IMPROVED STOCHASTIC DYNAMIC PROGRAMMING FOR OPTIMAL RESERVOIR OPERATION BASED ON THE ASYMPTOTIC CONVERGENCE OF BENEFIT DIFFERENCES

by

Alberto Guitron De Los Reyes

A Thesis Submitted to the Faculty of the

DEPARTMENT OF HYDROLOGY AND WATER RESOURCES

In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE
WITH A MAJOR IN WATER RESOURCES ADMINISTRATION

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 7 4

STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Request for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: A. Guitrón

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

Donald R Davis — Dec 3, 1974

Donald R. Davis
Professor of Hydrology and Water Resources

Date

ACKNOWLEDGMENTS

I am deeply indebted to individuals both on and off the campus of The University of Arizona for their support.

On campus, I wish to express my deep appreciation to Dr. Don Davis, who reviewed the thesis and provided helpful comments.

I am grateful to Dr. Hasan K. Qashu, for his willingness to assist me in many academic and personal problems.

Off campus, special gratitude is due the Water Resources Ministry of the Mexican Government and the National Council of Science and Technology of Mexico for their financial support during my stay in Tucson.

Special gratitude goes to my lovely wife Ana for her patience and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

ABSTRACT

Stochastic dynamic programming has been widely used to solve for optimal operating rules for a single reservoir system. In this thesis a new iterative scheme is given which results to be more efficient in terms of computational effort than the conventional stochastic dynamic approach. The scheme is a hybrid one composed of the conventional procedure alternating with iterations over a fixed policy in order to increase the chance of finding the optimal policy more rapidly. Likewise this thesis introduces a refined technique to derive transition probability matrices and the use of bounded variables in the recursive equation to provide an easier way to verify the convergence of the cyclic gain of the system. A computer program is developed to implement the new iterative scheme and then it is applied to a real world problem in order to derive quantitative comparisons. A real savings of twenty-five percent of the computational time required with the conventional procedure is obtained with the new iterative scheme.

CHAPTER 1

INTRODUCTION

There is a substantial body of literature on the solution of the optimal operating rule problem for a single stochastic reservoir. Linear programming models, dynamic programming models and policy iteration models have all been used, as reviewed by Loucks and Falkson (1970).

Most of the work on the stochastic linear programming approach to this problem has been done by Loucks (1969). This approach is ingenious but computationally difficult. An unreported experiment by Theodore G. Roefs (1974) was conducted in 1968. A twelve season problem was solved using both the linear programming approach and the stochastic dynamic approach. The linear programming approach required more than an order of magnitude more computer time and the dynamic programming approach. Hassitt (1968) shows the relationship between the models.

Most of the work on the policy iteration approach to this problem has been done by Howard (1960). The policy iteration method has some of the features of both linear and dynamic programming approaches. Like all linear programming problems it is necessary to solve sets of linear simultaneous equations, which even for small problems can become quite large. As in dynamic programming problems, the policy iteration method develops a recursive relationship which is divided into two steps in an effort to examine only the steady state values, i.e., those that apply when a large number of transitions remain before the economical and hydrological conditions change.

The difficulties associated with Howard's (1960) method when the number of states is large have been pointed out by White (1963), Odoni (1969) and MacQueen (1966). In fact, an experiment reported by Odoni (1969), for ten state and twenty-five state problems with three alternatives per state, the stochastic dynamic approach reduced computation time to one half and one fifth respectively, of the time required for policy iteration solution.

There is an additional drawback for linear programming models and policy iteration models. There is no doubt that the number of simultaneous linear equations that can be accurately solved on current day computers is less than the number that may be requested for a particular reservoir operation system. In the process of solving large linear programming problems, computer roundoff and truncation errors may result in an initially feasible solution becoming infeasible. This obviously constraints the size of the problem that can be examined using techniques such as in linear programming or those developed by Howard (1960). If the dynamic programming approach is used, the solution of simultaneous equations is not required.

White (1963) first modified the method of successive approximations (dynamic programming) developed by Bellman (1957), for solving Markov decision problems to focus attention on the convergence of costs relative to the cost of a base state (cost differences), rather than on convergence of the total cost function.

For the undiscounted case he proved that the modified cost function converged at least geometrically, irrespective of the sequence of policy chosen. Schweitzer (1965) proved convergence for the more general single-chain aperiodic case. MacQueen (1966) and Odoni (1969) have extended White's (1963) results to provide computable upper and lower bounds on the gain of the process at each iteration.

Su and Deininger (1972) likewise extended Odoni's (1969) results to the case where the Markovian decision process is periodic. The method used for solving the periodic Markovian decision problem consists in a simple backward recursive optimization routine. Owing to the periodicity of the Markovian process, the maximal expected payoff per cycle, instead of the maximal expected payoff per transition, is recursively computed until this value from every state sufficiently converges to a constant. The converged constant would be the desired approximated maximal expected average payoff per cycle, and the policy used in the last iteration (in the cycle) is the desired optimal policy.

Vleugels (1972) has demonstrated that Su and Deininger's (1972) model can be simplified in order to consider a Markov process with monthly transitions instead of a model with yearly transitions. By decomposing the problem into one of monthly transitions, the optimization problem decreases in size substantially. Additionally, Vleugels (1972) demonstrated rather elegantly the properties of convergence of this model through the use of contraction mapping techniques. Vleugels' (1972) model is basically what the author considers as the stochastic dynamic programming approach (SDP).

The technique used to solve the model consists of a simple backward recursive optimization algorithm linked with a stopping procedure. The stopping procedure most generally employed consists of the termination of the algorithm when the policy repeats itself in two successive cycles. However, in theory a policy that repeats itself need not be optimal, not even if there is only one optimal policy. In this case, convergence of the modified cost function guarantees optimality of a repeating policy. Obviously, after that point, one is indeed iterating a fixed policy. It is necessary for the modified cost function to converge to guarantee that this policy will, in fact, be optimal.

Morton (1971) has studied the asymptotic convergence rate of cost differences for the case of a fixed policy. He demonstrated that the modified cost function converges geometrically to the yearly gain of the process. Additionally, he extracted an interesting implication that can be summarized as follows: if the problem is "well behaved" in the sense that policies with gains close to that of the optimal policy also have transition matrices very similar to that of the optimal policy, then one would expect convergence characteristics "close to" that for a fixed policy.

This implication, together with the fact that the optimal policy (if there is one) is often reached very early, lead us to the conclusion that a hybrid iterative scheme composed of the conventional procedure alternating with iterations over a fixed policy might be more efficient in terms of computational effort than the conventional procedure. The object of this thesis is to establish the theoretical basis for the new algorithm and to establish its advantages.

To pursue such goals this thesis is organized in the following manner. Chapter 2 introduces the Markovian model of the single reservoir system and its corresponding conventional dynamic programming formulation. Chapter 3 extends the results of asymptotic convergence over a fixed policy, as derived by Morton (1971), to the more general case where the Markovian decision process is periodic. The new iterative scheme is presented at the end of this chapter and its principal characteristics are exposed. Chapter 4 presents a case study where the new technique is applied and compared with the traditional approach. First, a physical characterization of the system is given. The refined technique to compute state transition probabilities is then introduced. Thirdly, quantitative comparisons are made between both approaches and computational implications are drawn. Finally, Chapter 5 presents some general conclusions and discussion of the study.

CHAPTER 2

SINGLE RESERVOIR MODEL AND SOLUTION TECHNIQUE

The water reservoir system under consideration will be modelled as a Markovian Decision Process. The solution technique used to solve the model will be stochastic dynamic programming and it is presented in this chapter.

2.1 The Model

Consider Figure 2.1 which is a diagrammatic representation of the single reservoir system, where S_t, i_t and r_t are the initial storage volume, streamflow input and release, respectively, for some month t.

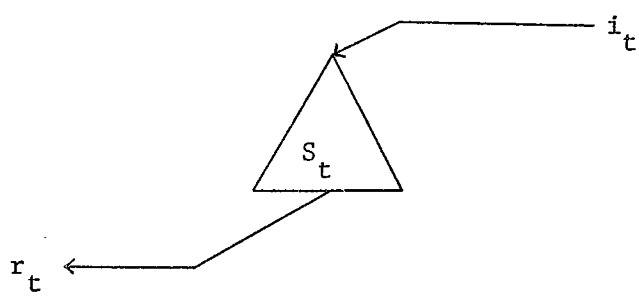


Figure 2.1. Single Reservoir System.

Two major assumptions with respect to the stochastic process of inflow should be established before the system can be modelled as a Markovian decision process.

Assumption 1: The inflows in every time interval (a month) to the reservoir are statistically dependent on the inflow of the preceding interval.

Assumption 2: The stochastic process of these inflows is periodic with period T; T time intervals define a cycle $(T = 12$ in this case).

Given Assumptions 1 and 2 it is possible to represent the stream flow input to the reservoir as a Markov chain which is periodic with a period of twelve months. The stochastic process is defined by a set of probabilities of the form $p(i_t/i_{t-1})$. Since these are conditional probabilities

$$\sum_{i_t} p(i_t/i_{t-1}) = 1, \qquad \text{for all } i_{t-1}$$

$$0 \leq p(i_t/i_{t-1}) \leq 1, \quad \text{for all } i_t, \; i_{t-1}, \; t$$

The following notation will be used in this thesis:

T - number of months in a year, $T = 12$;

t - number of time interval in the cycle, $t = 1, 2, \ldots, T$;

m - time horizon in years;

n - number of years remaining until the termination of the process, $n = 0, 1, 2, \ldots, m$;

S_t - storage volume of reservoir at the start of month t;

S_{t+1} - storage volume of reservoir at the start of month $t + 1$;

i_t - streamflow input to reservoir during month t;

i_{t+1} - streamflow input to reservoir during month $t + 1$;

r_t - release from the reservoir during month t.

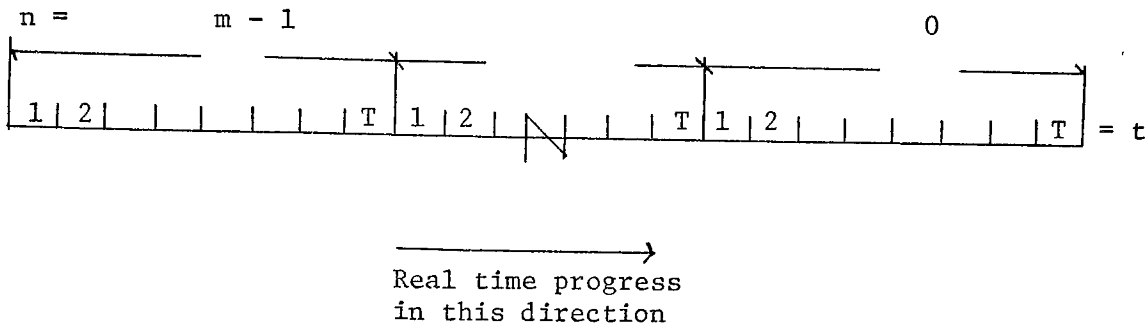
Figure 2.2 exhibits a schematic representation of the relationship between the variables previously defined.

The governing equation of the single reservoir system is:

$$S_{t+1} = S_t + i_t - r_t \tag{2.1}$$

which states that the storage volume at the beginning of month $t + 1$ is equal to the sum of the initial storage volume and streamflow input of month t minus the release made during month t, where the release is made according to the following decision process. The decision process is to decide upon the release r_t to be made after observing the state (S_t, i_{t-1}) of the system, i.e., the decision is made after observing the storage at the beginning of month t and previous inflow in month $t - 1$.

(a)



(b)

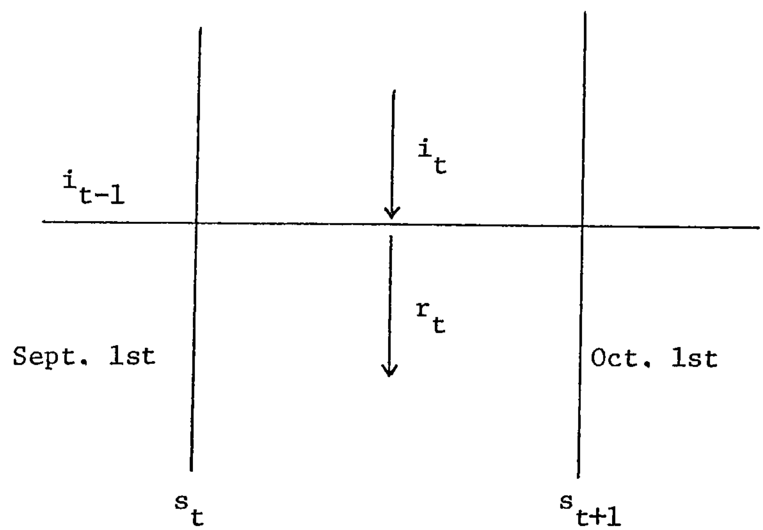


Figure 2.2. Schematic Representation of the Relationship Between the Variables. (a) Schematic Representation of the Time Variable. (b) Time Relationship Between Model Variables.

The decision variable r_t is constrained by $LC_t \leq r_r \leq UC_t$ for each t, where LC_t and UC_t are the prescribed minimum and maximum permissible drafts. Physically these limits could correspond with obligatory release and channel capacity, respectively. Additionally the water storage levels of the reservoir are bounded above by the physical size limitation of the reservoir; below by the minimum storage level at which the water can be extracted from the reservoir.

Any r_t, such that $LC_t \leq r_t \leq UC_t$, will be a feasible decision. An operating policy is a set of feasible decisions that indicates what action is to be taken after observing the state (S_t, i_{t-1}) of the system.

For mathematical manipulation, let all variables be discretized taking some finite number of discrete values. Now, assume that it is possible to derive monthly transition rewards of the form

$V_t(r_t)$ - immediate reward when the system makes a monthly transition from month t to month t + 1, when the release is r.

Let us assume also, that the immediate reward is independent of the year of the process, that is

$$V_{T+t}(r_t) = V_t(r_t) \quad \text{for all } t. \tag{2.2}$$

It is also possible to derive monthly transition probabilities of the form

$P[(S_t, i_{t-1})(S_{t+1}, i_t); r_t]$ Probability to transitioning to state (S_{t+1}, i_t) given the system is in state (S_t, i_{t-1}), where the release was r in month t.

Where by Assumptions 1 and 2 and the forcing equation (2.1) it is obvious that

$$P[(S_t, i_{t-1})(S_{t+1}, i_t); r_t] = \begin{cases} p(i_t/i_{t-1}) & \text{if } S_{t+1} = S_t + i_t - r_t \\ 0 & \text{otherwise} \end{cases} \quad (2.3)$$

which simply says that if the policy specifies a particular final reservoir volume S_{t+1} in period $t + 1$ for an initial storage volume S_t and inflow i_t in period t then the state transition probability equals the inflow transition probability, otherwise the probability of that particular state is zero.

Now consider the system as a stochastic process observed at discrete points t to be in one of a finite number of states (S_t, i_{t-1}); (S_t, i_{t-1}) is a periodic Markovian process with period T and has transition probabilities given by (2.3). The reservoir system can be thought of as making transitions from one state (S_t, i_{t-1}) to another state (S_{t+1}, i_t) in successive periods. If there are T periods in a year, then t proceeds from 1, 2, 3, ..., $T - 1$, T, 1, 2, ..., to some distant time horizon m years or mT periods in the future.

Given Assumptions 1 and 2 and equations (2.1) and (2.3), it is now possible to derive the mathematical model for the system. Consider that the system is currently at the beginning of a time interval that is $(n + 1)T - t + 1$ intervals from termination. At this time the system is observed to be in state (S_t, i_{t-1}) in period t. If a decision r_t is made from a finite set of feasible decisions, then an immediate reward $V_t(r_t)$ is obtained, and the system moves to a state (S_{t+1}, i_t) at the end of this interval following the probability function $p(i_t / i_{t-1})$. Note that the reward in each period is independent of the inflow i_t.

Let $f_{(n+1)T-t+1}(S_t, i_{t-1})$ be the maximal total expected undiscounted reward from the next $(n + 1)T - t + 1$ intervals. Thus with $(n + 1)T - t + 1 = 1$ period to go, the total expected reward is

$$f_1(S_t, i_{t-1}) = V_t(r_t), \quad \text{for all } S_t,\ i_{t-1},\ r_t = f(S_{t+1}, i_t, S_t). \qquad (2.4)$$

With $(n + 1)T - t + 1 = 2$ periods to go, the total expected reward is

$$f_2(S_t,i_{t-1}) = V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) f_1(S_{t+1},i_t),$$

for all S_t, i_{t-1}, $r_t = f(S_t,s_{t+1},i_t)$

where each $f_1(S_{t+1},i_t)$ was determined by equation (2.4). In general the recursive relationship for determining the total expected reward with $(n + 1)T - t + 1$ stages to go, given a particular operating policy (a fixed policy) $r_t = f(S_t,S_{t+1},i_t)$, is

$$f_{(n+1)T-t+1}(S_t,i_{t-1}) = V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) f_{(n+1)T-t}(S_{t+1},i_t).$$

An optimal policy is defined as one that maximizes the expected total reward, that is

$$f_{(n+1)T-t+1}(S_t,i_{t-1}) = \mathrm{Max}[V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) f_{(n+1)T-t}(S_{t+1},i_t)]$$

for all S_t, i_{t-1}, t. (2.5)

Starting at some time in the future and using the connection between the inflows in one time period and the adjacent time period it is possible to calculate values of r_t for each time period as a function of the state variable (S_t,i_{t-1}). These r_t's then form an optimal policy of the operation of the reservoir. The important point to note is that under conditions of ergodicity of the process this policy is said to converge when the values of r_t that are used to evaluate the function $f_{(n + 1)T - t + 1}(S_t, i_{t-1})$ repeat for all values of t when the number of transitions becomes larger.

Howard (1960) proved that an optimal policy will converge if the system is ergodic. This term implies that the final state of the system is independent of the starting state. In other words this is equivalent to stating that no matter what the state of the reservoir at the start of the computations the steady state of the system will be independent of the starting state.

2.2 Alternative Approach

Equation (2.5) is the application of the "Principle of Optimality" of dynamic programming to the Markovian decision process as stated by Bellman (1957). It can be solved straightforward using a backward optimization routine. However, an alternate procedure is still feasible if we now consider the Markovian process as an aperiodic one. That is, if we can derive an equivalent equation to that of (2.5) but now transforming the periodic process to one that is aperiodic, such alternate approach can be accomplished.

Now, let us introduce an operator Δ_t defined for any function h by

$$\Delta_t h(S_{t+1}, i_t) = \underset{r_t}{\mathrm{Max}}[V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) h(S_{t+1}, i_t)] \quad \text{for all}$$

S_{t+1}, i_t, t

with this new notation equation (2.5) becomes:

$$f_{(n+1)T-t+1}(S_t, i_{t-1}) = \Delta_t f_{(n+1)T-t}(S_{t+1}, i_t). \tag{2.6}$$

Now, let us consider the recursive application of equation (2.6) for some values of t. For $t = 1$

$$f_{(n+1)T}(S_1, i_{12}) = \Delta_1 f_{(n+1)T-1}(S_2, i_1) \tag{2.7}$$

$t = 2$

$$f_{(n+1)T-1}(S_2, i_1) = \Delta_2 f_{(n+1)T-2}(S_3, i_2) \tag{2.8}$$

now, using the values of $f_{(n+1)T-1}(S_2, i_1)$ derived in (2.8) and substituting in (2.7) we have

$$f_{(n+1)T}(S_1, i_{12}) = \Delta_1 \Delta_2 f_{(n+1)T-2}(S_3, i_2)$$

following the same reasoning for T periods we have

$$f_{(n+1)T}(S_1, i_{12}) = \Delta_1 \Delta_2 \cdots \Delta_T f_{nT}(S_1, i_{12}). \tag{2.9}$$

Equation (2.9) can be rewritten to consider any starting period t as

$$f_{(n+1)T}(S_t, i_{t-1}) = \Delta_t \Delta_{t+1} \cdots \Delta_{t+T} f_{nT}(S_{t+T}, i_{t+T-1})$$

or in short form

$$f_{(n+1)T}(S_t, i_{t-1}) = \theta_t f_{nT}(S_{t+T}, i_{t+T-1})$$

where

$$\theta_t = \Delta_t \Delta_{t+1} \cdots \Delta_{T+t}.$$

The operator θ_t was introduced for the following reason: instead of considering one step at a time, one looks at times 0, T, 2T, ..., from termination. At each such time, a policy (ζ) that consists of the next T decisions is chosen from a set of feasible policies Z, where $Z = r_1 \times r_2 \times r_3, \ldots, \times r_T$ and r_t is the set of feasibles relases for month t (T months in a year).

Any policy $\zeta \in Z$ chosen in state (S_t, i_{t-1}) leads to a T-step reward $R_t(\zeta)$ and a T-step transition probability $\overline{P}(\zeta) = [p(i_{t+T-1}/i_{t-1})]$ to state (S_{t+T}, i_{t+T-1}).

It is clear from the definition of θ_t, that each yearly transition probability $p(i_{T+t-1}/i_{t-1})$ is comprised of twelve individual monthly transition probabilities; hence, the yearly transition probability is the product of twelve monthly transition probabilities. The functional relationship between yearly and monthly transition probabilities is given by

$$p(i_{t+T-1}/i_{t-1}) = \prod_{\tau=t+1}^{T+t-1} \{ \sum_{i_{\tau-1}} p(i_{\tau-1}/i_{\tau-2}) \} p(i_{\tau}/i_{\tau-1}).$$

In a similar way, the yearly transition reward $R_t(\zeta)$ is comprised of twelve monthly transition rewards, each of which must be multiplied by the probability of obtaining these monthly rewards. The functional relationship between yearly and monthly rewards is given by

$$R_t(\zeta) = V_t(r_t) + \sum_{\tau=t+1}^{T+t-1} \{ \prod_{s=2}^{\tau} (\sum_{i_{s-1}} p(i_{s-1}/i_{s-2})) \} V_{\tau}(r_{\tau}).$$

Consequently the operator θ_t is therefore given by

$$\theta_t h(S_{t+T}, i_{t+T-1}) = \underset{\zeta \in Z}{\mathrm{Max}} [R_t(\zeta) + \sum_{i_{t+T-1}} p(i_{t+T-1}/i_{t-1}) h(S_{t+T}, i_{t+T-1})]$$

for all S_{t+T}, i_{t+T-1}, t.

Finally, the recursive equation that solves the model for any starting period t is given by

$$f_{(n+1)T}(S_t, i_{t-1}) = \underset{\zeta}{\mathrm{Max}} [R_t(\zeta) + \sum_{i_{t+T-1}} p(i_{t+T-1}/i_{t-1}) f_{nT}(S_{t+T}, i_{t+T-1})]. \tag{2.10}$$

It is important to point out that equation (2.10) will be solved recursively for a specific time period t, say $t = 1$, since all the remaining periods are taken implicitly into account by the equation.

CHAPTER 3

ASYMPTOTIC CONVERGENCE OF BENEFIT DIFFERENCES FOR PERIODIC MARKOVIAN PROCESSES

3.1 Asymptotic Convergence for a Fixed Policy

To derive the asymptotic convergence properties of the expected reward function for the case of a single policy, the alternative approach to solve the Markovian decision process will be used here. Some concluding remarks from the previous chapter will be recalled. First of all, it is necessary to consider that a finite state discrete Markovian system is controlled by a decision maker. After each yearly transition $(n = 0, 1, 2, \ldots, m)$ the system is in one of N states (S_t, i_{t-1}). Then the decision maker picks one of the Z policies resulting in a yearly reward $R_t(\zeta)$ for the cycle and a yearly transition probability row vector
$\overline{P}(\zeta) = [p(i_{t+T-1}/i_{t-1})]$, for all i_{t+T-1}, where

$[p(i_{t+T-1}/i_{t-1})]$ is the probability of going from state (S_t, i_{t-1}) to state (S_{t+T}, i_{t+T-1})

given policy (ζ). The maximal total expected reward for any starting period t for the next n cycles, if an optimal policy is followed, can be written as

$$f_{(n+1)T}(S_t, i_{t-1}) \tag{3.1}$$

$$= \underset{\zeta}{\mathrm{Max}}[R_t(\zeta) + \sum_{i_{t+T-1}} p(i_{t+T-1}/i_{t-1}) f_{nT}(S_{t+T}, i_{t+T-1})].$$

Since the term $\sum_{i_{t+T-1}} p(i_{t+T-1}/i_{t-1}) f_{nT}(S_{t+T}, i_{t+T-1})$ can be written as the product of a row vector $\overline{P}(\zeta)$ belonging to the transition matrix $\{P_t(\zeta)\}$, times a column vector $\overline{f}_{nT}$, equation (3.1) can be written as

$$f_{(n+1)T}(S_t, i_{t-1}) = \underset{\zeta}{\mathrm{Max}}[R_t(\zeta) + \overline{P}_t(\zeta)\ \overline{f}_{nT}]. \tag{3.2}$$

Considering now the vector $\overline{f}_{(n+1)T}$ that includes all the elements $f_{(n+1)T}(S_t, i_{t-1})$, the following vector equation can be written:

$$\overline{f}_{(n+1)T} = \underset{\zeta}{\mathrm{Max}}\{\overline{R}_t(\zeta) + \{P_t(\zeta)\}\overline{f}_{nT}\},$$

where $\overline{R}_t(\zeta)$ is the yearly reward vector and $\{P_t(\zeta)\}$ is the yearly transition matrix.

Let us consider the case of a fixed policy for which in the stationary condition a yearly transition probability matrix $\{P\}$ and a yearly transition reward vector $\overline{R}$ have been permanently achieved. This statement is translated mathematically to the following equation:

$$\overline{f}_{(n+1)T} = \overline{R} + \{P\}\overline{f}_{nT}. \qquad (3.3)$$

Writing out values of (3.3) for some particular values of n we have for $n = 1$;

$$\overline{f}_T = \overline{R} + \{P\}\overline{f}_0 \qquad (3.4)$$

for $n = 2$;

$$\overline{f}_{2T} = \overline{R} + \{P\}\overline{f}_T. \qquad (3.5)$$

Substituting the value of $\overline{f}_T$ obtained in (3.4) into (3.5) yields,

$$\overline{f}_{2T} = \overline{R} + \{P\}[\overline{R} + \{P\}\overline{f}_0].$$

Simplifying, we have

$$\overline{f}_{2T} = \overline{R} + \{P\}\overline{R} + \{P\}^2\overline{f}_0.$$

Following a similar reasoning an expression for $\overline{f}_{3T}$ is obtained as follows

$$\overline{f}_{3T} = \overline{R} + \{P\}\overline{R} + \{P\}^2\overline{R} + \{P\}^3\overline{f}_0.$$

In general, for $n = n$ we have,

$$\bar{f}_{(n+1)T} = \sum_{a=0}^{n} \{P\}^a \bar{R} + \{P\}^{n+1} \bar{f}_0. \tag{3.6}$$

Assuming that $\{P\}$ has N distinct real eigenvalues $1, \gamma_1, \gamma_2, \ldots, \gamma_{N-1}$, where $1 \geq |\gamma_1| \geq |\gamma_2| \geq \ldots \geq |\gamma_{N-1}|$, and using results obtained by Howard (1960, pp. 9-12), an expansion for $\{P\}^a$ can be written as

$$\{P\}^a = \{S\} + \sum_{j=1}^{N-1} \gamma_j^a \{T_j\},$$

where $\{S\}$ has identical rows which represent the stationary probabilities and $\{T_j\}$ are the transient matrices.

New terms are defined in the following way:

$$\{S\}\bar{R} = k \cdot \bar{e}, \quad \{T_j\}\bar{R} = \bar{q}_j; \quad \{S\}\bar{f}_0 = k_0\bar{e}$$

and $\{T_j\}\bar{f}_0 = \bar{r}_j$ where $\bar{e}$ = is a unit column vector.

Thus, k is the generalized yearly gain, while $\bar{q}_j$, and $\bar{r}_j$ are transient error vectors.

Then equation (3.6) is rewritten as:

$$\bar{f}_{(n+1)T} = (\sum_{a=0}^{n} (k\bar{e} + \sum_{j=1}^{N-1} \gamma_j^a \bar{q}_j)) + k_0\bar{e} + \sum_{j=1}^{N-1} \gamma_j^{n+1} \bar{r}_j. \tag{3.7}$$

In a similar manner it is possible to derive an expression for $\bar{f}_{nT}$ as follows:

$$\bar{f}_{nT} = \sum_{a=0}^{n-1} (k\bar{e} + \sum_{j=1}^{N-1} \gamma_j^a \bar{q}_j) + k_0 \bar{e} + \sum_{j=1}^{N-1} \gamma_j^n \bar{r}_j .$$

Defining a new vector $\bar{x}_n$ as:

$$\bar{x}_n = \bar{f}_{(n+1)T} - \bar{f}_{nT} .$$

From the above it follows directly that

$$\bar{x}_n = k\bar{e} + \gamma_1^n (\bar{q}_1 + (\gamma_1 - 1)\bar{r}_1) + \bar{g}(\gamma_2^n, \gamma_3^n, \ldots), \tag{3.8}$$

As $n \to \infty$, it is reasonable, in equation (3.8), to neglect the terms compounded by the eigenvalues of minor dominance and exclusively consider the term with γ_1.

It is clear from equation (3.8) that the sequence $\{\bar{X}_n\}$ will converge geometrically to the limiting value $k\bar{e}$ (by definition the cyclic gain of the process) as $n \to \infty$.

3.2 The New Iterative Scheme

Howard (1960, pp. 40-41) demonstrated that using values for the functional $f_{(n+1)T-t+1}(S_t, i_{t-1})$ obtained by searching over the steady state values for a particular given policy, the new policy derived from this new set is better than the older ones. Such improved values are obtained in Howard's (1960) policy iteration method through the solution of a set of N simultaneous linear equations, where N

is the number of discretized states. It is clear that such procedure is very limitative when the number of states is very large, which is the usual case in operation reservoir problems. However, in light of the results of Section 3.1, it is absolutely clear that it is no longer necessary to do the improved values step by solving a set of simultaneous linear equations. Simply repeatedly using the successive approximation machinery with the policy kept fixed will produce both the yearly gain of the process and the modified benefit function with geometric convergence. It is relevant to note that this machinery is used by the conventional stochastic dynamic approach in any event, so that the programming needed for the improvement procedure is actually a subset of that for the conventional approach.

In fact, these steady state values can be obtained through a routine that is quite similar to that developed for the full maximizing step (equation 2.5) of the conventional approach. However, instead of lookng through all feasible alternatives (releases from the reservoir) for each state, one computes only the cyclic gain of the system and the improved values for the releases fixed. The asymptotic convergence of the cyclic gain of the process guarantees that the improved values for the functional will, in fact, be drawn from the steady state condition.

Moreover, from the author's experience, the number of iterations required to obtain the desired convergence with a maximum error of one percent (1%) is at most two in the early stages when the current policy is far from the optimal, and one iteration in the middle and final stages when the current policy is very close to the optimal.

With these ideas in mind, a combined iterative routine can be sketched. Figure 3.1 shows the proposed algorithm. Here, instead of using the linear diverging variable $f_{(n+1)T-t+1}(S_t, i_{t-1})$ which is inconvenient from a practical point of view, the bounded variable $f'_{(n+1)T-t+1}(S_t, i_{t-1}) = f_{(n+1)T-t+1}(S_t, i_{t-1}) - f_{(n+1)T-t+1}(S'_t, i'_{t-1})$ is preferred, where $f_{(n+1)T-t+1}(s'_t, i'_{t-1})$ refers to the value of the functional for a base state (S'_t, i'_{t-1}).

Step 1 - Starting Procedure: Set $f'_{(n+1)T-t}(S_{t+1}, i_t)$ equal zero for $n = 0$, $t = T$ and considering that $T + 1 = 1$.

Set $n = 0$

Step 2 - Full Optimization Routine: Solve for $t = 12, 11, 10, \ldots, 1$

$$f_{(n+1)T-t+1}(S_t, i_{t-1}) = \underset{r_t}{\mathrm{Max}}(V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) f'_{(n+1)T-t}(S_{t+1}, i_t)),$$

for all (S_t, i_{t-1})

Store r_t^* which is optimal.

Compute Bounded Variables:

$$f'_{(n+1)T}(S_1, i_{12}) = f_{(n+1)T}(S_1, i_{12}) - f_{(n+1)T}(S'_1, i'_{12}),$$

for all (S_1, i_{12}), where (S'_1, i'_{12}) is a base state.

Step 3 - Stopping Procedure: If

$$(|f_{(n+1)T}(S'_1, i'_{12}) - f_{nT}(S'_1, i'_{12})| \leq \xi |f_{(n+1)T}(S'_1, i'_{12})|,$$

where ξ is a small positive number.

If "YES" Stop.

NO

Step 4 - Improved Values Routine: Compute for $t = 12, 11, 10, \ldots, 1$

$$f_{(n+1)T-t+1}(S_t, i_{t-1}) = V_t(r_t) + \sum_{i_t} p(i_t/i_{t-1}) f'_{(n+1)T-t}(S_{t+1}, i_t),$$

using the r_t^* of Step 2.

Update Bounded Variables:

$$f'_{(n+1)T}(S_1, i_{12}) = f_{(n+1)T}(S_i, i_{12}) - f_{(n+1)T}(S'_1, i'_{12}),$$

for all (S_1, i_{12}). Set $n = n + 1$. Go to Step 2.

Figure 3.1. New Iterative Scheme

CHAPTER 4

APPLICATION TO A REAL WORLD PROBLEM

The real system that was chosen with the purpose of testing the new iteration scheme is Marte R. Gomez Reservoir on the San Juan River in Northeastern Mexico.

4.1 Layout of the System

The system is composed of five elements, namely, the river that provides the inflow, the reservoir that stores the water, and three main channels that serve the water requirements downstream. A schematic layout of this system is shown in Figure 4.1. The system is part of a more complex one. The integrated by the set of reservoirs, diverging dams, main channels, flood ways, power plants, etc., that conforms the total Bajo Rio Bravo and Bajo Rio San Juan system.

4.2 Features of the System

The system was constructed in 1943 for irrigation purposes exclusively. The streamflow data correspond to the historical record that was registered in the period 1902-1941 at Santa Rosalia Station, that was located upstream of the dam. In the period 1944-1964 the streamflow data were deduced from the monthly functioning of the dam. The historical record consists of fifty-three years of mean monthly flows, expressed in millions of cubic méters. The mean monthly flows,

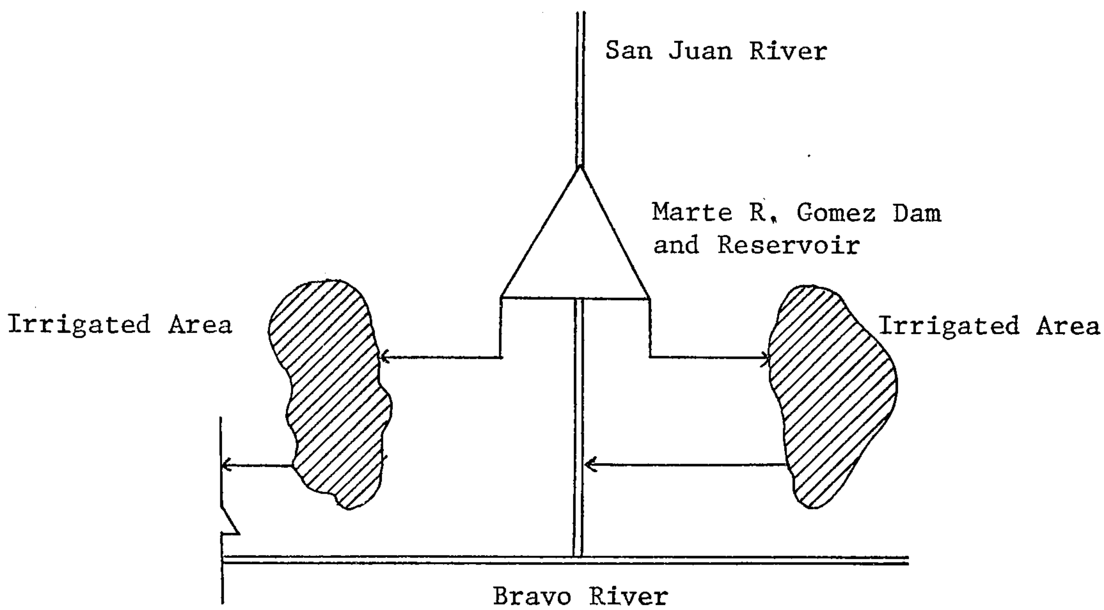


Figure 4.1. Layout of the System

monthly standard deviation, monthly coefficient of skew and the coefficient of correlation between successive months is included in Table 4.1. These statistical parameters were computed from the decimal logarithms of the real inflow data.

Table 4.1 Statistical Parameters of Inflow for each Month at Marte R. Gomez Reservoir.

Month	Coefficient of Skew	Standard Deviation	Mean	Coefficient of Correlation
January	- .87115	.79677	.92345	.62882
February	- .87738	.72186	.80940	.68698
March	- .82655	.62436	.88661	.51639
April	- .72603	.72433	1.12757	.21831
May	-1.19940	.61466	1.53958	.29417
June	- .54641	.69438	1.59177	.16176
July	- .28116	.59656	1.59633	.52752
August	.29834	.75622	1.54891	.45301
September	- .25921	.56365	2.07281	.61131
October	- .17223	.47114	2.00480	.37448
November	-1.01336	.72797	1.35531	.75940
December	-1.01570	.73163	1.16481	.54513

Marte R. Gomez dam is an earth dam of forty-nine meters height. The reservoir has a maximum capacity of 1100 million cubic meters and a dead storage volume of $100 \times 10^6 m^3$ approximately. Evaporation losses are expressed as the average monthly voluem lost and will be considered a constant for a particular month, independently of the year. The monthly evaporation volume losses are included in Table 4.2.

Table 4.2 Monthly Evaporation from Gomez Reservoir

Month	Volume ($10^6 m^3$)
January	9.2
February	10.7
March	15.6
April	18.6
May	20.7
June	22.5
July	35.9
August	22.2
September	18.3
October	16.3
November	11.8
December	9.4

The relase schedule of Gomez Dam is limited by a maximum amount, given by the total capacity of the conduits, totaling $200 \times 10^6 m^3$ per month.

No flood control is provided in Gomez Reservoir, since the spillway is of the free type, i.e., there are no control facilities. Minimum requirements for monthly releases are implicit in the benefit function.

4.3 Optimal Control Policy for Gomez Dam

4.3.1. Net Benefit Function

For the purpose of this study the unit price of water will be considered constant. A monthly target output of $100 \times 10^6 m^3$ is included in the benefit function. Penalty costs are imposed over monthly releases below the monthly target output.

A smooth variation of the net benefit function is obtained by fitting a parabolic equation to three match points given by the value of the function when the release is zero, one hundred and two hundred million cubic meters respectively. Equation (4.1) and Figure 4.2 express these concepts analytically and shematically.

$$V_t(r_t) = 52500 - 1.75(r_t - 200)^2 \qquad (4.1)$$

The returns in different time periods will not be discounted. This case is not covered in this study. However, no substantial

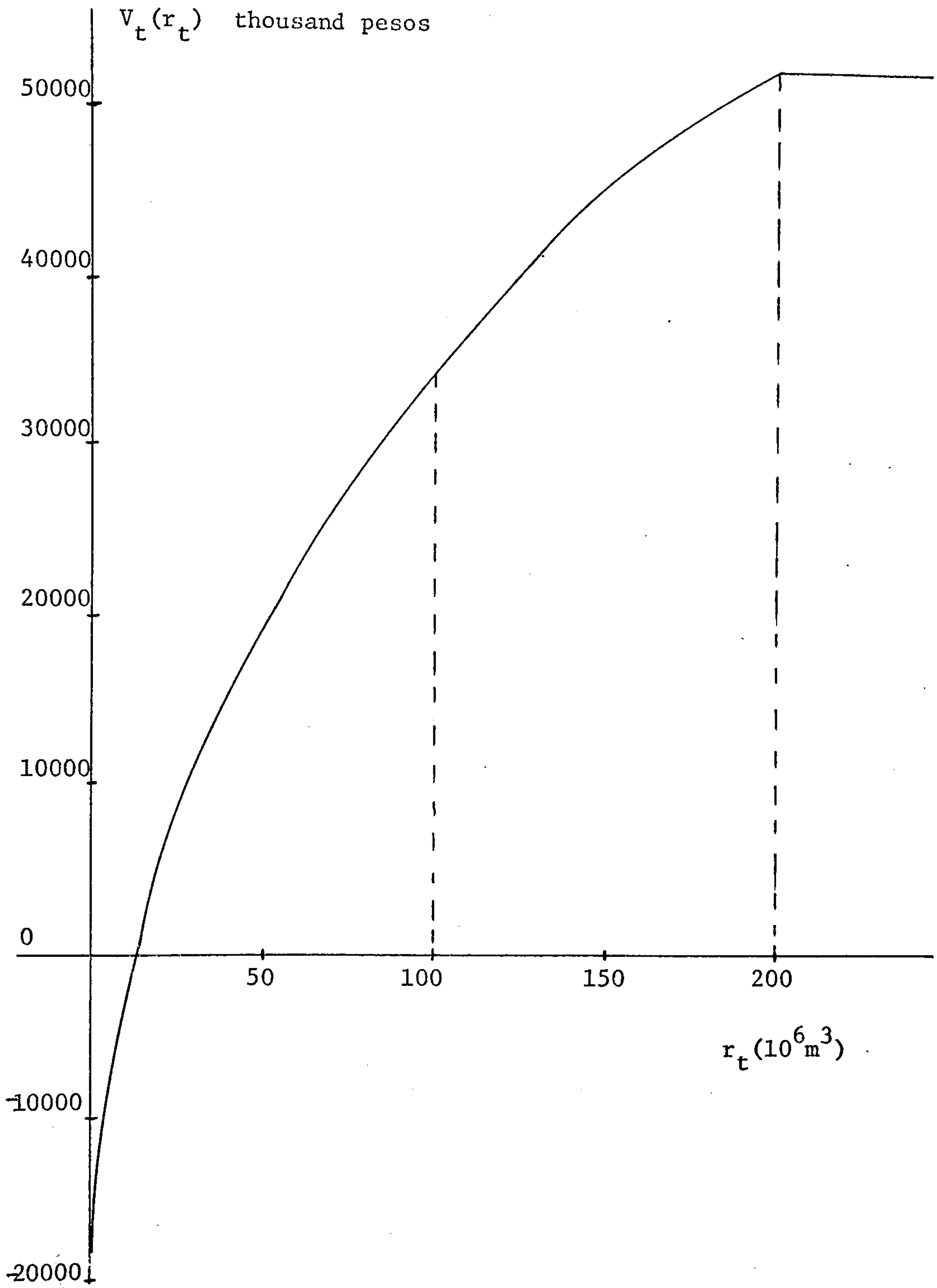


Figure 4.2. Net Benefit Function.

changes have to be made in the mathematical formulation of the process to include it.

4.3.2. Derivation of Periodic Markovian Chains for Depicting the Process

The state transition probabilities for the case study can be derived from the monthly inflow probability distributions (conditional and unconditional). The state transition probabilities can then be computed from these probabilities and the governing equation (equation 2.1). Computationally, only twelve sets of monthly inflow probability distributions must be stored in the computer, and every state transition probability is computed, when it is being used, according to the equations stated in (2.1) and (2.3).

The twelve sets of probabilities for the monthly inflows were derived using a refined technique developed by Clainos (1972) based on a stochastic streamflow synthesis technique.

At each time t the continuous value of each monthly inflow i_t is discretized into a finite number of equal intervals. Each interval is, then represented by the value at its midpoint. This discretization scheme provides a convenient way of indexing the various values of i_t and it is easier to program for the computer.

Using the definition of conditional probability, the conditional probability $P(A/B)$ is desired where $P(A/B) = P(A,B)/P(B)$.

Expressed in streamflows, the conditional probability

$$P(\underline{q}_0 \leq Q_0 \leq \overline{q}_0 / \underline{q}_s \leq Q_s \leq \overline{q}_s)$$

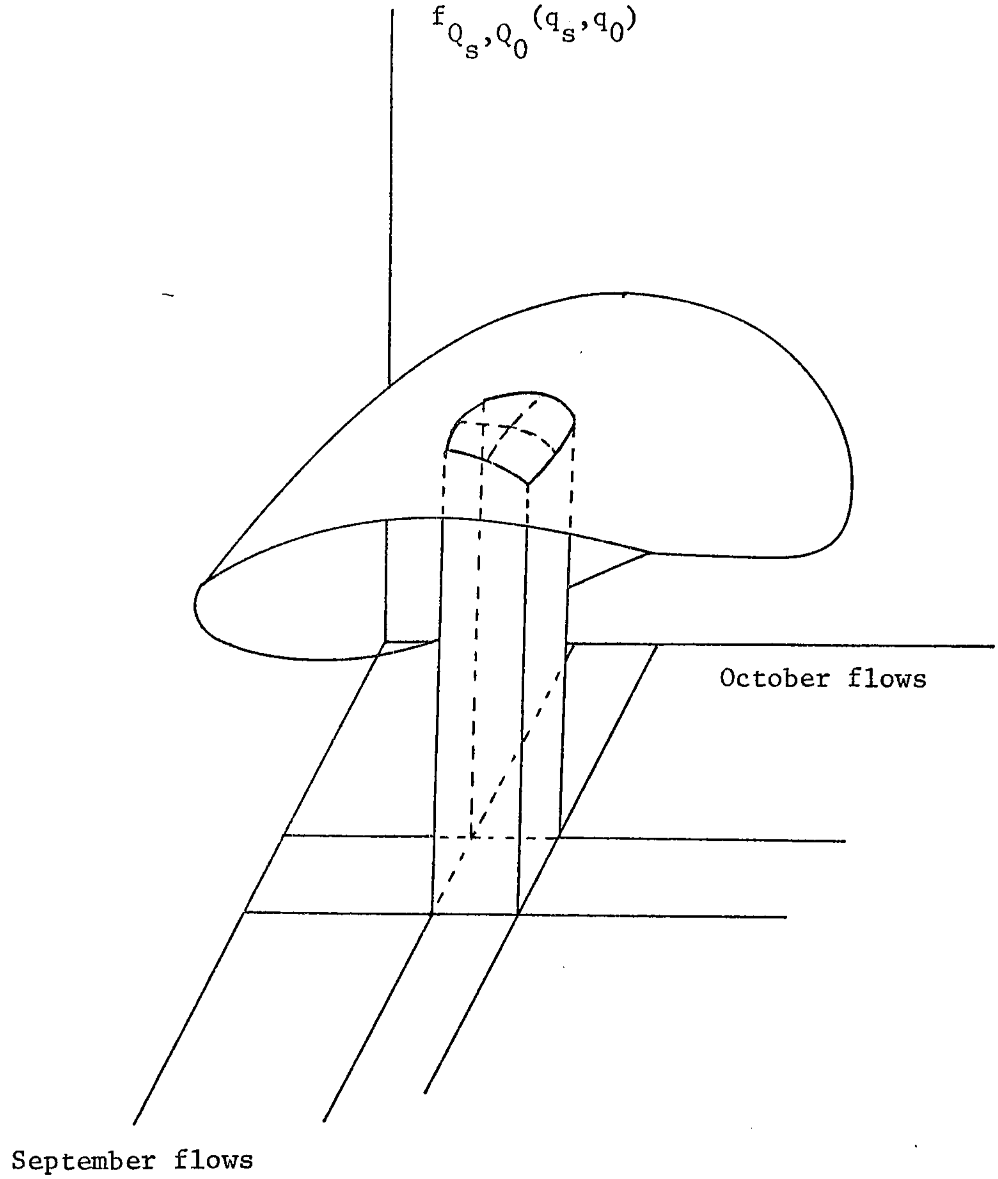
$$= P(\underline{q}_0 \leq Q_0 \leq \overline{q}_0, \underline{q}_s \leq Q_s \leq \overline{q}_s) / P(\underline{q}_s \leq Q_s \leq \overline{q}_s)$$

where $\underline{q}_0$ is the lower limit on the interval of October flows; $\overline{q}_0$ is the upper limit of October flows; $\underline{q}_s$ is the lower limit of September flows, and $\overline{q}_s$ is the upper limit of September flows.

A procedure to determine the unconditional probabilities is given by Clainos (1972):

1) Take the common logarithm of each streamflow.
2) Compute the standardized gamma deviate of the value in Step 1.
3) Transform the standardized gamma deviate of Step 2 to a standardized normal deviate.
4) Then using numerical integration, determine the unconditional probability of each streamflow.

The joint probability is expressed by means of a bivariate normal distribution. Graphically, the joint probability is represented by the shaded volume of Figure 4.3. One possibility to compute this probability is to perform double numerical integration. However, for practical problems the computer time required to derive this joint probability is too expensive. Roefs and Clainos (1971)



$$\text{Volume} = P(Q_s,Q_0) = \iint f_{Q_s,Q_0}(q_s,q_0)\,dq_s\,dq_0$$

Figure 4.3. Joint Density Function.

show that if the flow interval of the previous month was sufficiently small, it could be represented by a single point. If this approximation is made, the computational effort is reduced substantially. A computer program is provided by Clainos (1972) to derive conditional probabilities matrices.

Problems arising from high skewed streamflows $(g \geq 2)$ was studied by Clainos, Roefs, and Duckstein (1973). These problems arise because the Wilson-Hilferty transform is used to transform a Pearson type III deviate to a normal deviate. Clainos et al. (1973) provide a new transform based on Harter's (1969) work. Basically, he computed the percentage points of the chisquare distribution, which he then modified to percentage points of the Pearson III distribution. Clainos et al. (1973) show that this new transform works very well for coefficient of skew as great as three inclusive.

However, in this study the use of the new transform was not necessary since the coefficient of skew for all months are less than $\pm$ 1.5.

Appendix B shows the conditional probabilities dependency matrices derived for each month.

4.4 The Computational Algorithm

A computer program was written in Fortran IV language and the CDC 6400 computer was used to solve the recursive equation (2.5) of Chapter 2 subject to all the constraints specified previously. For the purposes of this study it was found that increments of 100 million cubic meters for the storage, and 10 million cubic meters for the release are appropriate. However, the size increment for the monthly inflow varies considerably depending upon the range of variation of these inflows for a particular month.

In view of the large range of variation of inflow for some months, it was necessary to establish an upper limit of five increments to maintain the problem within the limits of computer time assigned to this study. Consequently, the increments for the state and decision variables are not the same. This fact may lead to the result that during the iterations the next state is not necessarily one of the original discretized states. Hence, some kind of interpolation scheme must be used. The selected scheme for this study was linear interpolation, mostly due to its simplicity. However, since the benefit function is smooth and convex, a second order interpolation polynomial would fit better to the benefit function than the linear approach.

The computer program is composed of two parts. The first one corresponds to the main program, where the data are read and the full optimization step is located. The second part comprises the improved relative values step and it is located in a subroutine which is called every time a full maximizing step is executed. A flow chart of the proposed algorithm is included in Figure 4.3. A listing of the computer program is presented in Appendix A.

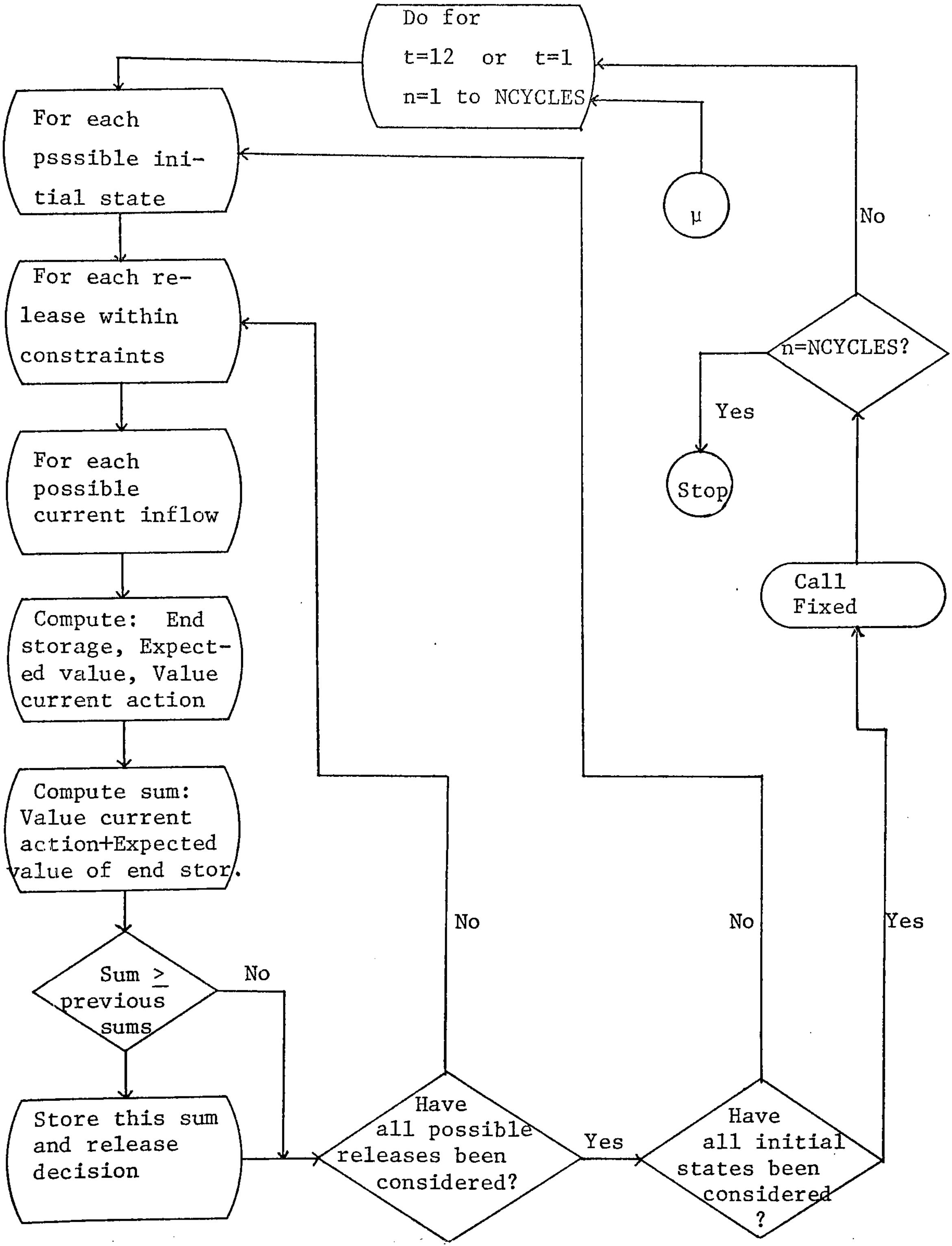


Figure 4.4. Flowchart for the Improved Stochastic Dynamic Programming Algorithm.

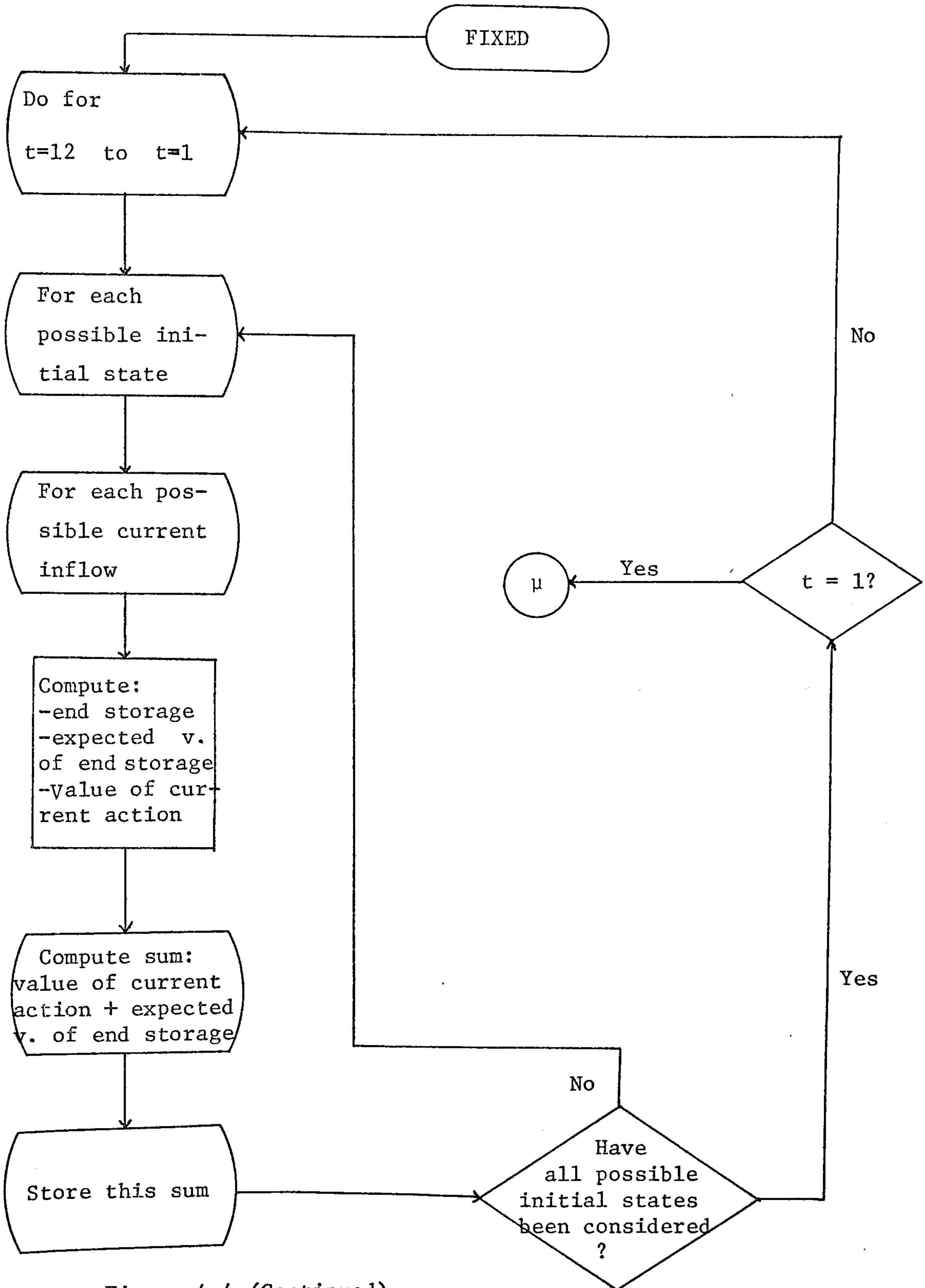


Figure 4.4 (Continued)

4.5 Optimal Release Policy Under the Two Approaches

The conventional stochastic dynamic programming and the new iterative scheme were tested in order to derive quantitative comparisons. As can be observed from the flowchart, the computational algorithm for the conventional approach forms part of the composed algorithm. Simply by avoiding the call to the subroutine FIXED the algorithm for the conventional approach is obtained.

A sample of the optimum policy determination for September is given in Figure 4.5 in graphical form and in Table 4.3 in tabular form.

The expected annual returns for the conventional approach, as determined by the gain of the system, was \$363,594,000 with the optimality criterion ξ being equal to .001 (with a maximal possible error of $100 \times \xi$ percent). This condition was reached after six full iterations and about fifty seconds of central processing time were required. The optimal policy was obtained in the sixth iteration.

The corresponding expected annual return for the new iterative scheme, for the same level of accuracy, was \$363,605,000. This amount is practically identical as that obtained with the conventional approach. Such condition was reached after four full iterations and three fixed policy iterations, requiring thirty-eight seconds of central processing time. The optimal policy was obtained in the fourth full iteration, stopping the algorithm at this point since the level of accuracy in the yearly gain of the system was satisfied and the optimal policy reached.

Table 4.3 Optimal Release Policy for September

Starting Storage	Inflow in August $(10^6 m^3)$				
$10^6 m^3$	150	450	750	1050	1350
100	70	80	80	90	90
200	80	90	100	100	100
300	90	100	100	110	110
400	100	110	110	110	120
500	110	120	130	130	130
600	120	130	130	130	130
700	130	130	130	140	140
800	130	140	140	140	140
900	140	150	160	160	160
1000	150	160	160	160	170
1100	150	160	160	170	170

It is important to notice that the decreasing computer time would be proportionally greater if the number of discrete values of the decision variable were increased. In this particular application, only twenty discrete values of release were analyzed; in this case it was necessary to expend approximately two seconds in the execution of the fixed policy routine and eight seconds for the full iterative step. However, in a more refined model where the number of discrete values were three or four times the number of discrete values used here, the fixed policy routine will remain costing exclusively two seconds per iteration as compared with the full maximizing routine which will be increased at least three or four times more.

For this particular application a saving of approximately twenty-five percent of central processing time was obtained. However, based on the arguments of the above paragraph this saving might jump to fifty percent.

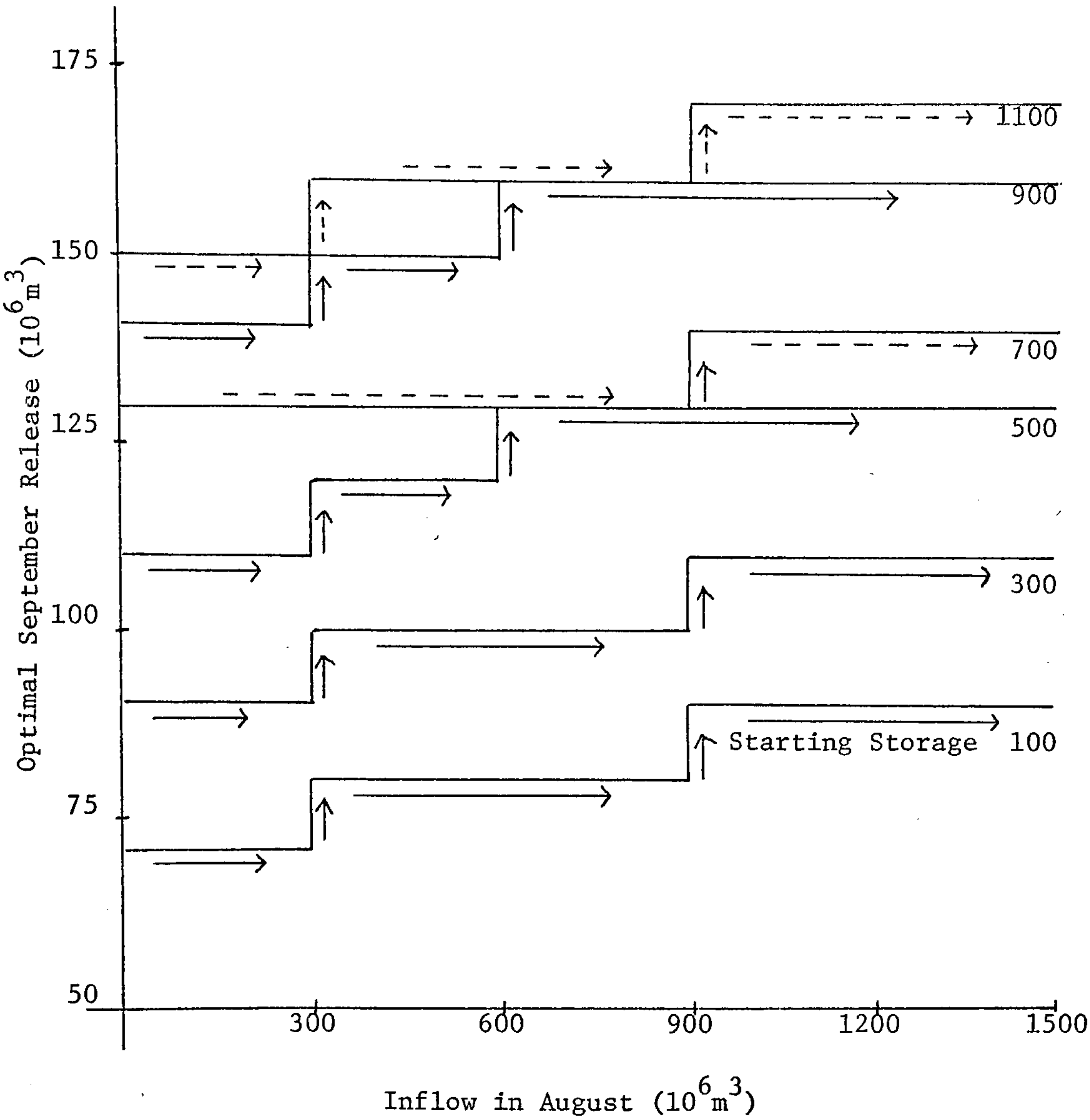


Figure 4.5. Optimal Release Control Policy for September.

CHAPTER 5

CONCLUSIONS AND DISCUSSION

The operating policies for a single stochastic reservoir system are commonly optimized by conventional dynamic programming with the use of high speed digital computers. However, this method usually encounters two difficulties that constrain its general application, namely, the untractable large computer memory requirement and the excessive computer time requirement. The new iterative scheme can, at least, ease the latter difficulty considerably. By examining only the steady state values, those produced for a fixed policy after the execution of each full maximizing step, an improvement in the values of the functional used in the subsequent policy determinations is obtained, increasing the chance of finding the optimal policy more rapidly. Such improved values are obtained by simple repeated use of the successive approximation machinery with the policy kept fixed. The procedure will produce the improved values with geometric convergence.

The time requirements with the proposed algorithm are reduced by, at least, twenty-five percent (25%) of the time required with the conventional approach. Moreover, if the number of discrete values for the decision variable is increased, the computer time saved might be raised proportionally up to fifty percent (50%). This saving will have direct effect on the feasibility to deal with more sophisticated systems that in the past have been untractable by the traditional technique.

At this point, it is important to mention that a general drawback which is shared by all the stochastic dynamic programming approaches to derive optimum operating policies is directly related to the absence of control over their probability of failure. The technique is designed to maximize expected net benefits, and in doing so, it may derive a policy that despite its economic advantages, will allow the system to fail more frequently than the operating authority is willing to accept. A further improvement of the algorithm presented is possible by introducing a technique developed by Askew (1974) that permits the derivation of the optimal policy subject to a constraint on the probability of failure. The technique consists basically in the imposition of an extra penalty cost in the recursive equation to be maximized.

APPENDIX A

CONDITIONAL PROBABILITIES DEPENDENCY MATRICES

FOR MARTE R. GOMEZ DAM

Table A.1 Conditional Probabilities Dependency Matrices for each Month at Marte R. Gomez Dam.

	20	60	100	140	180
			January to February		
20*	.89	.10	.01	.00	.00
60	.73	.21	.05	.01	.00
100	.59	.29	.09	.02	.01
140	.48	.34	.13	.04	.01
180	.38	.36	.18	.06	.02

	20	60	100	140	180
			February to March		
20	.87	.12	.01	.00	.00
60	.59	.33	.07	.01	.00
100	.35	.44	.17	.04	.00
140	.19	.43	.27	.09	.02
180	.08	.34	.35	.18	.05

	60	180	300	420	540
			March to April		
20	.91	.07	.01	.01	.00
60	.75	.18	.05	.01	.01
100	.60	.27	.09	.03	.01
140	.44	.32	.15	.06	.03
180	.29	.33	.21	.11	.06

*The values of the left axis are the discrete points of monthly inflow given in million cubic meters.

Table A.1 Conditional Probabilities Dependency Matrices for each Month at Marte R. Gomez Dam. (Continued)

	30	90	150	210	270
			April to May		
60	.51	.25	.14	.07	.03
180	.43	.27	.17	.09	.04
300	.38	.28	.18	.11	.05
420	.34	.28	.19	.12	.07
540	.32	.27	.20	.13	.08
	90	270	450	630	810
			May to June		
30	.85	.10	.03	.01	.01
90	.79	.13	.05	.02	.01
150	.74	.15	.06	.03	.02
210	.69	.18	.07	.04	.02
270	.63	.20	.09	.05	.03
	40	120	200	280	360
			June to July		
90	.69	.18	.07	.04	.02
270	.64	.20	.08	.05	.03
450	.62	.21	.09	.05	.03
630	.59	.21	.10	.06	.04
810	.55	.22	.11	.07	.05

Table A.1 Conditional Probabilities Dependency Matrices for each Month at Marte R. Gomez Dam. (Continued)

	150	450	750	1050	1350
			July to August		
40	.93	.04	.02	.01	.00
120	.85	.09	.03	.02	.01
200	.79	.12	.05	.02	.02
280	.74	.15	.06	.03	.02
360	.71	.15	.07	.04	.03
	150	450	750	1050	1350
			August to November		
150	.66	.21	.08	.03	.02
450	.56	.25	.11	.05	.03
750	.51	.27	.12	.06	.04
1050	.47	.28	.13	.07	.05
1350	.43	.29	.14	.08	.06
	150	450	750	1050	1350
			September to October		
150	.88	.10	.02	.00	.00
450	.68	.23	.06	.02	.01
750	.56	.29	.10	.04	.01
1050	.47	.32	.13	.05	.03
1350	.40	.37	.15	.07	.03

Table A.1 Conditional Probabilities Dependency Matrices for each Month at Marte R. Gomez Dam. (Continued)

	40	120	200	280	360
			October to November		
150	.73	.18	.06	.02	.01
450	.58	.24	.11	.05	.02
750	.50	.26	.14	.07	.03
1050	.45	.27	.16	.08	.04
1350	.40	.28	.17	.10	.05
	30	90	150	2.0	270
			November to December		
40	.85	.13	.02	.00	.00
120	.51	.35	.11	.03	.00
200	.27	.40	.23	.08	.02
280	.13	.34	.30	.17	.06
360	.05	.23	.33	.26	.13
	20	60	100	140	180
			December to January		
30	.78	.14	.05	.02	.01
90	.59	.23	.11	.05	.02
150	.45	.27	.15	.08	.05
210	.34	.28	.19	.12	.07
270	.24	.27	.22	.16	.11

APPENDIX B

COMPUTER PROGRAM

```
      PROGRAM STOICO (INPUT,OUTPUT)
C***************************************************************
C*        IMPROVED STOCHASTIC DYNAMIC PROGRAMMING              *
C*           OPTIMAL OPERATION POLICY                          *
C*           MARTE R. GOMEZ DAM,MEXICO                         *
C************************************************
      REAL INF
      COMMON P(5,5,12),ST(11),INF(5,12),E(12),FTP1(11,5),
     1FT(11,5),FASTRI(11,5),RASTRI(12,11,5),N,SMAX
C---------------------------------------------------------------
C     NOTATION.
C     P(I,J,K),TRANSITION PROBABILITY MATRIX
C     K,IDENTIFY A PARTICULAR PAIR OF MONTHS.
C     J,IDENTIFY THE DISCRETE VALUES OF INFLOW
C     FOR PREVIOUS MONTH.
C     I,IDENTIFY THE DISCRETE VALUES OF INFLOW
C     FOR CURRENT MONTH.
C     ST,CONTAINS THE DISCRETE VALUES OF STORAGE(STATES)
C     INF(I,J),CONTAINS THE DISCRETE VALUES OF INFLOW
C     J,IDENTIFY THE MONTH AND I A PARTICULAR VALUE.
C     E(I),CONTAINS AMOUNTS OF EVAPORATION PER MONTH.
C     DRT(I),CONTAINS FOR EACH MONTH THE INCREMENT DESIRED
C     FOR THE DECISION VARIABLE.
C     NYEARS,NUMBER OF CYCLES TO BE RUN THE MODEL.
C     SMAX,UPPER LIMIT FOR STORAGE(CAPACITY OF THE DAM)
C     SMIN,LOWER LIMIT OF STORAGE PERMISSIBLE.
C---------------------------------------------------------------
C**********************************************
C     READ   DATA
C**********************************************
      READ 500,(((P(I,J,K),I=1,5),J=1,5),K=1,12)
      READ 501,(ST(I),I=1,11)
      READ 502,((INF(I,J),I=1,5),J=1,12)
      READ 503,(E(I),I=1,12)
      READ 505,NYEARS,SMAX,SMIN,DRT,EPSILO
C**********************************************
C     INITIALIZE VALUES
C**********************************************
      DO 10 I=1,11
      DO 10 J=1,5
      FTP1(I,J)=0.0
      FT(I,J)=0.0
      FASTRI(I,J)=-999999999.9
   10 CONTINUE
      FTP1(11,5)=0.0
      ALPHA=FTP1(11,5)
C   ********************************************
C     DO FOR NUMBER OF YEARS AND TIME PERIODS
C   ********************************************
      DO 70 ITT=1,NYEARS
      PRINT 507
      DO 50 IT=1,12
C**********************************************
C     SET A FEASIBLE STATE
C     SET AN INITIAL STORAGE
C**********************************************
      DO 90 IST=1,11
```

```
      PRINT 508
      ***********************************************
      SET A PREVIOUS INFLOW
      ***********************************************
      DO 20 ITM1=1,5
      ***********************************************
      SET A FEASIBLE RELEASE
      ***********************************************
      DO 30 IRT=1,21
      IRTT=IRT-1
      RT=IRTT*DRT
      PROD=0.0
      ***********************************************
      FOR EACH CURRENTE INFLOW COMPUTE=
      (1) END STORAGE
      (2) EXPECTED VALUE OF END STORAGE
      (3) VALUE OF CURRENT ACTION
      ***********************************************
      DO 40 IIT=1,5
      STP1=ST(IST)+INF(IIT,IT)-RT-E(IT)
      IF(STP1.GT.SMAX) 305,396
  305 STP1=SMAX
      GO TO 397
  396 IF(STP1.LT.SMIN) GO TO 31
  397 ENT=STP1/100.
      NIND=INT(ENT)
      NINDI=NIND+1
      FX=FTP1(NIND,IIT)+(ENT-NIND)*(FTP1(NINDI,IIT)-
     1FTP1(NIND,IIT))
      PROD=PROD+P(IIT,ITM1,IT)*FX
   40 CONTINUE
      VR=52500.0-1.75*(RT-200.0)**2
      ***********************************************
      COMPUTE SUM OF VALUE OF CURRENT ACTION
      PLUS EXPECTED VALUE OF END STORAGE
      ***********************************************
      FT(IST,ITM1)=VR+PROD
      ***********************************************
      IS THIS SUM GREATER THAN THE PREVIOUS
      SUM COMPUTED
      YES,STORE THIS SUM AND RELEASE DECISION
      NO,DEFINE  A NEW RELEASE
      ***********************************************
      IF(FT(IST,ITM1).GT.FASTRI(IST,ITM1)) 1274,30
 1274 FASTRI(IST,ITM1)=FT(IST,ITM1)
      RASTRI(IT,IST,ITM1)=RT
   30 CONTINUE
   31 PRINT 506,IST,ITM1,FASTRI(IST,ITM1),
     1RASTRI(IT,IST,ITM1),IT
   20 CONTINUE
      ***********************************************
      HAVE ALL POSSIBLE INITIAL STATES BEEN ANALYZED
      ***********************************************
   90 CONTINUE
      DO 782 IST=1,11
      DO 782 ITM1=1,5
      FTP1(IST,ITM1)=FASTRI(IST,ITM1)
```

```
      FT(IST,ITM1)=0.0
      FASTRI(IST,ITM1)=-999999999.9
  782 CONTINUE
   50 CONTINUE
      ****************************************
      HAVE OPTIMIZATION BEEN REACHED
      ****************************************
      IF(ABS(FTP1(11,5)-ALPHA).LE.EPSILO *ABS(FTP1(11,5)))STOP
      DO 60 IST=1,11
      DO 60 ITM1=1,5
      FTP1(IST,ITM1)=FTP1(IST,ITM1)-FTP1(11,5)
   60 CONTINUE
      IF(ITT.EQ.1.OR.ITT.EQ.2) 61,62
      ****************************************
      ENTER SUBROUTINE FIXED
      ****************************************
   61 CALL FIXED
   62 ALPHA=FTP1(11,5)
   70 CONTINUE
  500 FORMAT(5F10.2)
  501 FORMAT(8F10.2)
  502 FORMAT(5F10.2)
  503 FORMAT(6F10.0)
  504 FORMAT(6F10.0)
  505 FORMAT(I10,5F10.0)
  506 FORMAT(10X,*IST =*,I5,5X,*ITM1 =*,I5,5X,*FASTRI =*,
     1F10.0,5X,*RASTRI =*,F10.0,5X,*MONTH =*,I5)
  507 FORMAT(1H1)
  508 FORMAT(1H )
      END
```

```
      SUBROUTINE FIXED
      REAL INF
      COMMON P(5,5,12),ST(11),INF(5,12),E(12),FTP1(11,5),
     1FT(11,5),FASTRI(11,5),RASTRI(12,11,5),N,SMAX
      NCYCLES=1
      DO 70 ITT=1,NCYCLES
      DO 50 IT=1,12
      ******************************************
      SET A FEASIBLE STATE
      SET AN INITIAL STORAGE
      ******************************************
      DO 90 IST=1,11
      ******************************************
      SET A PREVIOUS INFLOW
      ******************************************
      DO 20 ITM1=1,5
      PROD=0.
      ******************************************
      FOR EACH CURRENTE INFLOW COMPUTE=
      (1) END STORAGE
      (2) EXPECTED VALUE OF END STORAGE
      (3) VALUE OF CURRENT ACTION
      MAKING USE OF THE SAME RELEASE DECISION
      AS OBTAINED PREVIOUSLY
      ******************************************
      DO 40 IIT=1,5
      STP1=ST(IST)+INF(IIT,IT)-RASTRI(IT,IST,ITM1)-E(IT)
      IF(STP1.GT.SMAX) 305,397
  305 STP1=SMAX
  397 ENT=STP1/100.
      NIND=INT(ENT)
      NINDI=NIND+1
      FX=FTP1(NIND,IIT)+(ENT-NIND)*(FTP1(NINDI,IIT)-
     1FTP1(NIND,IIT))
      ******************************************
      COMPUTE SUM OF VALUE OF CURRENT ACTION
      PLUS EXPECTED VALUE OF END STORAGE
      ******************************************
      PROD=PROD+P(IIT,ITM1,IT)*FX
   40 CONTINUE
      VR=52500.-1.75*(RASTRI(IT,IST,ITM1)-200.)**2
      FT(IST,ITM1)=VR+PROD
      PRINT 506,IST,ITM1,FT(IST,ITM1),RASTRI(IT,IST,ITM1),IT
   20 CONTINUE
      ******************************************
      HAVE ALL POSSIBLE INITIAL STATES BEEN ANALYZED
      ******************************************
   90 CONTINUE
      DO 782 IST=1,11
      DO 782 ITM1=1,5
      FTP1(IST,ITM1)=FT(IST,ITM1)
      FT(IST,ITM1)=0.0
  782 CONTINUE
   50 CONTINUE
      DO 60 IST=1,11
      DO 60 ITM1=1,5
      FTP1(IST,ITM1)=FTP1(IST,ITM1)-FTP1(11,5)
```

```
 60 CONTINUE
 70 CONTINUE
506 FORMAT(10X,*IST =*,I5,5X,*ITM1 =*,I5,5X,*FASTRI =*,
   1F10.0,5X,*RASTRI =*,F10.0,5X,*MONTH =*,I5)
    RETURN
    END
```

REFERENCES

Askew, J. A., "Optimum Reservoir Operating Policies and the Imposition of a Reliability Constraint", Water Resources Research, Vol. 10, No. 1, pp. 51-56, 1974.

Bellman, R. E., Dynamic Programming, Princeton University Press, 1957.

Clainos, D. M., Input Specifications to Stochastic Decision Models, M.S. Thesis, University of Arizona, Tucson, 1972.

Clainos, D. M., Roefs, T. G., Duckstein, L. "Creation of Conditional Dependency Matrices Based on a Stochastic Streamflow Synthesis Technique", Water Resources Research, Vol. 9, No. 2, pp. 481-485, 1973.

Harter, H. L., "A New Table of Percentage Points of the Pearson Type 3 Distributions", Technometrics, Vol. 11 (1), pp. 177-187, 1969.

Hassitt, A., "Solution of the Stochastic Programming Model of Reservoir Regulation", Report No. 320-3506, 18 pp., IBM Washington Scientific Center, 1968.

Howard, R. A., Dynamic Programming and Markov Processes, M.I.T. Press, Cambridge, Mass., 1960.

Loucks, D. P., "Stochastic Methods for Analyzing River Basin Systems", Technical Report No. 16, Cornell University, Water Resources and Marine Sciences Center, Ithaca, N. Y., pp. V1-V21, 1969.

Loucks, D. P. and Falkson, L. M., "Comparison of some Dynamic, Linear, and Policy Iteration Methods for Reservoir Operation", Water Resources Bulletin, Vol. 6, No. 3, pp. 384-400, 1970.

MacQueen, J., "A Modified Dynamic Programming Method for Markovian Decision Problems", J. Math. Anal. Appl., Vol. 14, pp. 38-43, 1966.

Morton, T. E., "On the Asymptotic Convergence of Cost Differences for Markovian Decision Process", Operations Research, Vol. 19, pp. 244-249, 1971.

Odoni, A., "On Finding the Maximal Gain for Markov Decision Processes", Operations Research, Vol. 17, pp. 857-860, 1969.

Roefs, T. G., and Clainos, D. M., "Conditional Streamflow Probability Distributions", in Hydrology and Water Resources in Arizona and the Southwest, Vol. 1, pp. 153-170, American Water Resources Association, Tucson, Arizona, 1971.

Roefs, T. G., Personal Communication, University of Arizona, Tucson, 1974.

Schweitzer, P. J., "Perturbation Theory and Markovian Decision Processes", M.I.T. Operations Research Center, Technical Report No. 15, 1965.

Su, S. Y., and Deininger, R. A., "Generalization of White's Method of Successive Approximations to Periodic Markovian Decision Processes", Operations Research, Vol. 20, No. 2, pp. 318-326, 1972.

Vleugels, R. A., "Optimal Operation of Serially-Linked Water Reservoirs", Contribution No. 138, University of California Water Resources Center, Los Angeles, Calif., 54 pp. 1972.

White, D. J., "Dynamic Programming, Markov Chains and the Method of Successive Approximations", J. Math. Anal. Appl., Vol. 6, pp. 373-376, 1963.