970). The Influence of High Substrate Concentrations on Microbial Kinetics. Biotechnology and Bioengineering, v. XII, p. 679.

FYKHOFF, P. (1968). Process Parameter and State Estimation. Automatica, v. 4, p. 205.

FAN, L.T. (1966). "The Continuous Maximum Principle". John Wiley and Sons, New York and Toronto.

FISHER, D.G. and SEBORG (1976). "Multivariable Computer Control. - A Case Study". American Elsevier, New York.

FISHER, R.A. (1922). On the Mathematical Foundations of Theoretical-Statistics. Philosophical Transaction of Royal Society of London, Ser. A, v. 222, p.309.

FISHER, R.A. (1925). Theory of Statistical Estimation. Proceedings of Cambridge Philosophical Society.

GAUDY, A.F. Jr., RAMANATHAN, M. and RAO, B.S. (1967). Kinetic Behaviour of Heterogenous Populations in Completely Mixed Reactors. Biotechnology and Bioengineering, v. IX, p. 387.

GAUSS, K.F. (1821). Theory of Least Squares. English Translation by H.F. Trotter, Princeton University, Statistical Techniques Research Group, Technical Report # 5, 1957.

GAUSS, K.F. (1963). " Theory of the Motion of Heavenly Bodies Moving about the Sun in Conic Section ". Dover, New York.

GELB, A. (1974). " Applied Optimal Estimation ". M.I.T. Press, Cambridge and London.

GILLAY, J.W. and BUNGAY, H.R. III (1967). Oscillatory Growth Rate Responses of S. cerevisiae in Continuous Cultures. Biotechnology and Bioengineering, v. IX, p. 617.

GUNCKEL, T.F. and FRANKLIN, G.F. (1963). A General Solution for Linear Sampled Data Control Systems. Transaction of A.S.M.E., v. 85D, p. 197.

GUPTA, N.K. and MEHRA, R.K. (1974). Computational Aspects of Maximum Likelihood Estimation and Reduction in Sensitivity Function Calculations. I.E.E.E. Transactions on Automatic Control, v. AC-19, p. 774.

GUSTAVSSON, I. (1975). Survey of Applications of Identification in Chemical and Physical Processes. Automatica, v. 11, p. 3.

HAGGERTY, G.B. (1972). " Elementary Numerical Analysis with Programming ". Allyn and Bacon, Boston.

HALDANE, J.B.S. (1965). " Enzymes ". M.I.T. Press, Cambridge, Massachusetts.

HARTLEY, H.O. (1961). The Modified Gauss-Newton Method for the Fitting of Nonlinear Regression Function by Least Squares. Technometrics, v. 3, p. 269.

HERBERT, D., ELSWORTH, R. and TELLING, R.C. (1956). The Continuous Culture of Bacteria, a Theoretical and Experimental Study. Journal of General Microbiology, v. 14, p. 601.

HERBERT, D. (1961). A Theoretical Analysis of Continuous Culture Systems. Society of Chemical Industry, Monograph 12, p. 21.

HIMMELBLAU, D.M. (1972). " Applied Nonlinear Programming ". McGraw Hill, New York and Toronto.

HUMPHREY, A.E. (1963). Some Observations of Continuous Fermentation. Proceedings of I.A.M. Symposium, Institute of Applied Microbiology, University of Tokyo, Tokyo.

I.B.M. 1130 Continuous System Modeling Program. Program Description and Operations Manual. Program N.1130-CX-13X.

JARWINSKI, A. (1971). "Stochastic Processes and Filtering Theory". Academic Press, New York and Toronto.

JOHNSTON, J. (1963). "Econometric Methods". McGraw Hill, New York and Toronto.

JOSEPH, P.D. and Tou, J.T. (1961). On Linear Control Theory. A.I.E.E. Trans. on Application and Industry, v. 80, p. 193.

KALMAN, R.E. (1960a). Contributions to Theory of Optimal Control. Boletin de la Sociedad Matematica Mexicana, v. 5, p. 102.

KALMAN, R.E. (1960b). A New Approach to Linear Filtering and Prediction Problems. Trans. of A.S.M.E., v. 82D, p. 35.

KALMAN, R.E. and BUCY, R.S. (1961). New Results in Linear Filtering and Prediction Theory. Trans. of A.S.M.E., v. 83D, p. 95.

KING, R.E., ARAGONA, J. and CONSTANTINIDES, A. (1974). Specific Optimal Control of a Batch Fermentor. International Journal of Control, v. 20, p. 869.

KING, R.E. (1975). Optimum Energy Conservation in a Batch Penicillin Process. International Journal of Control, v. 22, p. 29.

KIRK, D.E. (1970). "Optimal Control Theory - An Introduction". Prentice Hall, Englewood Cliffs and Toronto.

KOLMOGOROV, A.N. (1941). Interpolation and Extrapolation of Stationary Random Sequences. Bulletin of Academy of Science

of U.S.S.R., Mathematical Series, v. 5. A Translation has been published by the Rand Corporation, Santa Monica, California as Memo # RM-3090-PR.

KONO, T. and ASAI, T. (1969). Kinetics of Continuous Cultivation. Biotechnology and Bioengineering, v. XI, p. 19.

KUSHNER, H.J. (1964). On the Differential Equations Satisfied by Conditional Probability Densities of Markov Processes. S.I.A.M. Journal of Control, v. 2, p. 106.

KWAKERNAAK, H. and SIVAN, R. (1972). "Linear Optimal Control Systems". Wiley-Interscience, New York and Toronto.

LAWTON, W.H. and SYLVESTRE, E.A. (1971). Elimination of Linear Parameters in Nonlinear Regression. Technometrics, v. 13, p. 461.

LEDUY, A. (1972). "Elaboration of Polysaccharide from Sucrose by Fermentation with Pulluria pullulans. M.A.Sc. Thesis, University of Sherbrooke, Sherbrooke, Quebec.

LEDUY, A. and ZAJIC, J.E. (1973). Sampling Procedures for Non-Newtonian Fermentation Broths. Biotechnology and Bioengineering, v. XV, p. 579.

LEDUY, A. (1975). "Cultivation of Morels in Waste Sulfite Liquors". Ph.D. Thesis, University of Western Ontario, London, Ontario.

LEE, E.S. (1968). "Quasilinearization and Invariant Imbedding".

Academic Press, New York and Toronto.

LEE, R.C.K. (1964). " Optimal Estimation, Identification and Control ". Research Monograph # 28, M.I.T. Press, Cambridge, Massachusetts.

LEVENBERG, K. (1944). A Method for Solution of Certain Non-linear Problems in Least Squares. Quarterly of Applied Mathematics, v. 2, p. 164.

LUEDEKING, R. and PIRET, E.L. (1959). A Kinetic Study of Lactic Acid Fermentation. Journal of Biochemical and Microbiological Technology and Engineering, v. 1, p. 393.

LUENBERGER, D.G. (1971). An Introduction to Observers. I.E.E.E. Transactions on Automatic Control, v. AC-16, p. 190.

MALEK, I. and FENCL, Z. (1966). " Theoretical and Methodological Basis of Continuous Cultures of Microorganisms ". Academic Press, New York and Toronto.

MARQUARDT, D.W. (1963). An Algorithm for Least Squares Estimation of Nonlinear Parameters. S.I.A.M. Journal on Applied Mathematics, V. 11, p. 431.

MATELES, R.J., RYU, D.Y. and YASUDA, J. (1965). Measurement of Unsteady State Growth Rate of Microorganisms. Nature, v. 208, p. 263.

MEDITCH, J.S. (1969). " Stochastic Optimal Linear Estimation and Control ". McGraw Hill, New York and Toronto.

MEHRA, R.K. (1970). Maximum Likelihood Identification of Aircraft Parameters. Preprints of the 1970 Joint Automatic Control Conference, Georgia.

MENDEL, J.M. (1973). " Discrete Techniques of Parameter Estimation ". Marcel Dekker, New York.

MOHLER, R.R. (1973). " Bilinear Control Processes ". Academic Press, New York and Toronto.

MONOD, J. (1950). La Technique de Cultures Continues : Theorie et Applications. Anhales de l'Institut Pasteur de Lille, v. 79, p. 390.

MORAINE, R.A. and ROGOVIN, P. (1966). Kinetics of Polysaccharide B-1459 Fermentation. Biotechnology and Bioengineering, v. VIII, p. 511.

MOSER, H. (1958). " The Dynamic of Bacterial Population Maintained in the Chemostat ". Carnegie Institute of Washington, Publication # 614.

NIEMAN, R.E., FISHER, D.G. and SEBORG, D.E. (1971). A Review of Process Identification and Parameter Estimation Techniques. International Journal of Control, v. 13, p. 209.

NOVICK, A. and SZILARD, L. (1950). Description of the Chemostat. Journal of Science, v. 112, p. 715.

N.R.C. TSS User's Manual. Nonlinear Regression and Nonlinear Equations (Marquardt). Regression, Distribution 01, 70.274,

p. 320.030.04.

POLAK, E. (1971). " Computational Methods in Optimization ". Academic Press, New York and Toronto.

PONTRYAGIN, L.S., BOLTYANSKII, V.G., GAMKRELIDZE, R.V. and MISHCHENKO, E.F. (1962). " The Mathematical Theory of Optimal Processes ". Interscience, New York and Toronto.

PRESS, S.J. (1972). " Applied Multivariate Analysis ". Holt, Rinehart and Winston, New York and Toronto.

RAI, V.R. and CONSTANTINDES, A. (1974). Mathematical Modeling and Optimization of the Gluconic Acid Fermentation. American Institute of Chemical Engineering Symposium Series, v. 69, p. 114.

RAJBMAN, N.S. (1973). Survey of Applications of Identification Methods in the U.S.S.R.. Proceedings of the 3rd I.F.A.C. Symposium on Identification and System Parameter Estimation, The Hague.

RAMAKER, B.L., SMITH, C.L. and MURRIL, P.L. (1967). Determination of Dynamic Model Parameters Using Quasilinearization. Technical Report # 6, T-TR-6, Louisiana State University, Baton Rouge, Louisiana.

RAMANATHAN, M. and GAUDY, A.F. Jr. (1969). Effect of High

Substrate Concentration and Cell Feedback on Kinetic Behaviour of Heterogenenous Populations in Completely Mixed Systems. Biotechnology and Bioengineering, v. XI, p. 207.

RAMANATHAN, M. and GAUDY, A.F. Jr. (1971). Steady State Model for Activated Sludge with Constant Recycle Sludge Concentration. Biotechnology and Bioengineering, v. XIII, p. 125.

SAGE, A.P. (1968). " Optimum Systems Control ". Prentice Hall, Englewood Cliff and Toronto.

SAGE, A.P. and MELSA, J.L. (1971a). " Estimation Theory with Applications to Communications and Control ". McGraw Hill, New York and Toronto.

SAGE, A.P. and MELSA, J.L. (1971b). " System Identification ". Academic Press, New York and Toronto.

SHAH, P. (1972). " Modeling, Analysis and Control of Biochemical Reactors ". Ph.D. Dissertation, Kansas State University, Manhattan, Kansas.

SIMON, H.A. (1956). Dynamic Programming Under Uncertainty with a Quadratic Criterion Function. Econometrica, v. 24, p. 74.

SMITH, F.B. Jr. and SHANNO, D. (1971). An Improved Marquardt Procedure for Nonlinear Regression. Technometrics, v. 13, p. 63.

SNELL, F.D. and SNELL, C.T. (1957). " Colorimetric Methods of Analysis, v. III ". Van Nostrand, New York.

SOONG, T.T. (1973). " Random Differential Equations in Science and Engineering ". Academic Press, New York and Toronto.

SPEEDY, C.B., BROWN, R.F. and GOODWIN, G.C. (1970). " Control Theory : Identification and Optimal Control ". Oliver and Boyd, Edinburgh.

SPEYER, J.L. and GUSTAFSON, D.E. (1974). Stochastic Optimal Control of Linear Dynamic Systems. A.I.A.A. Journal, v. 12, p. 1013.

STARK, P.A. (1970). " Introduction to Numerical Methods ". MacMillan, London and Toronto.

STRATONOVICH, R.L. (1960). Conditional Markov Processes. Theory of Probability and Its Applications, v. 5, p. 156.

SVRCEK, W.Y., ELLIOT, R.F. and ZAJIC, J.E. (1974). The Extended Kalman Filter Applied to a Continous Culture Model. Biotechnology and Bioengineering, v. XVI, p. 827.

TAKAMATSU, T., HASHIMOTO, I., SHIOYA, S., MIZUHARA, K., KOIKEE, T. and OHNO, H. (1975). Theory and Practice of Optimum Control in Continuous Fermentation Process. Automatica, v. 11, p. 141.

THEIL, H. (1957). A Note on Certainty Equivalence in Dynamic Planning. Econometrica, v. 25, p. 346.

TOPIWALA, H. and SINCLAIR, J.G. (1971). Temperature Relationship in Continuous Culture. Biotechnology and Bioengineering, v. XII, p. 795.

UNBEHAUEN, H and GOHRING, B. (1974). Application of Different Statistical Tests for the Determination of the Most Accurate Order of the Model in Parameter Estimation. Automatica, v. 10, p. 233.

VAN-DEN-BOOM, A.J.W. and VAN-DEN-ENDEN, A.W. (1974). The Determination of the Order of Process and Noise Dynamics. Automatica, v. 10, p. 245.

VISHWAKARMA, K.P. (1974). "Macro-Economic Regulation". Rotterdam University Press, Rotterdam, Netherlands.

WALD, A. (1949). Note on the Consistency of the Maximum Likelihood Estimate. Annals of Mathematical Statistics, p. 595.

WALLING, D. (1968). Nonlinear Least Squares Curves Fitting When some Parameters Are Linear. Texas Journal of Science, v. 20, p. 119.

WIENER, N. (1949). "The Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Appli-

cation ". Wiley, New York and Toronto. Originally issued as a Classified Report by M.I.T. Radiation Laboratory, Cambridge, Massachusetts, 1942.

WONHAM, W.M. (1963). Stochastic Problems in Optimal Control. I.E.E.E. International Convention Record, v. 11, p. 114.

WONHAM, W.M. (1968). On the Separation Theorem of Stochastic Control. S.I.A.M. Journal of Control, v. 6, p. 312.

WONNACOTT, A.J. and WONNACOTT, T.H. (1970). " Econometrics ". John Wiley and Sons, New York and Toronto.

WOODSIDE, C.M. (1970). A Dynamic Model for Electric Melting of Continuous Fed Prereduced Iron Pellets. The 99th Annual Meeting of the A.I.M.E..

WOODSIDE, C.M. (1971). Estimation of the Order of Linear Systems. Automatica, v. 7, p. 727.

WOODSIDE, C.M. and XUYEN, V.T. (1973). An Algorithm for Identification of Static and Dynamic Parameters in Batch Processing. Preprints of the 1973 Canadian Conference on Automatic Control, Paper 11.4, Federicton, New Brunswick.

ZUPP, G.A. and CHILDS, S.B. (1969). Applications of Quasi-linearization Theory to System Identification. N.A.S.A. Technical Note # NASA-TN-D-5300, Manned Spacecraft Centre, Houston, Texas.

VITA

NAME

VƯƠNG THANH XUYÊN

DATE & PLACE OF BIRTH

June 21, 1948 Phnom Penh, CAMBODIA

EDUCATION

(1) Trường Trung Học Hồ-Ngọc-Cẩn,
(Hồ-Ngọc-Cẩn High School),
Gia-Định, VIETNAM,
Sept. 1959 - June 1966.
(Science).

(2) Trường Quốc-Gia Kỹ-Sư Phú-Thọ,
(Phú-Thọ National School of Engineering),
Sài-Gòn, VIETNAM.
Sept. 1966 - Feb. 1967.
(Electrical Engineering).

(3) California State University at Sacramento,
(Formerly Sacramento State College),
Sacramento, California, U.S.A..
Feb. 1967 - Jan. 1971.
B.S.E.E. with Honors (Electrical Engineering).

(4) California State University at Long Beach,
(Formerly California State College, Long Beach),
Long Beach, California, U.S.A..
Summer 1968.
(General Education).

(5) Michigan State University
East Lansing, Michigan, U.S.A..
Feb. 1971.
Seminar Certificate (Communication Art).

(6) Carleton University,
Ottawa, Ontario, CANADA.
Sept. 1971 - Sept. 1973.
M. Eng. (Systems Engineering).

(7) University of Western Ontario,
London, Ontario, CANADA.
Sept. 1973 - Present.
Ph.D. (Systems Engineering).

HONORS & AWARDS

(1) Representative of the Hồ-Ngọc-Cẩn High School in the Annual National Competition in Mathematics, 1966.

(2) Representative of the California State University at Sacramento in the Annual North American William L. Putnam Competition in Mathematics, 1968.

(3) Recipent of the United States' A.I.D. Scholarships, 1967 - 1971.

(4) Recipent of the Ontario Graduate Fellowships, 1971 - 1973.

PROJECTS & THESES

(1) "Phép Nghịch Đảo và Chùm Vòng Tròn", ("The Inverse Transformation and Families of Circles"). Grade 12 Honor Project (1966), Supervisor : Mr. Hồ Văn Hòa, Hồ-Ngọc-Cẩn High School, Gia-Định, VIETNAM.

(2) "Design of a Low Ripple Power Supply Using a Digital Computer", Senior Project (1970), Supervisors : Dr. K.E. Stoffers & Dr. J. Vreeland, Department of Electrical Engineering, California State University at Sacramento, Sacramento, California, U.S.A..

(3) "Dynamic Batch Process Identification", M. Eng. Thesis (1973), Supervisor : Dr. C.M. Woodside, Department of Systems Engineering, Carleton University,

Ottawa, Ontario, CANADA.

(4) "Identification and Control Applied to Fermentation". Ph.D. Thesis (1976), Supervisors : Dr. J.E. Zajic & Dr. W.Y. Svrcek, Faculty of Engineering Science, University of Western Ontario, London, Ontario, CANADA.

SCIENTIFIC PAPERS

(1) C.M. WOODSIDE and V.T. XUYEN (1973). An Algorithm for Identification of Static and Dynamic Parameters in Batch Processing. Preprints of the 1973 Canadian Conference on Automatic Control, Paper 11.4, Fredericton, New Brunswick, CANADA.

(2) V.T. XUYEN and A. LEDUY (1974). Dynamic Batch Process Identification for Cell Growth. The 24th Canadian Chemical Engineering Conference, Ottawa, Ontario, CANADA.

(3) J.E. ZAJIC, W.Y. SVRCEK and V.T. XUYEN (1974). Kinetics of Methane Production in One and Two Stage Systems. Proceedings on Uses of Agricultural Wastes, Regina, Saskachewan, CANADA.

(4) J.E. ZAJIC, V.T. XUYEN and W.Y. SVRCEK (1975). Optimum Operating Conditions of Continuous Bioreactor

- A Theoretical Approach. Development in Industrial Microbiology, v. 16, p. 437. Also presented at the 1974 Annual Meeting of S.I.M., Memphis, Tennessee, U.S.A..

(5) V.T. XUYEN and W.Y. SVRCEK (1975). Elimination of Initial Conditions in Linear Dynamic Batch Process Identification by Least Squares. Canadian Journal of Chemical Engineering, v. 53, p. 443. Also presented at the 24th Canadian Chemical Engineering Conference, Ottawa, Ontario, CANADA, 1974.

(6) V.T. XUYEN, W.Y. SVRCEK and J.E. ZAJIC (1975). The Separation Principle Applied to Bioreactor Control. The 25th Canadian Chemical Engineering Conference, Montreal, Quebec, CANADA.

(7) V.T. XUYEN and W.Y. SVRCEK (1976). On Equivalence of the Gauss-Newton Technique, the Parameter Influence Coefficient (Sensitivity) Technique and the Quasi-linearization Technique in Dynamic System Identification by Least Squares. Accepted for publication in the Journal of Optimization Theory and Applications.

(8) W.Y. SVRCEK, V.T. XUYEN and J.E. ZAJIC (1976). Development of a Stochastic Model and a Regulating Control Policy for a Continuous Fermentation Process. Being submitted for publication.