OPTIMIZATION OF MANAGEMENT SYSTEMS

BY SECOND VARIATION

by 45

SHASHIKANT KRISHNARAO RANGNEKAR

B.E. (Mechanical Engineering), Bombay University
Bombay, India, 1966

A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1969

Approved by

G. Shing Lee

Major Professor

TABLE OF CONTENTS

1. INTRODUCTION

Optimization techniques can be divided into two classes, single stage and multistage. In multistage optimization techniques, a certain relationship is used to isolate the interconnections between the various stages. Thus one stage is searched at a time instead of all the N stages simultaneously. In this way, an N-dimensional problem is converted into N one-dimensional problems if the problem has only one control variable. The multistage optimization techniques can be classified into classical techniques (calculus of variation) and dynamic programming.

In case of calculus of variation, the resulting equations form a two-point boundary value problem (2,15). The differential equations encountered in practical applications are generally nonlinear and cannot be solved analytically. Finding numerical answers for this nonlinear boundary value problem is very tedious especially if there is a large number of equations with a large number of initial values missing. This has limited the use of the calculus of variation.

The maximum principle is a very powerful tool for obtaining analytical solutions of linear optimization problems with inequality constraints on control variable (7). But when the problem is nonlinear and an analytical solution cannot be obtained, the maximum principle gives rise to similar boundary value difficulties.

Dynamic programming, although free from the boundary value difficulty, has a serious drawback because of its storage requirements on the computer. Instead of solving any individual process, the dynamic programming technique solves a family of related processes (20). In here, as in the other multistage techniques, the problem of an N dimensional search is reduced to N one-dimensional search problems if the problem has only one control variable. However, in investigating one stage at a time, all possible combinations of the stage variables for the previously calculated stage must be stored in the memory of the computer.

This storage requirement, often referred to as the "curse of dimensionality," becomes too excessive to permit the use of dynamic programming for a problem in which more than three state variables are involved. Thus, if a three-dimensional problem, i.e. involving three state variables, is to be solved and if it is decided to have each state variable discretized into 50 values, then because of the interpolation required in the dynamic programming approach, $(50)^3$ values have to be stored. Thus it is frequently impossible to handle even a three-state variable problem with straight forward dynamic programming.

Thus it is seen that the dimensionality difficulty in dynamic programming and the boundary value problem in the classical methods limit the number of state variables in a problem that can be treated by these techniques. It should be noted, however, that these two difficulties are totally different from each other. The dimensionality difficulty requires more computer memory while the boundary value demands more computer time. Also, the classical boundary value problem approach represents an iterative procedure to obtain the numerical solution while dynamic programming represents an expansion of the original problem.

2. GRADIENT TECHNIQUES

The methods of gradients seem to remove the difficulties experienced in dynamic programming and the classical multistage techniques. Although there are various approaches with these methods, the basic philosophy remains the same. When use of the gradient methods is contemplated, the problem is formulated as a final value problem. In other words, the performance index or the objective function is selected as the value of some function at the end of the process. This is not a serious restriction. Thus if the performance index is

$$J = \int_0^{t_f} f(\underline{x})dt$$

Then

$$\frac{\partial J}{\partial t} = f(x)$$

Introducing an additional state variable x_{n+1}

$$\frac{dx_{n+1}}{dt} = f(n)$$

and $x_{n+1}(t_0) = 0$.

The original integral performance criterion is replaced by a criterion which calls for extremizing the final value of an element of the state vector. Philosophically at least, extremization of any performance criterion should be possible by using the following approach underlying the methods of gradients.

First a sequence of values of control vector is taken. Then a computation is made of the gradient of the performance index with respect to each control vector. Next each control vector is improved by moving it in the appropriate direction along the individual gradients. This improved sequence of control vectors then becomes the basis for the next iteration.

In the following sections, the first variation method, a technique suitable for optimizing nonlinear complex problems, is summarized. Then the second variation method, which is more sophisticated than the first variation method, is discussed. Three applications of this method in the field of production planning and control illustrate the advantages and disadvantages of this method.

2.1 The First Variation Method

Because of its computational appeal, various versions of the gradient methods have been developed for optimization calculations. A gradient technique for the numerical solution of dynamic optimization problems is generally known as the functional or serial gradient technique. This technique has been applied successfully to solve problems in aerospace, control and chemical engineering systems (5,6,10,16,17,20,21). The continuous version of the functional gradient technique was developed independently by Kelley(10) and by Brayson and his coworkers (5). A comprehensive treatment of this technique and of the gradient methods in general can be found in the article by Kelley(21).

In this method, the convergence is generally independent of the initial guess used in the iterative procedure, although the rate of convergence or, alternatively, the computer time, is affected by the initial guess. The number of equations to be integrated in the forward direction is (n+1); i.e. these equations are integrated from $t=0$ to $t=t_f$. There are (n+1) recursive equations. There are, however, no equations to be integrated in the backward direction from $t=t_f$ to $t=0$. The first variation equations are simpler than those of the second variation method.

The main drawback of the first variation method is that a very large number of iterations must be made in order to approach the optimal trajectory. More important is the fact that the trajectory approaches the optimum but does not actually reach it within a finite number of iterations. In some cases, the trajectory is far from the optimum after a large number of iterations and the rate of convergence becomes too slow to permit further iterations. This method cannot conveniently handle the problems with inequality constraints on the state variables.

2.2 Second Variation Method

The pioneer work in the area of second variation method has been carried out by Bryson and his coworkers (4,5), Kelley and his coworkers (10,11), Merriam (25) and Jaswinski (9). Mitter (26) and Breakwell and Ho (8) have also added to the work in this field.

This method is a natural evolution of the first order linearizations used in the first variation method in which the equations are linearized by truncating after all linear terms. The second order and higher order terms are thus ignored. It is well-known that the use of a linear approximation in a gradient search procedure is an excellent means for arriving near the optimum point quickly and from almost and stationary starting point. Near the optimum, however, the linear approximation becomes deficient and it is necessary to turn to a second order approximation to achieve the optimum. A useful optimization procedure is to initially use the first variation to get near the optimum trajectory and then to switch to the second order method for refinement.

2.3 Derivation of the Second Variation Method

Consider a process which can be represented by

$$\frac{d\underline{x}}{dt} = \underline{f}[\underline{x}(t), \underline{\theta}(t)] \tag{1}$$

where $\underline{x}$ is n dimensional state vector, $\underline{\theta}$ is r dimensional control vector and $\underline{x}(0)$ is prescribed. No terminal constraints are to be imposed on $\underline{x}(t_f)$, although the final time, t_f, may be specified.

Suppose it is desired to minimize the following performance index:

$$I[x(0),t_f] = I = \int_0^{t_f} J(\underline{x},\underline{\theta},t)dt \tag{2}$$

From Equation 2, this equation results

$$\frac{dI}{dt} = J(\underline{x},\underline{\theta},t) \tag{2A}$$

Since the performance index as given by Equation 2 is subject to the system constraints of Equation 1, consider the minimization of the unconstrained performance index as

$$I^* = I + \int_0^{t_f} \underline{z}'(\underline{f} - \frac{d\underline{x}}{dt})dt \tag{3}$$

where $\underline{z}$ is a vector of n Lagrangin multipliers. Substituting Equation 2 into Equation 3 results in

$$I^* = \int_0^{t_f} [J(\underline{x},\underline{\theta},t) + \underline{z}'(\underline{f} - \frac{d\underline{x}}{dt}]dt \tag{4}$$

In order to minimize I^*, an iteration algorithm can be constructed such that

$$I^{*(j+1)} = \int_0^{t_f} \left\{ J^{(j+1)} + \underline{z}'^{(j+1)} \left(\underline{f}^{(j+1)} - \frac{d\underline{x}^{(j+1)}}{dt} \right) \right\} dt \tag{5}$$

converges in a desirable way. The superscript (j+1) is used to indicate the number of iteration, and it is desired to have

$$I^{*(0)} > I^{*(1)} > \dots > I^{*(j)} > I^{*(j+1)} > \dots \tag{6}$$

To construct the desired iterative algorithm, the values of the functions at iteration (j+1) can be expressed in terms of the j^{th} iteration by means of Taylor's series expansion. Retaining only the terms up to the second order gives

$$J^{(j+1)} \simeq J^{(j)} + \left(\frac{\partial J^{(j)}}{\partial \underline{x}^{(j)}}\right) \delta \underline{x}^{(j)} + \left(\frac{\partial J^{(j)}}{\partial \underline{\theta}^{(j)}}\right) \delta \underline{\theta}^{(j)}$$

$$+ \frac{1}{2} \delta \underline{x}^{(j)'} \frac{\partial^2 J^{(j)}}{\partial \underline{x}^{(j)2}} \delta \underline{x}^{(j)} + \delta \underline{\theta}^{(j)'} \frac{\partial^2 J^{(j)}}{\partial \underline{\theta}^{(j)} \cdot \partial \underline{x}^{(j)}} \delta \underline{x}^{(j)}$$

$$+ \frac{1}{2} \delta \underline{\theta}^{(j)'} \frac{\partial^2 J^{(j)}}{\partial \underline{\theta}^{(j)2}} \delta \underline{\theta}^{(j)} \tag{7}$$

where,

$$\begin{aligned} \delta \underline{x}^{(j)} &= x^{(j+1)} - x^{(j)} \\ \delta \underline{\theta}^{(j)} &= \underline{\theta}^{(j+1)} - \underline{\theta}^{(j)} \end{aligned} \tag{8}$$

$$\frac{\partial^2 J}{\partial \underline{x}^2} = \begin{bmatrix} \frac{\partial^2 J}{\partial x_1^2} & \frac{\partial^2 J}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 J}{\partial x_1 \partial x_n} \\ \vdots & & & \\ \frac{\partial^2 J}{\partial x_n \partial x_1} & & \cdots & \frac{\partial^2 J}{\partial x_n^2} \end{bmatrix},$$

$$\frac{\partial^2 J}{\partial \underline{\theta} \partial \underline{x}} = \begin{bmatrix} \frac{\partial^2 J}{\partial \theta_1 \partial x_1} & \cdots & \frac{\partial^2 J}{\partial \theta_1 \partial x_n} \\ \vdots & & \\ \frac{\partial^2 J}{\partial \theta_r \partial x_1} & \cdots & \frac{\partial^2 J}{\partial \theta_r \partial x_n} \end{bmatrix} \quad (9)$$

The superscript (j) has been omitted in Equation (9) for clarity. Thus it is seen that

$$\delta \underline{\theta}' \frac{\partial^2 J}{\partial \underline{\theta}\, \partial \underline{x}} \delta \underline{x} = \sum_{i=1}^{n} \sum_{j=1}^{r} \frac{\partial^2 J}{\partial \theta_j \partial x_i} \delta \theta_j \; \delta x_i \quad (10)$$

Next, define the Hamiltonian

$$\bar{H} = \underline{z}' \; \underline{f} \quad (11)$$

and expand $\bar{H}$ at the $(j+1)^{th}$ iteration up to the second order terms as a function of $\bar{H}$ at the j^{th} iteration. Note that $\bar{H}$ is a function of $\underline{x}$, $\underline{\theta}$, and $\underline{z}$ and that $\frac{\partial^2 H}{\partial z^2} = 0$

$$\bar{H}^{(j+1)} = \bar{H}^{(j)} + \left(\frac{\partial \bar{H}^{(j)}}{\partial \underline{x}^{(j)}}\right)' \delta \underline{x}^{(j)} + \left(\frac{\partial \bar{H}^{(j)}}{\partial \underline{\theta}^{(j)}}\right)' \delta \underline{\theta}^{(j)}$$

$$+ \left(\frac{\partial \bar{H}^{(j)}}{\partial \underline{z}^{(j)}}\right)' \delta \underline{z}^{(j)} + \frac{1}{2}\, \delta \underline{x}^{(j)\prime} \frac{\partial^2 \bar{H}}{\partial x^{(j)2}}\, \delta x^{(j)} \tag{12}$$

$$+ \delta \underline{\theta}^{(j)\prime} \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{\theta}^{(j)} \partial \underline{x}^{(j)}}\, \delta \underline{x}^{(j)} + \frac{1}{2}\, \delta \theta^{(j)\prime} \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{\theta}^{(j)2}}\, \delta \underline{\theta}^{(j)}$$

$$+ \delta \underline{\theta}^{(j)\prime} \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{\theta}^{(j)} \partial \underline{z}^{(j)}}\, \delta \underline{z}^{(j)} + \delta \underline{x}^{(j)\prime} \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{x}^{(j)} \cdot \partial \underline{z}^{(j)}}\, \delta z^{(j)}$$

Now consider the nonlinear performance equations. If these equations are linearized by Taylor-series expansions and by retaining only the first order terms, the result is

$$\frac{d\underline{x}^{(j)}}{dt} = \left(\partial \underline{f}^{(j)\prime} / \partial \underline{x}^{(j)}\right)' \delta \underline{x}^{(j)} + \left(\partial \underline{f}^{(j)\prime} / \partial \underline{\theta}^{(j)}\right)' \delta \underline{\theta}^{(j)} \tag{13}$$

with $\delta \underline{x}(0) = \underline{0}$ since the initial conditions are constant. This last equation may be rearranged by noting that

$$\frac{d\underline{x}^{(j+1)}}{dt} = \delta\left(\frac{d\underline{x}^{(j)}}{dt}\right) + \underline{f}^{(j)} \tag{14}$$

Thus Equation 13 can be rewritten as

$$\frac{d\underline{x}^{(j+1)}}{dt} = \underline{f}^{(j)} + \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{z}^{(j)}\, \partial \underline{x}^{(j)}} \delta x^{(j)} + \frac{\partial^2 \bar{H}^{(j)}}{\partial \underline{z}^{(j)} \partial \underline{\theta}^{(j)}} \delta \underline{\theta}^{(j)} \qquad (15)$$

Furthermore,

$$\underline{z}^{(j+1)} = \underline{z}^{(j)} + \underline{P}^{(j)}\, \delta \underline{x}^{(j)}$$

so that $\delta \underline{z}^{(j)} = \underline{P}^{(j)}\, \delta \underline{x}^{(j)}$ (16)

where the matrix $\underline{P}$ is defined by

$$\underline{P} = \begin{bmatrix} \frac{\partial z_1}{\partial x_1} & \cdots & \frac{\partial z_1}{\partial x_n} \\ \vdots & & \\ \frac{\partial z_n}{\partial x_1} & \cdots & \frac{\partial z_n}{\partial x_n} \end{bmatrix} = \left(\frac{\partial \underline{z}'}{\partial \underline{x}}\right)' \qquad (17)$$

It is a symmetrical matrix,

$$\text{i.e.} \quad \frac{\partial z_i}{\partial x_j} = \frac{\partial z_j}{\partial x_i}$$

Clearly $\underline{P}$ is unknown explicitly at this point. For the sake of clarity, the superscript (j) is omitted in the subsequent derivation.

If now the normal Hamiltonian function is defined as $H = J + \underline{z}'\, \underline{f}$, then the above expressions can be substituted into Equation 4 to yield

$$I^{*(j+1)} = I^{*} + \int_0^{t_f} \left\{ \left(\frac{\partial H}{\partial \underline{x}}\right)' \delta\underline{x} + \left(\frac{\partial H}{\partial \underline{\theta}}\right)' \delta\underline{\theta} + \frac{1}{2}\,\delta\underline{x}' \frac{\partial^2 H}{\partial \underline{x}^2}\,\delta\underline{x} \right.$$

$$+ \delta\underline{\theta}' \frac{\partial^2 H}{\partial \underline{\theta}\,\partial \underline{x}}\,\delta\underline{x} + \frac{1}{2}\,\delta\underline{\theta}' \frac{\partial^2 H}{\partial \underline{\theta}^2}\,\delta\underline{\theta} + \delta\theta' \frac{\partial \underline{f}'}{\partial \underline{\theta}}\,\underline{P}\,\delta\underline{x} \qquad (18)$$

$$\left. + \delta\underline{x}' \frac{\partial \underline{f}'}{\partial x}\,\underline{P}\,\delta\underline{x} - \underline{z}'\delta\,\frac{d\underline{x}}{dt} - \delta\underline{x}'\,\underline{P}\,\delta\,\frac{d\underline{x}}{dt} \right\} dt$$

To further simplify Equation 18, use of the adjoint equation is made. Thus

$$\frac{d\underline{z}}{dt} = -\frac{\partial J}{\partial \underline{x}} - \frac{\partial \underline{f}'}{\partial x}\,\underline{z} \qquad (19)$$

This is easily obtained by defining

$$M = \min_{\theta}\ [J\ (\underline{x},\underline{\theta},t) + z'f] \qquad (20)$$

where the adjoint variable $\underline{z}$ is defined by

$$\underline{z} = \frac{\partial I^{o}}{\partial \underline{x}} \qquad (21)$$

But from the principle of optimality in dynamic programming, it follows that for

$$I^{o}(\underline{x},\ t) = \min_{\underline{\theta}} \int_0^{t_f} J(\underline{x},\underline{\theta},\lambda)\ d\lambda \qquad (22)$$

that

$$I^o(\underline{x},t) = \min_{\underline{\theta}} \left(\int_t^{t+\Delta t} J(\underline{x},\underline{\theta},\lambda)d\lambda + \int_{t+\Delta t}^{t_f} J(\underline{x},\underline{\theta},\lambda)d\lambda \right)$$

$$= \min_{\underline{\theta}} \int_t^{t+\Delta t} J(\underline{x},\underline{\theta},\lambda)d\lambda + I^o\left(\underline{x} + \frac{d\underline{x}}{dt}\Delta t,\ t + \Delta t)\right) .$$

As Δt approaches zero

$$I^o(\underline{x},t) = \min_{\underline{\theta}} \left(J(\underline{x},\underline{\theta},t)\ \Delta t + J^o(\underline{x},t) + \left(\frac{\partial I^o}{\partial \underline{x}}\right) \frac{d\underline{x}}{dt} \Delta t + \frac{\partial I^o}{\partial t} \cdot \Delta t\right)$$

i.e. $$J^o(\underline{x},\underline{\theta}^o,t) + \left(\frac{\partial I^o}{\partial \underline{x}}\right)' \frac{d\underline{x}}{dt} + \frac{\partial I^o}{\partial t} = 0$$

which may be written as

$$M + \frac{\partial I^o}{\partial t} = 0 \qquad (23)$$

The partial differentiation of Equation 23 w.r.t. $\underline{x}$ yields

$$\frac{\partial M}{\partial \underline{x}} + \frac{\partial^2 I^o}{\partial \underline{x} \cdot \partial t} = \underline{0}$$

or

$$\frac{\partial M}{\partial \underline{x}} + \frac{\partial \underline{z}}{\partial t} = \underline{0} . \qquad (24)$$

However, the total time derivative of $\underline{z}$ is

$$\frac{d\underline{z}}{dt} = \frac{\partial \underline{z}}{\partial t} + \left(\frac{\partial \underline{z}'}{\partial \underline{x}}\right)' \frac{d\underline{x}}{dt}$$

$$= -\frac{\partial J^o}{\partial \underline{x}} - \frac{\partial(\underline{z}'f)}{\partial \underline{x}} + \left(\frac{\partial \underline{z}'}{\partial \underline{x}}\right)' \frac{d\underline{x}}{dt} . \qquad (25)$$

Since

$$\frac{\partial J}{\partial \underline{\theta}} + \frac{\partial}{\partial \underline{\theta}} (\underline{z}'\underline{f}) = \underline{0} \tag{26}$$

due to the optimality condition. Expanding Equation 25 gives Equation 19, namely

$$\frac{d\underline{z}}{dt} = -\frac{\partial J^o}{\partial \underline{x}} - \frac{\partial \underline{f}'}{\partial \underline{x}} \underline{z} \tag{27}$$

Similarly,

$$\frac{d\underline{P}}{dt} = -\frac{\partial^2 J}{\partial \underline{x}^2} - \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{x}^2} - \left\{ \underline{P} \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right)' + \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right) \underline{P} \right\} + \underline{K}\,\underline{P} \tag{28}$$

where

$$\underline{K} = -\frac{\partial \underline{\theta}'}{\partial \underline{x}} \tag{29}$$

and

$$\underline{P} = \frac{\partial^2 J}{\partial \underline{\theta} \cdot \partial \underline{x}} + \frac{\partial \underline{f}'}{\partial \underline{\theta}} \underline{P} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}\, \partial \underline{x}} \quad . \tag{30}$$

To evaluate $\underline{K}$ it may be noted from Equation 17 that

$$\frac{\partial J}{\partial \underline{\theta}} + \frac{\partial \underline{f}'}{\partial \underline{\theta}} \underline{z} = \underline{0}$$

so that partially differentiating w.r.t. x gives

$$\left(\frac{\partial \underline{\theta}'}{\partial \underline{x}}\right) \frac{\partial^2 J}{\partial \underline{\theta}^2} + \frac{\partial^2 J}{\partial \underline{x} \cdot \partial \underline{\theta}} + \left(\frac{\partial \underline{\theta}'}{\partial \underline{x}}\right) \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta} \cdot \partial \underline{x}} + \frac{\partial \underline{z}'}{\partial \underline{x}} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right)' = \underline{0}$$

i.e. $$\frac{\partial \underline{\theta}'}{\partial \underline{x}} \left(\frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2} \right) = - \underline{R}'$$

and $$\underline{K} = \underline{R}' \left(\frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2} \right)^{-1} \quad . \tag{32}$$

Therefore it is possible to solve for $\frac{\partial^2 J}{\partial \underline{x}^2}$, namely

$$\frac{\partial^2 J}{\partial \underline{x}^2} = - \frac{d\underline{P}}{dt} - \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{x}^2} - \left\{ \underline{P} \left(\frac{\partial \underline{f}'}{\partial \underline{x}} \right)' + \left(\frac{\partial \underline{f}'}{\partial \underline{x}} \right) \underline{P} \right\} + \underline{K}\ \underline{R} \tag{33}$$

Now, substituting Equations 27 and 33 into Equation 18 yields

$$I^{*(j+1)} \simeq I^{*(j)} + \int_0^{t_f} \left\{ \frac{1}{2} \delta\underline{\theta}' \left(\frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2} \right) \delta\underline{\theta} \right.$$

$$+ \left(\left(\frac{\partial J}{\partial \underline{\theta}} \right)' + \sum_{i=1}^{n} z_i \left(\frac{\partial f_i}{\partial \underline{\theta}} \right)' \right) \delta\underline{\theta}$$

$$\left. + \delta\underline{\theta}' \ \underline{R} \ \delta\underline{x} + \frac{1}{2} \delta\underline{x}' \ \underline{K} \ \underline{R} \ \delta\underline{x} \right\} dt \quad . \tag{34}$$

In order that the performance index will converge to a minimum, the integral in Equation 34 must be less than zero, i.e.

$$\int_0^{t_f} \left\{\frac{1}{2}\,\delta\underline{\theta}'\left(\frac{\partial^2 J}{\partial\underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial\underline{\theta}^2}\right)\delta\underline{\theta} + \left[\left(\frac{\partial J}{\partial\underline{\theta}}\right)' + \sum_{i=1}^{n} z_i\left(\frac{\partial f_i}{\partial\underline{\theta}}\right)'\right]\delta\underline{\theta}\right.$$

$$\left. + \delta\underline{\theta}' \cdot \underline{R} \cdot \delta\underline{x} + \frac{1}{2}\,\delta\underline{x}' \cdot \underline{K} \cdot \underline{R} \cdot \delta\underline{x}\right\} dt < 0 \qquad (35)$$

In addition, the convergence idealy should be as fast as possible so the minimization of the integral is considered:

$$V(\delta\underline{x},\, t) = \int_t^{t_f} \left\{\frac{1}{2}\,\delta\underline{\theta}'\left(\frac{\partial^2 J}{\partial\underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial\underline{\theta}^2}\right)\delta\underline{\theta}\right.$$

$$+ \left[\left(\frac{\partial J}{\partial\underline{\theta}}\right)' + \sum_{i=1}^{n} z_i\left(\frac{\partial f_i}{\partial\underline{\theta}}\right)'\right]\delta\underline{\theta}$$

$$\left. + \delta\underline{\theta}'\,\underline{R}\,\delta\underline{x} + \frac{1}{2}\,\delta\underline{x}' \cdot \underline{K} \cdot \underline{R}\,\delta\underline{x}\right\} d\lambda \qquad (36)$$

Through the proper choice of $\delta\underline{\theta}$ and denoting the minimum by $V^o(\delta\underline{x}, t)$, since $V(\delta\underline{x}, t)$ as given by Equation 36 is quadratic in $\delta\underline{x}$, the minimum of $V(\delta\underline{x}, t)$ may be written as a quadratic expression, as

$$V^o(\delta\underline{x},\, t) = q(t) + (\underline{q}(t))'\,\delta\underline{x} + \delta\underline{x}'\,Q(t)\,\delta\underline{x} \qquad (37)$$

where $q(t)$ = scalar function of t

$\underline{q}(t)$ = (nxl) vector function of t

$Q(t)$ = (nxm) matrix function of t (symmetric)

and $q(t_f) = 0 \quad \underline{q}(t_f) = \underline{0} \quad \underline{Q}(t_f) = \underline{0}$.

From Equation 37

$$\frac{\partial V^o(\delta \underline{x},\ t)}{\partial t} = \frac{d\underline{q}(t)}{dt} + \left(\frac{d\underline{q}(t)}{dt}\right)' \delta\underline{x} + \delta\underline{x}' \frac{d\underline{Q}(t)}{dt} \cdot \delta\underline{x} \quad (38)$$

and

$$\frac{\partial V^o(\delta \underline{x},\ t)}{\partial \underline{x}} = \underline{q}(t) + 2\underline{Q}(t)\ \ \delta\underline{x} \quad (39)$$

Minimization of $V(\delta\underline{x},\ t)$ as given by Equation 36 gives

$$\frac{1}{2}\,\delta\underline{\theta}^{o\prime}\left(\frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2}\right)\delta\underline{\theta}^o + \left[\left(\frac{\partial}{\partial \underline{\theta}}\right)' + \sum_{i=1}^{n} z_i \left(\frac{\partial f_i}{\partial \underline{\theta}}\right)'\right]\delta\underline{\theta}^o$$

$$+\ \delta\underline{\theta}^{o\prime}\ R \cdot \delta\underline{x} + \frac{1}{2}\,\delta\underline{x}'\ \underline{K}\ \underline{R}\ \delta\underline{x}$$

$$+\ \left(\underline{q}'(t) + 2\delta\underline{x}'\underline{Q}(t)\right)\delta\left(\frac{dx}{dt}\right) + \frac{d\underline{q}(t)}{dt}$$

$$+\ \left(\frac{d\underline{q}(t)}{dx}\right)' \delta\underline{x}' + \delta\underline{x}' \frac{d\underline{Q}(t)}{dt}\,\delta\underline{x} = 0 \quad (40)$$

where

$$\delta\underline{\theta}^o = -\left(\frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2}\right)^{-1}\left[\left(\frac{\partial J}{\partial \underline{\theta}} + \sum_{i=1}^{n} z_i \frac{\partial f_i}{r\underline{\theta}}\right)\right.$$

$$\left.+\ \underline{R}\ \delta\underline{x} + \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right)\left(\underline{q}(t) + 2\underline{Q}(t)\delta\underline{x}\right)\right] \quad (41)$$

and

$$\frac{\delta d\underline{x}}{dt} = \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right) \delta \underline{x} + \left(\frac{\delta \underline{f}'}{\delta \underline{\theta}}\right) \delta \underline{\theta} \quad . \tag{42}$$

When the optimal control as given by Equation 41 is substituted into Equation 40 and the coefficients of $\delta \underline{x}$ and $\delta \underline{x}' \cdot \delta \underline{x}$ along with the terms not containing $\delta \underline{x}$ are all put equal to zero (to satisfy the identify for any $\delta \underline{x}$) the following results:

$$\frac{d\underline{q}}{dt} = \frac{1}{2} \left\{ \underline{S}' \, \underline{T}^{-1} \, \underline{S} + \underline{S}' \, \underline{T}^{-1} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{q} + \underline{q}' \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{T}^{-1} \underline{S} \right.$$

$$\left. + \, \underline{q}' \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right)' \underline{T}^{-1} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{q} \right\} \tag{43}$$

$$\frac{d\underline{q}}{dt} = \underline{R}' \, \underline{T}^{-1} \, \underline{S} + \underline{R}' \, T^{-1} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{q} - \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right)' \underline{q}$$

$$+ \, 2\underline{Q} \left[\left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right)' \underline{T}^{-1} \, \underline{S} + \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{T}^{-1} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{q} \right] \tag{44}$$

and

$$\frac{d\underline{Q}}{dt} = 2 \left\{ \underline{Q} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right)' \underline{T}^{-1} \, \underline{R} + \underline{Q} \left(\frac{\partial \underline{f}'}{\partial \theta}\right)' \underline{T}^{-1} \frac{\partial \underline{f}'}{\partial \theta} \underline{Q} - \underline{Q} \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right)' \right\} \tag{45}$$

where $\underline{S}$ and $\underline{T}$ are introduced to condense the notation and are given by

$$\underline{S} = \frac{\partial J}{\partial \underline{\theta}} + \sum_{i=1}^{n} z_i \frac{\partial f_i}{\partial \underline{\theta}} \tag{46}$$

$$\underline{T} = \frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2} \quad . \tag{47}$$

At this point it is noted that the matrix $\underline{Q}$ contributes only insignificantly to the control. Furthermore $\underline{Q}$ appears as a second order term itself. Therefore, to facilitate programming on the digital computer these $\frac{n(n+1)}{2}$ equations shall be discarded and $\underline{Q}$ shall be put equal to zero. Equation 43 is not required for the evaluation of control; therefore Equation 44 will take the form

$$\frac{d\underline{q}}{dt} = \underline{R}' \underline{T}^{-1} \underline{S} + \underline{R}' \underline{T}^{-1} \left(\frac{\partial \underline{f}'}{\partial \underline{\theta}}\right) \underline{q} - \left(\frac{\partial \underline{f}'}{\partial \underline{x}}\right) \underline{q} \tag{48}$$

and the change in the control simplifies to

$$\delta\underline{\theta} = -\underline{T}^{-1} \left(\underline{S} + \underline{R}\ \delta\underline{x} + \frac{\partial \underline{f}'}{\partial \underline{\theta}} \underline{q}\right). \tag{49}$$

To prevent overstepping in control adjustment, Memiam [23] has suggested the introduction of a constant ε where $0 < \varepsilon \leq 1$ in Equation 49 to give

$$\delta\underline{\theta} = -\varepsilon T^{-1}\left(\underline{S} + \frac{\partial \underline{f}'}{\partial \underline{\theta}} \underline{q}\right) - \underline{T}^{-1}\ \underline{R}\ \ \delta\underline{x}\ . \tag{50}$$

Thus it is now worthwhile to detail the application of the second variation method equations developed above.

(1) Assume a set of initial value for θ.

(2) Equations 1 and 2A are integrated forward from $t = 0$ to $t = t_f$; i.e. (n+1) equations are integrated forward in time, namely

$$\frac{d\underline{x}}{dt} = \underline{f}(\underline{x},\ \underline{\theta})$$

$$\frac{dI}{dt} = J(\underline{x},\ \theta,\ t)\ .$$

(3) While the integration is carried out, the values of $\underline{x}$ are retained in the computer memory at small time intervals to approximate the continuous system.

(4) The adjoint Equation 19 plus the additional Equations 28 and 48 are integrated backwards, i.e., $2n+\frac{n(n+1)}{2}$ equations are integrated backwards in time from t_f to 0, namely

$$\frac{d\underline{z}}{dt} = -\frac{\partial J}{\partial \underline{x}} - \left(\frac{\partial f'}{\partial \underline{x}}\right) \underline{z}$$

$$\frac{d\underline{P}}{dt} = -\frac{\partial^2 J}{\partial \underline{x}^2} - \sum_{i=1}^{n} z_i \frac{\partial^2 f_i}{\partial \underline{x}^2} - \left\{\underline{P}\left(\frac{\partial f'}{\partial \underline{x}}\right)' + \left(\frac{\partial f'}{\partial \underline{x}}\right) \underline{P}\right\} + \underline{R}' \, \underline{T} \, \underline{R}$$

$$\frac{d\underline{q}}{dt} = \underline{R}' \, \underline{T}^{-1} \, \underline{S} + \underline{R}' \, T^{-1} \left(\frac{\partial f'}{\partial \underline{\theta}}\right) \underline{q} - \left(\frac{\partial f'}{\partial \underline{x}}\right) \underline{q} \, .$$

(5) During the backward integration, the values of $\underline{T}$, $\underline{S}$, $\underline{q}$ and $\underline{R}$ are stored in computer memory.

(6) The new value of control is calculated from Equation 50, i.e.

$$\underline{\theta}^{(j+1)} = \underline{\theta}^{(j)} - \left\{\varepsilon \underline{T}^{-1}\left(\underline{S} + \frac{\partial \underline{f}'}{\partial \underline{\theta}} \underline{q}\right)\right\}^{(j)} - (\underline{T}^{-1}\underline{R})^{(j)} \left(\underline{x}^{(j+1)} - \underline{x}^{(j)}\right)$$

and steps 2-6 are carried out again.

(7) This iteration is continued until no further change in $\underline{\theta}$ is noticed or until the performance index does not change. The former is more sensitive [12].

If the performance index increases during some iteration, the parameter ε is halved and the iteration is continued.

For maximization problems, the derivation can be followed on the same lines and it will be seen that the resulting equations are the same.

2.4 Advantages and Disadvantages of the Second Variation Method.

The foremost advantage of the second variation method lies in its rapid convergence. Also, unlike the first variation, the optimum can be reached with reasonably high accuracy.

The theoretical attractiveness of this method, however, is more than offset by its disadvantages. First, and most important, the initially assumed trajectory of the control variable must be sufficiently close to the optimal trajectory for convergence to be obtained. Second, the number of equations to be integrated is considerably greater than required for the first variation method. In the second variation method, (n+1) equations are integrated in the forward direction, i.e. from t=0 to $t=t_f$, and (2n+n(n+1)/2) equations are integrated backwards where n is the number of state variables in the problem under consideration. The first variation method requires only (n+1) equations to be integrated in the forward direction and there are (n+1) recursive equations in the backward direction. Not only is the number of equations involved in the second variation method high but the equations themselves are more complicated. The main reason for this is that the calculations of all derivatives, both first and second order, becomes more and more tedious with the increasing number of state and control variables. All the multiplications are in terms of matrices. Again, the inverse of $\underline{T}$ has to be computed at each integration step in the backward integration. Hence the programming of the iteration scheme with the required equations can be quite complicated. Instability can arise from bad starting values, i.e. from an insufficiently good guess for the starting trajectory of the control variable. The values for the parameter ε have to be established by trial and error for the particular problem. The higher the value the faster the convergence. Finally, this technique cannot handle problems involving inequality constraints.

3. APPLICATIONS

To illustrate the use of the second variation method, three numerical problems in the field of production planning and control are solved in the following sections.

3.1 An Inventory Model

The Model

The following is a simple problem in the field of production scheduling and inventory control. Assume that the rate of sales Q(t) is known with certainty and that the rate of change of the inventory level I(t) is given by

$$\frac{dI(t)}{dt} = P(t) - Q(t) \tag{51}$$

where P(t) is the production rate at time t. The problem is to minimize the cost function.

$$C_T = \int_0^T \{C_I(I_M - I(t))^2 + C_P \exp (P_M - P(t))^2\} \, dt \tag{52}$$

where C_T is the total cost of inventory and production and C_P is the minimum production cost which occurs when the production rate equals P_M. The quantity P_M can be considered as the capacity of the manufacturing plant. Since the plant is designed for a capacity P_M, an increase in capacity may require additional equipment and manpower which, due to contract agreements cannot be reduced easily. I_M can be considered as the capacity for the storage of inventory and C_I is the inventory carrying cost. In many practical situations, the minimum storage cost is obtained when the storage capacity is completely filled. Furthermore, the cost function, Equation (2), has the smoothing capability which is frequently desirable for many manufacturing processes. In this case, I_M and P_M can be considered as the desirable inventory and production levels. It is further assumed that the sales forecast is known and is given by the linear relation

$$Q(t) = a+bt \tag{53}$$

and the initial inventory is

$$I(0) = c \tag{54}$$

Recursive Relations

This optimum production planning problem can be rewritten into the form required for the second variation method as

Let $x1(t) = I(t)$

$\theta(t) = P(t)$.

Equations (51) and (54) become

$$\frac{dx1(t)}{dt} = \theta(t) - a - b(t) \tag{55}$$

and $x1(0) = c$ (56)

Let

$$x2(t) = \int_0^t C_I(I_M - I(t))^2 + C_P \exp\,(P_M - \theta(t))^2\}dt \tag{57}$$

Then

$$x2(t) = C_T \tag{58}$$

$$\frac{dx2(t)}{dt} = C_I(I_M - x1(t))^2 + C_P \exp\,(P_M - \theta(t))^2 \tag{59}$$

$$x2(0) = 0 \tag{60}$$

Thus, in this problem there is one state variable, namely inventory x1. The control variable is the production rate $\theta(t)$. The numerical values used for this problem are:

$$a = 2 \qquad b = 1 \qquad c = 5$$

$$C_I = 0.1 \qquad I_M = 10 \qquad C_P = 0.001 \; .$$

$$P_M = 5 \qquad T = 1$$

The various derivatives required for obtaining the second variational equations are:

$$\frac{\partial J}{\partial \underline{x}} = \frac{\partial J}{\partial x_1} = - 2\, C_I \, (I_M - x1(t))$$

$$\frac{\partial^2 J}{\partial \underline{x}^2} = 2\, C_I$$

$$\frac{\partial J}{\partial \theta} = -2\, C_P \exp\, (P_M - \theta(t))^2 \cdot (P_M - \theta(t))$$

$$\frac{\partial^2 J}{\partial \underline{\theta}^2} = 2\, C_P \exp\, (P_M - \theta(t))^2 \;\; \{1 + 2\, (P_M - \theta(t))^2\}$$

$$\frac{\partial^2 J}{\partial \underline{\theta}\, \partial \underline{x}} = 0$$

$$\frac{\partial f_1}{\partial \underline{x}} = 0 \qquad \frac{\partial^2 f_1}{\partial \underline{x} \partial \underline{\theta}} = 0$$

$$\frac{\partial^2 f_1}{\partial \underline{x}^2} = 0 \qquad \frac{\partial f'}{\partial \underline{x}} = 0 \qquad \frac{\partial f'}{\partial \underline{\theta}} = 1$$

The expressions for the terms $\underline{R}$, $\underline{s}$, $\underline{T}$ are:

$$\underline{R} = \underline{P} = P \qquad \underline{P} \text{ being 1 dimensional}$$

$$\underline{s} = -2\, C_P \exp\,(P_M - \theta(t))^2\, (P_M - \theta(t)) + z_1$$

$$\underline{T} = 2\, C_P \exp\,(P_M - \theta(t))^2 \cdot \{1 + 2\,(P_M - \theta(t))^2\}$$

The second variational equations (19, 28, 48) become

$$\frac{d\underline{z}}{dt} = \frac{dz}{dt} = 2\, C_I\, [I_M - x_1(t)] \tag{61}$$

$$\frac{d\underline{P}}{dt} = \frac{dP}{dt} = -\,2\, C_I + P^2\, [2\, C_P \exp\,(P_M - \theta(t))^2\, \{1 + 2\,(P_M - \theta(t))^2\}] \tag{62}$$

$$\frac{d\underline{QF}}{dt} = \frac{dQF}{dt} = \frac{P\{-2C_P \exp(P_M - \theta(t))^2 (P_M - \theta(t)) + z + QF\}}{2C_P \cdot \exp[P_M - \theta(t)]^2\, \{1 + 2[P_M - \theta(t)]^2\}} \tag{63}$$

and

$$\theta^{(j+1)} = \theta^{(j)} - [\varepsilon(s + QF)]^{(j)} - [P^{(j)} (x_1^{(j+1)} - x_1^{(j)})] \,.$$

Thus Equations (61), (62) and (63) are the second variational equations and Equation (64) is the equation for finding the new value of the control.

Table 1

Effect of ε on the Rate of Convergence,

of Inventory, $\theta(t) = 7$, $x_1(t) = 5$, $0 \leq t \leq t_f$.

Iteration	ε = 0.1	ε = 0.3	ε = 0.5	ε = 0.7	ε = 1.0
1	9.49995	9.49995	9.49995	9.49995	9.49995
5	9.38148	9.39033	9.37763	9.36130	9.34562
10	9.39130	9.36100	9.33515	9.32687	9.32586
15	9.38672	9.33859	9.32642	9.32586	9.32586
20	9.32649	9.32585	9.32588	9.32586	9.32586
25	9.32597	"	9.32585	9.32586	9.32586
30	9.32587	"	"	"	"
35	9.32585	"	"	"	"
40	"	"	"	"	"
45	"	"	"	"	"
50	"	"	"	"	"
55	"	"	"	"	"

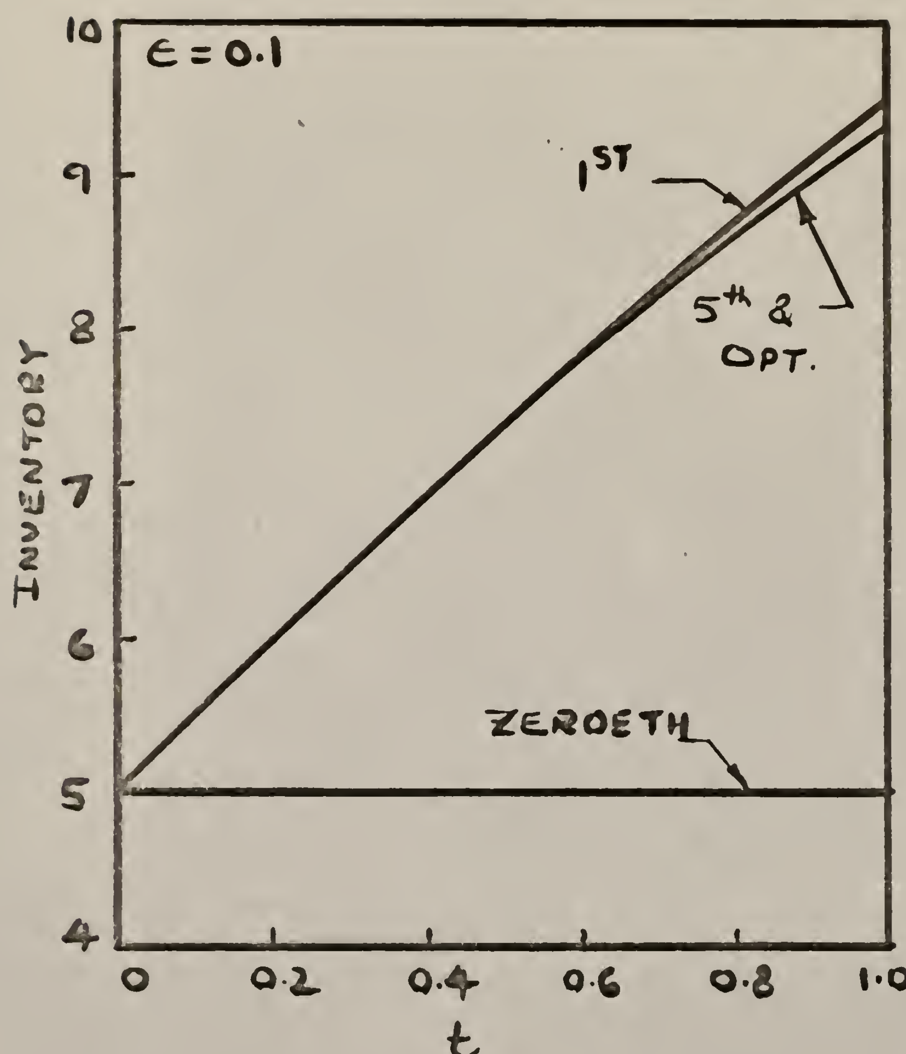

Fig. 1 Convergence rate of Inventory.

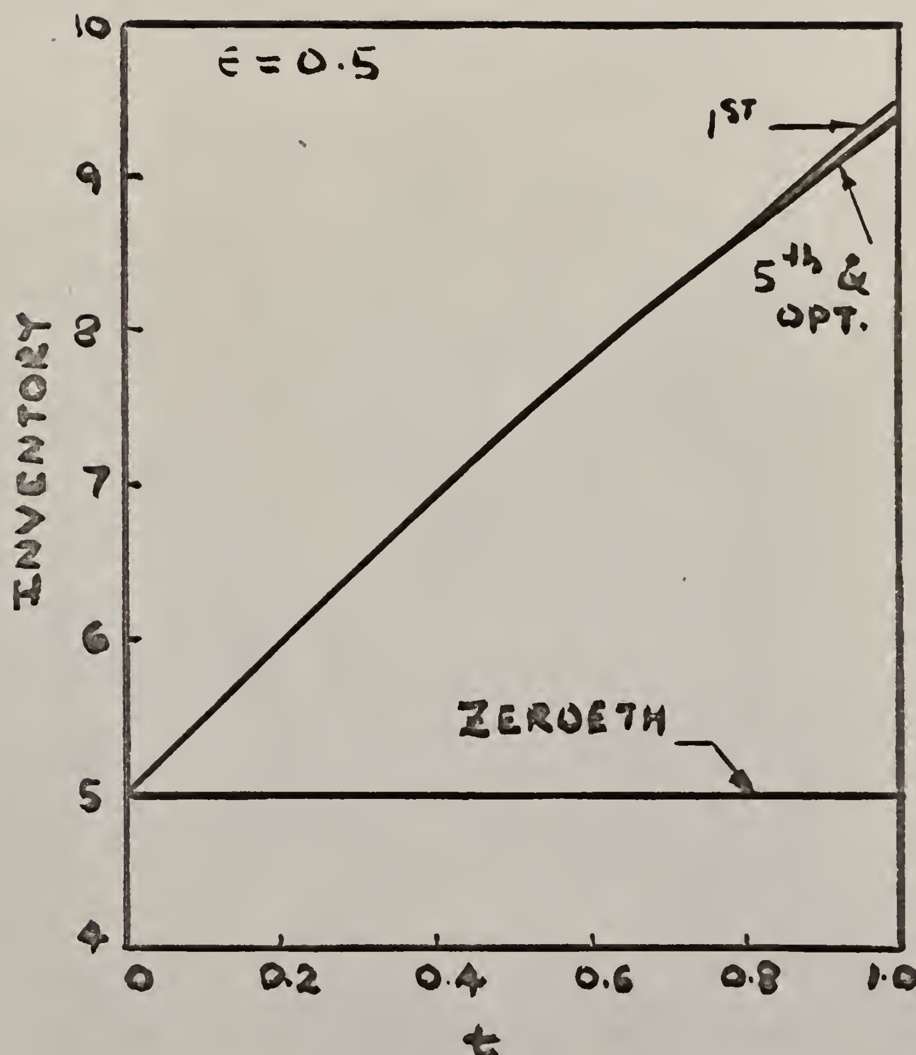

Fig. 2. Convergence rate of Inventory.

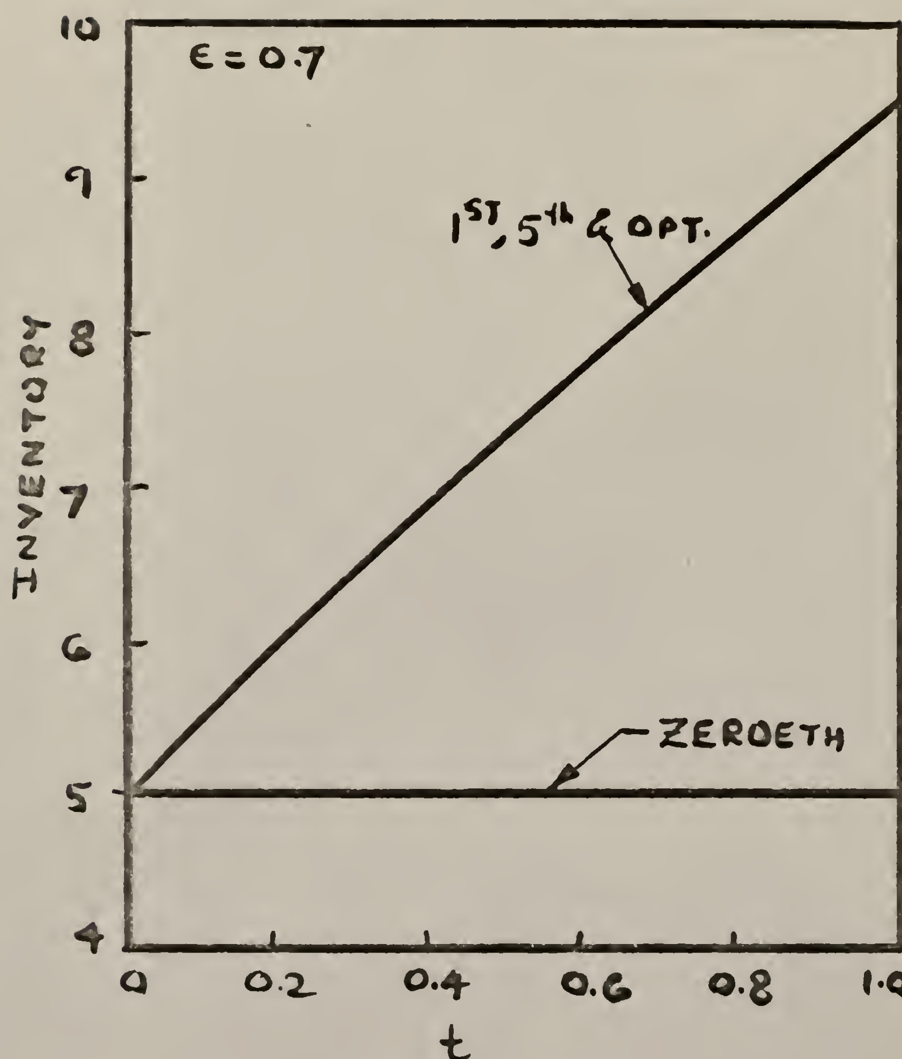

Fig. 3 Convergence rate of Inventory

Table 1A

Starting Trajectories

t	θ(t)	$x_1(t)$
0.00	7.189250000	5.000000000
0.01	7.187477000	5.051840000
0.02	7.079871000	5.103566000
0.03	7.077939000	5.154113000
0.04	7.104925000	5.204544000
0.05	7.132057000	5.255143000
0.06	7.149645000	5.305912000
0.07	7.157608000	5.356759000
0.08	7.158973000	5.407584000
0.09	7.156805000	5.458322000
0.10	7.153052000	5.508939000
0.11	7.148717000	5.559419000
0.12	7.144282000	5.609757000
0.13	7.140039000	5.659949000
0.14	7.135867000	5.709999000
0.15	7.131478000	5.759905000
0.16	7.127013000	5.809671000
0.17	7.122532000	5.859291000
0.18	7.118021000	5.908764000
0.19	7.113436000	5.958094000
0.20	7.108769000	6.007279000
0.21	7.104050000	6.056316000
0.22	7.099350000	6.105205000
0.23	7.094633000	6.153948000
0.24	7.089814000	6.202545000
0.25	7.084888000	6.250991000
0.26	7.079880000	6.299290000
0.27	7.074810000	6.347438000
0.28	7.069685000	6.395437000
0.29	7.064513000	6.443284000
0.30	7.059288000	6.490978000
0.31	7.054013000	6.538519000
0.32	7.048672000	6.585909000
0.33	7.043266000	6.633147000
0.34	7.037783000	6.680229000
0.35	7.032225000	6.727155000
0.36	7.026593000	6.773927000
0.37	7.020878000	6.820544000
0.38	7.015089000	6.867001000
0.39	7.009217000	6.913302000
0.40	7.003265000	6.959444000
0.41	6.997230000	7.005426000
0.42	6.991116000	7.051247000
0.43	6.984918000	7.096908000

Table 1A (continued)

0.44	6.978632000	7.142408000
0.45	6.972250000	7.187744000
0.46	6.965771000	7.232914000
0.47	6.959211000	7.277921000
0.48	6.952615000	7.322764000
0.49	6.946091000	7.367440000
0.50	6.939800000	7.411951000
0.51	6.933926000	7.456297000
0.52	6.928563000	7.500487000
0.53	6.923484000	7.544522000
0.54	6.917935000	7.588405000
0.55	6.910558000	7.632135000
0.56	6.899575000	7.675689000
0.57	6.883111000	7.719036000
0.58	6.859521000	7.762116000
0.59	6.827771000	7.804861000
0.60	6.787430000	7.847187000
0.61	6.738802000	7.889011000
0.62	6.682859000	7.930249000
0.63	6.620663000	7.970827000
0.64	6.553533000	8.010683000
0.65	6.482723000	8.049766000
0.66	6.409090000	8.088044000
0.67	6.333541000	8.125484000
0.68	6.256731000	8.162069000
0.69	6.179157000	8.197786000
0.70	6.101206000	8.232625000
0.71	6.023172000	8.266588000
0.72	5.945170000	8.299669000
0.73	5.867554000	8.331870000
0.74	5.790261000	8.363195000
0.75	5.713470000	8.393647000
0.76	5.637252000	8.423231000
0.77	5.561565000	8.451951000
0.78	5.486539000	8.479818000
0.79	5.412208000	8.506833000
0.80	5.338596000	8.533003000
0.81	5.265617000	8.558340000
0.82	5.193286000	8.582844000
0.83	5.121719000	8.606528000
0.84	5.050916000	8.629393000
0.85	4.980895000	8.651453000
0.86	4.911464000	8.672711000
0.87	4.842929000	8.693175000
0.88	4.775110000	8.712854000
0.89	4.708012000	8.731755000
0.90	4.641857000	8.749883000
0.91	4.576363000	8.767251000
0.92	4.511854000	8.783864000

Table 1A (continued)

0.93	4.448162000	8.799734000
0.94	4.385418000	8.814865000
0.95	4.323664000	8.829268000
0.96	4.262953000	8.842953000
0.97	4.203338000	8.855934000
0.98	4.144891000	8.868217000
0.99	4.087779000	8.879816000
1.00	4.087779000	8.890743000

Table 2

Effect of ε on the Rate of Convergence

of Cost Function x_2, $\theta(t) = 7$, $x_1(t) = 5$, $0 \leq t \leq t_f$.

Iteration	ε = 0.1	ε = 0.3	ε = 0.5	ε = 0.7	ε = 1.0
1	0.95957	0.95957	0.95956	0.95956	0.95957
5	0.95232	0.94536	0.94392	0.94356	0.94342
10	0.94694	0.94360	0.94337	0.94335	0.94335
15	0.94498	0.94339	0.94335	"	"
20	0.94415	0.94336	"	"	"
25	0.94376	0.94335	"	"	"
30	0.94356	"	"	"	"
35	0.94347	"	"	"	"
40	0.94342	"	"	"	"
45	0.94339	"	"	"	"
50	0.94339	"	"	"	"
55	0.94336	"	"	"	"

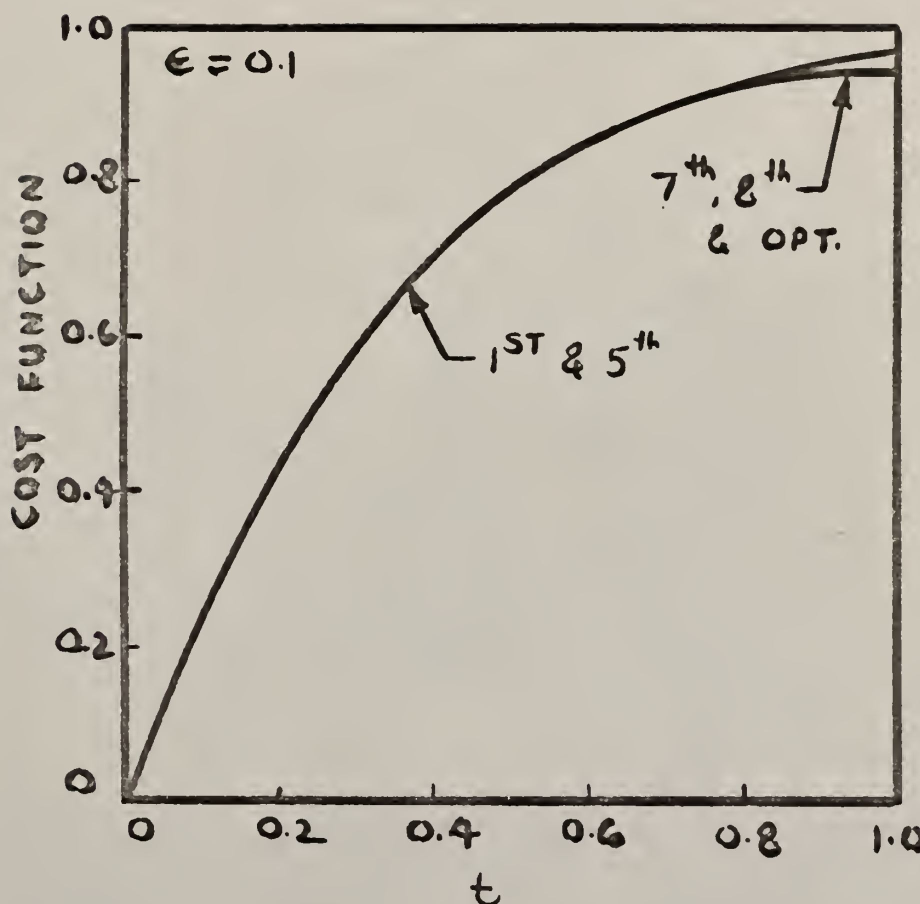

Fig. 4 Convergence rate of Cost Function.

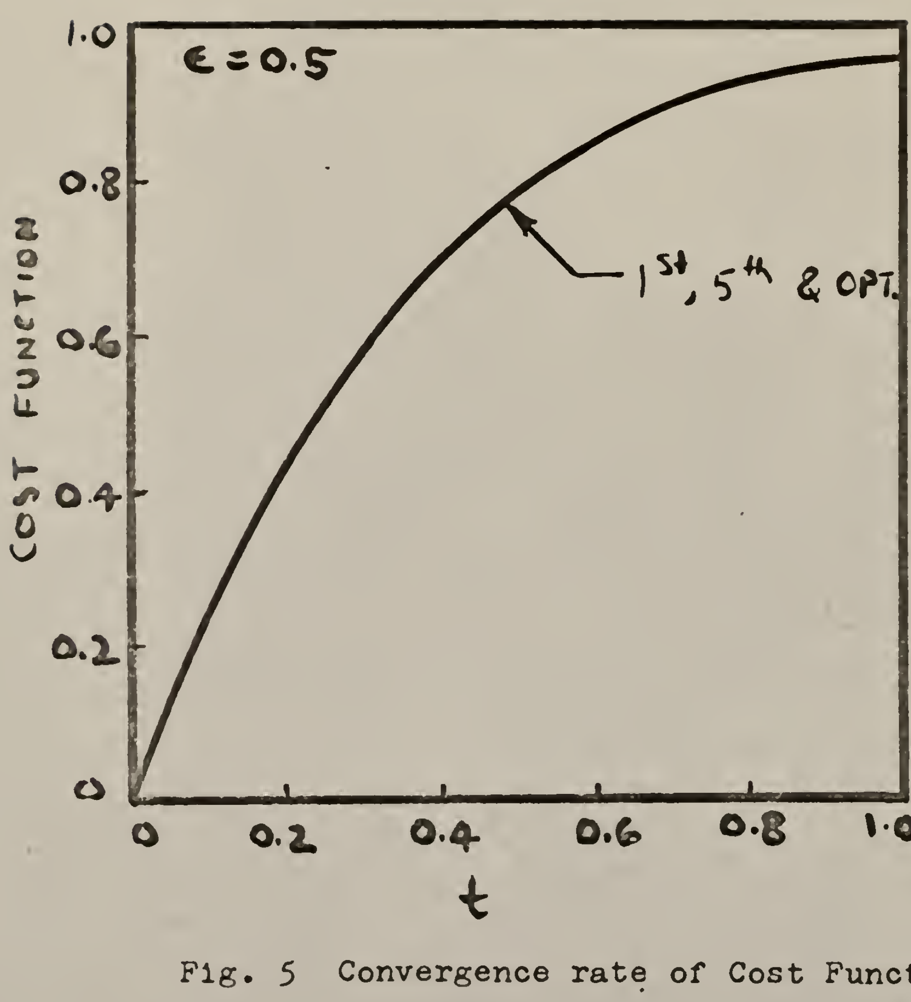

Fig. 5 Convergence rate of Cost Function.

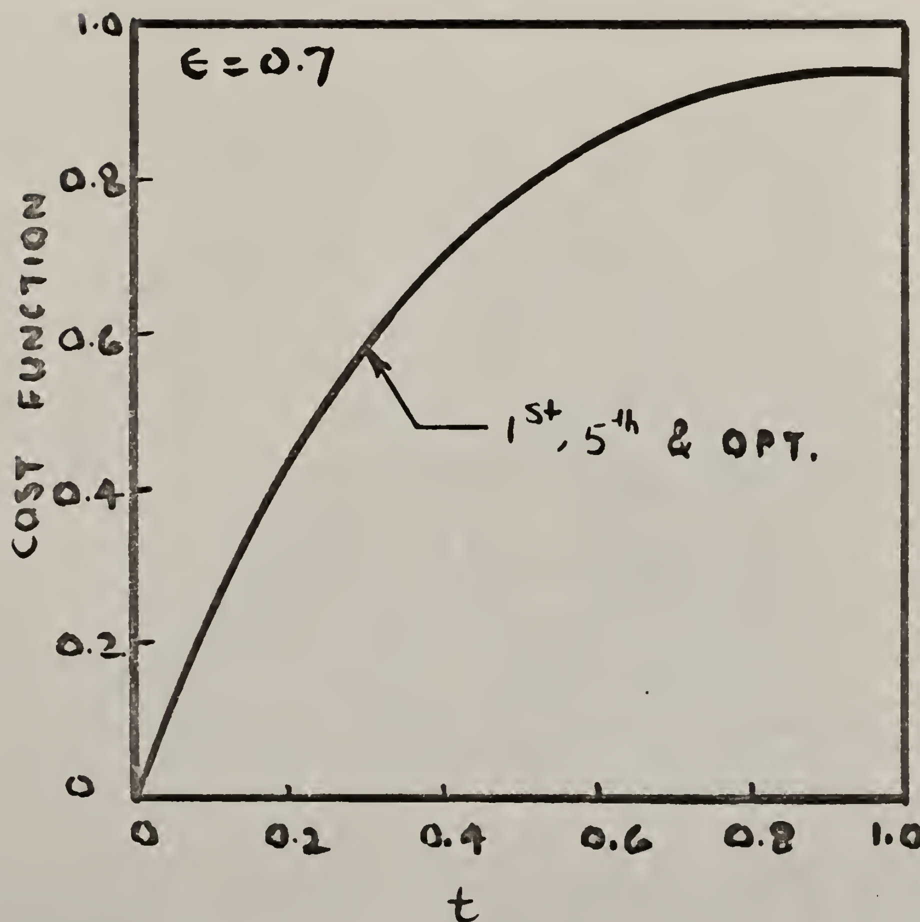

Fig. 6 Convergence rate of Cost Function.

Table 3

Effect of ε on Rate of Convergence of the

Production Rate θ, $\theta(t) = 7$, $x_1(t) = 5$, $0 \leq t \leq t_f$.

Iteration		$\varepsilon = 0.1$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$	$\varepsilon = 1.0$
1	$\theta(t_0)$	7.03101	7.09304	7.15507	7.21709	7.31013
	$\theta(t_f)$	6.97778	6.93333	6.88889	6.84444	6.77778
5	$\theta(t_0)$	7.10105	7.17094	7.18717	7.18952	7.18934
	$\theta(t_f)$	6.88690	6.64770	6.38816	6.10440	5.62812
10	$\theta(t_0)$	7.14225	7.18636	7.18927	7.18933	7.18933
	$\theta(t_f)$	6.76850	6.23612	5.58561	5.06248	5.0000
15	$\theta(t_0)$	7.16293	7.18884	7.18933	7.18933	7.18933
	$\theta(t_f)$	6.64410	5.74570	5.03792	5.00015	5.00000
20	$\theta(t_0)$	7.17416	7.18925	7.18933	7.18933	7.18933
	$\theta(t_f)$	6.51294	5.25530	5.00119	5.00000	5.00000
25	$\theta(t_0)$	7.18051	7.18932	7.18933	7.18933	7.18933
	$\theta(t_f)$	6.37410	5.04754	5.00004	5.00000	5.00000
30	$\theta(t_0)$	7.18416	7.18933	7.18933	7.18933	7.18933
	$\theta(t_f)$	6.22667	5.00802	5.00000	5.00000	5.00000
35	$\theta(t_0)$	7.18629	7.18933	7.18932	7.18933	7.18933
	$\theta(t_f)$	6.06984	5.00135	5.00000	5.00000	5.00000
40	$\theta(t_0)$	7.18754	7.18933	7.18933	7.18933	7.18933
	$\theta(t_f)$	5.90348	5.00023	5.00000	5.00000	5.00000
45	$\theta(t_0)$	7.18828	7.18932	7.18933	7.18933	7.18933
	$\theta(t_f)$	5.72933	5.00003	5.00000	5.00000	5.00000
50	$\theta(t_0)$	7.18871	7.18932	7.18933	7.18933	7.18933
	$\theta(t_f)$	5.55353	5.00000	5.00000	5.00000	5.00000
55	$\theta(t_0)$	7.18896	7.18932	7.18933	7.18933	7.18933
	$\theta(t_f)$	5.38931	5.00000	5.00000	5.00000	5.00000

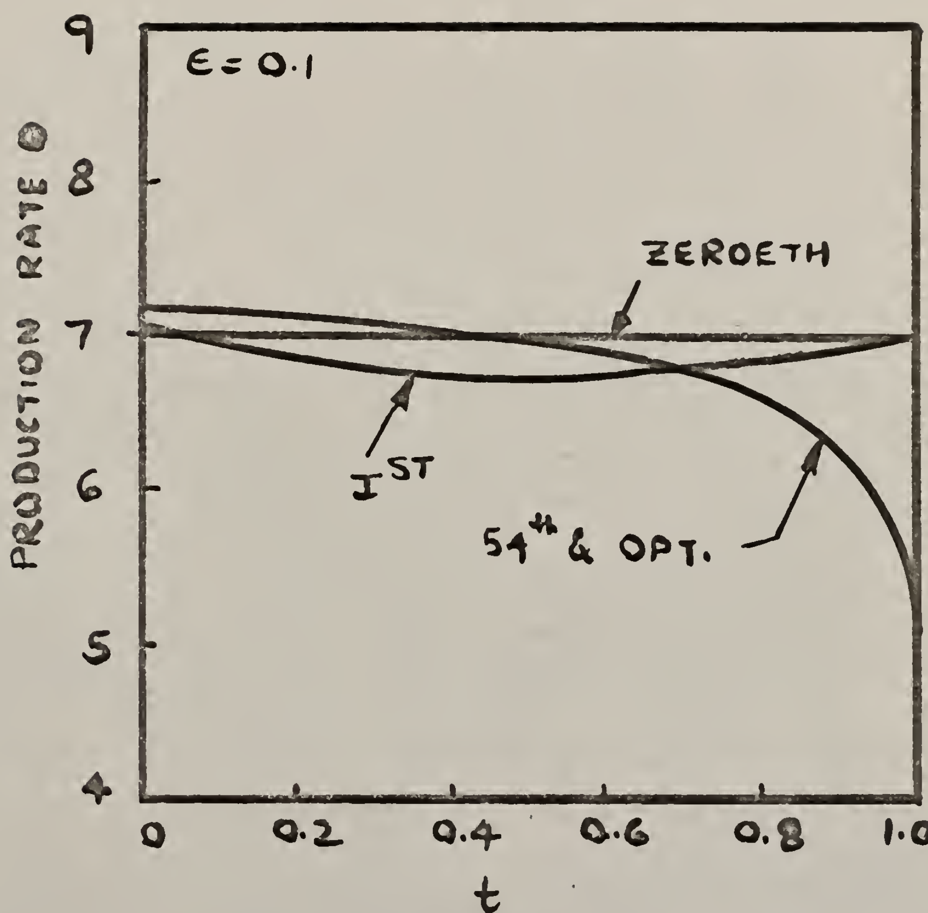

Fig. 7 Convergence rate of Production Rate.

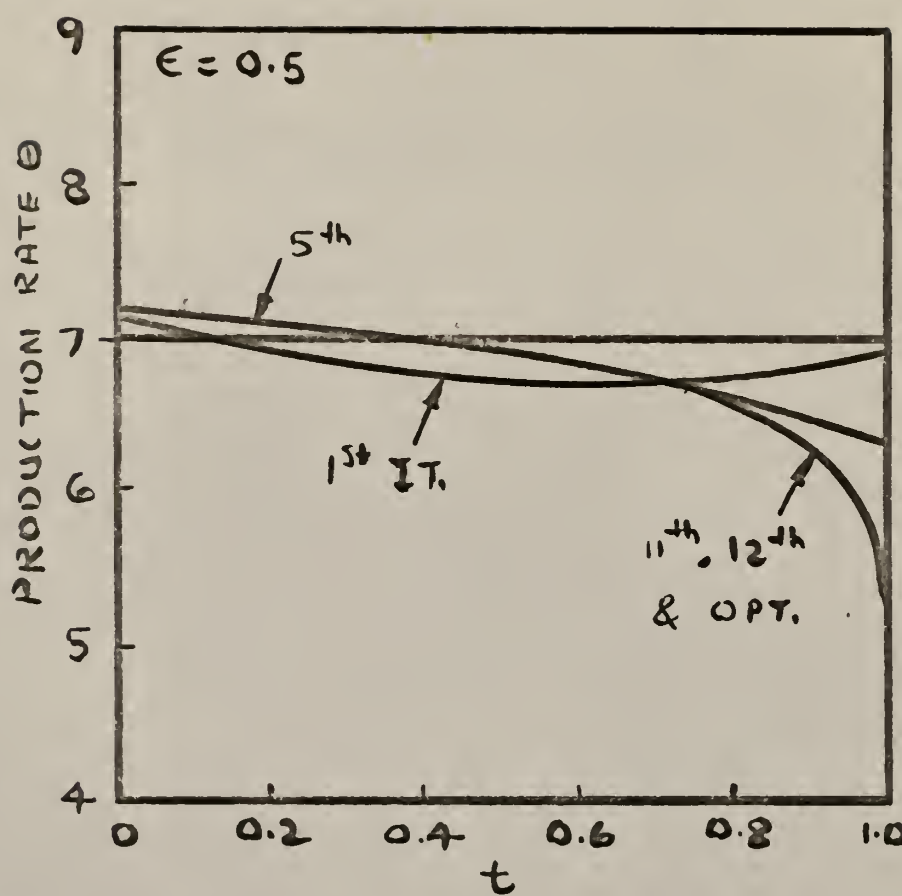

Fig. 8 Convergence rate of Production Rate.

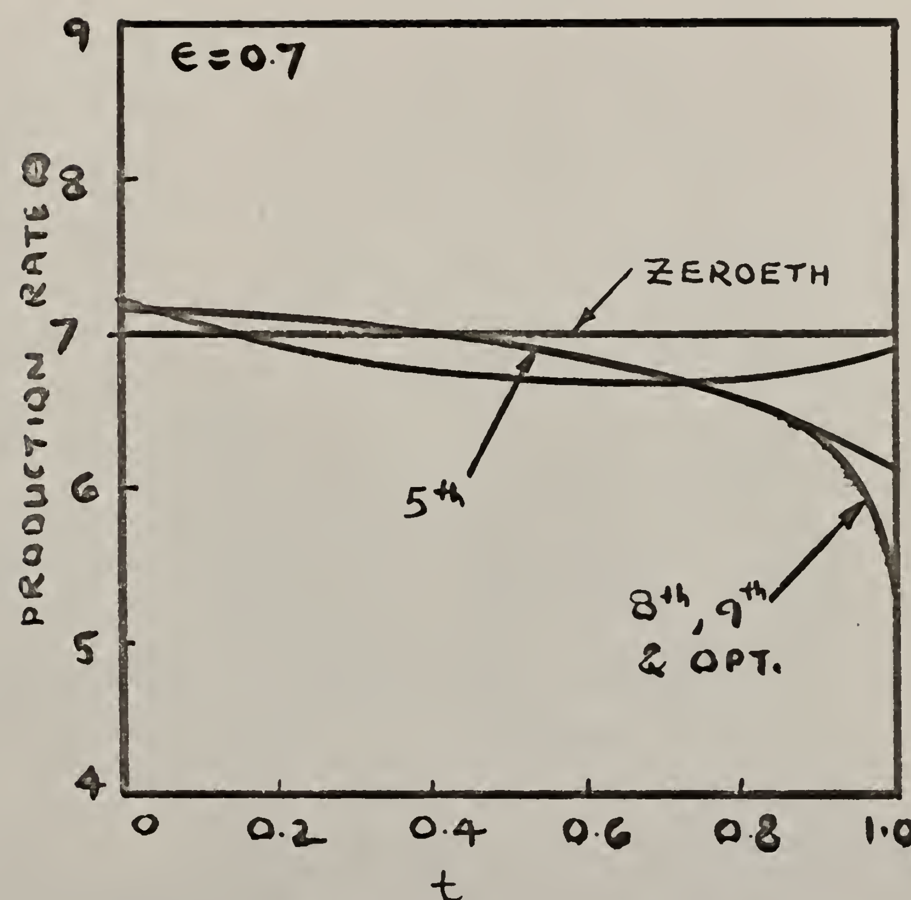

Fig. 9 Convergence rate of Production rate.

Numerical Results

This problem was solved by two approaches. In the first, the second variation method was used in combination with the first variation method. The same problem is solved by Lee and Shaikh (20). The values of the state and control variables (at all grid points) were taken from the results of the first variation. In particular, the values of X1(t) and θ(t) were taken from the 21st iteration of the first variation and fed as good starting values for the second variation. These values are listed in Table 1A. In the second approach, the second variation was tried directly by itself. For this, a guess was made for the starting values of the state variable and the control variables.

An interesting parameter in the computation is the step size ε which determines the magnitude of the step taken in each iteration. In the solution of this problem, a series of values of ε were selected and the computation was carried out for each. Tables 1, 2 and 3 show the convergence rate of inventory x1, cost function x2 and the production rate θ(t), respectively. These tables are for a constant starting value of the control variable, namely $\theta(t) = 7$, $0 \leq t \leq t_f$ and a constant starting value of the state variable, namely $x1(t) = 5$, $0 \leq t \leq t_f$. It is seen that for $\varepsilon = 1$, the fastest convergence rate is obtained while the convergence rate slows down when its value is decreased. Figures 1 through 9 show the rate of convergence of the inventory, production rate and the cost function for different values of ε .

Regarding the starting trajectory of the control variable, it was found that only the constant trajectories between $\theta(t) = 7$ and $\theta(t) = 8$ would lead to convergence. For all other control variable values, the

the problem would not converge. These other values were:

$$\theta(t) = 1,2,3,4,5,6 \quad \text{and } \theta(t) = 9,10,11, \quad 0 \le t \le t_f$$

Also, the combination of the first and the second variation required about 50 iterations to reach the optimal with $\varepsilon = 0.3$. A higher value of ε could not be used as it led to overstepping in this situation.

3.2 An Inventory and Advertising Model

The Model

This model is an extension of the one formulated by Teichroew (27). Consider a marketing situation where only a certain number of possible customers possess certain information about a firm's product. Suppose that the total number of such possible customers remains constant and that the diffusion of information occurs only through personal contact. The number of contacts made by an informed person in a unit time is known as contact coefficient. In a contact, the contactee receives information if he does not already have it; if he already has it, the contact is wasted insofar as increasing the number of informed persons is concerned.

Let $K(0) = K_0$ = number of informed persons at time t_0

N = total number of persons

c = contact coefficient, the number of contacts made by one informed person per unit time

$K(t)$= number of informed persons at time t.

Then $K(t)/N$ = proportion of informed persons at time t

$1 - K(t)/N$ = proportion of uninformed persons at time t

$c.K(t).dt$ = contacts made during a time interval dt.

Clearly $dK(t) = c.K(t).dt.(1-K(t)/N)$

Thus the equation governing the process is

$$\frac{dK(t)}{dt} = c.K(t).(1-K(t)/N) \tag{65}$$

Suppose next that the firm can influence the number of contacts by spending money on advertising. In particular it can increase the number of contacts made by the informed persons (above the ones included in c) by an additional number A per unit time.

Equation (65) now becomes

$$\frac{dK(t)}{dt} = K(t).(c+A(t)).(1-K(t)/N) \tag{66}$$

If each successful contact results in the sale of n units of the firm's product and if Q(t) represents the sale at time t, then

$$Q(t) = n\,K(t)$$

Letting n=1 and substituting Q(t) for K(t) in Equation (66), then

$$\frac{dQ(t)}{dt} = Q(t).(c+A(t)).(1-Q(t)/N) \tag{67}$$

The rate of change of the firm's inventory is given by

$$\frac{dX(t)}{dt} = P(t) - Q(t) \tag{68}$$

where P(t) = production rate at time t.

The production rate is assumed to be a linear function of time

$$P(t) = a+bt \tag{69}$$

where a and b are constants.

This assumption is made to simplify the model by avoiding a second control variable.

The firm's management wishes to maximize the profit

$$S_T = \int_0^T [F\cdot Q(t) - C_I\ (P_I - x(t))^2 - C_A\ A^2(t)\ Q(t)]dt \tag{70}$$

where S_T is the total net profit.

F is the revenue from the sale of one unit of the product. C_I is the inventory carrying cost and has the same significance as in the model described in Section 3.1. P_I can be considered as the capacity for the storage of inventory. C_A is the cost of advertising.

Equations (67) through (70) represent the system under consideration. The system has two state variables, inventory X(t) and sales Q(t), and there is one control variable, advertising A(t).

The initial conditions and the numerical values used are:

$a = 0.7 \quad b = 1.0 \quad c = 2.0 \quad N = 1.5 \quad F = 10.0$

$C_I = 0.15 \quad P_I = 1.0 \quad C_A = 1.0 \quad X(0) = 0.2 \quad Q(0) = 0.2$

<u>Recursive Relation</u>

The necessary relations for the second variation can be obtained in the following manner. Note that in these derivations $\underline{x}(t)$ denotes the state variable vector while x(t) denotes the inventory. From Equation 70, then,

$$J = Q \cdot F - C_I (P_I - x(t))^2 - C_A Q A^2(t).$$

The various derivatives required for obtaining the second variation equations are:

$$\frac{\partial J}{\partial \underline{x}} = \begin{bmatrix} \frac{\partial J}{\partial x} \\ \\ \frac{\partial J}{\partial Q} \end{bmatrix} = \begin{bmatrix} 2C_I (P_I - x(t)) \\ \\ F - C_A A^2(t) \end{bmatrix}$$

$$\frac{\partial^2 J}{\partial \underline{x}^2} = \begin{pmatrix} \frac{\partial}{\partial x}\left(\frac{\partial J}{\partial x}\right) & \frac{\partial}{\partial x}\left(\frac{\partial J}{\partial Q}\right) \\ \frac{\partial}{\partial Q}\left(\frac{\partial J}{\partial x}\right) & \frac{\partial}{\partial Q}\left(\frac{\partial J}{\partial Q}\right) \end{pmatrix} = \begin{pmatrix} -2C_I & 0 \\ 0 & 0 \end{pmatrix}.$$

$$\frac{\partial J}{\partial \underline{\theta}} = \frac{\partial J}{\partial A(t)} = -2C_A\ Q(t)\ A(t)$$

$$\frac{\partial^2 J}{\partial \underline{\theta}^2} = \frac{\partial^2 J}{\partial A^2(t)} = -2\ C_A\ Q(t)$$

$$\frac{\partial^2 J}{\partial \underline{\theta}\ \partial x} = \frac{\partial}{\partial A(t)} \begin{pmatrix} \frac{\partial J}{\partial x} \\ \frac{\partial J}{\partial Q} \end{pmatrix} = [0 \qquad -2C_A\ A(t)]$$

$$\frac{\partial f_1}{\partial \underline{x}} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} \qquad \frac{\partial^2 f_1}{\partial \underline{x}\ \partial \underline{\theta}} = [0\ ,\ 0]$$

$$\frac{\partial f_2}{\partial \underline{x}} = \begin{pmatrix} 0 \\ \left(C+A(t)\right)\left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix} \qquad \frac{\partial^2 f_2}{\partial \underline{x}\ \partial \underline{\theta}} = \left(0,\ \ (1 - \frac{2Q(t)}{N})\right)$$

$$\frac{\partial f_1}{\partial \underline{\theta}} = 0 \qquad\qquad \frac{\partial^2 f_1}{\partial \underline{\theta}^2} = 0$$

$$\frac{\partial f_2}{\partial \underline{\theta}} = Q(t)\left(1 - \frac{Q(t)}{N}\right) \qquad \frac{\partial^2 f_2}{\partial \underline{\theta}^2} = 0$$

$$\frac{\partial f'}{\partial \underline{x}} = \begin{pmatrix} \frac{\partial}{\partial x}[f_1] & \frac{\partial}{\partial x}[f_2] \\ \frac{\partial}{\partial Q}[f_1] & \frac{\partial}{\partial Q}[f_2] \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ -1 & [C+A(t)]\left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix}$$

$$\frac{\partial^2 f_1}{\partial \underline{x}^2} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\frac{\partial^2 f_2}{\partial \underline{x}^2} = \begin{pmatrix} 0 & 0 \\ 0 & -\frac{2}{N}(C + A(t)) \end{pmatrix}$$

$$\frac{\partial f'}{\partial \underline{\theta}} = \frac{\partial}{\partial A(t)} \quad [f_1 \quad f_2] = \left(0 \quad , \quad Q(t)\left(1 - \frac{Q(t)}{N}\right)\right)$$

Expressions for the terms $\underline{R}$, $\underline{s}$, $\underline{T}$ from Equation 30 result in

$$\underline{R} = [0 \quad -2C_A\, A(t)] + \left[0 \quad Q(t)\left(1 - \frac{Q(t)}{N}\right)\right] \begin{bmatrix} P_{11} & P_{12} \\ P_{12} & P_{22} \end{bmatrix}$$

$$+ \left[0 \quad z_2\left(1 - \frac{2Q(t)}{N}\right)\right] .$$

$$= [0 \quad -2C_A\, A(t)] + \left[P_{12}\, Q(t)\left(1 - \frac{Q(t)}{N}\right), \quad P_{22}\, Q(t)\left(1 - \frac{Q(t)}{N}\right)\right]$$

$$+ \left[0 \quad z_2\left(1 - \frac{2Q(t)}{N}\right)\right] .$$

Let $\underline{R} = [R_1, \quad R_2]$

where

$$R_1 = P_{12} \quad Q(t) \left(1 - \frac{Q(t)}{N}\right)$$

$$R_2 = -2\, C_A\, A(t) + P_{22}\, Q(t) \left(1 - \frac{Q(t)}{N}\right)$$

$$+ z_2 \left(1 - \frac{2Q(t)}{N}\right) .$$

Equation 46 gives

$\underline{S} = -2C_A\, Q(t)\, A(t) + z_2\, Q(t) \left(1 - \frac{Q(t)}{N}\right)$ and Equation 47 gives,

$\underline{T} = -2C_A\ Q(t).$

It is now possible to determine the $2n + \frac{n(n+1)}{2}$ i.e. (2+2+3) or seven equations to be integrated backwards. Equation (19) becomes

$$\frac{d\underline{z}}{dt} = \begin{pmatrix} -2C_I\ (P_I - x(t)) \\ \\ -F + C_A\ A^2(t) \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ \\ -1 & [C+A(t)]\left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix} \begin{pmatrix} z_1 \\ \\ z_2 \end{pmatrix}$$

$$= \begin{pmatrix} -2C_I\ (P_I - x(t) \\ \\ -F + C_A\ A^2(t) \end{pmatrix} - \begin{pmatrix} 0 \\ \\ -z_1 + z_2\ [C + A(t)]\left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix}$$

$$\therefore \quad \frac{dz_1}{dt} = -2C_I\ [P_I - x(t)] \tag{71}$$

$$\frac{dz_2}{dt} = -F + C_A\ A^2(t) + z_1 - z_2\ [C + A(t)]\left(1 - \frac{2Q(t)}{N}\right) \tag{72}$$

Thus Equations (71) and (72) correspond to Equation (19). Equation (28) in this case becomes

$$\frac{d\underline{P}}{dt} = \begin{bmatrix} 2C_I & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{2z_2}{N}\,[C + A(t)] \end{bmatrix}$$

$$- \begin{bmatrix} 0 & -P_{11} + P_{12}\,[C + A(t)] \cdot \left(1 - \frac{2Q(t)}{N}\right) \\ -P_{11} + P_{12}\,[C + A(t)]\left(1 - \frac{2Q(t)}{N}\right), & -2P_{12} + 2P_{22}\,[C + A(t)] \cdot \left(1 - \frac{2Q(t)}{N}\right) \end{bmatrix} .$$

Hence Equation (28) is represented by the following three equations:

$$\frac{dP_{11}}{dt} = 2C_I + R_1^2 T \qquad (73)$$

$$\frac{dP_{12}}{dt} = P_{11} - P_{12}\,[C + A(t)]\left(1 - \frac{2Q(t)}{N}\right) + R_1 R_2 T \qquad (74)$$

$$\frac{dP_{22}}{dt} = \frac{2z_2}{N}\,[C + A(t)] + 2P_{12} - 2P_{22}\,[C + A(t)\ \left(1 - \frac{2Q(t)}{N}\right) + R_2^2 T \qquad (75)$$

To avoid confusion, the $\underline{q}$ in the derivation of the method given in Equation (48) is denoted by QF here. Thus Q still represents the sales for this problem.

Equation (48) is given by

$$\frac{dQ_F}{dt} = \begin{pmatrix} R_1 \\ \\ R_2 \end{pmatrix} \left(\frac{1}{T}\right) S + \begin{pmatrix} R_1 \\ \\ R_2 \end{pmatrix} \left(\frac{1}{T}\right) \begin{pmatrix} 0 & , & Q(t)\left(1 - \frac{Q(t)}{N}\right) \end{pmatrix} \begin{pmatrix} QF_1 \\ \\ QF_2 \end{pmatrix}$$

$$- \begin{pmatrix} 0 & -1 \\ \\ 0 & [C + A(t)]\left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix} \begin{pmatrix} QF_1 \\ \\ QF_2 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{R_1 s}{T} \\ \\ \frac{R_2 s}{T} \end{pmatrix} + \begin{pmatrix} \frac{R_1}{T} \cdot QF_2 \cdot Q(t) \cdot \left(1 - \frac{Q(t)}{N}\right) \\ \\ \frac{R_2}{T} \cdot QF_2 \cdot Q(t) \cdot \left(1 - \frac{Q(t)}{N}\right) \end{pmatrix}$$

$$\begin{pmatrix} - QF_2 \\ \\ QF_2 \; [C + A(t)] \left(1 - \frac{2Q(t)}{N}\right) \end{pmatrix}$$

$$\frac{dQF_1}{dt} = \frac{R_1 s}{T} + \frac{R_1}{T} \cdot QF_2 \cdot Q(t) \cdot \left(1 - \frac{Q(t)}{N}\right) + QF_2 \qquad (76)$$

Table 4

Starting Trajectories

t	I(t)	Q(t)
0.00	0.200	0.200
0.01	0.203	0.200
0.02	0.205	0.201
0.03	0.208	0.208
0.04	0.210	0.212
0.05	0.211	0.220
0.06	0.212	0.228
0.07	0.217	0.230
0.08	0.220	0.238
0.09	0.222	0.245
0.10	0.226	0.250
0.11	0.228	0.261
0.12	0.230	0.272
0.13	0.233	0.283
0.14	0.238	0.300
0.15	0.242	0.317
0.16	0.250	0.339
0.17	0.252	0.410
0.18	0.260	0.430
0.19	0.264	0.450
0.20	0.270	0.460
0.21	0.274	0.483
0.22	0.280	0.503
0.23	0.284	0.520
0.24	0.290	0.540
0.25	0.293	0.560
0.26	0.300	0.580
0.27	0.301	0.600
0.28	0.306	0.620
0.29	0.310	0.648
0.30	0.318	0.665
0.31	0.320	0.690
0.32	0.324	0.702
0.33	0.330	0.724
0.34	0.336	0.745
0.35	0.340	0.760
0.36	0.342	0.785
0.37	0.346	0.800
0.38	0.350	0.814
0.39	0.351	0.830
0.40	0.355	0.842
0.41	0.359	0.857
0.42	0.360	0.870
0.43	0.362	0.880
0.44	0.368	0.890

Table 4 (continued)

0.45	0.370	0.900
0.46	0.371	0.910
0.47	0.372	0.920
0.48	0.375	0.923
0.49	0.378	0.930
0.50	0.380	0.936
0.51	0.381	0.940
0.52	0.382	0.945
0.53	0.387	0.950
0.54	0.389	0.951
0.55	0.390	0.958
0.56	0.391	0.962
0.57	0.392	0.970
0.58	0.393	0.971
0.59	0.398	0.978
0.60	0.400	0.982
0.61	0.400	0.990
0.62	0.401	0.995
0.63	0.403	1.000
0.64	0.408	1.002
0.65	0.410	1.010
0.66	0.411	1.015
0.67	0.412	1.020
0.68	0.416	1.028
0.69	0.417	1.030
0.70	0.419	1.037
0.71	0.420	1.040
0.72	0.421	1.045
0.73	0.422	1.050
0.74	0.426	1.051
0.75	0.430	1.056
0.76	0.435	1.060
0.77	0.438	1.065
0.78	0.440	1.070
0.79	0.441	1.071
0.80	0.447	1.078
0.81	0.450	1.081
0.82	0.451	1.090
0.83	0.453	1.092
0.84	0.460	1.100
0.85	0.461	1.101
0.86	0.462	1.110
0.87	0.468	1.112
0.88	0.470	1.120
0.89	0.473	1.130
0.90	0.475	1.135
0.91	0.480	1.140
0.92	0.482	1.143
0.93	0.488	1.150

Table 4 (continued)

0.94	0.490	1.160
0.95	0.496	1.165
0.96	0.500	1.170
0.97	0.503	1.173
0.98	0.510	1.180
0.99	0.513	1.190
1.00	0.519	1.200

$$\frac{dQF_2}{dt} = \frac{R_2 s}{T} + \frac{R_2}{T} \cdot QF_2 \cdot Q(t) \cdot \left(1 - \frac{Q(t)}{N}\right)$$

$$- QF_2 \, [\, C + A(t) \,] \cdot \left(1 - \frac{2Q(t)}{N}\right) \tag{77}$$

Thus Equations (76) and (77) represent Equation (48). The equation for improving the control variable becomes

$$A(t)^{(j+1)} = A(t)^{(j)} + \frac{\varepsilon}{T} \left[s + QF_2 \cdot Q(t) \cdot \left(1 - \frac{Q(t)}{N}\right) \right]$$

$$+ \frac{1}{T} \left\{ R_1 (x^{(j+1)} - x^{(j)}) + R_2 (Q^{(j+1)} - Q^{(j)}) \right\}$$

Equation (78) represents Equation (50) (78)

This problem illustrates how tedious the calculations become when the number of variables increases.

Numerical Results

In here, the starting trajectories of the two state variables, inventory I(t) and the sales Q(t), were fed from the results of the solution of the same problem by dynamic programming. These values are listed in Table 4. Actually these values are obtained after dividing the original results by 100. This was required to prevent the exponential overflow of the system of equations. The starting trajectory of the control variable was tried in the range of 0.001 to 6.0. It was found that all these values would work; however, the best value was found to be $\theta(t) = 0.5$, $0 \le t \le t_f$.

Table 5

Effect of ε on the Rate of Convergence

of $I(t_f)$ with $A_0(t) = 0.5$.

Iteration	$\varepsilon = 0.1$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$
1	0.8524	0.8524	0.8524	0.8524
5	0.7137	0.6274	0.6076	0.6277
10	0.6546	0.5990	0.5939	0.5929
14	0.6307	0.5948	0.5935	0.5934
16	0.6227	0.5941	"	0.5935
17	0.6194	0.5939	"	"
21	0.6096	0.5936	"	"

*The Values of $I_0(t)$ & $Q_0(t)$ are obtained from Table 4.

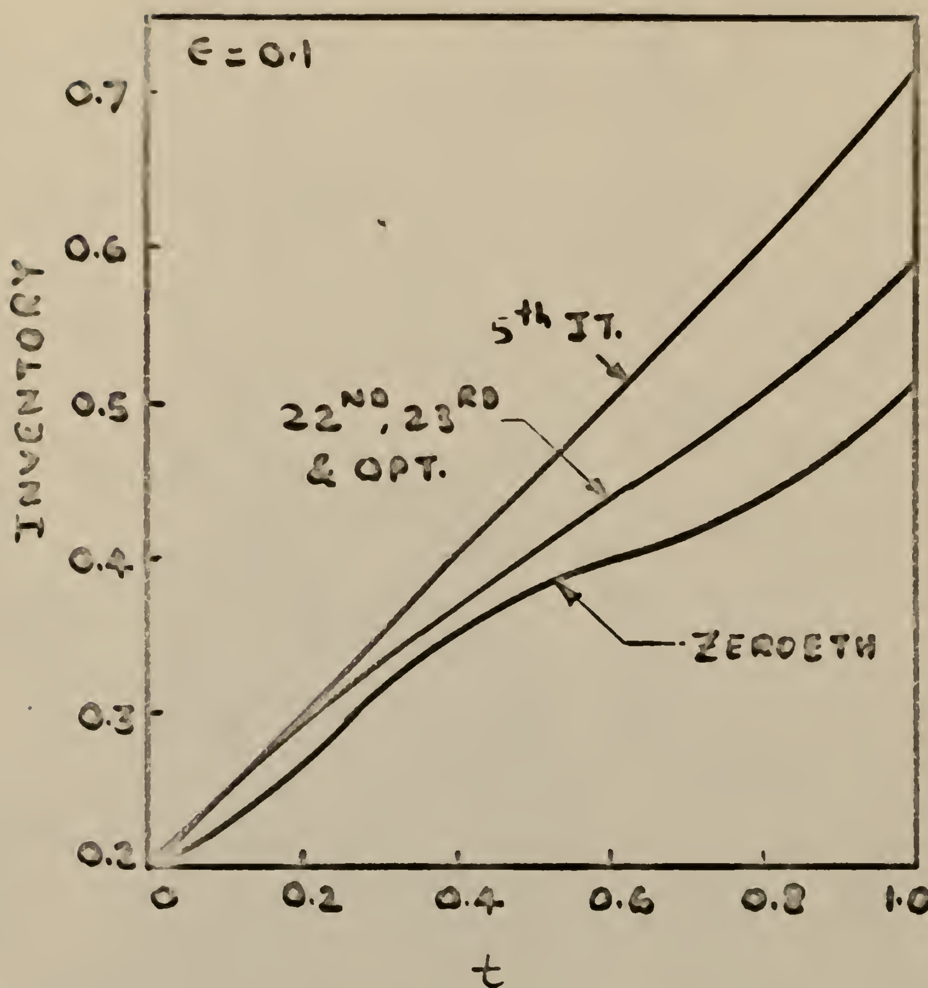

Fig. 10 Convergence rate of Inventory.

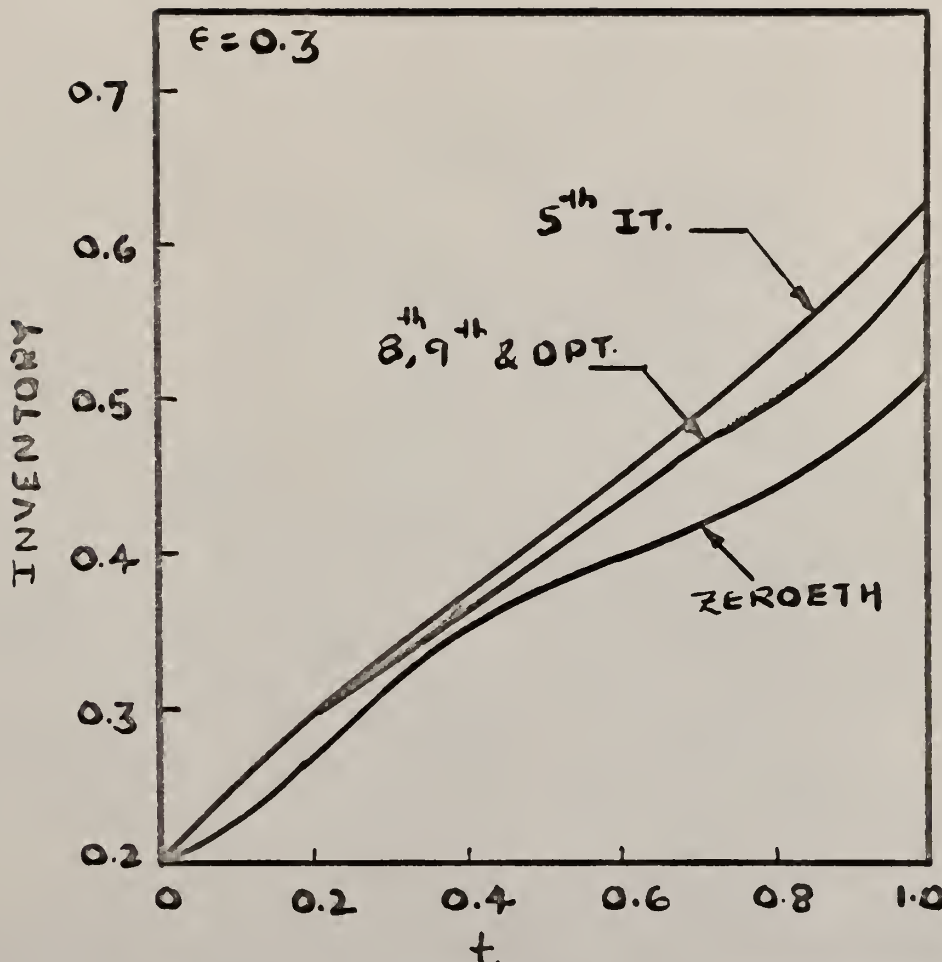

Fig. 11 Convergence rate of Inventory.

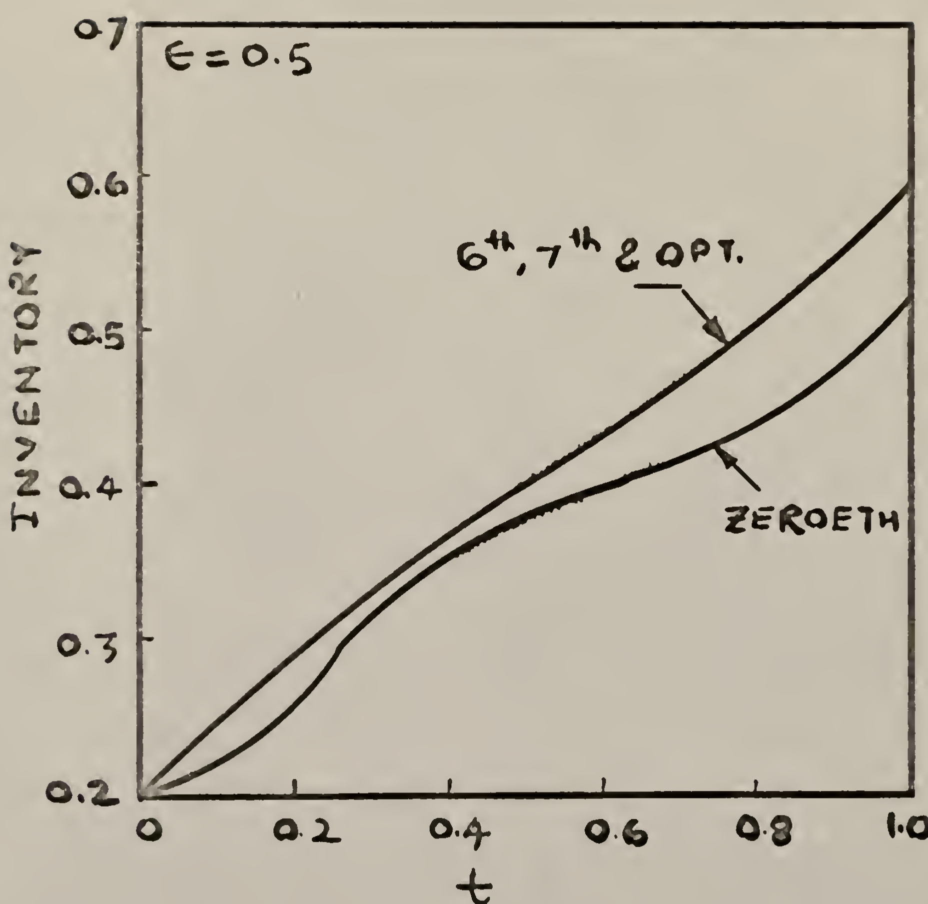

Fig. 12 Convergence rate of Inventory.

Table 6

Effect of ε on the Rate of Convergence

of $Q(t_f)$, with $A_0(t) = 0.5$.

Iteration	$\varepsilon = 0.1$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$
1	0.9781	0.9781	0.9781	0.9781
5	1.135	1.198	1.206	1.172
10	1.179	1.218	1.222	1.223
14	1.197	1.221	"	"
16	1.202	1.222	"	"
17	1.205	"	"	"
21	1.211	"	"	"

*The Values of $I_0(t)$ & $Q_0(t)$ are obtained from Table 4.

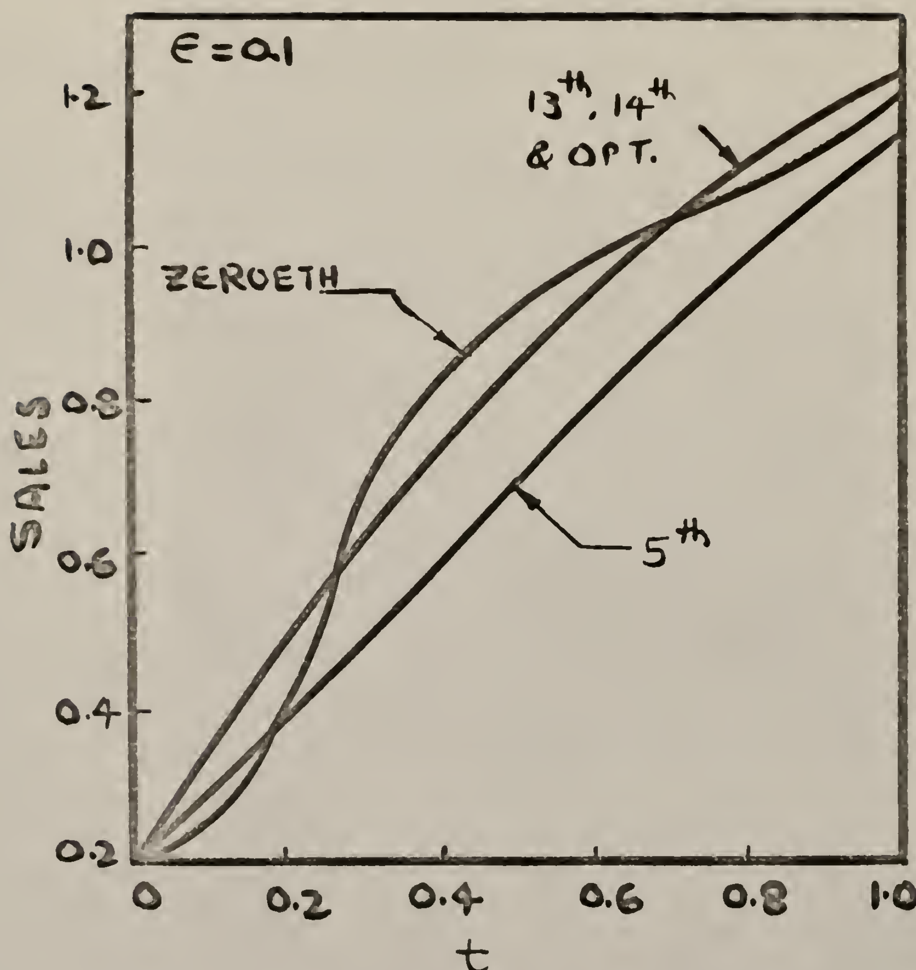

Fig. 13 Convergence rate of Sales.

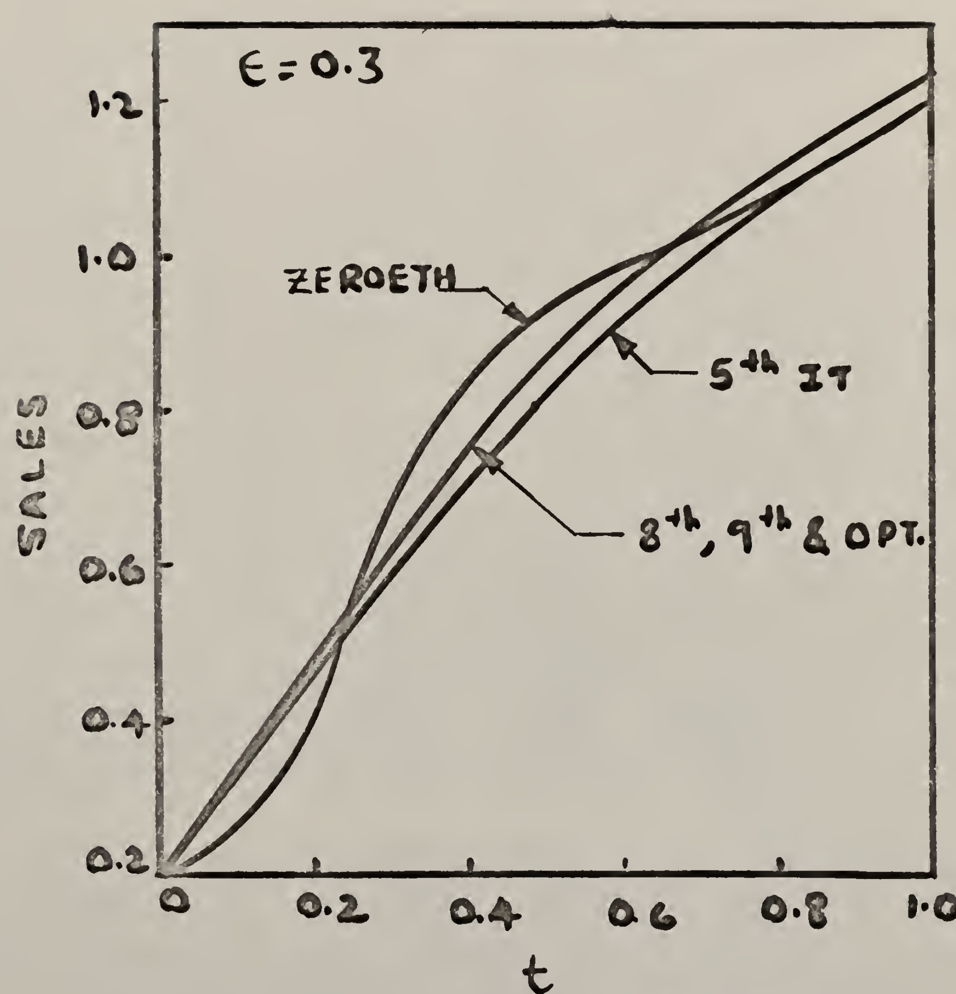

Fig. 14 Convergence rate of Sales.

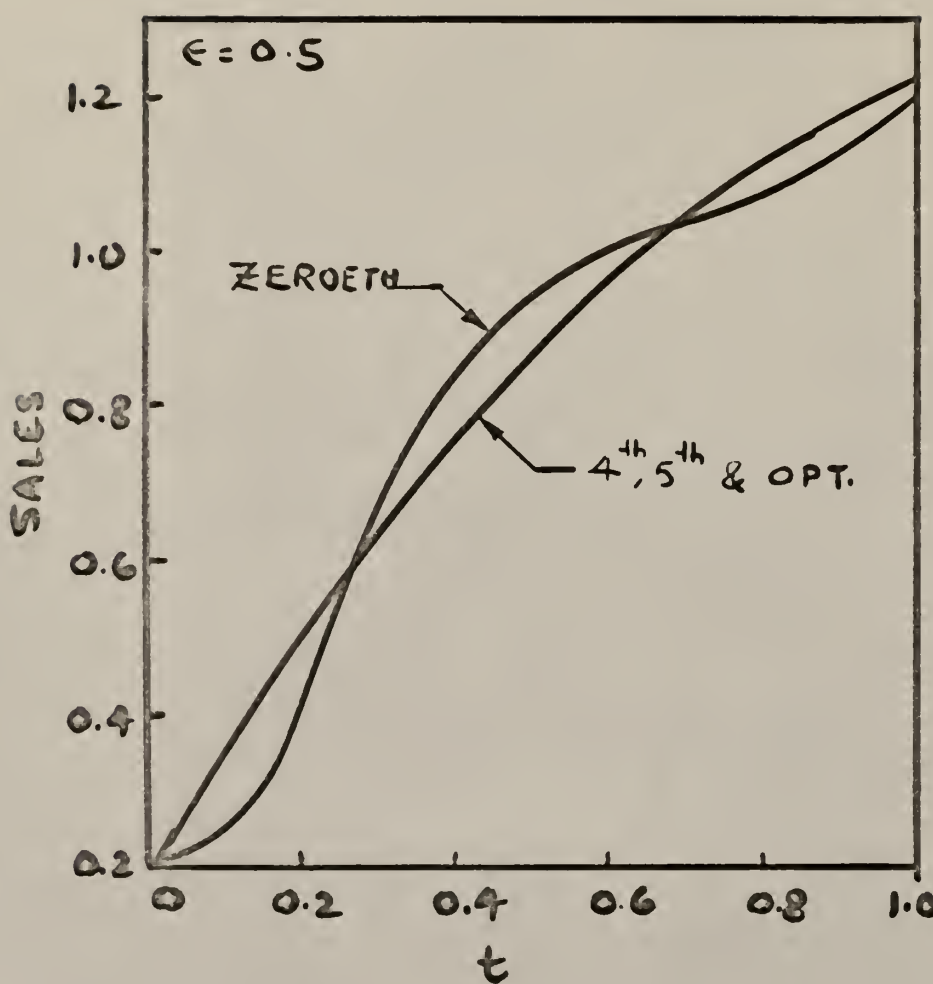

Fig. 15 Convergence rate of Sales.

Table 7

Effect of ε on the Rate of Convergence

of Total Profit, with $A_0(t) = 0.5$.

Iteration	$\varepsilon = 0.1$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$
1	5.298	5.298	5.298	5.298
5	6.260	6.596	6.621	6.571
10	6.527	6.626	6.626	6.626
14	6.589	"	"	"
16	6.604	"	"	"
17	6.609	"	"	"
21	6.620	"	"	"

*The Values of $I_0(t)$ & $Q_0(t)$ are obtained from Table 4.

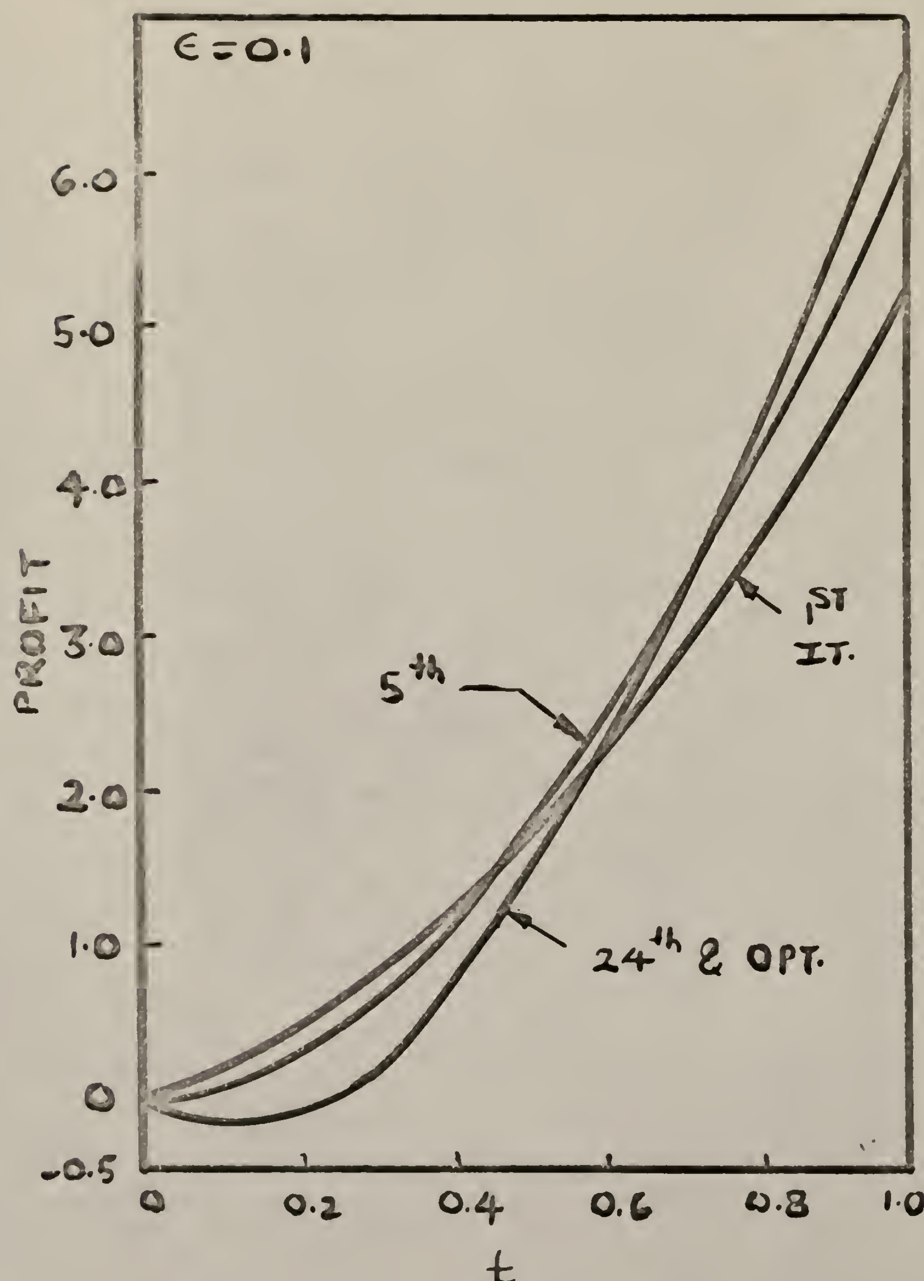

Fig. 16 Convergence rate of Profit

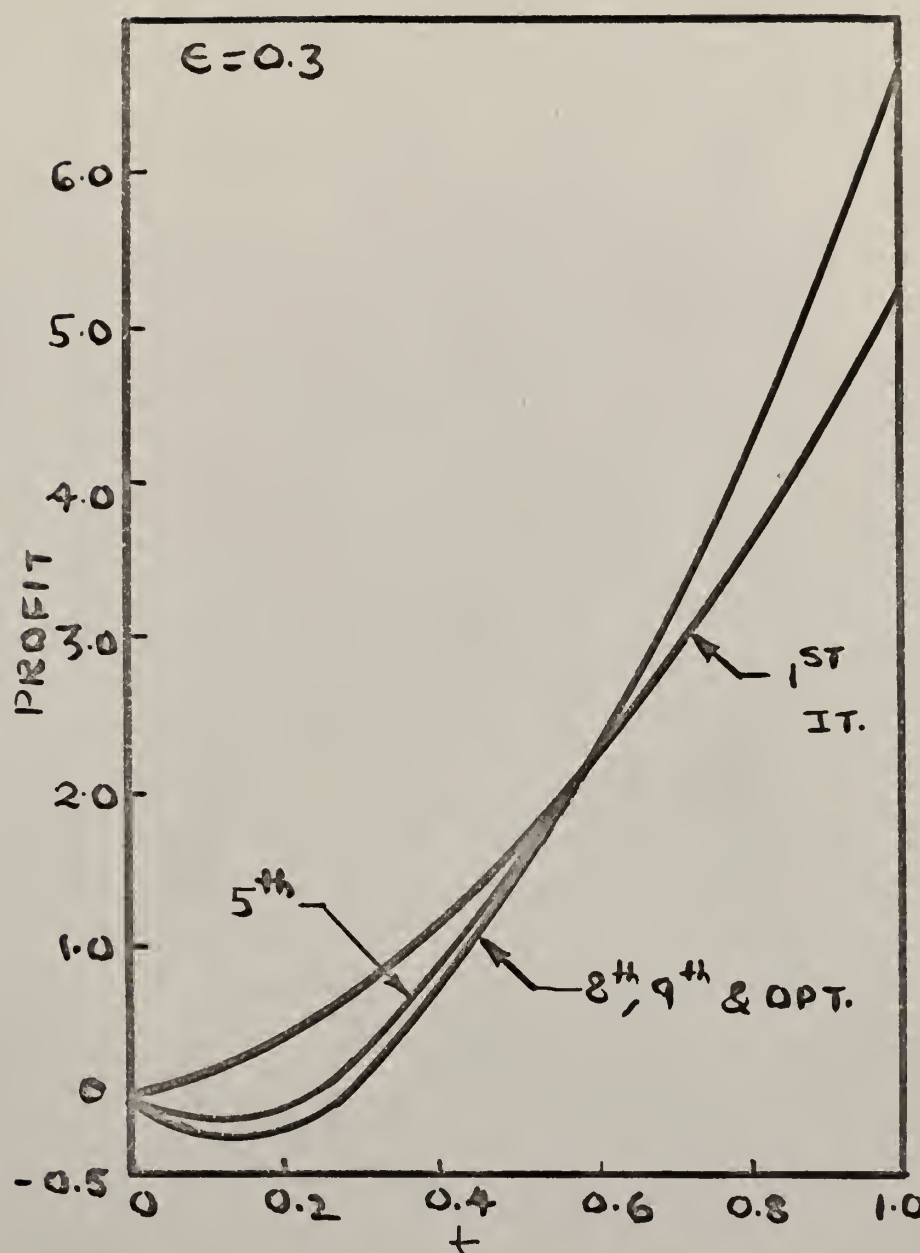

Fig. 17 Convergence rate of Profit.

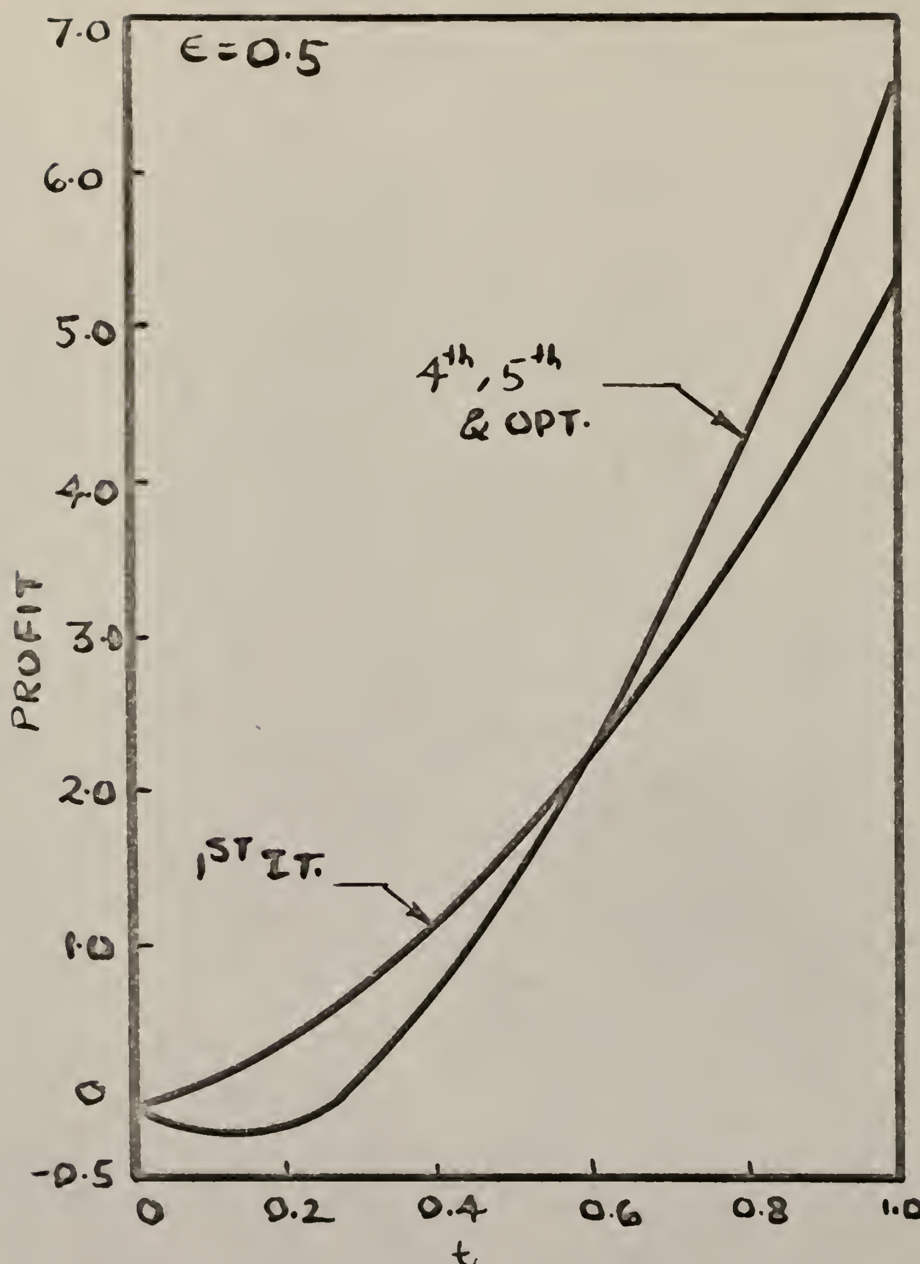

Fig. 18 Convergence rate of Profit.

Table 8

Effect of ε on the Rate of Convergence

of $A(t_0)$, with $A_0(t) = 0.5$.

Iteration	$\varepsilon = .01$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$
1	1.320	2.960	4.599	6.239
5	3.088	4.841	5.218	5.269
10	4.091	5.174	5.222	5.221
14	4.525	5.213	5.221	"
16	4.672	5.217	"	"
17	4.733	5.219	"	"
21	4.917	5.220	"	"

*The Values of $I_0(t)$ & $Q_0(t)$ are obtained from Table 4.

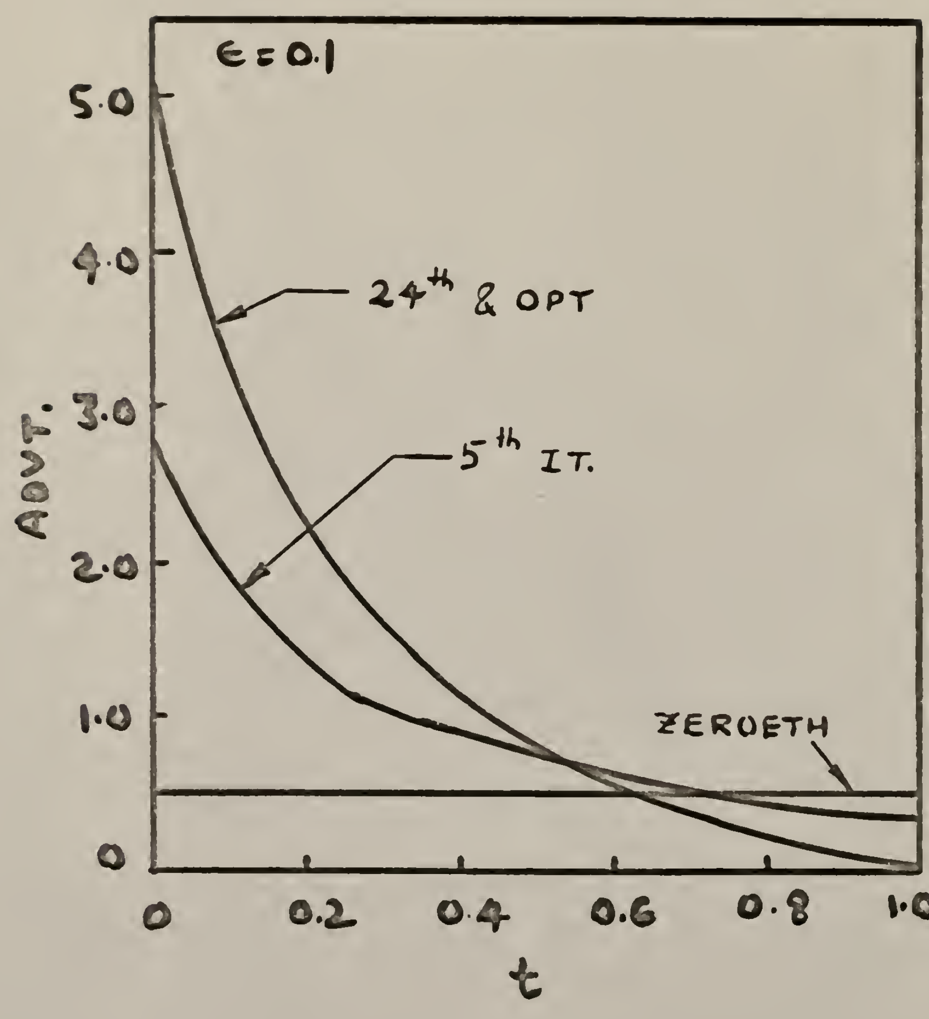

Fig. 19 Convergence rate of Advertisement.

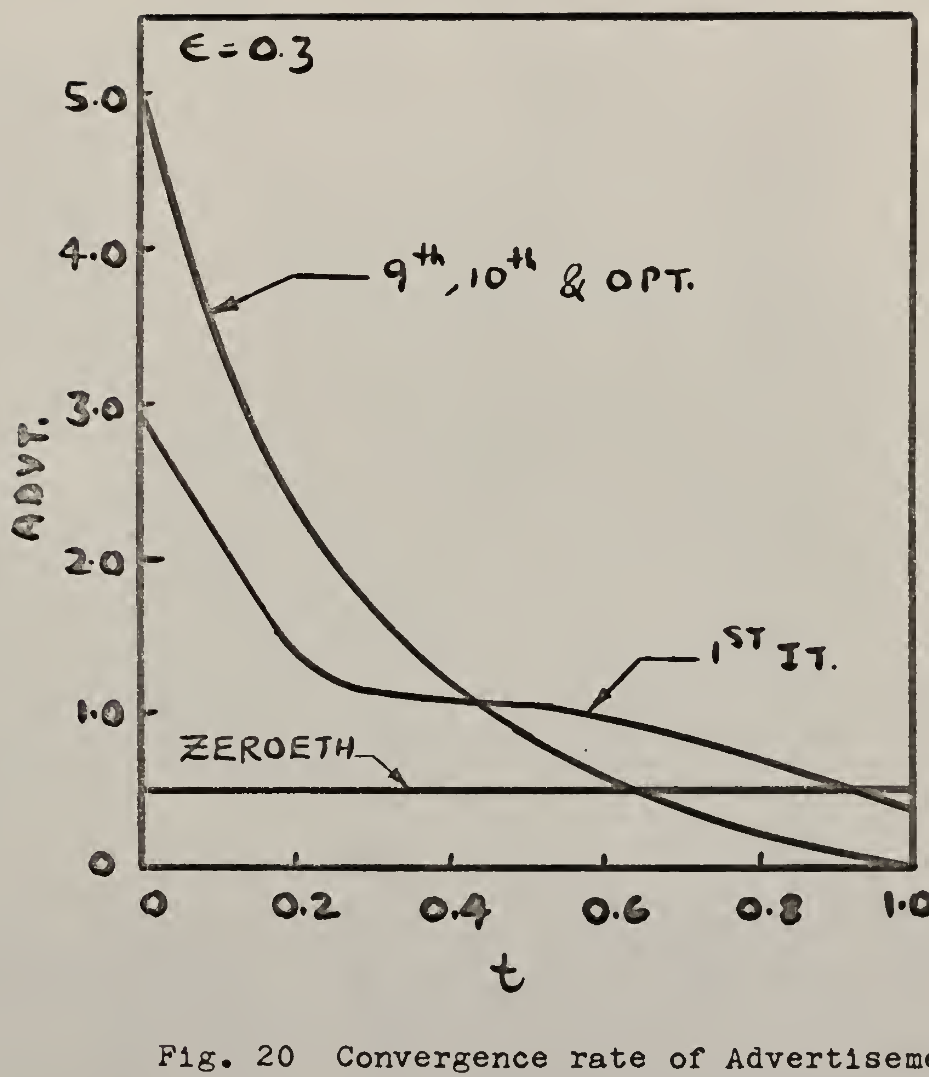

Fig. 20 Convergence rate of Advertisement.

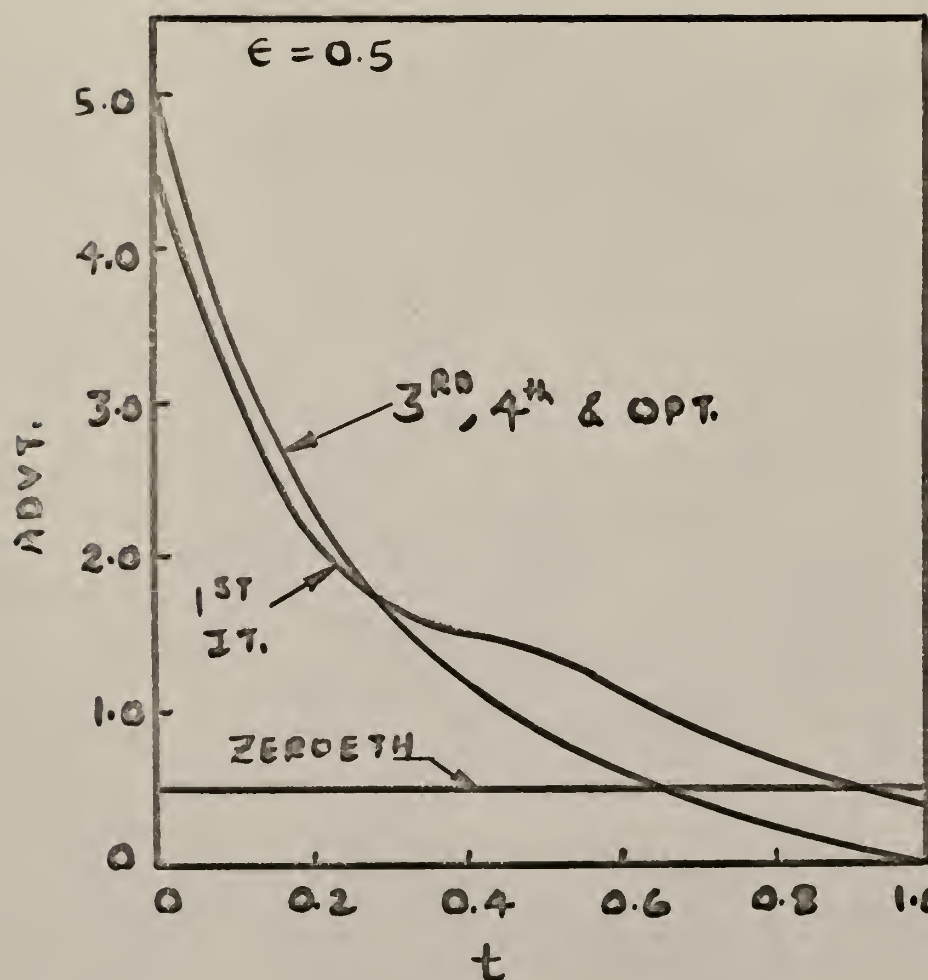

Fig. 21 Convergence rate of Advertisement.

Table 9

Starting Trajectories for Inventory, Sales and Advertisement, $0 \le t \le t_f$.

1	$I_0(t) = 0.2$		2	$I_0(t) = 0.5$
	$Q_0(t) = 0.2$			$Q_0(t) = 0.5$
	$A_0(t) = 0.5$			$A_0(t) = 2.0$
3	$I_0(t) = 0.5$		4	$I_0(t) = 0.6$
	$Q_0(t) = 1.0$			$Q_0(t) = 1.3$
	$A_0(t) = 2.0$			$A_0(t) = 2.0$
5	$I_0(t) = 0.6$			
	$Q_0(t) = 1.3$			
	$A_0(t) = 5.0$			

Table 10

Effect of ε on Rate of Convergence of

$I(t_f)$ with $I_0(t) = Q_0(t) = 0.2$, $A_0(t) = 0.5$, $0 \le t \le t_f$.

Iteration	ε = 0.1	ε = 0.3	ε = 0.5	ε = 0.7
1	0.8524	0.8524	0.8524	0.8524
5	0.7264	0.6343	0.6114	0.6322
10	0.6624	0.5999	0.5940	0.5928
15	0.6309	0.5945	0.5935	0.5935
20	0.6142	0.5936	"	"
25	0.6051	0.5935	"	"

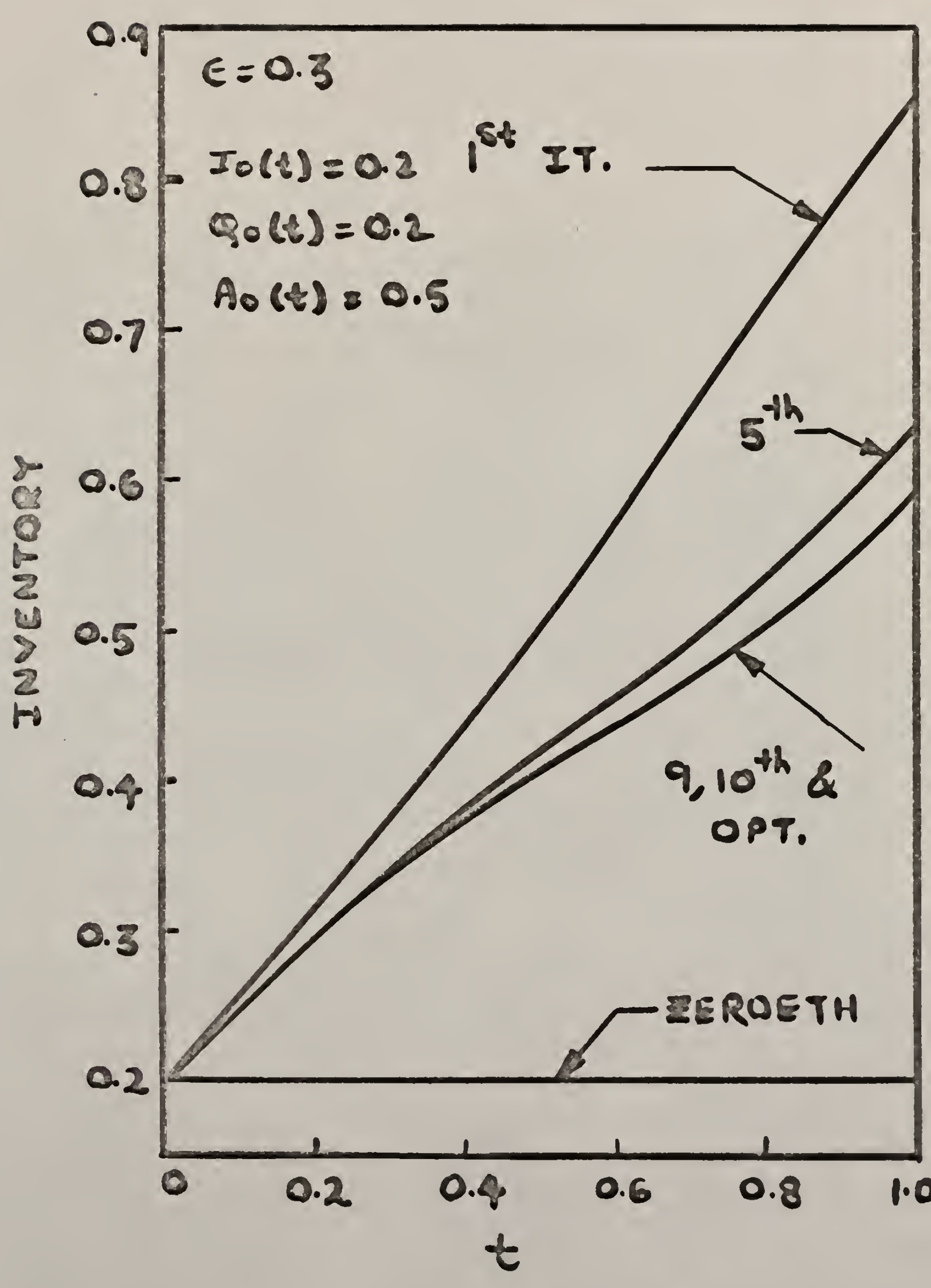

Fig. 22 Convergence rate of Inventory.

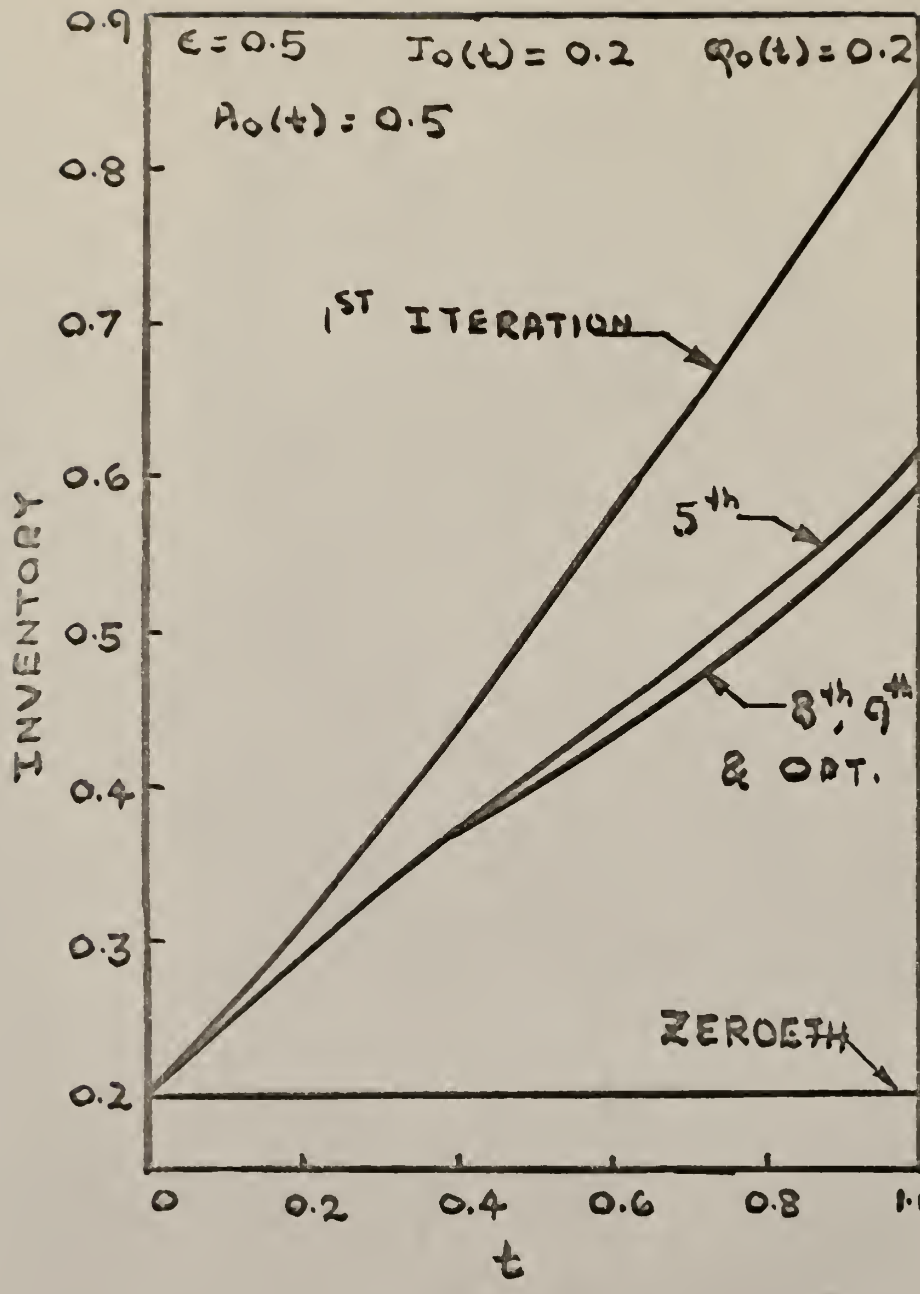

Fig. 23 Convergence rate of Inventory.

Table 11

Effect of ε on Rate of Convergence of

$Q(t_f)$, $I_0(t) = Q_0(t) = 0.2$, $A_0(t) = 0.5$, $0 \leq t \leq t_f$.

Iteration	ε = 0.1	ε = 0.3	ε = 0.5	ε = 0.7
1	0.9781	0.9781	0.9781	0.9781
5	1.083	1.178	1.198	1.165
10	1.153	1.215	1.222	1.223
15	1.185	1.221	"	1.222
20	1.202	1.222	"	"
25	1.211	"	"	"

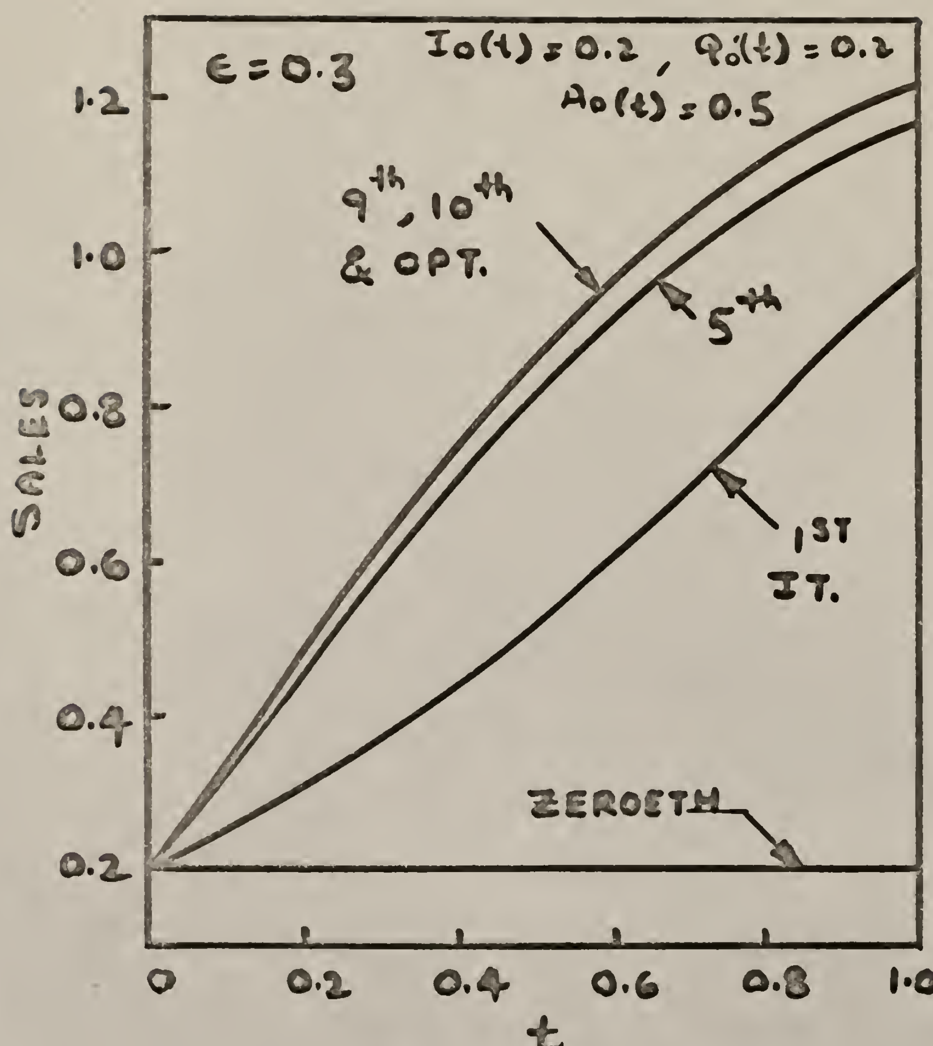

Fig. 24 Convergence rate of Sales.

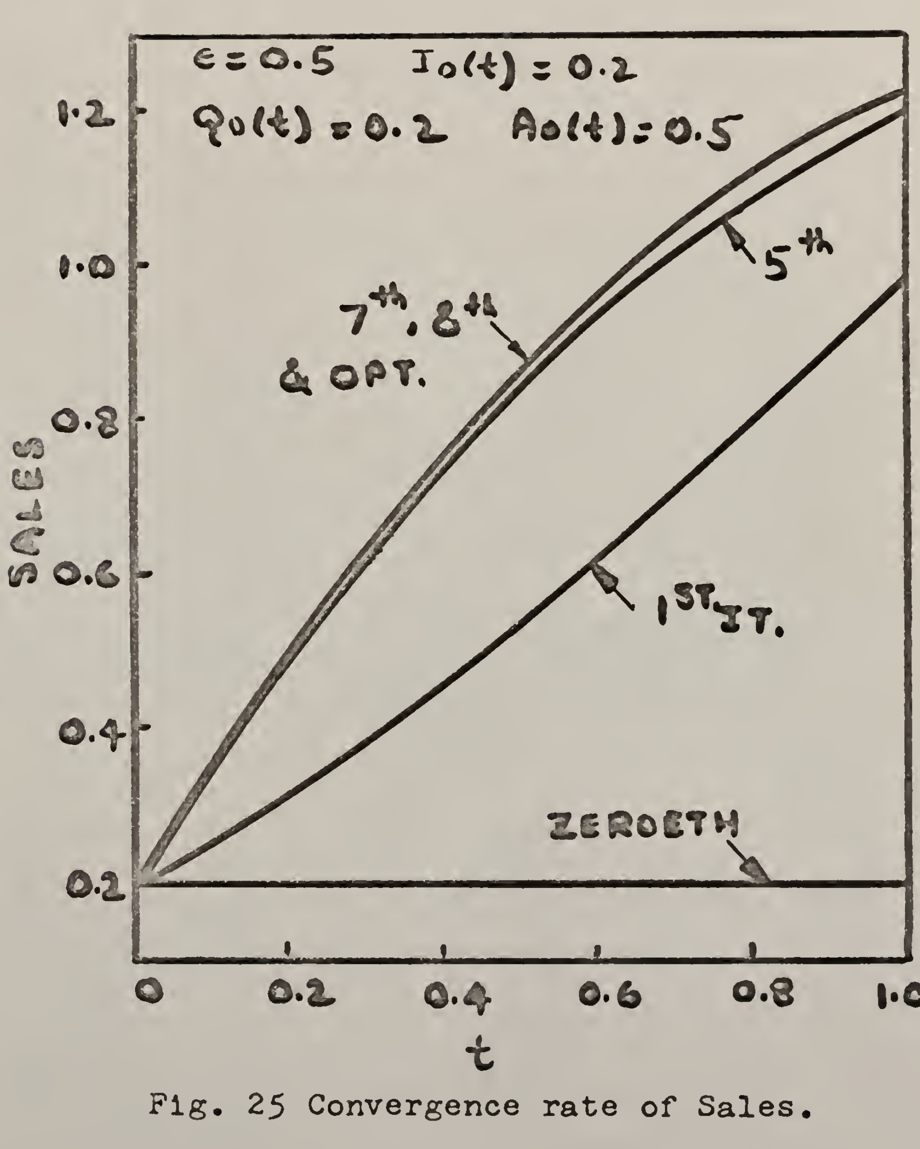

Fig. 25 Convergence rate of Sales.

Table 12

Effect of ε on Rate of Convergence of

Total Profit, $I_0(t) = Q_0(t) = 0.2$, $A_0(t) = 0.5$, $0 \leq t \leq t_f$.

Iteration	ε = 0.1	ε = 0.3	ε = 0.5	ε = 0.7
1	5.298	5.298	5.298	5.298
5	6.246	6.586	6.614	6.554
10	6.518	6.626	6.626	6.626
15	6.595	"	"	"
20	6.617	"	"	"
25	6.624	"	"	"

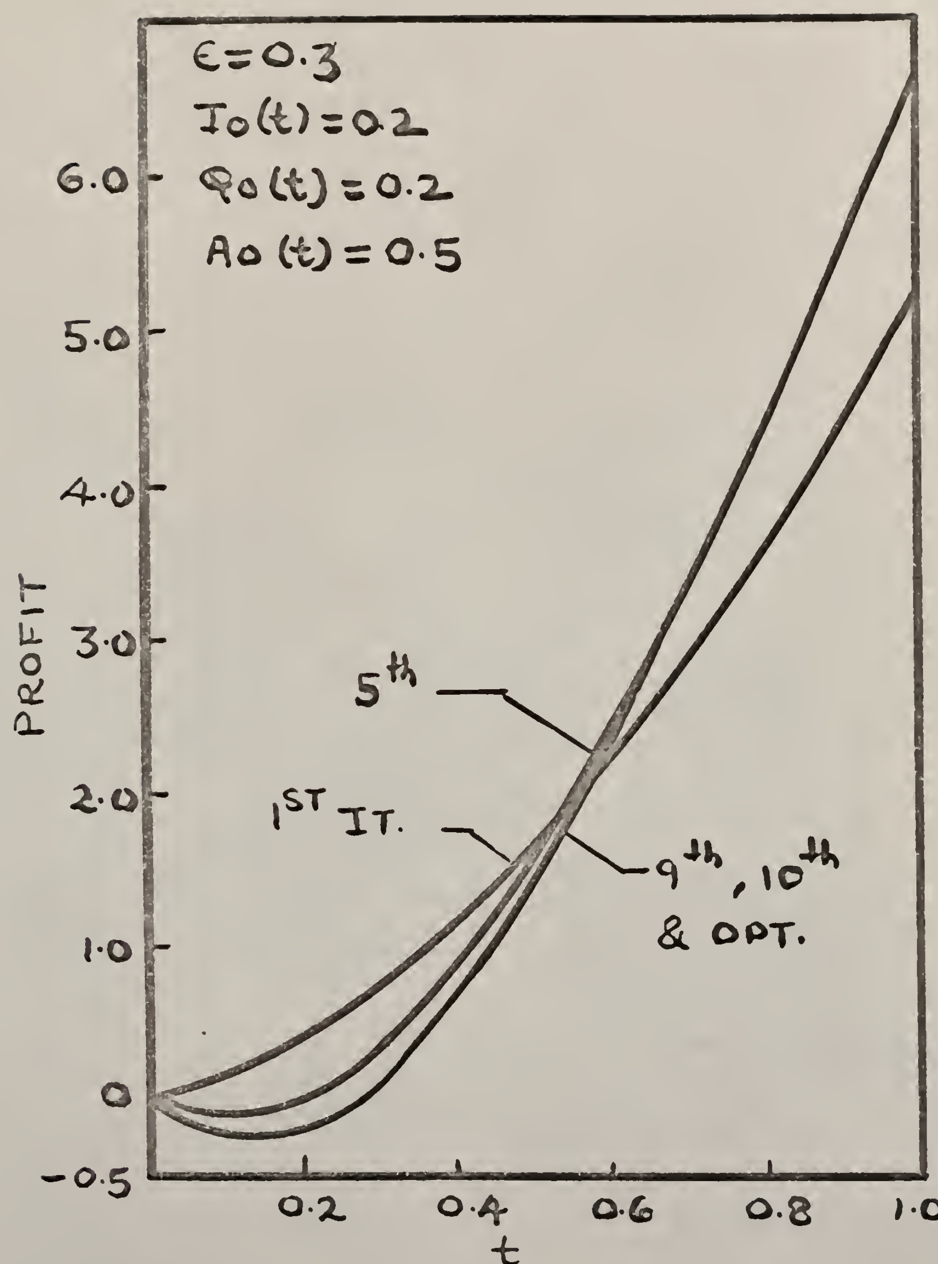

Fig. 26 Convergence rate of Profit

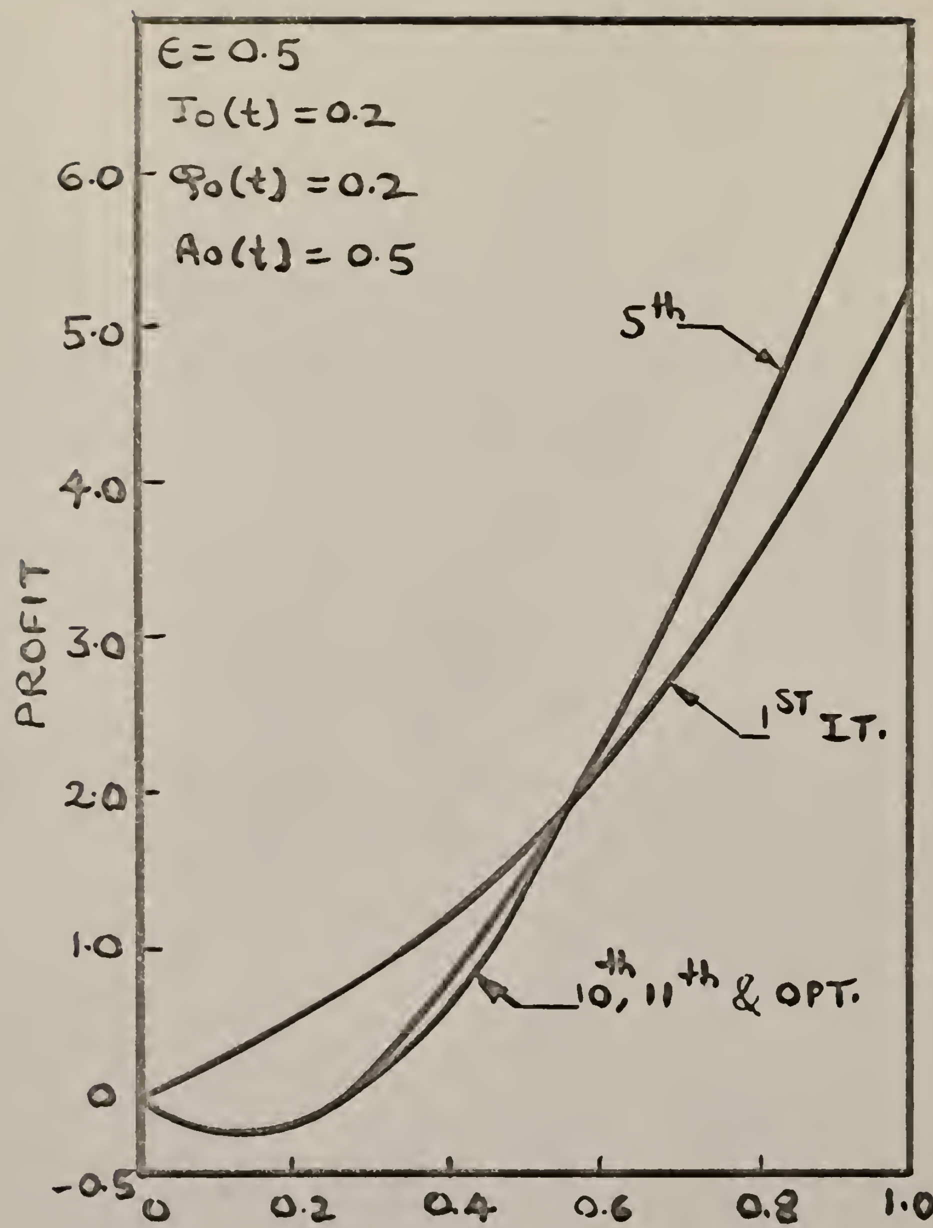

Fig. 27 Convergence rate of Profit.

Table 13

Effect of ε on Rate of Convergence of

$A(t_0)$, $I_0(t) = Q_0(t) = 0.2$, $A_0(t) = 0.5$, $0 \leq t \leq t_f$.

Iteration	$\varepsilon = 0.1$	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.7$
1	1.320	2.960	4.599	6.239
5	2.973	4.804	5.223	5.284
10	4.005	5.169	5.223	5.220
15	4.549	5.215	5.221	5.221
20	4.847	5.220	"	"
25	5.012	5.221	"	"

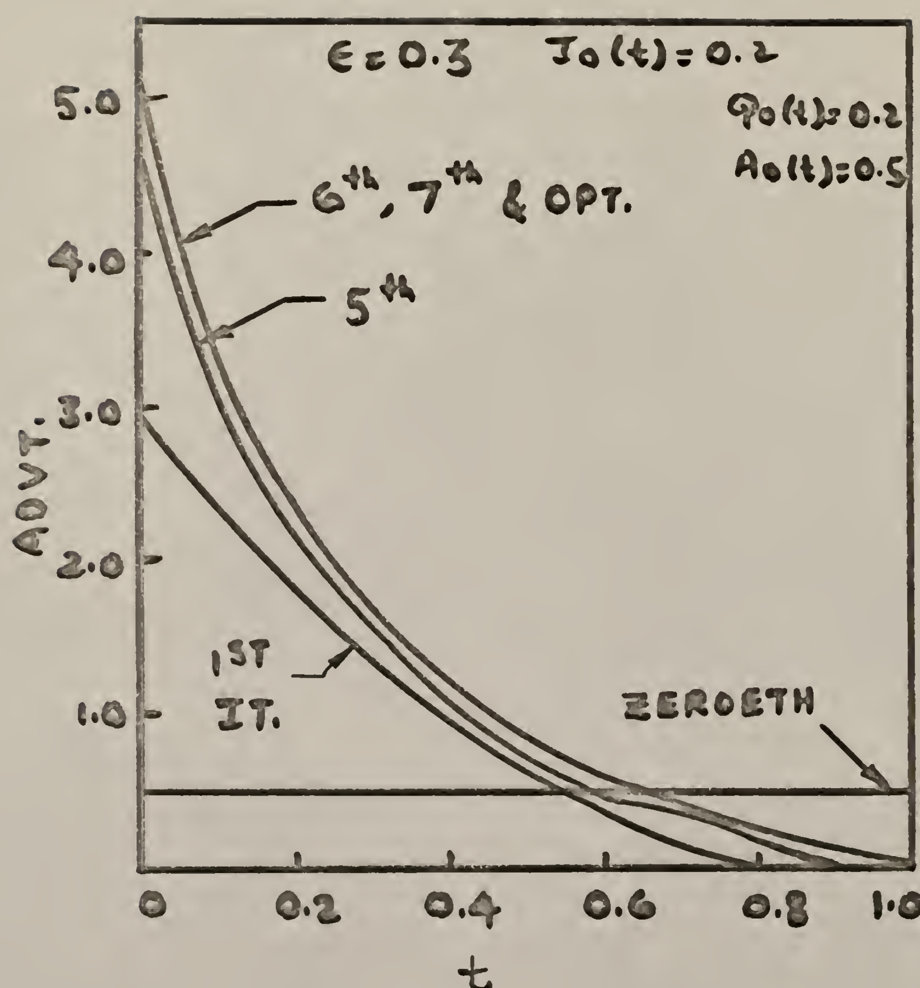

Fig. 28 Convergence rate of Advertisement.

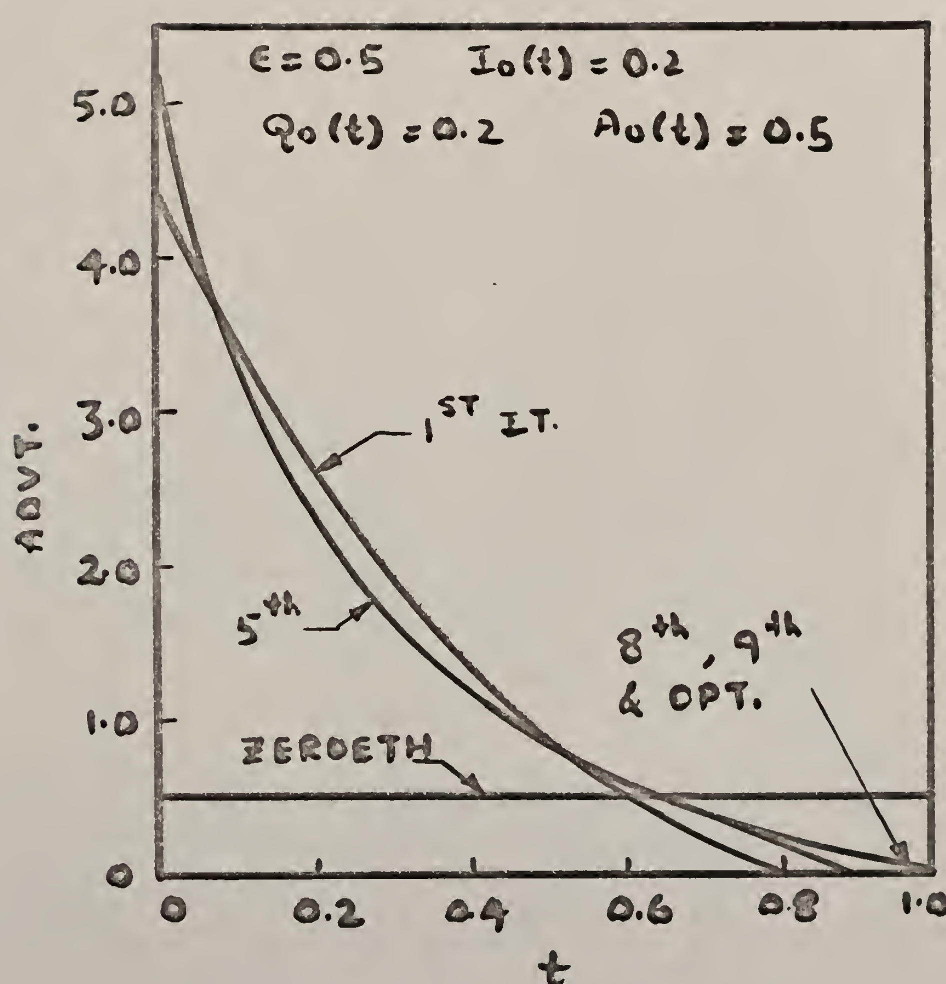

Fig. 29 Convergence rate of Advertisement.

Table 14

Effect of Different Starting Trajectories

on $I(t_f)$ with $\varepsilon = 0.4$.

Iteration	$I_0(t) = 0.5$ $Q_0(t) = 0.5$ $A_0(t) = 2.0$	$I_0(t) = 0.5$ $Q_0(t) = 1.0$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 5.0$
1	0.6134	0.6134	0.6134	0.3305
5	0.6024	0.5922	0.5876	0.5356
10	0.5945	0.5939	0.5936	0.5887
15	0.5936	0.5936	0.5935	0.5932
20	0.5935	0.5935	"	0.5934

Table 15

Effect of Different Starting Trajectories on

Rate of Convergence of $Q(t_f)$ with $\varepsilon = 0.4$.

Iteration	$I_0(t) = 0.5$ $Q_0(t) = 0.5$ $A_0(t) = 2.0$	$I_0(t) = 0.5$ $Q_0(t) = 1.0$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 5.0$
1	1.340	1.340	1.340	1.491
5	1.219	1.243	1.255	1.316
10	1.222	1.224	1.225	1.232
15	"	1.222	1.222	1.223
20	"	"	"	1.222

Table 16

Effect of Different Starting Trajectories on

Rate of Convergence of Total Profit with $\varepsilon = 0.4$.

Iteration	$I_0(t) = 0.5$ $Q_0(t) = 0.5$ $A_0(t) = 2.0$	$I_0(t) = 0.5$ $Q_0(t) = 1.0$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 5.0$.
1	4.668	4.668	4.668	-16.120
5	6.621	6.588	6.551	6.249
10	6.626	6.625	6.625	6.621
15	"	6.626	6.626	6.626
20	"	"	"	"

Table 17

Effect of Different Starting Trajectories on

Rate of Convergence of $A(t_0)$ with $\varepsilon = 0.4$.

Iteration	$I_0(t) = 0.5$ $Q_0(t) = 0.5$ $A_0(t) = 2.0$	$I_0(t) = 0.5$ $Q_0(t) = 1.0$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 2.0$	$I_0(t) = 0.6$ $Q_0(t) = 1.3$ $A_0(t) = 5.0$
1	2.330	1.442	-1.176	1.823
5	4.910	4.677	4.556	4.476
10	5.210	5.182	5.173	5.151
15	5.220	5.218	5.217	5.215
20	5.221	5.221	5.221	5.220

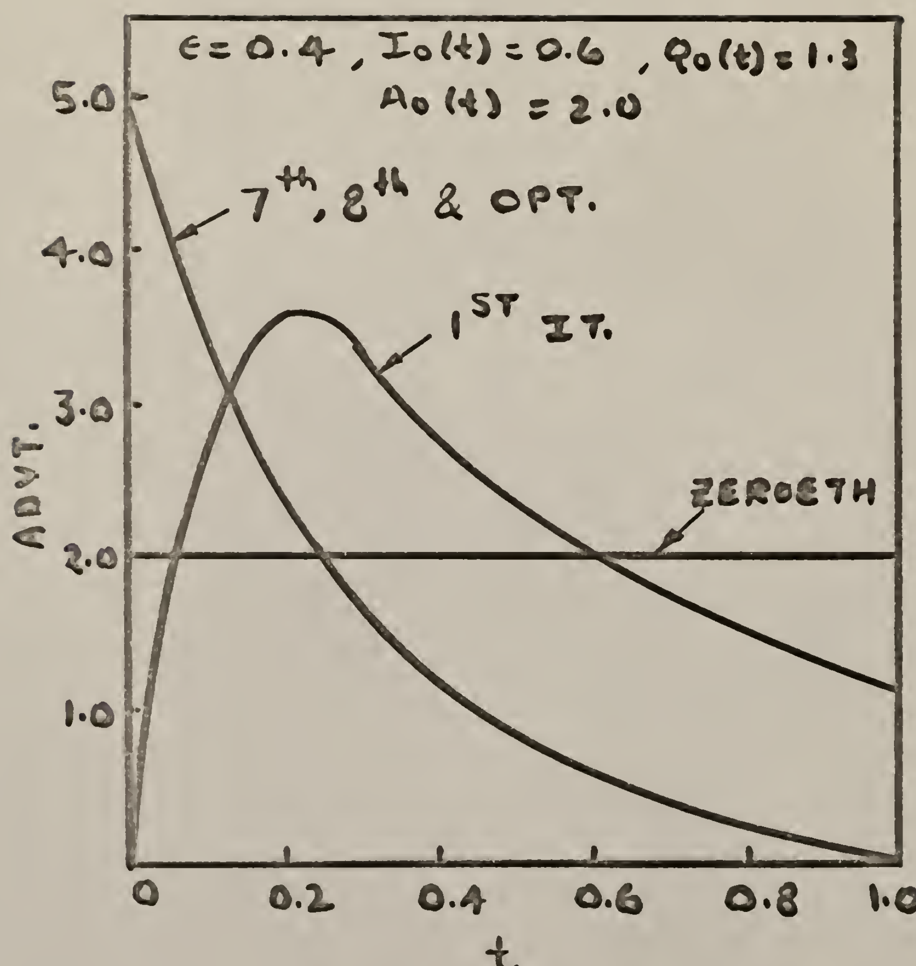

Fig. 30 Convergence rate of Advertisement.

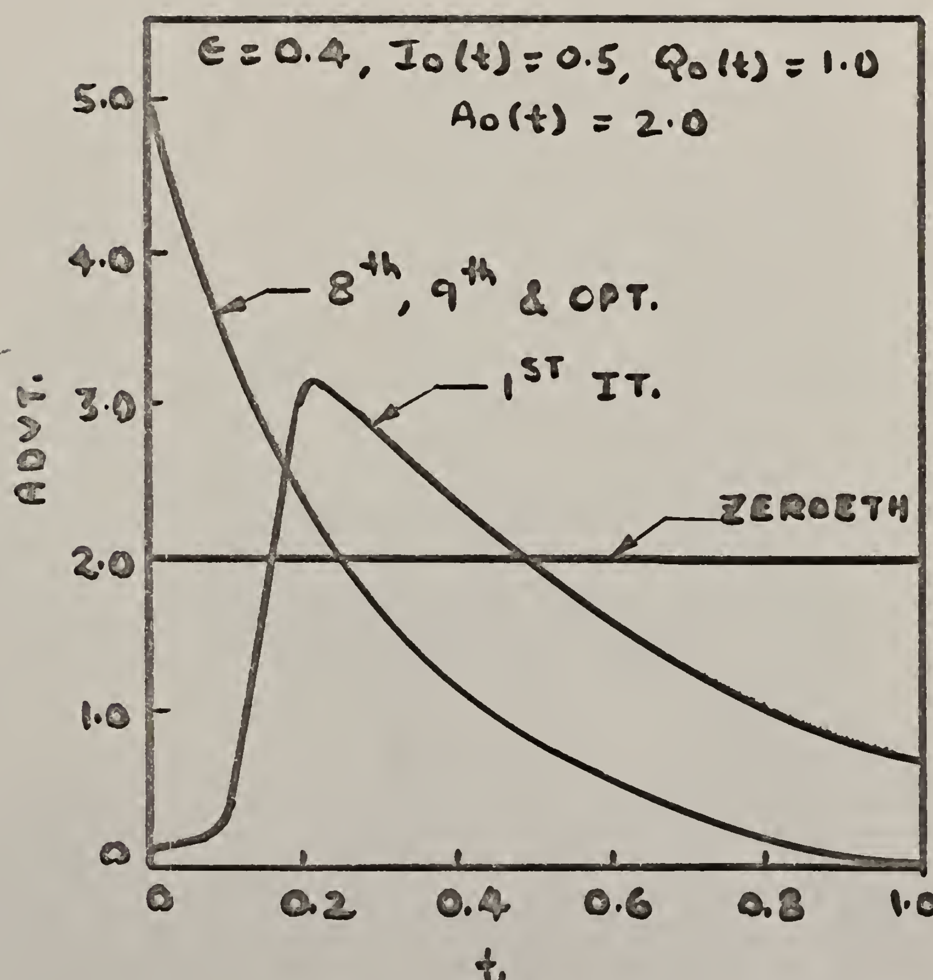

Fig. 31 Convergence rate of Advertisement.

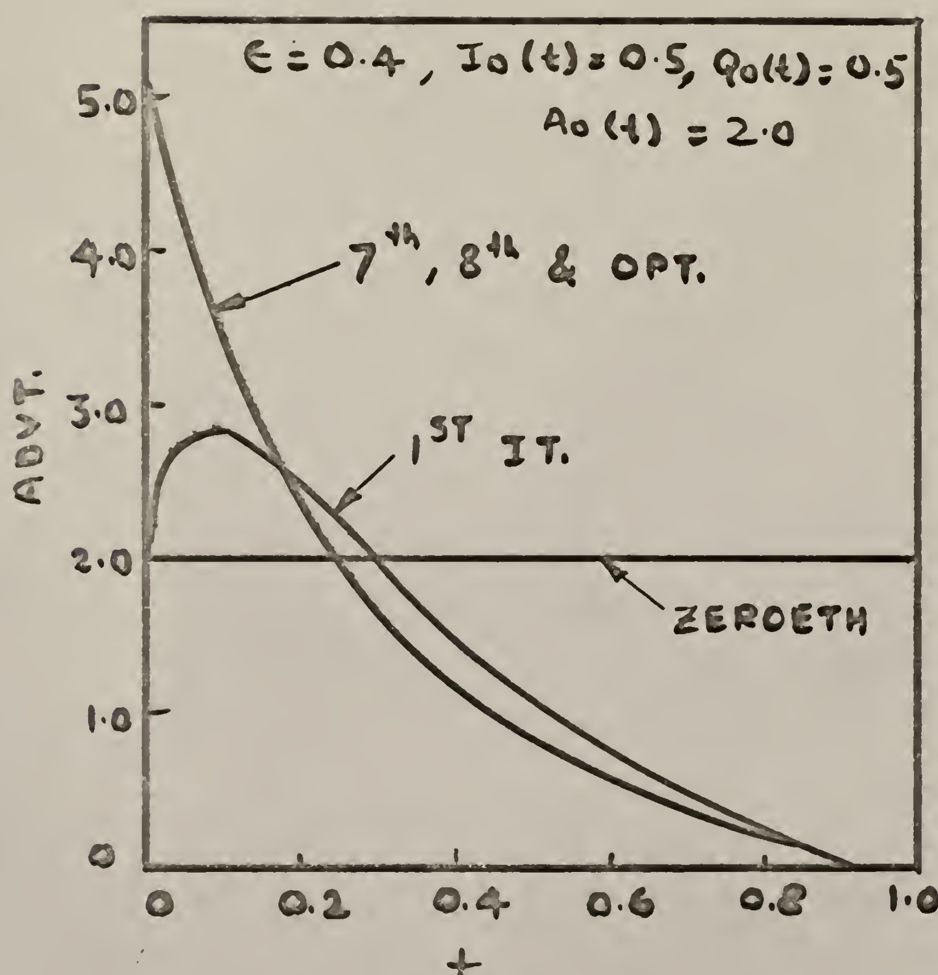

Fig. 32 Convergence rate of Advertisement.

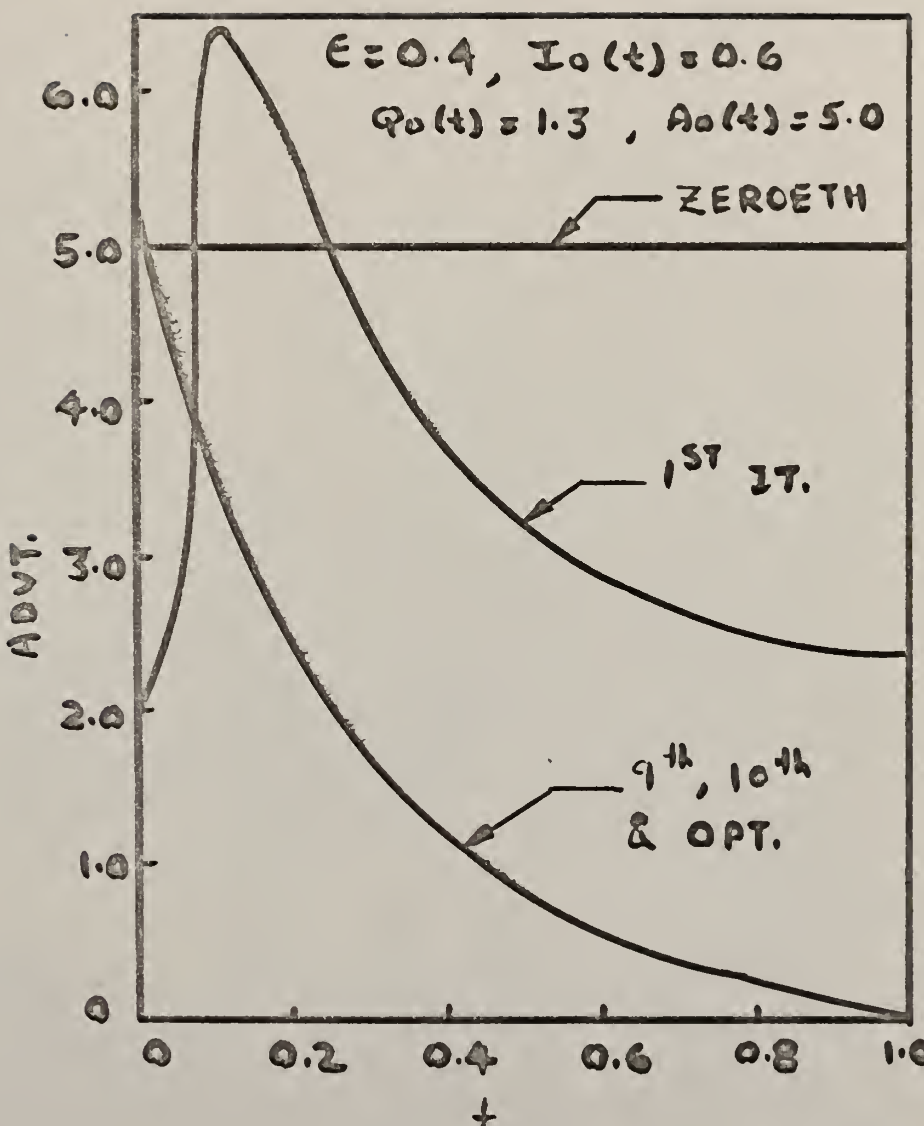

Fig. 33 Convergence rate of Advertisemnt.

The results starting with this trajectory were explored in detail with different value of ε . Tables 5 through 8 show the convergence rate of inventory, sales, profit function and advertisement, respectively, for the different values of ε .

Figures 10 through 12 show the convergence rate of inventory for different values of ε . Similarly, Figs.13 through 15 show the convergence rate of sales, Figs. 16 through 18 of profit function and Figs. 19 through 21 of advertisement for different values of ε . The maximum value of ε that would lead to convergence in this case was found to be 0.7. $\varepsilon = 1.0$ would lead to exponential overflow in this situation. Another interesting point noted was that almost the same convergence rate was obtained with $\varepsilon = 0.5$ and with $\varepsilon = 0.7$. Thus a higher ε did not increase the convergence rate.

In an another approach to this problem, a number of different starting trajectories for inventory, sales and advertisement were used. These are listed in Table 9. Set (1) of the various trajectories listed in Table 9 was explored in detail with different values of ε .

Tables 10 through 13 show the convergence rate of inventory, sales, profit function and advertisement respectively, for different values of ε . Figures 22 and 23 show the convergence rate of inventory for different values of ε . Similarly Figs. 24 and 25 show the convergence rate of sales, Figs. 26 and 27 of profit function and Figs. 28 and 29 of advertisement for different values of ε .

The remaining starting trajectories from Table 9, namely sets (1) through (5) were tried with $\varepsilon = 0.4$. Tables 14 through 17 list the convergence rate of advertisement for these trajectories. Figs. 30 through 33 show the convergence rate of advertisement for these different trajectories.

The starting trajectories (1) through (5) from Table 9 led to convergence almost in the same number of iterations. The maximum value of ε that would lead to convergence was found to be 0.7 in this case also.

Thus it is seen that the problem is very stable and that the optimum can be reached almost with any reasonable values of the starting trajectories.

Another computational feature that was encountered in the solution of this problem was regarding the numerical solution of the differential equations. As their number increased it was found advisible to use the IBM subroutine "RKGS" for their numerical solution. However, this subroutine imposed the problem of accuracy which has to be specified by the user. This is the accuracy against which the results are checked after each integration step. If the accuracy is too low, the integration step size is halved and this continues until the specified accuracy is obtained. Thus, if the accuracy is not appropriate, the grid points may not be the ones desired by the user. The calculations of $\underline{R}$, $\underline{S}$, and $\underline{T}$ should be done both in subroutine "FCT" and "OUTP". (See appendix 7.2) Also, to test the fact that this method would lead to convergence at the nearest stationary point regardless of whether it is a maximization or a minimization problem, the objective function was made negative and the same problem solved again. The results agree in both the cases. Thus whether a maximum or minimum will be reached all depends on the nature of the curve of the objective function.

3.3 A Chemical Manufacturing Problem with Advertisement

The Model

Figure 34 represents a chemical manufacturing process and stages 1 and 2 represent two reactors. The raw material entering the first reactor is a mixture of A and B. After the second stage, the product A and product B are separated, as is the remaining raw material, product C. Product B is the more valuable of the three products and, to enhance its sale, it has to be advertised. Also, to meet the fluctuations in its demand, a certain amount of inventory has to be kept. It shall be assumed that the demands for products A and C are unlimited.

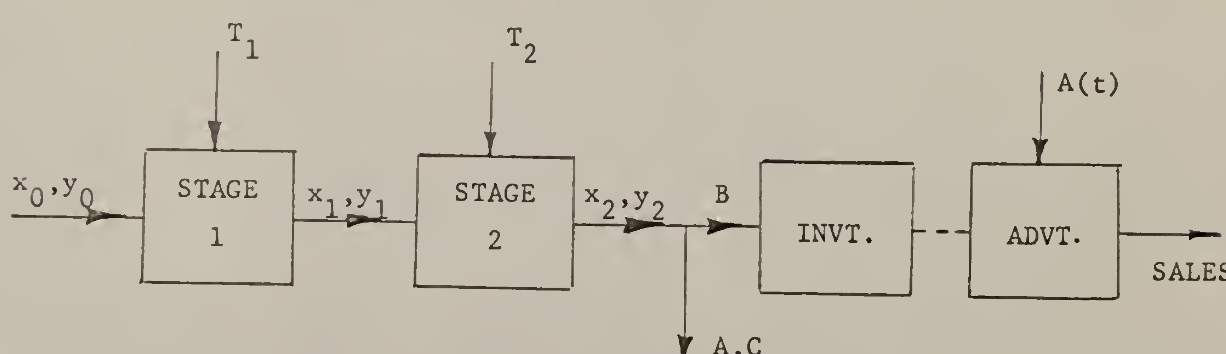

Fig. 34

Let x_0, and y_0 represent the concentration of A and B in the original raw material before it enters the first stage or reactor. Similarly, let x_1, y_1 and x_2, y_2 represent the concentrations of A and B before and after the second stage, respectively. To bring about this reaction, temperatures T_1 and T_2 have to be applied to the two reactors. The reactions in the reactor can be represented by the following equations:

Let q = flow rate

v_1 = volume of the first reactor

v_2 = volume of the second reactor.

Then,

$$v_1 \frac{dx_1}{dt} = q(x_0 - x_1) - v_1 \, Ka_1 \, x_1 \tag{79}$$

$$v_1 \frac{dy_1}{dt} = q(y_0 - y_1) - v_1 \, Kb_1 \, y_1 + v_1 \, Ka_1 \, x_1 \tag{80}$$

$$v_2 \frac{dx_2}{dt} = q(x_1 - x_2) - v_2 \, Ka_2 \, x_2 \tag{81}$$

$$v_2 \frac{dy_2}{dt} = q(y_1 - y_2) - v_2 \, Kb_2 \, y_2 + v_2 \, Ka_2 \, x_2 \tag{82}$$

where

$$Ka_1 = G_a \exp(-Ea/RT_1)$$

$$Ka_2 = G_a \exp(-Ea/RT_2)$$

$$Kb_1 = G_b \exp(-Eb/RT_1)$$

$$Kb_2 = G_b \exp(-Eb/RT_2).$$

This completes the production part of the system. Now consider the inventory. The rate of change of inventory is the difference between the rate of production of B and its rate of sale. If I(t) represents the inventory at time t, then

$$\frac{dI(t)}{dt} = qy_2 - C_a\, K(t) \tag{83}$$

The sales equation is assumed similar to the problem in Para. 3.2.

$$\frac{dK(t)}{dt} = [C + A(t)] \cdot K(t) \cdot \left(1 - \frac{K(t)}{N}\right) \tag{84}$$

Equations (79) through (84) represents the performance equations of the whole system under consideration.

This problem has six state variables, namely x_1, y_1, x_2, y_2, I(t). K(t) and three control variables namely T_1, T_2 and A(t).

The profit function can be formulated as:

Profit = (sales revenue from A,B,C)-(cost of holding the inventory for B) - (cost of advertising for B) - (cost of production)

Sales revenue from A, B and C is = $C_1 C_q K(t) + C_2 q x_2 + C_3 q\,(1 - x_2 - y_2)$

where, C_1, C_2, C_3 represent the unit sales prices for A, B, C respectively.

Cost of holding the inventory of B = $C_I\,(I_M - I(t))^2$ where I_M is the capacity of the warehouse and C_I = inventory carrying cost.

Cost of advertising = $C_A\, A^2(t)\, K^2(t)$.

Cost of production comes from the fact that the two reactors have to be supplied with heat energy in order to obtain the desired temperature. Let C_T represent the cost of raising the reactor temperature by a unit degree. Then the cost of production becomes

$$= C_T\left\{(T_{1m} - T_1)^2 + (T_1 - T_2)^2\right\}$$

where T_{1m} is the temperature of the entering raw material. Thus the function to be maximized is

$$J = \int_0^{t_f} C_1C_qK(t) + C_2qx_2 + C_3q(1-x_2-y_2) - C_I[I_m-I(t)]^2$$

$$- C_AA^2(t)K^2(t) - C_T[(T_{1m}-T_1)^2 + (T_1-T_2)^2]\Big\} \, dt$$

$$\frac{dJ}{dt} = C_1C_qK(t) + C_2qx_2 + C_3q(1-x_2-y_2) - C_I[I_m-I(t)]^2$$

$$- C_AA^2(t)K^2(t) - C_T[(T_{1m}-T_1)^2 - (T_1-T_2)^2] \quad . \tag{85}$$

Recursive Relations

The necessary relations for the second variation can be obtained in the following manner. The various derivatives can be obtained as follows

$$\frac{\partial J}{\partial \underline{x}} = \begin{pmatrix} \frac{\partial J}{\partial x_1} \\ \frac{\partial J}{\partial y_1} \\ \frac{\partial J}{\partial x_2} \\ \frac{\partial J}{\partial y_2} \\ \frac{\partial J}{\partial I} \\ \frac{\partial J}{\partial K} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ q(C_2-C_3) \\ -C_3q \\ 2C_I(I_m-I(t)) \\ C_1C_q-C_AA^2(t)\cdot 2\cdot K(t) \end{pmatrix}$$

$$\frac{\partial^2 J}{\partial \underline{x}^2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2C_I & 0 \\ 0 & 0 & 0 & 0 & 0 & -2C_A A^2(t) \end{bmatrix}$$

$$\frac{\partial^2 J}{\partial \underline{\theta} \partial \underline{x}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\frac{\partial J}{\partial \underline{\theta}} = \begin{bmatrix} 2C_T(T_{1m}-T_1) + 2C_T(T_1-T_2) \\ 2C_T(T_1-T_2) \\ -2C_A A(t) K^2(t) \end{bmatrix}$$

$$\frac{\partial^2 J}{\partial \underline{\theta}^2} = \begin{bmatrix} 0 & 2C_T & 0 \\ -2C_T & -2C_T & 0 \\ 0 & 0 & -2C_A K^2(t) \end{bmatrix}$$

$$\frac{\partial f'}{\partial \underline{x}} = \begin{bmatrix} BM_1 & Ga\cdot EAT_1 & (q/V_2 & 0 & 0 & 0 \\ 0 & BM_2 & 0 & q/V_2 & 0 & 0 \\ 0 & 0 & BM_3 & Ga\cdot EAT_2 & 0 & 0 \\ 0 & 0 & 0 & BM_4 & q & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -C_q & BM_5 \end{bmatrix} .$$

where,

$$BM_1 = -(q/V_1) - Ga\cdot EAT_1, \qquad EAT_1 = \exp(-\frac{EA}{RT_1})$$

$$BM_2 = -(q/V_1) - Gb\cdot EBT_1, \qquad EAT_2 = \exp(-\frac{EA}{RT_2})$$

$$BM_3 = -(q/V_2) - Ga\cdot EAT_2, \qquad EBT_1 = \exp(-\frac{EB}{RT_1})$$

$$BM_4 = -(q/V_2) - Gb\cdot EBT_2, \qquad EBT_2 = \exp(-\frac{EB}{RT_2})$$

$$BM_5 = (C+A(t)) - Gb\cdot EBT_2$$

$$[\frac{\partial f'}{\partial \underline{x}}]' = \begin{pmatrix} BM_1 & 0 & 0 & 0 & 0 & 0 \\ Ga\cdot EAT_1 & BM_2 & 0 & 0 & 0 & 0 \\ q/V_2 & 0 & BM_3 & 0 & 0 & 0 \\ 0 & q/V_2 & Ga\cdot EAT_2 & BM_4 & 0 & 0 \\ 0 & 0 & 0 & q & 0 & -C_q \\ 0 & 0 & 0 & 0 & 0 & BM_5 \end{pmatrix}$$

$$\frac{\partial f'}{\partial \underline{\theta}} = \begin{pmatrix} FT_1 & FT_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & FT_3 & FT_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & FT_5 \end{pmatrix}$$

where

$$FT_1 = - \frac{Ea}{RT_1^2} G_a x_1 e^{-\frac{Ea}{RT_1}}$$

$$FT_2 = - \frac{Eb}{RT_1^2} G_b y_1 e^{-\frac{Eb}{RT_1}} + \frac{Ea}{RT_1^2} G_a x_1 e^{-\frac{Ea}{RT_1}}$$

$$FT_3 = - \frac{Ea}{RT_2^2} G_a x_2 e^{-\frac{Ea}{RT_2}}$$

$$FT_4 = -\frac{Eb}{RT_2^2} G_b y_2 e^{-\frac{Eb}{RT_2}} + \frac{Ea}{RT_2^2} G_a x_2 e^{-\frac{Ea}{RT_2}}$$

$$FT_5 = K(t)\ [1 - \frac{K(t)}{N}]$$

$$\frac{\partial^2 f_1}{\partial \underline{x}^2} = \frac{\partial^2 f_2}{\partial \underline{x}^2} = \frac{\partial^2 f_3}{\partial \underline{x}^2} = \frac{\partial^2 f_4}{\partial \underline{x}^2} = \frac{\partial^2 f_5}{\partial \underline{x}^2} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial^2 f_6}{\partial \underline{x}^2} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{2}{N}[C+A(t)] \end{pmatrix}$$

$$\frac{\partial f_1}{\partial \underline{x}} = \begin{pmatrix} BM_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_1}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} G_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_2}{\partial \underline{x}} = \begin{pmatrix} Ga \cdot EAT_1 \\ BM_2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_2}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} -G_1 & G_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_3}{\partial \underline{x}} = \begin{pmatrix} q/V_2 \\ 0 \\ -q/V_2 - Ga \cdot EAT_2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_3}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & G_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_4}{\partial \underline{x}} = \begin{pmatrix} 0 \\ q/V_2 \\ Ga \cdot EAT_2 \\ BM_4 \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_4}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -G_3 & G_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_5}{\partial \underline{x}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ q \\ 0 \\ -C_q \end{pmatrix} \qquad \frac{\partial^2 f_5}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_6}{\partial \underline{x}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ [1 - \frac{2K(t)}{N}][C+A(t)] \end{pmatrix} \qquad \frac{\partial^2 f_6}{\partial \underline{x}\, \partial \underline{\theta}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & [1 - \frac{2K(t)}{N}] \end{pmatrix}$$

where,

$$G_1 = -\frac{EA}{RT_1^2} Ga \exp(-\frac{EA}{RT_1})$$

$$G_3 = -\frac{EA}{RT_2^2} Ga \exp(-\frac{EA}{RT^2})$$

$$G_2 = -\frac{EB}{RT_1^2} Gb \exp(-\frac{EB}{RT_1})$$

$$G_4 = -\frac{EB}{RT_2^2} Gb \exp(-\frac{Eb}{RT_2})$$

$$\frac{\partial f_1}{\partial \underline{\theta}} = \begin{pmatrix} -\frac{Ea}{RT_1^2} G_a x_1 e^{-\frac{Ea}{RT_1}} \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_1}{\partial \underline{\theta}^2} = \begin{pmatrix} -G_1 x_1 (\frac{2}{T_1} - \frac{Ea}{RT_1^2}) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_2}{\partial \underline{\theta}} = \begin{pmatrix} \left(-\frac{Eb}{RT_1^2} G_b y_1 e^{-\frac{Eb}{RT_1}} + \frac{Ea}{RT_1^2} G_a x_1 e^{-\frac{Ea}{RT_1}} \right) \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_2}{\partial \underline{\theta}^2} = \begin{pmatrix} -G_2 y_1 (\frac{2}{T_1} - \frac{Eb}{RT_1^2}) + G_1 x_1 (-\frac{2}{T_1} + \frac{Ea}{RT_1^2}) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_3}{\partial \underline{\theta}} = \begin{pmatrix} 0 \\ -\frac{Ea}{RT_2^2} G_a x_2 e^{-\frac{Ea}{RT_2}} \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_3}{\partial \underline{\theta}^2} = \begin{pmatrix} 0 & 0 & 0 \\ & -G_3(x_2)(\frac{2}{T_2} - \frac{Ea}{RT_2^2}) & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_4}{\partial \underline{\theta}} = \begin{pmatrix} 0 \\ \left(-\frac{Eb}{RT_2^2} G_b y_2 e^{-\frac{Eb}{RT_2}} + \frac{Ea}{RT_2^2} G_a x_2 e^{-\frac{Ea}{RT_2}} \right) \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_4}{\partial \underline{\theta}^2} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -G_4 y_2(\frac{2}{T^2} - \frac{Eb}{RT_2^2}) - G_3 x_2(-\frac{2}{T_2} + \frac{Ea}{RT_2^2}) & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_5}{\partial \underline{\theta}} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad \frac{\partial^2 f_5}{\partial \underline{\theta}^2} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\frac{\partial f_6}{\partial \underline{\theta}} = \begin{bmatrix} 0 \\ 0 \\ K(t)(1 - \frac{K(t)}{N}) \end{bmatrix} \qquad \frac{\partial^2 f_6}{\partial \underline{\theta}^2} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\underline{P} = \begin{bmatrix} P_{11} & P_{21} & P_{31} & P_{41} & P_{51} & P_{61} \\ P_{21} & P_{22} & P_{32} & P_{42} & P_{52} & P_{62} \\ P_{31} & P_{32} & P_{33} & P_{43} & P_{53} & P_{63} \\ P_{41} & P_{42} & P_{43} & P_{44} & P_{54} & P_{64} \\ P_{51} & P_{52} & P_{53} & P_{54} & P_{55} & P_{65} \\ P_{61} & P_{62} & P_{63} & P_{64} & P_{65} & P_{66} \end{bmatrix}$$

Now the expressions for $\underline{R}$, $\underline{S}$, $\underline{T}$ which are required for obtaining the second variational Equations (19), (28) and (48) may be determined. Thus

$$\underline{R} = \frac{\partial^2 J}{\partial \underline{\theta}\ \partial \underline{x}} + \frac{\partial f'}{\partial \underline{\theta}} \underline{P} + \sum_{i=1}^{6} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}\ \partial \underline{x}}$$

Let

$$\underline{R} = \begin{bmatrix} R_1 & R_4 & R_7 & R_{10} & R_{13} & R_{16} \\ R_2 & R_5 & R_8 & R_{11} & R_{14} & R_{17} \\ R_3 & R_6 & R_9 & R_{12} & R_{15} & R_{18} \end{bmatrix}$$

where

$$R_1 = P_{11}\ FT1 + P_{21}\ FT2 + z_1G_1 - z_2G_1$$

$$R_2 = P_{31}\ FT3 + P_{41}\ FT4$$

$$R_3 = P_{61}\ FT5$$

$$R_4 = P_{21}\ FT1 + P_{22}\ FT2 + z_2G_2$$

$$R_5 = P_{32}\ FT3 + P_{42}\ FT4$$

$$R_6 = P_{62}\ FT5$$

$$R_7 = P_{31}\ FT1 + P_{32}\ FT2$$

$$R_8 = P_{33}\ FT3 + P_{43}\ FT4 + z_3G_3 - z_4G_3$$

$$R_9 = P_{63}\ FT5$$

$$R_{10} = P_{41}\ FT1 + P_{42}\ FT2$$

$$R_{11} = P_{43}\ FT3 + P_{44}\ FT4 + z_4G_4$$

$$R_{12} = P_{64}\ FT5$$

$$R_{13} = P_{51}\ FT1 + P_{52}\ FT2$$

$$R_{14} = P_{53}\ FT3 + P_{54}\ FT4$$

$$R_{15} = P_{65}\ FT5$$

$$R_{16} = P_{61}\ FT1 + P_{62}\ FT2$$

$$R_{17} = P_{63}\ FT3 + P_{64}\ FT4$$

$$R_{18} = P_{66}\ FT5 + z_6\ [1 - \frac{2K(t)}{N}] - 4C_A A(t)\ K(t)$$

From Equation (46)

$$S = \frac{\partial J}{\partial \underline{\theta}} + \sum_{i=1}^{6} z_i \frac{\partial f_i}{\partial \underline{\theta}}$$

Let

$$S = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \end{bmatrix}$$

$$\equiv \begin{bmatrix} 2C_T(T_{1m}-T_1) + 2C_T(T_1-T_2) + z_1G_1x_1 + (G_2y_1 - G_1x_1)\ z_2 \\ 2C_T(T_1-T_2) + z_3x_2G_3 + z_4(y_2G_4 - x_2G_3) \\ z_6\ K(t)\ [1 - \frac{K(t)}{N}] - 2C_AA(t)\ K^2(t) \end{bmatrix}$$

From Equation 47

$$\underline{T} = \frac{\partial^2 J}{\partial \underline{\theta}^2} + \sum_{i=1}^{6} z_i \frac{\partial^2 f_i}{\partial \underline{\theta}^2}$$

Let

$$\underline{T} = \begin{bmatrix} DT1 & DT4 & DT7 \\ DT2 & DT5 & DT9 \\ DT3 & DT6 & DT9 \end{bmatrix}$$

where

$$DT1 = - z_1G_1x_1 \left(\frac{2}{T_1} - \frac{Ea}{RT_1^2}\right) - G_2z_2y_1 \left(\frac{2}{T_1} - \frac{Eb}{RT_1^2}\right)$$

$$- z_2G_1x_1 \left(- \frac{2}{T_1} + \frac{Ea}{RT_1^2}\right)$$

$$DT2 = - 2C_T$$

$$DT3 = 0$$

$DT4 = 2C_T$

$$DT5 = -\ 2C_T - z_3G_3x_2\ (\frac{2}{T_2} - \frac{Ea}{RT_2^2}) - z_4G_4y_2\ (\frac{2}{T_2} - \frac{Eb}{RT_2^2})$$

$$-\ G_3z_4x_2\ (-\frac{2}{T_2} + (-\frac{2}{T_2} + \frac{Ea}{RT_2^2})$$

$DT6 = 0$

$DT7 = 0$

$DT8 = 0$

$DT9 = -\ 2C_A\ K^2(t)$

The equations for $\frac{d\underline{z}}{dt}$, i.e. Equation (19), take the following form:

$$\frac{dz_1}{dt} = -\ z_1BM1 - z_2G_a\ EAT1 - (q/V_2)\ z_3 \tag{86}$$

$$\frac{dz_2}{dt} = -\ z_2BM2 - z_4(q/V_2) \tag{87}$$

$$\frac{dz_3}{dt} = -\ q(C_2-C_3) - z_3BM3 - G_a\ EAT2\ z_4 \tag{88}$$

$$\frac{dz_4}{dt} = C_3q - BM4\ z_4 - z_5\ q \tag{89}$$

$$\frac{dz_5}{dt} = 2C_I\ (I_M - I(t)) \tag{90}$$

$$\frac{dz_6}{dt} = -\ C_1C_q + C_a\ A^2(t)\ 2K(t) + z_5C_q - z_6BM5 \tag{91}$$

Let

$$\underline{R}'\underline{TR} = \begin{bmatrix} RTR1 & RTR2 & RTR3 & RTR4 & RTR5 & RTR6 \\ RTR2 & RTR7 & RTR8 & RTR9 & RTR10 & RTR11 \\ RTR3 & RTR8 & RTR12 & RTR13 & RTR14 & RTR15 \\ RTR4 & RTR9 & RTR13 & RTR16 & RTR17 & RTR18 \\ RTR5 & RTR10 & RTR14 & RTR17 & TRT19 & RTR18 \\ RTR6 & RTR11 & RTR15 & RTR18 & RTR20 & RTR21 \end{bmatrix}$$

Equation (28) now becomes

$$\frac{dP_{11}}{dt} = - 2P_{11}\ BM1 - 2P_{21}\ Ga\ EAT1 - 2P_{31}\ (q/V_2) + RTR1 \tag{92}$$

$$\frac{dP_{21}}{dt} = - P_{21}\ (BM1 + BM2) - P_{22}\ Ga\ EAT1 - (P_{32} + P_{41})(q/V_2) + RTR2 \tag{93}$$

$$\frac{dP_{31}}{dt} = - P_{31}\ (BM1 + BM3) - P_{41}\ Ga\ EAT2 - P_{32}\ Ga \cdot EAT1 - P_{33}\ (q/V_2) + RTR3 \tag{94}$$

$$\frac{dP_{41}}{dt} = - P_{41}\ (BM1 + BM4) - P_{42}\ Ga\ EAT1 - P_{43}\ (q/V_2) - P_{51}\ q + RTR4 \tag{95}$$

$$\frac{dP_{51}}{dt} = - P_{51}\ BM1 - P_{52}\ Ga\ EAT1 - P_{53}\ (q/V_2) + RTR5 \tag{96}$$

$$\frac{dP_{61}}{dt} = P_{51}\ C_q - P_{61}\ (BM1 + BM5) - P_{62}\ Ga\ EAT1 - P_{63}\ (q/V_2) + RTR6 \tag{97}$$

$$\frac{dP_{22}}{dt} = - 2P_{22} BM2 - 2P_{42}\ (q/V_2) + RTR7 \tag{98}$$

$$\frac{dP_{32}}{dt} = - P_{32}(BM2 + BM3) - P_{42}\ Ga\ EAT2 - P_{43}\ (q/V_2) + RTR8 \tag{99}$$

$$\frac{dP_{42}}{dt} = - P_{42} (BM2 + BM4) - P_{44}(q/V_2) - P_{52} q + RTR9 \qquad (100)$$

$$\frac{dP_{52}}{dt} = - P_{52} BM2 - P_{54} (q/V_2) + RTR10 \qquad (101)$$

$$\frac{dP_{62}}{dt} = P_{52} CQ - P_{62} (BM2 + BM5) - P_{64} (q/V_2) + RTR11 \qquad (102)$$

$$\frac{dP_{33}}{dt} = - 2P_{33} BM3 - 2P_{43} Ga\ EAT2 + RTR12 \qquad (103)$$

$$\frac{dP_{43}}{dt} = - P_{43} (BM3 + BM4) - P_{44} Ga\ EAT2 - P_{53} \cdot q + RTR13 \qquad (104)$$

$$\frac{dP_{53}}{dt} = - P_{53} BM3 - P_{54} Ga\ EAT2 + RTR14 \qquad (105)$$

$$\frac{dP_{63}}{dt} = P_{53} Cq - P_{63} (BM3 + BM5) - P_{64} Ga\ EAT2 + RTR15 \qquad (106)$$

$$\frac{dP_{44}}{dt} = - 2P_{44} BM4 - 2P_{54} q + RTR16 \qquad (107)$$

$$\frac{dP_{54}}{dt} = - P_{54} BM4 - P_{55} Q + RTR17 \qquad (108)$$

$$\frac{dP_{64}}{dt} = P_{54} Cq - P_{64} (BM4 + BM5) - P_{65} q + RTR18 \qquad (109)$$

$$\frac{dP_{55}}{dt} = 2C_I + RTR19 \qquad (110)$$

$$\frac{dP_{65}}{dt} = P_{55} Cq - P_{65} BM5 + RTR20 \qquad (111)$$

$$\frac{dP_{66}}{dt} = 2z_6 \left[C + \frac{A(t)}{4}\right] + 2Ca (A^2(t)) + 2P_{65} Cq - 2P_{66} BM5 + RTR21 \qquad (112)$$

Equation (48) is given by

$$\frac{d\underline{Q}}{dt} = \underline{R}'\ \underline{T}^{-1}\ \underline{S} + R'\ T^{-1}\ (\frac{\partial f'}{\partial \underline{\theta}})\ \underline{Q} - (\frac{\partial f'}{\partial \underline{x}})\ \underline{Q}$$

where $\underline{Q}$ is six dimensional. Here all the terms were obtained by the matrix multiplication. The new control is calculated as given by Equation (50), i.e.

$$\begin{pmatrix} T_1 \\ T_2 \\ A(t) \end{pmatrix}^{(j+1)} = \begin{pmatrix} T_1 \\ T_2 \\ A(t) \end{pmatrix}^{(j)} - \left[\varepsilon\ \underline{T}^{-1}(\underline{S} + \frac{\partial f'}{\partial \underline{\theta}}\ \underline{Q})\right]^{(j)} - (\underline{T}^{-1}\ \underline{R})^{j} \cdot (\underline{x}^{(j+1} - \underline{x}^{(j)})$$

Table 18

Numerical Values of the Constants

Set # 1

$q = 60$	$v_1 = 12.0$	$v_2 = 12.0$
$C_q = 1.0$	$C = 1.0$	$N = 100.0$
$Im = 20.0$	$T_{1m} = 340.$	$C_1 = 5.0$
$C_2 = 0.0$	$C_3 = 0.0$	$C_I = 1.0$
$C_A = 0.0002$	$C_T = 0.0005$	$R = 2.000$
$E_A = 18000.0$	$E_B = 30000.0$	$x_I = 0.530$
$y_I = 0.430$	$Ga = 0.535 \times 10^{11}$	$Ga = 0.461 \times 10^{18}$

Set # 2

$q = 60.$	$v_1 = 12.0$	$v_2 = 12.0$
$C_q = 1.0$	$C = 1.0$	$N = 100$
$Im = 10.$	$T_{1m} = 340.$	$C_1 = 5.0$
$C_2 = 0.$	$C_3 = 0$	$C_I = 1.0$
$C_A = 0.01$	$C_T = 0.001$	$R = 2.0$
$E_A = 18000$	$EB = 30000$	$x_I = 0.53$
$y_I = 0.43$	$Ga = 0.535 \times 10^{11}$	$Ga = 0.461 \times 10^{18}$

Set # 3

Same as Set # 2

except $Im = 20.$

$C_T = 0.0005$

and $K(t_0) = 1.0$

Discussion

This particular problem reveals how the theoretical attractiveness of the second variation method is more than offset by both the complexity and by the number of equations to be integrated.

In this problem (6+1) or seven equations are to be integrated in the forward direction and (6+6(7)/2+6) or 33 equations in the backward direction. In addition, the calculations of $\underline{R}$, $\underline{S}$, and $\underline{T}$ are in terms of matrix multiplications and $\underline{T}^{-1}$ has to be calculated at each step of the integration in Equations (28) and (48).

This program was tried with three different sets of numerical values which are shown in Table 18. These values were taken from the solution of the same problem first by variation and quasilinearization respectively.

This problem was found to be unstable as far as its solution by the second variation is concerned. With all the various values tried, the program could make a complete iteration. However, it fails in the backward integration of the second variational equations because of exponential overflow.

4. CONCLUSION

The second variation method has been shown to be a fairly useful tool for attacking the complex practical optimization problems involving a fairly large number of variables. The convergence is very fast, provided the initial or starting guess is sufficiently close to the optimal trajectory. This, however, becomes more and more difficult when more than one control variable are involved. In that case, the number of combinations that could be used as the starting trajectory is quite large and makes the initial guess a difficult task. This can be overcome by using the first variation method for the first few iterations and then switching to the second variation method. This combination provides rapid convergence to the optimum from almost any realistic starting trajectory. The theoretical attractiveness of this method is removed by its disadvantages like the guess of the initial trajectory for the state variables in addition to that of control variables. Also the number of equations and their complexity make the use of this technique tedious.

The first variation method, of which the second variation method is a natural evolution, should be used in combination with the second variation. The first variation method, unlike the second variation, will approach optimum from almost any realistic starting trajectory. The results of the first variation method could then be used as the starting trajectories for the second variation. In this way, the convergence problem of the second variation can be partly overcome. This combination provides a rapid convergence from almost any realistic starting trajectory for most engineering problems. While evaluating the merits and demerits of this technique, it should be borne in mind that no single optimization technique is suitable for all classes of problems that will be encountered. Each technique will be most efficient only for a particular type or types of problems. It is left to the decision of the engineer to select any one or a combination of these techniques for the problem he is facing.

5. ACKNOWLEDGMENT

The author is deeply indebted to Dr. E.S. Lee, major professor, for his patient guidance, valuable comments, constructive critism and deep interest in preparing this thesis.

6. BIBLIOGRAPHY

1. Aris, R. and Amundson N. R., Chem. Engg. Sci., 7, 121, 132, 148 (1958).
2. Bellman R. and Dreyfus S. E., Applied Dynamic Programming, Princeton University Press, Princeton, New Jersey (1962).
3. Bliss, G., Lectures on the Calculus of Variations, University of Chicago Press, Chicago, Illinois (1946).
4. Bryson A. E., Breakwell J. V. and Speyer J. L., Journal SIAM Control 1, 193 (1963).
5. Bryson A. E. and Denham W. F., "A Steepest Ascent Method for Solving Optimum Programming Problems", J. Appl. Mech., 29, 247 (1962).
6. Dreyfus, S., "Variational Problems with State Variable Inequality Constraints", P-2605, The Rand Corp., Santa Monica, Calif. (July, 1962).
7. Fan . T., "The Continuous Maximum Principle, John Wiley & Sons, New York (1966).
8. Ho Y - and Breakwell J. V., Int., J. Eng. Sci. 2, 265 (1965).
9. Jaswinski, A. H., AIAA Journal, 2, 1371 (1964); 3, 925 (1965).
10. Kelley, H. J., "Gradient Theory of Optimal Flight Paths", ARS Journal, 30, 947 (1960).
11. Kelley, H. J.; Kopp, R. E. and Moyer, N. G., Preprint 63-145 at AIAA Astrodynamics Conference (Aug. 1963).
12. Lapidus L. and Luus R., "Optimal Control of Chemical Engineering Systems", Blaisdell Publishing Co., New York (1966).
13. Lapidus L. and Luus R., "Control of Non-Linear Systems I. Iteration on the Performance INdex", Submitted for Presentation at Columbus A.I.Ch.E. Meeting (1966).

14. Lee E. S., "Optimum Design and Operation of Chemical Processes", Ind. Eng. Chem. 55, 30 (Aug. 1963).

15. Lee E. S., A.I.Ch.E. Journal, 10, 309 (1964).

16. Lee E. S. "Optimization by a Gradient Technique", I&Ec. Fundamentals, 3, 373 (1964).

17. Lee E. S. "Optimization of Complex Chemical Plants by a Gradient Technique" Presented at the A.I.Ch.E. National Meeting", Feb. 18-21, 1968.

18. Lee E. S., "Optimal Production Planning of Non-Linear, Multiproduct Systems", Presented at the Joint Meeting of ORSA and TIMS, May 1-3, 1968.

19. Lee E. S. "Quasilinearization and Invariant Imbedding", Academic Press (1968).

20. Lee E. S. & Shaikh M. A., Optimal Production Planning by a Gradient Technique I First Variation.

21. Leitmann, G. Ed., "Optimization Techniques with Application to Aerospace Systems" Academic Press, New York (1962).

22. McReynolds, S. R. and Brayson, A. E., "A Successive Sweep Method for Solving Optimal Programming Problems", Presented at 1965 JACC (June 1965).

23. Merriam, C. W., J. Siam Control Ser A, 2, 1, (1964).

24. Merriam, C. W., "Information and Control" 8, 215 (1965).

25. Merriam, C. W., "Optimization Theory and the Design of Feedback Control Systems", McGraw Hill Book Co., New York (1964).

26. Mitter S. K., Automatica, 3, 135 (1966).

27. Teichroew, D. "An Introduction to Management Science" - Deterministic Models, John Wiley, New York, (1966).

28. Zangwill, W. I., "A Deterministic Multiproduct Multifacility Production and Inventory Model", O.R., 14, 486 (1966).

7. APPENDIX

7.1 Computer Program for the Inventory Model

7.2 Computer Program for the Inventory and Advertising Model

7.3 Computer Program for the Chemical Manufacturing Problem with Advertisement

```
$JOB              GRADIENT,RUN=CHECK,TIME=9,PAGES=200,LINES=50
C   SECOND VARIATIONAL GRADIENT TECHNIQUE---RANGNEKAR
          DIMENSION P1(150),P2(150),P3(150),P4(150),DX1(150),X1(150),Q1(150)
          DIMENSION Q2(150),Q3(150),Q4(150),DX2(150),X2(150)
          DIMENSION Z(150),M(150),T(150)
          DIMENSION RST(150),RT(150),R(150)
          DIMENSION L(150),TL(150),X(150),P(150),TX(150),Q(150),S(150)
      100 FORMAT (1H-,'NO.            X1                        X2              Z
        1          P                Q                  THETA               TX')
      101 FORMAT (1H-,(1H ,I3,2X,E15.8,2X,E15.8,2X,E15.8,2X,E15.8,2X,E15.8,2
        1X,E15.8,1X,E15.8))
      102 FORMAT (8F9.7)
      103 FORMAT (1H ,9(F7.3))
      104 FORMAT (1H-,'***************************************     ITERATION
        1 NO.',I4,'    ****************************************')
          ITMAX=58
          EP=.1
          READ 103,PA,CP,XM,CI,D,B,A,X1(1),X2(1)
          PRINT 103,PA,CP,XM,CI,D,B,A,X1(1),X2(1)
          CP=.001
          DO 12 I=1,101
          T(I)=9.
       12 X(I)=5.
          DO 1 N=1,ITMAX
          L(N)=N
        2 DO 6 I=1,101
          M(I)=I
C***************************************     FORWARD INTEGRATION OF X1
          P1(I)=D*(T(I)-A-B*D*(I-1))
          P2(I)=D*(T(I)-A-B*(D*(I-1)+.5*D))
          P3(I)=P2(I)
          P4(I)=D*(T(I)-A-B*(D*(I-1)+D))
          DX1(I)=(1./6.)*(P1(I)+2.*P2(I)+2.*P3(I)+P4(I))
          X1(I+1)=X1(I)+DX1(I)
C***************************************     FORWARD INTEGRATION OF X2
          Q1(I)=D*(CI*(XM-X1(I))*(XM-X1(I))+CP*EXP((PA-T(I))*(PA-T(I))))
          Q2(I)=D*(CI*(XM-X1(I)+.5*P1(I))**2+CP*EXP((PA-T(I))**2))
          Q3(I)=D*(CI*(XM-X1(I)+.5*P2(I))**2+CP*EXP((PA-T(I))**2))
          Q4(I)=D*(CI*(XM-X1(I)+P3(I))**2+CP*EXP((PA-T(I))**2))
          DX2(I)=(1./6.)*(Q1(I)+2.*Q2(I)+2.*Q3(I)+Q4(I))
          X2(I+1)=X2(I)+DX2(I)
        4 TX(I)=2.*CP*(EXP((PA-T(I))**2))*(1.+2.*((PA-T(I))**2))
        6 S(I)=2.*CP*(PA-T(I))*(EXP((PA-T(I))**2))
          DO 7 I=1,100
          K=101
          Z(101)=0.
          P(101)=0.
          Q(101)=0.
C***************************************      BACKWARD INTEGRATION OF Z
          ZM1=-D*2.*CI*(XM-X1(K+1-I))
```

```
      ZM2=ZM1
      ZM3=ZM1
      ZM4=ZM1
    5 Z(K-I)=Z(K+1-I)+(1./6.)*(ZM1+2.*ZM2+2.*ZM3+ZM4)
C*************************************BACKWARD INTEGRATION OF DP/DT
C     ****************************            DP/DT=-2CI+(P)SQUARE(TX)
      PM1=-D*(-2.*CI+TX(K+1-I)*(P(K+1-I)**2))
      PM2=-D*(-2.*CI+TX(K+1-I)*((P(K+1-I)+PM1/2.)**2))
      PM3=-D*(-2.*CI+TX(K+1-I)*((P(K+1-I)+PM2/2.)**2))
      PM4=-D*(-2.*CI+TX(K+1-I)*((P(K+1-I)+PM3)**2))
      P(K-I)=P(K+1-I)+(1./6.)*(PM1+2.*PM2+2.*PM3+PM4)
C*************************************BACKWARD INTEGRATION OF DQ/DT
C*************************************       DQ/DT=(P(Z+Q-S))/TX
      QM1=-D*((P(K+1-I)*(Z(K+1-I)+Q(K+1-I)-S(K+1-I)))/TX(K+1-I))
      QM2=-D*(((P(K+1-I)+.5*PM1)*(Z(K+1-I)+.5*ZM1+Q(K+1-I)+.5*QM1-S(K+1-
     1I)))/TX(K+1-I))
      QM3=-D*(((P(K+1-I)+.5*PM2)*(Z(K+1-I)+.5*ZM2+Q(K+1-I)+.5*QM2-S(K+1-
     1I)))/TX(K+1-I))
      QM4=-D*(((P(K+1-I)+PM3)*(Z(K+1-I)+ZM3+Q(K+1-I)+QM3-S(K+1-I)))/TX(K
     1+1-I))
    7 Q(K-I)=Q(K+1-I)+(1./6.)*(QM1+2.*QM2+2.*QM3+QM4)
      DO 8 I=1,101
    8 T1(I)=T(I)-EP*((Z(I)-S(I)+Q(I))/TX(I))-(P(I)/TX(I))*(X1(I)-X(I))
      PRINT 104,L(N)
      PRINT 100
      PRINT 101,(M(I),X1(I),X2(I),Z(I),P(I),Q(I),T1(I),TX(I),I=1,101)
      DO 9 I=1,101
      T(I)=T1(I)
    9 X(I)=X1(I)
    1 CONTINUE
      STOP
      END
```

```
$JCB              RANGNEKAR,RUN=CHECK,TIME=30,PAGES=400,LINES=50
          EXTERNAL FCT,OUTP
          DIMENSION PRMT(10),DERY(10),AUX(8,12),YO(2,101),Y1(9,101),Y(10),AT
         1(101),ATNEW(101)
          COMMON Y1,AT,ATNEW,YO,A,B,C,AN,F,CI,PI,CA,K,NK,EP,R1,R2,S,T,J,N
C --------- THIS PROBLEM HAS 2 STATE VARIABLES AND 1 CONTROL VARIABLE
C -- THE STATE VARIABLES ARE THE INVENTORY AND THE SALES
C -- THE CONTROL VARIABLE IS THE ADVERTISEMNT MADE
C
C ----------------------- READ THE VARIOUS CONSTANTS ----------------
      100 FORMAT (9F8.3)
          READ 100,A,B,C,AN,F,CI,PI,CA,EP
      101 FORMAT (2I4)
          READ 101,NK,ITMAX
      102 FORMAT (' FOLLOWING VALUES OF THE VARIOUS CONSTANTS ARE READ IN')
      103 FORMAT ('0A=',F8.3,'        B=',F8.3,'          C=',F8.3,'      AN
         1=',F8.3)
      104 FORMAT ('0F=',F8.3,'        CI=',F8.3,'         PI=',F8.3,'      CA
         1=',F8.3)
      105 FORMAT ('0EP=',F8.3,'       NK=',I4,'           ITMAX=',I4)
          PRINT 102
          PRINT 103,A,B,C,AN
          PRINT 104,F,CI,PI,CA
          PRINT 105,EP,NK,ITMAX
C  VALUES OF THE STATE AND CONTROL VARIABLES AT 101 GRID POINTS --------
      200 FORMAT ('0THE FOLLOWING VALUES OF STATE AND CONTROL VARIABLES AT 1
         101 GRID POINTS ARE READ IN')
          PRINT 200
      201 FORMAT (12F6.3)
          READ 201,(YO(1,NM),NM=1,NK)
          READ 201,(YO(2,NM),NM=1,NK)
          READ 201,(AT(NM),NM=1,NK)
      203 FORMAT ('0NO.          INVENTORY                SALES            ADVT.')
          PRINT 203
      202 FORMAT (1H-,(1H ,I3,4X,3(F6.3,5X))/)
          PRINT 202,(NM,YO(1,NM),YO(2,NM),AT(NM),NM=1,NK)
C ----------------------- MAIN DO LOOP FOR ITERATIONS  -------------
          DO 2 IJ=1,ITMAX
        3 FORMAT ('0*********************************************  ITERATION
         1 NO. ',I4,' ************************************ ')
          PRINT 3,IJ
C  VARIOUS PARAMETERS FOR FORWARD INTEGRATION ------------------
          PRMT(1)=0.
          PRMT(2)=1.
          PRMT(3)=.01
          PRMT(4)=.001
          NDIM=3
          DERY(1)=NDIM
          DERY(1)=1./DERY(1)
          DO 4 I=2,NDIM
```

```
      4 DERY(I)=DERY(1)
        Y(1)=.2
        Y(2)=.2
        Y(3)=0.
        K=1
C   ------------ CALL RKGS FOR THE FORWARD INTEGRATION -------------
        CALL RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C   VARIOUS PARAMETERS FOR THE BACKWARD INTEGRATION
        PRMT(1)=1.
        PRMT(2)=0.
        PRMT(3)=-.01
        PRMT(4)=.01
        NDIM=7
        DERY(1)=NDIM
        DERY(1)=1./DERY(1)
        DO 5 I=2,NDIM
      5 DERY(I)=DERY(1)
        DO 6 I=1,7
      6 Y(I)=0.
        K=2
C   ----------- CALL RKGS FOR BACKWARD INTEGRATION -----------------
        CALL RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
        DO 7 L=1,NK
        YO(1,L)=Y1(1,L)
        YO(2,L)=Y1(2,L)
      7 AT(L)=ATNEW(L)
      2 CONTINUE
        STOP
        END
C
```

```
      SUBROUTINE FCT(X,Y,DERY)
      DIMENSION PRMT(10),DERY(10),AUX(8,12),YO(2,101),Y1(9,101),Y(10),AT
     1(101),ATNEW(101)
      COMMON Y1,AT,ATNEW,YO,A,B,C,AN,F,CI,PI,CA,K,NK,EP,R1,R2,S,T,J,N
C  DEPENDING ON THE VALUE OF K,EITHER THE FIRST PART OR THE SECOND PART
C  OF THIS SUBROUTINE IS USED FOR THE FORWARD AND THE BACKWARD
C  INTEGRATION RESPECTIVELY.
      IF (K.EQ.2) GO TO 10
C  -- THIS PART OF SUBROUTINE IS FOR FORWARD INTEGRATION ONLY  ---------
      IF (X.NE.0) GO TO 11
      J=1
C Y(1),Y(2) DENOTE X AND Q IN THE PROBLEM  ----------------------------
C  INDEPENDENT VARIABLE T IN THE ORIGINAL EQNS. IS DENOTED BY X IN THE
C   PROGRAM
   11 DERY(1)=A+B*X-Y(2)
      DERY(2)=Y(2)*(C+AT(J))*(1.-Y(2)/AN)
      DERY(3)=Y(2)*F-CI*((PI-Y(1))**2)-CA*Y(2)*(AT(J)**2)
      RETURN
C  ------- FIRST PART FOR FORWARD INTEGRATION ENDS  -------------------
C
C  ----- SECOND PART FOR BACKWARD INTEGRATION   ------------------------
   10 IF(X.NE.1) GO TO 12
      N=NK
   12 R1=Y(4)*Y1(2,N)*(1.-Y1(2,N)/AN)
      R2=-2.*CA*AT(N)+Y(5)*Y1(2,N)*(1.-Y1(2,N)/AN)+Y(2)*(1.-2.*Y1(2,N)/A
     1N)
      S=-2.*CA*Y1(2,N)*AT(N)+Y(2)*Y1(2,N)*(1.-Y1(2,N)/AN)
      T=-2.*CA*Y1(2,N)
      DERY(1)=-2.*CI*(PI-Y1(1,N))
      DERY(2)=-F+CA*(AT(N)**2)+Y(1)-Y(2)*(C+AT(N))*(1.-2.*Y1(2,N)/AN)
C ---------  BACKWARD INTEGRATION OF DP/DT ---------  3 EQUATIONS  -----
      DERY(3)=2.*CI+(R1**2)*T
      DERY(4)=Y(3)-Y(4)*(C+AT(N))*(1.-2.*Y1(2,N)/AN)+R1*R2*T
      DERY(5)=(2.*Y(2)/AN)*(C+AT(N))+2.*Y(4)-2.*Y(5)*(C+AT(N))*(1.-2.*Y1
     1(2,N)/AN)+(R2**2)*T
C  ---- BACKWARD INTEGRATION OF DQF/DT  --------------------------------
C  ----- THE SCALAR FUNCTION Q IN THE ORIGINAL DERIVATION IS DENOTED BY
C  QF IN THIS PROBLEM AND IS DENOTED BY Y(6) AND Y(7) RESP., IN
C   THIS PROGRAM
      DERY(6)=(R1*S)/T+(R1/T)*Y(7)*Y1(2,N)*(1.-Y1(2,N)/AN)+Y(7)
      DERY(7)=(R2*S)/T+(R2/T)*Y(7)*Y1(2,N)*(1.-Y1(2,N)/AN)-Y(7)*(C+AT(N)
     1)*(1.-2.*Y1(2,N)/AN)
      RETURN
C   ------ SECOND PART ENDS
      END
C
```

```
      SUBROUTINE OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      DIMENSION PRMT(10),DERY(10),AUX(8,12),YO(2,101),Y1(9,101),Y(10),AT
     1(101),ATNEW(101)
      COMMON Y1,AT,ATNEW,YO,A,B,C,AN,F,CI,PI,CA,K,NK,EP,R1,R2,S,T,J,N
      IF(K.EQ.2) GO TO 20
C ----- THIS PART OF THE SUBROUTINE IS FOR THE FORWARD INTEGRATION ONLY
      IF (X.NE.0) GO TO 21
      J=0
   23 FORMAT ('-  NO.   GRID.PT.       INVENTORY              SALES
     1     PROFIT                 ADVT.')
      PRINT 23
   21 J=J+1
C ---- STORING THE VALUES OF STATE VARIABLES AT EACH GRID POINT,TO BE
C USED IN THE SE CND PART OF THIS SUBROUTINE FOR THE CALCULATION OF
C THE NEW VALUES OF THE CONTROL VARIABLE AT 101 GRID POINTS
      DO 22 M=1,2
   22 Y1(M,J)=Y(M)
      ABC=-Y(3)
   24 FORMAT (1H ,I4,4X,F6.2,5X,4(E12.4,7X))
      PRINT 24,J,X,Y(1),Y(2),ABC,AT(J)
      IF (J.EQ.101) PRMT(5)=1.
      RETURN
C --- FIRST PART FOR FORWARD INTEGRATION ENDS ------------------------
C
C --------------- SECOND PART FOR BACKWARD INTEGRATION ONLY   ---------
   20 IF (X.NE.1) GO TO 25
  306 FORMAT ('-NO.  GRID PT.     Z1            Z2            P11           P1
     12           P22              QF1          QF2           S             T'
     2)
      PRINT 306
      N=NK+1
   25 N=N-1
      R1=Y(4)*Y1(2,N)*(1.-Y1(2,N)/AN)
      R2=-2.*CA*AT(N)+Y(5)*Y1(2,N)*(1.-Y1(2,N)/AN)+Y(2)*(1.-2.*Y1(2,N)/A
     1N)
      S=-2.*CA*Y1(2,N)*AT(N)+Y(2)*Y1(2,N)*(1.-Y1(2,N)/AN)
      T=-2.*CA*Y1(2,N)
C -------- BACKWARD INTEGRATION OF DZ/DT  ------  2 EQUATIONS  --------
C --------- CALCULATION OF THE NEW VALUE OF THE CONTROL VARIABLE AT
C     N TH GRID POINT  ------
   30 XX1=Y1(1,N)-YO(1,N)
      XX2=Y1(2,N)-YO(2,N)
      ATNEW(N)=AT(N)-(EP/T)*(S+Y(7)*Y1(2,N)*(1.-Y1(2,N)/AN))-(R1*XX1+R2*
     1XX2)/T
  308 FORMAT (1H ,I4,1X,F5.2,1X,9(E12.4,1X))
      PRINT 308,N,X,(Y(IM),IM=1,7),S,T
      IF (N.EQ.1) PRMT(5)=1.
  310 RETURN
C ---- SECOND PART FOR THE BACKWARD INTEGRATION ENDS   --------------
      END
```

```
      SUBROUTINE RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
C
      DIMENSION Y(1),DERY(1),AUX(8,1),A(4),B(4),C(4),PRMT(1)
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.
      CALL FCT(X,Y,DERY)
C
C
C     PREPARATIONS FOR RUNGE-KUTTA METHOD
    2 A(1)=.5
      A(2)=.2928932
      A(3)=1.707107
      A(4)=.1666667
      B(1)=2.
      B(2)=1.
      B(3)=1.
      B(4)=2.
      C(1)=.5
      C(2)=.2928932
      C(3)=1.707107
      C(4)=.5
C
C     PREPARATIONS OF FIRST RUNGE-KUTTA STEP
      DO 3 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
      AUX(3,I)=0.
    3 AUX(6,I)=0.
C
C     RECORDING OF INITIAL VALUES OF THIS STEP
    7 CALL OUTP(X,Y,DERY,IREC,NDIM,PRMT)
      IF(PRMT(5))40,8,40
C
C
C     START OF INNERMOST RUNGE-KUTTA LOOP
    8 J=1
   10 AJ=A(J)
      BJ=B(J)
      CJ=C(J)
      DO 11 I=1,NDIM
      R1=H*DERY(I)
      R2=AJ*(R1-BJ*AUX(6,I))
      Y(I)=Y(I)+R2
      R2=R2+R2+R2
   11 AUX(6,I)=AUX(6,I)+R2-CJ*R1
      IF(J-4)12,15,15
   12 J=J+1
      IF(J-3)13,14,13
```

```
   13 X=X+.5*H
   14 CALL FCT(X,Y,DERY)
      GOTO 10
C     END CF INNERMOST RUNGE-KUTTA LOOP
C
C
   15 DO 29 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
   29 AUX(6,I)=AUX(3,I)
      CALL CUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))40,30,40
   30 DO 31 I=1,NDIM
      Y(I)=AUX(1,I)
   31 DERY(I)=AUX(2,I)
      GO TC 8
   40 RETURN
      END
```

```
      READ 60,(AT(I),I=1,NK)
      DO 515 JK=1,101
  515 YO(6,JK)=YO(6,JK)/100.
   63 FORMAT(' FOLLOWING VALUES OF THE 6 STATE VARIABLES AND 3 CONTROL
     1 VARIABLES ARE READ IN')
      PRINT 63
   61 FORMAT (1H ,(1H ,I3,1X,9(F10.5,1X))/)
      PRINT 61,(I,YO(1,I),YO(2,I),YO(3,I),YO(4,I),YO(5,I),YO(6,I),T1(I)
     1T2(I),AT(I),I=1,NK)
C ---------------- MAIN DO LOOP FOR ITERATIONS ------------------
      DO 100 IJ=1,ITMAX
  200 FORMAT (' ***********************************     ITERATION  NO. =
     1 ',I4,'  ***********************************')
      PRINT 200,IJ
      PRMT(1)=0.0
      PRMT(2)=1.
      PRMT(3)=.01
      PRMT(4)=.1
      NDIM=7
      DERY(1)=NDIM
      DERY(1)=1.E0/DERY(1)
      DO 1 I=2,NDIM
    1 DERY(I)=DERY(1)
      Y(1)=.53
      Y(2)=.43
      Y(3)=.53
      Y(4)=.43
      Y(5)=8.
      Y(6)=.01
      Y(7)=0
      KSL=1
   41 CALL RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
      PRMT(1)=1.
      PRMT(2)=0.
      PRMT(3)=-.01
      PRMT(4)=10.
      NDIM=33
      DERY(1)=NDIM
      DERY(1)=1.E0/DERY(1)
      DO 3 I=2,NDIM
    3 DERY(I)=DERY(1)
      DO 4 I=1,33
    4 Y(I)=0.
      KSL=2
   46 CALL RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
      DO 120 L=1,NK
      YO(1,L)=Y1(1,L)
      YO(2,L)=Y1(2,L)
      YO(3,L)=Y1(3,L)
      YO(4,L)=Y1(4,L)
```

```
      Y0(5,L)=Y1(5,L)
      Y0(6,L)=Y1(6,L)
      T1(L)=T1NEW(L)
      T2(L)=T2NEW(L)
  120 AT(L)=ATNEW(L)
  100 CONTINUE
      STOP
      END
```

```
      SUBROUTINE FCT (X,Y,DERY)
      DIMENSION PRMT(10),Y(42),DERY(42),AUX(8,43),Y1(40,202),T1NEW(202)
     1T2NEW(202),ATNEW(202),T1(202),T2(202),AT(202),Y0(6,202),A(10),L(1
     2),M(10)
      COMMON Y1,T1NEW,T2NEW,ATNEW,T1,T2,AT,NK,XX1,XX2,XX3,XX4,XX5,XX6,Y
     1,FT1,FT2,FT3,FT4,FT5,S1,S2,S3,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,
     212,R13,R14,R15,R16,R17,R18,A,J,N,EP,KSL,R,DIS,V1,V2,CQ,C,AN,AIM,T
     3M,C1,C2,C3,CI,CA,CT,EA,EB,XI,YI,GA,GB,RTR1,RTR2,RTR3,RTR4,RTR5,RT
     46,RTR7,RTR8,RTR9,RTR10,RTR11,RTR12,RTR13,RTR14,RTR15,RTR16,RTR17
      COMMON RTR18,RTR19,RTR20,RTR21,RTS1,RTS2,RTS3,RTS4,RTS5,RTS6,RTFQ
     1,RTFQ2,RTFQ3,RTFQ4,RTFQ5,RTFQ6,FQ1,FQ2,FQ3,FQ4,FQ5,FQ6,EAT1,EAT2,
     1BT1,EBT2,BM1,BM2,BM3,BM4,BM5,G1,G2,G3,G4
      IF (KSL.EQ.2) GO TO 25
      IF (X.NE.0) GO TO 42
      J=1
   42 DERY(1)=(DIS/V1)*(XI-Y(1))-GA*EXP(-EA/(R*T1(J)))*Y(1)
      DERY(2)=(DIS/V1)*(YI-Y(2))-GB*EXP(-EB/(R*T1(J)))*Y(2)+GA*EXP(-EA/(
     1R*T1(J)))*Y(1)
      DERY(3)=(DIS/V2)*(Y(1)-Y(3))-GA*EXP(-EA/(R*T2(J)))*Y(3)
      DERY(4)=(DIS/V2)*(Y(2)-Y(4))-GB*EXP(-EB/(R*T2(J)))*Y(4)+GA*EXP(-EA
     1/(R*T2(J)))*Y(3)
      DERY(5)=DIS*Y(4)-CQ*Y(6)
      DERY(6)=(C+AT(J))*(Y(6)-((Y(6)**2)/AN))
      DERY(7)=CQ*C1*Y(6)+C2*DIS*Y(3)+C3*DIS*(1-Y(3)-Y(4))-CI*((AIM-Y(5))
     1**2)-CA*(AT(J)**2)*(Y(6)**2)-CT*((T1M-T1(J))**2+(T1(J)-T2(J))**2)
      RETURN
   25 IF (X.NE.1) GO TO 47
      N=NK
   47 CALL CALCL(X,Y,DERY)
C ---------------------- INTEGRATION OF DZ/DT ---------------------------
      DERY(1)=-BM1*Y(1)-Y(2)*GA*EAT1-(DIS/V2)*Y(3)
      DERY(2)=-BM2*Y(2)-Y(4)*(DIS/V2)
      DERY(3)=DIS*(C3-C2)-BM3*Y(3)-Y(4)*GA*EAT2
      DERY(4)=C3*DIS-Y(4)*BM4-DIS*Y(5)
      DERY(5)=2.*CI*(Y1(5,N)-AIM)
      DERY(6)=-C1*CQ+2.*CA*(AT(N)**2)*Y1(6,N)+Y(6)*CQ*(C+AT(N))*(1.-2.*Y
     11(6,N)/AN)
C ---------------  INTEGRATION OF  DP/DT   ----------------------------
      DERY(7)=-2.*Y(7)*BM1-2.*Y(8)*GA*EAT1-2.*Y(9)*(DIS/V2)+RTR1
      DERY(8)=-Y(8)*(BM1+BM2)-Y(13)*GA*EAT1-(Y(14)+Y(10))*(DIS/V2)+RTR2
      DERY(9)=-Y(9)*(BM1+BM3)-Y(10)*GA*EAT2-Y(14)*GA*EAT1-Y(18)*(DIS/V2)
     1+RTR3
      DERY(10)=-Y(10)*(BM1+BM4)-Y(15)*GA*EAT1-Y(19)*(DIS/V2)-Y(11)*DIS+R
     1TR4
      DERY(11)=-Y(11)*BM1-Y(16)*GA*EAT1-Y(20)*(DIS/V2)+RTR5
      DERY(12)=Y(11)*CQ-Y(12)*(BM1+BM5)-Y(17)*GA*EAT1-Y(21)*(DIS/V2)+RTR
     16
      DERY(13)=-2.*Y(13)*BM2-2.*Y(15)*(DIS/V2)+RTR7
      DERY(14)=-Y(14)*(BM2+BM3)-Y(15)*GA*EAT2-Y(19)*(DIS/V2)+RTR8
      DERY(15)=-Y(15)*(BM2+BM4)-Y(22)*(DIS/V2)-Y(16)*DIS+RTR9
```

```
      DERY(16)=-Y(16)*BM2-Y(23)*(DIS/V2)+RTR10
      DERY(17)=Y(16)*CQ-Y(17)*(BM2+BM5)-Y(24)*(DIS/V2)+RTR11
      DERY(18)=-2.*Y(18)*BM3-2.*Y(19)*GA*EAT2+RTR12
      DERY(19)=-Y(19)*(BM3+BM4)-Y(22)*GA*EAT2-Y(20)*DIS+RTR13
      DERY(20)=-Y(20)*BM3-Y(23)*GA*EAT2+RTR14
      DERY(21)=Y(20)*CQ-Y(21)*(BM3+BM5)-Y(24)*GA*EAT2+RTR15
      DERY(22)=-2.*Y(22)*BM4-2.*Y(23)*DIS+RTR16
      DERY(23)=-Y(23)*BM4-Y(25)*DIS+RTR17
      DERY(24)=Y(23)*CQ-Y(24)*(BM4+BM5)-Y(26)*DIS+RTR18
      DERY(25)=RTR19+2.*CI
      DERY(26)=Y(25)*CQ-Y(26)*BM5+RTR20
      DERY(27)=2.*Y(6)*(C+AT(N))/AN+2.*CA*(AT(N)**2)+2.*Y(26)*CQ-2.*Y(2
     1)*BM5+RTR21
C -------------------- INTEGRATION OF DQ/DT -----------------------------
      DERY(28)=RTS1+RTFQ1-FQ1
      DERY(29)=RTS2+RTFQ2-FQ2
      DERY(30)=RTS3+RTFQ3-FQ3
      DERY(31)=RTS4+RTFQ4-FQ4
      DERY(32)=RTS5+RTFQ5-FQ5
      DERY(33)=RTS6+RTFQ6-FQ6
      RETURN
      END
```

```
      SUBROUTINE OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      DIMENSION PRMT(10),Y(42),DERY(42),AUX(8,43),Y1(40,202),T1NEW(202)
     1T2NEW(202),ATNEW(202),T1(202),T2(202),AT(202),YO(6,202),A(10)
      COMMON Y1,T1NEW,T2NEW,ATNEW,T1,T2,AT,NK,XX1,XX2,XX3,XX4,XX5,XX6,Y
     1,FT1,FT2,FT3,FT4,FT5,S1,S2,S3,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,
     212,R13,R14,R15,R16,R17,R18,A,J,N,EP,KSL,R,DIS,V1,V2,CQ,C,AN,AIM,T
     3M,C1,C2,C3,CI,CA,CT,EA,EB,XI,YI,GA,GB,RTR1,RTR2,RTR3,RTR4,RTR5,RT
     46,RTR7,RTR8,RTR9,RTR10,RTR11,RTR12,RTR13,RTR14,RTR15,RTR16,RTR17
      COMMON RTR18,RTR19,RTR20,RTR21,RTS1,RTS2,RTS3,RTS4,RTS5,RTS6,RTFQ
     1,RTFQ2,RTFQ3,RTFQ4,RTFQ5,RTFQ6,FQ1,FQ2,FQ3,FQ4,FQ5,FQ6,EAT1,EAT2,
     1BT1,EBT2,BM1,BM2,BM3,BM4,BM5,G1,G2,G3,G4
      IF (KSL.EQ.2) GO TO 36
      IF (X.NE.0) GO TO 31
      J=0
   31 J=J+1
      DO 32 K=1,6
   32 Y1(K,J)=Y(K)
   15 FORMAT (1H ,I3,3X,F6.3,2X,7(E12.4,3X))
   16 PRINT 15,J,X,(Y(I),I=1,7)
      IF (J.EQ.NK) PRMT(5)=1.
   17 RETURN
   36 IF (X.NE.1) GO TO 33
      N=NK+1
   33 N=N-1
      CALL CALCL(X,Y,DERY)
   20 FORMAT (1H ,I3,1X,F5.3,1X,8(E12.4,3X))
      PRINT 20,N,X,(Y(I),I=1,8)
      PRINT 20,N,X,(Y(I),I=9,16)
      PRINT 20,N,X,(Y(I),I=17,24)
      PRINT 20,N,X,(Y(I),I=25,32)
C     ------------------------- (DF'/DTHETA).(Q) -----------------------
      DO 91 I=1,33
   91 Y1(I+7,N)=Y(I)
   75 FDTQ1=FT1*Y1(35,N)+FT2*Y1(36,N)
      FDTQ2=FT3*Y1(37,N)+FT4*Y1(38,N)
      FDTQ3=FT5*Y1(40,N)
C     ---------------------- S+(DF'/DTHETA).Q --------------------------
      SFD1=S1+FDTQ1
      SFD2=S2+FDTQ2
      SFD3=S3+FDTQ3
C     ---------------------- T(INVERSE).(S+(DF'/DTHETA).Q) -------------
      TISFC1=A(1)*SFD1+A(4)*SFD2+A(7)*SFD3
      TISFC2=A(2)*SFD1+A(5)*SFD2+A(8)*SFD3
      TISFC3=A(3)*SFD1+A(6)*SFD2+A(9)*SFD3
      EP=.1
C     ---------------------- T(INVERSE).R ------------------------------
      TIR1=A(1)*R1+A(4)*R2+A(7)*R3
      TIR4=A(1)*R4+A(4)*R5+A(7)*R6
      TIR7=A(1)*R7+A(4)*R8+A(7)*R9
      TIR1C=A(1)*R10+A(4)*R11+A(7)*R12
```

```
      TIR13=A(1)*R13+A(4)*R14+A(7)*R15
      TIR16=A(1)*R16+A(4)*R17+A(7)*R18
      TIR2=A(2)*R1+A(5)*R2+A(8)*R3
      TIR5=A(2)*R4+A(5)*R5+A(8)*R6
      TIR8=A(2)*R7+A(5)*R8+A(8)*R9
      TIR11=A(2)*R10+A(5)*R11+A(8)*R12
      TIR14=A(2)*R13+A(5)*R14+A(8)*R15
      TIR17=A(2)*R16+A(5)*R17+A(8)*R18
      TIR3=A(3)*R1+A(6)*R2+A(9)*R3
      TIR6=A(3)*R4+A(6)*R5+A(9)*R6
      TIR9=A(3)*R7+A(6)*R8+A(9)*R9
      TIR12=A(3)*R10+A(6)*R11+A(9)*R12
      TIR15=A(3)*R13+A(6)*R14+A(9)*R15
      TIR18=A(3)*R16+A(6)*R17+A(9)*R18
C     ----------------------  X(J+1)-X(J)  ------------------------------
      XX1=Y1(1,N)-Y0(1,N)
      XX2=Y1(2,N)-Y0(2,N)
      XX3=Y1(3,N)-Y0(3,N)
      XX4=Y1(4,N)-Y0(4,N)
      XX5=Y1(5,N)-Y0(5,N)
      XX6=Y1(6,N)-Y0(6,N)
      TIRX1=XX1*TIR1+XX2*TIR4+XX3*TIR7+XX4*TIR10+XX5*TIR13+XX6*TIR16
      TIRX2=XX1*TIR2+XX2*TIR5+XX3*TIR8+XX4*TIR11+XX5*TIR14+XX6*TIR17
      TIRX3=XX1*TIR3+XX2*TIR6+XX3*TIR9+XX4*TIR12+XX5*TIR15+XX6*TIR18
C     --------------------  IMPROVED VALUES OF CONTROL VARIABLES  -------
      T1NEW(N)=T1(N)-EP*TISFQ1-TIRX1
      T2NEW(N)=T2(N)-EP*TISFQ2-TIRX2
      ATNEW(N)=AT(N)-EP*TISFQ3-TIRX3
   21 FORMAT (1H ,I3,3X,F5.2,2X,4(E12.4,2X))
      PRINT 21,N,X,Y(33),T1NEW(N),T2NEW(N),ATNEW(N)
      IF (N.EQ.1) PRMT(5)=1.
   19 RETURN
      END
```

```
      SUBRCUTINE CALCL(X,Y,DERY)
      DIMENSION PRMT(10),Y(42),DERY(42),AUX(8,43),Y1(40,202),T1NEW(202)
     1T2NEW(202),ATNEW(202),T1(202),T2(202),AT(202),YO(6,202),A(10),L(1
     2),M(10)
      COMMCN Y1,T1NEW,T2NEW,ATNEW,T1,T2,AT,NK,XX1,XX2,XX3,XX4,XX5,XX6,Y
     1,FT1,FT2,FT3,FT4,FT5,S1,S2,S3,R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,
     212,R13,R14,R15,R16,R17,R18,A,J,N,EP,KSL,R,DIS,V1,V2,CQ,C,AN,AIM,
     3M,C1,C2,C3,CI,CA,CT,EA,EB,XI,YI,GA,GB,RTR1,RTR2,RTR3,RTR4,RTR5,RT
     46,RTR7,RTR8,RTR9,RTR10,RTR11,RTR12,RTR13,RTR14,RTR15,RTR16,RTR17
      COMMCN RTR18,RTR19,RTR20,RTR21,RTS1,RTS2,RTS3,RTS4,RTS5,RTS6,RTFQ
     1,RTFC2,RTFQ3,RTFQ4,RTFC5,RTFQ6,FQ1,FQ2,FQ3,FQ4,FQ5,FQ6,EAT1,EAT2,
     1BT1,EBT2,BM1,BM2,BM3,BM4,BM5,G1,G2,G3,G4
   47 EAT1=EXP(-EA/(R*T1(N)))
      EAT2=EXP(-EA/(R*T2(N)))
      EBT1=EXP(-EB/(R*T1(N)))
      EBT2=EXP(-EB/(R*T2(N)))
      BM1=-(DIS/V1)-GA*EAT1
      BM2=-(DIS/V1)-GB*EBT1
      BM3=-(DIS/V2)-GA*EAT2
      BM4=-(DIS/V2)-GB*EBT2
      BM5=(C+AT(N))*(1.-2.*Y1(6,N)/AN)
      G1=(-EA/(R*(T1(N)**2)))*GA*EAT1
      G2=(-EB/(R*(T1(N)**2)))*GB*EBT1
      G3=(-EA/(R*(T2(N)**2)))*GA*EAT2
      G4=(-EB/(R*(T2(N)**2)))*GB*EBT2
      FT1=Y1(1,N)*G1
      FT2=Y1(2,N)*G2-FT1
      FT3=Y1(3,N)*G3
      FT4=Y1(4,N)*G4-FT3
      FT5=Y1(6,N)*(1.-Y1(6,N)/AN)
C ---------- ALL MATRICES TREATED AS VECTORS , COLUMNWISE -----------
C ------------------------- MATRIX  R  (3X6)  -----------------------
      R1=Y(7)*FT1+Y(8)*FT2+Y(1)*G1-Y(2)*G1
      R2=Y(9)*FT3+Y(10)*FT4
      R3=Y(12)*FT5
      R4=Y(8)*FT1+Y(13)*FT2+Y(2)*G2
      R5=Y(14)*FT3+Y(15)*FT4
      R6=Y(17)*FT5
      R7=Y(9)*FT1+Y(14)*FT2
      R8=Y(18)*FT3+Y(19)*FT4-G3*Y(4)+G3*Y(3)
      R9=Y(21)*FT5
      R10=Y(10)*FT1+Y(15)*FT2
      R11=Y(19)*FT3+Y(22)*FT4+G4*Y(4)
      R12=Y(24)*FT5
      R13=Y(11)*FT1+Y(16)*FT2
      R14=Y(20)*FT3+Y(23)*FT4
      R15=Y(26)*FT5
      R16=Y(12)*FT1+Y(17)*FT2
      R17=Y(21)*FT3+Y(24)*FT4
      R18=Y(27)*FT5+Y(6)*(1.-2.*Y1(6,N)/AN)-4.*CA*AT(N)*Y1(6,N)
```

```
      C ---------------------- MATRIX   S  (3X1) ----------------------------
            S1=2.*CT*(T1M-T1(N))+2.*CT*(T1(N)-T2(N))+Y1(1,N)*G1*Y(1)+Y1(2,N)*G
           12*Y(2)-Y1(1,N)*G1*Y(2)
            S2=2.*CT*(T1(N)-T2(N))+Y1(3,N)*G3*Y(3)+Y1(4,N)*G4*Y(4)-Y1(3,N)*G3
           1*Y(4)
            S3=-2.*CA*AT(N)*(Y1(6,N)**2)+Y(6)*(Y1(6,N)-(Y1(6,N)**2)/AN)
      C --------------------- MATRIX    T   (3X3)  -------------------------
            DT1=-2.*(Y1(1,N)/T1(N))*G1*Y(1)+(Y(1)*G1*EA*Y1(1,N))/(R*(T1(N)**2)
           1)-2.*Y1(2,N)*G2*Y(2)/T1(N)+EB*Y1(2,N)*G2*Y(2)/(R*(T1(N)**2))+(2.*Y
           21(1,N)*G1*Y(2))/T1(N)-EA*Y1(1,N)*G1*Y(2)/(R*(T1(N)**2))
            DT2=-2.*CT
            DT3=0.
            DT4=2.*CT
            DT5=-2.*CT-2.*Y1(3,N)*G3*Y(3)/T2(N)+EA*Y1(3,N)*G3*Y(3)/(R*(T2(N)**
           12))-2.*Y1(4,N)*G4*Y(4)/T2(N)+EB*Y1(4,N)*G4*Y(4)/(R*(T2(N)**2))+2.*
           2Y1(3,N)*G3*Y(4)/T2(N)-EA*Y1(3,N)*Y(4)*G3/(R*(T2(N)**2))
            DT6=0.
            DT7=0.
            DT8=0.
            DT9=-2.*CA*(Y1(6,N)**2)
      C ------------------------- MATRIX  R'T      (6X3)   ------------------
            RT1=R1*DT1+R2*DT2+R3*DT3
            RT2=R4*DT1+R5*DT2+R6*DT3
            RT3=R7*DT1+R8*DT2+R9*DT3
            RT4=R10*DT1+R11*DT2+R12*DT3
            RT5=R13*DT1+R14*DT2+R15*DT3
            RT6=R16*DT1+R17*DT2+R18*DT3
            RT7=R1*DT4+R2*DT5+R3*DT6
            RT8=R4*DT4+R5*DT5+R6*DT6
            RT9=R7*DT4+R8*DT5+R9*DT6
            RT10=R10*DT4+R11*DT5+R12*DT6
            RT11=R13*DT4+R14*DT5+R15*DT6
            RT12=R16*DT4+R17*DT5+R18*DT6
            RT13=R1*DT7+R2*DT8+R3*DT9
            RT14=R4*DT7+R5*DT8+R6*DT9
            RT15=R7*DT7+R8*DT8+R9*DT9
            RT16=R10*DT7+R11*DT8+R12*DT9
            RT17=R13*DT7+R14*DT8+R15*DT9
            RT18=R16*DT7+R17*DT8+R18*DT9
      C ------------------- MATRIX  R'TR      (6X6)    ------------------
      C    MATRIX  DP/DT BEING SYMMETRICAL ONLY HALF THE MATRIX R'TR IS TAKEN
            RTR1=R1*RT1+R2*RT7+R3*RT13
            RTR2=R1*RT2+R2*RT8+R3*RT14
            RTR3=R1*RT3+R2*RT9+R3*RT15
            RTR4=R1*RT4+R2*RT10+R3*RT16
            RTR5=R1*RT5+R2*RT11+R3*RT17
            RTR6=R1*RT6+R2*RT12+R3*RT18
            RTR7=R4*RT2+R5*RT8+R6*RT14
            RTR8=R4*RT3+R5*RT9+R6*RT15
            RTR9=R4*RT4+R5*RT10+R6*RT16
```

```
      RTR10=R4*RT5+R5*RT11+R6*RT17
      RTR11=R4*RT6+R5*RT12+R6*RT18
      RTR12=R7*RT3+R8*RT9+R9*RT15
      RTR13=R7*RT4+R8*RT10+R9*RT16
      RTR14=R7*RT5+R8*RT11+R9*RT17
      RTR15=R7*RT6+R8*RT12+R9*RT18
      RTR16=R10*RT4+R11*RT10+R12*RT16
      RTR17=R10*RT5+R11*RT11+R12*RT17
      RTR18=R10*RT6+R11*RT12+R12*RT18
      RTR19=R13*RT5+R14*RT11+R15*RT17
      RTR20=R13*RT6+R14*RT12+R15*RT18
      RTR21=R16*RT6+R17*RT12+R18*RT18
C -------------------- CALCULATION OF  T-INVERSE ----------------------
      A(1)=DT1
      A(2)=DT2
      A(3)=DT3
      A(4)=DT4
      A(5)=DT5
      A(6)=DT6
      A(7)=DT7
      A(8)=DT8
      A(9)=DT9
      NN=3
      CALL MINV(A,NN,D,L,M)
C  -------------------- MATRIX R'.T(INVERSE)          (6X3) ----------
      RTI1=R1*A(1)+R2*A(2)+R3*A(3)
      RTI2=R4*A(1)+R5*A(2)+R6*A(3)
      RTI3=R7*A(1)+R8*A(2)+R9*A(3)
      RTI4=R10*A(1)+R11*A(2)+R12*A(3)
       RTI5=R13*A(1)+R14*A(2)+R15*A(3)
       RTI6=R16*A(1)+R17*A(2)+R18*A(3)
      RTI7=R1*A(4)+R2*A(5)+R3*A(6)
      RTI8=R4*A(4)+R5*A(5)+R6*A(6)
      RTI9=R7*A(4)+R8*A(5)+R9*A(6)
      RTI10=R10*A(4)+R11*A(5)+R12*A(6)
      RTI11=R13*A(4)+R14*A(5)+R15*A(6)
      RTI12=R16*A(4)+R17*A(5)+R18*A(6)
      RTI13=R1*A(7)+R2*A(8)+R3*A(9)
      RTI14=R4*A(7)+R5*A(8)+R6*A(9)
      RTI15=R7*A(7)+R8*A(8)+R9*A(9)
      RTI16=R10*A(7)+R11*A(8)+R12*A(9)
      RTI17=R13*A(7)+R14*A(8)+R15*A(9)
      RTI18=R16*A(7)+R17*A(8)+R18*A(9)
C   ---------------------  MATRIX  R'.T(INVERSE).S     (6X1)  -------
      RTS1=S1*RTI1+S2*RTI7+S3*RTI13
      RTS2=S1*RTI2+S2*RTI8+S3*RTI14
      RTS3=S1*RTI3+S2*RTI9+S3*RTI15
      RTS4=S1*RTI4+S2*RTI10+S3*RTI16
      RTS5=S1*RTI5+S2*RTI11+S3*RTI17
      RTS6=S1*RTI6+S2*RTI12+S3*RTI18
```

```
C ------------------------------------------------------------------------
      RTF1=RTI1*FT1
      RTF2=RTI2*FT1
      RTF3=RTI3*FT1
      RTF4=RTI4*FT1
      RTF5=RTI5*FT1
      RTF6=RTI6*FT1
      RTF7=RTI1*FT2
      RTF8=RTI2*FT2
      RTF9=RTI3*FT2
      RTF10=RTI4*FT2
      RTF11=RTI5*FT2
      RTF12=RTI6*FT2
      RTF13=RTI7*FT3
      RTF14=RTI8*FT3
      RTF15=RTI9*FT3
      RTF16=RTI10*FT3
      RTF17=RTI11*FT3
      RTF18=RTI12*FT3
      RTF19=RTI7*FT4
      RTF20=RTI8*FT4
      RTF21=RTI9*FT4
      RTF22=RTI10*FT4
      RTF23=RTI11*FT4
      RTF24=RTI12*FT4
      RTF25=0.
      RTF26=0.
      RTF27=0.
      RTF28=0.
      RTF29=0.
      RTF30=0.
      RTF31=RTI13*FT5
      RTF32=RTI14*FT5
      RTF33=RTI15*FT5
      RTF34=RTI16*FT5
      RTF35=RTI17*FT5
      RTF36=RTI18*FT5
C ------------------------------------------------------------------------
      RTFQ1=Y(28)*RTF1+Y(29)*RTF7+Y(30)*RTF13+Y(31)*RTF19+Y(32)*RTF25+Y
     133)*RTF31
      RTFQ2=Y(28)*RTF2+Y(29)*RTF8+Y(30)*RTF14+Y(31)*RTF20+Y(32)*RTF26+Y
     133)*RTF32
      RTFQ3=Y(28)*RTF3+Y(29)*RTF9+Y(30)*RTF15+Y(31)*RTF21+Y(32)*RTF27+Y
     133)*RTF33
      RTFQ4=Y(28)*RTF4+Y(29)*RTF10+Y(30)*RTF16+Y(31)*RTF22+Y(32)*RTF28+
     1(33)*RTF34
      RTFQ5=Y(28)*RTF5+Y(29)*RTF11+Y(30)*RTF17+Y(31)*RTF23+Y(32)*RTF29+
     1(33)*RTF35
      RTFQ6=Y(28)*RTF6+Y(29)*RTF12+Y(30)*RTF18+Y(31)*RTF24+Y(32)*RTF30+
     1(33)*RTF36
```

```
      C ----------------------------------------------------------------------
            FQ1=Y(28)*BM1
            FQ2=Y(28)*GA*EAT1+Y(29)*BM2
            FQ3=Y(28)*(DIS/V2)+Y(30)*BM3
            FQ4=Y(29)*(DIS/V2)+Y(30)*GA*EAT2+Y(31)*BM4
            FQ5=Y(31)*DIS
            FQ6=Y(33)*(C+AT(N))*(1.-2.*Y1(6,N)/AN)
            RETURN
            END
      C
      C     ...................................................................
      C
      C        SUBROUTINE MINV
      C
      C        PURPOSE
      C           INVERT A MATRIX
      C
      C        USAGE
      C           CALL MINV(A,N,D,L,M)
      C
      C        DESCRIPTION OF PARAMETERS
      C           A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
      C               RESULTANT INVERSE.
      C           N - ORDER OF MATRIX A
      C           D - RESULTANT DETERMINANT
      C           L - WORK VECTOR OF LENGTH N
      C           M - WORK VECTOR OF LENGTH N
      C
      C        REMARKS
      C           MATRIX A MUST BE A GENERAL MATRIX
      C
      C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
      C           NONE
      C
      C        METHOD
      C           THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
      C           IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
      C           THE MATRIX IS SINGULAR.
      C
      C     ...................................................................
      C
```

```
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C
C        ...............................................................
C
C        IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C        C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C        STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C        THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C        APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C        ROUTINE.
C
C        THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C        CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT
C        10 MUST BE CHANGED TO DABS.
C
C        ...............................................................
C
C        SEARCH FOR LARGEST ELEMENT
C
      D=1.0
      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
   10 IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)
      L(K)=I
      M(K)=J
   20 CONTINUE
C
C        INTERCHANGE ROWS
C
      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
C
C        INTERCHANGE COLUMNS
C
   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
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   45 IF(BIGA) 48,46,48
   46 D=0.0
      RETURN
   48 DO 55 I=1,N
      IF(I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
C
C        REDUCE MATRIX
C
      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
C
C        DIVIDE ROW BY PIVOT
C
      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
C
```

```
C          PRODUCT OF PIVOTS
C
           D=D*BIGA
C
C          REPLACE PIVOT BY RECIPROCAL
C
           A(KK)=1.0/BIGA
        80 CONTINUE
C
C          FINAL ROW AND COLUMN INTERCHANGE
C
           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 KI=K-N
           DO 130 I=1,N
           KI=KI+N
           HOLD=A(KI)
           JI=KI-K+J
           A(KI)=-A(JI)
       130 A(JI) =HOLD
           GO TO 100
       150 RETURN
           END
```

```
      SUBROUTINE RKGS(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
C
      DIMENSION Y(1),DERY(1),AUX(8,1),A(4),B(4),C(4),PRMT(1)
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.
      CALL FCT(X,Y,DERY)
C
C
C     PREPARATIONS FOR RUNGE-KUTTA METHOD
    2 A(1)=.5
      A(2)=.2928932
      A(3)=1.707107
      A(4)=.1666667
      B(1)=2.
      B(2)=1.
      B(3)=1.
      B(4)=2.
      C(1)=.5
      C(2)=.2928932
      C(3)=1.707107
      C(4)=.5
C
C     PREPARATIONS OF FIRST RUNGE-KUTTA STEP
      DO 3 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
      AUX(3,I)=0.
    3 AUX(6,I)=0.
C
C     RECORDING OF INITIAL VALUES OF THIS STEP
    7 CALL OUTP(X,Y,DERY,IREC,NDIM,PRMT)
      IF(PRMT(5))40,8,40
C
C
C     START OF INNERMOST RUNGE-KUTTA LOOP
    8 J=1
   10 AJ=A(J)
      BJ=B(J)
      CJ=C(J)
      DO 11 I=1,NDIM
      R1=H*DERY(I)
      R2=AJ*(R1-BJ*AUX(6,I))
      Y(I)=Y(I)+R2
      R2=R2+R2+R2
   11 AUX(6,I)=AUX(6,I)+R2-CJ*R1
      IF(J-4)12,15,15
   12 J=J+1
      IF(J-3)13,14,13
```

```
   13 X=X+.5*H
   14 CALL FCT(X,Y,DERY)
      GOTO 10
C     END OF INNERMOST RUNGE-KUTTA LOOP
C
C
   15 DO 29 I=1,NDIM
      AUX(1,I)=Y(I)
      AUX(2,I)=DERY(I)
   29 AUX(6,I)=AUX(3,I)
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))40,30,40
   30 DO 31 I=1,NDIM
      Y(I)=AUX(1,I)
   31 DERY(I)=AUX(2,I)
      GO TO 8
   40 RETURN
      END
```

OPTIMIZATION OF MANAGEMENT SYSTEMS

BY SECOND VARIATION

by

SHASHIKANT KRISHNARAO RANGNEKAR

B.E. (Mechanical Engineering), Bombay University
Bombay, India, 1966

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Industrial Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1969

ABSTRACT

There are many difficulties in using either the classical multistage optimization techniques or dynamic programming for solving nonlinear complex problems involving a fairly large number of variables. The former gives boundary value difficulties while the latter has the difficulty of dimensionality. The methods of gradients and other techniques such as quasilinearization partially overcome these difficulties.

The basic philosophy of the methods of gradients is fairly simple. First a sequence of values of the control vector is selected. Then the gradient of the performance index with respect to each of the control vector is calculated. Finally each control vector is improved by moving it in the direction of the gradient. This improved sequence of control vectors then becomes the basis for the next iteration.

The functional gradient technique, one of the many versions of the gradient methods, has been developed for optimal control problems. The second variation method overcomes certain difficulties of the functional gradient technique. The convergence rate of the second variation method, provided the method converges, is very fast. However, the initial guess of the trajectory for the control variable has to be near the optimal trajectory in order to obtain convergence. Too, the number of equations to be integrated and their complexity tend to suppress its advantage of rapid convergence.

First, the method of second variation is discussed in detail. Then the method is applied to three problems in the field of production and inventory control to illustrate the approach.

The first application is a simple inventory model involving one state variable and one control variable. The objective function is the cost function, which is to be minimized. The second application is an inventory and advertising model where it is desired to maximize the profit function. This problem has two state variables and one control variable. The last application is that of a chemical manufacturing problem with advertisement. It has six state variables and three control variables.

These examples suggest that the first variation method, of which the second variation method is a natural evolution, should be used in combination with the second variation. The first variation method, unlike the second variation, will approach optimum from almost any realistic starting trajectory. The results of the first variation method could then be used as the starting trajectories for the second variation. In this way, the convergence problem of the second variation can be partly overcome. Furthermore, this combination provides a rapid convergence from almost any realistic starting trajectory for most engineering problems.