OPTIMAL OPERATION OF RESERVOIRS USING SYSTEM ANALYSIS

(Optimization of Hydropower Output from A 5-Reservoir System by Incremental Dynamic Programming)

DISSERTATION
submitted in partial fulfilment of requirements for
ME (WRD) Degree of the University of Roorkee

By
SRI B. K. BANERJEE

WATER RESOURCES DEVELOPMENT TRAINING CENTRE
UNIVERSITY OF ROORKEE
ROORKEE
1975

C E R T I F I C A T E

CERTIFIED that the dissertation entitled "OPTIMAL OPERATION OF RESERVOIRS USING SYSTEMS ANALYSIS" which is being submitted by SRI B. K. BANERJEE in partial fulfilment for the award of the degree of MASTER OF ENGINEERING in WATER RESOURCES DEVELOPMENT of the University of Roorkee is a record of the candidates own work carried out by him under my supervision and guidance. The matter embodied in this dissertation has not been submitted for the award of any other degree or diploma.

This is further to certify that he has worked from Ist October, 1974 to 30th June, 1975 for preparing this dissertation for Master of Engineering of this University.

(Hari Krishna)
Professor
Water Resources Development
Training Centre,
University of Roorkee, ROORKEE

ROORKEE:

Dated: 17.9.75.

A C K N O W L E D G E - M E N T

The author is grateful to Prof. Harikrishna for his help and guidance in preparing this dissertation.

C O N T E N T S

LIST OF FIGURES

List of Tables

List of Tables

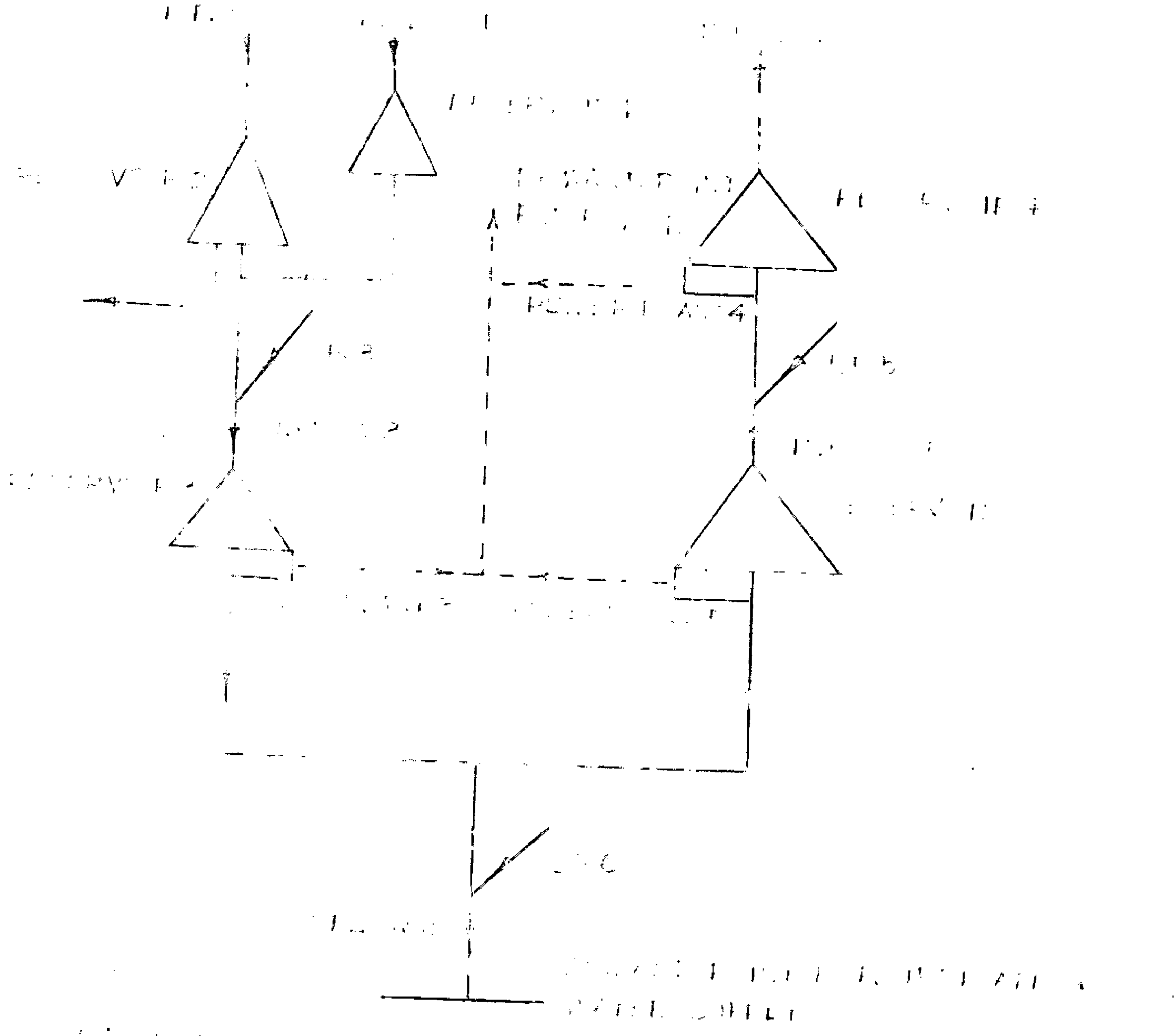
INTRODUCTION :

This study is meant to develop a model to find out the optimal operating policy for a multipurpose, multi-reservoir water resources system. The objective is to maximize hydro-power output from a five-reservoir system with firm water and flood control as parameters and the design of the system defined by the storage capacity, power plant characteristics etc. as fixed constants. The hydrology used in this study is deterministic and is defined by the critical period. The model is applied to the DVC system to obtain optimal joint operation policy, for all the five reservoirs, namely, Konar, Tenughat, Panchet Hill, Tilaiya and Maithon,which will maximize the hydro-power production from the system with certain constraints imposed over it.

Since the problem involves a sequential decision process, dynamic programming technique (1,2) can be suitably applied. The conventional procedure, however, calls for a large number of computations involving a great deal of computer time and storage requirement. This difficulty has been significantly reduced by a new approach known as State Incremental Dynamic Programming (3). This technique has been used by Yeh and Trott (4) in optimizing capacity specifications for water resource systems and by Hall etal (6) in optimizing firm power output from a two reservoir system, Shasta and Folsom in the USA. The present study is an application of this technique to a simplified five-reservoir system for a given deterministic hydrological input.

Since the installed capacity of the hydro-power plants in the DVC system is only a small fraction of the total installed capacity including thermal and hydro-electric power, the entire continuous hydro-power generation has been considered as firm power. This treatment simplifies the system to an isolated hydro-power system with interconnected power plants operating to supply energy to the power grid which always has the demand for whatever power can be generated from the hydro-power plants.

Energy losses in transmission lines have been ignored. No constraint has been imposed on the source of energy which goes to satisfy the demand.



[illegible]

[illegible]

THE D.V.C SYSTEM :

Figure 1 shows the schematic representation of the DVC water resources system with its five reservoirs, namely, Konar, Tenughat, Panchet Hill, Tilaiya and Maithon numbered in that order. Three of these have variable head power plants with a total installed capacity of 104 MW. These are connected to the DVC grid which has three thermal power plants with a total plant capacity of over 1000 MW. This, in turn, is connected to the Rihand, Hirakud, West Bengal, Bihar and Assam grids.

There are two main parallel water-systems, - one on River Damodar and the other on its principal tributary, Barakar. On River Damodar, the upstream reservoir is at Tenughat (Reservoir 2) which supplies water to the Bokaro Steel Complex. Another reservoir (Reservoir 1) on River Konar which is also a tributary to River Damodar supplies cooling water for Bokaro Thermal Power Station. The releases from Tenughat and Konar, after meeting the consumptive requirements for industrial and domestic use, enter the downstream reservoir (Reservoir 3) at Panchet Hill. The present withdrawal for consumptive use between Konar and Panchet Hill is 3.03 cumecs(107CFS) only, but projected figure of 7.08 cumecs (250 CFS) has been adopted. The consumptive requirements for the steel complex has been taken as 8.50 cumecs (300 CFS).

Whereas Konar and Tenughat have no power plants at present, Panchet Hill has a hydel power station with an installed capacity of 40 MW.

On River Barakar, the upstream reservoir (Reservoir 4) is at Tilaiya with a small power plant having an installed capacity of only 4 MW. The releases from this reservoir enter the downstream reservoir (Reservoir 5) at Maithon which has a power plant with 60 MW of installed capacity.

The demand point for the irrigation water supply of the combined system is at Durgapur Barrage which is located downstream of the confluence of the two rivers, Damodar and Barakar. The firm water contract upto this point consists of requirements of Kharif irrigation of 393676 hectares (973000 ACRES) and Rabi irrigation for 22253 hectares (55000 ACRES) of land, industrial and domestic requirements for the region below the two upstream dams, mandatory release of 2.83 cumecs (100 CFS) below Durgapur Barrage and navigational requirements during the non-monsoon season. Nominal enroute losses have been considered during the dry periods only.

Data for the reservoirs, power plants and different water requirements are shown in Tables 1-9 and 18-19.

DYNAMIC PROGRAMMING:

The three types of variables forming system equations are the stage variables, the state variables and the decision or control variables. The stage variables determine the order of occurence of events in the system. These are the monthly time periods over which the system is operated and defined as the sequence k = 1,2, ...K starting from the first day of June when the reservoirs have been taken at their minimum storage levels.

The state variables are Si(i=1, 2, 3, 4 or 5) which is the amount of water in storage in each of the five reservoirs at the beginning of each monthly period. The decision variables are Di, the amount of water in storage at the end of each monthly period.

Another set of decision variables can be Ri, the releases from each reservoir during each monthly time period. These will be completely defined by the initial storage, the end storage, inflow and evaporation losses which can be considered as a function of initial and final storages for each monthly time period.

Taking Si as the state variable and Di as the decision variable for Reservoir i, the state transformation equations can be written as :

$$Si_{k+1} = Di_k$$

$$(i = 1, 2, 3, 4 \text{ or } 5)$$

$$(k = 1, 2, \ldots\ldots K)$$

Where :

Si_k is the water in storage in Reservoir i at the beginning of month k,

Di_k is the water in storage in Reservoir i at the end of month k.

If Ri_k is the release from Reservoir i during period k, then :

$$Ri_k = Si_k - Di_k - Ei_k + Ii_k$$

Where : Ei_k = Net evaporation losses from Reservoir i deducting direct precipitation over it during period k.

Ii_k = Net inflows into Reservoir i during period k.

Evaporation losses can be expressed as :

$$Ei_k = EVARi_k \cdot AREAi \frac{(Si_k + Di_k)}{2}$$

Where :

$EVARi_k$ = The rate of evaporation from Reservoir i during period k.

AREAi = A function giving the area of Reservoir i as a function of average storage during any period.

Keeping in view that only Reservoirs 3, 4 and 5 have power plants, the objective function can be written as :

$$\sum_{1}^{5} UUi = \max \sum_{1}^{K} \left[PRj_k \cdot ERj \left(\frac{Sj_k + Dj_k}{2} \right) \right]$$

$$j = 3, 4, 5$$

Where :

PRj_k is the release through power plant j during period k.

ERj is a function that gives the energy production rate for power plant j as a function of average storage in Reservoir i during any period.

The different constraints are :

(1) Energy Capacity Constraint

$$0 \leq Uj_k = \left(PRj_k, Ri_k, \frac{Sj_k + Dj_k}{2} \right)$$

$$\leq ENMAXj_k \left(PREMj_k, \frac{Sj_k + Dj_k}{2} \right)$$

Where :

Uj_k = A function giving energy produced through the operation of power plant j during period k.

$ENMAXj_k$ = A function giving maximum energy that can be generated at power plant j during period k.

Also, $0 \leq PRj_k \leq PREMj_k$

Where :

$PREMj_k$ is the maximum release that can be made through power plant j during period k.

$$ENMAXj_k = PRHRS_k \cdot CAPAj \left(\frac{Sj_k + Dj_k}{2} \right)$$

Where :

$PRHRS_k$ = Number of hours available during period k for the production of firm energy.

CAPAj = A function that gives the maximum capacity at which energy can be generated at power plant j in relation to the average storage $\left(\frac{Sj_k + Dj_k}{2} \right)$ in any period k.

(2) Reservoir Capacity Constraint:

$$SMINi \leqslant Si_{k+1} \leqslant SMAXi_k$$

Where : $SMINi$ = The minimum storage level in Reservoir i.

$SMAXi_k$ = The maximum permissible storage level in Reservoir i during period k, from flood control point of view.

(3) Water Availability Constraint :

$$COREi_k \leqslant Ri_k \leqslant Si_k + Ii_k - Ei_k - SMINi$$

Where :

$COREi_k$ = Compulsory release below Reservoir i during period k.

(4) Continuity Constraint :

$$Di_k = Si_k + Ii_k - Ri_k - Ei_k$$

(5) Downstream Requirement Constraint :

$$DRMIN_k \leqslant R3_k + R5_k + UR6_k \leqslant DRMAX_k$$

Where : $DRMIN_k$ = The minimum downstream requirement during period k.

$DRMAX_k$ = The minimum permissible flow through the downstream channel during period k.

$R3_k$ = Release from Reservoir 3 during period k

$R5_k$ = Release from Reservoir 5 during period k

$UR6_k$ = Runoff from the catchment between Reservoirs 3 & 5 and Durgapur Barrage during period k.

The recursive equation for maximizing annual firm energy from a reservoir with a power plant can be written as :

$$f_{k+1}(Sj_{k+1}) = \max_{Dj_k} \left[Uj_k + f_k \; (Sj_k) \right]$$

Where : $f_k \; (Sj_k)$ = Optimum energy generation at power plant from operation of Reservoir j upto month k following an optimal policy to arrive at state Sj_k in month k.

Uj_k = Amount of energy generated at power plant j by operation of Reservoir j during month k

$$= PRj_k \, . \, ERj \; \left(\frac{Sj_k + Dj_k}{2}\right) .$$

Since there are five reservoirs 1,2,3,4 and 5 with power plants at Reservoirs 3, 4 and 5 only the recursive equation for the system will reduce to

$$f_{k+1}\ (S1_{k+1},\ S2_{k+1},\ S3_{k+1},\ S4_{k+1},\ S5_{k+1})$$
$$= \underset{\substack{D1_k \\ D2_k \\ D3_k \\ D4_k \\ D5_k}}{\text{Max}} \left[U3_k + U45_k + f_k\ (S1_k,\ S2_k,\ S3_k,\ S4_k,\ S5_k\ ) \right]$$

(k = 1, 2, ...K

and $S1_1$ is taken at the minimum storage level).

Where :

$$U45_k = U4_k + U5_k$$

SUCCESSIVE APPROXIMATIONS :

The recursive equation stated above has five state variables and five control variables. Problems of such high dimensionality can be simplified by successive approximations (4) which will reduce the system to a number of subsystems of lower dimensionality inter-related in space and time. In this method the sequence of optimizations over the subsystems is made to converge back to the original one. With this technique the solution of an n - dimensional dynamic programming problem can be broken down to the solution of a sequence of one - dimensional subproblems. The advantage is that the requirements of computational time and storage vary linearly as n rather than exponentially.

Although the convergence to the true optimum cannot be guaranteed in all cases, it can be shown that convergence is always monotonic and for certain classes of problems, convergence to the true optimum can be rigorously proved.(5).

In this method, an initial feasible solution is first assumed for all the reservoirs. The initial policy, which is the sequence of states through which the system must pass, will specify the storages at the beginning and end of each monthly time period from which the total firm energy output from the system can be computed. After determining the initial policy the problem is decomposed into a number of subproblems, considering each subproblem separately where state of only one reservoir is changed at a time for optimization while the sequence of states in the other reservoirs is held fixed. The process is repeated and iterative study is made.

In this study, when the states of an upstream reservoir are being varied, the inflows into the downstream reservoirs and the outflows from them will vary but their states will be kept constant. Incremental dynamic programming is applied over the reservoirs starting with an initial policy which may or may not be optimal. This technique will pick up a new sequence of states which is a fixed increment above or below the initial sequence of states and is either better than or atleast equal to the original one.(6) This becomes the initial sequence of states for the reservoirs for the next iteration.

In our problem, first incremental dynamic programming has been applied to Reservoir 1 while keeping the states in the other reservoirs unchanged. This modifies the initial policy in this reservoir and the inflows and releases for Reservoir 3 and gives a better or atleast an equal value of the objective function.

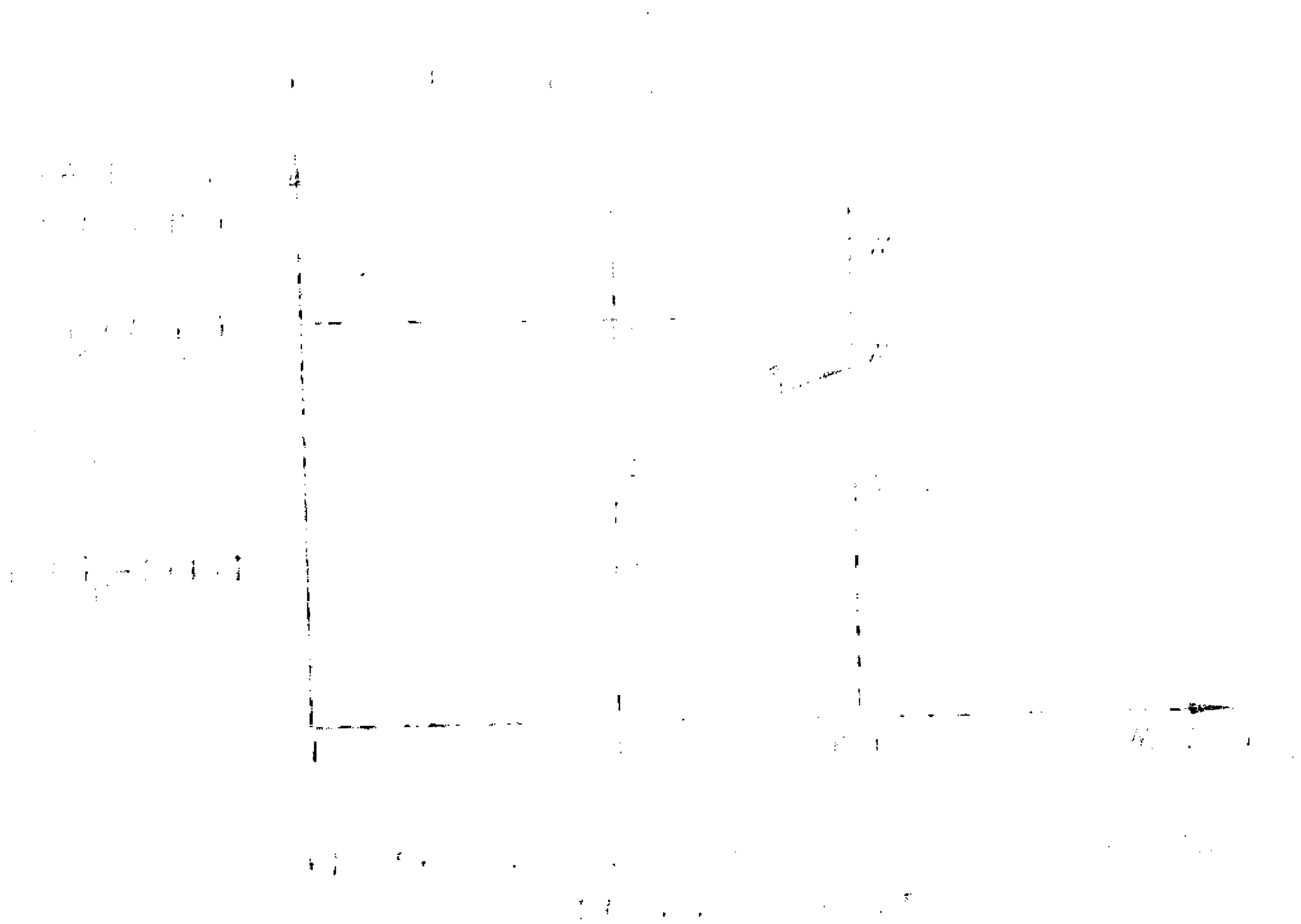
The process passes on to Reservoirs 2, 3, 4 and 5 and then, again, it comes back to Reservoir 1. This is continued until there is no further increase in the firm energy output irrespective of the reservoir selected. This final operation policy gives the optimum firm power for the system.

INCREMENTAL DYNAMIC PROGRAMMING ALGORITHM:

The incremental dynamic programming technique is an iterative procedure which keeps on replacing the initial policy chosen in each iteration and solving the problem repeatedly over and over again until convergency is obtained. The initial policy must be a feasible policy which means that the sequence of states in each monthly period must be such that all constraints imposed over the system are satisfied. Starting with such an initial policy for any reservoir, search for a new policy is conducted in order to obtain either a better value of the objective function or atleast equal to it. The new policy to be analyzed is kept only a fixed amount above or below the original policy or at the same state as the original policy itself. The new policy is next considered to be the original policy for the next iteration and the procedure is repeated. When the value of the objective function converges, no change in the policy produces ~~only~~ any farther increase in the value of the objective function.

COMPUTATIONAL PROCEDURE :

The state and decision variables for any reservoir during any period k are allowed to vary by a fixed amount of deviation



from the initial policy. This can be represented by :

$$Si_k = STORi_k + (I-2).\ DELTi,\ I = 1,\ 2 \text{ or } 3$$

$$Di_k = Si_{k+1} = STORi_{k+1} + (M-2).\ DELTi,\ M = 1,2 \text{ or } 3$$

Where :

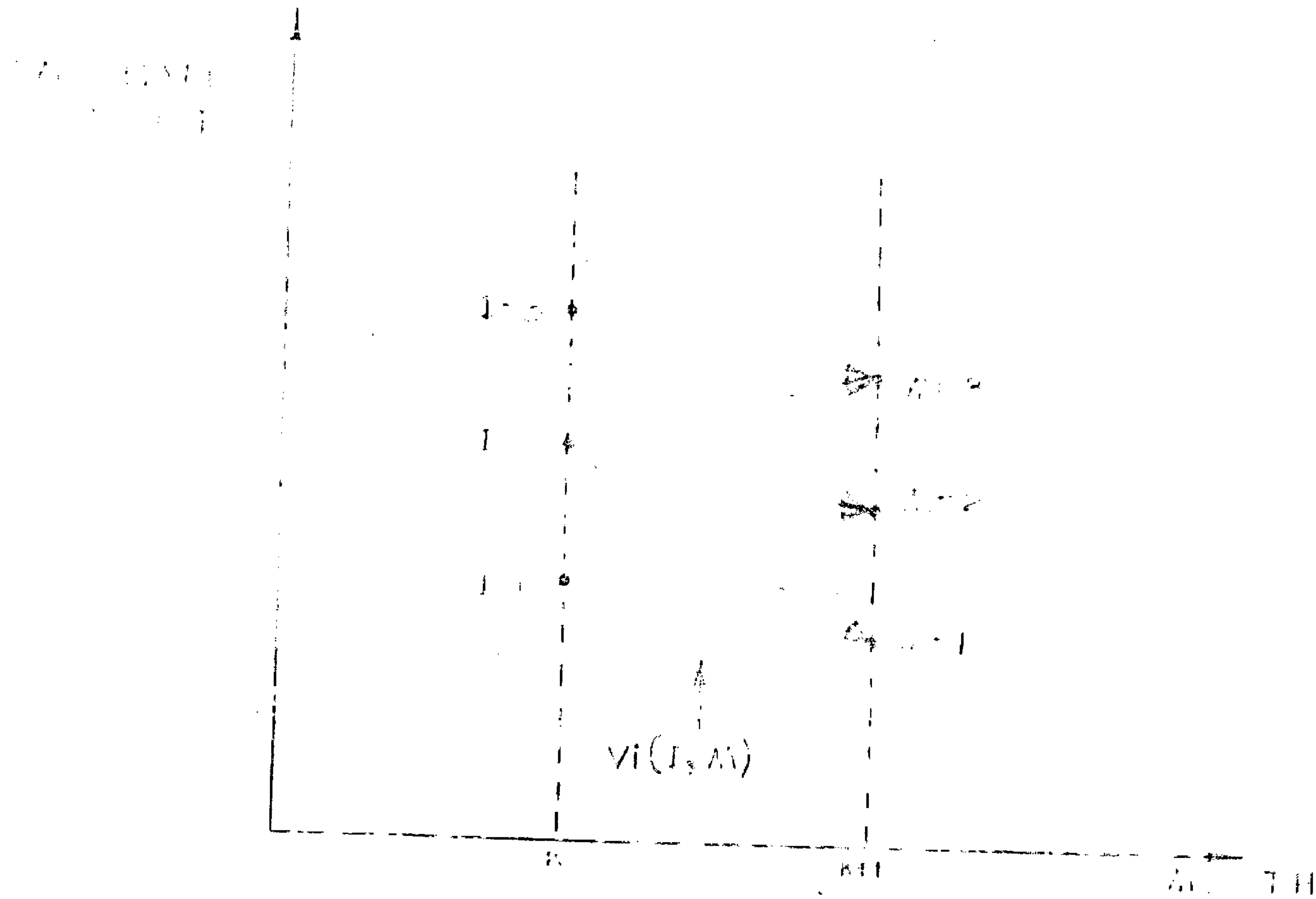
$STORi_k$ = The storage in Reservoir i in the beginning of period k as per initial feasible policy.

DELTi = The amount of deviation allowed from the initial policy in Reservoir i.

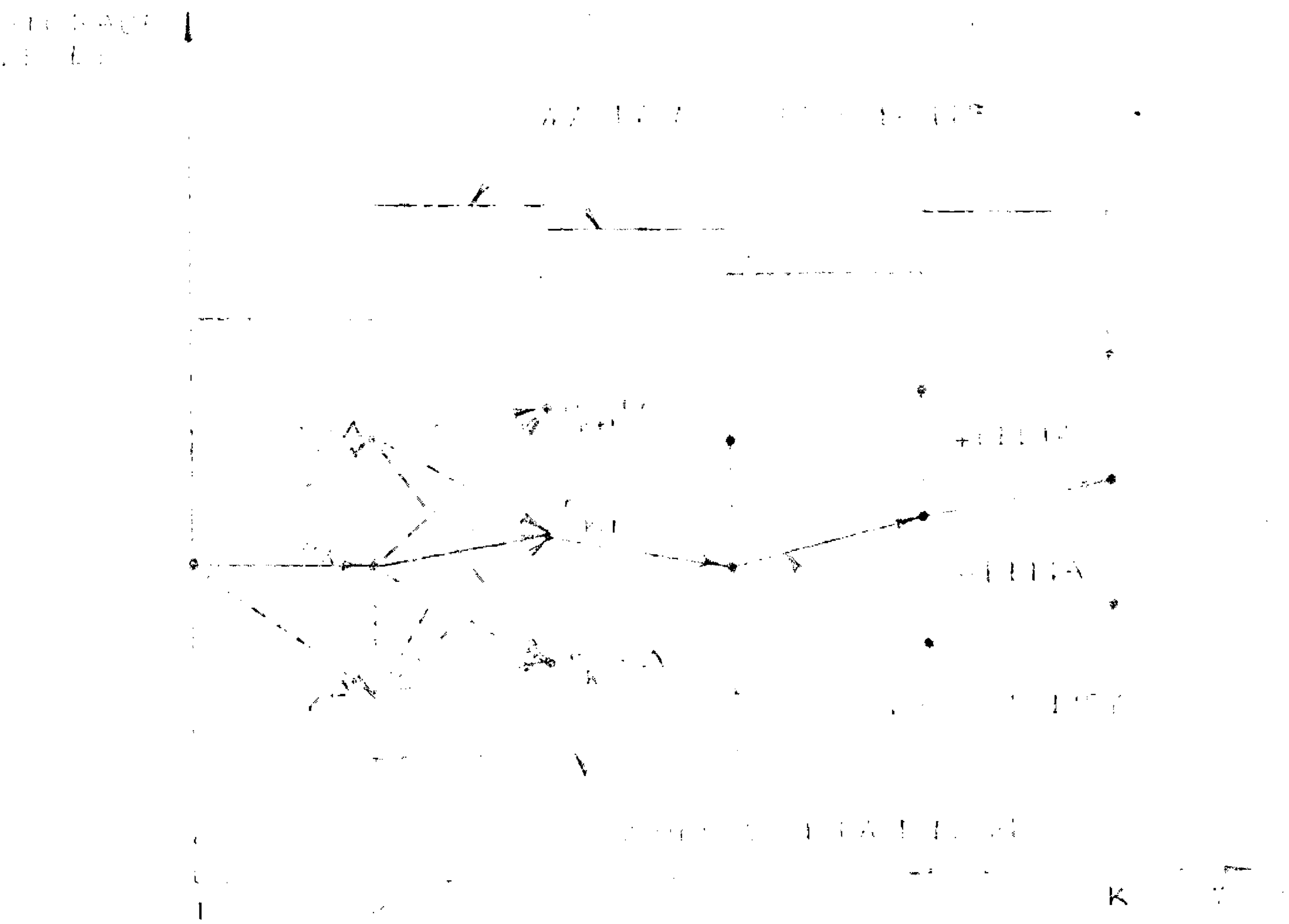
Here the state variable is represented by (I) and the decision variable by (M). A graphical representation of this is shown in Figure 2. It can be seen from this figure that the initial policy is defined by the state I = 2 and the decision M=2 in each month. It can also be seen that there are only three possible states for a reservoir in each month and for each of these three states, there are only three possible decisions. In all, there are nine possible paths, (1,1), (1,2), (1,3); (2,1), (2,2), (2,3);(3,1), (3,2), (3,3). Corresponding to these nine paths for each month k, there are only nine different values of the variable Uj_k.

Controls are applied to find the value of the objective function corrosponding to each sequence of states. If a policy is found infeasible, a penalty value of (-)'999999 is assigned to it in order that it may get eliminated when the best out of these policies is searched.

The computation starts with a fixed initial value of storage at the beginning of the first time period. Three different values of the decision variable are analysed and the values of the objective function UUi(I), I = 1, 2, 3 are calculated. This has been illustrated in Figure 3.



[illegible]
PERIODS OTHER THAN THE FIRST.



With the three values of UUi(I) corrosponding to I = 1, 2 and 3 the next stage is entered and the matrix of firm energy corrosponding to beginning state I and ending state M are calculated. This is expressed as Vi (I,M) and is represented by Figure -4. Adding UUi(I) and Vi(I,M) together three sets of values are obtained corresponding to M= 1,2 and 3. The best out of each set is picked up and is redesignated as UUi(I), I = 1,2,3 upto stage 3 to represent the energy generated through the operation of Reservoir i upto month 3 with states corresponding to I = 1,2 and 3. This process is repeated upto the last stage and the best value of UUi is picked up. The subscripts I and M are used to trace back the optimum path.

A representation of the system is shown in Figure 5 and a computer program for a five-reservoir system in FORTRAN II is enclosed in Appendix D.

APPLICATION OF THE MODEL:

The model has been applied to the DVC system comprising of five reservoirs on River Damodar and its tributaries, Konar and Barakar. The inflow record for all the five reservoirs and the downstream barrage which is the demand point is for a period of 10 years between June 1956 and March 1966. The initial period for all the reservoirs is defined by the year starting June 1962. The canal requirements for paddy during the monsoon of 1962 has been estimated on the basis of consumptive use, percolation loss and effective precipitation. Canal and outlet efficiencies have been adopted as 70 and 85 percent respectively.(7)

The minimum storage levels for the reservoirs are:

6044 hectare metres (49 KAF) for Konar
20970 hectare metres (170 KAF) for Tenughat
17146 hectare metres (139 KAF) for Panchet Hill
7523 Hectare metres (61 KAF) for Tilaiya and
16776 hectare metres (136 KAF) for Maithon.

Maximum storage levels for different months have been defined by the flood operation policy.(8)

Other relavent data have been given in Tables 1-6.

INITIAL POLICY :

The initial policy for each reservoir must be a feasible policy considering the entire system. Also, the policy should be as near optimal as possible so that convergence is obtained with the minimum number of iterations.

For the upstream reservoirs, Konar, Tenughat and Tilaiya initial policy was obtained for the given firm water contracts by a simple mass balance. The state at the beginning of the first stage was taken at the minimum level. Release during any period was equated to the firm water contract for that period. A maximum possible value of evaporation loss was estimated. The decision variable was computed as :

$$Di_k = Si_k + Ii_k - Ei_k - Ri_k$$

If Di_k became more than $SMAXi_k$, it was reduced to $SMAXi_k$ by increasing Ri_k. If it fell below SMINi, it was increased to SMINi by decreasing Ri_k. A computer program in FORTRAN II is enclosed in Appendix A.

It was found out from this that Konar Reservoir (Reservoir 1) failed to deliver the prescribed amount of water during the critical period. The firm water contract for this reservoir had to be redefined as shown in Table 21. Since sufficient water was available from Tenughat Reservoir (Reservoir 2) to meet the consumptive use requirements upto Panchet Hill, the release policy of this reservoir was also redefined.(Tables 20 and 21).

The initial policy of Tilaiya Reservoir, which has a power plant with it, was refined by incremental dynamic programming considering this as an isolated single reservoir system. A computer program in FORTRAN II for optimization of energy generation relevent to this system is enclosed in Appendix B. The refined initial policy has been shown in Table 22.

After deciding the initial policies in the three upstream reservoirs, the inflows into the downstream reservoirs at Panchet Hill (Reservoir 3) and Maithon (Reservoir 5) and the initial policies were determined by hand calculation satisfying the constraints due to continuity, storage capacity, water availability and downstream requirements. This is shown in Tables 25 and 26.

These initial policies were refined by incremental dynamic programming technique applied to optimization of power in single reservoir systems with combined operational constraints. This has been shown in Table 27.

A computer program for this computation is enclosed in Appendix C.

OVERALL OPTIMIZATION :

Since Konar Reservoir (Reservoir 1) was just able to meet the redefined firm water contract, any change in its initial policy would have resulted in an infeasible policy. For this reason, the states in this reservoir were always held constant while incrementing the states of the other reservoirs in a successive rotation. The computer program for optimization of operations in a five-reservoir system shown in Appendix D was slightly modified by removing the operations relevant to Reservoir 1. Thus, successive approximations were conducted through reservoirs 2, 3, 4 and 5 while keeping the initial policy of Reservoir 1 unchanged. The inputs and outputs have been shown in Appendices E and F respectively.

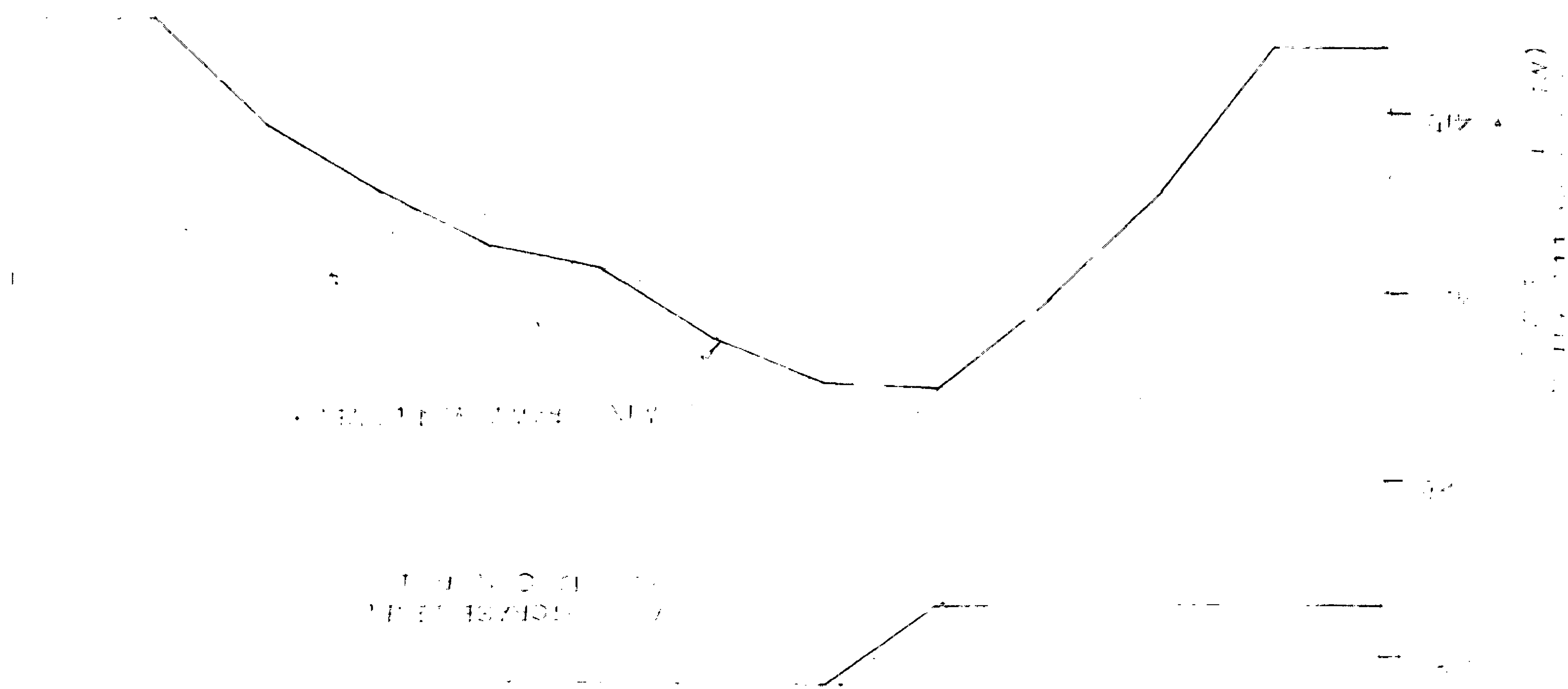
RESULTS AND CONCLUSION :

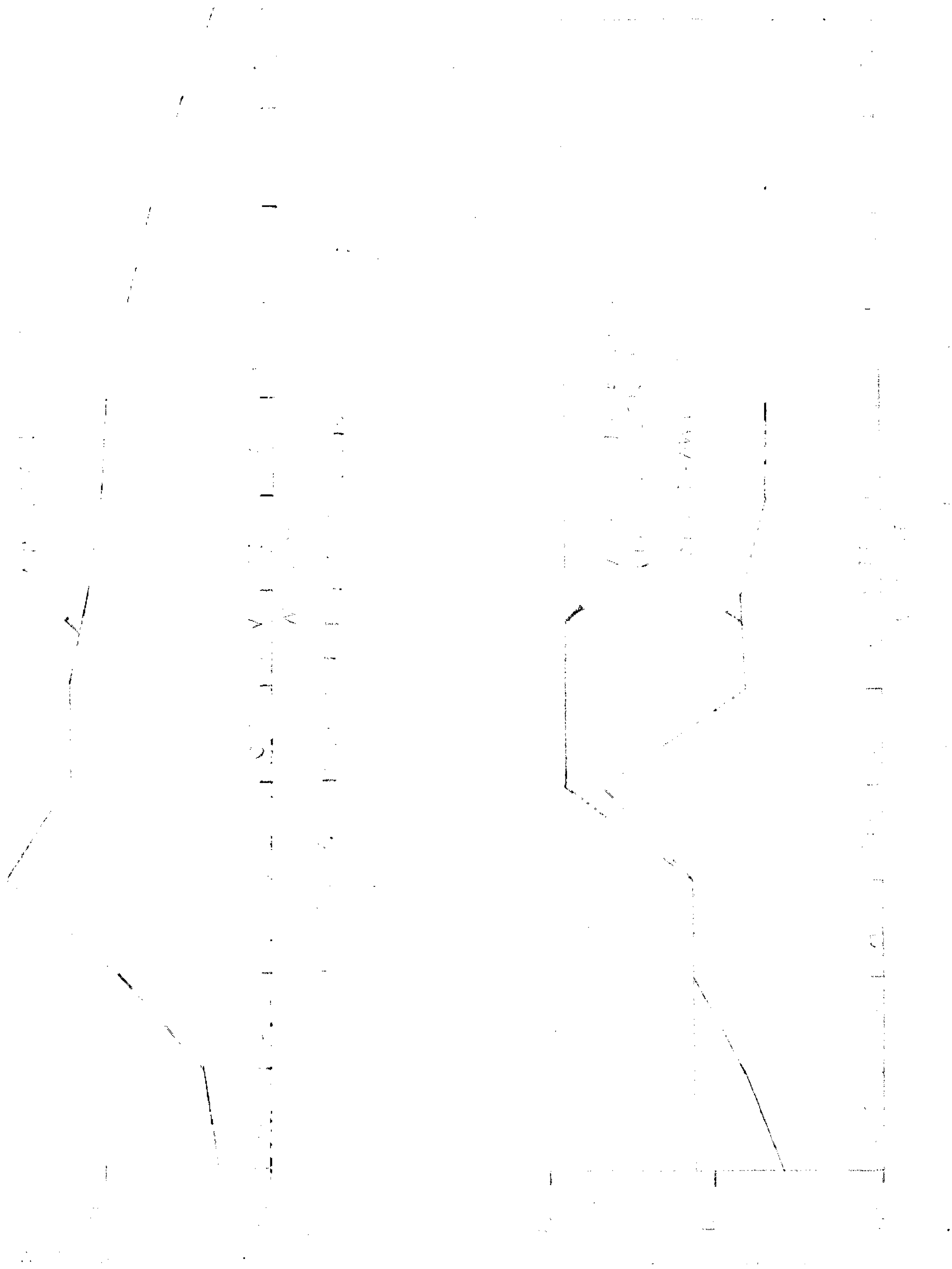
The total firm power output from the system converged to 248701.78 MWH/YEAR in 14 iterations. This took about 40 minutes time in an IBM - 1620 machine.

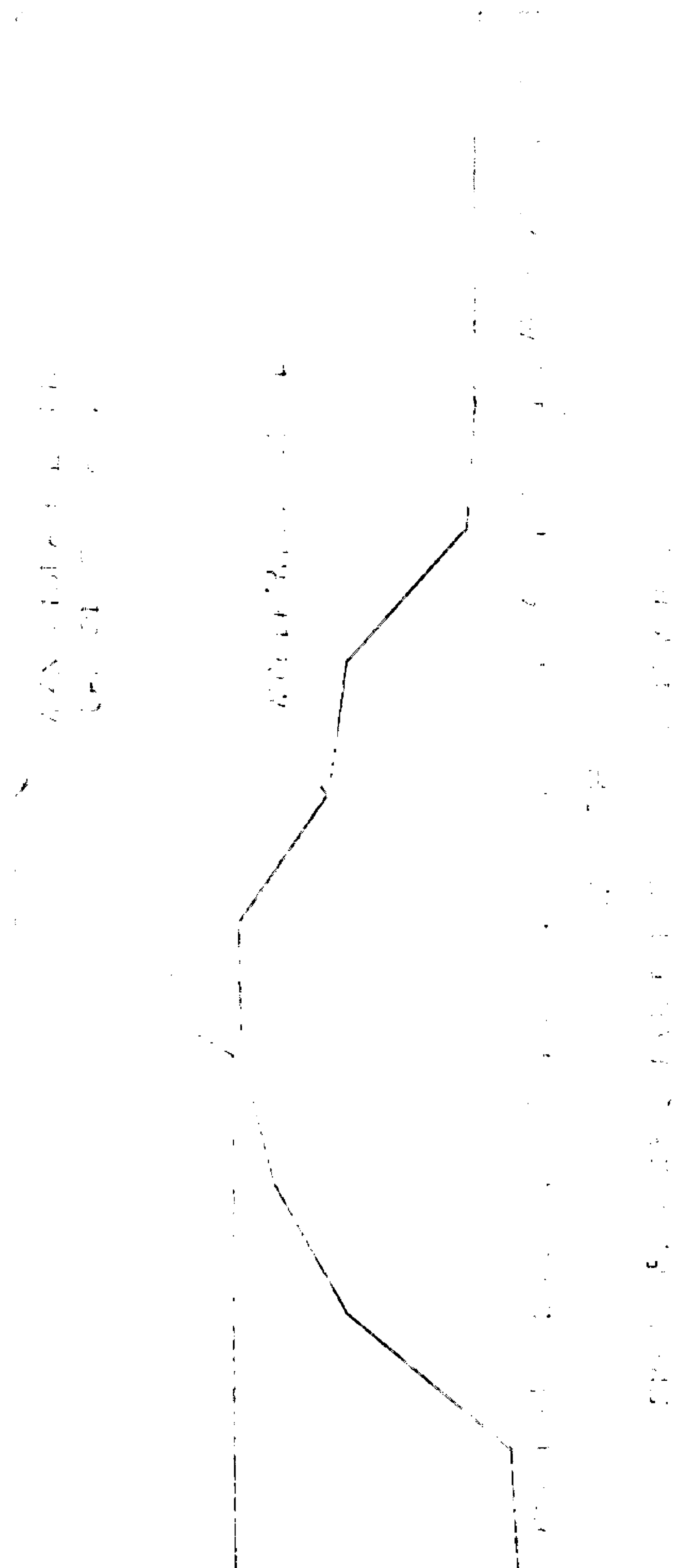
From the states of different reservoirs corresponding to this, the rule curves for the operation of the different reservoirs shown in Figures 6 -10 were constructed.

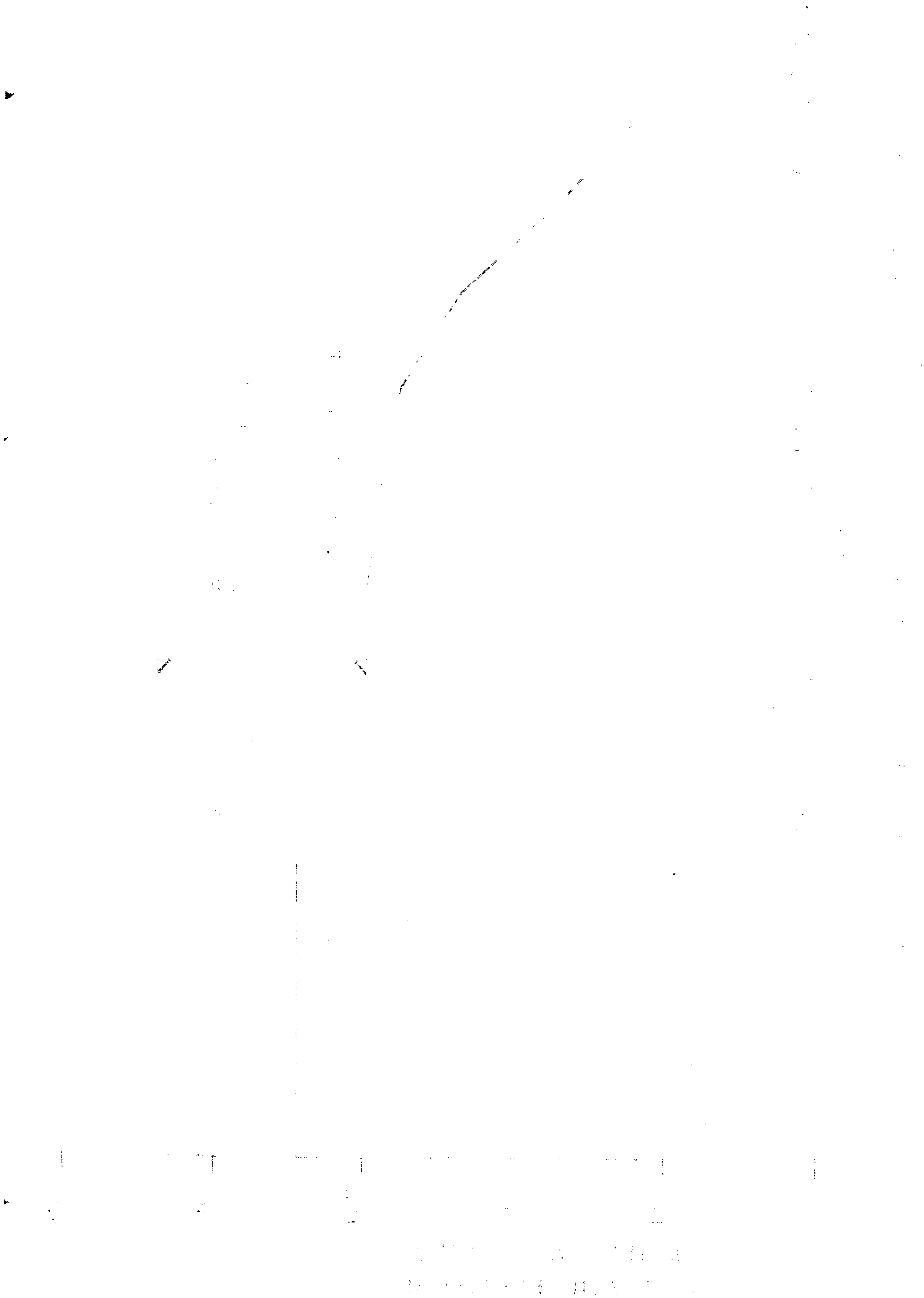
The firm power output at the end of different iterations is shown in Table 29. This shows an increase of about 20 percent over the initial policy which was defined by single reservoir optimizations.

This study was a simplified system analysis with assumptions already stated. Further refinement can be obtained by treating the problem in a stochastic environment rather than deterministically. Stochastic problems in water resources development cannot, however, be solved satisfactorily at present time, the principal reason being that it is difficult to define suitable measures for risks and uncertainty and introduce them into the objective function(9).









Further, this study was confined to the optimization of power output only, treating irrigation, flood control, navigation etc. as fixed parameters. A comprehensive study must consider each of these items and optimize for overall effect.

Further refinement in power output optimization can be obtained by considering the entire DVC power system, or even the entire grid jointly with the hydel generation system.

REFERENCES :

1. Bellman, R., Dynamic Programming, Princeton University Press, Princeton, N.J., 1957.

2. Bellman, R., and S. Dreyfus, Applied Dynamic Programming, Princeton University Press, Princeton, N.J., 1962.

3. Larson, R.E., State Incremental Dynamic Programming, American Elsevier Publishing Company, Inc., New York, 1968

4. Yeh, W.W-G. and W.J. Trott, Optimization of Water Resources Development : Optimization of Capacity Specifications for Components of Regional, Complex integrated, Multipurpose Water Resources Systems, (Submitted) Water Resources Research, U.S. Department of the Interior, Washington, D.C., 1972

5. Korsak, A.J. and R.E. Larson, Convergence Proofs for a Dynamic Programming Successive Appoxomation Technique, (Submitted) Fourth IFAC Congress, Warsaw, Poland, 1969

6. Hall, W. A., R.C. Harboe, W.W.-G. Yeh and A. J. Askew, Optimum Firm Power Output from a Two Reservoir System by Incremental Dynamic Programming, Water Resources centre Contribution No. 130, University of California, Los Angeles, California,1969

7. Note on the Water Requirement of crops for DVC Projects, Water Management Division, Ministry of Food and Agriculture, Govt. of India.

8. Regulation Manual for Damodar Valley Reservoirs, C.W. and P.C., Govt. of India, 1969

9. Windsor, J.S. and V.T. Chow, Multi Reservoir Optimization Model, Journal Hydraulics Division, Proc. A.S.C.E., 1972

10. Chinnappa, K. M., Maithon Underground Hydro-Electric Power Station, Indian Journal of Power and River Valley Development, Maithon Dam Special, 1957.

11. Deb, S. B., Panchet Hill Hydro Electric Power Station, Indian Journal of Power and River Valley Development, DVC Special No., 1959

12. Hydrological Data for Damodar Basin, D.V.C, Calcutta, India.

TABLE - 1.

CHARACTERISTICS OF KONAR RESERVOIR

Storage (KAF)	Area (KA)
0	0
50	2.0
100	3.3
150	4.3
200	5.2
250	6.0
300	6.8

TABLE - 2.

CHARACTERISTICS OF TENUGHAT RESERVOIR

Storage (KAF)	Area (KA)	Storage (KAF)	Area (KA)
0	0	450	10.8
50	2.1	500	11.6
100	3.6	550	12.4
150	5.0	600	13.1
200	6.2	650	13.8
250	7.2	700	14.5
300	8.1	750	15.2
350	9.1	800	15.8
400	10.0		

TABLE - 3.

CHARACTERISTICS OF PANCHET HILL RESERVOIR AND POWER PLANT

Storage (KAF)	Area (KA)	Capability (MW)	Energy Rate (MWH/KAF)
0	0	0	0
50	2. 9	0	0
100	3. 3	35.0	53. 0
150	8. 4	37.5	62. 0
200	9. 8	41.0	68. 5
250	11. 5	43.0	73. 0
300	13. 4	44.0	78. 5
350	15. 1	44.0	81. 5
400	16. 8	44.0	84. 5
450	18. 4	44.0	86. 5
500	20.0	44.0	88. 5
550	21. 5	44.0	90. 5
600	22. 9	44.0	92. 0
650	24. 3	44.0	93. 5
700	25. 9	44.0	94. 5
750	27. 6	44.0	95. 5
800	29. 3	44.0	96. 5

Cont'd......

Storage (KAF)	Area (KA)	Capability (MW)	Energy Rate (MWH/KAF)
850	31.1	44.0	97.0
900	33.1	44.0	97.0
950	35.2	44.0	98.0
1000	37.3	44.0	98.0
1050	39.5	44.0	98.0

TABLE - 4.

CHARACTERISTICS OF TILAIYA RESERVOIR AND POWER PLANT

Storage (KAF)	Area (KA)	Capability (MW)	Energy Rate (MWH/KAF)
0	0	0	0
50	3.8	0	0
100	6.2	2.68	37.5
150	8.5	3.24	45.2
200	10.5	3.51	49.2
250	12.4	3.68	52.0
300	14.0	3.94	53.7
350	16.4	4.00	55.5

TABLE - 5.

CHARACTERISTICS OF MAITHON RESERVOIR AND POWER PLANT

Storage (KAF)	Area (KA)	Capability (MW)	Energy Rate (MWH/KAF)
0	0	0	0
50	2.4	0	0
100	4.3	0	0
150	5.8	43.8	75.0
200	7.4	50.4	82.5
250	9.0	54.3	88.0
300	10.4	57.0	94.0
350	11.7	58.8	96.0
400	13.0	60.0	99.0
450	14.2	60.0	101.0
500	15.4	60.0	104.0
550	16.6	60.0	106.0
600	17.7	60.0	107.5
650	18.7	60.0	108.5
700	19.7	60.0	110.0
750	26.6	60.0	111.5
800	21.5	60.0	112.5
850	22.4	60.0	113.0
900	23.4	60.0	113.5
950	24.5	60.0	114.0
1000	25.6	60.0	114.5
1050	26.7	60.0	115.0

TABLE - 6.

EVAPORATION RATE FOR DIFFERENT RESERVOIRS

Month	Net Evaporation Rate(FEET/MONTH)				
	KONAR.	TENUGHAT	PANCHET HILL	TILAIYA	MAITHON
June	0.55	0.55	0.57	0.68	0.46
July	0.32	0.32	0.32	0.34	0.28
August	0.28	0.28	0.28	0.28	0.22
September	0.31	0.31	0.30	0.30	0.24
October	0.32	0.32	0.30	0.30	0.29
November	0.31	0.31	0.27	0.28	0.28
December	0.28	0.28	0.24	0.26	0.27
January	0.28	0.28	0.30	0.28	0.27
February	0.34	0.34	0.34	0.38	0.32
March	0.53	0.53	0.56	0.63	0.48
April	0.67	0.67	0.71	0.84	0.62
May	0.75	0.75	0.77	0.94	0.66

TABLE - 7.

MAXIMUM STORAGE LEVELS FOR DIFFERENT RESERVOIRS

Begining of Month	Maximum permissible storage(KAF)				
	KONAR	TENUGHAT	PANCHET HILL	TILAIYA	MAITHON
June	228.0	830.0	318.0	175.5	597.0
July	228.0	830.0	318.0	175.5	597.0
August	228.0	830.0	318.0	175.5	597.0
September	228.0	830.0	318.0	175.5	597.0
October	228.0	830.0	876.0	175.5	910.0
November	272.0	830.0	876.0	319.4	910.0
December	272.0	830.0	876.0	319.4	910.0
January	272.0	830.0	876.0	319.4	910.0
February	272.0	830.0	876.0	319.4	910.0
March	272.0	830.0	876.0	319.4	910.0
April	272.0	830.0	876.0	319.4	910.0
May	272.0	830.0	876.0	319.4	910.0

TABLE # 8.

COOLING WATER REQUIREMENTS FOR BOKARO THERMAL STATION FROM KONAR RESERVOIR *

Month	Cooling Water Requirements (CFS)	(KAF)
June	450	27
July	450	28
August	500	31
September	450	27
October	350	22
November	250	15
December	150	9
January	100	6
February	150	8
March	250	16
April	300	18
May	350	22

Cooling requirement is non consumptive.

* Projected consumptive requirement below Konar upto Panchet Hill Reservoir is 250 CFS (7.08 cumecs)

TABLE - 9.

MAXIMUM RELEASE THROUGH POWER HOUSES

Maximum Release Capacity (KAF)

Month	PANCHET HILL	TILAIYA	MAITHON
June	411.30	51.80	428.40
July	425.01	53.50	442.68
August	425.01	53.50	442.68
September	411.30	51.80	428.40
October	425.01	53.50	442.68
November	411.30	51.80	428.40
December	425.01	53.50	442.68
January	425.01	53.50	442.68
February	383.88	48.40	397.84
March	425.01	53.50	442.68
April	411.30	51.80	428.40
May	425.01	53.50	442.68

TABLE -10.

RUNOFF FROM THE CATCHMENT ABOVE KONAR DAM

Runoff (KAF)

Month - Year	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May
1956-57	20	48	48	68	39	6	2	9	2	3	4	14
1957-58	12	65	37	80	6	2	3	5	8	7	6	7
1958-59	32	61	69	106	27	5	5	5	4	3	7	10
1959-60	45	120	116	160	140	8	6	4	4	4	10	23
1960-61	29	44	111	83	49	3	5	4	11	10	23	10
1961-62	76	125	141	133	108	11	4	4	4	4	4	22
1962-63	23	57	59	60	23	4	2	2	2	3	4	8
1963-64	30	34	83	123	160	16	5	2	2	1	2	12
1964-65	31	74	62	51	11	1	1	0	1	0	1	4
1965-66	9	73	47	63	11	0	0	2	1	1	1	1

TABLE-11.

TUNOFF FROM THE CATCHMENT ABOVE TENUGHAT DAM

Runoff (KAF)

Month - Year	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
1956-57	176	269	182	387	186	0	0	0	0	0	0	0
1957-58	25	377	172	397	31	0	0	0	0	0	0	0
1958-59	68	401	332	468	130	0	0	0	0	0	0	0
1959-60	108	223	364	402	547	0	0	0	0	0	0	0
1960-61	34	223	453	326	193	0	0	0	0	0	0	0
1961-62	307	448	714	758	320	0	0	0	0	0	0	0
1962-63	39	265	339	275	50	0	0	0	0	0	0	0
1963-64	68	170	260	335	501	0	0	0	0	0	0	0
1964-65	93	444	468	414	192	0	0	0	0	0	0	0
1965-66	44	442	117	239	18	0	0	0	0	0	0	0

TABLE - 12.

RUNOFF FROM THE EFFECTIVE CATCHMENT ABOVE PANCHET HILL DAM

Runoff (KAF)

Month - Year.	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
1956-57	33	347	380	694	284	134	30	42	10	13	5	10
1957-58	30	117	65	16	0	9	7	8	8	2	3	3
1958-59	22	131	290	391	32	37	15	22	20	0	0	1
1959-60	129	530	627	908	1179	116	39	27	6	5	0	0
1960-61	9	210	714	646	295	52	20	16	117	0	0	0
1961-62	24	331	669	684	674	130	37	18	16	5	10	0
1962-63	60	199	278	237	138	22	20	5	0	0	0	0
1963-64	54	170	150	194	442	106	22	16	2	2	1	17
1964-65	96	389	294	306	122	53	20	15	2	5	22	0
1965-66	18	321	493	317	81	20	7	16	0	0	14	8

TABLE - 13.

RUNOFF FROM THE CATCHMENT ABOVE TILAIYA DAM

Runoff (KAF)

Month - Year.	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
1956-57	12	26	19	75	27	3	3	11	3	4	6	8
1957-58	19	88	29	54	8	4	3	3	5	10	9	6
1958-59	7	49	80	103	3	7	4	13	10	8	8	19
1959-60	40	93	110	87	105	12	9	8	6	8	5	10
1960-61	6	34	91	98	34	6	5	4	10	6	6	7
1961-62	113	122	146	101	110	7	6	3	3	9	5	8
1962-63	9	55	44	68	56	8	5	10	2	2	12	9
1963-64	12	16	30	62	148	10	1	18	9	5	6	5
1964-65	17	69	35	69	28	3	4	4	4	1	1	1
1965-66	17	46	61	43	7	0	2	4	3	3	0	1

TABLE - 14.

RUNOFF FROM THE EFFECTIVE CATCHMENT ABOVE MAITHON DAM

Runoff (KAF)

Month - Year.	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
1956-57	233	367	320	629	176	92	16	43	4	10	5	0
1957-58	123	642	281	365	1	5	0	1	0	4	1	2
1958-59	34	225	459	518	207	10	8	18	53	0	0	4
1959-69	165	490	525	505	1406	46	11	4	6	8	0	10
1960-61	61	224	595	661	342	19	2	3	86	0	0	4
1961-62	124	372	742	512	802	46	10	4	4	0	9	0
1962-63	67	156	392	329	155	2	2	2	2	2	2	13
1963-64	132	273	218	468	1103	70	0	0	0	0	6	26
1964-65	107	403	311	393	142	3	5	0	0	5	4	0
1965-66	116	293	343	383.	36	2	0	14	1	0	0	1

TABLE - 15.

RUNOFF FROM THE UNCONTROLLED CATCHMENT BELOW MAITHON AND PANCHET HILL DAMS UPTO DURGAPUR BARRAGE

Runoff (KAF)

Month - Year.	June.	Jul.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
1956-57	0	164	1067	919	320	0	0	0	0	0	0	0
1957-58	0	632	272	687	72	0	0	0	0	0	0	0
1958-59	0	288	286	519	242	0	0	0	0	0	0	0
1959-60	0	177	312	656	1183	0	0	0	0	0	0	0
1960-61	0	45	307	280	126	0	0	0	0	0	0	0
1961-62	0	210	242	283	338	0	0	0	0	0	0	0
1962-63	0	144	250	145	82	0	0	0	0	0	0	0
1963-64	0	165	98	120	344	0	0	0	0	0	0	0
1964-65	0	158	352	236	100	0	0	0	0	0	0	0
1965-66	0	221	264	217	86	0	0	0	0	0	0	0

TABLE - 16.

TOTAL INFLOW IN THE DVC SYSTEM IN DIFFERENT YEARS

Inflow (KAF)

Years	KONAR	TENUGHAT	PANCHET HILL	TILAIYA	MAITHON	DURGAPUR	TOTAL
1956-57	267	1200	1982	197	1895	1266	6807
1957-58	238	1102	268	238	1425	1663	4934
1958-59	331	1389	961	320	1536	1289	5826
1959-60	640	1644	3566	493	3176	2337	11856
1960-61	382	1229	2079	257	1996	738	6681
1961-62	636	2547	2598	633	2625	1073	10112
1962-63	247	968	959	280	1124	621	4199
1963-64	470	1334	1324	316	2296	727	6467
1964-65	237	1611	1176	236	1373	846	5479
1965-66	211	860	1294	187	1189	788	4529

TABLE - 17.

MONTHLY RUNOFF FROM EFFECTIVE CATCHMENTS ABOVE DAMS AND DURGAPUR BARRAGE DURING 1962 - 1963

Runoff (KAF)

Month	KONAR	TENUGHAT	PANCHET HILL	TILAIYA	MAITHON	DURGAPUR
June	23	39	60	9	67	0
July	57	265	199	55	156	144
Aug.	59	339	278	44	392	250
Sept	60	275	237	68	329	145
Oct	23	50	138	56	155	82
Nov	4	0	22	8	2	0
Dec	2	0	20	5	2	0
Jan	2	0	5	10	2	0
Feb	2	0	0	2	2	0
Mar	3	0	0	2	2	0
Apr	4	0	0	12	2	0
May	8	0	0	9	13	0
Total	247	968	959	280	1124	621

TABLE - 18.

DOWNSTREAM REQUIREMENTS DURING 1962 - 63

Downstream Requirements (KAF)

Month / Purpose	June.	July.	Aug.	Sept.	Oct.	Nov.	Dec.	Jan.	Feb.	Mar.	Apr.	May.
Kharif	0	0	177	354	1118	0	0	0	0	0	0	0
Rabi	0	0	0	0	0	0	7	14	14	14	7	0
Industrial	28	29	29	28	29	28	29	29	27	29	28	29
Mandatory	0	0	0	0	0	6	6	6	6	6	6	6
Navigation	24	0	0	0	0	24	25	24	24	24	25	25
Losses	6	-	-	-	-	7	7	7	7	7	7	6
Total	58	29	206	382	1147	72	80	80	78	80	73	66

TABLE - 19.

WATER REQUIREMENT FOR PADDY DURING MONSOON 1962 - 63

Month	Period	Total Requirement (INCHES) (1)	Rainfall (INCHES)	Rainfall Deficiency (INCHES)	Canal Requirement (FEET)	Canal Requirement (KAF)
July	16 - 20	1.8	4.0	0	0	0
	21 - 31	3.8	3.8	0	0	0
Aug	1 - 10	3.4	2.4	1.0	0.140	136
	11 - 20	3.5	4.0	0	0	0
	21 - 31	3.8	3.5	0.3	0.042	41
Sept	1 - 10	3.5	0.9	2.6	0.365	354
	11 - 20	3.5	3.9	0	0	0
	21 - 30	3.5	4.7	0	0	0
Oct	1 - 10	3.5	0.9	2.6	0.365	354
	11 - 20	3.4	0.0	3.4	0.477	464
	21 - 31	3.4	1.2	2.2	0.308	300

(1) Consumptive use + percolation loss.

TABLE - 20.

INITIAL POLICY FOR KONAR, TENUGHAT AND TILAIYA
(FROM MASS BALANCE)

Beginning Storage (KAF)

Month	KONAR	TENUGHAT	TILAIYA
June	49.00	170.00	61.00
July	49.00	184.62	65.00
Aug	76.69	427.87	117.00
Sept	103.48	745.08	158.00
Oct	135.06	825.14	162.00
Nov	134.50	824.82	164.00
Dec	121.87	801.80	168.00
Jan	106.43	779.31	169.00
Feb	91.03	756.87	175.00
Mar	77.37	729.52	172.00
Apr	61.36	698.26	166.00
May	49.00	662.93	168.00
June	49.00	616.48	166.00

TABLE - 21.

REDEFINED FIRM WATER CONTRACT FOR KONAR AND TENU GHAT

Month	CORE 1 (KAF)	CORE 2 (KAF)
June	20	18
July	28	18
Aug	31	18
Sept	27	18
Oct	22	18
Nov	15	18
Dec	16	18
Jan	16	18
Feb	14	22
Mar	16	23
Apr	13	25
May	4	35

TABLE - 22.

REFINED INITIAL POLICY FOR TILAIYA RESERVOIR
(BY INCREMENTAL DYNAMIC PROGRAMMING) *

Month	Beginning Storage (KAF)
June	61
July	65
Aug	117
Sept	158
Oct	174
Nov	174
Dec	170
Jan	171
Feb	127
Mar	108
Apr	104
May	110
June	61

* Energy production = 9957.06 MWH / YEAR at Tilaiya.

TABLE - 23.

INFLOW INTO PANCHET HILL RESERVOIR WITH INITIAL POLICY AT KONAR AND TENUGHAT

Month	Runoff from effective catchment (KAF)	Releases R1 (KAF)	Releases R2 (KAF)	Consumptive use Steel (KAF)	Consumptive use Industry (KAF)	Loss (KAF)	Net Inflow (KAF)
June	60	20	18	18	15	0	65
July	199	28	18	18	16	0	211
Aug	278	31	18	18	16	0	293
Sept	237	27	190	18	15	0	421
Oct	138	22	45	18	16	0	171
Nov	22	15	18	18	16	5	17
Dec	20	16	18	18	16	5	15
Jan	5	16	18	18	16	5	0
Feb	0	14	22	17	14	5	0
Mar	0	16	23	18	16	5	0
Apr	0	13	25	18	15	5	0
May	0	4	35	18	16	5	0

TABLE - 24.

INFLOW INTO MAITHON RESERVOIR WITH INITIAL POLICY AT TILAIYA

Month	Runoff from effective catchment (KAF)	Release from Tilaiya (KAF)	Total (KAF)	Loss (KAF)	Net inflow (KAF)
June	67	1.99	68.99	0.99	68
July	156	1.03	157.03	1.03	156
Aug	392	0.78	392.78	0.78	392
Sept	329	49.25	378.25	0.25	378
Oct	155	53.16	208.16	0.16	208
Nov	2	9.37	11.37	1.37	10
Dec	2	1.57	3.57	1.57	2
Jan	2	51.63	53.63	1.63	52
Feb	2	18.33	20.33	1.33	19
Mar	2	1.92	3.92	1.92	2
Apr	2	0.52	2.52	0.52	2
May	13	52.82	65.82	1.82	64

TABLE - 25.

ROUGH INITIAL POLICY FOR PANCHET HILL RESERVOIR (HAND COMPUTATION)

Month	Storage(KAF) Beginning	Storage(KAF) Ending	Max. Evaporation Loss (KAF)*	Inflow (KAF)	Min. Release (KAF)
June	139	194	10	65	0
July	194	318	6	211	81
Aug	318	318	8	293	285
Sept	318	610	10	421	119
Oct	610	238	10	171	533
Nov	238	215	4	17	36
Dec	215	186	4	15	40
Jan	186	183	3	0	0
Feb	183	179	4	0	0
Mar	179	173	6	0	0
Apr	173	165	8	0	0
May	165	139	8	0	18
June	139				

* Approximate.

TABLE - 26.

ROUGH INITIAL POLICY FOR MAITHON RESERVOIR
(HAND COMPUTATION)

Month	Storage(KAF) Beginning	Storage(KAF) Ending	Max. Evaporation Loss(KAF)*	Inflow (KAF)	Min. Release (KAF)	Combined Release (KAF)**
June	136	136	3	68	65	65
July	136	288	4	156	0	81
Aug	288	597	5	392	78	363
Sept	597	851	6	378	118	237
Oct	851	521	6	208	532	1065
Nov	521	488	5	10	38	74
Dec	488	403	5	2	82	122
Jan	403	369	4	52	82	82
Feb	369	304	5	19	79	79
Mar	304	217	7	2	82	82
Apr	217	138	6	2	75	75
May	138	136	5	64	61	79
June	136					

* Approximate.
** Maithon and Panchet Hill

TABLE - 27.

REFINED INITIAL POLICY FOR PANCHET HILL AND MAITHON
(BY INCREMENTAL DYNAMIC PROGRAMMING) *

Month	Beginning Storage (KAF) PANCHET HILL	MAITHON
June	139	136
July	194	146
Aug	318	298
Sept	318	597
Oct	565	876
Nov	193	546
Dec	170	513
Jan	176	438
Feb	173	404
Mar	169	339
Apr	163	252
May	155	173
June	139	136

* Energy generation at Panchet Hill = 81002.75 MWH/YEAR

Energy generation at Maithon = 116860.60 MWH/YEAR

TABLE - 28.

RELEASE FROM THE RESERVOIRS CORRESPONDING TO THE INITIAL POLICY (Computer Program in Appdx. G)

Release (KAF)

Month	KONAR	TENUGHAT	PANCHET HILL	TILAIYA	MAITHON
June	20.00	18.00	6.09	1.99	55.45
July	28.00	18.00	83.24	1.03	1.73
Aug	31.00	18.00	289.07	0.78	89.88
Sept	27.00	190.00	168.56	49.25	94.11
Oct	22.00	45.00	538.17	53.16	532.22
Nov	15.00	18.00	37.79	9.37	38.48
Dec	16.00	18.00	7.21	1.57	73.00
Jan	16.00	18.00	0.73	51.63	82.35
Feb	14.00	22.00	1.53	18.33	80.07
Mar	16.00	23.00	2.18	1.92	84.06
Apr	13.00	25.00	3.60	0.52	76.16
May	4.00	35.00	11.93	52.82	97.07

TABLE - 29.

RESULT OF JOINT OPERATION

ITERATION NO.	FIRM ENERGY (MWH/ YEAR)	DELT 2 (KAF)	DELT 3 (KAF)	DELT 4 (KAF)	DELT 5 (KAF)
0	207820.41	-	-	-	-
1	217238.20	100	100	50	100
2	226053.43	100	100	50	100
3	234353.15	100	100	50	100
4	239031.34	50	50	20	50
5	243725.83	50	50	20	50
6	243725.83	50	50	20	50
7	244745.83	10	20	10	20
8	245565.41	10	20	10	20
9	246350.09	10	20	10	20
10	247431.01	5	10	5	10
11	247908.37	5	10	5	10
12	248301.97	5	10	5	10
13	248627.48	2	5	2	5
14	248701.78	2	5	2	5

Appendix A

A COMPUTER PROGRAM IN FORTRAN II FOR INITIAL POLICY OF RESERVOIR 1 AND 2

(Mass Balance)

```
**
      DIMENSIONEVAR(12),SMAX(12),AREA(50)
      DIMENSIONRELES(12),CORE(12),S1(13),S2(13),AINF(12),EVAP(12)
  400 READ100,NOBS
      READ600,(EVAR(J),J=1,12),(SMAX(J),J=1,12)
      READ600,(CORE(J),J=1,12),(AINF(J),J=1,12)
      READ601,(AREA(L),L=1,NOBS)
      READ600,SMIN, STOR, DELT
  100 FORMAT(I10)
  600 FORMAT(6F10.2)
  601 FORMAT(8F10.2)
      S1(1)=STOR
      DO500N=1,12
      N1=N+1
      RELES(N)=CORE(N)
      AVST=0.01*(S1(N)+SMAX(N))+1.0
      I=AVST $ B=I $ I1=I+1
      EVAP(N)=(AREA(I)+(AVST-B)*(AREA(I1)-AREA(I)))*EVAR(N)
      S2(N)=S1(N)+AINF(N)-RELES(N)
      IF(S2(N)-SMAX(N))496,496,497
  497 S2(N)=SMAX(N)
  496 S2(N)=S2(N)-EVAP(N)
      IF(S2(N)-SMIN)498,499,499
  498 S2(N)=SMIN
  499 RELES(N)=S1(N)+AINF(N)-EVAP(N)-S2(N)
  500 S1(N1)=S2(N)
      PUNCH601,(S1(K),K=1,13)
      PUNCH600,(RELES(N),N=1,12)
      GOTO400
      END
```

Appendix B

A COMPUTER PROGRAM IN FORTRAN II FOR INITIAL POLICY OF RESERVOIR 4

(Optimal Operating Policy For a Single Reservoir System Using Incremental Dynamic Programming)

**

```
      DIMENSIONEVAR(12),SMAX(12),AINF(12),PREMA(12),PRHRS(12)
      DIMENSIONAREA(50),ENGR(50),CAPA(50)
      DIMENSIONAIND(12,3),STOR(13),UU(3),V(3,3),W(3,3)
      COMMONSMIN,SMAX,AINF,AREA,EVAR,ENGR,CAPA,PREMA,PRHRS
      READ600,NOBS
      READ601,(EVAR(L),L=1,12),(SMAX(L),L=1,12),(AINF(L),L=1,12)
      READ601,(PREMA(L),L=1,12),(PRHRS(L),L=1,12)
      READ602,(STOR(I),I=1,13)
      READ603,(AREA(K),CAPA(K),ENGR(K),K=1,NOBS)
      READ604,SMIN,DELT,DEL,DE,D,ED
  600 FORMAT(I10)
  601 FORMAT(6F10.3)
  602 FORMAT(7F10.3)
  603 FORMAT(6F10.3)
  604 FORMAT(6F10.4)
  605 FORMAT(2F10.4,F20.4)
      PRMAX=0.0
      AITER=0.0
  800 AITER=AITER+1.0
      K=1 $ K1=K+1 $ U=0.0
      S1=STOR(K)
      DO101I=1,3
      AI=I
      S2=STOR(K1)+(AI-2.)*DELT
      CALLPOWER(S1,S2,K,U)
  101 UU(I)=U
      DO650M=1,3
  650 AIND(K,M)=2.
      DO500K=2,12
      K1=K+1
      DO106I=1,3
      AI=I
      S1=STOR(K)+(AI-2.)*DELT
      DO105M=1,3
      AM=M
      S2=STOR(K1)+(AM-2.)*DELT
      CALLPOWER(S1,S2,K,U)
  105 V(I,M)=U
  106 CONTINUE
      DO107M=1,3
      DO108I=1,3
  108 W(M,I)=UU(I)+V(I,M)
  107 CONTINUE
```

```
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO109M=1,3
      DO110I=1,3
      IF(W(M,I)+999999.)10,20,20
   10 W(M,I)=-999999.
   20 IF(UU(M)-W(M,I))150,150,110
  150 UU(M)=W(M,I)
      AIND(K,M)=I
  110 CONTINUE
  109 CONTINUE
  500 CONTINUE
      DO901I=1,3
      IF(PRMAX-UU(I))900,900,901
  900 PRMAX=UU(I)
      M=I $ AM=M
  901 CONTINUE
      PUNCH605,AITER,DELT,PRMAX
      STOR(13)=STOR(13)+(AM-2.)*DELT
      DO903I=1,12
      K=12-I+1
      STOR(K)=STOR(K)+(AIND(K,M)-2.)*DELT
  903 M=AIND(K,M)
      PUNCH601,(STOR(K),K=1,13)
      IF(AITER-5.)800,400,400
  400 DELT=DEL
      IF(AITER-10.0)800,300,300
  300 DELT=DE
      IF(AITER-15.0)800,200,200
  200 DELT=D
      IF(AITER-20.0)800,100,100
  100 DELT=ED
      IF(AITER-25.0)800,850,850
  850 STOP
      END
```

**

```
      SUBROUTINEPOWER(S1,S2,K,U)
      DIMENSIONEVAR(12),SMAX(12),AINF(12),PREMA(12),PRHRS(12)
      DIMENSIONAREA(50),ENGR(50),CAPA(50)
      COMMONSMIN,SMAX,AINF,AREA,EVAR,ENGR,CAPA,PREMA,PRHRS
      IF(S1-SMAX(K))399,399,400
  399 IF(S1-SMIN)400,401,401
  401 IF(S2-SMIN)400,402,402
  402 IF(S2-SMAX(K))403,403,400
  403 IF(AINF(K)+S1-S2)400,404,404
  404 AVST=(S1+S2)/100.+1.0
      I=AVST $ I1=I+1 $ B=I
      EVAP=(AREA(I)+(AVST-B)*(AREA(I1)-AREA(I)))*EVAR(K)
      RELES=S1-S2+AINF(K)-EVAP
      IF(RELES)400,405,405
  405 IF(RELES-PREMA(K))406,406,407
  406 PRELS=RELES $ GOTO408
  407 PRELS=PREMA(K)
  408 U=PRELS*(ENGR(I)+(AVST-B)*(ENGR(I1)-ENGR(I)))
      ENMAX=PRHRS(K)*(CAPA(I)+(AVST-B)*(CAPA(I1)-CAPA(I)))
      IF(U-ENMAX)409,409,410
  410 U=ENMAX
  409 RETURN
  400 U=-999999.
      RETURN
      END
```

Appendix C

A COMPUTER PROGRAM IN FORTRAN II FOR
INITIAL POLICY OF RESERVOIRS 3 AND 5
(Optimal Operating Policy For a 2 Reservoir System
Using Incremental Dynamic Programming)

```
**
      DIMENSIONEVAR3(12),AREA3(25),CAPA3(25),ENGR3(25),AINF3(12)
      DIMENSIONAIND(12,3),UU(3),V(3,3),W(3,3)
      DIMENSIONSTOR3(13),SMAX3(13),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),PREM3(12),PRHRS(12)
      COMMONSMAX3,SMIN3,AINF3,DELT3,AREA3,EVAR3,CAPA3,ENGR3,R5,UR6,
     1STOR3,DRMIN,DRMAX,PREM3,PRHRS
      READ4,NOBS,PRM3
      READ1,(EVAR3(I),I=1,12)
      READ1,(SMAX3(N),N=1,13)
      READ1,(AREA3(K),CAPA3(K),ENGR3(K),K=1,NOBS)
      READ1,(AINF3(I),I=1,12)
      READ1,(R5(I),I=1,12),(UR6(I),I=1,12)
      READ2,(STOR3(N),N=1,13)
      READ1,(DRMIN(I),I=1,12),(DRMAX(I),I=1,12)
      READ1,(PREM3(I),I=1,12),(PRHRS(I),I=1,12)
      READ1,SMIN3,DELT3
    1 FORMAT(6F10.2)
    2 FORMAT(7F10.2)
    4 FORMAT(I10,2F20.2)
      ITER=0
  800 ITER=ITER+1
      K=1 $ K1=K+1 $ U=0.0
      S31=STOR3(K)
      DO301I=1,3
      AI=I
      S32=STOR3(K1)+(AI-2.)*DELT3
      CALLRELP3(S31,S32,K,U)
  301 UU(I)=U
      DO670M=1,3
  670 AIND(K,M)=2.
      DO530K=2,12
      K1=K+1
      DO306I=1,3
      AI=I
      S31=STOR3(K)+(AI-2.)*DELT3
      DO305M=1,3
      AM=M
      S32=STOR3(K1)+(AM-2.)*DELT3
      CALLRELP3(S31,S32,K,U)
  305 V(I,M)=U
  306 CONTINUE
      DO307M=1,3
      DO308I=1,3
  308 W(M,I)=UU(I)+V(I,M) .
  307 CONTINUE
```

```
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO309M=1,3
      DO310I=1,3
      IF(W(M,I)+999999.)10,20,20
   10 W(M,I)=-999999.
   20 IF(UU(M)-W(M,I))350,350,310
  350 UU(M)=W(M,I)  $ AIND(K,M)=I
  310 CONTINUE
  309 CONTINUE
  530 CONTINUE
      DO701I=1,3
      IF(PRM3-UU(I))702,702,701
  702 PRM3=UU(I)  $ M=I  $ AM=M
  701 CONTINUE
      PUNCH4,ITER,DELT3,PRM3
      STOR3(13)=STOR3(13)+(AM-2.)*DELT3
      DO703I=1,12
      K=12-I+1
      STOR3(K)=STOR3(K)+(AIND(K,M)-2.)*DELT3
  703 M=AIND(K,M)
      PUNCH2,(STOR3(N),N=1,13)
      IF(ITER-5)800,400,400
  400 DELT3=50.
      IF(ITER-10)800,300,300
  300 DELT3=25.
      IF(ITER-15)800,200,200
  200 DELT3=10.
      IF(ITER-20)800,100,100
  100 STOP
      END
```

**

```
      SUBROUTINERELP3(S31,S32,K,U)
      DIMENSIONEVAR3(12),AREA3(25),CAPA3(25),ENGR3(25),AINF3(12)
      DIMENSIONDRMIN(12),DRMAX(12),PREM3(12),PRHRS(12)
      DIMENSIONSTOR3(13),SMAX3(13),R5(12),UR6(12)
      COMMONSMAX3,SMIN3,AINF3,DELT3,AREA3,EVAR3,CAPA3,ENGR3,R5,UR6,
     1STOR3,DRMIN,DRMAX,PREM3,PRHRS
      K1=K+1
      IF(S31-SMAX3(K))399,399,400
  399 IF(S31-SMIN3)400,401,401
  401 IF(S32-SMIN3)400,402,402
  402 IF(S32-SMAX3(K1))403,403,400
  403 IF(AINF3(K)+S31-S32)400,404,404
  404 AVST3=(S31+S32)/100.+1.
      J=AVST3 $ J1=J+1 $ C=J
      EVAP=(AREA3(J)+(AVST3-C)*(AREA3(J1)-AREA3(J)))*EVAR3(K)
      RELES=S31-S32+AINF3(K)-EVAP
      IF(RELES)400,405,405
  405 AINF6=RELES+R5(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))351,351,400
  351 IF(RELES-PREM3(K))406,406,407
  406 PREL=RELES $ GOTO408
  407 PREL=PREM3(K)
  408 U=PREL*(ENGR3(J)+(AVST3-C)*(ENGR3(J1)-ENGR3(J)))
      ENMAX=PRHRS(K)*(CAPA3(J)+(AVST3-C)*(CAPA3(J1)-CAPA3(J)))
      IF(U-ENMAX)409,409,410
  410 U=ENMAX
  409 RETURN
  400 U=-999999.
      RETURN
      END
```

Appendix D

A COMPUTER PROGRAM IN FORTRAN II FOR OVERALL OPTIMIZATION

(Optimal Operating Policy For a 5 Reservoir System Using Incremental Dynamic Programming)

**

```
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      DIMENSIONAIND(12,3),UU(3),V(3,3),RR(12),W(3,3)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      READ1,(EVAR1(I),I=1,12),(EVAR2(I),I=1,12),(EVAR3(I),I=1,12)
      READ1,(EVAR4(I),I=1,12),(EVAR5(I),I=1,12)
      READ1,(SMAX1(N),N=1,13)
      READ1,(SMAX2(N),N=1,13)
      READ1,(SMAX3(N),N=1,13)
      READ1,(SMAX4(N),N=1,13)
      READ1,(SMAX5(N),N=1,13)
      READ2,(AREA1(K),K=1,7)
      READ2,(AREA2(L),L=1,18)
      READ2,(STOR1(N),N=1,13)
      READ2,(STOR2(N),N=1,13)
      READ2,(STOR3(N),N=1,13)
      READ2,(STOR4(N),N=1,13)
      READ2,(STOR5(N),N=1,13)
      READ1,(R1(I),I=1,12),(R2(I),I=1,12),(R3(I),I=1,12)
      READ1,(R4(I),I=1,12),(R5(I),I=1,12)
      READ1,(UR6(I),I=1,12)
      READ1,(AINF1(I),I=1,12),(AINF2(I),I=1,12),(AINF3(I),I=1,12)
      READ1,(AINF4(I),I=1,12),(AINF5(I),I=1,12)
      READ1,(AREA3(K),CAPA3(K),ENGR3(K),K=1,22)
      READ1,(AREA4(L),CAPA4(L),ENGR4(L),L=1,8)
      READ1,(AREA5(K),CAPA5(K),ENGR5(K),K=1,22)
      READ1,(CORE1(I),I=1,12),(CORE2(I),I=1,12)
      READ1,(DRMIN(I),I=1,12),(DRMAX(I),I=1,12)
      READ1,(PREM3(I),I=1,12),(PREM4(I),I=1,12),(PREM5(I),I=1,12)
      READ1,(PRHRS(I),I=1,12)
      READ1,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5
       READ1,DELT1,DELT2,DELT3,DELT4,DELT5
      READ1,DEL1,DEL2,DEL3,DEL4,DEL5
      READ1,DE1,DE2,DE3,DE4,DE5
      READ1,D1,D2,D3,D4,D5
      READ1,ED1,ED2,ED3,ED4,ED5
      READ1,ELD1,ELD2,ELD3,ELD4,ELD5
      READ3,PRM3,PRM45
```

```
    1 FORMAT(6F10.2)
    2 FORMAT(8F10.2)
    3 FORMAT(2F20.2)
    4 FORMAT(F10.2,F20.2,5F10.2)
    5 FORMAT(7F10.2)
      AITER=0.
  800 AITER=AITER+1.
      K=1 $ K1=K+1 $ U=0.
      S11=STOR1(K)
      DO101I=1,3
      AI=I
      S12=STOR1(K1)+(AI-2.)*DELT1
      CALLRELP1(S11,S12,K,U)
  101 UU(I)=U
      DO650M=1,3
  650 AIND(K,M)=2.
      DO510K=2,12
      K1=K+1
      DO106I=1,3
      AI=I
      S11=STOR1(K)+(AI-2.)*DELT1
      DO105M=1,3
      AM=M
      S12=STOR1(K1)+(AM-2.)*DELT1
      CALLRELP1(S11,S12,K,U)
  105 V(I,M)=U
  106 CONTINUE
      DO107M=1,3
      DO108I=1,3
  108 W(M,I)=UU(I)+V(I,M)
  107 CONTINUE
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO109M=1,3
      DO110I=1,3
      IF(W(M,I)+999999.)10,20,20
   10 W(M,I)=-999999.
   20 IF(UU(M)-W(M,I))150,150,110
  150 UU(M)=W(M,I)
      AIND(K,M)=I
  110 CONTINUE
  109 CONTINUE
  510 CONTINUE
      DO901I=1,3
      IF(PRM3-UU(I))902,902,901
  902 PRM3=UU(I) $ M=I $ AM=M
  901 CONTINUE
      STOR1(13)=STOR1(13)+(AM-2.)*DELT1
      DO903I=1,12
      K=12-I+1
      STOR1(K)=STOR1(K)+(AIND(K,M)-2.)*DELT1
  903 M=AIND(K,M)
      DO11K=1,12
      K1=K+1
      AVST=(STOR1(K)+STOR1(K1))/100.+1.
```

```
      I=AVST $ I1=I+1 $ B=I
      EVAP=(AREA1(I)+(AVST-B)*(AREA1(I1)-AREA1(I)))*EVAR1(K)
   11 RR(K)=STOR1(K)-STOR1(K1)+AINF1(K)-EVAP
      DO12K=1,12
      RRK=RR(K)-R1(K)
      AINF3(K)=AINF3(K)+RRK
      R3(K)=R3(K)+RRK
   12 R1(K)=RR(K)
      K=1 $ K1=K+1 $ U=0.0
      S21=STOR2(K)
      DO201I=1,3
      AI=I
      S22=STOR2(K1)+(AI-2.)*DELT2
      CALLRELP2(S21,S22,K,U)
  201 UU(I)=U
      DO660M=1,3
  660 AIND(K,M)=2.
      DO520K=2,12
      K1=K+1
      DO206I=1,3
      AI=I
      S21=STOR2(K)+(AI-2.)*DELT2
      DO205M=1,3
      AM=M
      S22=STOR2(K1)+(AM-2.)*DELT2
      CALLRELP2(S21,S22,K,U)
  205 V(I,M)=U
  206 CONTINUE
      DO207M=1,3
      DO208I=1,3
  208 W(M,I)=UU(I)+V(I,M)
  207 CONTINUE
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO209M=1,3
      DO210I=1,3
      IF(W(M,I)+999999.)30,40,40
   30 W(M,I)=-999999.
   40 IF(UU(M)-W(M,I))250,250,210
  250 UU(M)=W(M,I)
      AIND(K,M)=I
  210 CONTINUE
  209 CONTINUE
  520 CONTINUE
```

```
      DO801I=1,3
      IF(PRM3-UU(I))802,802,801
  802 PRM3=UU(I) $ M=I $ AM=M
  801 CONTINUE
      STOR2(13)=STOR2(13)+(AM-2.)*DELT2
      DO803I=1,12
      K=12-I+1
      STOR2(K)=STOR2(K)+(AIND(K,M)-2.)*DELT2
  803 M=AIND(K,M)
      DO21K=1,12
      K1=K+1
      AVST=(STOR2(K)+STOR2(K1))/100.+1.
      I=AVST $ I1=I+1 $ B=I
      EVAP=(AREA2(I)+(AVST-B)*(AREA2(I1)-AREA2(I)))*EVAR2(K)
   21 RR(K)=STOR2(K)-STOR2(K1)+AINF2(K)-EVAP
      DO22K=1,12
      RRK=RR(K)-R2(K)
      R3(K)=R3(K)+RRK
      AINF3(K)=AINF3(K)+RRK
   22 R2(K)=RR(K)
      K=1 $ K1=K+1 $ U=0.0
      S31=STOR3(K)
      DO301I=1,3
      AI=I
      S32=STOR3(K1)+(AI-2.)*DELT3
      CALLRELP3(S31,S32,K,U)
  301 UU(I)=U
      DO670M=1,3
  670 AIND(K,M)=2.
      DO530K=2,12
      K1=K+1
      DO306I=1,3
      AI=I
      S31=STOR3(K)+(AI-2.)*DELT3
      DO305M=1,3
      AM=M
      S32=STOR3(K1)+(AM-2.)*DELT3
      CALLRELP3(S31,S32,K,U)
  305 V(I,M)=U
  306 CONTINUE
      DO307M=1,3
      DO308I=1,3
  308 W(M,I)=UU(I)+V(I,M)
  307 CONTINUE
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO309M=1,3
      DO310I=1,3
      IF(W(M,I)+999999.)50,60,60
   50 W(M,I)=-999999.
   60 IF(UU(M)-W(M,I))350,350,310
  350 UU(M)=W(M,I) $ AIND(K,M)=I
  310 CONTINUE
  309 CONTINUE
  530 CONTINUE
```

```
      DO701I=1,3
      IF(PRM3-UU(I))702,702,701
  702 PRM3=UU(I)  $  M=I  $  AM=M
  701 CONTINUE
      PUNCH3,PRM3
      STOR3(13)=STOR3(13)+(AM-2.)*DELT3
      DO703I=1,12
      K=12-I+1
      STOR3(K)=STOR3(K)+(AIND(K,M)-2.)*DELT3
  703 M=AIND(K,M)
      DO31K=1,12
      K1=K+1
      AVST=(STOR3(K)+STOR3(K1))/100.+1.
      I=AVST  $  I1=I+1  $  B=I
      EVAP=(AREA3(I)+(AVST-B)*(AREA3(I1)-AREA3(I)))*EVAR3(K)
   31 R3(K)=STOR3(K)-STOR3(K1)+AINF3(K)-EVAP
      K=1  $  K1=K+1  $  U=0.0
      S41=STOR4(K)
      DO401I=1,3
      AI=I
      S42=STOR4(K1)+(AI-2.)*DELT4
      CALLRELP4(S41,S42,K,U)
  401 UU(I)=U
      DO680M=1,3
  680 AIND(K,M)=2.
      DO540K=2,12
      K1=K+1
      DO406I=1,3
      AI=I
      S41=STOR4(K)+(AI-2.)*DELT4
      DO405M=1,3
      AM=M
      S42=STOR4(K1)+(AM-2.)*DELT4
      CALLRELP4(S41,S42,K,U)
  405 V(I,M)=U
  406 CONTINUE
      DO407M=1,3
      DO408I=1,3
  408 W(M,I)=UU(I)+V(I,M)
  407 CONTINUE
      UU(1)=-999999.  $  UU(2)=-999999.  $  UU(3)=-999999.
      DO409M=1,3
      DO410I=1,3
      IF(W(M,I)+999999.)70,80,80
   70 W(M,I)=-999999.
   80 IF(UU(M)-W(M,I))450,450,410
  450 UU(M)=W(M,I)   $   AIND(K,M)=I
  410 CONTINUE
  409 CONTINUE
  540 CONTINUE
```

```
      DO601I=1,3
      IF(PRM45-UU(I))602,602,601
  602 PRM45=UU(I) $ M=I $ AM=M
  601 CONTINUE
      STOR4(13)=STOR4(13)+(AM-2.)*DELT4
      DO603I=1,12
      K=12-I+1
      STOR4(K)=STOR4(K)+(AIND(K,M)-2.)*DELT4
  603 M=AIND(K,M)
      DO41K=1,12
      K1=K+1
      AVST=(STOR4(K)+STOR4(K1))/100.+1.
      I=AVST $ I1=I+1 $ B=I
      EVAP=(AREA4(I)+(AVST-B)*(AREA4(I1)-AREA4(I)))*EVAR4(K)
   41 RR(K)=STOR4(K)-STOR4(K1)+AINF4(K)-EVAP
      DO42K=1,12
      RRK=RR(K)-R4(K)
      R5(K)=R5(K)+RRK
      AINF5(K)=AINF5(K)+RRK
   42 R4(K)=RR(K)
      K=1 $ K1=K+1 $ U=0.0
      S51=STOR5(K)
      DO501I=1,3
      AI=I
      S52=STOR5(K1)+(AI-2.)*DELT5
      CALLRELP5(S51,S52,K,U)
  501 UU(I)=U
      DO690M=1,3
  690 AIND(K,M)=2.
      DO550K=2,12
      K1=K+1
      DO506I=1,3
      AI=I
      S51=STOR5(K)+(AI-2.)*DELT5
      DO505M=1,3
      AM=M
      S52=STOR5(K1)+(AM-2.)*DELT5
      CALLRELP5(S51,S52,K,U)
  505 V(I,M)=U
  506 CONTINUE
      DO507M=1,3
      DO508I=1,3
  508 W(M,I)=UU(I)+V(I,M)
  507 CONTINUE
      UU(1)=-999999. $ UU(2)=-999999. $ UU(3)=-999999.
      DO509M=1,3
      DO511I=1,3
      IF(W(M,I)+999999.)90,100,100
   90 W(M,I)=-999999.
  100 IF(UU(M)-W(M,I))551,551,511
  551 UU(M)=W(M,I)  $  AIND(K,M)=I
  511 CONTINUE
```

```
509 CONTINUE
550 CONTINUE
    DO591I=1,3
    IF(PRM45-UU(I))592,592,591
592 PRM45=UU(I) $ M=I $ AM=M
591 CONTINUE
    STOR5(13)=STOR5(13)+(AM-2.)*DELT5
    DO593I=1,12
    K=12-I+1
    STOR5(K)=STOR5(K)+(AIND(K,M)-2.)*DELT5
593 M=AIND(K,M)
    DO51K=1,12
    K1=K+1
    AVST=(STOR5(K)+STOR5(K1))/100.+1.
    I=AVST $ I1=I+1 $ B=I
    EVAP=(AREA5(I)+(AVST-B)*(AREA5(I1)-AREA5(I)))*EVAR5(K)
 51 R5(K)=STOR5(K)-STOR5(K1)+AINF5(K)-EVAP
    PRMAX=PRM3+PRM45
    PUNCH4,AITER,PRMAX,DELT1,DELT2,DELT3,DELT4,DELT5
    PUNCH5,(STOR1(I),I=1,13)
    PUNCH5,(STOR2(I),I=1,13)
    PUNCH5,(STOR3(I),I=1,13)
    PUNCH5,(STOR4(I),I=1,13)
    PUNCH5,(STOR5(I),I=1,13)
    PUNCH1,(AINF3(I),I=1,12)
    PUNCH1,(AINF5(I),I=1,12)
    IF(AITER-3.)800,850,850
850 DELT1=DEL1 $ DELT2=DEL2 $ DELT3=DEL3 $ DELT4=DEL4 $ DELT5=DEL5
    IF(AITER-6.)800,851,851
851 DELT1=DE1 $ DELT2=DE2 $ DELT3=DE3 $ DELT4=DE4 $ DELT5=DE5
    IF(AITER-9.)800,852,852
852 DELT1=D1 $ DELT2=D2 $ DELT3=D3 $ DELT4=D4 $ DELT5=D5
    IF(AITER-12.)800,853,853
853 DELT1=ED1 $ DELT2=ED2 $ DELT3=ED3 $ DELT4=ED4 $ DELT5=ED5
    IF(AITER-15.)800,854,854
854 DELT1=ELD1 $ DELT2=ELD2 $ DELT3=ELD3 $ DELT4=ELD4 $ DELT5=ELD5
    IF(AITER-20.)800,855,855
855 PUNCH1,(R1(I),I=1,12)
    PUNCH1,(R2(I),I=1,12)
    PUNCH1,(R3(I),I=1,12)
    PUNCH1,(R4(I),I=1,12)
    PUNCH1,(R5(I),I=1,12)
    STOP
    END
```

**

```
      SUBROUTINERELP1(S11,S12,K,U)
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      K1=K+1
      IF(S11-SMAX1(K))399,399,400
  399 IF(S11-SMIN1)400,401,401
  401 IF(S12-SMIN1)400,402,402
  402 IF(S12-SMAX1(K1))403,403,400
  403 IF(AINF1(K)+S11-S12)400,404,404
  404 AVST1=(S11+S12)/100.+1.
      I=AVST1 $ I1=I+1 $ B=I
      AVST3=(STOR3(K)+STOR3(K+1))/100.+1.
      J=AVST3 $ J1=J+1 $ C=J
      EVAP=(AREA1(I)+(AVST1-B)*(AREA1(I1)-AREA1(I)))*EVAR1(K)
      RELES=S11-S12+AINF1(K)-EVAP
      IF(RELES-CORE1(K))400,405,405
  405 DREL1=RELES-R1(K)
      REL3=R3(K)+DREL1
      AINF6=REL3+R5(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))350,350,400
  350 IF(REL3-PREM3(K))406,406,407
  406 PREL=REL3 $ GOTO408
  407 PREL=PREM3(K)
  408 U=PREL*(ENGR3(J)+(AVST3-C)*(ENGR3(J1)-ENGR3(J)))
      ENMAX=PRHRS(K)*(CAPA3(J)+(AVST3-C)*(CAPA3(J1)-CAPA3(J)))
      IF(U-ENMAX)409,409,410
  410 U=ENMAX
  409 RETURN
  400 U=-999999.
      RETURN
      END
```

**

```
      SUBROUTINERELP2(S21,S22,K,U)
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      K1=K+1
      IF(S21-SMAX2(K))399,399,400
  399 IF(S21-SMIN2)400,401,401
  401 IF(S22-SMIN2)400,402,402
  402 IF(S22-SMAX2(K1))403,403,400
  403 IF(AINF2(K)+S21-S22)400,404,404
  404 AVST2=(S21+S22)/100.+1.
      I=AVST2 $ I1=I+1 $ B=I
      AVST3=(STOR3(K)+STOR3(K+1))/100.+1.
      J=AVST3 $ J1=J+1 $ C=J
      EVAP=(AREA2(I)+(AVST2-B)*(AREA2(I1)-AREA2(I)))*EVAR2(K)
      RELES=S21-S22+AINF2(K)-EVAP
      IF(RELES-CORE2(K))400,405,405
  405 DREL2=RELES-R2(K)
      REL3=R3(K)+DREL2
      AINF6=REL3+R5(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))350,350,400
  350 IF(REL3-PREM3(K))406,406,407
  406 PREL=REL3 $ GOTO408
  407 PREL=PREM3(K)
  408 U=PREL*(ENGR3(J)+(AVST3-C)*(ENGR3(J1)-ENGR3(J)))
      ENMAX=PRHRS(K)*(CAPA3(J)+(AVST3-C)*(CAPA3(J1)-CAPA3(J)))
      IF(U-ENMAX)409,409,410
  410 U=ENMAX
  409 RETURN
  400 U=-999999.
      RETURN
      END
```

**

```
      SUBROUTINERELP3(S31,S32,K,U)
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      K1=K+1
      IF(S31-SMAX3(K))399,399,400
  399 IF(S31-SMIN3)400,401,401
  401 IF(S32-SMIN3)400,402,402
  402 IF(S32-SMAX3(K1))403,403,400
  403 IF(AINF3(K)+S31-S32)400,404,404
  404 AVST3=(S31+S32)/100.+1.
      J=AVST3 $ J1=J+1 $ C=J
      EVAP=(AREA3(J)+(AVST3-C)*(AREA3(J1)-AREA3(J)))*EVAR3(K)
      RELES=S31-S32+AINF3(K)-EVAP
      IF(RELES)400,405,405
  405 AINF6=RELES+R5(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))350,350,400
  350 IF(RELES-PREM3(K))406,406,407
  406 PREL=RELES $ GOTO408
  407 PREL=PREM3(K)
  408 U=PREL*(ENGR3(J)+(AVST3-C)*(ENGR3(J1)-ENGR3(J)))
      ENMAX=PRHRS(K)*(CAPA3(J)+(AVST3-C)*(CAPA3(J1)-CAPA3(J)))
      IF(U-ENMAX)409,409,410
  410 U=ENMAX
  409 RETURN
  400 U=-999999.
      RETURN
      END
```

**

```
      SUBROUTINERELP4(S41,S42,K,U)
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      K1=K+1
      IF(S41-SMAX4(K))399,399,400
  399 IF(S41-SMIN4)400,401,401
  401 IF(S42-SMIN4)400,402,402
  402 IF(S42-SMAX4(K1))403,403,400
  403 IF(AINF4(K)+S41-S42)400,404,404
  404 AVST4=(S41+S42)/100.+1.0
      I=AVST4 $ I1=I+1 $ B=I
      AVST5=(STOR5(K)+STOR5(K+1))/100.+1.
      J=AVST5 $ J1=J+1 $ C=J
      EVAP=(AREA4(I)+(AVST4-B)*(AREA4(I1)-AREA4(I)))*EVAR4(K)
      RELES=S41-S42+AINF4(K)-EVAP
      IF(RELES)400,405,405
  405 DREL4=RELES-R4(K)
      REL5=R5(K)+DREL4
      AINF6=REL5+R3(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))350,350,400
  350 IF(REL5-PREM5(K))406,406,407
  406 PREL=REL5 $ GOTO408
  407 PREL=PREM5(K)
  408 U5=PREL*(ENGR5(J)+(AVST5-C)*(ENGR5(J1)-ENGR5(J)))
      ENMAX=PRHRS(K)*(CAPA5(J)+(AVST5-C)*(CAPA5(J1)-CAPA5(J)))
      IF(U5-ENMAX)409,409,410
  410 U5=ENMAX
  409 IF(RELES-PREM4(K))411,411,412
  411 PREL=RELES $ GOTO413
  412 PREL=PREM4(K)
  413 U4=PREL*(ENGR4(I)+(AVST4-B)*(ENGR4(I1)-ENGR4(I)))
      ENMAX=PRHRS(K)*(CAPA4(I)+(AVST4-B)*(CAPA4(I1)-CAPA4(I)))
      IF(U4-ENMAX)414,414,415
  415 U4=ENMAX
  414 U=U5+U4
      RETURN
  400 U=-999999.
      RETURN
      END
```

**

```
      SUBROUTINERELP5(S51,S52,K,U)
      DIMENSIONEVAR1(12),EVAR2(12),EVAR3(12),EVAR4(12),EVAR5(12)
      DIMENSIONAREA1(25),AREA2(25),AREA3(25),AREA4(25),AREA5(25)
      DIMENSIONENGR3(25),ENGR4(25),ENGR5(25)
      DIMENSIONCAPA3(25),CAPA4(25),CAPA5(25)
      DIMENSIONAINF1(12),AINF2(12),AINF3(12),AINF4(12),AINF5(12)
      DIMENSIONSTOR1(13),STOR2(13),STOR3(13),STOR4(13),STOR5(13)
      DIMENSIONSMAX1(13),SMAX2(13),SMAX3(13),SMAX4(13),SMAX5(13)
      DIMENSIONR1(12),R2(12),R3(12),R4(12),R5(12),UR6(12)
      DIMENSIONDRMIN(12),DRMAX(12),CORE1(12),CORE2(12)
      DIMENSIONPREM3(12),PREM4(12),PREM5(12),PRHRS(12)
      COMMONSMAX1,SMAX2,SMAX3,SMAX4,SMAX5,SMIN1,SMIN2,SMIN3,SMIN4,SMIN5,
     1CORE1,CORE2,AINF1,AINF2,AINF3,AINF4,AINF5,DELT1,DELT2,DELT3,DELT4,
     2DELT5,AREA1,AREA2,AREA3,AREA4,AREA5,EVAR1,EVAR2,EVAR3,EVAR4,EVAR5,
     3 ENGR3,ENGR4,ENGR5,CAPA3,CAPA4,CAPA5,R1,R2,R3,R4,R5,PREM3,PREM4,
     4PREM5,STOR1,STOR2,STOR3,STOR4,STOR5,UR6,DRMIN,DRMAX,PRHRS
      K1=K+1
      IF(S51-SMAX5(K))399,399,400
  399 IF(S51-SMIN5)400,401,401
  401 IF(S52-SMIN5)400,402,402
  402 IF(S52-SMAX5(K1))403,403,400
  403 IF(AINF5(K)+S51-S52)400,404,404
  404 AVST5=(S51+S52)/100.+1.
      J=AVST5 $ J1=J+1 $ C=J
      AVST4=(STOR4(K)+STOR4(K1))/100.+1.0
      I=AVST4 $ I1=I+1 $ B=I
      EVAP=(AREA5(J)+(AVST5-C)*(AREA5(J1)-AREA5(J)))*EVAR5(K)
      RELES=S51-S52+AINF5(K)-EVAP
      IF(RELES)400,405,405
  405 AINF6=RELES+R3(K)+UR6(K)
      IF(AINF6-DRMIN(K))400,300,300
  300 IF(AINF6-DRMAX(K))350,350,400
  350 IF(RELES-PREM5(K))406,406,407
  406 PREL=RELES $ GOTO408
  407 PREL=PREM5(K)
  408 U5=PREL*(ENGR5(J)+(AVST5-C)*(ENGR5(J1)-ENGR5(J)))
      ENMAX=PRHRS(K)*(CAPA5(J)+(AVST5-C)*(CAPA5(J1)-CAPA5(J)))
      IF(U5-ENMAX)409,409,410
  410 U5=ENMAX
  409 IF(R4(K)-PREM4(K))411,411,412
  411 PREL=R4(K) $ GOTO413
  412 PREL=PREM4(K)
  413 U4=PREL*(ENGR4(I)+(AVST4-B)*(ENGR4(I1)-ENGR4(I)))
      ENMAX=PRHRS(K)*(CAPA4(I)+(AVST4-B)*(CAPA4(I1)-CAPA4(I)))
      IF(U4-ENMAX)414,414,415
  415 U4=ENMAX
  414 U=U5+U4
      RETURN
  400 U=-999999.
      RETURN
      END
```

Appendix E

INPUT DATA FOR JOINT OPERATION

0.55	0.32	0.28	0.31	0.32	0.31		
0.28	0.28	0.34	0.53	0.67	0.75		
0.55	0.32	0.28	0.31	0.32	0.31		
0.28	0.28	0.34	0.53	0.67	0.75		
0.57	0.32	0.28	0.30	0.30	0.27		
0.24	0.30	0.34	0.56	0.71	0.77		
0.68	0.34	0.28	0.30	0.30	0.28		
0.26	0.28	0.38	0.63	0.84	0.94		
0.46	0.28	0.22	0.24	0.29	0.28		
0.27	0.27	0.32	0.48	0.62	0.66		
228.0	228.0	228.0	228.0	228.0	272.0		
272.0	272.0	272.0	272.0	272.0	272.0		
228.0							
830.0	830.0	830.0	830.0	830.0	830.0		
830.0	830.0	830.0	830.0	830.0	830.0		
830.0							
318.0	318.0	318.0	318.0	876.0	876.0		
876.0	876.0	876.0	876.0	876.0	876.0		
318.0							
175.5	175.5	175.5	175.5	175.5	319.4		
319.4	319.4	319.4	319.4	319.4	319.4		
175.5							
597.0	597.0	597.0	597.0	910.0	910.0		
910.0	910.0	910.0	910.0	910.0	910.0		
597.0							
0.0	2.0	3.3	4.3	5.2	6.0	6.8	
0.0	2.1	3.6	5.0	6.2	7.2	8.1	9.1
10.0	10.8	11.6	12.4	13.1	13.8	14.5	15.2
15.8	16.5						
49.00	49.00	76.69	103.48	135.06	134.50	121.87	106.43
91.03	77.37	61.36	49.00	49.00			
170.00	184.62	427.87	745.08	825.14	824.82	801.80	779.31
756.87	729.52	698.26	662.93	616.48			
139.0	194.0	318.0	318.0	565.0	193.0	170.0	176.0
173.0	169.0	163.0	155.0	139.0			
61.00	65.00	117.00	158.00	174.00	174.00	170.00	171.00
127.00	108.00	104.00	110.00	61.00			
136.0	146.0	298.0	597.0	876.0	546.0	513.0	438.0
404.0	339.0	252.0	173.0	136.0			
20.00	28.00	31.00	27.00	22.00	15.00		
16.00	16.00	14.00	16.00	13.00	4.00		
18.00	18.00	18.00	190.00	45.00	18.00		
18.00	18.00	22.00	23.00	25.00	35.00		
6.09	83.24	289.07	168.56	538.17	37.79		
7.21	.73	1.53	2.18	3.60	11.93		
1.99	1.03	0.78	49.25	53.16	9.37		
1.57	51.63	18.33	1.92	0.52	52.82		
55.45	1.73	89.88	94.11	532.22	38.48		
73.00	82.35	80.07	84.06	76.16	97.07		
0.0	144.0	250.0	145.0	82.0	0.0		
0.0	0.0	0.0	0.0	0.0	0.0		
23.0	57.0	59.0	60.0	23.0	4.0		
2.0	2.0	2.0	3.0	4.0	8.0		
39.0	265.0	339.0	275.0	50.0	0.0		
0.0	0.0	0.0	0.0	0.0	0.0		
65.0	211.0	293.0	421.0	171.0	17.0		
15.0	0.0	0.0	0.0	0.0	0.0		
9.0	55.0	44.0	68.0	56.0	8.0		
5.0	10.0	2.0	2.0	12.0	9.0		

68.0	156.0	392.0	378.0	208.0	10.0
2.0	52.0	19.0	2.0	2.0	64.0
0.0	0.0	0.0	2.9	0.0	0.0
3.3	35.0	53.0	5.4	37.5	62.0
9.8	41.0	68.5	11.5	43.0	73.0
13.4	44.0	78.5	15.1	44.0	81.5
16.8	44.0	84.5	18.4	44.0	86.5
20.0	44.0	88.5	21.5	44.0	90.5
22.9	44.0	92.0	24.3	44.0	93.5
25.9	44.0	94.5	27.6	44.0	95.5
29.3	44.0	96.5	31.1	44.0	97.0
33.1	44.0	97.5	35.2	44.0	98.0
37.3	44.0	98.0	39.5	44.0	98.0
0.0	0.0	0.0	3.8	0.0	0.0
6.2	2.68	37.5	8.5	3.24	45.2
10.5	3.51	49.2	12.4	3.68	52.0
14.0	3.94	53.7	15.4	4.00	55.5
0.0	0.0	0.0	2.4	0.0	0.0
4.3	0.0	0.0	5.8	43.8	75.0
7.4	50.4	82.5	9.0	54.3	88.0
10.4	57.0	94.0	11.7	58.8	96.0
13.0	60.0	99.0	14.2	60.0	101.0
15.4	60.0	104.0	16.6	60.0	106.0
17.7	60.0	107.0	18.7	60.0	108.5
19.7	60.0	110.0	20.6	60.0	111.5
21.5	60.0	112.5	22.4	60.0	113.0
23.4	60.0	113.5	24.5	60.0	114.0
25.6	60.0	114.5	26.7	60.0	115.0
20.0	28.0	31.0	27.0	22.0	15.0
16.0	16.0	14.0	16.0	13.0	4.0
18.0	18.0	18.0	18.0	18.0	18.0
18.0	18.0	22.0	23.0	25.0	35.0
58.0	29.0	206.0	382.0	1147.0	72.0
80.0	80.0	78.0	80.0	73.0	66.0
15000.0	15500.0	15500.0	15000.0	15500.0	15000.0
15500.0	15500.0	14000.0	15500.0	15000.0	15500.0
411.30	425.01	425.01	411.30	425.01	411.30
425.01	425.01	383.88	425.01	411.30	425.01
51.8	53.5	53.5	51.8	53.5	51.8
53.5	53.5	48.4	53.5	51.8	53.5
428.40	442.68	442.68	428.40	442.68	428.40
442.68	442.68	397.84	442.68	428.40	442.68
720.0	744.0	744.0	720.0	744.0	720.0
744.0	744.0	672.0	744.0	720.0	744.0
49.0	170.0	139.0	61.0	136.0	
50.0	100.0	100.0	50.0	100.0	
20.0	50.0	50.0	20.0	50.0	
10.0	10.0	20.0	10.0	20.0	
5.0	5.0	10.0	5.0	10.0	
2.0	2.0	5.0	2.0	5.0	
1.0	1.0	2.0	1.0	2.0	
	0.0		0.0		

Appendix F

JOINT OPERATION OUTPUT

	90415.93						
1.00		217238.20	50.00	100.00	100.00	50.00	100.00
170.00	184.62	427.87	745.08	725.14	724.82	701.80	
679.31	656.87	629.52	598.26	562.93	516.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	170.00	
171.00	127.00	108.00	104.00	110.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	212.11	293.17	521.29	171.56	17.46		
15.46	.50	.69	1.21	1.73	2.27		
68.00	156.00	392.00	378.00	208.00	10.00		
2.00	52.00	19.00	2.00	2.00	64.00		
	99231.16						
2.00		226053.43	50.00	100.00	100.00	50.00	100.00
170.00	184.62	427.87	745.08	625.14	624.82	601.80	
579.31	556.87	529.52	498.26	462.93	416.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	170.00	
171.00	127.00	108.00	104.00	110.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	212.11	293.17	621.50	172.01	17.90		
15.85	.89	1.17	1.99	2.76	3.47		
68.00	156.00	392.00	378.00	208.00	10.00		
2.00	52.00	19.00	2.00	2.00	64.00		
	107530.88						
3.00		234353.15	50.00	100.00	100.00	50.00	100.00
170.00	184.62	427.87	645.08	525.14	524.82	501.80	
479.31	456.87	429.52	398.26	362.93	316.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	170.00	
171.00	127.00	108.00	104.00	110.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	212.11	393.37	621.94	172.48	18.35		
16.28	1.33	1.71	2.84	3.86	4.77		
68.00	156.00	392.00	378.00	208.00	10.00		
2.00	52.00	19.00	2.00	2.00	64.00		
	111420.56						
4.00		239031.34	20.00	50.00	50.00	20.00	50.00
170.00	184.62	377.87	595.08	475.14	474.82	451.80	
429.31	406.87	379.52	348.26	312.93	266.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	150.00	
151.00	107.00	88.00	84.00	90.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	

438.00	404.00	339.00	252.00	173.00	136.00		
68.26	262.26	393.60	622.16	172.73	18.60		
16.50	1.55	1.99	3.30	4.49	5.51		
68.00	156.00	392.00	378.00	208.00	30.11		
2.21	52.26	19.35	2.59	2.79	44.45		
	115315.31						
5.00		243725.83	20.00	50.00	50.00	20.00	50.00
170.00	184.62	327.87	545.08	425.14	424.82	401.80	
379.31	356.87	329.52	298.26	262.93	216.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
131.00	87.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	312.41	393.82	622.41	172.99	18.85		
16.73	1.80	2.30	3.81	5.13	6.20		
68.00	156.00	392.00	378.00	208.00	50.22		
2.45	52.51	19.72	3.20	3.60	24.90		
	115315.31						
6.00		243725.83	20.00	50.00	50.00	20.00	50.00
170.00	184.62	327.87	545.08	425.14	424.82	401.80	
379.31	356.87	329.52	298.26	262.93	216.48		
139.00	194.00	318.00	318.00	565.00	193.00	170.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
131.00	87.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	876.00	546.00	513.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	312.41	393.82	622.41	172.99	18.85		
16.73	1.80	2.30	3.81	5.13	6.20		
68.00	156.00	392.00	378.00	208.00	50.22		
2.45	52.51	19.72	3.20	3.60	24.90		
	116217.24						
7.00		244745.83	10.00	10.00	20.00	10.00	20.00
170.00	184.62	337.87	535.08	415.14	414.82	391.80	
369.31	346.87	319.52	288.26	252.93	206.48		
139.00	194.00	318.00	318.00	565.00	193.00	190.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
131.00	87.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	856.00	526.00	513.00	
458.00	424.00	359.00	252.00	173.00	136.00		
68.26	302.38	413.82	622.46	173.04	18.90		
16.78	1.85	2.37	3.92	5.25	6.35		
68.00	156.00	392.00	378.00	208.00	50.22		
2.45	52.51	19.72	3.20	3.60	24.90		

	117011.77						
8.00		245565.41	10.00	10.00	20.00	10.00	20.00
170.00	184.62	327.87	525.08	415.14	414.82	381.80	
359.31	336.87	309.52	278.26	242.93	196.48		
139.00	194.00	318.00	318.00	565.00	193.00	190.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
131.00	87.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	836.00	506.00	493.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	312.41	413.87	612.49	173.04	28.92		
16.83	1.90	2.43	4.02	5.37	6.50		
68.00	156.00	392.00	378.00	208.00	50.22		
2.45	52.51	19.72	3.20	3.60	24.90		
	117796.45						
9.00		246350.09	10.00	10.00	20.00	10.00	20.00
170.00	184.62	317.87	515.08	405.14	404.82	371.80	
349.31	326.87	299.52	268.26	232.93	186.48		
139.00	194.00	318.00	318.00	565.00	193.00	190.00	
176.00	173.00	169.00	163.00	155.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
131.00	87.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	836.00	506.00	493.00	
438.00	404.00	339.00	252.00	173.00	136.00		
68.26	322.43	413.91	612.54	173.09	28.98		
16.88	1.95	2.50	4.12	5.49	6.65		
68.00	156.00	392.00	378.00	208.00	50.22		
2.45	52.51	19.72	3.20	3.60	24.90		
	118704.80						
10.00		247431.01	5.00	5.00	10.00	5.00	10.00
170.00	184.62	317.87	515.08	400.14	404.82	366.80	
344.31	321.87	294.52	263.26	227.93	181.48		
139.00	194.00	318.00	318.00	565.00	193.00	200.00	
166.00	163.00	159.00	153.00	145.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
126.00	82.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	826.00	496.00	493.00	
448.00	414.00	349.00	242.00	173.00	136.00		
68.26	322.43	413.91	617.55	168.10	33.99		
16.91	1.98	2.54	4.16	5.56	6.72		
68.00	156.00	392.00	378.00	208.00	50.22		
7.48	52.58	14.77	3.20	3.60	24.90		
	119097.37						
11.00		247908.37	5.00	5.00	10.00	5.00	10.00
170.00	184.62	312.87	510.08	395.14	399.82	361.80	
339.31	316.87	289.52	258.26	222.93	176.48		
139.00	194.00	318.00	318.00	565.00	193.00	200.00	
166.00	163.00	159.00	153.00	145.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
121.00	82.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	816.00	486.00	483.00	
458.00	424.00	359.00	242.00	173.00	136.00		

68.26	327.45	413.93	617.57	168.13	34.02		
16.93	2.01	2.57	4.21	5.63	6.80		
68.00	156.00	392.00	378.00	208.00	50.22		
12.51	47.61	14.77	3.20	3.60	24.90		
	119490.97						
12.00		248301.97	5.00	5.00	10.00	5.00	10.00
170.00	184.62	307.87	505.08	390.14	394.82	356.80	
334.31	311.87	284.52	253.26	217.93	171.48		
139.00	194.00	318.00	318.00	565.00	193.00	200.00	
166.00	163.00	159.00	153.00	145.00	139.00		
61.00	65.00	117.00	158.00	174.00	174.00	130.00	
121.00	82.00	68.00	64.00	70.00	61.00		
136.00	146.00	298.00	597.00	816.00	486.00	483.00	
458.00	424.00	359.00	242.00	173.00	136.00		
68.26	332.47	413.96	617.60	168.16	34.05		
16.96	2.04	2.60	4.26	5.69	6.89		
68.00	156.00	392.00	378.00	208.00	50.22		
12.51	47.61	14.77	3.20	3.60	24.90		
	119721.92						
13.00		248627.48	2.00	2.00	5.00	2.00	5.00
170.00	186.62	307.87	505.08	388.14	394.82	354.80	
332.31	309.87	282.52	251.26	217.93	171.48		
139.00	194.00	318.00	318.00	565.00	193.00	200.00	
161.00	158.00	154.00	153.00	145.00	139.00		
61.00	65.00	117.00	158.00	172.00	172.00	128.00	
121.00	80.00	66.00	62.00	70.00	61.00		
136.00	146.00	298.00	597.00	811.00	481.00	478.00	
458.00	424.00	359.00	242.00	173.00	136.00		
66.25	334.46	413.96	619.60	166.16	36.06		
16.97	2.05	2.61	4.28	3.71	6.89		
68.00	156.00	392.00	380.02	208.02	50.25		
10.52	49.62	14.80	3.26	1.64	24.90		
	119749.95						
14.00		248701.78	2.00	2.00	5.00	2.00	5.00
170.00	186.62	305.87	503.08	386.14	392.82	352.80	
330.31	307.87	280.52	251.26	217.93	171.48		
139.00	194.00	318.00	318.00	565.00	193.00	200.00	
161.00	158.00	154.00	153.00	145.00	139.00		
61.00	65.00	117.00	158.00	172.00	172.00	130.00	
123.00	78.00	64.00	62.00	70.00	61.00		
136.00	146.00	298.00	597.00	806.00	476.00	473.00	
453.00	424.00	359.00	242.00	173.00	136.00		
66.25	336.47	413.96	619.61	166.18	36.07		
16.98	2.06	2.63	2.29	3.71	6.89		
68.00	156.00	392.00	380.02	208.02	48.23		
10.50	53.62	14.84	1.29	1.64	24.90		