AL-IV

Water Supply Allocation Model

Program Documentation
And
Users Manual

TEXAS WATER DEVELOPMENT BOARD

September 1975

PROGRAM DOCUMENTATION
AND
USER'S MANUAL

Water Supply Allocation Model

AL-IV

Prepared By
Systems Engineering Division
of the
Texas Water Development Board

September 1975

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

LIST OF TABLES

LIST OF FIGURES

PREFACE

The AL-IV Model described in this document is a computer program that simulates and optimizes the operation of an interconnected system of reservoirs, pump canals, pipelines, and river reaches.

AL-IV is an improved version of the previously developed Allocation Model (TWDB, 1972). The major improvement to the previous model (AL-III) consisted of incorporating reservoir evaporation, channel seepage, and consumptive use directly into the Allocation Model as system variables. In past versions of the model, it was required that the water losses from these processes be specified in advance since the solution procedure for the network model did not allow water losses which were flow dependent. The current version of the Allocation Program represents the physical system in a more realistic manner by allowing water losses in impoundments and channels to vary with storage and discharge levels.

The improved representation of the physical system was made possible by adapting the Allocation Program to utilize a recently developed network optimization algorithm written by Dr. Paul Jensen and Dr. Gora Bhaumik. Their Network With Gains Algorithm replaced the Out-Of-Kilter Algorithm used in AL-III.

The Allocation Model was originally developed by Water Resources Engineers, Inc. under contract to the Texas Water Development Board, with subsequent improvements and modifications made by the staff of the Texas Water Development Board. The current version of the program was adapted from AL-III by Dr. Quentin W. Martin of the Systems Engineering Division of the Texas Water Development Board.

The model was developed on the UNIVAC 1108/1106 computer systems but is designed to be essentially machine independent. Core requirements are approximately 41,000 decimal words. Computation time is a function of the size of the problem being analyzed; however, for a two-year network such as the example problem discussed herein, approximately 25 seconds of UNIVAC 1106 execution time is required to obtain a solution. Any inquires concerning the use of this program should be directed to Dr. Quentin W. Martin.

ALLOCATION MODEL

PURPOSE

The water resources allocation program, AL-IV, is a general hydrologic optimization model of surface water resources systems. It is designed to analyze the simulated multi-period operation of any interconnected configuration of reservoirs, pump canals, and pipelines on a steady state monthly or seasonal basis. The program may be utilized by itself to find:

1) the minimum cost operating plan for a system of reservoirs, river junctions, canals, and river reaches,

2) the minimum cost sizing of individual reservoirs, canals, or closed conduits,

3) the reservoir operating rule coefficients for use in the SIM-IV and SIMYLD-II Models (4);

or jointly with the DPSIM program (5) to determine

4) the minimal cost construction sizing and sequencing of a number of water storage and conveyance projects in a multiple purpose river basin system.

CONCEPTS

The general concept behind this program is to describe the structure of a surface water resources system in terms of a network flow model. The minimum cost operation of the prototype system is thus established by solving the associated network representation for the least cost set of flows through the network. The structure of this network model is described in the following paragraphs.

First, the physical system is represented in space by a node-link configuration. For the physical system, nodes represent either storage reservoirs or non-storage link junctions, and links portray either river reaches or canals. An example of such a spatial representation is shown in Figure I. This example system consists of six nodes (four reservoirs and two link junctions) and eight links (seven canals and one river reach). All reservoirs that do not have a river reach leaving them must have an outlet for spilling any excess water that enters them. These spills leave the network and are no longer available for use. The spill outlets are also illustrated in Figure I.

Next, this spatial representation is expanded to include time. For each time period in the problem, there is a corresponding node-link representation. The representations are connected by the rates at which reservoir storage contents are carried forward in time. These connections are referred to as "storage arcs" (arcs refer to all node connections in the problem, including canal and river links). Thus, the time-space representation of the problem can be envisioned as a layered network, each layer representing a time period with storage arcs connecting the layers. The example system illustrated in Figure I, expanded to include four time periods, is shown in Figure II.

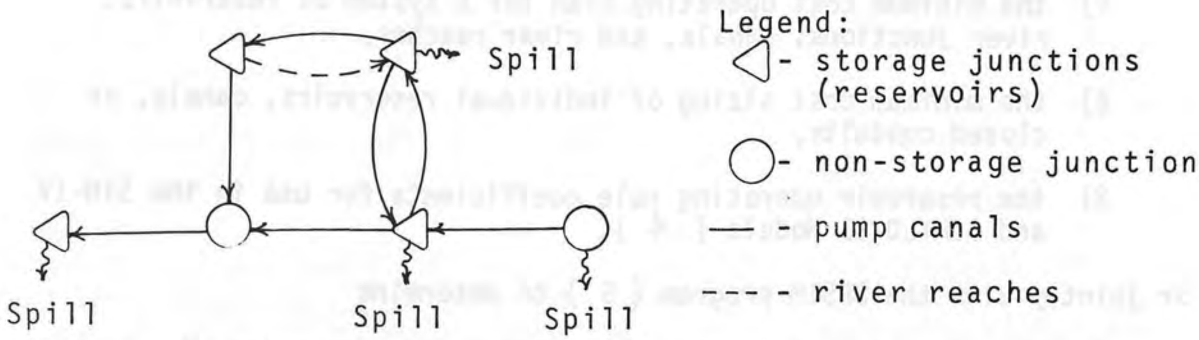


Figure I
Spatial Representation of the System Configuration

This expanded network still does not completely represent the problem. The system must have initial reservoir storage contents; inputs to and demands from the system must be made; imports must be allowed to enter and spills permitted to leave the system; and following the last time period, specifications must be made for the final reservoir storage contents. However, all of these considerations are accomodated by adding additional arcs and nodes. The complete network, including these arcs and nodes, is as shown schematically in Figure III.

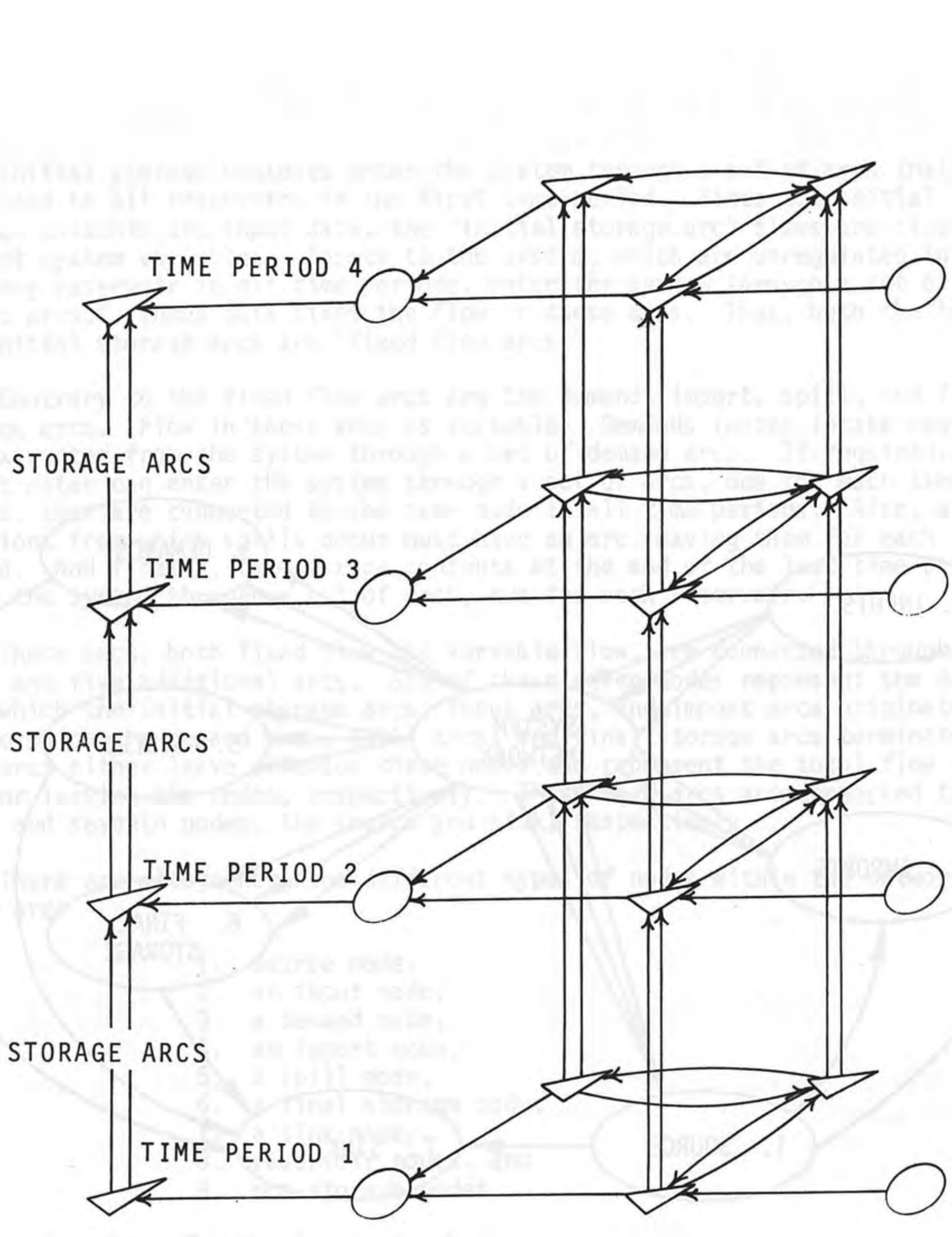


Figure II
Spatial Representation of the System Configuration Expanded to Include Four Time Periods

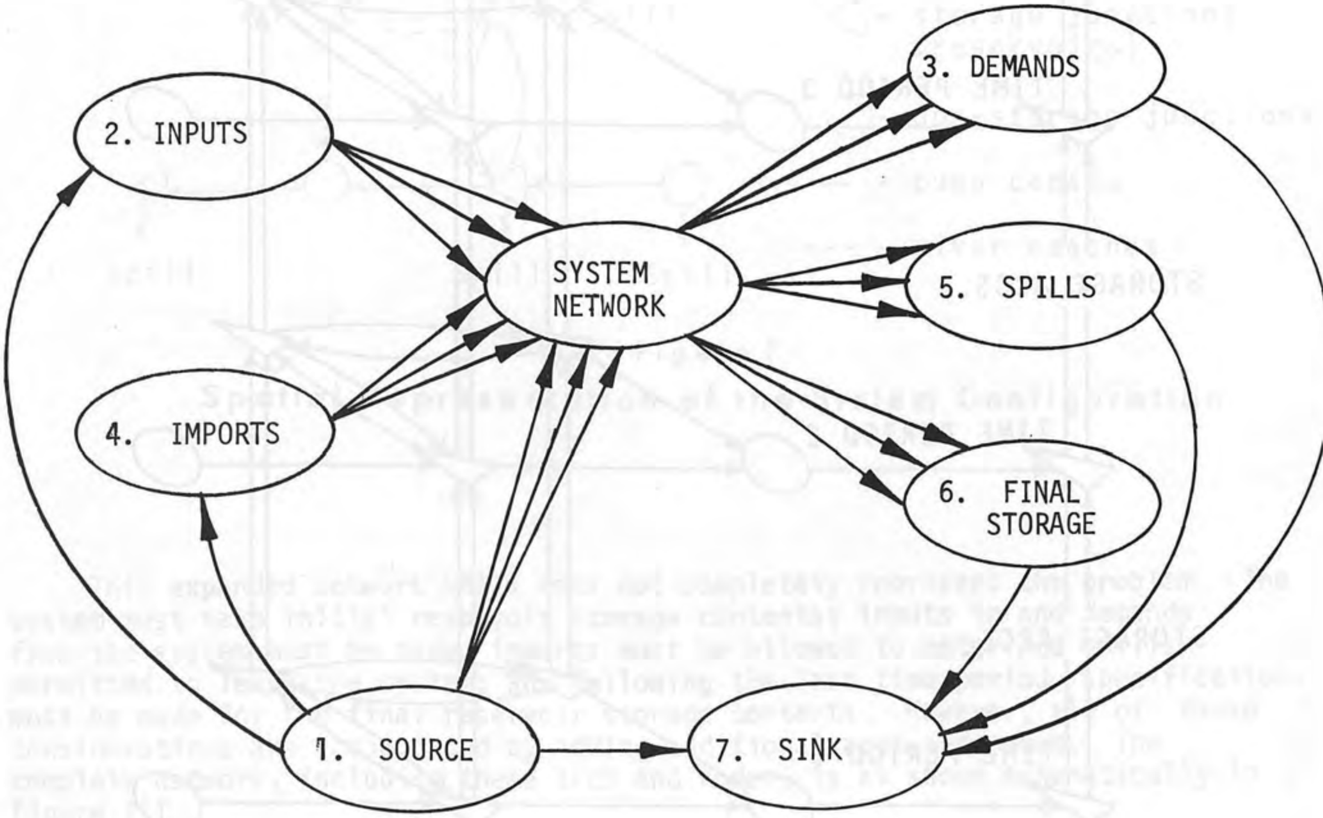


FIGURE III

THE NETWORK FOR THE ALLOCATION PROBLEM

Initial storage contents enter the system through a set of arcs that are connected to all reservoirs in the first time period. Since the initial storage contents are input data, the "initial storage arc" flows are fixed and are not system variables. Inputs to the system, which are unregulated inflows to every reservoir in all time periods, enter the system through a set of "input arcs." Input data fixes the flow in these arcs. Thus, both the input and initial storage arcs are "fixed flow arcs."

Contrary to the fixed flow arcs are the demand, import, spill, and final storage arcs. Flow in these arcs is variable. Demands (water intake requirements) are extracted from the system through a set of demand arcs. If required, import water can enter the system through a set of arcs, one for each time period, that are connected to the same node in all time periods. Also, all junctions from which spills occur must have an arc leaving them for each time period. And finally, the storage contents at the end of the last time period leave the system through a set of arcs, two for each reservoir.

These arcs, both fixed flow and variable flow, are connected through seven nodes and five additional arcs. Six of these seven nodes represent the nodes from which the initial storage arcs, input arcs, and import arcs originate, and to which the demand arcs, spill arcs, and final storage arcs terminate. The five arcs either leave or enter these nodes and represent the total flow going into or leaving the nodes, respectively. These five arcs are connected to the sixth and seventh nodes, the source and sink, respectively.

There are altogether nine different types of nodes within the network. These are:

1. source node,
2. an input node,
3. a demand node,
4. an import node,
5. a spill node,
6. a final storage node,
7. a sink node,
8. reservoir nodes, and
9. non-storage nodes.

The total number of nodes in a network is

$$N = Ln_n + 7$$

where L is the number of time periods in the problem, n_n is the number of nodes in the spatial representation of the problem (reservoirs plus non-storage nodes), and 7 is the number of special nodes in the problem (items 1 through 7 above).

Connecting these nine nodes are ten different types of arcs; they are listed in Table I. Flow in these arcs is constrained to be within lower and upper limits. These limits are also summarized in Table I.

Table I
Arc Types and Definitions of Their Upper and Lower Bounds

	ARC TYPE	LOWER BOUND	UPPER BOUND
1.	River	Minimum Required Flow	River Capacity
2.	Canal/Pipeline	Minimum Required Flow	Canal Capacity
3.	Storage	Zero	Reservoir Capaci Target Storage
4.	Target Storage	Minimum Storage	Target Reservoir Storage
5.	Initial Storage	Initial Storage	Initial Storage
6.	Input	Unregulated Inflow	Unregulated Infl
7.	Demand	Required Portion of Demand to be Met	Water Demand
8.	Import	Zero	Maximum Availabl
9.	Spill	Zero	Maximum Permissi
10.	Net Balance		
	a. Total Inputs	Σ Inputs	Σ Inputs
	b. Total Imports	Zero	Σ Maximum Import
	c. Total Demands	Zero	Σ Demands
	d. Total Spills	Zero	Σ Maximum Spills
	e. Total Final Storage	Zero	Σ Reservoir Capacities
	f. Flow By-Pass	Zero	Large Positive Value

Flow in river arcs is permitted to be between zero and the maximum river capacity. If there are low flow requirements for water quality control or other purposes, the lower limit can be set to satisfy these needs. Canal flows and reservoir storage contents are both constrained to be between zero and their design capacities. This upper limit can also be used to stage or time their addition to the system, which is accomplished by placing an upper limit of zero on those elements not yet built and, in the year each is constructed, increasing the constraint to the design capacity of the element.

Both the initial storage arcs and the input arcs have equal upper and lower bounds. This forces the initial storage contents and inputs of the system to be constants. The demands (water intake requirements) withdrawn from the system can range from a lower bound that permits a tolerable shortage to the maximum needed. The amount of shortage allowed can be different for every month and every node. The amount of water imported is allowed to be anywhere between zero and the maximum quantity available. The latter depends on the season of the year. Flow in spill arcs is limited to between zero and an arbitrarily high value that will not be reached.

An additional arc, called a flow by-pass link, is inserted into the network between the flow source and sink. The solution technique used to solve the allocation network model requires that a fixed flow level be supplied to the sink. The by-pass arc supplies, directly from the source node, whatever portion of the water requirement at the sink that is not furnished through the system network. Since it is important to put as much beneficial flow as possible through the system network, a small unit cost ($.001) is placed on flow through the by-pass arc. Thus flow will go through the by-pass link only after the net cost to put additional flow through the system network exceeds $.001 per unit of flow.

The net balance arcs have constraints equal to the sum of the constraints on their components. For example, the bounds on the total input arc is equivalent to the sum of the bounds on the individual input arcs.

The total number of arcs in the network to be solved by the AL-IV Model is expressed as

$$\text{Number of Arcs} = [(n_L + 2\, n_r + 2n_n + n_s + 1) \times L] + 6 + n_r$$

where n_L is the number of links (canals plus river reaches); n_r is the number of reservoir nodes; n_n is the number of storage and/non-storage junctions; n_s is the number of nodes from which spills can occur; 1 represents the single import arc in each time period; L is the number of time periods in the problem; and 6 represents the number of net balance arcs.

If monthly time increments are used and the problem contains a large number of nodes and links, the computer storage requirements can become excessive. When this occurs, the Allocation Model can span only a few years at one time. In this case, the number of years spanned is the number of years in the network. For explanatory purposes, assume the total problem involves a ten-year period and the maximum number of years the network can span is four. The first problem solved would involve only the first four years of the total ten-year period, and produce a valid solution for the first year. The first year is then deleted from the network and the fifth year added. A solution for the second year is obtained from this problem. This process is repeated until the network problem formed by the last four of the ten years has been solved.

With this concept of structuring the problem in terms of a network, one can seek a solution that will minimize cost yet satisfy the inputs to and demands from the system.

MODELING ASSUMPTIONS

The abstraction of a prototype physical system into a mathematical representation usually requires numerous assumptions. These assumptions thus limit the generality of the resulting model and place restrictions on its valid use. The major assumptions of the network approach utilized by the Allocation program are described below.

- Only surface waters are modeled. That is, no water quality parameters or conjunctive use of groundwater is included in the modeling capability.

- Monthly time increments are the shortest time increments which may be used to simulate the system; thus, operations of canals and reservoirs for routing flood waves is not considered.

- All demands for and inputs of water must be pre-specified except for the case of import waters where the maximum available will be pre-specified. Thus, runoff, evaporation rates, and intake demands for water are forced upon the system, but import water is drawn upon only when needed.

- Import can occur at any one storage or non-storage junction in the system during any limited part of the year up to the maximum monthly availability that was pre-specified. Import water has a constant unit cost.

- A minimum cost objective function is used in conjunction with penalty cost functions. Thus, if these two parameters are properly used, a net benefits maximization criterion can be imposed.

- Because an economic objective criterion is specified, a pre-specified economic value for meeting demands versus the economic value of spilling water and the economic value of storing water is required. Therefore, it is assumed that demands for water will be met only if the value for meeting demands is greater than the penalty for not meeting them. The value of having water in specific reservoirs on a seasonal basis can be specified.

- Unit penalty costs for incurred shortages can be varied by node by season whereas storage arc pricing preferences can be varied by reservoir by year.

- Demands for water, reservoir inflow quantities, and evaporation rates are capable of being varied on a month-by-month basis to permit accounting for a demand build-up, a runoff depletion, and stochastic variability in all of these quantities.

- Canal costs must be divided into two components - that component which cannot be staged (e.g. ditch and right-of-way costs) and that component which can be staged (e.g. pump, motor, and housing costs).

- Both reservoirs and canals can be added to the network of active facilities at any given year in the simulation period.

- Both minimum and maximum flow and storage capacities must be specified for canals and reservoirs, respectively.

- The physical system is represented by a set of interconnected nodes and links. Links correspond to river reaches, pump-canals and pipelines, while nodes represent reservoir and link junction points.

- All demands for water and runoff quantities occur at nodes and reservoirs.

- Canal evaporation losses are computed as a percentage of the flow along the canal.

- Initial storage contents of all reservoirs are known.

- Lower bound constraints can be set on demand arcs to reflect, at each node, how much of a pre-specified demand must be met regardless of the magnitude of shortages incurred. If the lower bounds are set too high, an infeasible solution may result.

- Spills out of the system can be controlled to occur only at those reservoirs specified as spill nodes.

- The resolution of modeling accuracy is currently set at 1000 acre-feet as supplied via input data.

- Only reservoir storage allocated for "conservation" purposes can be used for reregulation.

- The unit cost of pumping is the product of a monthly power cost per unit of flow per foot of lift and a constant lift. These parameters are under input control.

- A pumping efficiency of .8 is assumed for all artifical conveyance links.

MATHEMATICAL DESCRIPTION

The mathematic statement of the allocation problem may be expressed as

$$\text{minimize} \sum_{i=1}^{N} \sum_{j=1}^{N} q_{ij} \; C_{ij} \qquad [1]$$

subject to

material balance at all nodal points,

$$\sum_{i=1}^{N} q_{ij} - q'_{ji} = 0 \;, \; j=1, \ldots, N \qquad [2]$$

material losses along links,

$$q'_{ij} - (1 - \gamma_{ij}) \, q_{ij} = 0 \;, \text{ for all i and j,} \qquad [3]$$

and bounds on material conveyed along links,

$$L_{ij} \leq q_{ij} \leq U_{ij} \;, \text{ for all i and j,} \qquad [4]$$

where the terms are as defined in Table II.

The mathematical structure is described by four sets of constraint equations and an objective function. One set of constraint equations requires that continuity be satisfied at all nodes in the network, except at the source and sink nodes. The second set of constraints indicates the amount of water lost by conveyance through each of the arcs. The remaining two sets of equations describe the upper and lower limits on flow in all arcs in the network. Thus, there is one equation for each node and three equations for each arc.

Using the terminology defined in Table II, continuity equations for the nodes listed on page 12, with the exception of the source and sink nodes, can be written as

1. Input Node:

$$\sum_{k=1}^{L} \sum_{j=1}^{n} \alpha_{jk} = X_i \quad \text{(Inflow from Source Node)}$$

2. Demand Node:

$$\sum_{k=1}^{L} \sum_{j=1}^{n} D_{jk} = X_d \quad \text{(Outflow to Sink Node)}$$

3. Import Node:

$$\sum_{k=1}^{L} \sum_{j=1}^{n} \delta_j \; I_k = X_m \qquad \text{(Inflow from Source Node)}$$

4. Spill Node:

$$\sum_{k=1}^{L} \sum_{j=1}^{n} \Theta_j \; P_{jk} = X_s \qquad \text{(Outflow to Sink Node)}$$

5. Final Storage Node:

$$\sum_{j=1}^{n_r} \frac{S_{j,L+1}}{\Delta t} = X_f \qquad \text{(Outflow to Sink Node)}$$

6. Reservoir Nodes:

$$\sum_{i=1}^{N} Q'_{ijk} - \sum_{i=1}^{N} Q_{jik} - \Theta_j \; P_{jk} - \frac{S_{j,k+1}}{\Delta t} + \frac{S_{jk}}{\Delta t} +$$

$$\delta_j \; I_k + \alpha_{jk} - D_{jk} - \beta_{ij} = 0$$

$$j = 1,2,\ldots,n_r \qquad k = 1,2,\ldots,L$$

7. Link Junction Nodes:

$$\sum_{i=1}^{N} Q'_{ijk} - \sum_{i=1}^{N} Q_{jik} - \Theta_j \; P_{jk} + \delta_j \; I_k - D_{jk} + \alpha_{jk} = 0$$

$$j = n_r + 1, n_r + 2,\ldots,n \qquad k = 1,2,\ldots,L$$

All of these equations can be reduced to the common form

$$\sum_{i=1}^{N} q'_{ij} - \sum_{i=1}^{N} q_{ji} = 0 \qquad j = 1,\ldots, n.$$

Table II

Definition of Terms in the AL-IV Program

NETWORK FLOW PROBLEM			UNITS*
q_{ij}	=	flow entering arc from node i to node j	l^3/t
q'_{ij}	=	flow leaving arc from node i to node j	l^3/t
L_{ij}	=	lower bound on flow from node i to node j	l^3/t
U_{ij}	=	upper bound on flow from node i to node j	l^3/t
C_{ij}	=	cost of moving one unit of flow from node i to node j	$\$/l^3/t$
N	=	number of nodes in the network	

NODE BALANCE EQUATIONS			
Q'_{ijk}	=	flow into reservoir or link junction i from reservoir or link junction j in time period k	l^3/t
Q_{ijk}	=	flow out of reservoir or link junction i toward reservoir or link junction j in time period k	l^3/t
P_{jk}	=	rate of spill from reservoir or link junction j in time period k	l^3/t
S_{jk}	=	storage contents of reservoir j at the beginning of time period k	l^3
I_k	=	rate of importation of water in time period k	l^3/t
D_{jk}	=	rate of demand at node j in time period k	l^3/t

α_{jk}	=	rate of input to junction j (unregulated inflow)	l^3/t
β_{jk}	=	rate of water loss due to evaporation from reservoir j in time period k	l^3/t
Θ_j	=	1, if j is a spill node 0, if j is not a spill node	
δ_j	=	1, if j is an import node 0, if j is not an import node	
γ_{ij}	=	Fraction of flow lost along the arc from node i to node j	

SUBSCRIPTS AND SUBSCRIPT LIMITS

i,j	=	nodes
n_r	=	number of reservoirs
n	=	number of reservoirs plus number of link junctions
k	=	time period
L	=	number of time periods

---

* l is used to designate unit of length, and
t is used to designate unit of time
(e.g. l^3/t designates volume per unit time)

In this basic equation q_{ij} represents the flow leaving node i along an arc connecting it with node j and, similarly, q'_{ij} represents the flow entering node j along an arc from any other node i. Flows entering and leaving a node occur only through the arcs connected to it. Flow entering these arcs is constrained to be within a range defined by the arc's lower and upper limits. These constraints can be expressed as

$$q_{ij} \geq L_{ij} \qquad \text{and} \qquad q_{ij} \leq U_{ij}.$$

The constraints vary considerably depending upon what the arc flow represents. As described previously, there are ten arc types (see Table I) each having different limits.

The objective function that must be satisfied while solving these equations is the minimization of the cost of transferring water through the network. This is expressed as

$$\text{Minimize } Z = \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij} \quad q_{ij} .$$

In this network, the canal, demand, storage and spill arcs are the only types of arcs which have cost assoicated with them. In the canal arc the unit cost represents the cost of power needed to pump one unit of flow while in the demand arc there is a negative unit cost associated with each unit of deficit. Reservoir storage arcs take on user-designated costs according to the operating rule option desired. Similarly, the unit spillage cost is specified by the user; however, to prevent reservoirs from spilling when they could be storing water, a minimum unit cost of $.001 per thousand acre-feet is assigned to water released from the system.

SOLUTION METHODOLOGY

The multi-period network flow optimization problem described above represents a more general problem formulation than was considered in previous versions of the Allocation program. The prior Allocation models required that flow be conserved throughout the network, hence the flow out of the source node was specified to equal the flow into the sink node. Under this assumption, the network flow problem specified in the previous section (without constraint [3]) was rapidly and efficiently solved using the "Out-of-Kilter" Algorithm (OKA) (2) to determine the minimum-cost operating policy for the system. The imposition of this complete material continuity assumption for the entire network, however, restricted the ability of the previous Allocation programs to accurately account for water losses which are dependent upon the channel flows. Such losses occur in actual water distribution systems through evaporation, channel seepage or consumptive use. These leakages were treated in the previous Allocation programs by estimating the losses and adjusting the input hydrology for the network accordingly. While this procedure is reasonably accurate for single-period (monthly) network models considerable error may occur when multiple period network problems are solved using the OKA.

Recent advances in network flow theory by Jensen and Bhaumik (3) have extended the capabilities of network models to incorporate consideration of leakage mechanisms which are flow dependent. The primary theoretical advancement has been development of computationally efficient algorithms for finding the minimum cost flow circulation in network models with gains. A network with gains is simply a standard (or pure) network with an additional descriptive parameter associated with each arc. This added parameter is termed the gain factor of the arc and has value equal to the ratio between the flow leaving and the flow entering the arc. A link having a gain of .9, for instance, will transmit 90% of its inflow, while losing the remaining 10% to leakage.

The value assigned to an arc's gain factor will depend upon the phenomenon causing the water loss. For water conveyance links, the complement of the gain should represent the net losses (evaporation plus ground water recharge) per unit of flow. For network arcs corresponding to consumer water demands, the value of the gain factor should be set to that of the user's S/W ratio. The treatment is somewhat more complicated for the losses due to reservoir evaporation since evaporation is proportional to the water surface area and not directly proportional to storage volume. However, by approximating the capacity - area curve for each reservoir with two continuous piece-wise linear segments, evaporation losses can be expressed as proportional to reservoir storages. To illustrate, the capacity-acre curve for the Cuero I Reservoir in the Guadalupe River Basin of Texas can be approximated by several connected line segments as shown in Figure IV. This approximating function, A(S), for the Cuero I capacity -area relationship is defined by

$$A(S) = \begin{cases} C_1 S & , 0 \le S \le 130 \\ C_2 (S-130) + 130C_1, & 130 \le S, \end{cases}$$

where A (S) is the water surface area (1000 Acres),

S is the reservoir storage volume (1000 Acre-Feet),

$C_1 = .0923$,

$C_2 = .0311$.

By separating the storage variable S into two components, one corresponding to the storage below 130,000 Acre-Feet (S_1) and one to the storage above 130,000 Acre-Feet (S_2), the evaporation loss can be reasonably approximated as a proportion of storage. The approximate evaporation loss E (1000 Acre-Feet/Month), as a function of reservoir storage, is given by

$$E = e\ (C_1\ S_1 + C_2\ S_2),$$

where e is the evaporation rate (feet/month). The percentage losses due to evaporation for each of the storage segments S_1 and S_2 are thus eC_1 and eC_2 respectively.

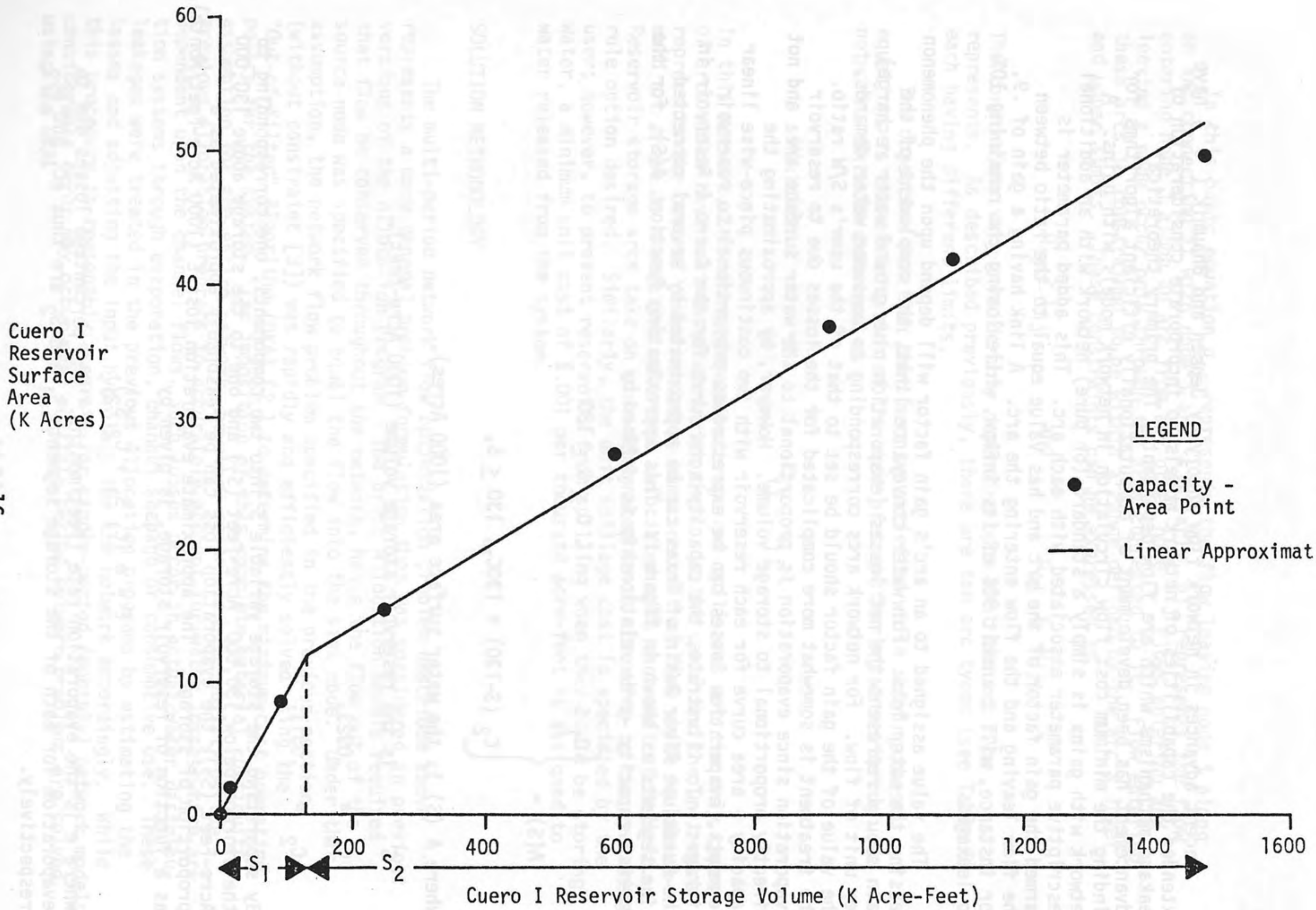

Cuero I Reservoir Surface Area (K Acres)
0
10
20
30
40
50
60
0
200
400
600
800
1000
1200
1400
1600
S1
S2
Cuero I Reservoir Storage Volume (K Acre-Feet)
LEGEND
Capacity - Area Point
Linear Approximat

Figure IV: Piecewise Linear Approximation of a Capacity-Area Curve

The network model in the AL-IV allocation program uses the above equation to account for the effect of evaporation of reservoir storage. The two storage components S_1 and S_2 are represented in the network by two carry-over storage arcs (see Figure II). The gain factors for these arcs are determined by subtracting each of the evaporation loss percentages (eC_1 and eC_2) from unity. These gains may vary by month according to the monthly evaporation rate e. A $1/(1000 Acre-Feet) benefit is assigned in the network for the lower range storages (S_1), while no unit benefit (or cost) is assessed for volume retained in the upper storage range (S_2). This helps insure that the arcs corresponding to the lower storage range are saturated before flow is allowed through the links representing the upper storages. Under some circumstances, however, the algorithm may attempt to put water in the upper range of storage before the lower segment of storage capacity is full. Remedies for this difficulty include setting the minimum storage in the impoundment equal to the storage level where the linear segments join, or specifying operating rules designed to require the lower segment of storage to be full at certain periods of low total storage. These and other corrective procedures will be described in the section dealing with model limitations.

The advantages gained by the enhanced modeling capabilities of the network models with gains are somewhat offset by increased computational requirements. Reported analysis indicated that the GAIN program developed by Jensen and Bhaumik (3) required, when solving large pure network problems, approximately double the execution time of the fastest currently available OKA-type network algorithms.

INPUT REQUIREMENTS

The following physical, economic, and water usage data elements are required for input into the Allocation program.

System Structure Data

(1) The network's dimensions.
(2) A list of each link entering each node.
(3) A list of the nodes at the end of each link (link direction implied).

Junction (node) Data

(1) A literal name for each junction.
(2) The maximum storage capacity at each junction.
(3) The minimum storage allowed at each junction.
(4) The storage versus area relationship for each junction.
(5) The return flow S/W ratio (effluent to influent ratio) and number of the node where the return flow will be discharged.
(6) The simulation year each reservoir is built.

Canal (Link) Data

(1) A literal name for each canal.
(2) The maximum flow capacity for each canal.
(3) The minimum flow allowed for each canal.
(4) The unit cost of pumping in each canal.
(5) The total dynamic pumping head of each canal.
(6) The simulation year each canal is built.
(7) Average annual percentage conveyance loss along each canal.

Cost Data

(1) Interest rates and repayment periods.
(2) Capital and operating cost for each reservoir.
(3) Capital and operating cost for each canal.
(4) The unit cost of deficit at each demand node by season.

Hydraulic and Hydrologic Data

(1) The unregulated inflow to each junction site for each time frame.
(2) The intake demand at each junction for each time frame.
(3) The evaporation rate at each reservoir for each time frame.

System Simulation Data

(1) Delimiters for time interval and total simulation span.
(2) A list of nodes where spills may occur and a unit cost of spills.
(3) Total annual amount, seasonal distribution and unit cost of available import water.
(4) Fraction full for each reservoir at start of simulation.

CAPABILITIES

The AL-IV Program is capable of analyzing the operation of an interconnected system of as many as 30 reservoirs and non-storage junctions and 45 artifical and natural water conveyance channels. Up to twelve periods within a year may be considered in simulating the operation of a surface water resources system. The entire simulation time horizon may be as much as 45 years in length.

The multi-period network model solved by Allocation may contain as many as 1800 arcs and 500 nodes. For a particular problem, the number of arcs and nodes are computed according to the equations on pages 5 and 7. The user should verify before executing the program that the spatial and temporal dimensions of the problem do not generate a network model which exceeds the specified limits.

The Allocation Program is capable of performing a number of analytical functions either singly or in conjunction with other Texas Water Development Board models (eg., DPSIM (5)). Included in the model's capabilities are (1) determining the minimum cost operating policy for a multiple reservoir system over a finite time horizon, (2) sizing reservoir or canal projects and (3) finding monthly reservoir operating rules. Details of how these capabilities are achieved are presented in the following paragraphs.

Minimum Cost Operating Plan

A minimum cost operating plan can be found for any development plan. The link-node configuration and the length of the planning horizon define the network. Reservoir and canal capacities, inputs, and demands describe the major constraints. A set of deterministic inflows and return flows describe the inputs. Projected demands constitute the water requirements. The solution to the network flow problem associated with these conditions is a minimum cost operating plan. It represents the least costly method of transferring water through this network.

The network model solved by the Allocation program has the ability to look ahead in time, up to four years, to operate the system. If the critical drought is no more than four years in duration then the model can adequately convey water within the system to provide the minimum cost operating plan. However should the drought last more than four years then the network solution may not foresee the coming water scarcity and thus may fail to take the appropriate corrective measures to meet future water shortages. This situation can be alleviated by analyzing the system's operation over such long term droughts with the DPSIM program (5). Such an analysis provides annual terminal reservoir storages which can be forced on the system by the Allocation program so that water is adequately moved within the system to insure minimum cost operation over the critical drought.

Canal and Reservoir Sizing

The optimal staging, sizing and sequencing of a number of reservoir and conveyance projects within a large-scale water resources system may be determined by using the Allocation program in conjunction with the DPSIM capacity expansion program. The computational details of this procedure are described in the documentation of the DPSIM program.

If the system under study has only one proposed canal or reservoir project to be added then the Allocation program may be utilized singly to size this facility. The optimal sizing of individual canals or reservoirs can be achieved by utilizing an iterative process requiring manual intervention and engineering judgements to converge on an optimal solution. Results from the GAIN Algorithm include the marginal costs of the canal and reservoir sizes (constraints). The marginal cost of a capacity constraint increases as the constraint decreases because the increment of water associated with the decrease in capacity must find another route to its destination. Since the solution with the higher constraint is the least costly, the lower constraint must increase operating costs. This increase in cost is the marginal cost of the constraint. It represents the savings that would result if the upper bound (maximum) capacity constraints were increased by one unit of flow or, conversely, if the lower bound (minimum) capacity constraints were reduced by one unit of flow.

The manual procedure begins by setting the upper bound constraints at arbitrarily high values that will not be reached. As a result, all marginal costs will be zero. The solution to this problem, which is essentially unconstrained, is characterized by high ratios of maximum to mean arc flows. These ratios are reduced by decreasing the canal and reservoir sizes until their marginal costs become non-zero. These sizes can be further reduced until the increase in their marginal costs becomes equal to the decrease in their marginal capital costs.

Since power costs are incurred in each time interval and capital costs occur only once, the two items are not directly comparable. With a system that experiences increasing demands over time, the power cost for each time period should be discounted and all accumulated before making a comparison with the incremental capital costs.

A recommended procedure is to initially size the canal or reservoir under some assumed design conditions and refine them in the later evaluation process. For example, canal size could be preliminarily found under conditions of average hydrology and maximum demand. Reservoir size, on the other hand, could be estimated for extreme hydrologic and demand conditions. This initial size would be refined by interatively reducing or enlarging the conveyance or storage capacity of the facility according to the variations in operating costs computed by the allocation program.

It should be noted that the iterative process described in the above paragraphs is not as computational efficient as using the DPSIM program if more than a few (two or three) alternative sizes are to be considered for the project.

Reservoir Operating Rules

The generation of seasonal target reservoir storage levels (impoundment operating rules) for each reservoir can be obtained by using the Allocation program iteratively as described in TWDB Report 179 (4). An alternative and possible more flexible approach than the procedure developed in Report 179 would be to generate multiple regression equations similar to those developed by Young (6). These equations could be utilized to predict the required end-of-season storage in key reservoirs as a function of various states or conditions of the system. Parameters such as current reservoir storages, season of the year, water demands, and predicted and antecedent seasonal inflows could be utilized as independent variables in such regression equations. The seasonal storage levels would be computed by the Allocation model and these numbers in turn would be utilized as the dependent variable in the regression analysis.

LIMITATIONS

The Allocation Program is designed to optimize the simulated operation of a general configuration of water storage and conveyance facilities. The characteristics of the Network With Gains Algorithm (3) utilized to solve the associated network model does, however, place some restrictions on the type of configurations that can be analyzed.

Networks Containing Loops

The Network With Gains Algorithm requires that an initial solution be provided which is optimal for a particular level of inflow into the sink node. The Allocation Program specifies an initial solution having all flows set to zero. It is assumed that this set of flows is the minimal cost solution for an inflow into the sink of zero. This assumption will not be valid if the network model contains any closed loops or circuits having arcs with negative unit costs either entering or within the circuit. The term closed circuit is defined here to be any internal set of nodes and linking arcs through which a path may be directed back to any starting node.

The valid use of the Allocation Program thus requires that the network model, corresponding to the spatial representation of physical system, contain no closed circuit along which flow may move. Unfortunately, such circuits do arise in actual water resource systems, usually in the form of pump canals or pipelines. To analyze the operation of water storage and distribution network having such characteristic structures, the network representation of the system must be modified to remove the circuit. This is possible by adding artificial junctions and links to the network to redirect the flows. Figure V indicates how this may be done for a simple circuit between two junction points. Nodes A and B are joined by two pump-conveyance links (P_1 and P_2) moving water in opposite directions. By adding two pseudo-junctions A' and A'' and B' and B'', the circuit between A and B is removed. All inflows into A are now directed into A' and all demands on A are located at node A''.

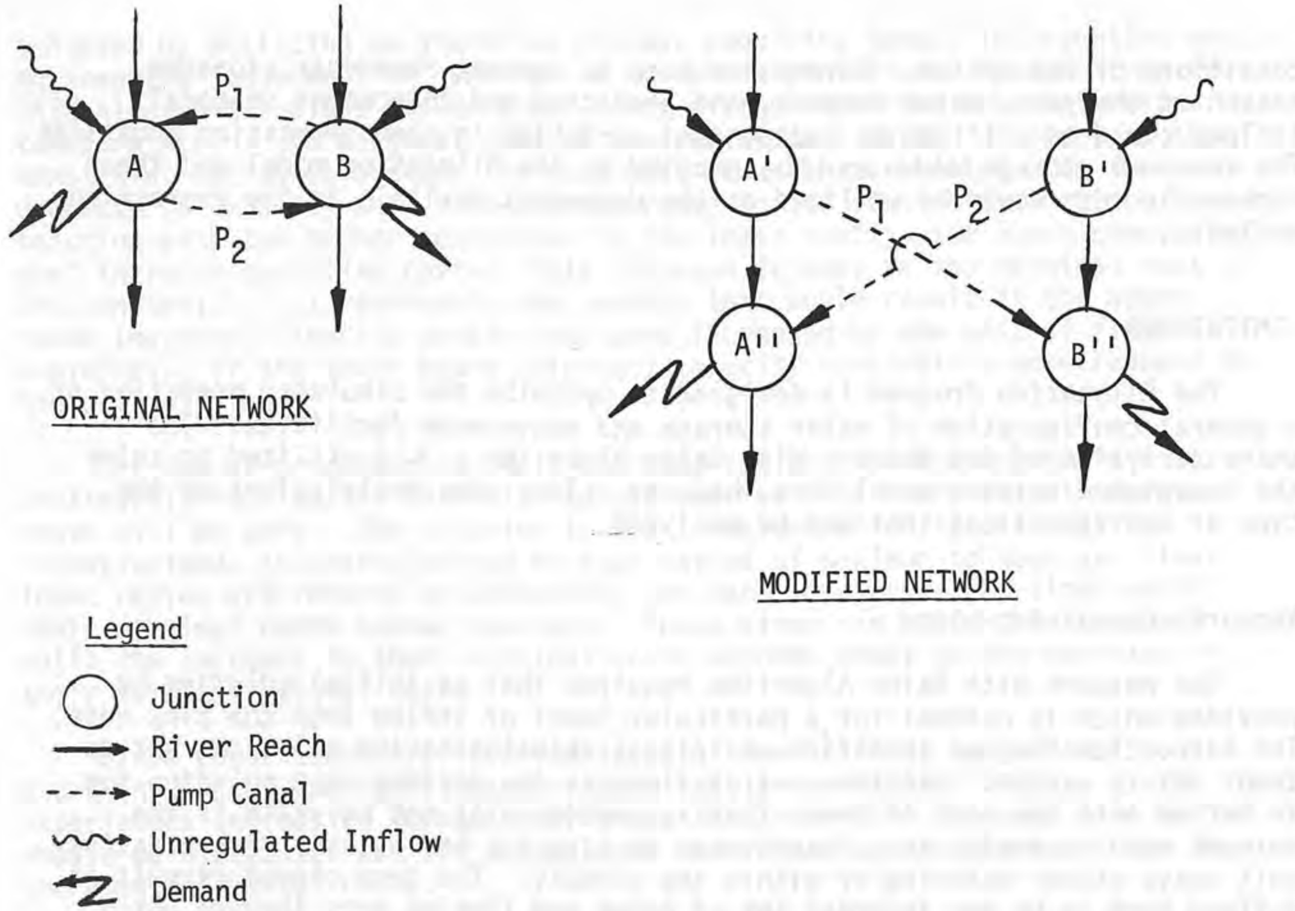


Figure V

Method to Eliminate Loops Between Junctions

A similar procedure may be utilized to remove closed circuits containing reservoirs. Figure VI illustrates how additional nodes and arcs may be included in the network to remove such circuits. This approach is, however, not completely satisfactory since an upstream reservoir will be unable to make downstream releases using water storaged in previous time periods. For example, reservoir A'' in the modified network in Figure VI has no outlet for discharging stored water except to meet local demands, although unregulated inflow (into A') in any given period may be routed downstream or impounded.

To completely overcome this difficulty a slight modification is required in the multiperiod network model as presented in this documentation. Instead of having the storage arcs terminate at the node corresponding to the reservoir, these arcs should terminate at the assoicated artifical node. Thus in Figure VI the storage arcs for reservoir A (node A'' in the modified network) would terminate at node A'. At any given time interval, storage arcs for reservoir A would still originate at node A'' , but by directing these arcs to node A' in the next time period, all water entering node A' could be either restored or released downstream. The required modifications in the computer code would occur in subroutine ARC and would involve simply redefining the terminal nodes for the storage arcs of the reservoirs concerned.

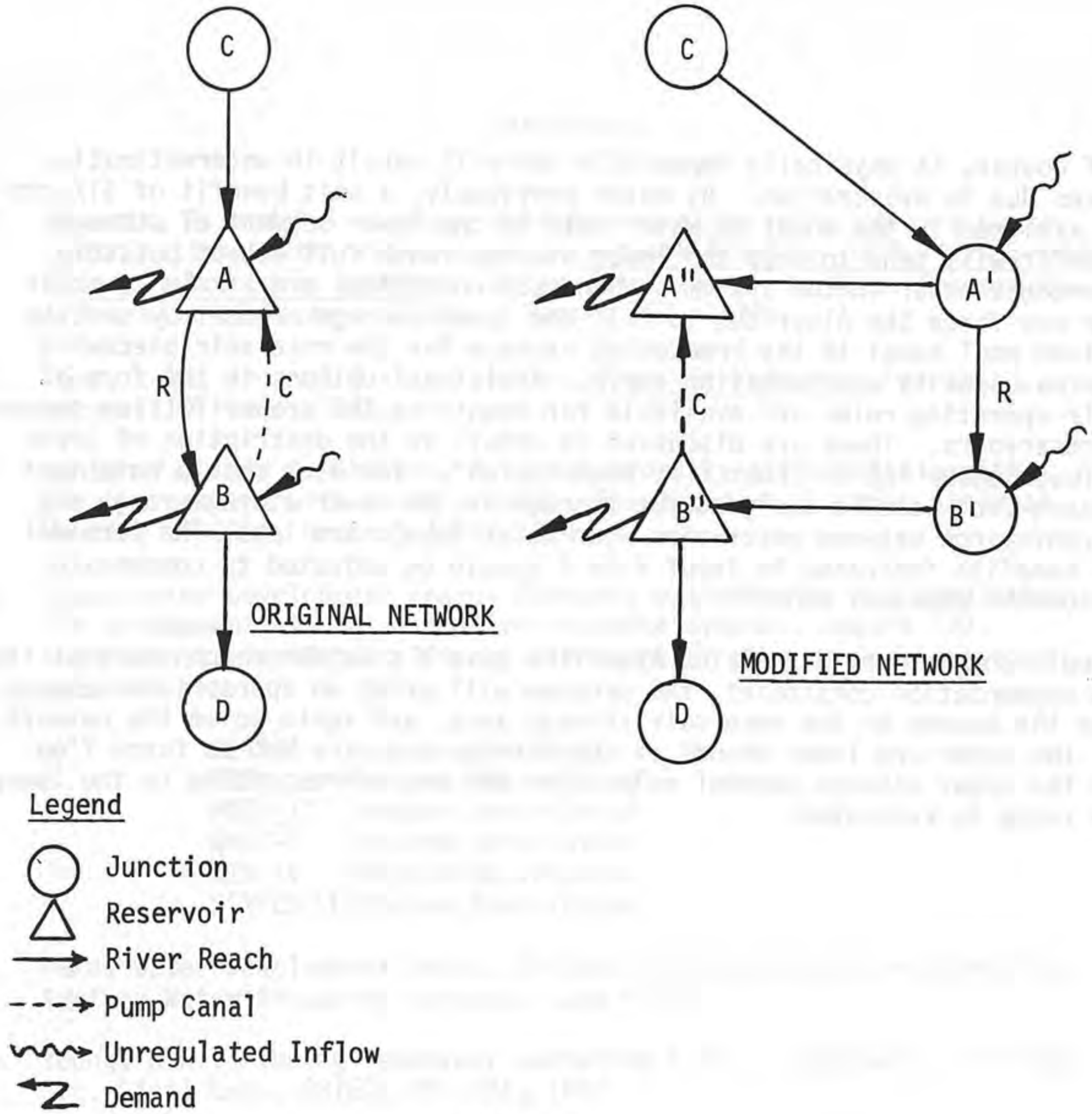


Figure VI

Method to Eliminate Loops Between Reservoirs

Possible Reservoirs Storage Errors

The Allocation Program has an additional limitation which the prospective user should be prepared to deal with. As noted previously, the network model divides the storage in each reservoir into two segments so that evaporation losses may be more accurately computed. The lower segment of storage (represented by S_1 in Figure IV) will invariably be less efficient (in terms for evaporation losses) in storing water than the upper storage segment (S_2 in Figure IV). That is, a unit of volume added to the upper range of storage will add less surface area to the reservoir than would be added if the lower storage segment were augmented by the same volume. Hence, if a shortage occurs in the system, the GAIN Algorithm may attempt to take advantage of this storage efficiency by placing water in the upper segment of storage without first filling the lower segment.

This, of course, is physically impossible and will result in underestimating the losses due to evaporation. As noted previously, a unit benefit of $1/acre-foot is assigned in the model to water held in the lower segment of storage. This benefit will tend to keep the lower storage range full except possibly when shortages occur in the system. When such conditions are likely to occur the user may force the Algorithm to fill the lower storage segment by setting the minimum pool equal to the breakpoint storage for the reservoir piecewise linear area-capacity approximation curve. Additional options in the form of reservoir operating rules are available for requiring the proper filling sequence in the reservoirs. These are discussed in detail in the description of Input File P found under "AL-IV Input File Description". The user should note that the $1/acre-foot benefit assigned for storage in the lower storage range may effect conveyance between reservoirs when water levels are low. The terminal storage benefits indicated in Input File P should be adjusted to compensate this automatic seasonal benefit.

Should the Network With Gains Algorithm give a solution which violates the storage segmentation constraint, the program will print an appropriate message, redefine the bounds on the reservoir storage arcs, and again solve the network model. The upper and lower bounds on the storage arcs are set to force flow through the upper storage segment only after the arc corresponding to the lower storage range is saturated.

REFERENCES

1. Bhaumik, G. and P.A. Jensen, Optimum operating policies of a water distribution system with losses,in Network Modeling of Multireservoir Water Distribution Systems, Center for Research in Water Resources, University of Texas at Austin, Report 107, 1974.

2. Ford, L.R., Jr.,and D.R. Fulkerson, Flows in Networks, Princeton University Press, 1962.

3. Jensen P.A. and G. Bhaumik, A computationally efficient algorithm for the network with gains problem, Presented at the 45th National Meeting of ORSA, Boston, April 1974.

4. Texas Water Development Board, Economic optimization and simulation for management of regional water resource systems, Report 179. Supplementary to Report 179 are six computer program documentation volumes:

   AL-III Program Description
   DES Program Description
   MOSS-III Program Description
   QNET-I Program Description
   SIM-IV Program Description
   SIMYLD-II Program Description

5. Texas Water Development Board, Optimal Capacity Expansion Model for Surface Water Resources Systems, June 1975.

6. Young, G.K., Finding reservoir operating rules, J.Hydraul. Div. Amer. Soc. Civil Eng., 93(6), 297-321, 1967.

ACKNOWLEDGEMENTS

The water supply allocation model AL-IV was adapted from previous versions of the model by Dr. Quentin Martin of the Systems Engineering Division of the Texas Water Development Board. The network with gains algorithm was developed by Dr. Paul Jensen of the University of Texas at Austin and Dr. Gora Bhaumik of the University of California at Fullerton. A source deck of the GAIN Program was graciously provided by Dr. Jensen. Glenn Merschbrock, Nick Carter and Glenda Leftwich of the Systems Engineering Division assisted in assembling the program documentation.

AL-IV PROGRAM DESCRIPTION

The Allocation Model consists of one main program and 21 subroutines, each of which is described in the following sections. A schematic of the interaction between subprograms is given on the following page. The AL-IV Program was coded in Fortran IV for execution on the Univac 1106 Computer. Modification of some constants in Subroutine GAIN may be necessary if the number of significant figures carried by the user's computer is fewer than the nine significant digits in the Univac 1100 Series machines. A further discussion of the possible alterations is provided in the section below describing Subroutine GAIN.

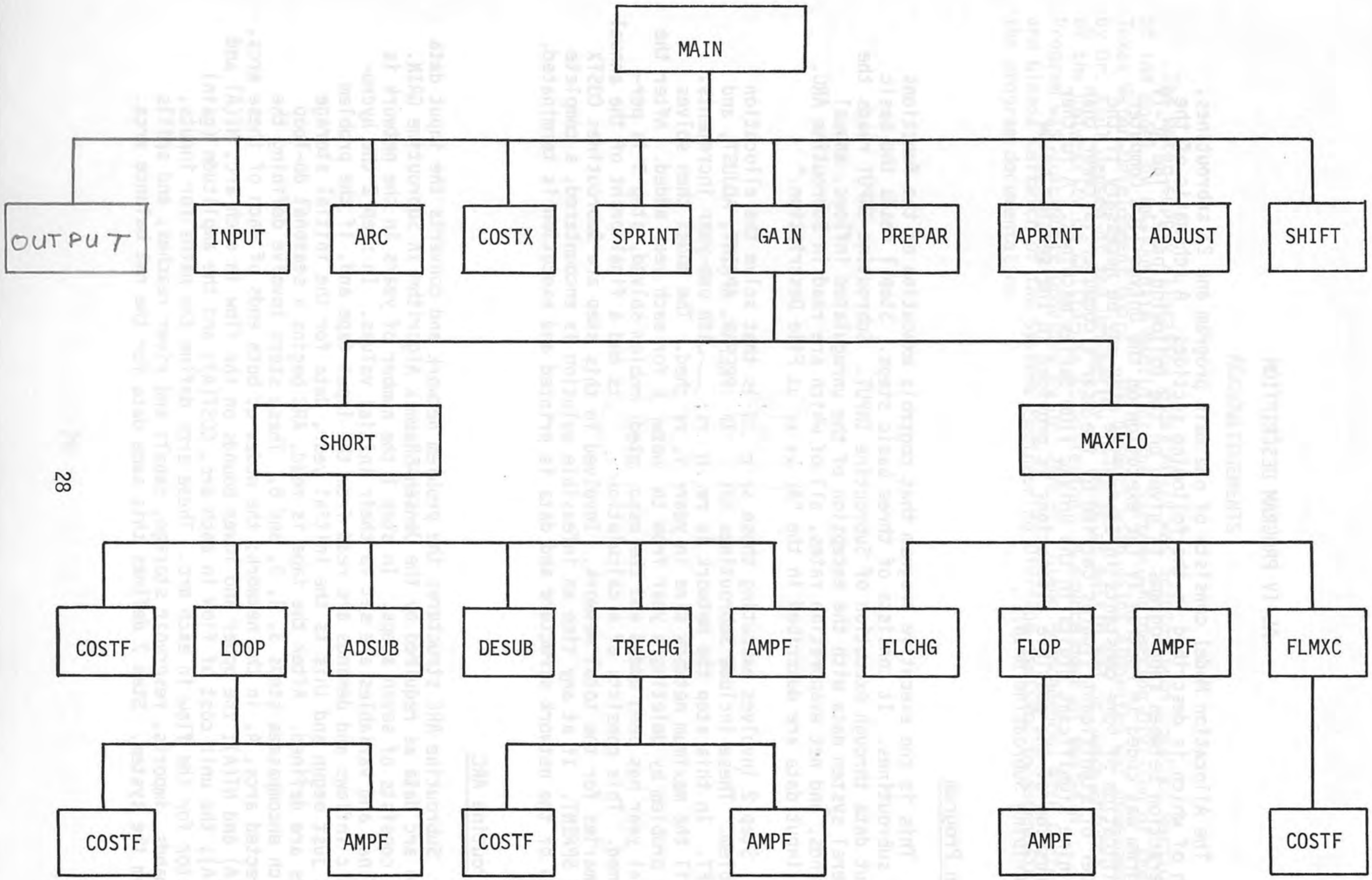
Main Program

This is the executive program that controls execution of the functions and subroutines. It consists of three basic steps. Step 1 reads the basic input data through execution of Subroutine INPUT. Subroutine INPUT reads the general system data with the exception of the unregulated inflows, annual demands, and net evaporation rates, all of which are read in Subroutine ARC. All input data are described in the "AL-IV Input File Description."

Step 2 involves executing those subroutines that solve the allocation problem. These include Subroutines ARC, GAIN, PREPAR, APRINT, ADJUST, and SHIFT. In this step the network is reconstructed with one-year increments, until the maximum network size in years is reached. The model then solves the problem by deleting a year from the network for each year added. After the final year has been added and the assoicated problem solved, step 3 is performed. This consists of a calculation of costs and a final print of the annual summaries for the total network. Involved in this step are Subroutines COSTX and SPRINT. If at any time an infeasible solution is encountered, a complete dump of the network structure and data is printed and execution is terminated.

Subroutine ARC

Subroutine ARC structures the problem network and converts the input data into arc data as required by the Jensen-Bhaumik Algorithm in Subroutine GAIN. ARC consists of seven steps. In step 1 the number of years in the network is defined and variables are set to their initial values. In step 2 the hydrologic inflows and demands are read from the input tape and, if the problem has just begun and this is the initial year, data for the initial storage arcs are defined. After the tape is read, ARC begins a seasonal do-loop which encompasses steps 3, 4, 5, and 6. These steps involve defining the directed arcs, A, in the network; the nodes at both ends of each of these arcs, NF(A) and NT(A); the upper and lower bounds on the flow in each arc, HI(A) and LO(A); the unit cost of flow in each arc, COST(A); and the amplitude (gain) factor for the flow in each arc. These arcs define the paths for inputs, demands, imports, reservoir storage, canals and river reaches, and spills from the system. Step 7 defines this same data for the net balance arcs.


MAIN
OUTPUT
INPUT
ARC
COSTX
SPRINT
GAIN
PREPAR
APRINT
ADJUST
SHIFT
SHORT
MAXFLO
COSTF
LOOP
ADSUB
DESUB
TRECHG
AMPF
FLCHG
FLOP
AMPF
FLMXC
COSTF
AMPF
COSTF
AMPF
AMPF
COSTF

The net balance arcs are the first six arcs in the arc numbering system and they represent the following:

1. Source to sink by-pass arc,
2. Total inputs to the network,
3. Total demands from the network,
4. Total imports to the network,
5. Total spills from the network, and
6. Total final storage in the reservoirs.

The initial storage arcs follow the net balance arcs and are numbered from 7 to 6+NRES (or AMIN). Following these arcs is an arc sequence that is repeated for every season in the network. This are sequence and its relation to the entire arc numbering system is shown in Figure VII on the following page where:

I	=	season number
NRES	=	number of reservoirs
NJ	=	number of link junctions
NL	=	number of links
NS	=	number of spills
NSEAS	=	number of seasons per year
NSOLVE	=	number of years in network
AMIN	=	NRES + 6
AIMIN	=	AMIN + ((I-1)*(2*NJ + 1+2*NRES +NL+NS))
ARMIN	=	AIMIN + (NJ+1)
ALMIN	=	ARMIN + 2(NRES)
ASMIN	=	ALMIN + NL
ASMIN	=	ASMIN + NS

In steps 3, 4, 5, 6, and 7, ARC builds the network one year at a time. After each year has been added, Subroutine GAIN solves the network flow problem and ARC then adds another year to the network. This process is repeated until the maximum network size is reached. At this point, each time a network flow problem is solved, the next year is added to the network and the earliest year deleted.

ARC initializes the flows in all arcs in the network each time a new year is encountered. Flows in all arcs are set to zero.

Subroutine INPUT

Subroutine INPUT reads from cards all the system data except the unregulated inflows, demands and the net seasonal evaporation rates. Reading these data involves ten basic steps. Each step reads one or more records related to the type of data being read. These types of data pertain to (1) system constants, (2) nodes, (3) links, (4) economic parameters, (5) seasonal variations in power costs, import availability and channel losses, (6) annual print control options, (7) spills, (8) reservoir and canal construction times, (9) reservoir operating rules, and (10) shortage costs. These data elements are all described in the "AL-IV Input File Description" section. INPUT also prints, in tabled form, all data read in the subroutine.

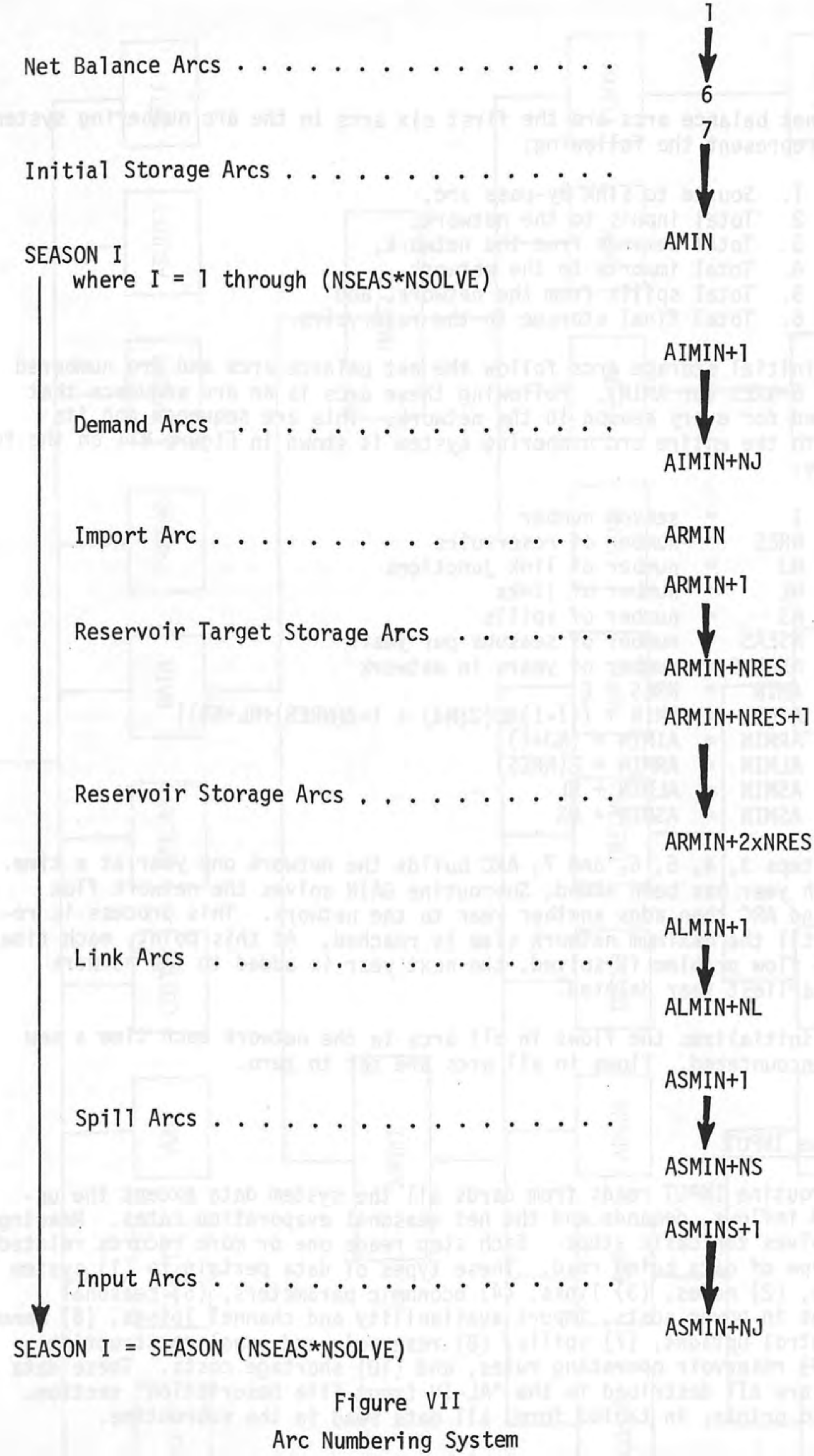


Figure VII

Arc Numbering System

Subroutine GAIN

This subroutine executes the program developed by Dr. Paul Jensen and Dr. Gora Bhaumik (3) for solving the network with gains problem for a given quantity of output flow. All counters and constants for the algorithm are specified in this subprogram.

The network with gains algorithm is executed by iteratively calling the subroutines SHORT and MAXFLO in a two step sequence. In step 1, SHORT determines, on the first iteration, a set of connected arcs providing a minimum cost path for flow from the source to the sink. In each successive iteration, subroutine SHORT deletes a saturated arc from the current flow augmenting path and adds the most promising available arc to create a new path. In step 2, subroutine MAXFLO examines the current minimum cost path and determines the maximum flow that can be directed through that path into the sink. Flows in the appropriate arcs are then increased to the levels needed to increase the flow into the sink the maximum computed amount. If the total flow into the sink is equal to the required inflow then the procedure is terminated; otherwise, the algorithm returns to step 1 and the process is repeated.

Upon completion of steps 1 and 2, GAIN calls subroutine OUTPUT to determine if the network flow solution satisfies the required minimum and maximum link flow and reservoir storage constraints. If the solution is infeasible (ie., violates any constraints) then an appropriate message is printed, the network model is dumped and execution of the program is terminated.

The algorithm for solving the network model requires that an initial solution be provided which is optimal for a given flow into the sink. The set of zero arc flows is used as a starting solution under the assumption that this is the minimal cost set of flows for an input of zero into the sink. This is a valid supposition if the network model has no directed loops or circuits. Modification of the basic network model will general eliminate any directed circuits and allow the operational analysis of the system by the AL-IV Program. A further discussion of this problem may be found under the section entitled "Limitations".

The original version of the GAIN Program was written for the CDC 6600 computer which carries fourteen significant figures. The Allocation Program documented herein was developed on a Univac 1106 which is limited in accuracy to nine significant digits. This difference in machine precision required that the constants EPS and BIG in Subroutine GAIN be redefined from 10^{-6} and 10^{6} to 10^{-3} and 10^{3} respectively. By resetting these small and large tolerance values, the program was allowed to ignore small differences (less than .001) between flow rates. Roundoff errors generated by the machine had been responsible for introducing small differences between supposedly equal values. These discrepencies sometimes resulted in the program trying to remove from the flow augmenting path the identical arc to be added into the path. When such a condition occurred the program began to cycle and would terminate only when the time limit for execution was exceeded.

Subroutine PREPAR

Subroutine PREPAR makes all unit conversions and stores the results from GAIN in matrices for printing in Subroutine APRINT. It performs these operations in five steps for only the first year in each network. Step 1 initiates the values for the variables in the print routine, APRINT, to zero. In step 2 the link flows, their maximum values, and their average annual values are converted to cubic feet per second and stored in the matrix LQ; link power costs and their average annual values are stored in the matrix LP in terms of thousands of dollars; and the marginal link power costs are converted to thousands of dollars per cfs and stored in the matrix LC. Also, annual power costs, APOW, and annual import quantities, AIMP, are determined in step 2. Step 3 places the reservoir storage levels and their annual average in the matrix RS in units of thousands of acre-feet, and the marginal reservoir power costs in the matrix RC in units of dallars per acre-foot. In step 4 the Subroutine computes the marginal costs of the continuity constraints and puts them in the matrix CC in units of dollars per acre-foot, stores the deficit quantities in matrix IDEF and ISHORT in units of thousands of acre-feet, and computes the annual cost of the deficits, ICOST, in dollars per acre-foot. Lastly, step 5 stores the spills in the matrix JQ in units of thousands of acre-feet. All of these matrices are then used in APRINT, the annual print routine.

Subroutine APRINT

Subroutine APRINT prints the annual results of the first year in the network as obtained from PREPAR. APRINT consists of eight steps; each of which prints a specific type of results. The correspondents between steps and results is as follows:

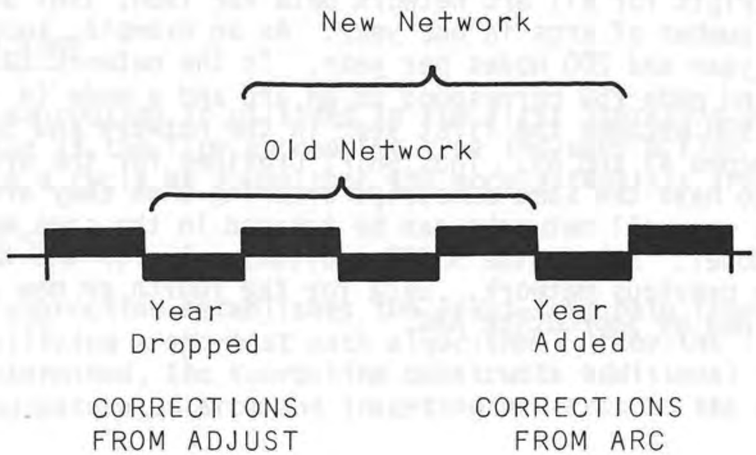
Step 1 - Prints Link Flows
Step 2 - Prints Link Power Costs, Import Costs, Penalty Costs
Step 3 - Prints Marginal Link Costs
Step 4 - Prints Reservoir Storage Levels
Step 4a - Prints Deficits
Step 5 - Prints Marginal Reservoir Costs
Step 6 - Prints Marginal Costs of Continuity Equations
Step 7 - Prints System Spills
Step 8 - Prints Maximum Link Flows

The reader should review "AL-IV Output File Description" for more detailed explanations of the output and examples of the formats used in printing.

Subroutine ADJUST

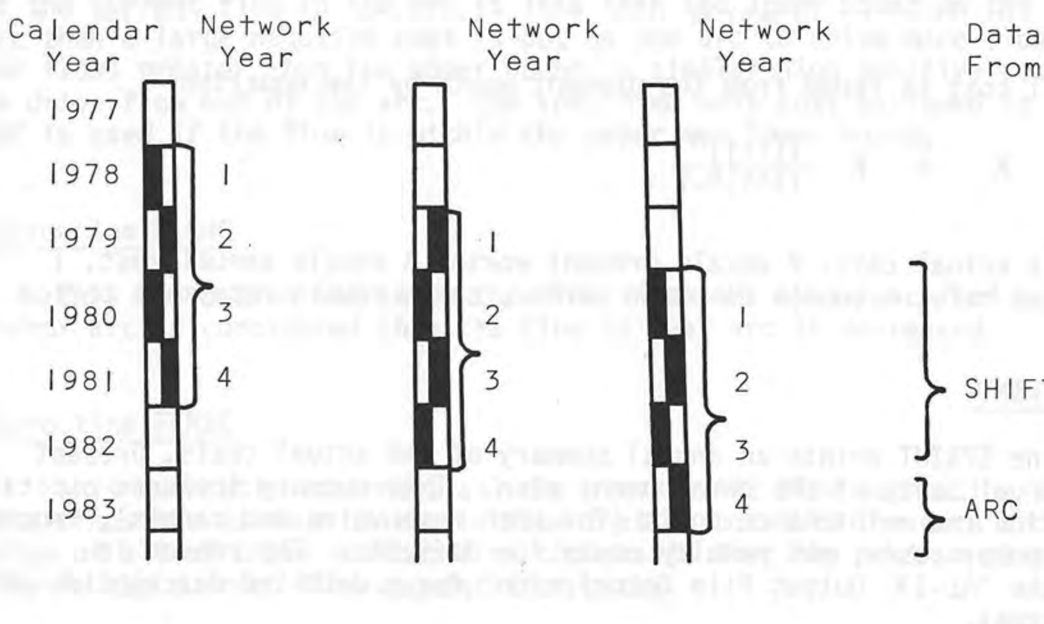
Subroutine ADJUST finds the adjustments that must be made in the lower and upper bounds of, and the flows in, the net balance arcs when a year is dropped from the network. This subroutine is not needed during the initial construction of the reservoir when years are added sequentially to the network

However, once the maximum network size is reached, ADJUST provides the corrections needed when a later year is added to and the earliest year dropped from the network to maintain a constant size. These corrections pertain to the year dropped. The corrections needed for the year added to the network are made in Subroutine ARC. The sketch below illustrates this. Subroutine ARC uses the corrections made in Subroutine ADJUST to update the net balance arcs.



Subroutine SHIFT

Subroutine SHIFT shifts arc and node network data when a year is added to the network as a year is deleted from the network. The following sketch and description describes this data shift:



The foregoing sketch shows the period 1977 through 1983, and three four-year networks within this period. Take, for example, the last two four-year networks. Assume that the problem involving the period 1979 through 1982 has just been solved and the network for the years 1980 through 1983 is the next to be established. When the main program calls Subroutine SHIFT, the results for 1979 have been printed by APRINT and 1979 can be dropped from the network. The first year in the new network is now 1980, 1981 the second, and 1982 the third. Thus, the subscripts for all arc network data for 1980, 1981 and 1982 must be shifted by the number of arcs in one year. As an example, suppose there are 400 arcs per year and 200 nodes per year. In the network 1979 through 1983, arc 441 and node 269 correspond to an arc and a node in 1980. When 1979 is dropped, 1980 becomes the first year in the network and SHIFT cause these arc numbers to become 41 and 69. Thus SHIFT provides for the arc and node data for all networks to have the same subscript ordering when they are under consideration. In this way, all networks can be treated in the same manner in other portions of the Model. Subroutine SHIFT provides only for the three years retained from the previous network. Data for the fourth or new year added to the network is provided by Subroutine ARC.

Subroutine COSTX

Subroutine COSTX computes the actual cost, the present worth, and the average annual cost of the plan being evaluated. Reservoir capital costs are given as input data and their operation and maintenance costs are expressed as a fraction of their capital costs. Canal capital costs are based on maximum canal flow, and operation and maintenance costs are a fraction of this value. Power and imported water costs are determined from the figures developed in PREPAR. Present worth is obtained from the actual cost (sum of the costs of actual capital, operation and maintenance, power, imported water, and deficit) by the equation

$$P = S \frac{1}{(1+i)^m}$$

and the annual cost is found from the present worth by the equation

$$A = P \frac{i(1+i)^n}{(1+i)^n - 1}$$

where S equals actual cost, P equals present worth, A equals annual cost, i equals interest rate, m equals discount period, and n equals repayment period.

Subroutine SPRINT

Subroutine SPRINT prints an annual summary of the actual costs, present worth, and annual costs of the development plan. This summary includes capital costs, operation and maintenance costs (for both reservoirs and canals), importe water costs, power costs, and penalty costs for deficits. The reader is referred to the "AL-IV Output File Description" for a detailed description of the output format.

Subroutine OUTPUT

Subroutine OUTPUT prints all the arc and network data when the program encounters an infeasible problem. A message is printed noting the problem cannot be solved and a dump of the data follows. This includes arc numbers, gain factors, connecting nodes, upper and lower bounds, unit costs, and arc flows when infeasibility was realized.

Subroutine LOOP

This subroutine is utilized in the first iteration of the GAIN subroutine to determine if the flow augmenting tree includes a flow generating cycle. Should such a cycle be found then the node potentials are adjusted.

Subroutine SHORT

This subroutine establishes the least-cost path from the source to the sink by utilizing a shortest path algorithm. After the initial flow augmenting path is determined, the subroutine constructs additional flow augmenting trees by deleting saturated arcs and inserting new arcs to the current tree.

Subroutine FLOP

This subroutine determines the flow amplitude in a flow generating cycle in the current flow augmenting path. The maximum flow change in the cycle is also computed, as is the constraining arc on maximum flow augmentation.

Subroutine COSTF

This subroutine computes the unit cost of flow transmitted along an arc. If the current flow in the arc is less than the lower bound on the flow in that arc then a large negative cost is put on the arc to drive more flow through it. For flows greater than the upper bound, a similar high positive cost is imposed to drive flow out of the arc. The specified unit cost assigned in subroutine ARC is used if the flow is within the upper and lower bounds.

Subroutine FLCHG

This subroutine increases the flow in an arc by a specified amount. If a mirror arc is considered then the flow in that arc is decreased.

Subroutine FLMXC

This subroutine searches all the arcs in the flow augmenting tree and determines the maximum change in flow that is possible for each arc. The minimum value of all of these maximum flow changes is then the maximum change in the flow in the current flow augmentation path.

Subroutine TRECHG

This subroutine updates the flow augmenting tree by deleting a flow saturated arc and inserting an addition arc. In this manner the tree is continua changed to enable more flow transmission between the source and the sink.

Subroutine MAXFLO

This subroutine calculates the maximum increase in flow into the sink alo the arcs in the current flow augmenting tree. The constraining (flow saturate arc in the path is also established. The flow in the augmenting tree is increa by the computed maximum amount.

Subroutine DESUB

This subroutine deletes a constraining arc from the triple label represent tion of the current flow augmenting path.

Subroutine ADSUB

This subroutine adds an arc to the triple label representation of the flow augmenting tree.

Function AMPF

This function computes the flow amplitude of all arcs and their mirror ar

AL-IV INPUT FILE DESCRIPTION

All input data for the Allocation Program is read from cards and is divided into 20 files labeled from A to T. A detailed description of the individual data files and their associated input parameters is provided in Figure VIII. The required formats for the cards in each file are specified on the input coding sheets (Figure IX). Figure X displays the sequence in which the card files must be arranged for input into the computer.

INPUT FILE DESCRIPTION

CARD OR FIELD ID	DESCRIPTION	PROGRAM VARIABLE
	FILE A - SYSTEM PARAMETERS	
PARAMETERS	Listed below are detailed descriptions of the variables defining some of the spatial and temporal dimensions of the physical system and its associated network formulation (one card).	
NJ	Number of nodes (reservoir and non-storage junctions) in the system (maximum of 30). Columns (11-15)	NJ
NRES	Number of reservoirs in the system (maximum of 30). Columns (16-20)	NRES
NJUNC	Number of non-storage junctions in the system (maximum of 30). Columns (21-25)	NJUNC
NL	Number of conveyance links (canal, pipelines, and river channels) in the system (maximum of 45). Columns (26-30)	NL
NC	Number of artifical conveyance links (canals and pipelines) in the system (maximum of 45). Columns (31-35)	NC
NR	Number of river reaches in the system (maximum of 45). Columns (36-40)	NR
NYR	Number of years for which the operation of the system will be simulated (maximum of 45). Columns (41-45).	NYR
NSEAS	Number of equal length subdivisions of a year used in analyzing the operation of the system (maximum of 12). Columns (46-50)	NSEAS
IDATE	Calendar year corresponding to the first year of simulated operation. Also calendar year to which present value costs are referenced. (Columns 51-55)	IDATE
IMP	Number of the junction (node) where import into the system may occur. Leave blank is no import is desired. Columns (55-60)	IMP
	FILE B - STREAM JUNCTION DESCRIPTION	
	Listed below are detailed descriptions of the variables describing the system junctions (storage and non-storage). Consecutive numbering of the junctions (nodes) is required with the reservoirs being numbered first followed by the non-storage junctions. There must be one card for each of the NJ (FILE A) junctions.	
NODE NO.	Numerical designation of the junction. Columns (1-2)	J
IDENT	Alphanumeric descriptor of the junction. Columns (3-10)	RNAME
MAXIMUM STORAGE	Maximum storage capacity of the junction (in 1000 acre-feet). Columns (11-20)	RCAP
MINIMUM STORAGE	Minimum allowed storage capacity of the junction (in 1000 acre-feet). Columns (21-30)	RMIN
RESERVOIR CAPITAL COST	Cost of constructing the storage facility at the junction (in millions of dollars). This field may be left blank if the user does not require an analysis of the capital cost over time for the system. Columns (31-40)	CRES
S/W RATIO FOR DEMAND	Decimal fraction of water removed from the junction (to satisfy the demands at the junction) that is not consumptively used. This fraction must be greater than zero and should be set to 1 if the return flow does not re-enter the system. Columns (41-45)	RETSW
RETURN FLOW NODE	Numerical designation of the junction where the effluent from the demand at this junction is discharged back into the stream system. Leave this field blank if the effluent does not re-enter the system. Columns (51-55)	NLINK
MINIMUM DEMAND FRACTION	Fraction of water demand at the junction which must be met. A decimal value between 0 and 1 is required. The value in this field imposes a constraint on the network model that requires the indicated fraction of the intake demand to be supplied. Should the system be unable to meet any of these minimum demands then the program will stop with an infeasible solution. Columns (51-55)	DLO
BREAKPOINT STORAGE	Reservoir storage (in 1000 acre-feet) were the two linear segments join, in the piecewise linear approximation of surface area-storage capacity curve for this junction. Leave this field blank if the junction is not a reservoir. This storage value may be smaller than the minimum allowed storage but no greater than the maximum capacity of the reservoir. Columns (56-65)	BND
LOWER STORAGE SLOPE	Slope of line approximating the area-capacity curve in the lower range of storage. Columns (71-75)	ACO(1)
UPPER STORAGE SLOPE	Slope of line approximating the area-capacity curve in the upper range of storage. Columns (76-80)	ACO(2)
	FILE C - LINK DESCRIPTION AND SYSTEM CONFIGURATION	
	Listed below are detailed descriptions associated with the various links in the system. One data card is required for each of the NL (FILE A) conveyance links. Links should be numbered consecutively from one, beginning with river reaches followed by pump conveyance links. The user is directed to the prior section dealing with the limitations in the program for a discussion of the considerations required if the network contains circuits or loops.	
LINK NO.	Numerical designation of the conveyance link. Columns (1-2)	L
IDENT	Alphanumeric descriptor of the conveyance link. Columns (3-10)	CNAME
MAXIMUM FLOW CAPACITY	Maximum flow rate for the link (in ft^3/sec). Columns (11-20)	CCAP
MINIMUM FLOW ALLOWED	Minimum flow rate for the link (in ft^3/sec). Columns (21-30)	CMIN
TOTAL DYNAMIC HEAD	Total dynamic head (in feet) which must be over come when pumping water along the link. Leave this data field blank if the link is a natural river channel. Columns (31-40)	CLIFT
LINK CONVEYANCE FACTOR	Fraction of water entering the link which also leaves the link. This value should represent the yearly average fraction of water conveyed through the link, and must have a value greater than zero. Columns (41-50). Set = 1 if and only if link is a pipeline.	EL
ORIGIN NODE	Originating node for the link. Columns (71-75)	LNODE(1)
TERMINAL NODE	Terminal node for the link. Columns (76-80)	LNODE(2)
	FILE D - CONVEYANCE LINK CONSTRUCTION COST COEFFICIENTS	
	Listed below are detailed descriptions of the data associated with the construction cost versus flow capacity of the conveyance links in the system. Each link is assumed to have two associated second-order polynominal capital cost equations: 1) canal ditch cost (in dollars) vs. flow capacity (in acre-feet/year) and 2) canal pump station cost (in dollars) vs. flow capacity (in acre-feet/year). The first equation could be utilized to represent the cost of installed pipe if the link is a closed conduit. There must be one card for each of the NL (FILE A) links, however those cards corresponding to natural river channels should have blanks in the coefficient data fields. Should the user not require any output concerning the time history of capital costs over the simulated operating horizon then all coefficient data fields should be left blank.	
LINK NO.	Numerical designation for the link. Columns (1-2)	L
IDENT	Alphanumeric descriptor for the link. Columns (3-10)	-
CANAL DITCH COST VS. CAPACITY	Construction cost equation of the form $\text{Ditch Cost (\$)} = A + B*Q + C*Q^2$ where A, B, and C are constants and Q is the conveyance capacity of the link in acre-feet/year.	
INTERCEPT COEFF.	Constant coefficient A in the above equation. Columns (11-20)	CLINK(1,L,1)
1st ORDER COEFF.	Constant coefficient B in the above equation. Columns (21-30)	CLINK(1,L,2)
2nd ORDER COEFF.	Constant coefficient C in the above equation. Columns (31-40)	CLINK(1,L,3)
CANAL PUMP STATION COST VS. CAPACITY	Construction cost equation of the form $\text{Pump Station Cost (\$)} = A + B*Q + C*Q^2$ where A,B, and C are constants and Q is flow in acre-feet/year.	
INTERCEPT COEFF.	Constant coefficient A in the above equation. Columns (41-50)	CLINK(2,L,1)
1st ORDER COEFF.	Constant coefficient B in the above equation. Columns (51-60)	CLINK(2,L,2)
2nd ORDER COEFF.	Constant coefficient C in the above equation. Columns (61-70)	CLINK(2,L,3)

Figure VIII-A

AL-IV Input File Description

INPUT FILE DESCRIPTION

CARD OR FIELD ID	DESCRIPTION	PROGRAM VARIABLE
	FILE E - ECONOMIC PARAMETERS	
ECON FCTOR	Listed below are detailed descriptions of the data required to specify the economic parameters for the system. There must be one card in this file.	
ANNUAL DISCOUNT RATE	Annual discount rate used in computing present value. Columns (11-20)	PERINT
DEPT REPAYMENT PERIOD	Period (in years) over which debts, incurred for constructing reservoirs and conveyance links, must be repayed. Columns (21-30)	NPAY
ELECTRICAL POWER COST	Average annual unit cost of electricity in $/KW-HR. Columns (31-40)	POWR
ANNUAL OM&R COST FOR RES.	Annual fixed operating, maintenance, and replacement cost for reservoir facilities as a decimal fraction of the construction cost of the reservoir. Columns (41-50)	RESONM
ANNUAL OM&R COST FOR CANALS	Annual fixed operating, maintenance, and replacement cost for conveyance facilities as a decimal fraction of the construction cost of the canal or pipeline. Columns (51-60)	CONONM
UNIT COST OF IMPORTED WATER	Unit cost of water imported into the system in $/acre-foot. Columns (61-70)	CIMP
CONSTRUCTION PERIOD FOR RES.	Time period (in years) required between initiation of construction and final completion of reservoir projects. Columns (71-75)	LAGR
CONSTRUCTION PERIOD FOR CANALS	Time period (in years) required between initiation of construction and final completion of canal and pipeline projects. Columns (76-80)	LAGC
	FILE F - SEASONAL VARIATIONS IN POWER COSTS	
POWER DIST	Listed below are detailed descriptions of the data specifying the variations in unit electrical power costs within a year. There must be one card in this file.	
SEASONAL ELECTRICAL POWER COST FACTORS	List of decimal fractions corresponding to the seasonal variations in the average annual power cost specified in FILE E. Columns (11-15; 16-20; 21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70).	POWFAC
	FILE G - SEASONAL AVAILABILITY OF IMPORT WATER	
IMPORT	Listed below are detailed descriptions of the data specifying the seasonal availability of import water into the system. There must be one card in this file.	
MAXIMUM IMPORT	Maximum volume (in 1,000 acre-feet) of water available for import into the system in each year. Columns (11-15)	XIMP
SEASONAL DISTRIBUTION OF AVAILABLE WATER IMPORT	List of decimal fractions specifying the portion of the total annual import water that is available in each season of the year (maximum of 12 values). Columns (16-20; 21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70; 71-75).	XX
	FILE H - SEASONAL VARIATION OF LINK CONVEYANCE FACTORS	
TRAN FACTRS	Listed below are detailed descriptions of the data describing the variations within the year of the conveyance ratios for the links. There must be one card in this file. Pipelines are not effected by these factors.	
SEASONAL CORRECTION FACTORS FOR CONVEYANCE COEFFICIENTS	List of decimal fractions specifying the seasonal transmission factors (ratios of flows leaving link to flows entering link) as fractions of the average annual link conveyance factor (FILE C) (maximum of 12 nonzero values). Columns (11-15; 16-20; 21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70)	ELF
	FILE I - PROGRAM OUTPUT CONTROL	
CONTROL	Listed below are detail descriptions of the data designating the type of output information to be provided for each of the years in the operating horizon. There must be one card in this file.	
ANNUAL OUTPUT CONTROL	List of integer values defining the output mode for each year of the simulation. The print options are as indicated below: 0 - Print all output from the solution of the network model but not the network model itself; 1 - Print all output from the solution of the network model and the model formulation, 2 - Print all output specified under option "0" except marginal costs, 3 - Suppress all output for this year A maximum of 45 values may be listed with one value per column. It is assumed that the yearly print options are read sequentially beginning with the first year of simulation in column 15. Columns (15-59)	IPRNT
	FILE J - TIME DIMENSIONS OF ALLOCATION PROBLEM	
SOLVE YRS	Listed below are detailed descriptions of the data elements describing the time dimensions of the operational simulation by the Allocation Program. There must be one card in this file.	
NUMBER OF YEARS TO BE SOLVED	Number of years to be simulated by the Allocation Program (maximum of 45 years). Columns (21-30)	NUMYRS
NUMBER OF YEARS IN EACH SOLUTION NETWORK	Number of yearly networks that are to be solved in each multiyear network model as program moves over the simulation period. The procedure used by the program for adding and deleting yearly networks, from the multiyear network, is described in the discussion of Subroutine SHIFT. The number of years in the multiyear network must be at least one and can be as many as desired so long as the total number of nodes and arcs do not exceed 500 and 1,800 respectively. This point is discussed further in the earlier section discussing the capabilities of the program.	NSOLVE
	FILE K - SYSTEM SPILL DESCRIPTION	
SPILLAGE	Listed below are detailed descriptions of the data associated with specifying the unit cost and maximum magnitude of spills from the system. There must be one card in this file.	
MAXIMUM NODE SPILLAGE	Maximum rate of spillage (in 1,000 acre-feet per season) from each spill nodes. A large value (e.g., 5,000) should be used to insure that water can leave the system. If the value is too small then an infeasible solution may result. Columns (11-20)	SCAP
UNIT SPILL COST	Cost of each unit of water spilled from the system (in $1/acre-foot). The program specifies a minimum unit cost of $.001. Columns (21-30)	KSPILL
NUMBER OF SPILL NODES	Total number of nodes in the system which are allowed to spill water from the system (maximum of ten (10) nodes). Columns (31-40)	NSPNDS
	FILE L - SYSTEM SPILL NODES	
SPILL NDS	Listed below are detailed descriptions of the data defining the locations within the system where spills may occur. Care should be taken in specifying the spill nodes so that excess water will be able to exit the system. Infeasible solutions may occur if the spill nodes are improperly located. The spill nodes are read five nodes per card with a maximum of two cards in this file. The number of spill nodes must correspond to the number indicated in FILE K.	
NODE NUMBER OF THE DESIGNATED SPILL NODES	Numerical designation of the nodes which are specified as system spill nodes. These nodes may correspond to either reservoir or non-storage junctions. Columns (19-20; 29-30; 39-40; 49-50; 59-60)	JS
	FILE M - RESERVOIR CONSTRUCTION TIMES	
RES START	Listed below are detailed descriptions of the data defining the installation time of reservoirs within the Simulation period. Starting times for up to four reservoirs are read per card. There must be enough cards in this file to read data for all NR (FILE A) reservoir facilities.	
NUMBER	Numerical designation of the reservoir. Columns (19-20; 34-35; 49-50; 64-65)	I
YEAR	Simulation year for completion of the reservoir. Columns (21-25; 36-40; 51-55; 66-70)	JBLT
	FILE N - LINK CONSTRUCTION TIMES	
LINK START	Listed below are detailed descriptions of the data elements specifying the installation times of all conveyance links in the system. Starting times for as many as 4 links are read per card. There must be sufficient cards in this file to read data for all NL (FILE A) links.	
NUMBER	Numerical designations for link. Columns (19-20; 34-35; 49-50; 64-65)	I
YEAR	Simulation year for completion of the link. Columns (21-25; 36-40; 51-55; 66-70)	LBLT

FIGURE VIII-B

AL-IV INPUT FILE DESCRIPTION

INPUT FILE DESCRIPTION

FILE O - INITIAL RESERVOIR STORAGE

CARD OR FIELD ID	DESCRIPTION	PROGRAM VARIABLE
PCT FULL	Listed below are detailed descriptions of the data elements specifying the storage conditions to be assigned to the reservoirs upon their installation in the system. There must be a sufficient number of cards in this file to provide starting storages for all NRES (FILE A) reservoirs, with a maximum of four (4) storage values per card.	
NUMBER	Numerical designation for the reservoir. Columns (19-20; 34-35; 49-50; 64-65)	I
PERCENT	Percentage of maximum storage capacity corresponding to the initial storage level desired in the reservoir. Columns (21-25; 36-40; 51-55; 66-70)	FSTART

FILE P - RESERVOIR OPERATING RULES

RES OPER

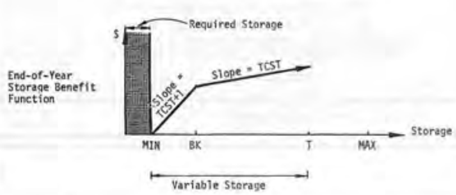
Listed below are detailed descriptions of the date elements required to specify the manner in which the reservoirs in the system are to be operated. The Allocation Program allows the user to specify one of two general operating policies for each of the reservoirs in the system. These rules control the storage level in the reservoirs by specifying the end-of-year storage conditions. The two reservoir policies will be designated as the "upper bound" and "lower bound" terminal storage rules. These rules are predicated on dividing the available storage space in each impoundment into several storage segments. Complete description of these rules are provided below. There must be NRES (FILE A) cards in this file.

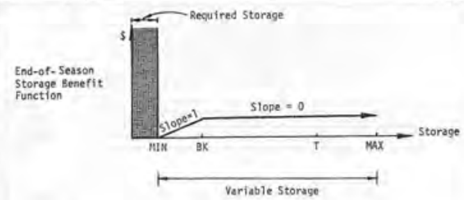
"Upper Bound" Storage Rule (Option 1) -

This operating policy option requires that the user specify for each year of simulated operation a maximum level of storage (T) that is allowed to be carried over to the beginning of the next year. These storage volumes must be no greater than the maximum water conservation capacity of the impoundment. In addition, a unit benefit ($/acre-foot) must be assigned by the user to the annual terminal storages in the reservoir. This number should reflect the relative desirability of storing water in this reservoir as opposed to storing the water in another impoundment or releasing it to meet demands.

In the network model, the reservoir storages at the the end of each season and year are assigned various benefit values. The associated benefits for this operating rule option are indicated in the figures below. The BK storage level in these figures corresponds to the breakpoint storage in the piecewise linear approximation to the area-capacity curve (FILE B).

"UPPER - BOUND" OPERATING RULE





"Lower Bound Storage Rule (Option 2) -

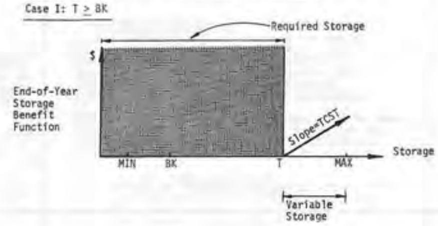
In contrast with the previous operating option that varies the maximum amount of carry-over storage, this option requires that the user specify target terminal storages that are required to be met or exceeded. As with Option 1, a unit storage benefit is assigned for annual terminal storages between the maximum storage capacity and the terminal target.

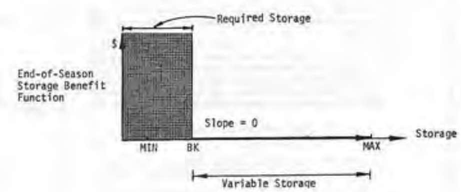
This reservoir operating option can be useful in cases where the network model tries to fill the upper segment of storage in a reservoir before the lower segment is filled. Such situations usually arise during periods when reservoir storages are quite low and shortages are occurring at the demand centers.

The manner in which this operating option functions is illustrated in the figure below. The variable range of storage is controlled by the minimum target storage (T) designated. By specifying any value of T between the minimum and maximum capacities of the reservoir, the network model is forced to fill the storage in the impoundment in the proper sequence. If the required target level is set too high, however, it is possible that either an infeasible solution may result or a shortage may occur in the demands supplied from the reservoir.

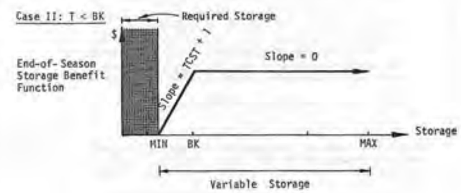
"LOWER BOUND" OPERATING RULE

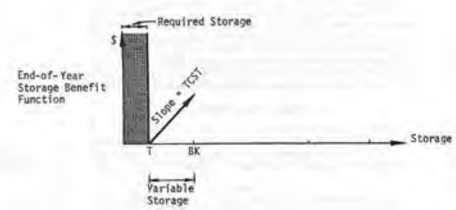
Case I: T ≥ BK





Case II: T < BK





CARD OR FIELD ID	DESCRIPTION	PROGRAM VARIABLE
NUMBER	Numerical designation of the reservoir. Columns (14-15)	J
OPER OPT	Operating Option for the reservoir. A value of either 1 (for "Upper Bound" Rule) or 2 (for "Lower Bound" Rule) must be used. Columns (19-20)	IOPT
STORAGE BENEFIT	Unit storage benefit (in $/acre-foot) for water stored at the end of each year in the reservoir. Columns (21-25)	TCST

FIGURE VIII-C

AL-IV INPUT FILE DESCRIPTION

INPUT FILE DESCRIPTION

CARD OR FIELD ID	DESCRIPTION	PROGRAM VARIABLE
END-OF-YEAR TARGET RESERVOIR STORAGE LEVELS	Set of as many as 13 yearly target terminal storage levels for the reservoir as decimal fractions of the maximum reservoir storage (FILE B). Each value must be no greater than 1 and no less than the fraction associated with the minimum storage pool (FILE B). If the number of years NYR (FILE A) in the simulation period exceeds 13 then the target annual terminal storage levels for years 14 and beyond are all set equal by the program to the target given for year 13. Columns (26-29; 30-33; 34-37; 38-41; 42-45; 46-49; 50-53; 54-57; 58-61; 62-65; 66-69; 70-73; 74-77)	TFAC
α	A decimal fraction specifying the probable underestimation in each required yearly target terminal storage level given for the reservoir. This variable is applicable for use with operating Option 2 when the storage target may slightly underestimate the required carry-over storage needed in future years. The value for α may vary with the accuracy of the targets; however, experiences with storage targets generated by the DPSIM program indicated that α=.1 was satisfactory. A nonzero value for α should be used only only under Option 2 and then only at key supply reservoirs in the system. The specified value of α supplied by the user is utilized in the program to increase the targets from later to earlier years in the simulation period. These increases are cumulative since they take into account intervening losses. (Columns (78-80)	ALP
	FILE Q - DEMAND SHORTAGE COSTS	
SHORT COST	Listed below are detailed descriptions of the data specifying the unit cost incurred at each junction for failing to supply the maximum intake demand. The unit shortage costs must be provided for each season of the year, with a maximum of four seasons on each card.	
NUMBER	Numerical designation for the junction. Columns (19-20)	J
SEASON	Numerical designation of the season within the year. This must be a positive integer value no greater than 12. Columns (21-25; 36-40; 51-55; 66-70)	I
UNIT COST	Unit Cost of shortage in the demand (in $/acre-foot) at the junction in the indicated season. Columns (26-35; 41-50; 56-65; 71-78)	CST (I, J)
	FILE R - JUNCTION UNREGULATED INFLOWS	
UNREG INF	Listed below are detailed descriptions of the data specifying the seasonal unregulated (natural) inflow into all junctions in one simulation year. Each card contains up to twelve seasonal inflow values for a single junction. There must be NJ (FILE A) cards in this data file, and junction inflows must be read in sequential order beginning with junction number 1.	
YEAR NO.	Simulation year in which the seasonal inflows on the card occur. Columns (13-15)	
NODE NO.	Numerical designation for the junction. Columns (19-20)	
SEASONAL INFLOW VALUES	List of unregulated seasonal inflows into the junction (in 1,000 acre-feet). Columns (21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70; 71-75; 76-80). Non-Negative flows required.	U
	FILE S - JUNCTION WATER DEMANDS	
DEMANDS	Listed below are detailed descriptions of the data specifying intake demands (not the consumptive use) of the reservoir and non-storage junctions in the system for a single year. The demands at any junction may vary by season by year over the simulation period. There must be NJ (FILE A) cards in this file, with junctions read in sequential order.	
YEAR NO.	Year in the simulation in which the seasonal demands on the card occur. Columns (13-15)	
NODE NO.	Numerical designation for the junction. Columns (19-20)	
SEASONAL INFLOR VALUES	List of seasonal intake demands at the junction (in 1,000 acre-feet). Columns (21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70; 71-75; 76-80)	D
	FILE T - RESERVOIR NET EVAPORATION RATES	
EVAP RATE	Listed below are detailed descriptions of the data specifying the seasonal net evaporation rates in a single year at the reservoirs in the system. These must be NRES (FILE A) cards in this file, with the reservoirs read in sequential order.	
YEAR NO.	Year in the simulation in which the seasonal evaporation rates on the card occur. Columns (13-15)	
NODE NO.	Numerical designation for the reservoir. Columns (19-20)	
SEASONAL EVAPORATION RATES	List of seasonal net evaporation rates at the reservoir (in feet/season). Columns (21-25; 26-30; 31-35; 36-40; 41-45; 46-50; 51-55; 56-60; 61-65; 66-70; 71-75; 76-80)	E

Figure VIII-D

AL-IV Input File Description

TEXAS WATER DEVELOPMENT BOARD
ALLOCATION MODEL
AL-IV

SYSTEM PARAMETERS

FILE A

	NJ	NRES	NJUNC	NL	NC	NR	NYR	NSEAS	IDATE	IMP
PARAMETERS										

STREAM JUNCTION DESCRIPTION

FILE B

NODE NO.	IDENT	MAXIMUM STORAGE (1000 ACRE-FEET)	MINIMUM STORAGE (1000 ACRE-FEET)	RESERVOIR CAPITAL COST ($10^6)	S/W RATIO FOR DEMAND	RETURN FLOW NODE	MINIMUM DEMAND FRACTION	PIECEWISE LINEAR APPROX. OF RESERVOIR AREA-CA... BREAKPOINT STORAGE (1000 AC-FT)	LOWER SLOPE	UP... SL...

LINK DESCRIPTION AND SYSTEM CONFIGURATION

FILE C

LINK NO.	IDENT	MAXIMUM FLOW CAPACITY (CFS)	MINIMUM FLOW ALLOWED (CFS)	TOTAL DYNAMIC HEAD (FEET)	LINK CONVEYANCE FACTOR	ORIGIN NODE	TERM. NO...

CONVEYANCE LINK CONSTRUCTION COST COEFFICIENTS

FILE D

LINK NO.	IDENT	CANAL DITCH COST VS. CAPACITY			CANAL PUMP STATION COST VS. CAPACITY		
		INTERCEPT COEFF.	1st ORDER COEFF.	2nd ORDER COEFF.	INTERCEPT COEFF.	1st ORDER COEFF.	2nd ORDER COEFF.
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E
		E	E	E	E	E	E

ECONOMIC PARAMETERS

FILE E

	ANNUAL DISCOUNT RATE	DEBT REPAYMENT PERIOD	ELECTRICAL POWER COST ($/KW-HR)	ANNUAL OM&R COST FOR RESERVOIRS	ANNUAL OM&R COST FOR CANALS	UNIT COST OF IMPORTED WATER ($/ACRE-FOOT)	RES. CONSTR. PER. (YEAR)	CAN. CON. PER.
ECON FCTOR								

SEASONAL VARIATION IN POWER COSTS

FILE F

	SEASONAL ELECTRICAL POWER COST FACTORS											
	SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SEASON 12
POWER DIST												

SEASONAL AVAILABILITY OF IMPORT WATER

FILE G

	MAXIMUM IMPORT (1000 AC-FT)	SEASONAL DISTRIBUTION OF AVAILABLE WATER IMPORT											
		SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SEASON 12
IMPORT													

SEASONAL VARIATION OF LINK CONVEYANCE FACTORS

FILE H

	SEASONAL CORRECTION FACTORS FOR CONVEYANCE COEFFICIENTS											
	SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SEASON 12
TRAN FCTRS												

Note: All numeric fields without decimals are integer

Figure IX-A
AL-IV Input Sheet

TEXAS WATER DEVELOPMENT BOARD
ALLOCATION MODEL
AL-IV

PROGRAM OUTPUT CONTROL

FILE I

		ANNUAL OUTPUT CONTROL (1COL/YR) "1"-print all output plus network model; "0"-print all output; "2"-suppress marginal costs; "3"-suppress all output
CONTROL =		

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

TIME DIMENSIONS OF ALLOCATION PROBLEM

FILE J

		NUMBER OF YEARS TO BE SOLVED		NUMBER OF YEARS IN EACH SOLUTION NETWORK	
SOLVE YRS=					

SYSTEM SPILL DESCRIPTION

FILE K

	MAXIMUM NODE SPILLAGE (1000 AC-FT/SEASON)	UNIT SPILL COST ($/AC-FT)	NUMBER OF SPILL NODES	
SPILLAGE	.			

SYSTEM SPILL NODES

FILE L

NODE NUMBERS OF THE DESIGNATED SPILL NODES

		NUMBER		NUMBER		NUMBER		NUMBER		NUMBER	
SPILL NDS		RES		RES		RES		RES		RES	
SPILL NDS		RES		RES		RES		RES		RES	

RESERVOIR CONSTRUCTION TIMES

FILE M

RESERVOIR START TIMES

		NUMBER	YEAR		NUMBER	YEAR		NUMBER	YEAR		NUMBER	YEAR	
RES START		RES			RES			RES			RES		
RES START		RES			RES			RES			RES		
RES START		RES			RES			RES			RES		
RES START		RES			RES			RES			RES		
RES START		RES			RES			RES			RES		
RES START		RES			RES			RES			RES		

LINK CONSTRUCTION TIMES

FILE N

RIVER REACH AND PUMP CANAL STARTING TIMES (RIVER REACH #'S MUST BE SET TO 01)

		NUMBER	YEAR		NUMBER	YEAR		NUMBER	YEAR		NUMBER	YEAR	
LINK START		LNK			LNK			LNK			LNK		
LINK START		LNK			LNK			LNK			LNK		
LINK START		LNK			LNK			LNK			LNK		
LINK START		LNK			LNK			LNK			LNK		
LINK START		LNK			LNK			LNK			LNK		
LINK START		LNK			LNK			LNK			LNK		

INITIAL RESERVOIR STORAGE

FILE O

INITIAL RESERVOIR STORAGE CONDITIONS (PERCENTAGE OF MAXIMUM STORAGE)

		NUMBER	PERCENT		NUMBER	PERCENT		NUMBER	PERCENT		NUMBER	PERCENT	
PCT FULL		RES	.		RES	.		RES	.		RES	.	
PCT FULL		RES	.		RES	.		RES	.		RES	.	
PCT FULL		RES	.		RES	.		RES	.		RES	.	
PCT FULL		RES	.		RES	.		RES	.		RES	.	
PCT FULL		RES	.		RES	.		RES	.		RES	.	
PCT FULL		RES	.		RES	.		RES	.		RES	.	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

RESERVOIR OPERATING RULES

FILE P

END-OF-YEAR TARGET RESERVOIR STORAGE LEVELS (DECIMAL FRACTION OF MAXIMUM STORAGE)

	NUMBER	OPER. OPTION (1 or 2)	STORAGE BENEFIT ($/AC-FT)	YEAR 1	YEAR 2	YEAR 3	YEAR 4	YEAR 5	YEAR 6	YEAR 7	YEAR 8	YEAR 9	YEAR 10	YEAR 11	YEAR 12	YEAR 13	α
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
RES OPER	RES	OPT	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

DEMAND SHORTAGE COSTS

FILE Q

SEASONAL SHORTAGE COSTS FOR NODE DEMANDS

		NUMBER	SEASON	UNIT COST ($/AC-FT)		SEASON	UNIT COST ($/AC-FT)		SEASON	UNIT COST ($/AC-FT)		SEASON	UNIT COST ($/AC-FT)	
SHORT COST		JCT		.			.			.			.	
SHORT COST		JCT		.			.			.			.	
SHORT COST		JCT		.			.			.			.	
SHORT COST		JCT		.			.			.			.	
SHORT COST		JCT		.			.			.			.	
SHORT COST		JCT		.			.			.			.	

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

Note: All numeric fields without decimals are integer

FIGURE IX-B
AL-IV INPUT SHEET

TEXAS WATER DEVELOPMENT BOARD
ALLOCATION MODEL
AL-IV

JUNCTION UNREGULATED INFLOWS

FILE R

	YEAR NO.	NODE NO.	SEASONAL INFLOW VALUES (1000 ACRE-FEET) *											
			SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SE
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
UNREG INF	YR	JCT												
1–10	11–15	16–20	21–25	26–30	31–35	36–40	41–45	46–50	51–55	56–60	61–65	66–70	71–76	

JUNCTION WATER DEMANDS

FILE S

	YEAR NO.	NODE NO.	SEASONAL DEMAND VALUES (1000 ACRE-FEET) *											
			SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SE
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												
DEMANDS	YR	JCT												

RESERVOIR NET EVAPORATION RATES

FILE T

	YEAR NO.	RES. NO.	SEASONAL EVAPORATION RATES (FEET/SEASON) *											
			SEASON 1	SEASON 2	SEASON 3	SEASON 4	SEASON 5	SEASON 6	SEASON 7	SEASON 8	SEASON 9	SEASON 10	SEASON 11	SE
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
EVAP RATE	YR	RES												
1–10	11–15	16–20	21–25	26–30	31–35	36–40	41–45	46–50	51–55	56–60	61–65	66–70	71–76	

* *Left justify and insert decimal points; all other numeric fields integer*

FIGURE IX-C
AL-IV INPUT SHEET

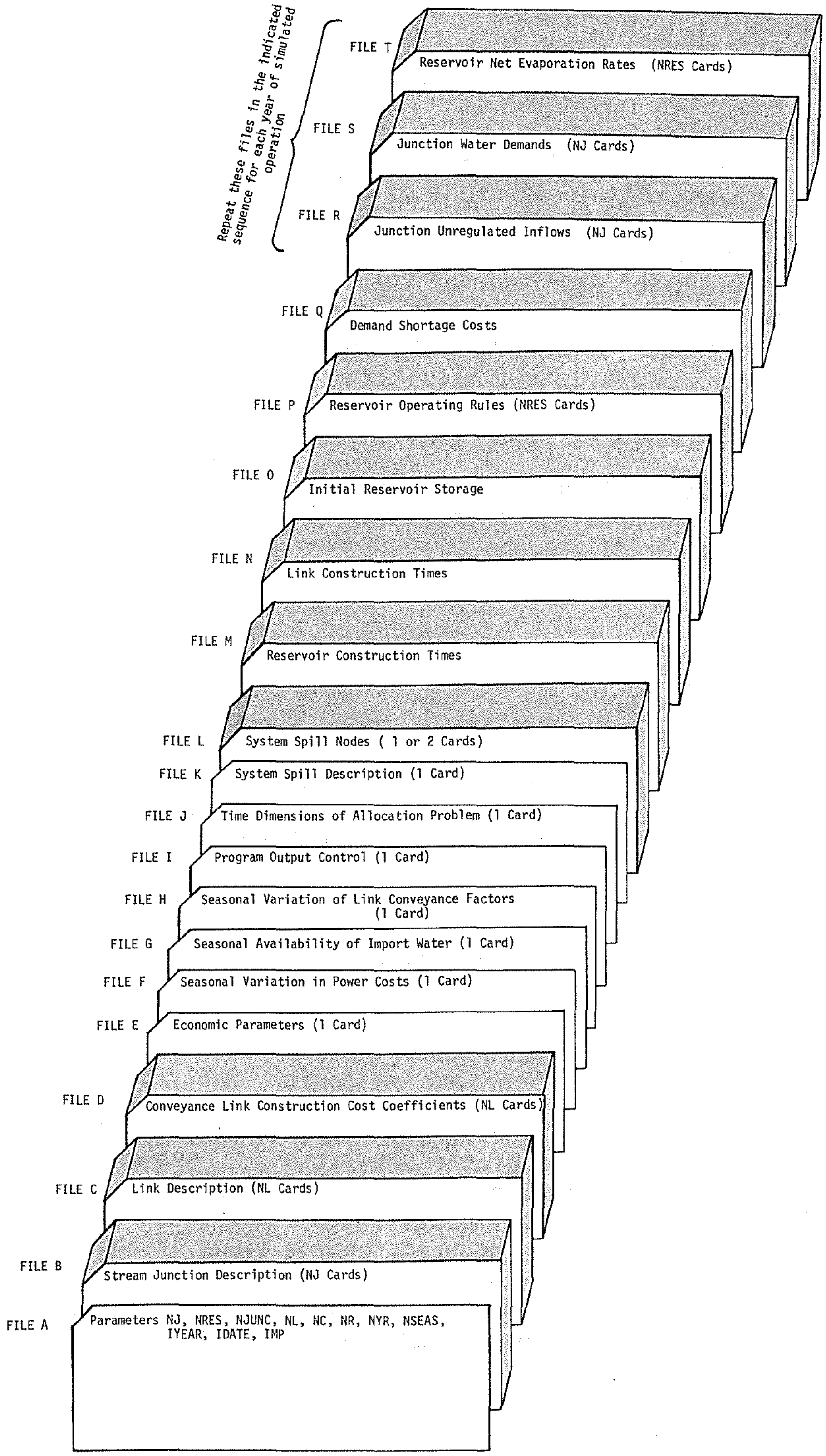


Figure X
AL-IV Input Data Organization

AL-IV OUTPUT FILE DESCRIPTION

The initial output of the Allocation Model consists of printing the card input data and a summary of the structure of the problem. Following this is output from Subroutine APRINT describing the annual simulation results. As previously described, this output is controlled by the variable IPRNT. Data that can be printed for each year of the simulation are described in the following sections. At the end of the simulation output, Subroutine SPRINT prints total costs, present values of total costs, and average annual costs.

INPUT DATA

Table III (page 55) illustrates a sample printing of the card input data. Information on the problem is grouped by junctions, links, the beginning year, the number of seasons in each year and the number of years in each network.

ANNUAL SIMULATION RESULTS

Link Flows

Table IV (page 63) shows seasonal and yearly average link flows and is printed for each year of the simulation. Flows are expressed in cubic feet per second for each link by season. An arithmetic average of all seasonal link flows is printed to the right of the seasonal flow. The first 12 links represent river reaches while the remaining 11 are pipelines. Water imported to the junction specified in Table IV is the last entry in this table. The yearly value is the total amount imported, in acre-feet, during the year. Seasonal values are in units of cubic feet per second.

Link Power Costs

Table V (page 64) shows seasonal and yearly average link power costs that can be printed for each year of the simulation. Costs are given in thousands of dollars. An arithmetic average of all seasonal power costs is printed to the right of all seasonal costs. Since the first 12 links are river reaches, no power costs are incurred for the flows in these links. Power costs in Table V correspond to link flows in Table IV; however, when the power costs is less than one thousand dollars the cost is truncated to zero. The total annual pumping cost (in thousands of dollars) is printed at the bottom of this table.

Marginal Costs for Link Constraints

Table VI (page 65) shows the costs that would be incurred if it was necessary to change the link constraints by one cfs. These costs are termed "marginal costs for link constraints." They can be printed by season at the end of each year in the simulation period or suppressed entirely. The values may be positive, negative, or zero. Positive marginal costs are associated with the lower flow (bound) constraints. For example, if it is necessary to provide a minimum flow in a particular link and the optimum (minimum cost) solution is constrained by this minimum flow, the penalty (cost) we pay for having that constraining flow is the marginal cost. Since the cost of one cfs in a link is dependent on the status of the entire system network, the marginal cost refers only to the last unit increment (one cfs) of link flow.

Negative marginal costs relate to upper flow (bound) constraints. For example, if the flow in a particular link is at its upper limit and the optimum (minimum cost) solution is constrained by this condition, a negative marginal cost indicates the saving that would be achieved if the upper flow constraint was increased by one cfs.

Zero marginal cost indicates that the link flow is not being constrained.

Reservoir Storage Levels

Table VII (page 66) shows seasonal and yearly average storage for reservoirs and is printed for each year of the simulation. Seasonal and yearly average storage is expressed in thousands of acre-feet. Of the 15 nodes within the network, the first 10 are reservoirs capable of storing water. The remaining 5 nodes are junctions that cannot store water, and hence there is no output for them.

Marginal Costs for Reservoir Constraints

Table VIII(page 67) shows the costs that would be incurred if it were necessary to change the volume of water stored in a particular reservoir by one acre-foot. These costs are termed "marginal costs for the reservoir constraints." They are printed by season at the end of each year in the simulation period. These values may be positive, negative, or zero. Positive marginal costs relate to lower storage (bound) constraints. For example, if it is necessary to maintain a minimum pool elevation, and the optimum (minimum cost) solution is constrained by that elevation, the marginal cost is the cost of retaining the last acre-foot of water in the reservoir. Because the cost of storage depends on the status of the entire network, the marginal cost refers only to the last acre-foot of storage.

Negative marginal costs are associated with upper storage (bound) constraints. For example, if a pool elevation is at its upper limit, and the optimum (minimum cost) solution is constrained by this elevation, a negative marginal cost indicates the saving that could be realized if that reservoir could store an additional acre-foot of water.

Zero marginal costs indicate that the reservoir storage is between its upper and lower bound (elevations) for an optimum solution and that no gain or loss will occur if the constraints are changed.

Marginal Costs for Continuity Constraints

Table IX (page 68) shows the cost of delivering one additional acre-foot of water to each node in the network. These values, termed "marginal costs for continuity constraints," are expressed in dollars per acre-foot.

Average Spills

Table X (page 69) shows the seasonal and yearly average spills from those nodes whose spills leave the system. These results may be printed for each year in the simulation period or they can be suppressed. The nodes are identified in Table III. Seasonal and yearly average spills are expressed in thousands of acre-feet.

Maximum Link Flows

Table XI (page 70) shows the maximum link flows that have occurred up to and including the year being printed. Values are in cubic feet per second.

COST RESULTS

Total Costs

Table XII summarizes total costs at the end of the simulation. Costs in thousands of dollars are accumulated by year. Costs are shown separately for physical facilities and for capital and operation and maintenance.

Present Values of Total Costs

Table XIII summarizes the present values of the total costs in Table XII. Present values refer to the base year shown in the title of this table. Like total costs, present values are accumulated. Costs are shown separately for physical facilities and for present values of capital and operation and maintenance. The entry corresponding to the last year in this table shown the present value of total costs to construct, operate, and maintain all facilities, including the cost of imported water over the simulation period.

Annual Costs

Table XIV summarizes annual costs for construction, operation, maintenance imported water, and power for the duration of the simulation period. The repayment period is shown in the title of this table. The totals at the end of this table summarize the estimated total annual cost outlay over the indicated repayment period.

PROBLEM INFEASIBILITY

The program will terminate if any solution to the network problem is found to be infeasible (ie., fails to satisfy all constraints). Solution infeasibility is generally due to one or more of the following conditions:

- improper specifications of the desired system configuration,
- inadequate number of spill nodes,
- a minimum river or canal capacity that is too binding,
- an unregulated inflow occurring where there is no possible outlet for the water,
- improper specification of the basic hydrologic and demand data.

If problem infeasibility occurs then the user should examine the input specifications and the network dump printed by the program to verify the accuracy of the input data.

AL-IV EXAMPLE PROBLEM

The Allocation Program was executed for a typical water resources system to provide a demonstration of the program's capabilities. The example system treated in this analysis is the San Antonio - Guadalupe River Basin in South Central Texas. The major existant and proposed water storage and conveyance projects in the basin are depicted in Figure XI. The physical elements in this system were aggregated into an associated network representation as shown in Figure XII.

In addition to the existing reservoirs indicated in Figure XI, this example operational analysis of the San Antonio - Guadalupe System included the Cibolo and Cuero I & II reservoirs, pipelines from Cuero to Cibolo to San Antonio, and a pipeline from Lake McQueeney to San Antonio. The water demands on the system are the projected intake requirements over the 2011-202 period. The natural hydrologic inputs are derived from the critical historic period 1947-1957.

Two yearly periods, each subdivided into four seasons, were spanned by each operational network. The critical annual terminal storage targets spec for the Cibolo Reservoir were provided by the output of the DPSIM Program applied to this system (5).

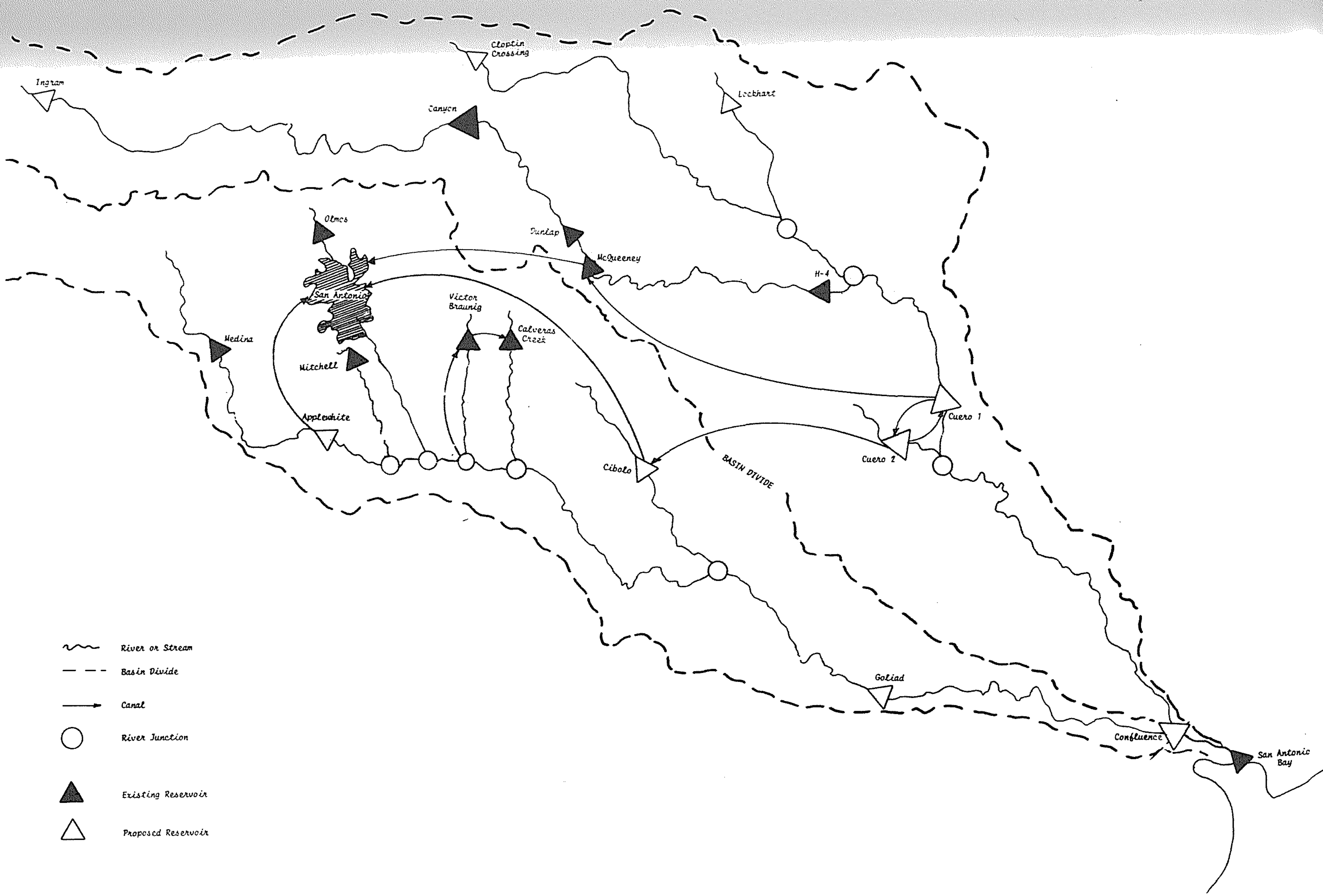


Figure XI
Guadalupe and San Antonio River Basins

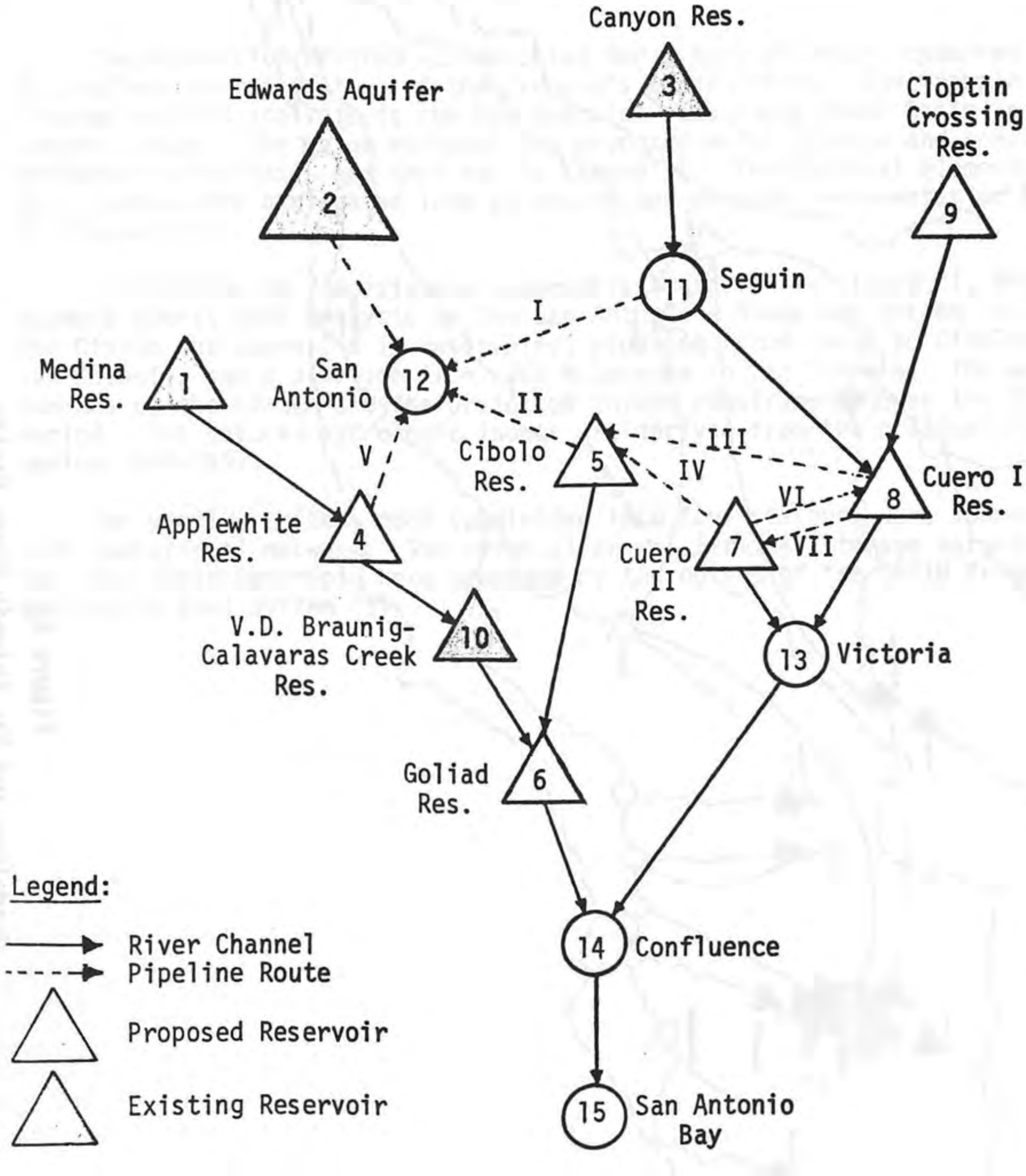


Figure XII

Network Representation of the San Antonio -

Guadalupe River System

AL-IV
EXAMPLE OUTPUT

Figure 11

Network Representation of the San Antonio -

Guadalupe River System

```
****************************************************************
            TEXAS WATER DEVELOPMENT BOARD
                  ALLOCATION MODEL
****************************************************************


DETAILS OF THE PROBLEM ARE AS FOLLOWS

 1) THERE ARE 15 JUNCTIONS, OF THESE

   A) THE FIRST 10 ARE RESERVOIRS
   B) THE LAST   5 ARE LINK JUNCTIONS
   C) IMPORTED WATER ENTERS AT NODE  0
   D) SPILLS FROM THE SYSTEM CAN OCCUR AT 1
      NODES, THESE ARE NODES
             15

 2) THERE ARE 23 LINKS,OF THESE

   A) THE FIRST 12 ARE RIVER REACHES
   B) THE LAST  11 ARE PUMP-CANALS

 3) THERE ARE 11 YEARS IN THE PROBLEM
   A) THE FIRST YEAR IS  1
   B) EACH YEAR HAS  4 SEASONS
   C) EACH YEARLY SOLUTION REQUIRES A 2
      YEAR NETWORK
   D) THE LAST YEAR IS 11
```

Table III-A

Tabled Printing of Input Data

****

NODE DATA, EITHER RESERVOIR OR JUNCTION

****

NODE NO.	RES OR JNC NAME	RESEVR MAX 1000AF	RESEVR MIN 1000AF	RESEVR COST 1000$	RETURN FLOW NODE	RETURN FLOW S/W	BRKPT STORAGE 1000AF	LOWER STORAGE SLOPE	UPPER STORAGE SLOPE	SIMULATION YEAR RES BUILT
1	MEDINA	254.0	2.3	.0	0	.00	62.0	.0368	.0174	1
2	EDWARDA	20000.0	.0	.0	11	.65	.0	.0000	.0000	1
3	CANYON	386.2	3.0	.0	0	.00	43.0	.0595	.0158	1
4	APLE WHT	40.0	1.0	23500.0	0	.00	40.0	.0550	.0000	13
5	CIBOLO	200.0	28.0	46920.0	0	.00	38.0	.0853	.0412	1
6	GOLIAD	400.0	42.0	38621.0	0	.00	220.0	.0681	.0227	13
7	CUERO2	2886.0	50.0	94885.0	0	.00	750.0	.0613	.0273	1
8	CUERO1	1416.0	42.0	114000.0	0	.00	130.0	.0923	.0311	13
9	CLOP CRX	147.0	3.2	55436.0	0	.00	60.0	.0381	.0179	13
10	CAL-VDB	88.7	55.5	.0	6	.55	13.0	.1150	.0430	1
11	SEGUIN	.0	.0	.0	8	.60	.0	.0000	.0000	0
12	SA M.-I.	.0	.0	.0	10	.65	.0	.0000	.0000	0
13	CU1-CU2	.0	.0	.0	14	.61	.0	.0000	.0000	0
14	CONFLU	.0	.0	.0	0	.00	.0	.0000	.0000	0
15	SA BAY	.0	.0	.0	0	.00	.0	.0000	.0000	0

Table III-B

Tabled Printing of Input Data

LINK DATA CANAL OR RIVER

LINK NO.	LINK NAME	ORIGIN NODE	TERMINAL NODE	MAXIMUM FLOW CFS	MINIMUM FLOW CFS	TOTAL DYNAMIC PUMPING HEAD FT	SEASONAL LOSS COEFF.	SIMULATION YEAR LINK BUILT
1	MED-APWT	1	4	.0	.0	.0	.950	1
2	APLWT-10	4	10	90000.0	.0	.0	.950	1
3	10-GOLAD	10	6	90000.0	.0	.0	.950	1
4	CIB-GOLD	5	6	90000.0	.0	.0	.950	1
5	GOL-CONF	6	14	90000.0	.0	.0	.950	1
6	CONF-SAB	14	15	90000.0	.0	.0	.950	1
7	CAYN-SEG	3	11	90000.0	.0	.0	.950	1
8	SEG-CUR1	11	8	90000.0	.0	.0	.950	1
9	LX-CUR1	9	8	90000.0	.0	.0	.950	1
10	CUR2-13	7	13	90000.0	.0	.0	.950	1
11	CUR1-13	8	13	90000.0	.0	.0	.950	1
12	13-CONFL	13	14	90000.0	.0	.0	.950	1
13	MED-EDWD	1	2	.0	.0	.0	.950	1
14	EDWD-CR1	2	8	.0	.0	.0	.950	1
15	EDWD-SEG	2	11	.0	.0	.0	.950	1
16	CUR1-CR2	8	7	90000.0	.0	.0	.950	1
17	EDWD-SAD	2	12	300.0	.0	60.0	1.000	1
18	SEG-SADM	11	12	100.0	100.0	500.0	1.000	1
19	CIB-SADM	5	12	360.0	.0	654.0	1.000	1
20	CUR2-CIB	7	5	325.0	.0	358.0	1.000	1
21	APLWT-SA	4	12	.0	.0	86.0	1.000	1
22	CUR1-CIB	8	5	.0	.0	358.0	1.000	1
23	CUR2-CR1	7	8	.0	.0	.0	1.000	1

Table III-C

Tabled Printing of Input Data

CANAL COST COEFF.

	DITCH			PUMP		
	C1	C2	C3	C1	C2	C3
1	.0000	.0000	.0000	.0000	.0000	.0000
2	.0000	.0000	.0000	.0000	.0000	.0000
3	.0000	.0000	.0000	.0000	.0000	.0000
4	.0000	.0000	.0000	.0000	.0000	.0000
5	.0000	.0000	.0000	.0000	.0000	.0000
6	.0000	.0000	.0000	.0000	.0000	.0000
7	.0000	.0000	.0000	.0000	.0000	.0000
8	.0000	.0000	.0000	.0000	.0000	.0000
9	.0000	.0000	.0000	.0000	.0000	.0000
10	.0000	.0000	.0000	.0000	.0000	.0000
11	.0000	.0000	.0000	.0000	.0000	.0000
12	.0000	.0000	.0000	.0000	.0000	.0000
13	.0000	.0000	.0000	.0000	.0000	.0000
14	.0000	.0000	.0000	.0000	.0000	.0000
15	.0000	.0000	.0000	.0000	.0000	.0000
16	.0000	.0000	.0000	.0000	.0000	.0000
17	.0000	.0000	.0000	.0000	.0000	.0000
18	868.1	.4516	.0000	.0000	.0000	.0000
19	1081.	.4769	.0000	.0000	.0000	.0000
20	3670.	.1909	.0000	.0000	.0000	.0000
21	1219.	.0000	.0000	.0000	.0000	.0000
22	3670.	.1909	.0000	.0000	.0000	.0000
23	.0000	.0000	.0000	.0000	.0000	.0000

Table III-D

****
CONSTANT COST DATA
****

INTEREST RATE =	.0600	REPAYMENT PERIOD YRS =	20 YRS
POWER COST =	.0080 $/KW-HR	O+M ANN DEC PCT RESVR =	.0010
O+M ANN DEC PCT CANAL =	.0010	COST OF IMPORT WATER =	1.0000 $/AF
RESVR FINANCE LAG TIME =	0 YRS	CANAL FINANCE LAG TIME=	0 YRS

POWER FACTOR

SEASONS

1	2	3	4	5	6	7	8	9	10	11	12
1.0000	1.0000	1.0000	1.0000								

IMPORT COEFFS

SEASONS

1	2	3	4	5	6	7	8	9	10	11	12
1.0000	1.0000	1.0000	1.0000								

LOSS COEFFS

SEASONS

1	2	3	4	5	6	7	8	9	10	11	12
1.0300	1.0500	1.0000	1.0300								

Table III-E

Tabled Printing of Input Data

UNIT SHORTAGE COST

($/AC-FT)

NODE NO.	SEASONS 1	2	3	4	5	6	7	8	9	10	11	12
1	100.	100.	100.	100.								
2	100.	100.	100.	100.								
3	115.	115.	115.	115.								
4	40.	40.	40.	40.								
5	40.	40.	40.	40.								
6	60.	60.	60.	60.								
7	40.	40.	40.	40.								
8	105.	105.	105.	105.								
9	90.	90.	90.	90.								
10	80.	80.	80.	80.								
11	110.	110.	110.	110.								
12	100.	100.	100.	100.								
13	103.	103.	103.	103.								
14	40.	40.	40.	40.								
15	0.	0.	0.	0.								

Table III-F

Tabled Printing of Input Data

****
RESERVOIR START CONTENTS
****

NODE NO.	RES OR JNC NAME	PCT FULL-DECIMAL STRT STOR
1	MEDINA	100.0000
2	EDWARDA	100.0000
3	CANYON	100.0000
4	APLE WHT	75.0000
5	CIBOLO	75.0000
6	GOLIAD	75.0000
7	CUERO2	75.0000
8	CUERO1	75.0000
9	CLOP CRX	75.0000
10	CAL-VDB	75.0000

Table III-G

Tabled Printing of Input Data

****

RESERVOIR OPERATING POLICIES

****

RES. NO.	OPERATING OPTION	STORAGE BENEFIT	END-OF-YEAR TARGET STORAGE LEVELS BY YEAR (FRACTION OF MAX. STORAGE) 1	2	3	4	5	6	7	8	9	10	11	12	13	ALPHA
1	1	1.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
2	1	1.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
3	1	4.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
4	1	1.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
5	2	.0	.20	.20	.20	.20	.20	.20	.29	.29	.22	.14	.14	.14	.14	.10
6	1	1.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
7	1	.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
8	1	.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
9	1	1.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00
10	1	5.0	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.00

Table III-H

Tabled Printing of Input Data

SEASONAL AND YEARLY AVERAGE LINK FLOWS IN CFS - YEAR 1

SEASON / LINK	1	2	3	4	
1	0	0	0	0	0
2	94	72	50	72	72
3	488	218	341	131	294
4	0	0	0	0	0
5	699	415	528	315	489
6	763	539	321	227	462
7	754	374	95	90	328
8	766	342	0	0	277
9	216	66	28	17	81
10	0	0	0	102	25
11	90	111	146	0	86
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	1610	783	250	112	688
17	300	300	300	300	300
18	100	100	100	100	100
19	247	247	247	247	247
20	0	0	33	325	89
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
IMP	0	0	0	0	0

Table IV

Seasonal and Yearly Average Link Flows

SEASONAL AND YEARLY AVERAGE LINK POWER COSTS IN THOUSAND DOLLARS - YEAR 1

SEASON / LINK	1	2	3	4	
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	33	33	33	33	33
18	92	92	92	92	92
19	298	299	299	299	298
20	0	0	21	215	59
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0

TOTAL PUMPING COST = 1931
TOTAL IMPORT COST = 0
TOTAL PENALTY COST = 0

TOTAL COST = 1931

Table V

Seasonal and Yearly Average Link Power Costs

SEASONAL DUAL VALUES FOR THE LINK CONSTRAINTS IN THOUSAND DOLLARS PER CFS - YEAR 1

SEASON / LINK	1	2	3	4
1	.50	.52	.49	.52
2	.00	.00	.01	.00
3	.00	.00	.01	.00
4	.81	.82	.83	.89
5	.00	.00	.01	.00
6	.00	.00	.01	.00
7	-.00	-.00	-.07	-.03
8	-.00	-.00	1.43	1.43
9	-.00	.00	-.00	.00
10	.00	-.00	-.00	.00
11	-.00	-.00	-.00	.00
12	.18	.18	.17	.18
13	-.02	-.02	-.02	-.01
14	.36	.36	.36	.36
15	.36	.36	-1.07	-1.07
16	-.00	.00	.00	-.00
17	-1.37	-1.38	-1.40	-1.45
18	-.92	-.93	.49	.44
19	.00	.00	-.00	-.00
20	.03	.02	.00	-.05
21	-1.85	-1.87	-1.86	-1.93
22	.03	.02	.00	-.05
23	.00	-.00	-.00	.00

Table VI

Marginal Costs for Link Constraints

SEASONAL AND YEARLY AVERAGE RESERVOIR STORAGE LEVELS IN THOUSAND ACRE-FEET - YEAR 1

SEASON	1	2	3	4	
RES.					
1	253	249	241	238	245
2	19943	19884	19825	19763	19855
3	386	386	376	367	378
4	0	0	0	0	0
5	121	92	50	63	81
6	0	0	0	0	0
7	2453	2573	2468	2345	2459
8	0	0	0	0	0
9	0	0	0	0	0
10	56	89	61	89	73

SEASONAL JUNCTION DEFICITS IN THOUSANDS OF ACRE-FEET YEAR- 1

SEASON	1	2	3	4
RES.				

Table VII

Seasonal and Yearly Average Reservoir Storage

SEASONAL DUAL VALUES FOR THE RESERVOIR CONSTRAINTS IN DOLLARS PER ACRE-FOOT - YEAR 1

SEASON / RES.	1	2	3	4
1	-.01	-.02	-.05	-.01
2	.00	.00	.00	-.00
3	-.00	-7.54	-.23	-.05
4	.07	-.17	.10	.00
5	-.04	-.10	-.27	-.15
6	.04	-.09	.06	.00
7	-.00	.00	.00	.00
8	-.00	.00	.00	-.00
9	-.00	.00	-.00	-.00
10	.05	-.13	.08	-5.00

Table VIII

Marginal Costs for Reservoir Constraints

SEASONAL DUAL VALUES FOR THE CONTINUIY CONSTRAINTS IN DOLLARS PER ACRE-FOOT - YEAR 1

SEASON NODE	1	2	3	4
1	1.87	1.88	1.90	1.95
2	2.00	2.00	2.00	2.00
3	.00	.00	7.54	7.77
4	-.92	-.99	-.81	-.92
5	3.53	3.57	3.67	3.94
6	-.96	-.99	-.90	-.96
7	.00	.00	.00	.00
8	.00	.00	.00	.00
9	.00	.00	.00	.00
10	-.94	-.99	-.86	-.94
11	.00	.00	7.94	7.94
12	10.22	10.26	10.36	10.64
13	.00	.00	.00	.00
14	-.98	-1.00	-.95	-.98
15	-1.00	-1.00	-1.00	-1.00

Table IX

Marginal Costs for Continuity Constraints

SEASONAL AND YEARLY AVERAGE SPILLS FROM THE SYSTEM IN THOUSAND ACRE-FEET - YEAR 1

SEASON / NODE	1	2	3	4	
15	134	97	55	40	81

Table X

Seasonal and Yearly Average Spills

MAXIMUM LINK FLOWS IN CFS - YEAR 1

LINK	FLOW	LINK	FLOW
1	0	2	94
3	488	4	0
5	699	6	763
7	754	8	766
9	216	10	102
11	146	12	0
13	0	14	0
15	0	16	1610
17	300	18	100
19	247	20	325
21	0	22	0
23	0		

Table XI

Maximum Link Flows

TOTAL COSTS IN THOUSAND DOLLARS

YEAR	RESERVOIRS CAPITAL	RESERVOIRS O AND M	CONDUITS CAPITAL	CONDUITS O AND M	IMPORTS	POWER	DEFICITS	TOTAL
1	141805.	142.	207421.	207.	0.	1931.	0.	351507.
2	141805.	284.	207421.	415.	0.	4230.	0.	354155.
3	141805.	425.	207421.	622.	0.	6457.	0.	356731.
4	141805.	567.	207421.	830.	0.	9052.	0.	359675.
5	141805.	709.	207421.	1037.	0.	11653.	0.	362625.
6	141805.	851.	207421.	1245.	0.	14353.	0.	365675.
7	141805.	993.	207421.	1452.	0.	17261.	0.	368932.
8	141805.	1134.	207421.	1659.	0.	20228.	0.	372248.
9	141805.	1276.	207421.	1867.	0.	23255.	0.	375624.
10	141805.	1418.	207421.	2074.	0.	26294.	482.	379495.
11	141805.	1560.	207421.	2282.	0.	28873.	1132.	383073.

Table XII

Total Costs at the End of the Simulation

PRESENT VALUES IN THOUSAND DOLLARS (1980 BASE)

YEAR	RESERVOIRS CAPITAL	RESERVOIRS O AND M	CONDUITS CAPITAL	CONDUITS O AND M	IMPORTS	POWER	DEFICITS	TOTAL
1	141805.	142.	207421.	207.	0.	1931.	0.	351507.
2	141805.	276.	207421.	403.	0.	4100.	0.	354005.
3	141805.	402.	207421.	588.	0.	6082.	0.	356298.
4	141805.	521.	207421.	762.	0.	8261.	0.	358770.
5	141805.	633.	207421.	926.	0.	10321.	0.	361107.
6	141805.	739.	207421.	1081.	0.	12330.	0.	363385.
7	141805.	839.	207421.	1227.	0.	14382.	0.	365681.
8	141805.	933.	207421.	1365.	0.	16362.	0.	367887.
9	141805.	1022.	207421.	1495.	0.	18261.	0.	370005.
10	141805.	1106.	207421.	1618.	0.	20060.	285.	372296.
11	141805.	1186.	207421.	1734.	0.	21500.	648.	374294.

Table XIII

Present Values of the Total Costs

AVERAGE ANNUAL COSTS IN THOUSAND DOLLARS (20-YEAR PAYMENTS)

YEAR	RESERVOIRS CAPITAL	RESERVOIRS O AND M	CONDUITS CAPITAL	CONDUITS O AND M	IMPORTS	POWER	DEFICITS	TOTAL
1	12363.	12.	18084.	18.	0.	168.	0.	30646.
2	12363.	24.	18084.	35.	0.	357.	0.	30864.
3	12363.	35.	18084.	51.	0.	530.	0.	31064.
4	12363.	45.	18084.	66.	0.	720.	0.	31279.
5	12363.	55.	18084.	81.	0.	900.	0.	31483.
6	12363.	64.	18084.	94.	0.	1076.	0.	31682.
7	12363.	73.	18084.	107.	0.	1254.	0.	31882.
8	12363.	81.	18084.	119.	0.	1426.	0.	32074.
9	12363.	89.	18084.	130.	0.	1592.	0.	32259.
10	12363.	96.	18084.	141.	0.	1749.	25.	32458.
11	12363.	103.	18084.	151.	0.	1874.	57.	32633.

Table XIV

Annual Costs

AL-IV

PROGRAM LISTING

- MAIN PROGRAM -

```
C***********************************************************************
C                 THIS IS THE MAIN PROGRAM FOR SOLVING THE ALLOCA-
C                 TION PROBLEM. ALL SUBROUTINES ARE CALLED FROM IT. THE
C                 RESULTS ARE THE LEAST COSTLY WAY TO OPERATE A SYSTEM
C                 OF RESERVOIRS AND CANALS TO MEET A GIVEN SET OF
C                 DEMANDS.
C***********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /JDATA/ RNAME(30,2),RCAP(30),RETSW(30),CRES(30),RMIN(30),NL
     1INK(30,8),FSTART(30),DLO(30)
      COMMON /LDATA/ CNAME(45,2),CLINK(2,45,3),CCAP(45),CPUMP(45),CMIN(4
     15),CLIFT(45),EL(45),LNODE(45,2)
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LOWER(1800),ARCS
      COMMON /MDATA/ XX(12),ELF(12),XIMP,KSPILL
      COMMON /MISC/ AMIN,ADELT,NDELT
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      REAL LOWER
      INTEGER ARCS,SOURCE,SINK
C***********************************************************************
C                 READ SYSTEM INPUT DATA FROM CARDS
C***********************************************************************
      CALL INPUT
      IFIG=1
      NREAD=1
C***********************************************************************
C                 SOLVE THE ALLOCATION PROBLEM.
C                 FIRST CONSTRUCT A NETWORK OF
C                 NSOLVE YEARS. THEN PROCEED
C                 THROUGH TIME BY ADDING A YEAR
C                 TO AND DROPPING A YEAR FROM THE
C                 NETWORK AND SOLVING THE NETWORK
C                 PROBLEM.  EACH SOLUTION PROVIDES
C                 THE RESULTS FOR THE FIRST YEAR
C                 IN THE NETWORK.  THIS YEAR WILL
C                 BE DROPPED FROM THE NETWORK IN
C                 THE NEXT SOLUTION.
C***********************************************************************
      DO 60 IYEAR=NREAD,NYEAR
      CALL ARC
      IF (NSOLVE.GT.IYEAR) GO TO 60
      CALL GAIN
20    CALL OUTPUT
C***********************************************************************
C                 INITIATE RESTART PROCEDURE FOR NETWORK MODEL
C                 IF RESERVOIR STORAGE SEGMENTATION CONSTRAINTS
C                 ARE VIOLATED
C***********************************************************************
      IF (TIMAX-1.) 50,30,30
30    DO 40 I=1,ARCS
40    FLOW(I)=0.
      CALL GAIN
      GO TO 20
50    CALL PREPAR
      CALL APRINT
      CALL ADJUST
      CALL SHIFT
60    CONTINUE
      IF (NSOLVE.EQ.1) GO TO 80
      DO 70 ISOLVE=2,NSOLVE
      IYEAR=NYEAR-1+ISOLVE
      CALL PREPAR
      CALL APRINT
      CALL SHIFT
70    CONTINUE
80    CONTINUE
C***********************************************************************
C                 CALCULATE COSTS AND PRINT
C                 ANNUAL SUMMARIES.
C***********************************************************************
      CALL COSTX
      CALL SPRINT
      STOP
      END
```

```
      SUBROUTINE ADJUST
C***********************************************************************
C                 THIS SUBROUTINE FINDS THE ADJUSTMENTS THAT MUST
C                 BE MADE IN THE LOWER AND UPPER BOUNDS OF THE NET
C                 BALANCE ARCS WHEN A SOLUTION HAS BEEN FOUND FOR
C                 ONE YEAR.THIS YEAR IS THEN DROPPED FROM THE
C                 NETWORK
C***********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LOWER(1800),ARCS
      COMMON /MISC/ AMIN,ADELT,NDELT
      COMMON /ADJST/ IADJ,SADJ,MADJ,DADJ,IDROP,JDROP,MDROP,SDROP,ISTORE,
     1ISADJ
      INTEGER A,AA,AMIN,AIMIN,ARMIN,ASMIN,DADJ,SADJ,ARCS
      INTEGER SDROP,AB
      REAL LOWER
C***********************************************************************
C                 INITIALIZE VARIABLES
C***********************************************************************
      IADJ=0
      MADJ=0
      DADJ=0
      ISADJ=0
      SADJ=0
      IDROP=0
      JDROP=0
      MDROP=0
      SDROP=0
C***********************************************************************
C                 MAIN COMPUTATIONAL LOOP
C***********************************************************************
      DO 40 I=1,NSEAS
      AIMIN=AMIN+(I-1)*(NJ+1+2*NRES+NL+NS+NJ)
      ARMIN=AIMIN+NJ+1
      ASMIN=ARMIN+2*NRES+NL+NS
C***********************************************************************
C                 CALCULATE ADJUSTMENTS FOR THE BOUNDS ON THE
C                 NET INPUT AND DEMAND ARCS
C***********************************************************************
      DO 20 J=1,NJ
      A=AIMIN+J
      DADJ=DADJ+HI(A)
      JDROP=JDROP+FLOW(A)
20    CONTINUE
      A=ARMIN
      MADJ=MADJ+HI(A)
      MDROP=MDROP+FLOW(A)
C***********************************************************************
C                 CALCULATE ADJUSTMENTS FOR THE UPPER BOUND ON
C                 THE NET SPILL ARC
C***********************************************************************
      DO 30 J=1,NJ
      A=ASMIN+J
      IADJ=IADJ+HI(A)
30    CONTINUE
40    CONTINUE
C***********************************************************************
C                 CALCULATE ADJUSTMENTS IN THE INITIAL STORAGE ARCS
C***********************************************************************
      ISTORE=0
      DO 50 J=1,NRES
      A=J+6
      AA=J+ARMIN
      AB=AA+NRES
      LOWER(A)=FLOW(AA)*AMP(AA)+FLOW(AB)*AMP(AB)
      HI(A)=LOWER(A)
      FLOW(A)=HI(A)
      ISTORE=ISTORE+FLOW(A)
50    CONTINUE
      IDROP=IADJ
      RETURN
      END
```

```
      SUBROUTINE ADSUB (I,II)
C***********************************************************************
C            SUBROUTINE TO ADD AN ARC TO THE TRIPLE LABEL
C            REPRESENTATION OF THE FLOW AUGMENTING TREE.
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      INTEGER ARCS
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
C***********************************************************************
C            ADDS ARC I TO THE LIST OF SUBSEQUENT ARCS TO NODE II.
C***********************************************************************
      IF (FARC(II).NE.0) GO TO 20
      FARC(II)=I
      GO TO 50
20    CONTINUE
      MM=FARC(II)
30    CONTINUE
      MN=JARC(MM)
      IF (RARC(MN).NE.0) GO TO 40
      RARC(MN)=I
      GO TO 50
40    CONTINUE
      MM=RARC(MN)
      GO TO 30
50    CONTINUE
      RETURN
      END
```



```
      FUNCTION AMPF (I)
C***********************************************************************
C            THIS FUNCTION COMPUTES THE AMPLITUDE (GAIN) OF
C            ARC I
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      IF (I.GT.NARC) GO TO 20
      AMPF=AMP(I)
      RETURN
20    CONTINUE
      AMPF=1./AMP(I-NARC)
      RETURN
      END
```

```
      SUBROUTINE APRINT
C*********************************************************************
C                   THIS SUBROUTINE PRINTS THE ANNUAL SUMMARIES OF THE
C                   LINK FLOWS,    POWER COSTS AND DUAL VALUES,THE
C                   RESERVOIR STORAGE LEVELS AND  DUAL VALUES,AND THE
C                   DUAL VALUES FOR THE CONTINUITY EQUATIONS
C*********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /RDATA/ JS(10),JBLT(30),LBLT(45)
      COMMON /APRNT/ APOW(50),AIMP(50),QM(45),ISHORT(50),ICOST(50)
      COMMON /PRT/ IPRNT(45)
      COMMON LQ(13,45),LP(13,45),LC(12,45),RS(13,30),RC(12,30),CC(12,30)
     1,JQ(13,10),IDEF(13,30),IMPC(50),ITOT(50)
      INTEGER AIMP,RS,QM
      INTEGER APOW
      REAL LC
      IL=NL+1
      M=NSEAS+1
      IYR=IYEAR-NSOLVE+1
20    FORMAT (1H1,5X,45HSEASONAL AND YEARLY AVERAGE LINK FLOWS IN CFS,8H
     1 - YEAR ,I4)
30    FORMAT (1H1,5X,45HSEASONAL AND YEARLY AVERAGE LINK POWER COSTS ,27
     1HIN THOUSAND DOLLARS - YEAR ,I4)
40    FORMAT (1H1,5X,45HSEASONAL DUAL VALUES FOR THE LINK CONSTRAINTS,36
     1H IN THOUSAND DOLLARS PER CFS - YEAR ,I4)
50    FORMAT (1H1,5X,45HSEASONAL AND YEARLY AVERAGE RESERVOIR STORAGE,37
     1H LEVELS IN THOUSAND ACRE-FEET - YEAR ,I4)
60    FORMAT (1H1,5X,45HSEASONAL DUAL VALUES FOR THE RESERVOIR CONSTR,38
     1HAINTS IN DOLLARS PER ACRE-FOOT - YEAR ,I4)
70    FORMAT (1H1,5X,45HSEASONAL DUAL VALUES FOR THE CONTINUIY CONSTR,38
     1HAINTS IN DOLLARS PER ACRE-FOOT - YEAR ,I4)
80    FORMAT (1H1,5X,44HSEASONAL AND YEARLY AVERAGE SPILLS FROM THE ,37H
     1SYSTEM  IN THOUSAND ACRE-FEET - YEAR ,I4)
90    FORMAT (1H1,5X,33HMAXIMUM LINK FLOWS IN CFS - YEAR ,I2)
100   FORMAT (17X,4HLINK,16X,4HFLOW,17X,4HLINK,16X,4HFLOW//(4I20))
110   FORMAT (7H SEASON,I4,11I8,4X,4HYEAR)
120   FORMAT (7H SEASON,I4,11I8)
130   FORMAT (5H LINK)
140   FORMAT (5H RES.)
150   FORMAT (5H NODE)
160   FORMAT (4H IMP,13I8)
170   FORMAT (1H I3,13I8)
180   FORMAT (1H I3,13F8.2)
C*********************************************************************
C                   PRINT SEASONAL AND YEARLY
C                   AVERAGE LINK FLOWS.
C*********************************************************************
      IF (IPRNT(IYR).EQ.3) RETURN
      WRITE (6,20) IYR
      WRITE (6,110) (I,I=1,NSEAS)
      WRITE (6,130)
      DO 190 L=1,NL
      WRITE (6,170) L,(LQ(I,L),I=1,M)
190   CONTINUE
      WRITE (6,160) (LQ(I,IL),I=1,NSEAS),AIMP(IYR)
C*********************************************************************
C                   PRINT SEASONAL AND YEARLY
C                   AVERAGE LINK POWER COSTS.
C*********************************************************************
      WRITE (6,30) IYR
      WRITE (6,110) (I,I=1,NSEAS)
      WRITE (6,130)
      DO 200 L=1,NL
      WRITE (6,170) L,(LP(I,L),I=1,M)
200   CONTINUE
      WRITE (6,210) APOW(IYR),IMPC(IYR),ICOST(IYR),ITOT(IYR)
210   FORMAT (//22H TOTAL PUMPING COST = ,I12,/22H TOTAL IMPORT  COST =
     1,I12,/22H TOTAL PENALTY COST = ,I12,//22H TOTAL         COST = ,I1
     22)
C*********************************************************************
C                   PRINT SEASONAL DUAL VALUES FOR
C                   THE LINK CONSTRAINTS. THESE ARE
C                   THE COST OF MOVING ONE MORE TAF
C                   THROUGH THE LINK.
C*********************************************************************
      IF (IPRNT(IYR).EQ.2) GO TO 230
      WRITE (6,40) IYR
      WRITE (6,120) (I,I=1,NSEAS)
      WRITE (6,130)
      DO 220 L=1,NL
      WRITE (6,180) L,(LC(I,L),I=1,NSEAS)
220   CONTINUE
230   CONTINUE
C*********************************************************************
C                   PRINT SEASONAL AND YEARLY AVERAGE
C                   RESERVOIR STORAGE LEVELS.
C*********************************************************************
      WRITE (6,50) IYR
      WRITE (6,110) (I,I=1,NSEAS)
      WRITE (6,140)
      DO 240 J=1,NRES
      WRITE (6,170) J,(RS(I,J),I=1,M)
240   CONTINUE
C*********************************************************************
C                   PRINT SEASONAL JUNCTION DEFICITS
C*********************************************************************
      WRITE (6,260) IYR
      WRITE (6,120) (I,I=1,NSEAS)
      WRITE (6,140)
      JJ=NSEAS+1
      DO 250 J=1,NJ
      IF (IDEF(JJ,J).EQ.0) GO TO 250
      WRITE (6,170) J,(IDEF(I,J),I=1,NSEAS)
250   CONTINUE
260   FORMAT (///,5X,45H SEASONAL JUNCTION DEFICITS IN THOUSANDS OF A,17
     1HCRE-FEET    YEAR-,I4)
C*********************************************************************
C                   PRINT SEASONAL DUAL VALUES FOR
C                   THE RESERVOIR CONSTRAINTS. THESE
C                   REPRESENT THE COST OF ONE MORE
C                   AF  OF STORAGE.
C*********************************************************************
      IF (IPRNT(IYR).EQ.2) GO TO 290
      WRITE (6,60) IYR
      WRITE (6,120) (I,I=1,NSEAS)
      WRITE (6,140)
      DO 270 J=1,NRES
      WRITE (6,180) J,(RC(I,J),I=1,NSEAS)
270   CONTINUE
C*********************************************************************
C                   PRINT SEASONAL DUAL VALUES FOR
C                   THE CONTINUITY CONSTRAINTS.
C                   THESE REPRESENT THE COST OF
C                   PUMPING ONE MORE AF TO THE NODE.
C*********************************************************************
      WRITE (6,70) IYR
      WRITE (6,120) (I,I=1,NSEAS)
      WRITE (6,150)
      DO 280 J=1,NJ
      WRITE (6,180) J,(CC(I,J),I=1,NSEAS)
280   CONTINUE
290   CONTINUE
C*********************************************************************
C                   PRINT SEASONAL AND YEARLY AVERAGE
C                   SPILLS FROM THE SYSTEM.
C*********************************************************************
      WRITE (6,80) IYR
      WRITE (6,110) (I,I=1,NSEAS)
      WRITE (6,150)
      DO 300 J=1,NS
      WRITE (6,170) JS(J),(JQ(I,J),I=1,M)
300   CONTINUE
C*********************************************************************
C                   WRITE MAXIMUM LINK FLOWS
C*********************************************************************
      WRITE (6,90) IYR
      WRITE (6,100) (L,QM(L),L=1,NL)
      RETURN
      END
```

```
      SUBROUTINE ARC
C***********************************************************************
C                 THIS SUBROUTINE CONVERTS THE INPUT DATA INTO
C                 ARC DATA AS REQUIRED BY SUBROUTINE GAIN.
C                 THESE DATA ARE
C                      1) ORIGIN NODE          = NF(A)
C                      2) DESTINATION NODE     = NT(A)
C                      3) LOWER BOUND ON FLOW = LO(A)
C                      4) UPPER BOUND ON FLOW = HI(A)
C                      5) UNIT COST OF FLOW    = COST(A)
C                      6) GAIN FACTOR          = AMP(A)
C***********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /JDATA/ RNAME(30,2),RCAP(30),RETSW(30),CRES(30),RMIN(30),NL
     1INK(30),FSTART(30),DLO(30)
      COMMON /LDATA/ CNAME(45,2),CLINK(2,45,3),CCAP(45),CPUMP(45),CMIN(4
     15),CLIFT(45),EL(45),LNODE(45,2)
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LO(1800),ARCS
      COMMON /MDATA/ XX(12),ELF(12),XIMP,KSPILL
      COMMON /PRT/ IPRNT(45)
      COMMON /RDATA/ JS(10),JBLT(30),LBLT(45)
      COMMON /CDATA/ PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC,POWFA
     1C(12)
      COMMON /SHORTS/ BND(30),CST(12,30),ACO(30,2),TFAC(13,30),TCST(30),
     1IOPT(30),ALP(30)
      COMMON /MISC/ AMIN,ADELT,NDELT
      COMMON /ADJST/ IADJ,SADJ,MADJ,DADJ,IDROP,JDROP,MDROP,SDROP,ISTORE,
     1ISADJ
      COMMON /XDATA/ PI(500),BARC(500),RARC(500),FARC(500),DISSET(500),G
     1AN(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      COMMON D(12,30),U(30,12),E(12,30)
      DIMENSION STAR(30)
      INTEGER AMIN,AIMIN,ARMIN,ALMIN,ASMIN,ADELT,SADJ,DADJ
      INTEGER DEMAND,SPILL,SCAP,ARCS,A,AA,SDROP,DSHORT
      INTEGER AB,AC,SOURCE,SINK
      REAL LO
C***********************************************************************
C                 INITIALIZE VARIABLES
C***********************************************************************
      IDUM=IYEAR
      KIN=5
      KOUT=6
      IF (IDUM.GE.NSOLVE) GO TO 20
      ISOLVE=IDUM
      GO TO 30
20    ISOLVE=NSOLVE
30    CONTINUE
      NI=ISOLVE*NSEAS
      ARCS=NRES+NI*(NJ+1+2*NRES+NL+NS+NJ)+6
      IF (ARCS.GT.1800) GO TO 40
      NODES=NI*NJ+7
      IF (NODES.GT.500) GO TO 40
      GO TO 60
40    WRITE (KOUT,50) ARCS,NODES
50    FORMAT (10X,71HTHE PROBLEM FORMULATION HAS EITHER MORE THAN  1800
     1 ARCS OR  500  NODES//20X,17HNUMBER OF ARCS = ,I5,5X,19H NUMBER OF
     2 NODES = ,I5//)
      STOP
60    CONTINUE
      FSTORE=0
      IF (IYEAR.GT.NREAD) GO TO 100
      NN=NSOLVE*NSEAS
      AMIN=NRES+6
      ADELT=NSEAS*(NJ+1+2*NRES+NL+NS+NJ)
      NDELT=NSEAS*NJ
      ISTORE=0
      INPUT=0
      DEMAND=0
      DSHORT=0
      IMPORT=0
      SPILL=0
      IADJ=0
      DADJ=0
      MADJ=0
      SADJ=0
      IDROP=0
      JDROP=0
      MDROP=0
      SDROP=0
      SUBS=12./NSEAS
      EFF=.8
      CPUMP(1)=(30.2*24.0*62.4*.75*POWR)/(EFF*550.)/1000.
      DO 70 J=1,NL
70    CPUMP(J)=CPUMP(1)
C***********************************************************************
C                 GENERATE DATA FOR INITIAL STORAGE ARCS
C***********************************************************************
      DO 90 J=1,NRES
      A=J+6
      NF(A)=1
      NT(A)=J+7
      LO(A)=0
      HI(A)=0
      AMP(A)=1.
      COST(A)=0.
      IF (JBLT(J).GT.IYEAR) GO TO 80
      LO(A)=FSTART(J)*RCAP(J)/100.
      HI(A)=LO(A)
80    ISTORE=ISTORE+HI(A)
90    CONTINUE
100   CONTINUE
      DO 110 J=1,ARCS
110   FLOW(J)=0
C***********************************************************************
C                 READ NATURALIZED INFLOWS, PROJECTED DEMANDS,
C                 AND NET EVAPORATION RATES
C                 - INPUT FILES  R,S,T
C***********************************************************************
      DO 120 J=1,NJ
120   READ (KIN,150) (U(J,I),I=1,12)
      DO 130 J=1,NJ
130   READ (KIN,150) (D(I,J),I=1,12)
      DO 140 J=1,NRES
140   READ (KIN,150) (E(I,J),I=1,12)
150   FORMAT (20X,12F5.0)
C***********************************************************************
C                 GENERATE DATA FOR INPUT, DEMAND, AND IMPORT ARCS
C***********************************************************************
      DO 410 ISEAS=1,NSEAS
      I=(ISOLVE-1)*NSEAS+ISEAS
      AIMIN=AMIN+(I-1)*(NJ+1+2*NRES+NL+NS+NJ)
      ARMIN=AIMIN+NJ+1
      ALMIN=ARMIN+2*NRES
      ASMIN=ALMIN+NL
      ASMINS=ASMIN+NS
      DO 190 J=1,NJ
      A=AIMIN+J
      INPUT=INPUT+U(J,ISEAS)
      HI(A)=D(ISEAS,J)
      LO(A)=DLO(J)*HI(A)
      IF(LO(A).GT.HI(A))LO(A)=HI(A)
      AMP(A)=RETSW(J)
      IF (AMP(A)) 160,160,170
160   AMP(A)=1.
170   NF(A)=7+(I-1)*NJ+J
      NT(A)=7+(I-1)*NJ+NLINK(J)
      IF(NLINK(J).EQ.0)NT(A)=3
      LO(A)=0
      DEMAND=DEMAND+HI(A)
      COST(A)=-CST(ISEAS,J)
190   CONTINUE
```

```
      A=ARMIN
      NF(A)=4
      NT(A)=7+(I-1)*NJ+IMP
      IF(IMP.EQ.0)NT(A)=7
      LO(A)=0
      HI(A)=XIMP*XX(ISEAS)
      COST(A)=CIMP
      AMP(A)=1.
      IMPORT=IMPORT+HI(A)
C***********************************************************************
C                    GENERATE DATA FOR RESERVOIR STORAGE ARCS
C***********************************************************************
      NYSTR=NYEAR
      INCRM=-1
      DO 220 J=1,NRES
      STAR(J) = 0.
      IF (IOPT(J).EQ.1) GO TO 220
      AP=ALP(J)
      IF (AP) 220,220,200
200   CONTINUE
      AP1=1.-AP
      SUMT=0
      DO 210 IJ=NYSTR,IYEAR,INCRM
      II=IJ
      IF(IJ.GT.13)II=13
210   SUMT=(TFAC(II,J)+SUMT)/AP1-TFAC(II,J)
      STAR(J)=SUMT
220   CONTINUE
      DO 330 J=1,NRES
      JF=J
      A=ARMIN+J
      NF(A)=7+(I-1)*NJ+JF
      NT(A)=NF(A)+NJ
      LO(A)=0
      HI(A)=0
      COST(A)=0.
      AB=ARMIN+NRES+J
      NF(AB)=NF(A)
      NT(AB)=NT(A)
      LO(AB)=0
      HI(AB)=0
      COST(AB)=0.
      AMP(A)=1.-(E(ISEAS,J)*ACO(J,1))
      AMP(AB)=1.-(E(ISEAS,J)*ACO(J,2))
      IF (JBLT(J).GT.IYEAR) GO TO 300
      SLOW=AMIN1(RMIN(J),BND(J))
      LO(A)=SLOW/AMP(A)
      IF(RMIN(J).GT.BND(J))LO(AB)=(RMIN(J)-BND(J))/AMP(A)
      IYX=IYEAR-JBLT(J)+1
      IF(IYX.GT.13)IYX=13
      TF=TFAC(IYX,J)+STAR(J)
      IF(TF.GT.1.)TF=1.
      IF(TF*RCAP(J).LT.RMIN(J))TF=RMIN(J)/RCAP(J)
      BN=BND(J)/RCAP(J)
      IF (IOPT(J)-1) 300,230,260
C***********************************************************************
C                    RESERVOIR OPERATING RULE - OPTION 1
C***********************************************************************
230   HI(A)=BND(J)/AMP(A)
      IF (ISEAS-NSEAS) 250,240,240
240   IF(TF.LT.BN) TF=BN
      HI(AB)=RCAP(J)*(TF-BN)/AMP(AB)
      COST(AB)=-TCST(J)
      COST(A)=-TCST(J)-1.
      GO TO 300
250   HI(AB)=RCAP(J)*(1.-BN)/AMP(AB)
      COST(A)=-1.
      GO TO 300
C***********************************************************************
C                    RESERVOIR OPERATING RULE - OPTION 2
C***********************************************************************
260   HI(A)=BND(J)/AMP(A)
      IF (ISEAS-NSEAS) 270,280,280
270   HI(AB)=RCAP(J)*(1.-BN)/AMP(AB)
      COST(A)=-TCST(J)-1.
      IF (BN.GT.TF) GO TO 300
      LO(A)=HI(A)
      GO TO 300
280   IF (BN.GT.TF) GO TO 290
      LO(A)=HI(A)
      HI(AB)=RCAP(J)*(1.-BN)/AMP(AB)
      IF(TF*RCAP(J).GT.RMIN(J))LO(AB)=RCAP(J)*(TF-BN)/AMP(AB)
      COST(AB)=-TCST(J)
      GO TO 300
290   LO(A)=RCAP(J)*TF/AMP(A)
      COST(A)=-TCST(J)
300   CONTINUE
C***********************************************************************
C                    CORRECT FINAL STORAGE ARCS
C***********************************************************************
      IF (I.NE.NI) GO TO 330
      NT(A)=6
      NT(AB)=6
      FSTORE=FSTORE+HI(A)+HI(AB)
      IF (ISOLVE.EQ.1) GO TO 330
      AA=A-ADELT
      NT(AA)=NF(AA)+NJ
      AC=AB-ADELT
      IF (IOPT(J).EQ.1) GO TO 320
      TF=TFAC(IYX-1,J)
      IF (BN.GT.TF) GO TO 310
      IF(TF*RCAP(J).LT.RMIN(J))TF=RMIN(J)/RCAP(J)
      LO(AC)=RCAP(J)*(TF-BN)/AMP(AC)
      GO TO 320
310   CONTINUE
      LO(AC)=0.
      HI(AC)=0.
      LO(AA)=RCAP(J)*TF/AMP(AA)
320   CONTINUE
      NT(AC)=NT(AA)
330   CONTINUE
C***********************************************************************
C                    GENERATE DATA FOR CANAL AND RIVER ARCS
C***********************************************************************
      DO 380 L=1,NL
      JF=LNODE(L,1)
      JT=LNODE(L,2)
      A=ALMIN+L
      NF(A)=7+(I-1)*NJ+JF
      NT(A)=7+(I-1)*NJ+JT
      LO(A)=0
      HI(A)=0
C***********************************************************************
C                    CONVEYANCE FACTORS FOR PIPELINES ARE NOT
C                    ADJUSTED SEASONALLY
C***********************************************************************
      IF (EL(L)-1) 350,340,350
340   AMP(A)=EL(L)
      GO TO 360
350   AMP(A)=EL(L)*ELF(ISEAS)
360   CONTINUE
      IF(LBLT(L).LE.IYEAR)HI(A)=CCAP(L)/16.6*SUBS
      IF(LBLT(L).LE.IYEAR)LO(A)=CMIN(L)/16.6*SUBS
      COST(A)=CPUMP(L)*CLIFT(L)*POWFAC(ISEAS)*16.6
      IF (COST(A)) 370,370,380
370   COST(A)=.0
380   CONTINUE
C***********************************************************************
C                    GENERATE DATA FOR SPILL ARCS
C***********************************************************************
      DO 390 J=1,NS
      JF=JS(J)
      A=ASMIN+J
      NF(A)=7+(I-1)*NJ+JF
      NT(A)=5
      LO(A)=0
      HI(A)=SCAP
      AMP(A)=1.
      COST(A)=KSPILL
      IF(KSPILL.LT.1)COST(A)=.001
      SPILL=SPILL+HI(A)
390   CONTINUE
```

```
C***********************************************************************
C                 GENERATE DATA FOR INPUT ARCS
C***********************************************************************
      DO 400 J=1,NJ
      A=ASMINS+J
      NF(A)=2
      NT(A)=7+(I-1)*NJ+J
      AMP(A)=1.
      HI(A)=U(J,ISEAS)
      LO(A)=HI(A)
      COST(A)=0.
400   CONTINUE
410   CONTINUE
C***********************************************************************
C                 GENERATE DATA FOR NET BALANCE ARCS
C***********************************************************************
      A=1
      NF(A)=1
      NT(A)=7
      HI(A)=99999.
      AMP(A)=1.
      COST(A)=.001
C***********************************************************************
C                 TOTAL INPUT ARC
C***********************************************************************
      A=2
      NF(A)=1
      NT(A)=2
      INPUT=INPUT-IADJ
      LO(A)=INPUT
      HI(A)=LO(A)
      AMP(A)=1.
      COST(A)=0.
C***********************************************************************
C                 TOTAL DEMAND ARC
C***********************************************************************
      A=3
      NF(A)=3
      NT(A)=7
      LO(A)=0.
      HI(A)=9999.
      AMP(A)=1.
      COST(A)=0.
C***********************************************************************
C                 TOTAL IMPORT ARC
C***********************************************************************
      A=4
      NF(A)=1
      NT(A)=4
      LO(A)=0
      IMPORT=IMPORT-MADJ
      HI(A)=9999.
      AMP(A)=1.
      COST(A)=0
C***********************************************************************
C                 TOTAL SPILL ARC
C***********************************************************************
      A=5
      NF(A)=5
      NT(A)=7
      LO(A)=0
      SPILL=SPILL-SADJ
      HI(A)=SPILL
      AMP(A)=1.0
      COST(A)=0
C***********************************************************************
C                 TOTAL FINAL RESERVOIR STORAGE
C***********************************************************************
      A=6
      NF(A)=6
      NT(A)=7
      LO(A)=0
      HI(A)=FSTORE*2.
      AMP(A)=1.
      COST(A)=0
      IYR=IYEAR-NSOLVE+1
      IF (ISOLVE.LT.NSOLVE) RETURN
      IF (IPRNT(IYR).EQ.1) GO TO 420
      RETURN
C***********************************************************************
C                 PRINT NETWORK MODEL
C***********************************************************************
420   PRINT 440, IYR
      PRINT 450
      PRINT 460
      DO 430 I=1,ARCS
      PRINT 470, I,NF(I),NT(I),LO(I),HI(I),COST(I),AMP(I)
430   CONTINUE
440   FORMAT (1H1//24H *****NETWORK MODEL*****/7X,9HFOR YEAR ,I2,//)
450   FORMAT (79H          ARC      START        END       LOWER      UPPER
     1     COST      AMPLIFICATION)
460   FORMAT (52H          NO.       NODE       NODE       BOUND      BOUND//)
470   FORMAT (3I10,6F11.2)
      RETURN
      END
```

```
      FUNCTION COSTF (I)
C*************************************************************************
C                    FUNCTION TO CALCULATE THE COSTS ON ARC I
C                    CONSIDERING UPPER AND LOWER BOUNDS
C*************************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1IN(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      IF (I.GT.NARC) GO TO 60
C*************************************************************************
C                    COMPUTE COST ON FORWARD ARC
C*************************************************************************
      IF (FLOW(I)-LOWER(I)) 20,40,40
20    CONTINUE
      COSTF=-BIG
      RETURN
30    CONTINUE
      COSTF=COST(I)
      RETURN
40    CONTINUE
      IF (UPPER(I)-FLOW(I)) 50,30,30
50    CONTINUE
      COSTF=BIG
      RETURN
C*************************************************************************
C                    COMPUTE COST ON MIRROR ARC
C*************************************************************************
60    CONTINUE
      K=I-NARC
      IF (FLOW(K)-LOWER(K)) 90,70,80
70    CONTINUE
      COSTF=-COST(K)/AMP(K)
      RETURN
80    CONTINUE
      IF (UPPER(K)-FLOW(K)) 100,70,70
90    CONTINUE
      COSTF=BIG/AMP(K)
      RETURN
100   CONTINUE
      COSTF=-BIG/AMP(K)
      RETURN
      END
```



```
      SUBROUTINE COSTX
C*************************************************************************
C                    THIS SUBROUTINE COMPUTES THE ECONOMIC COSTS
C                    OF THE CURRENT SYSTEM CONFIGURATION
C*************************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /JDATA/ RNAME(30,2),RCAP(30),RETSW(30),CRES(30),RMIN(30),NL
     1INK(30,8),FSTART(30),DLO(30)
      COMMON /LDATA/ CNAME(45,2),CLINK(2,45,3),CCAP(45),CPUMP(45),CMIN(4
     15),CLIFT(45),EL(45),LNODE(45,2)
      COMMON /CDATA/ PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC,POWFA
     1C(12)
      COMMON /RDATA/ JS(10),JBLT(30),LBLT(45)
      COMMON /APRNT/ APOW(50),AIMP(50),QM(45),ISHORT(50),ICOST(50)
      COMMON CRESC(50),CRESOM(50),CLINKC(50),CLINOM(50),CIMPT(50),CPOWR(
     150),CTOTAL(50),PVRES(50),PVROM(50),PVLINC(50),PVLIOM(50),PVIMP(50)
     2,PVPOWR(50),PVTOT(50),ANRESC(50),ANRESO(50),ANLINC(50),ANLINO(50),
     3ANIMP(50),ANPOWR(50),ANTOT(50),CPEN(50),PVPEN(50),ANPEN(50)
      INTEGER QM,AIMP,APOW
C*************************************************************************
C                    INITIALIZE SUMS AND COMPUTE INTEREST
C                    RATE FACTORS FOR PRESENT VALUE AND AVG.
C                    ANNUAL COST COMPONENTS
C*************************************************************************
      CRESC(1)=0.
      CRESOM(1)=0.
      CLINKC(1)=0.
      CLINOM(1)=0.
      CIMPT(1)=0.
      CPOWR(1)=0.
      PVRES(1)=0.
      PVROM(1)=0.
      PVLINC(1)=0.
      PVLIOM(1)=0.
      PVIMP(1)=0.
      PVPOWR(1)=0.
      ANRESC(1)=0.
      ANRESO(1)=0.
      ANLINC(1)=0.
      ANLINO(1)=0.
      ANIMP(1)=0.
      ANPOWR(1)=0.
      ANPEN(1)=0.
      PVPEN(1)=0.
      CPEN(1)=0.
      IPAY=-NPAY
      ANFAC=PERINT/(1.-(1.+PERINT)**IPAY)
C*************************************************************************
C                    SET UP DO LOOP FOR ANNUAL VALUES AND
C                    COMPUTE COST FACTORS WITH RESERVOIR
C                    AND CONDUIT CAPITAL COSTS INCURRED
C                    LAGR AND LAGC YEARS EARLIER RESPECTIVELY
C*************************************************************************
      DO 90 IYR=1,NYEAR
      J=IYR+1
      IF (IYR.EQ.1) GO TO 20
      IYR1=IYR-1
      PVFAC=1./(1.+PERINT)**IYR1
      GO TO 30
20    PVFAC=1.0
      CMPD=1.0
30    CONTINUE
C*************************************************************************
C                    COMPUTE RESERVOIR COST COMPONENTS
C*************************************************************************
      ANSUM=0.
      ANSUMO=0.0
      DO 50 K=1,NRES
      IF (JBLT(K).GT.IYR) GO TO 40
      ANSUMO=ANSUMO+CRES(K)*RESONM
40    CONTINUE
```

```
      IBL=JBLT(K)-LAGR
      IF(IBL.LT.1)IBL=1
      IF (IBL.NE.IYR) GO TO 50
      ANSUM=ANSUM+CRES(K)
50    CONTINUE
      CRESC(J)=CRESC(J-1)+ANSUM
      CRESOM(J)=CRESOM(J-1)+ANSUMO
      PVRES(J)=PVRES(J-1)+ANSUM*PVFAC
      PVROM(J)=PVROM(J-1)+ANSUMO*PVFAC
      ANRESC(J)=PVRES(J)*ANFAC
      ANRESO(J)=PVROM(J)*ANFAC
C***********************************************************************
C                   FIND MAXIMUM LINK FLOWS AND
C                   COMPUTE CONDUIT COST COMPONENTS
C                   NOTE*** ZERO CAPITAL COST FOR
C                   MAXIMUM LINK-FLOW.LT.1 CFS
C***********************************************************************
      ANSUM=0.
      ANSUMO=0.0
      DO 80 L=1,NL
      CONCAP=0.
      Q=QM(L)
      IF (Q.LT.1.) GO TO 60
      Q=Q*723.6
      CONCAP=CLINK(1,L,1)+CLINK(1,L,2)*Q+CLINK(1,L,3)*Q**2
60    CONTINUE
      IBL=LBLT(L)-LAGC
      IF(IBL.LT.1)IBL=1
      IF (IBL.NE.IYR) GO TO 70
      ANSUM=ANSUM+CONCAP
70    IF (LBLT(L).GT.IYR) GO TO 80
      ANSUMO=ANSUMO+CONCAP*CONONM
80    CONTINUE
      CLINKC(J)=CLINKC(J-1)+ANSUM
      CLINOM(J)=CLINOM(J-1)+ANSUMO
      PVLINC(J)=PVLINC(J-1)+ANSUM*PVFAC
      PVLIOM(J)=PVLIOM(J-1)+ANSUMO*PVFAC
      ANLINC(J)=PVLINC(J)*ANFAC
      ANLINO(J)=PVLIOM(J)*ANFAC
C***********************************************************************
C                   COMPUTE IMPORT WATER COST COMPONENTS
C***********************************************************************
      XIMP=AIMP(IYR)
      ANSUM=XIMP*CIMP
      CIMPT(J)=CIMPT(J-1)+ANSUM
      PVIMP(J)=PVIMP(J-1)+ANSUM*PVFAC
      ANIMP(J)=PVIMP(J)*ANFAC
C***********************************************************************
C                   COMPUTE PUMP-POWER COST COMPONENTS
C                   AND PENALTY COSTS
C***********************************************************************
      ANSUM=APOW(IYR)
      CPOWR(J)=CPOWR(J-1)+ANSUM
      PVPOWR(J)=PVPOWR(J-1)+ANSUM*PVFAC
      ANPOWR(J)=PVPOWR(J)*ANFAC
      CPEN(J)=CPEN(J-1)+ICOST(IYR)
      PVPEN(J)=PVPEN(J-1)+FLOAT(ICOST(IYR))*PVFAC
      ANPEN(J)=ANFAC*PVPEN(J)
C***********************************************************************
C                   SUM COST COMPONENTS TO GET CUMULATIVE
C                   ANNUAL TOTALS
C***********************************************************************
      CTOTAL(J)=CRESC(J)+CRESOM(J)+CLINKC(J)+CLINOM(J)+CIMPT(J)+CPOWR(J)
     1+CPEN(J)
      PVTOT(J)=PVRES(J)+PVROM(J)+PVLINC(J)+PVLIOM(J)+PVIMP(J)+PVPOWR(J)+
     1PVPEN(J)
      ANTOT(J)=ANRESC(J)+ANRESO(J)+ANLINC(J)+ANLINO(J)+ANIMP(J)+ANPOWR(J
     1)+ANPEN(J)
90    CONTINUE
C***********************************************************************
C                   END OF ANNUAL COST COMPUTATIONS
C***********************************************************************
      RETURN
```



```
      SUBROUTINE DESUB (I,II)
C***********************************************************************
C                   SUBROUTINE TO DELETE AN ARC FROM THE TRIPLE LABEL
C                   REPRESENTATION OF THE FLOW AUGMENTING TREE.
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
C***********************************************************************
C                   DELETES ARC I FROM THE LIST OF SUBSEQUENT ARCS
C                   TO NODE II.
C***********************************************************************
      JJ=JARC(I)
      IF (FARC(II).NE.I) GO TO 20
      FARC(II)=RARC(JJ)
      RETURN
20    CONTINUE
      MM=FARC(II)
30    CONTINUE
      MN=JARC(MM)
      IF (RARC(MN).NE.I) GO TO 40
      RARC(MN)=RARC(JJ)
      RETURN
40    CONTINUE
      MM=RARC(MN)
      GO TO 30
      END
```

```
      END
```



```
      SUBROUTINE FLCHG (II,FLONOW)
C*********************************************************************
C                    SUBROUTINE TO INCREASE THE FLOW IN AN ARC BY A
C                    GIVEN AMOUNT
C*********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
      IF (II.GT.NARC) GO TO 20
C*********************************************************************
C                    CHANGE FLOW IN A FORWARD ARC.
C*********************************************************************
      FLOW(II)=FLOW(II)+FLONOW
      IF(ABS(FLOW(II)-LOWER(II)).LT.EPS)FLOW(II)=LOWER(II)
      IF(ABS(FLOW(II)-UPPER(II)).LT.EPS)FLOW(II)=UPPER(II)
      CSTNOW=CSTNOW+FLONOW*COST(II)
      GO TO 30
C*********************************************************************
C                    CHANGE FLOW IN A MIRROR ARC.
C*********************************************************************
20    CONTINUE
      KK=II-NARC
      FLONOW=FLONOW/AMP(KK)
      FLOW(KK)=FLOW(KK)-FLONOW
      CSTNOW=CSTNOW-FLONOW*COST(KK)
30    CONTINUE
      RETURN
      END
```

```
      FUNCTION FLMXC (I,S)
C*********************************************************************
C                    FUNCTION TO DETERMINE MAXIMUM FLOW CHANGE IN AN ARC
C*********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL COSTF,AMPF
      II=IARC(I)
      JJ=JARC(I)
      IF (S) 20,20,70
C*********************************************************************
C                    FLOW IS TO BE DECREASED
C*********************************************************************
20    CONTINUE
      IF (I.GT.NARC) GO TO 40
      IF (FLOW(I).GT.UPPER(I)) GO TO 30
      FLMXC=FLOW(I)-LOWER(I)
      RETURN
30    CONTINUE
      FLMXC=FLOW(I)-UPPER(I)
      RETURN
40    CONTINUE
      K=I-NARC
      IF (FLOW(I).LT.LOWER(I)) GO TO 60
      FLMXC=(UPPER(K)-FLOW(K))*AMP(K)
      RETURN
60    CONTINUE
      FLMXC=(LOWER(K)-FLOW(K))*AMP(K)
      RETURN
C*********************************************************************
C                    FLOW IS TO BE INCREASED
C*********************************************************************
70    CONTINUE
      IF (I.GT.NARC) GO TO 100
      IF (FLOW(I).GE.UPPER(I)) GO TO 130
      IF (FLOW(I).LT.LOWER(I)) GO TO 90
      FLMXC=UPPER(I)-FLOW(I)
      RETURN
90    CONTINUE
      FLMXC=LOWER(I)-FLOW(I)
      RETURN
100   CONTINUE
      K=I-NARC
      IF (FLOW(K).GT.UPPER(K)) GO TO 120
      IF (FLOW(K).LE.LOWER(K)) GO TO 130
      FLMXC=(FLOW(K)-LOWER(K))*AMP(K)
      RETURN
120   CONTINUE
      FLMXC=(FLOW(K)-UPPER(K))*AMP(K)
      RETURN
130   CONTINUE
      FLMXC=0.0
      RETURN
      END
```

```
      SUBROUTINE FLOP (JJ,FLMAX,GN,IROOTL)
C**********************************************************************
C                   SUBROUTINE TO DETERMINE THE GAIN, MAXIMUM FLOW
C                   CHANGE, AND ROOT OF A FLOW GENERATING CYCLE
C                   IN THE FLOW AUGMENTING TREE.
C**********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
      FLMAX=9999999999.0
      GN=1
      IJ=JJ
20    CONTINUE
      IJK=BARC(IJ)
      GN=GN*AMPF(IJK)
      FLMXT=FLMXC(IJK,1.)*GN
      IF (FLMXT.GT.FLMAX) GO TO 30
      FLMAX=FLMXT
      IROOTL=JARC(IJK)
30    CONTINUE
      IF (IARC(IJK).EQ.JJ) GO TO 40
      IJ=IARC(IJK)
      GAN(IJ)=GAN(JJ)*GN
      GO TO 20
40    CONTINUE
      FLMAX=FLMAX*(1.-1./GN)
      RETURN
      END
```

```
      SUBROUTINE GAIN
C**********************************************************************
C                   PROGRAM TO SOLVE THE NETWORK WITH GAINS PROBLEM FOR
C                   A GIVEN QUANTITY OF OUTPUT FLOW.
C                   PROGRAM BY P. JENSEN AND GORA BHAUMIK, UNIVERSITY
C                   OF TEXAS, 1973.
C**********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      COMMON /PRT/ IPRNT(45)
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
C**********************************************************************
C                   INITIALIZE COUNTERS AND CONSTANTS
C**********************************************************************
      NARC=ARCS
      NDEG=0
      NLOP=0
      ITER=0
      FLONET=0.0
      TOTCST=0.0
      IPRINT=1
      IFS=0
      EPS=.001
      BIG=1.E3
      SOURCE=1
      SINK=7
      OUTFLO=99999.
C**********************************************************************
C                   INITIALIZE FLOWS AND CREATE MIRROR ARCS.
C**********************************************************************
      DO 20 I=1,NARC
      NN=NARC+I
      IARC(NN)=JARC(I)
      JARC(NN)=IARC(I)
20    CONTINUE
      NARCS=NARC*2
30    CONTINUE
C**********************************************************************
C                  EXECUTE ITERATIVE SOLUTION PROCEDURE
C**********************************************************************
      CALL SHORT (IENTER,ILEAV)
      IF (IENTER.EQ.0) GO TO 60
      CALL MAXFLO
      TOTCST=TOTCST+CSTNOW
      ITER=ITER+1
      IF (IPRINT) 50,50,40
40    PRINT 150, ITER,FLONET,TOTCST
      PRINT 160, ILEAV,IENTER,SICH
50    CONTINUE
      IF (ABS(FLONET-OUTFLO).LE.EPS) GO TO 60
      GO TO 30
60    CONTINUE
      IYR=IYEAR-NSOLVE+1
      IF (IPRNT(IYR).EQ.1) GO TO 70
      GO TO 90
```

```
C*****************************************************************************
C                    PRINT OPTIMAL SOLUTION FOR NETWORK PROBLEM
C*****************************************************************************
70     PRINT 100
       PRINT 110
       DO 80 I=1,NARC
       PRINT 120, I,IARC(I),JARC(I),LOWER(I),UPPER(I),COST(I),AMP(I),FLOW
      1(I)
80     CONTINUE
90     IF (IPRINT.EQ.-1) RETURN
       PRINT 130, TOTCST
       PRINT 140, ITER,NDEG,NLOP
       RETURN
100    FORMAT (1H1///,30H*****OPTIMAL FLOW PATTERN*****,///)
110    FORMAT (77H  ARC START  END      LOWER      UPPER       COST       GAI
      1N       FLOW              )
120    FORMAT (I5,2X,2I5,5F10.2)
130    FORMAT (///,21H *****TOTAL COST*****,F20.4)
140    FORMAT (22H NUMBER OF ITERATIONS ,I10/33H NUMBER OF DEGENERATE ITE
      1RATIONS ,I10/27H NUMBER OF LOOP ITERATIONS ,I10/)
150    FORMAT (11H ITERATION ,I5,5X,6H FLOW ,F10.2,5X,6H COST ,F20.2)
160    FORMAT (7H REMOVE,I5,5X,5HENTER,I5,5X,5HDELTA,F20.5)
       END
```

```
      SUBROUTINE INPUT
C*******************************************************************
C                    THIS SUBROUTINE READS ALL THE SYSTEM DATA,THE
C                    COST DATA, AND THE FIXED MONTHLY COEFFICIENTS.
C*******************************************************************
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /MDATA/ XX(12),ELF(12),XIMP,KSPILL
      COMMON /JDATA/ RNAME(30,2),RCAP(30),RETSW(30),CRES(30),RMIN(30),NL
     1INK(30),FSTART(30),DLO(30)
      COMMON /LDATA/ CNAME(45,2),CLINK(2,45,3),CCAP(45),CPUMP(45),CMIN(4
     15),CLIFT(45),EL(45),LNODE(45,2)
      COMMON /SHORTS/ BND(30),CST(12,30),ACO(30,2),TFAC(13,30),TCST(30),
     1IOPT(30),ALP(30)
      COMMON /PRT/ IPRNT(45)
      COMMON /RDATA/ JS(10),JBLT(30),LBLT(45)
      COMMON /APRNT/ APOW(50),AIMP(50),QM(45),ISHORT(50),ICOST(50)
      COMMON /CDATA/ PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC,POWFA
     1C(12)
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LOWER(1800),ARCS
      INTEGER SCAP,ARCS
      REAL LOWER
C*******************************************************************
C                    READ AND WRITE
C                    HEADING
C*******************************************************************
      INCRD=5
      IPAGE=6
      KOUT=6
      KIN=5
      WRITE (KOUT,20)
20    FORMAT (1H1/125(1H*))
      WRITE (KOUT,30)
30    FORMAT (42X,29HTEXAS WATER DEVELOPMENT BOARD,/44X,25H    ALLOCATIO
     1N MODEL     ,/125(1H*)////)
C*******************************************************************
C                    READ AND WRITE SYSTEM DATA
C                    - INPUT FILE A
C*******************************************************************
      READ (5,40) NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IDATE,IMP
40    FORMAT (10X,14I5)
C*******************************************************************
C                    READ AND WRITE NODE DATA
C                    - INPUT FILE B
C*******************************************************************
      READ (5,50) (J,(RNAME(J,K),K=1,2),RCAP(J),RMIN(J),CRES(J),RETSW(J)
     1,NLINK(J),DLO(J),BND(J),ACO(J,1),ACO(J,2),N=1,NJ)
50    FORMAT (I2,2A4,3F10.0,F5.2,I5,F5.2,F10.0,5X,2F5.4)
C*******************************************************************
C                    READ AND WRITE LINK DATA (CANAL OR RIVER)
C                    - INPUT FILE C
C*******************************************************************
      READ (5,60) (L,(CNAME(L,K),K=1,2),CCAP(L),CMIN(L),CLIFT(L),EL(L),(
     1LNODE(L,K),K=1,2),M=1,NL)
60    FORMAT (I2,2A4,3F10.0,F10.3,20X,2I5)
C*******************************************************************
C                    READ AND WRITE LINK COST DATA
C                    - INPUT FILE D
C*******************************************************************
      READ (5,70) (L,((CLINK(I,L,K),K=1,3),I=1,2),M=1,NL)
70    FORMAT (I2,8X,6E10.4)
C*******************************************************************
C                    READ AND WRITE ECONOMIC DATA
C                    - INPUT FILE E
C*******************************************************************
      READ (5,80) PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC
80    FORMAT (10X,F10.4,I10,4F10.4,2I5)
C*******************************************************************
C                    READ AND WRITE POWER FACTOR COEFFS
C                    - INPUT FILE F
C*******************************************************************
      READ (5,90) (POWFAC(I),I=1,NSEAS)
90    FORMAT (10X,12F5.3)
```

```
C***********************************************************************
C                    READ AND WRITE SEASONAL IMPORT COEFFS
C                    - INPUT FILE G
C***********************************************************************
      READ (5,100) XIMP,(XX(I),I=1,NSEAS)
100   FORMAT (10X,F5.0,12F5.3)
C***********************************************************************
C                    READ AND WRITE SEASONAL CHANNEL LOSS ADJUSTMENT
C                    FACTORS
C                    - INPUT FILE H
C***********************************************************************
      READ (5,90) (ELF(I),I=1,NSEAS)
C***********************************************************************
C                    READ ANNUAL OUTPUT CONTROL OPTIONS
C                    - INPUT FILE I
C***********************************************************************
      READ (5,110) IPRNT
110   FORMAT (14X,45I1)
C***********************************************************************
C                    READ TIME DIMENSIONS OF PROBLEM
C                    - INPUT FILE J
C***********************************************************************
      READ (KIN,120) NUMYRS,NSOLVE
120   FORMAT (20X,I10,10X,I10)
      NYEAR=NUMYRS
      NREAD=1
C***********************************************************************
C                    READ SPILL COST AND MAXIMUM SPILL RATES
C                    - INPUT FILE K
C***********************************************************************
      READ (KIN,130) SCAP,KSPILL,NSPNDS
130   FORMAT (10X,3I10)
      NS=NSPNDS
      IF (NSPNDS.LE.10) GO TO 150
      ERROR=1.0
      WRITE (IPAGE,140)
140   FORMAT (1X,48HERROR---MORE SPILL NODES THAN 10 ALLOWED---ERROR)
150   CONTINUE
C***********************************************************************
C                    READ NODE NUMBERS OF NODES THAT CAN SPILL
C                    - INPUT FILE L
C***********************************************************************
      READ (KIN,160) (JS(I),I=1,NSPNDS)
160   FORMAT (10X,5(8X,I2))
C***********************************************************************
C                    READ THE YEARS THAT RESERVOIRS WERE BUILT
C                    - INPUT FILE M
C***********************************************************************
      NRS4=NRES
170   NLOOP=MIN0(4,NRS4)
      READ (KIN,190) (I,JBLT(I),J=1,NLOOP)
      NRS4=NRS4-4
      IF (NRS4) 180,180,170
180   CONTINUE
190   FORMAT (10X,4(8X,I2,I5))
C***********************************************************************
C                    READ THE YEARS THAT CANALS WERE BUILT
C                    - INPUT FILE N
C***********************************************************************
      NRS4=NL
200   NLOOP=MIN0(4,NRS4)
      READ (KIN,190) (I,LBLT(I),J=1,NLOOP)
      NRS4=NRS4-4
      IF (NRS4) 210,210,200
210   CONTINUE
C***********************************************************************
C                    READ THE INITIAL RESERVOIR STORAGE CONTENTS
C                    AS A PERCENTAGE OF MAXIMUM STORAGE VOLUME
C                    - INPUT FILE O
C***********************************************************************
      NRS4=NRES
220   NLOOP=MIN0(4,NRS4)
      READ (KIN,240) (I,FSTART(I),J=1,NLOOP)
      NRS4=NRS4-4
      IF (NRS4) 230,230,220
230   CONTINUE
240   FORMAT (10X,4(8X,I2,F5.0))
C***********************************************************************
C                    READ RESERVOIR OPERATING RULES AND OPTIONS
C                    - INPUT FILE P
C***********************************************************************
      READ (KIN,250) (J,IOPT(J),TCST(J),(TFAC(I,J),I=1,13),ALP(J),K=1,NR
     1ES)
250   FORMAT (13X,I2,3X,I2,F5.0,13F4.2,F3.2)
C***********************************************************************
C                    READ SHORTAGE COSTS
C                    - INPUT FILE Q
C***********************************************************************
      NCRDS=NSEAS/4
      IF(4*NCRDS.LT.NSEAS)NCRDS=NCRDS+1
      DO 270 K=1,NJ
      NST=1
      NSTP=4
      DO 270 L=1,NCRDS
      READ (KIN,260) J,(I,CST(I,J),M=NST,NSTP)
      NST=NST+4
      NSTP=NSTP+4
260   FORMAT (18X,I2,4(I5,F8.2,2X))
270   CONTINUE
C***********************************************************************
C                    PRINT THE DETAILS OF HOW THE
C                    PROBLEM IS BEING APPROACHED
C***********************************************************************
      WRITE (6,280)
      WRITE (6,290) NJ,NRES,NJUNC,IMP,NS,(JS(K),K=1,NS)
      WRITE (6,300) NL,NR,NC
      WRITE (6,310) NUMYRS,NREAD,NSEAS,NSOLVE,NYEAR
280   FORMAT (37X,38H DETAILS OF THE PROBLEM ARE AS FOLLOWS)
290   FORMAT (1H0,37X,13H 1) THERE ARE,I3,20H JUNCTIONS, OF THESE//37X,1
     16H    A) THE FIRST,I3,15H ARE RESERVOIRS/37X,16H    B) THE LAST ,I
     23,19H ARE LINK JUNCTIONS/37X,36H    C) IMPORTED WATER ENTERS AT NO
     3DE,I3/37X,42H    D) SPILLS FROM THE SYSTEM CAN OCCUR AT,I2/37X,29H
     4        NODES, THESE ARE NODES/(50X,I3))
300   FORMAT (1H0,37X,13H 2) THERE ARE,I3,15H LINKS,OF THESE//37X,16H
     1 A) THE FIRST,I3,18H ARE RIVER REACHES/37X,16H    B) THE LAST ,I3,
     216H ARE PUMP-CANALS)
310   FORMAT (1H0,37X,13H 3) THERE ARE,I3,21H YEARS IN THE PROBLEM/37X,2
     14H    A) THE FIRST YEAR IS,I3/37X,20H    B) EACH YEAR HAS,I3,8H SE
     2ASONS/37X,38H    C) EACH YEARLY SOLUTION REQUIRES A,I2/37X,19H
     3   YEAR NETWORK,/37X,23H    D) THE LAST YEAR IS,I3)
       WRITE (KOUT,20)
       WRITE (KOUT,320)
320    FORMAT (34X,39HNODE DATA, EITHER RESERVOIR OR JUNCTION,/125(1H*)//
     1/4X,16HNODE  RES OR JNC,5X,6HRESEVR,4X,6HRESEVR,3X,6HRESEVR,4X,6HR
     2ETURN,4X,6HRETURN,4X,5HBRKPT,4X,5HLOWER,4X,5HUPPER,3X,10HSIMULATIO
     3N)
       WRITE (KOUT,330)
330    FORMAT (5X,3HNO.,5X,4HNAME,9X,3HMAX,7X,3HMIN,7X4HCOST
     1              6X,4HFLOW,6X,4HFLOW,4X,7HSTORAGE,2X,7HSTORAGE,2X,7
     2HSTORAGE,2X,     8HYEAR RES/25X,6H1000AF,4X,6H1000AF,3X,5H1000$,6X,
     34HNODE,6X,       4H S/W,5X,6H1000AF,3X,5HSLOPE,4X,5HSLOPE ,5X,5HBUI
     4LT/)
       DO 340 J=1,NJ
       WRITE (KOUT,350) J,(RNAME(J,K),K=1,2),RCAP(J),RMIN(J),CRES(J),NLIN
     1K(J),RETSW(J),BND(J),ACO(J,1),ACO(J,2),JBLT(J)
340    CONTINUE
350    FORMAT (3X,I5,3X,2A4,2X,F9.1,1X,F9.1,1X,F9.1,4X,I5,6X,F4.2,2X,F9.1
     1,2F9.4,I10)
       WRITE (KOUT,20)
       WRITE (KOUT,360)
360    FORMAT (43X,24HLINK DATA CANAL OR RIVER,/125(1H*)///,10X,15HLINK
     1LINK NAME,3X,77HORIGIN  TERMINAL   MAXIMUM   MINIMUM   TOTAL DYNAM
     2IC   SEASONAL     SIMULATION)
       WRITE (KOUT,370)
370    FORMAT (11X,3HNO.,15X,4HNODE,5X,4HNODE,5X,4HFLOW,7X,4HFLOW,5X,12HP
```

20X,6HCOEFF.,8X,5HBUILT)

```
      SUBROUTINE OUTPUT
C*********************************************************************
C                    THIS SUBROUTINE WRITES THE RESULTS FROM
C                    SUBROUTINE GAIN IF THE NETWORK PROBLEM IS
C                    INFEASIBLE
C*********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LO(1800),ARCS
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      DATA DASH/2H**/
      DATA BLANK/2H  /
      REAL LO,LOCF
      INTEGER SCAP,A,ARCS,SOURCE,SINK,AB
C*********************************************************************
C                    DETERMINE IF THE FLOW IN ANY ARC VIOLATES THE
C                    UPPER OR LOWER BOUNDS ON THE FLOW IN THAT ARC
C*********************************************************************
      KOUT=6
      TIMAX=0.
      DO 30 I=1,ARCS
      F=FLOW(I)
      IF (F-HI(I)-EPS) 20,20,100
20    IF (LO(I)-F-EPS) 30,30,100
30    CONTINUE
C*********************************************************************
C                    DETERMINE IF CONSTRAINTS ON RESERVOIR STORAGE
C                    SEGMENTATION HAVE BEEN VIOLATED
C*********************************************************************
      DO 80 I=1,NSEAS
      AIMIN=NRES+6+(I-1)*(2*NJ+1+2*NRES+NL+NS)
      ARMIN=AIMIN+NJ+1
      DO 80 J=1,NRES
      A=ARMIN+J
      AB=A+NRES
      IF (FLOW(A)-HI(A)+EPS) 40,70,70
40    IF (FLOW(AB)-EPS) 60,60,50
50    WRITE (KOUT,310) J,I,FLOW(A),FLOW(AB),HI(A)
      TIMAX=1.
      IF ((FLOW(A)+FLOW(AB)).GT.HI(A)) GO TO 70
60    HI(AB)=0.
      LO(AB)=0.
      GO TO 80
70    LO(A)=HI(A)
80    CONTINUE
      IF (TIMAX.EQ.1.) WRITE (KOUT,90)
90    FORMAT (1H1///125(1H*)//50X,27HRESTART PROCEDURE INITIATED//40X,55
     1HSTORAGE BOUNDS HAVE BEEN REDEFINED AND NETWORK SOLVED   ,//125(1H*
     2),1H1)
      RETURN
C*********************************************************************
C                    WRITE ARC DATA
C*********************************************************************
100   CONTINUE
      SUBS=12./NSEAS
      WRITE (KOUT,110)
110   FORMAT (1H1//8X,67HASTERISKS IN OUTPUT BELOW PINPOINT FLOWS WHERE
     1FLOWS=MAX CAPACITY  )
      WRITE (6,120)
      WRITE (6,130) NODES,ARCS
      IDYR=1
      IDSEA=1
      IPAGE=1
      WRITE (KOUT,140)
      WRITE (KOUT,150)
120   FORMAT (53H1* * *  S O L U T I O N    I N F E A S I B L E  * * *)
130   FORMAT (18H1NUMBER OF NODES =,I5//18H NUMBER OF ARCS  =,I5)
140   FORMAT (7X,3HARC,6X,4HFROM,6X,4H TO ,6X,4HFLOW,6X,4HFLOW,6X,4HCOST
     1,5X,14H---MAX FLOW---,3X,3X,14H---MIN FLOW---,3X)
150   FORMAT (7X,3HNO ,6X,4HNODE,6X,4HNODE,4X,6H1000AF,7X,3HCFS,3X,8H
     1 $/AF,4X,6H1000AF,7X,3HCFS,4X,6H1000AF,7X,3HCFS/)
160   FORMAT (1H1)
      WRITE (KOUT,170)
170   FORMAT (/40X,16HNET BALANCE ARCS/)
      DO 300 A=1,ARCS
      NDUM=(2*NJ+1+2*NRES+NL+NS)*((IDYR-1)*NSEAS+(IDSEA-1))+(6+NRES)
      IF (A.NE.IPAGE*30) GO TO 180
      WRITE (6,160)
      WRITE (KOUT,140)
      WRITE (KOUT,150)
      IPAGE=IPAGE+1
180   CONTINUE
      IF (A.EQ.7) WRITE (6,190)
190   FORMAT (/,40X,20HINITIAL STORAGE ARCS/)
      IF (A.NE.(NDUM+1)) GO TO 210
      WRITE (6,200) IDYR,IDSEA
200   FORMAT (/,40X,20HDEMAND ARCS FOR YEAR,I3,8H ,SEASON,I3/)
210   CONTINUE
      IF (A.EQ.(NDUM+(NJ+1))) WRITE (6,220) IDYR,IDSEA
220   FORMAT (/,40X,19HIMPORT ARC FOR YEAR,I3,8H ,SEASON,I3/)
      IF (A.EQ.(NDUM+NJ+2)) WRITE (6,230) IDYR,IDSEA
230   FORMAT (/,40X,27HRESERVOIR STORAGE ARCS,YEAR,I3,8H ,SEASON,I3/)
      IF (A.EQ.(NDUM+NJ+2+2*NRES)) WRITE (6,240) IDYR,IDSEA
240   FORMAT (/,40X,18HLINK ARCS FOR YEAR,I3,8H ,SEASON,I3/)
      IF (A.EQ.(NDUM+NJ+2+2*NRES+NL)) WRITE (6,250) IDYR,IDSEA
250   FORMAT (/,40X,19HSPILL ARCS FOR YEAR,I3,8H ,SEASON,I3)
      IF (A.EQ.(NDUM+NJ+2+2*NRES+NL+NS)) WRITE (6,260) IDYR,IDSEA
260   FORMAT (/40X,21HINFLOW ARCS FOR YEAR ,I3,9H ,SEASON ,I3/)
      FLOWC=FLOW(A)*16.6/SUBS
      LOCF=LO(A)*16.6/SUBS
      HICF=HI(A)*16.6/SUBS
      IF (FLOWC.LT.LOCF.OR.FLOWC.GT.HICF) PRINT 270, A
270   FORMAT (////,17H INFEASIBLE ARC -,I4////)
      DASHO=BLANK
      IF((FLOWC.EQ.HICF).AND.(FLOWC.NE.0.))DASHO=DASH
280   FORMAT (3I10,2F10.2,A2,F9.3,4F10.2)
      WRITE (KOUT,280) A,NF(A),NT(A),FLOW(A),FLOWC,DASHO,COST(A),HI(A),H
     1ICF,LO(A),LOCF
      IF (A.NE.(NDUM+NJ+1+2*NRES+NL+NS+NJ)) GO TO 290
      IDSEA=IDSEA+1
      IF (IDSEA.LE.NSEAS) GO TO 290
      IDSEA=1
      IDYR=IDYR+1
290   CONTINUE
300   CONTINUE
      STOP
310   FORMAT (1H1/125(1H*)//50X,19HCOMPUTATIONAL ERROR//1X,52HTHE LOWER
     1SEGMENT OF CARRY-OVER STORAGE IN RESERVOIR,I3,1X,9HIN SEASON,I3,62
     2H WAS NOT FULL BEFORE THE UPPER SEGMENT OF STORAGE WAS UTILIZED//5
     3X,31HSTORAGE IN THE LOWER SEGMENT = ,F6.1,4X,31HSTORAGE IN THE UPP
     4ER SEGMENT = ,F6.1,4X,38HMAXIMUM STORAGE IN THE LOWER SEGMENT =,F6
     5.1///20X,42HRESERVOIR OPERATING RULES MUST BE ADJUSTED/20X,94HEITH
     6ER OPTION 2  MAY BE USED OR THE UNIT STORAGE BENEFIT  MAY BE INCRE
     7ASED IF OPTION 1 IS USED//125(1H*)/////)
      END
```

```
      SUBROUTINE MAXFLO
C***********************************************************************
C                     SUBROUTINE TO CALCULATE THE MAXIMUM FLOW INCREASE
C                     INTO THE SINK. ARC TO LEAVE THE TREE IS ALSO
C                     DETERMINED. THE FLOW IS CHANGED IN THE AUGMENTING
C                     PATH.
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
C***********************************************************************
C                     FIND OUT IF THERE IS A LOOP. IF SO JJ IS THE
C                     JUNCTION OF THE LOOP.
C***********************************************************************
      DO 20 I=1,NODES
      ICHK(I)=0
20    CONTINUE
      JJ=SOURCE
      I=SINK
30    CONTINUE
      ICHK(I)=1
      IF (I.EQ.SOURCE) GO TO 50
      II=BARC(I)
      IF (II.EQ.0) GO TO 140
      I=IARC(II)
      IF (ICHK(I).EQ.1) GO TO 40
      GO TO 30
40    CONTINUE
      JJ=I
      NLOP=NLOP+1
C***********************************************************************
C                     FIND MAXIMUM FLOW CHANGE POSSIBLE.
C***********************************************************************
50    CONTINUE
      FLMX=999999.
      GN=1
      I=SINK
      GAN(I)=1.
60    CONTINUE
      IF (I.EQ.JJ) GO TO 70
      KK=BARC(I)
      GN=GN*AMPF(KK)
      FLMXT=FLMXC(KK,1.)*GN
      I=IARC(KK)
      GAN(I)=GN
      IF (FLMXT.GT.FLMX) GO TO 60
      FLMX=FLMXT
      IROOT=JARC(KK)
      GO TO 60
70    CONTINUE
      IF (JJ.EQ.SOURCE) GO TO 80
      CALL FLOP (JJ,FLMAX,GLOOP,IROOTL)
      FLMXT=FLMAX*GN
      IF (FLMXT.GT.FLMX) GO TO 80
      FLMX=FLMXT
      IROOT=IROOTL
80    CONTINUE
C***********************************************************************
C                     INCREASE TOTAL FLOW BY THE MAXIMUM FLOW CHANGE.
C***********************************************************************
      FLO=OUTFLO-FLONET
      IF(FLO.GT.FLMX)FLO=FLMX
      IF(FLO.LT.1.E-4)NDEG=NDEG+1
      CSTNOW=0.0
      IF (FLO.LT.1.E-4) GO TO 130
      FLONET=FLONET+FLO
C***********************************************************************
C                     CALCULATE FLOW CHANGE ON EACH ARC.
C***********************************************************************
      I=SINK
90    CONTINUE
      IF (I.EQ.JJ) GO TO 100
      II=BARC(I)
      IF (II.EQ.0) GO TO 140
      I=IARC(II)
      FLONOW=FLO/GAN(I)
      CALL FLCHG (II,FLONOW)
      GO TO 90
100   CONTINUE
      IF (JJ.EQ.SOURCE) GO TO 130
      FLOOP=FLO/(GAN(I)*(1.-(1./GLOOP)))
      I=JJ
      FLGA=FLOOP*GAN(JJ)
110   CONTINUE
      II=BARC(I)
      I=IARC(II)
      FLONOW=FLGA/GAN(I)
      IF (I.NE.JJ) GO TO 120
      J=JARC(II)
      FLONOW=FLGA/(GAN(J)*AMPF(II))
120   CONTINUE
      CALL FLCHG (II,FLONOW)
      IF (I.EQ.JJ) GO TO 130
      GO TO 110
130   CONTINUE
      RETURN
140   CONTINUE
      PRINT 150, FLONET,TOTCST
      CALL EXIT
150   FORMAT (///,' PROBLEM IS INFEASIBLE. THE MAXIMUM FLOW IS  ',F10.2,
     1'   AT A TOTAL COST OF  ',F10.2)
      END
```

```
      DO 380 L=1,NL
      WRITE (KOUT,390) L,(CNAME(L,K),K=1,2),(LNODE(L,K),K=1,2),CCAP(L),C
     1MIN(L),CLIFT(L),EL(L),LBLT(J)
380   CONTINUE
390   FORMAT (10X,I3,3X,2A4,4X,I5,4X,I5,4X,F8.1,2X,F8.1,5X,F11.1,3X,F8.3
     1,I10)
      WRITE (6,20)
      WRITE (6,400)
400   FORMAT (50X,17HCANAL COST COEFF./125(1H*)//42X,5HDITCH,32X,4HPUMP/
     133X,2HC1,9X,2HC2,9X,2HC3,13X,2HC1,9X,2HC2,11X,2HC3/)
      DO 410 L=1,NL
410   WRITE (6,420) L,((CLINK(I,L,K),K=1,3),I=1,2)
420   FORMAT (22X,I5,2X,3(G10.4,1X),4X,3(G10.4,1X))
      WRITE (KOUT,20)
      WRITE (KOUT,430) PERINT,NPAY,POWR,RESONM
430   FORMAT (45X,18HCONSTANT COST DATA,/125(1H*)///15X,13HINTEREST RATE
     1,10X,1H=,F10.4,15X,23HREPAYMENT PERIOD YRS  =,5X,I5,1X,3HYRS,//15X
     2,10HPOWER COST,13X,1H=,F10.4,8H $/KW-HR,7X,23HO+M ANN DEC PCT RESV
     3R =,F10.4,/)
      WRITE (KOUT,440) CONONM,CIMP,LAGR,LAGC
440   FORMAT (15X,24HO+M ANN DEC PCT CANAL  =,F10.4,15X,23HCOST OF IMPOR
     1T WATER  =,F10.4,1X,4H$/AF,//15X,24HRESVR FINANCE LAG TIME =,5X,I5
     2,1X,3HYRS,11X,23HCANAL FINANCE LAG TIME=,5X,I5,1X,3HYRS)
      WRITE (KOUT,450)
      WRITE (KOUT,450)
450   FORMAT (///)
      WRITE (KOUT,460) (J,J=1,12)
460   FORMAT (48X,12HPOWER FACTOR,//51X,7HSEASONS/12X,12I7/)
      WRITE (KOUT,470) (POWFAC(I),I=1,NSEAS)
470   FORMAT (13X,12F7.4)
      WRITE (6,480) (J,J=1,12),(XX(I),I=1,NSEAS)
480   FORMAT (///,48X,13HIMPORT COEFFS//51X,7HSEASONS/12X,12I7//13X,12F7
     1.4)
      WRITE (6,490) (J,J=1,12),(ELF(I),I=1,NSEAS)
490   FORMAT (///48X,11HLOSS COEFFS//51X,7HSEASONS/12X,12I7//13X,12F7.4)
      WRITE (KOUT,20)
      WRITE (KOUT,500) (J,J=1,12)
500   FORMAT (48X,18HUNIT SHORTAGE COST/52X,9H($/AC-FT)/125(1H*)///51X,7
     1HSEASONS/12X,12I7/5X,4HNODE/6X,3HNO./)
      DO 510 J=1,NJ
      WRITE (KOUT,520) J,(CST(I,J),I=1,NSEAS)
510   CONTINUE
520   FORMAT (6X,I2,5X,12F7.0)
      WRITE (KOUT,20)
      WRITE (KOUT,530)
530   FORMAT (42X,24HRESERVOIR START CONTENTS,/125(1H*)///29X,4HNODE,16X
     1,10HRES OR JNC,10X,16HPCT FULL-DECIMAL/30X,4HNO. ,18X,4HNAME,16X,9
     2HSTRT STOR/)
      WRITE (KOUT,540) (J,(RNAME(J,K),K=1,2),FSTART(J),J=1,NRES)
540   FORMAT (29X,I3,17X,2A4,10X,F10.4)
      WRITE (KOUT,20)
      WRITE (KOUT,550) (J,J=1,13)
550   FORMAT (44X,28HRESERVOIR OPERATING POLICIES/125(1H*)///10X,4HRES.,
     121H  OPERATING   STORAGE,11X,68HEND-OF-YEAR TARGET STORAGE LEVELS
     2BY YEAR (FRACTION OF MAX. STORAGE),3X,5HALPHA/11X,24HNO.   OPTION
     3    BENEFIT,3X,13I6/)
      DO 560 J=1,NRES
      WRITE (KOUT,570) J,IOPT(J),TCST(J),(TFAC(I,J),I=1,13),ALP(J)
560   CONTINUE
570   FORMAT (10X,I3,I8,5X,F9.1,3X,13F6.2,5X,F3.2)
      IF (ERROR) 600,600,580
580   WRITE (IPAGE,590)
590   FORMAT (10X,45HERRORS ENCOUNTERED IN DATA DECK   JOB STOPPED//)
      STOP
600   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LOOP (I,JJ)
C**********************************************************************
C                   SUBROUTINE TO DETERMINE IF THE FLOW AUGMENTING TREE
C                   INCLUDES A  FLOW GENERATING CYCLE.  IF SO NODE
C                   POTENTIALS ARE ADJUSTED ACCORDINGLY. USED ONLY IN
C                   THE FIRST ITERATION.
C**********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER ARCS
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
      COMMON ICHT(500)
C**********************************************************************
C                   DETERMINE IF POINTERS INDICATE A LOOP.
C**********************************************************************
      DO 20 K=1,NODES
      ICHT(K)=0
20    CONTINUE
      ICHT(JJ)=1
      IJ=JJ
30    CONTINUE
      IJK=BARC(IJ)
      IF (IJK.EQ.0) RETURN
      IA=IARC(IJK)
      IF (IA.EQ.JJ) GO TO 40
      IF (ICHT(IA).EQ.1) RETURN
      ICHT(IA)=1
      IJ=IA
      GO TO 30
40    CONTINUE
      TEMP=0
      GN=1
      IJ=JJ
C**********************************************************************
C                   CALCULATE THE COST TO OBTAIN ONE UNIT OF FLOW INTO JJ
C**********************************************************************
50    CONTINUE
      IJK=BARC(IJ)
      GN=GN*AMPF(IJK)
      TEMP=TEMP+COSTF(IJK)/GN
      IA=IARC(IJK)
      IF (IA.EQ.JJ) GO TO 60
      IJ=IA
      GO TO 50
60    CONTINUE
C**********************************************************************
C                   CALCULATE COST TO OBTAIN ONE UNIT OF FLOW OUT OF
C                   LOOP AT JJ
C**********************************************************************
      V(JJ)=TEMP/(1.-1./GN)
70    CONTINUE
      IJ=IA
      IJK=BARC(IJ)
      IA=IARC(IJK)
      IF (IA.EQ.JJ) RETURN
      V(IA)=V(IJ)*AMPF(IJK)-COSTF(IJK)
      GO TO 70
      END
```

```
      SUBROUTINE PREPAR
C**********************************************************************
C                  THIS SUBROUTINE TAKES THE RESULTS FROM GAIN AND
C                  PREPARES THEM FOR PRINTING IN SUBROUTINE APRINT.
C**********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LOWER(1800),ARCS
      COMMON /APRNT/ APOW(50),AIMP(50),QM(45),ISHORT(50),ICOST(50)
      COMMON /MISC/ AMIN,ADELT,NDELT
      COMMON /CDATA/ PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC,POWFA
     1C(12)
      COMMON /XDATA/ PI(500),BARC(500),RARC(500),FARC(500),DISSET(500),G
     1AN(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      COMMON LQ(13,45),LP(13,45),LC(12,45),RS(13,30),RC(12,30),CC(12,30)
     1,JQ(13,10),IDEF(13,30),IMPC(50),ITOT(50)
      INTEGER A,AL,AIMP,APOW,AMIN,ALMIN,ARMIN,ASMIN,QM,RS,ARCS,AB,SOURCE
     1,SINK,BARC,RARC,FARC,DISSET
      REAL LC,LOWER
C**********************************************************************
C                  INITIALIZE VARIABLES
C**********************************************************************
      IYR=IYEAR-NSOLVE+1
      SUBS=12./NSEAS
      ROFF=.499
      KOUT=6
      IL=NL+1
      M=NSEAS+1
      APOW(IYR)=0
      AIMP(IYR)=0
      ISHORT(IYR)=0
      ICOST(IYR)=0
      IMPC(IYR)=0
      DO 20 L=1,NL
      LQ(M,L)=0
      LP(M,L)=0
      IF(IYR.EQ.1)QM(L)=0
20    CONTINUE
      DO 30 J=1,NRES
      RS(M,J)=0
30    CONTINUE
      DO 40 J=1,NS
      JQ(M,J)=0
40    CONTINUE
      DO 50 J=1,NJ
50    IDEF(M,J)=0
      DO 100 I=1,NSEAS
      AIMIN=AMIN+(I-1)*(NJ+1+2*NRES+NL+NS+NJ)
      ARMIN=AIMIN+NJ+1
      ALMIN=ARMIN+2*NRES
      ASMIN=ALMIN+NL
      AL=ARMIN
C**********************************************************************
C                  PREPARE LINK FLOWS,POWER COSTS,
C                  AND DUAL VALUES FOR PRINTING.
C**********************************************************************
      DO 60 L=1,NL
      A=ALMIN+L
      IA=NF(A)
      JA=NT(A)
C**********************************************************************
C                  16.6 CONVERTS TAF/MON TO CFS
C**********************************************************************
      LQ(I,L)=FLOAT(FLOW(A))*16.6/SUBS+ROFF
      LQ(M,L)=LQ(M,L)+LQ(I,L)
      IF(LQ(I,L).GT.QM(L))QM(L)=LQ(I,L)
      LP(I,L)=FLOW(A)*COST(A)
      LP(M,L)=LP(M,L)+LP(I,L)
C**********************************************************************
C                  16.6 CONVERTS $1000/TAF/MON TO
C                                  $1000/CFS
C**********************************************************************
      LC(I,L)=COST(A)+PI(IA)-PI(JA)
      LC(I,L)=LC(I,L)/16.6*SUBS
      APOW(IYR)=APOW(IYR)+FLOW(A)*COST(A)
      IF(I.EQ.NSEAS)LQ(M,L)=LQ(M,L)/NSEAS
      IF(I.EQ.NSEAS)LP(M,L)=LP(M,L)/NSEAS
60    CONTINUE
      LQ(I,IL)=FLOW(AL)
      AIMP(IYR)=AIMP(IYR)+FLOW(AL)
      IMPC(IYR)=IMPC(IYR)+(FLOW(AL)*CIMP)
C**********************************************************************
C                  PREPARE RESERVOIR STORAGE LEVELS,
C                  AND DUAL VALUES FOR PRINTING.
C**********************************************************************
      DO 70 J=1,NRES
      A=ARMIN+J
      AB=A+NRES
      IA=NF(A)
      JA=NT(A)
      RS(I,J)=FLOW(A)*AMP(A)+FLOW(AB)*AMP(AB)+ROFF
      RS(M,J)=RS(M,J)+RS(I,J)
      RC(I,J)=COST(AB)+PI(IA)-PI(JA)
      IF(I.EQ.NSEAS)RS(M,J)=RS(M,J)/NSEAS
70    CONTINUE
C**********************************************************************
C                  PREPARE DUAL VALUES OF CONTINUITY
C                  CONSTRAINTS FOR PRINTING.   THESE
C                  REPRESENT THE VALUE OF ONE MORE
C                  ACRE-FOOT OF WATER DELIVERED TO
C                  THE NODE.
C**********************************************************************
      DO 80 J=1,NJ
      N=7+(I-1)*NJ+J
      CC(I,J)=PI(N)
      A=AIMIN+J
      ISHORT(IYR)=ISHORT(IYR)+HI(A)-FLOW(A)
      ICOST(IYR)=ICOST(IYR)-(COST(A)*(HI(A)-FLOW(A)))
      IDEF(I,J)=HI(A)-FLOW(A)
      IDEF(M,J)=IDEF(M,J)+IDEF(I,J)
80    CONTINUE
C**********************************************************************
C                  PREPARE THE SPILLS THAT LEAVE THE
C                  SYSTEM FOR PRINTING.
C**********************************************************************
      DO 90 J=1,NS
      A=ASMIN+J
      JQ(I,J)=FLOW(A)
      JQ(M,J)=JQ(M,J)+JQ(I,J)
      IF(I.EQ.NSEAS)JQ(M,J)=JQ(M,J)/NSEAS
90    CONTINUE
100   CONTINUE
      NT2=51
      NZ=0
      ITOT(IYR)=APOW(IYR)+ICOST(IYR)+IMPC(IYR)
      RETURN
      END
```

```
      SUBROUTINE SHIFT
C***********************************************************************
C                    THIS SUBROUTINE SHIFTS ALL ARC DATA AND ALL NODE
C                    DATA BY THE NUMBER OF ARCS AND NODES IN ONE YEAR,
C                    RESPECTIVELY, EACH TIME A SOLUTION FOR ONE YEAR
C                    HAS BEEN FOUND.
C***********************************************************************
C
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /ADATA/ NF(3600),NT(3600),COST(1800),AMP(1800),FLOW(1800),H
     1I(1800),LO(1800),ARCS
      COMMON /MISC/ AMIN,ADELT,NDELT
      REAL LO
      INTEGER A,AA,AMIN,ARCS,AX,ADELT
      INTEGER AIMIN
C***********************************************************************
C                    SHIFT ARC DATA
C***********************************************************************
      IF (ISOLVE.EQ.1) RETURN
      DO 50 ISOLVE=2,NSOLVE
      DO 40 ISEAS=1,NSEAS
      I=(ISOLVE-1)*NSEAS+ISEAS
      AIMIN=AMIN+(I-1)*(NJ+1+2*NRES+NL+NS+NJ)
      DO 30 J=1,NJ
      AA=AIMIN+J
      A=AA-ADELT
      IF (NT(AA).EQ.3) GO TO 20
      NF(A)=NF(AA)-NDELT
      NT(A)=NT(AA)-NDELT
      GO TO 30
20    NF(A)=NF(AA)-NDELT
      NT(A)=3
30    CONTINUE
40    CONTINUE
50    CONTINUE
      AX=AMIN+ADELT+1
      DO 60 AA=AX,ARCS
      A=AA-ADELT
      LO(A)=LO(AA)
      HI(A)=HI(AA)
      COST(A)=COST(AA)
      FLOW(A)=FLOW(AA)
      AMP(A)=AMP(AA)
60    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SHORT (IENTER,ILEAV)
C***********************************************************************
C                    SUBROUTINE TO FIND THE INITIAL AND SUBSEQUENT FLOW
C                    AUGMENTING TREE
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      INTEGER ARCS
      REAL LOWER
      EXTERNAL FLMXC,AMPF,COSTF
      IF (IFS.NE.0) GO TO 160
C***********************************************************************
C                    SET UP POINTERS TO FIND INITIAL TREE.
C***********************************************************************
      DO 20 I=1,NODES
      BARC(I)=0
      FARC(I)=0
      RARC(I)=0
      DISSET(I)=1
      GAN(I)=1
      V(I)=9999.
20    CONTINUE
      V(SOURCE)=0
      ICHANG=0
      IENTER=1
      ITF=0
      IFS=1
C***********************************************************************
C                    SET UP SHORTEST PATH TREE FOR FIRST ITERATION.
C***********************************************************************
30    CONTINUE
      DO 80 K=1,NARC
      IF ((UPPER(K)-FLOW(K)).GT.EPS) GO TO 40
      IF ((FLOW(K)-LOWER(K)).LT.EPS) GO TO 80
      II=JARC(K)
      JJ=IARC(K)
      POT=V(II)*AMP(K)-COST(K)
      I=K+NARC
      GO TO 60
40    CONTINUE
      I=K
      JJ=JARC(I)
      II=IARC(I)
      IF ((LOWER(I)-FLOW(I)).GT.EPS) GO TO 50
      POT=(V(II)+COST(I))/AMP(I)
      GO TO 60
50    CONTINUE
      POT=(V(II)-BIG)/AMP(I)
60    CONTINUE
      IF ((V(JJ)-POT).LT.EPS) GO TO 80
      V(JJ)=POT
      BARC(JJ)=I
      IF (ITF.EQ.0) GO TO 70
      CALL LOOP (I,JJ)
70    CONTINUE
      ICHANG=1
80    CONTINUE
      IF (ICHANG.EQ.0) GO TO 90
      ICHANG=0
      ITF=1
      GO TO 30
90    CONTINUE
C***********************************************************************
C                    CALCULATE FORWARD POINTERS FOR FIRST ITERATION.
C***********************************************************************
      DO 130 I=1,NODES
      IF (DISSET(I).EQ.0) GO TO 130
      KK=BARC(I)
      IF (KK.EQ.0) GO TO 130
```

```
      LL=IARC(KK)
      IF (FARC(LL).NE.0) GO TO 100
      FARC(LL)=KK
      GO TO 130
100   CONTINUE
      MM=FARC(LL)
110   CONTINUE
      MN=JARC(MM)
      IF (RARC(MN).NE.0) GO TO 120
      RARC(MN)=KK
      GO TO 130
120   CONTINUE
      MM=RARC(MN)
      GO TO 110
130   CONTINUE
140   CONTINUE
      IF (IPRINT.LT.2) GO TO 150
      PRINT 330, (BARC(I),I=1,NODES)
      PRINT 340, (FARC(I),I=1,NODES)
      PRINT 350, (RARC(I),I=1,NODES)
      PRINT 310, (DISSET(I),I=1,NODES)
      PRINT 300, (V(I),I=1,NODES)
      PRINT 320, (GAN(I),I=1,NODES)
      PRINT 360, (FLOW(I),I=1,NARC)
150   CONTINUE
      RETURN
C*********************************************************************
C                FIND NEW FLOW AUGMENTING TREE AFTER THE FIRST
C                ITERATION.
C*********************************************************************
160   CONTINUE
      DO 170 I=1,NODES
      DISSET(I)=0
170   CONTINUE
C*********************************************************************
C                DELETE BRANCH FROM BASIS AFTER THE FIRST ITERATION.
C*********************************************************************
      II=IROOT
      I=BARC(II)
      ILEAV=I
      IA=IARC(I)
      CALL DESUB (I,IA)
      RARC(II)=0
      BARC(II)=0
C*********************************************************************
C                SET NEW NODE GAINS ON DISCONNECTED NODES.
C*********************************************************************
180   CONTINUE
      I=FARC(II)
      IF (I.EQ.0) GO TO 190
      JJ=JARC(I)
      IF(I.GT.NARC)GAN(JJ)=GAN(II)*AMP(I-NARC)
      IF(I.LE.NARC)GAN(JJ)=GAN(II)/AMP(I)
      II=JJ
      IF (II.EQ.IROOT) GO TO 190
      GO TO 180
190   CONTINUE
      J=RARC(II)
      DISSET(II)=1
      IF (J.EQ.0) GO TO 200
      II=JARC(J)
      K=BARC(II)
      JJ=IARC(K)
      IF(K.GT.NARC)GAN(II)=GAN(JJ)*AMP(K-NARC)
      IF(K.LE.NARC)GAN(II)=GAN(JJ)/AMP(K)
      GO TO 180
200   CONTINUE
      K=BARC(II)
      IF (II.EQ.IROOT) GO TO 210
      II=IARC(K)
      GO TO 190
210   CONTINUE
C*********************************************************************
C                DETERMINE THE NEW BRANCH TO ENTER THE BASIS.
C*********************************************************************
      SICH=1.E+10
      IENTER=0
      DO 260 K=1,NARC
      IF ((UPPER(K)-FLOW(K)).GT.EPS) GO TO 220
C*********************************************************************
C                LOOKING AT A BACKWARD BRANCH
C*********************************************************************
      IF ((FLOW(K)-LOWER(K)).LT.EPS) GO TO 260
      JJ=IARC(K)
      IF (DISSET(JJ).EQ.0) GO TO 260
      II=JARC(K)
      I=K+NARC
      IF (DISSET(II).EQ.0) GO TO 250
C*********************************************************************
C                NEW BRANCH FORMS A LOOP
C*********************************************************************
      GO TO 230
220   CONTINUE
      I=K
      JJ=JARC(I)
      IF (DISSET(JJ).EQ.0) GO TO 260
      II=IARC(I)
      IF (DISSET(II).EQ.0) GO TO 250
C*********************************************************************
C                NEW BRANCH FORMS A LOOP.
C*********************************************************************
230   CONTINUE
      GALPIV=GAN(II)/(GAN(JJ)*AMPF(I))
      IF (GALPIV.GE..999) GO TO 260
      POTCH=(((V(II)+COSTF(I))/AMPF(I))-V(JJ))/((1.-GALPIV)*GAN(JJ))
240   CONTINUE
      IF (POTCH.GE.SICH) GO TO 260
      IF(POTCH.LT.0.)POTCH=0.
      SICH=POTCH
      IENTER=I
      GO TO 260
C*********************************************************************
C                NEW BRANCH DOES NOT FORM A LOOP.
C*********************************************************************
250   CONTINUE
      POTCH=(((V(II)+COSTF(I))/AMPF(I))-V(JJ))/GAN(JJ)
      GO TO 240
260   CONTINUE
      IF (IENTER.EQ.0) GO TO 280
C*********************************************************************
C                CHANGE NODE LABELS AND POINTERS TO REFLECT
C                ENTERING BRANCH.  CHANGE POINTERS.
C*********************************************************************
      CALL TRECHG (IENTER,ILEAV)
C*********************************************************************
C                CHANGE NODE POTENTIALS.
C*********************************************************************
      DO 270 II=1,NODES
      IF (DISSET(II).EQ.0) GO TO 270
      V(II)=SICH*GAN(II)+V(II)
270   CONTINUE
      GO TO 140
280   CONTINUE
      PRINT 290, FLONET
      II=IARC(ILEAV)
      CALL ADSUB (ILEAV,II)
      JJ=JARC(ILEAV)
      BARC(JJ)=ILEAV
      GO TO 140
290   FORMAT (1X,20H MAXIMUM FLOW FOUND ,F20.10)
300   FORMAT (1X,10H  LAB      ,(21F5.1))
310   FORMAT (1X,10H  DISSET   ,(21I5))
320   FORMAT (1X,10H  GAIN     ,(1X,21F5.3))
330   FORMAT (1X,10H  BARC     ,(21I5))
340   FORMAT (1X,10H  FARC     ,(21I5))
350   FORMAT (1X,10H  RARC     ,(21I5))
360   FORMAT (1X,10H  FLOW     ,(21F5.1))
      END
```

```
      SUBROUTINE SPRINT
C***********************************************************************
C                    THIS SUBROUTINE PRINTS AN ANNUAL SUMMARY OF ACTUAL
C                    COSTS, PRESENT VALUES, AND AVERAGE ANNUAL COSTS
C***********************************************************************
      COMMON /NDATA/ NSOLVE,LSOLVE,NYEAR,NS,SCAP,NI,NREAD,MAXYR
      COMMON /SDATA/ NJ,NRES,NJUNC,NL,NC,NR,NYR,NSEAS,IYEAR,IDATE,IMP,IN
      COMMON /CDATA/ PERINT,NPAY,POWR,RESONM,CONONM,CIMP,LAGR,LAGC,POWFA
     1C(12)
      COMMON CRESC(50),CRESOM(50),CLINKC(50),CLINOM(50),CIMPT(50),CPOWR(
     150),CTOTAL(50),PVRES(50),PVROM(50),PVLINC(50),PVLIOM(50),PVIMP(50)
     2,PVPOWR(50),PVTOT(50),ANRESC(50),ANRESO(50),ANLINC(50),ANLINO(50),
     3ANIMP(50),ANPOWR(50),ANTOT(50),CPEN(50),PVPEN(50),ANPEN(50)
20    FORMAT (1H1,20X,31HTOTAL COSTS IN THOUSAND DOLLARS)
30    FORMAT (1H1,20X,36HPRESENT VALUES IN THOUSAND DOLLARS (,I5,6H BASE
     1))
40    FORMAT (1H1,20X,42HAVERAGE ANNUAL COSTS IN THOUSAND DOLLARS (,I3,1
     15H-YEAR PAYMENTS))
50    FORMAT (1H0,14X,10HRESERVOIRS,15X,8HCONDUITS,10X,7HIMPORTS,8X,6H P
     1OWER,6X,8HDEFICITS,9X,5HTOTAL/18H  YEAR     CAPITAL,5X,31HO AND M
     2    CAPITAL     O AND M)
60    FORMAT (I6,3X,4(F9.0,3X),3(F9.0,5X),F10.0)
C***********************************************************************
C                    PRINT ACTUAL COSTS BY YEARS
C***********************************************************************
      WRITE (6,20)
      WRITE (6,50)
      DO 70 IYR=1,NYEAR
      J=IYR+1
      WRITE (6,60) IYR,CRESC(J),CRESOM(J),CLINKC(J),CLINOM(J),CIMPT(J),C
     1POWR(J),CPEN(J),CTOTAL(J)
70    CONTINUE
C***********************************************************************
C                    PRINT PRESENT VALUE BY YEARS
C***********************************************************************
      ICAL=IDATE
      WRITE (6,30) ICAL
      WRITE (6,50)
      DO 80 IYR=1,NYEAR
      J=IYR+1
      WRITE (6,60) IYR,PVRES(J),PVROM(J),PVLINC(J),PVLIOM(J),PVIMP(J),PV
     1POWR(J),PVPEN(J),PVTOT(J)
80    CONTINUE
C***********************************************************************
C                    PRINT AVERAGE ANNUAL COST COMPONENTS
C                    BY YEARS
C***********************************************************************
      WRITE (6,40) NPAY
      WRITE (6,50)
      DO 90 IYR=1,NYEAR
      J=IYR+1
      WRITE (6,60) IYR,ANRESC(J),ANRESO(J),ANLINC(J),ANLINO(J),ANIMP(J),
     1ANPOWR(J),ANPEN(J),ANTOT(J)
90    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRECHG (IENTER,ILEAV)
C***********************************************************************
C                    SUBROUTINE TO FORM THE FLOW AUGMENTING TREE BY
C                    DELETING ONE ARC FROM THE TREE AND INSERTING
C                    A NEW ARC INTO THE TREE
C***********************************************************************
      COMMON /ADATA/ IARC(3600),JARC(3600),COST(1800),AMP(1800),FLOW(180
     10),UPPER(1800),LOWER(1800),ARCS
      COMMON /XDATA/ V(500),BARC(500),RARC(500),FARC(500),DISSET(500),GA
     1N(500),ICHK(500),LIST(500)
      COMMON /V/ SOURCE,SINK,NARC,OUTFLO,FLONET,CSTNOW,TOTCST,NODES,IFS,
     1IROOT,EPS,BIG,NDEG,NLOP,SICH,ITER,NPRIT,TIMAX,TIME,IPRINT
      INTEGER SOURCE,SINK,BARC,FARC,RARC,DISSET
      REAL LOWER
      INTEGER ARCS
      EXTERNAL FLMXC,AMPF,COSTF
      IROOT=JARC(ILEAV)
      NLIS=0
      JJ=JARC(IENTER)
C***********************************************************************
C                    DELETE PATH FROM JARC(IENTER) TO IROOT
C***********************************************************************
20    CONTINUE
      JB=BARC(JJ)
      IF (JJ.EQ.IROOT) GO TO 30
      NLIS=NLIS+1
      II=IARC(JB)
      CALL DESUB (JB,II)
      IF(JB.LE.NARC)I=JB+NARC
      IF(JB.GT.NARC)I=JB-NARC
      ICHK(NLIS)=I
      BARC(JJ)=0
      RARC(JJ)=0
      JJ=II
      GO TO 20
30    CONTINUE
C***********************************************************************
C                    ADD IN THE REVERSE OF THE PATH JUST DELETED
C***********************************************************************
      IF (NLIS.EQ.0) GO TO 40
      I=ICHK(NLIS)
      NLIS=NLIS-1
      II=IARC(I)
      JJ=JARC(I)
      CALL ADSUB (I,II)
      BARC(JJ)=I
      GO TO 30
40    CONTINUE
      II=IARC(IENTER)
      CALL ADSUB (IENTER,II)
      JJ=JARC(IENTER)
      BARC(JJ)=IENTER
      RARC(JJ)=0
      RETURN
      END
```

```
      SUBROUTINE SUMRY
C*******************************************************************
C     THIS SUBROUTINE PRINTS AN ANNUAL SUMMARY OF ACTUAL
C     COSTS, PRESENT VALUES, AND AVERAGE ANNUAL COSTS.
C*******************************************************************
[illegible]
      ... TOTAL COSTS IN THOUSAND DOLLARS ...
[illegible]
```

AL-IV

VARIABLE DESCRIPTIONS

BLOCK COMMON

SUBROUTINES	ADATA	ADJUST	APRNT	CDATA	JDATA	LDATA	MDATA	MISC	NDATA	PRT	RDATA	SDATA	SHORTS	V	XDATA	BLANK1	BLANK2	BLANK3	BLANK4					
MAIN	●				●	●	●	●	●			●		●										
ADJUST	●	●						●	●			●												
ADSUB	●													●	●									
AMPF	●													●	●									
APRINT			●						●	●	●	●					●							
ARC	●	●		●	●	●	●	●	●	●	●	●	●	●	●	●								
COSTF	●													●	●									
COSTX			●	●	●	●			●		●	●						●						
DESUB	●													●	●									
FLCHG	●													●	●									
FLMXC	●													●	●									
FLOP	●													●	●									
GAIN	●								●	●		●		●	●									
INPUT	●		●	●	●	●	●		●	●	●	●	●											
LOOP	●													●	●				●					
MAXFLO	●													●	●									
OUTPUT	●								●			●												
PREPAR	●		●	●				●	●			●		●	●		●							
SHIFT	●							●	●			●												
SHORT	●													●	●									
SPRINT				●					●			●						●						
TRECHG	●													●	●									

ALLOCATION PROGRAM

BLOCK COMMON USAGE IN SUBROUTINES

VARIABLE DESCRIPTIONS FOR SUBROUTINE

ADJUST

VARIABLE NAME	DEFINITION
A	Arc Number
AA	Arc Number for Determination of the Bounds and Flow in the New Initial Storage Arcs
AIMIN ARMIN ASMIN	Working Variables

VARIABLE DESCRIPTIONS FOR SUBROUTINE

APRINT

VARIABLE NAME	DEFINITION
IL	Link Number for Imports
IYR	Year for which Results are Being Printed
M	Subscript for Mean Values to be Printed

Variable Descriptions for Subroutine ARC

Variable Name	Definition
A, AB	Arc Number
ANEW	Number of the First Arc to be Added to the Network
DEMAND	Upper Bound of Total Demand from the Network in Thousands of Acre-Feet
DSHORT	Lower Bound of Total Demand from the Network in Thousands of Acre-Feet
EFF	Pump and Motor Efficiency (Set to .8)
FSTORE	Total Amount of Storage in the Last Time Period In the Network in Thousands of Acre-Feet
I	Season Number in the Network
IFLOW	Total Amount of Input in the Year to be Added to the Network in Thousands of Acre-Feet
IMIN	Number of the First Season to be Added to the Network
IMPORT	Total Amount of Import to the Network in Thousands of Acre-Feet
INPUT	Total Amount of Input to the Network in Thousands of Acre-Feet
JFLOW	Total Amount of Input in the Year to be Added to the Network in Thousands of Acre-Feet
KIN	Fortran Logical Unit Assigned to Input File
NDEL	Number of Nodes per Year
NEVAP	Temporary Value of Reservoir Evaporation in Thousands of Acre-Feet
NN	Number of Seasons in the Network

SPILL	Total Amount of Spill from the Network in Thousands of Acre-Feet
SUBS	Number of Months in Each Season
AA, AIMIN,ALMIN ARMIN,ASMIN,INEW, JF,JT,K,KK,NMIN	Working Variables

Variable Descriptions for Subroutine FLMXC

Variable Name	Definition
I	Arc Number
FLMXC	Maximum Allowable Change in the Flow in Arc I
S	$\leq$ 0, if Flow is to be Decreased > 0, if Flow is to be Increased
II, JJ K	Working Variables

Variable Descriptions for Subroutines FLOP

Variable Name	Definition
FLMAX	Maximum Possible Change in Flow for a Flow Generating Cycle
GN	Gain Factor for a Flow Generating Cycle
IROOTL	Root Node for a Flow Generating Cycle
JJ	Initial Node in Flow Generating Cycle
IJ,IJK, FLMXT	Working Variables

Variable Descriptions for Subroutine PREPAR

Variable Name	Definition
A	Arc Number
AL	Link Number for Import Arc
IYR	The Year for which Output is Being Printed
MINP	Minimum Value in the Vector PI(N)
AIMIN, ALMIN, ARMIN, ASMIN, IA, IL, JA, M	Working Variables

Variable Descriptions for Subroutine SHIFT

Variable Name	Definition
A	Arc Number
AA	Arc Number whose Data will be Shifted to Arc Number (A)
AIMIN	Working Variable
AX	First Arc to be Shifted
I	Season Number
N	Node Number
NN	Node Number whose Data will be Shifted to Node Number (N)
NX	First Node to be Shifted

Variable Descriptions for Subroutines
SHORT

Variable Name	Definition
IENTER	Arc Entering the Flow Augmenting Tree
IFS	1, if Flow Augmenting Trees Subsequent to the First Tree are being Found 0, if Otherwise.
ILEAV	Arc Leaving the Flow Augmenting Tree
ITF	1, if the Initial Flow Augmenting Tree is to be Found 0, Otherwise.
POTCH	Change in Node Potential for Node I
AMX,CSTX, GALPIV,II, JJ,KK,LL,I,K, POT,ICHANG	Working Variables

Variable Descriptions for Subroutine
SPRINT

Variable Name	Definition
ICAL	Calendar Date (Year) to which Present Value Costs are Referenced
J	Working Variable

Variable Descriptions for Subroutine TRECHG

Variable Name	Definition
IENTER	Arc to Enter Flow Augmenting Tree
ILEAV	Arc to Leave Flow Augmenting Tree
NLIS,JJ, JB, II	Working Variables

Variable Descriptions for Block Common /ADATA/

Variable Name	Definition
AMP(A)	Amplitude Factor for flow in Arc (A) (Must be positive value greater than zero)
ARCS	Number of Arcs in the Network
COST(A)	Unit Cost of Flow in Arc (A) in Thousands of Dollars per Thousand Acre-Foot
FLOW(A)	Flow entering Arc (A) in Thousands of Acre-Feet
HI(A)	Upper Bound on Flow entering Arc (A) in Thousands of Acre-Feet
LO(A)	Lower Bound on Flow entering Arc (A) in Thousands of Acre-Feet
NF(A)	Origin Node of Arc (A)
NT(A)	Destination Node of Arc (A)

Variable Descriptions for Block Common /ADJST/

Variable Name	Definition
DADJ	Flow Adjustment in the Bounds of the net Demand Arc for Dropping a Year from the Network in Thousands of Acre-Feet
IADJ	Flow Adjustment in the Bounds of the Net Input Arc for Dropping a Year from the Network in Thousands of Acre-Feet
IDROP	Amount of Inputs in the Year to be Dropped from the Network in Thousands of Acre-Feet
ISADJ	Flow Adjustment in the Lower Bound of the Net Demand Arc for Dropping a Year from the Network in Thousands of Acre-Feet
ISTORE	Total Initial Storage in the System in Thousands of Acre-Feet
JDROP	Amount of Demands in the Year to be Dropped from the Network in Thousands of Acre-Feet
MADJ	Flow Adjustment in the Bounds of the Net Import Arc for Dropping a Year from the Network in Thousands of Acre-Feet
MDROP	Amount of Imports in the Year to be Dropped from the Network in Thousands of Acre-Feet
SADJ	Flow Adjustment in the Bounds of the Net Spill Arc for Dropping a Year from the Network in Thousands of Acre-Feet
SDROP	Amount of Spills in the Year to be Dropped from the Network in Thousands of Acre-Feet

Variable Descriptions for Block Common
/APRNT/

Variable Name	Definition
AIMP(IYR)	Annual Power Cost in Year (IYR) in Thousands of Dollars
APOW(IYR)	Annual Imports in Year (IYR) in Acre-Feet
QM(L)	Maximum Flow in Link (L) in CFS
ISHORT(IYR)	Junction Shortage Costs in Year (IYR) in Thousands of Dollars
ICOST(IYR)	Total Penalty Costs in Year (IYR) in Thousands of Dollars

Variable Descriptions for Block Common
/CDATA/

Variable Name	Definition
CIMP	Unit Cost of Imported Water in Dollars per Acre-Foot
CONONM	Conduit Annual Operation and Maintenance Cost Factor as a Fraction of Construction Cost
LAGC	Number of Years Lag Between Incurring Construction Cost for Conduits and First Year of Operation
LAGR	Number of Year Lag Between Incurring Construction Cost for Reservoirs and First Year of Operation
NPAY	Repayment Period in Years, Used to Compute Average Annual Costs
PERINT	Interest Rate (Decimal) for all Incurred Costs
POWFAC(I)	Seasonal Power Cost Coefficient
POWR	Unit Cost of Power in Dollars per Kilowatt-Hour
RESONM	Reservoir Annual Operation and Maintenance Cost Factor as a Fraction of Construction Cost

VARIABLE DESCRIPTIONS FOR BLOCK COMMON
/JDATA/

VARIABLE NAME	DEFINITION
CRES(J)	Capital Cost of Reservoir J in Thousands of Dollars
DLO(J)	Fraction of Seasonal Demand at Junction J which Must be Met
FSTART(J)	Fraction of Reservoir Full for Starting Storage
NLINK(J)	Node Number for Effluent Discharge of Wastewater Generated by Water Demand at Node J
RCAP(J)	Reservoir J Capacity in Thousands of Acre-Feet
RMIN(J)	Minimum Capacity of Reservoir J in Thousands of Acre-Feet
RNAME(J)	Reservoir or Junction Name
RETSW(J)	Effluent S/W Ratio (Sewage to Intake Water Ratio) for Demand at Node J

Variable Descriptions for Block Common /LDATA/

Variable Name	Definition
CCAP(L)	Maximum Capacity of Link L in CFS
CLIFT(L)	Pump Lift of Link L in Feet
CLINK(K,L,J)	Cost-Capacity Coefficients for a Second-Order Polynomial for Link L
CMIN(L)	Minimum Capacity of Link L in CFS
CNAME(L)	Link Name
CPUMP(L)	Unit Cost of Pumping in Thousands of Dollars Per CFS Per Foot
EL(L)	Fraction of Flow Entering Link L Which Leaves Link L (Amplitude of Link L)
LNODE(L,K)	Node Numbers at the Ends of Link L

Variable Descriptions for Block Common
/MDATA/

Variable Name	Definition
ELF(ISEAS)	Seasonal Coefficient for Link Amplitudes in Period ISEAS
KSPILL	Spill Cost in Dollars per Acre-Foot
XIMP	Maximum Annual Available Import in Thousands of Acre-Feet
XX(ISEAS)	Seasonal Import Coefficient for Period ISEAS

Variable Descriptions for Block Common
/NDATA/

Variable Name	Definition
LSOLVE	Number of Years Contained in the Network Flow Problem
MAXYR	Not Used
NI	Number of Seasons in the Network Flow Problem (Equals NSEAS times NSOLVE)
NREAD	The First Year of the Problem to be Solved by the Allocation Program (=1)
NS	Total Number of Nodes from Which Spills can Occur
NSOLVE	Maximum Number of Years in a Network the Allocation Program can Solve
NYEAR	The Last Year of the Problem to be Solved by the Allocation Program
SCAP	Capacity of Spill Arcs in Thousands of Acre-Feet

Variable Descriptions for Block Common
/MISC/

Variable Name	Definition
ADELT	Number of Arcs in One Year
AMIN	Number of Initial Reservoir Storage and Net Balance Arcs
NDELT	Number of Nodes in One Year

Variable Descriptions for Block Common
/PRT/

Variable Name	Definition
IPRNT(IYR)	0, 1, 2, or 3 If 0 -- Print All Output for Year (IYR) except the Network Model for that Year If 1 -- Print All Output for Year (IYR) and Print the Network Model for that Year If 2 -- Omit Printing of Dual Values for Year (IYR) If 3 -- Omit All Printing for Year (IYR)

VARIABLE DESCRIPTIONS FOR BLOCK COMMON /SDATA/

VARIABLE NAME	DEFINITION
IDATE	Calendar Starting Date of the Problem
IMP	Import Node Number
IN	Input Tape Number
IYEAR	First Year
NC	Number of Canals
NJ	Number of Nodes (Reservoirs and Junctions)
NJUNC	Number of Junctions
NL	Number of Links
NR	Number of River Reaches
NRES	Number of Reservoir
NSEAS	Number of Seasons per Year
NYR	Number of Years

VARIABLE DESCRIPTIONS FOR BLOCK COMMON /RDATA/

VARIABLE NAME	DEFINITION
JBLT(J)	Year Reservoir J is Built
JS(J)	Node J from Which Spills can Occur
LBLT(L)	Year Link L is Built

VARIABLE DESCRIPTIONS FOR BLOCK COMMON

/SHORTS/

VARIABLE NAME	DEFINITION
ACO(I,J)	Slope of Lines Approximating the Area-Capacity Curve for Reservoir I in the Lower (J=1) and Upper (J=2) Range of Storage
BND(J)	Fraction of Maximum Storage in Reservoir I Corresponding to the Junction Point between the Line Segments Approximating the Area-Capacity Curve
CST(J,I)	Unit Cost of Demands from Junction I in Season J in Dollars per Acre-Foot
IOPT(I)	Bivalued variable specifying the operating option (1 or 2) for Reservoir I
TCST(I)	Unit Benefit of Storage at Target Capacity in Reservoir I in Dollars per Acre-Foot
TFAC(J,I)	Fraction of Full Reservoir I to Obtain Target Capacity in Year J ($J \leq 13$)
ALP(I)	Safety Factor used to Increase the Annual Terminal Storage Targets for Reservoir I under Operating Rule 2.

Variable Descriptions for Block Common

/V/

Variable Name	Definitions
BIG	Large Positive Value (Set to 1000 in Subroutine GAIN)
CSTNOW	Cost for the Network Flows in the Current Solution
EPS	Small Positive Value (Set to .001 in Subroutine GAIN)
FLONET	Inflow into the Sink Node for the Current Solution
IFS	Used as a Counter to Indicate First and Subsequent Calls to Subroutine SHORT
IPRINT	Print Option Variable for Providing Intermediate Results from Network With Gains Algorithms: -1, No Intermediate Printing Executed 0, Print Only Total Network Cost and Number of Iterations Required for Solution 1, Print All of Above Plus Summary by Iteration giving Inflow into Sink, Network Cost, Arc Leaving Flow Augmeting Path, Arc Entering Path, and Increase in Potential at the Sink 2, Print All of Above Plus Summary by Iteration of forward, backward and right pointers, Nodes in the Flow Tree, Node Potentials, Node Gains, and Arc Flows
IROOT	Terminal Node for Arc Leaving the Flow Augmenting Path
ITER	Number of Iterations Required in Subroutine GAIN to Solve the Network Model
NARC	Number of Arcs in the Network Model
NDEG	Number of Degenerate Iterations Executed by the Program in Obtaining the Solution

NLOP	Number of Loop Iterations in Obtaining the Solution
NODES	Total Number of Nodes in the Network Model (Including Source and Sink)
NPRIT	Not Used
OUTFLO	Required Flow into the Sink Node
SICH	Increase in the Node Potential at the Sink for the Current Iteration in Subroutine GAIN
SINK	Node Number for the Flow Sink (Set to 7 in Subroutine GAIN)
SOURCE	Node Number for the Supply of Flow to the Network (Set to 1 in Subroutine GAIN)
TIMAX	Switch Used to Restart Algorithm when Storage Segmentation Constraints are Violated in any Network Solution
TIME	Not Used
TOTCST	Total Cost of the Optimal Solution to the Network Problem

Variable Descriptions for Block Common /XDATA/

Variable Name	Definition
BARC(I)	Backward Arc Pointer for Node I in the Three Label Tree Representation
DISSET(I)	1, if Node I is in the Flow Augmenting Path, 0, if Otherwise
FARC(I)	Forward Arc Pointer for Node I
GAIN(I)	Gain Factor for Node I
ICHK(I)	Counter Array Used in Determining Loops in the Network
LIST(I)	List of Nodes
RARC(I)	Right Arc Pointer for Node I
V(I)	Node Potential for Node I

Variable Descriptions for Blank Common

1

Variable Name	Definition
D(ISEAS,J)	Seasonal Water Demand for Junction J in Thousands of Acre-Feet in Period ISEAS
E(ISEAS,J)	Seasonal Evaporation Coefficients for Reservoir J in Period ISEAS
U(J,ISEAS)	Seasonal Unregulated Inflow to Junction J in Period ISEAS in Thousands of Acre-Feet

Variable Descriptions for Blank Common

2

Variable Name	Definition
CC(I,J)	Dual Value for the Reservoir J Continuity Constraints in Season I in Dollars per Acre-Foot
IDEF(I,J)	Deficits at Node J in Season I in Thousands of Acre-Feet
IMPC(IYR)	Cost of Imported Water in Year IYR in Thousands of Dollars
ITOT(IYR)	Total Cost of Deficits Plus Imports Plus Pumping in Year IYR in Thousands of Dollars
JQ(I,J)	Spills Leaving System from Node JS(J) in Season I in Thousands of Acre-Feet
CL(I,L)	Dual Value for Link L in Season I in Thousands of Dollars per CFS
LP(I,L)	Power Cost in Link L During Season I in Thousands of Dollars
LQ(I,L)	Flow in Link L During Season I in CFS
RC(I,J)	Dual Value for the Reservoir J Storage Contents in Season I in Dollars per Acre-Foot
RS(I,J)	Reservoir J Storage Contents for Season I in Thousands of Acre-Feet

Variable Descriptions for Blank Common

3

Variable Name	Definition
CIMPT(IYR) PVIMP(IYR) ANIMP(IYR)	Input Cost in Thousands of Dollars
CLINKC(IYR) PVLINC(IYR) ANLINC(IYR)	Construction Cost for Link in Thousands of Dollars
CLINOM(IYR) PVLION(IYR) ANLINO(IYR)	Operation and Maintenance Costs for Link in Thousands of Dollars
CPOWR(IYR) PVPOWR(IYR) ANPOWR(IYR)	Power Costs in Thousands of Dollars
CRESC(IRY) PVRES(IYR) ANRESC(IYR)	Reservoir Construction Costs in Thousands of Dollars
CRESOM(IYR) PVROM(IYR) ANRESO(IYR)	Reservoir Operation and Maintenance Costs in Thousands of Dollars
CPEN(IYR) PVPEN(IYR) ANPEN(IYR)	Penalty Costs for Deficits Taken in Thousands of Dollars
CTOTAL(IYR) PVTOT(IYR) ANTOT(IYR)	Total for Cost Components Listed in Thousands of Dollars

NOTE: Prefix C stands for cumulative costs through year IYR
Prefix PV stands for present values through year IYR
Prefix AN stands for average annual costs through year IYR

Variable Descriptions for Blank Common

4

Variable Name	Definition
ICHT(I)	Pointer for Node I that is Set to 1 if an Arc in the Flow Augmenting Tree Originates from that Node