DEVELOPMENT OF A MORE COMPREHENSIVE WATER PLANNING MODEL FOR THE STATE OF WYOMING

Duane D. Stroup
Victor R. Hasfurther

May 1993
WWRC-93-11

Technical Report

Submitted to

Wyoming Water Development Commission
Cheyenne, Wyoming

and

Wyoming Water Resources Center
University of Wyoming
Laramie, Wyoming

Submitted by

Duane D. Stroup
and
Victor R. Hasfurther
Department of Civil & Architectural
Engineering
University of Wyoming
Laramie, Wyoming

May 1993

ABSTRACT

Two programs (WIRSOS and HYDROSS) for modeling river systems were compared for use by the state of Wyoming. It was found that the Wyoming Integrated River System Operation Study (WIRSOS) planning model best fit the needs of the state. However, it was also found that WIRSOS did not completely satisfy every aspect of river system modeling. In fact, HYDROSS, the program of the United States Bureau of Reclamation, is much superior at modeling the operation of reservoirs in a river system, but HYDROSS is lacking in several other areas. Therefore, to improve the performance of WIRSOS, many of the reservoir modeling capabilities of HYDROSS were programmed into the WIRSOS model. The improved WIRSOS program models river systems with greater accuracy and provides for a more detailed analysis of water issues within the state of Wyoming.

ACKNOWLEDGMENTS

The authors wish to express sincere gratitude to several people who provided assistance, advice, and support during this research. Special thanks to Becky Mathisen of the Wyoming State Engineer's Office, who provided expertise on the WIRSOS program.

Funding was provided by the Wyoming Water Resources Center and the Wyoming Water Development Commission.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

CHAPTER PAGE

TABLE OF CONTENTS

CHAPTER PAGE

TABLE OF CONTENTS

CHAPTER PAGE

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

Water is a precious commodity in the Western United States. It provides the catalyst for economic development and sustains life. However, there is not an unlimited supply, and many separate interests often compete for the available water resources. Uses for water in the arid west include; industry, recreation, municipalities, agriculture, livestock, and domestic purposes (Brosz and Jacobs, 1980). To provide a means of ensuring a somewhat dependable supply of water for these applications, the Wyoming State Engineers Office (WSEO) allocates the states available water resources between all users based on the appropriation doctrine.

The actual allocation of surface water resources is a complex task, one that involves determining the flow in a river and distributing that water to the appropriate users. The distribution of this water must be performed according to Wyoming State Water Law (WSWL), which is based on the prior appropriation doctrine ("first in time, first in right"). In addition, the WSEO determines how water is stored in reservoirs and how water is released from these reservoirs to the river system. Besides the uses listed by

Brosz and Jacobs (1980), the WSEO must also consider water consumed by conveyance losses and evaporation. Furthermore, since all water diverted from a stream is not consumed and some of the water returns to the river system, the return flow must be restored to the river system. Through the coordination of all water uses, an accounting of the available water in the river system is made. It is then possible for the WSEO to evaluate and adjudicate additional demands for this water.

The WSEO must consider an intricate array of variables and a large amount of stream data when allocating water resources. This complexity creates a favorable environment for a computer model capable of simulating river basin operations. The Wyoming Integrated River System Operation Study (WIRSOS) planning model was developed to simulate river basin operations. In addition to simulating river basin operations, the model does a water accounting (water balance) of the water in the river system.

WIRSOS models a stream system corresponding to WSWL. However, many nuances to river system operation exist that cannot be modeled according to WSWL, and hence do not allow WIRSOS to predict the water available in the river more precisely.

The purpose of this thesis was to develop a WIRSOS model that would conform to WSWL. One that could improve the precision of river system simulation in situations where WSWL is not strictly enforced.

To accomplish this purpose, the following tasks were established:

1. Develop a modular subroutine based program that would be easier for a user to understand and modify.

2. Examine previous versions of WIRSOS and the United States Bureau of Reclamation Program HYDROSS (Hydrologic River Operation Study System) and incorporate those features into the modularized version of WIRSOS which would improve its capabilities to simulate a river system in Wyoming.

3. Develop subroutines in WIRSOS which improve reservoir operations and improve the ability of WIRSOS to more precisely simulate true river system management in Wyoming.

Before any alterations were made to WIRSOS, the model was used in Wyoming to model the Wind River and the Green River. After calibration, the Wind River model was able to produce river flow that were within plus or minus 5% of actual river flows for the period of record (Wind 1991). The Green River produced flow that were within plus or minus 8% of actual river flows (Meena 1993). The objective of the current model is to improve this accuracy and to make it easier to model a river system.

A literature search was conducted to determine what previous work had been done in the area of water planning models. This information is discussed in Chapter II. WIRSOS modifications are discussed in Chapter III, and a description of WIRSOS5 is given in Chapter IV. The testing of the model is discussed in Chapter V; finally, the summary and recommendations are presented in Chapter VI.

CHAPTER II

LITERATURE REVIEW

INTRODUCTION

Surface water has a significant impact on life in the arid west. According to Loucks (1976), surface water is the only source of hydro-electric power, water-based recreation, and flow for maintaining fisheries. Surface water also accounts for the majority of the arid west's irrigation and flood damage. Therefore, the first efforts to use mathematical models for water resource planning focused on surface water. This has resulted in the existence of many of methods for mathematically modeling all aspects of surface water.

RESERVOIR MODELING

The majority of surface water models deal with some aspect of reservoir operation. These models simulate reservoir operation, provide for real time operation of a reservoir system, and optimize reservoir operation, hydro-power output, and flood control. Many programming techniques were also used to create these models. Yeh (1985) provides an excellent review of the various models and modeling techniques used to simulate reservoir operations. A significant amount of research has occurred in the field of reservoir modeling since Yeh (1985) did his review. This later research focused on applying expert systems to real-time reservoir operations (Anibal, et al. 1990), optimization of a system involving multiple reservoirs (Changchit and Terrell 1989; Kuo, et al. 1990), methods to incorporate social objectives into the reservoir models (Flug and Ahmed 1990; Flug, et al. 1990), optimizing hydropower output from a system of reservoirs (Diaz and Fontane 1989; Tejada-Guibert, et al. 1990), and integrating multi-objective planning into reservoir operation (Loganathan and Bhattacharya 1990). There has also been a tendency to adapt large models for use on personal computers (Rockwood, et al. 1988; Eichert and Franke 1988; Mohammadi 1988) and to create models with interactive graphic user interfaces for easier model operation (Ford 1990). This is not a complete list of all the articles that have been written about reservoir system modeling, but indicates the types of articles available.

Of the numerous models that represent reservoir operation, two models warrant further discussion. These models represent major advances in the area of reservoir operation modeling.

CVP MODEL

The first model is the California Central Valley Project Power (CVPOWER) program for optimizing hydropower output from the Central Valley Project (CVP). The program optimizes the output from nine reservoirs and ten power plants on three rivers, subject to the water available in the system (Tejada-Guibert, et al. 1990). Other work includes efforts to integrate CVPOWER's optimization with the peak power demands of the local utility Pacific Power and Gas (Crower, et al. 1988; Staschus, et al. 1988). It is believed that the next step with the CVPOWER model will be to make it real-time operational. Real-time operation refers to the ability to model and operate a system at the same time, so that the results from the model change system operation.

TVA MODEL

The second program to be discussed is the Tennessee Valley Authority (TVA) reservoir operations model. This model provides a weekly schedule of releases from 40 major dams with hydropower capabilities in two river basins (Shane, et al. 1988). The model tries to optimize flood control and hydropower generation while meeting multi-objective uses at twenty-one of the reservoirs. This is accomplished by performing reservoir operation optimization at 19 of the reservoirs. The optimizations are based on past reservoir operations. Flows into the system are selected by the system operator from 52 years of historical flow data. In addition, the model uses climatic data, current river flow, and current reservoir levels to predict flows into the reservoirs for periods up to one week (Shane and Gilbert 1982). Courtney and Whitlock (1988) describe how remote sensors and micro-computers have replaced the old system of observers and main-frame computers for gathering and processing climatic and flow data. This eliminates the need to manually input the large amounts of data required for model operation which reduces the amount of time needed to run a simulation. Using these remote sensors and micro-computers, the model is currently able to simulate real-time operation of the system of reservoirs. This model is unique in the large number of reservoirs modeled and the ability to model such a large system on a real time basis.

RIVER BASIN MODELING

There are very few models that attempt to represent a complete river system. Wurbs (1991) provides an excellent review of programs for modeling, analyzing, and optimizing reservoir system operations. This review covers many of the water planning models discussed in this chapter.

Currently there are five techniques used to model a complete river system: Graphic Information Systems (GIS), water balancing, mass-balancing of inflows and outflows, network analysis, and water accounting. Examples of these types of models are presented and discussed in this chapter. Besides the models presented, numerous models were found that have been developed specifically for a single river or a single river basin. The models discussed either demonstrate one of the five techniques listed above, or a model that is directly related to this project or the state of Wyoming. In addition, a number of California models are described as an example of how a model evolves with the development of new programming techniques.

GIS MODELS

The main application of GIS in water resources has been in representing groundwater quantity and quality (Weghorst, et al. 1991). However, there are instances where GIS has been applied to a surface water system or to an integrated surface water/groundwater system. Weghorst, et al. (1991) describe a GIS model for the Santa Ana River Basin that accounts for the availability of surface water resources for groundwater recharge, but other than that the model's primary purpose is to model groundwater. Johnson (1991) presents a model that graphically represents changes in reservoir levels and the impact of such changes on riparian areas below the reservoir. This, allows the operator to propose different reservoir operating rules and evaluate their impact on environmentally critical reaches of the river downstream form the reservoir. GIS's strong point is its ability to visually display the results of a model run, demonstrating its suitability to information that is spatial in nature.

The states of Idaho (Anderson and Shaw 1987) and Oregon (Grainey 1987) have taken the data from their adjudicated water rights and entered it into a GIS system. These databases will allow a water engineer to display water rights and to produce maps of irrigated land.

GIS can not decide who gets water based on a priority date and enormous amounts of data required for GIS. Therefore, it is a good tool for showing the effects of more or less water in a river system and for modeling groundwater quality and quantity. However, it can only assign water on a spatial basis with no value given to priority. This would require taking GIS values and inputting them into a model for a watershed which handles water flow and water rights. Fisher (1989) describes an interface that accepts GIS information and formats the data for input into the Hydrological Simulation Program-Fortran (HSPF) watershed model.

WATER BALANCE MODELS

The Streamflow Synthesis and Reservoir Regulation (SSARR) model developed by the U. S. Corps of Engineers is a water balance model (Rockwood 1982). This is a complete hydrologic model that generates runoff for a drainage basin and predicts streamflow for that basin based on the runoff generated by the model. The runoff is generated from precipitation, humidity, snowmelt, infiltration, soil conditions, groundwater returns, and measured runoff. When combined with an automated data collection system the model is capable of real-time river system simulation (Rockwood, et al. 1988). The model provides for the operational management of reservoirs. A method is also available to account for diversions and irrigation requirements. This model is the first to introduce the concept of assigning flows at the headwaters of rivers and distributing the flow through the system. The model is highly generalized and has been adapted to meet the needs of many of users around the world. However, all inflow and outflows (diversions) are treated as part of the water-balance, and the priority of diversions is not of major interest in the model.

MASS BALANCE MODELS

The use of mass-balance for water planning has decreased since mass-balance is not capable of assigning priorities to diversions. Historically, water planning models have relied upon mass-balance to simulate reservoir operation. Two of the earliest reservoir simulation models developed were the HEC-3 and HEC-5 models produced by the

U. S. Hydraulic Engineering Center (HEC). The HEC models use mass-balance and flow routing to determine releases from a reservoir or a system of reservoirs. The HEC models are generalized for application to any reservoir system, and have been the prototypes for many models developed for specific river systems.

Sigvaldason (1976) describes the development of the ACRES model, (based on HEC-3) for modeling the Trent River basin. The ACRES model has no capacity to account for diversions from the river system. Instead it is used strictly to regulate river operation for hydropower generation and navigation. The model is unique because it was the first to incorporate rule curves and storage zones in the same model.

Most reservoir optimization models only use rule curves to simulate reservoir operations. Rule curves are a set of curves that describe how a reservoir is operated given a measurable reservoir parameter (inflow, demand, current storage, etc...).

Many simulation models find it easier to account for reservoir water using zones of storage as developed by the U. S. Corps of Engineers. For storage zones, the reservoir is divided into zones of storage and each storage zone has separate rules for reservoir operation. Each storage zone is also attributed to some function, such as flood storage, conservation storage, and/or a buffer zone. Sigvaldason

(1976) defined additional storage zones, such as minimum pool level, inactive storage, a spill zone, multiple conservation zones, and a rule curve zone.

Another model that uses mass balance principles is the Colorado River Simulation Model (CRSM) (Schuster 1989). CRSM models water availability and salt concentrations and tries to satisfy the many decrees and compacts that regulate the Colorado River. Only the major features of the river are modeled since the Colorado river is such a large system that covers a massive area. Major rivers and the large reservoirs are the main model parameters. Smaller rivers are modeled as inflows into the Colorado River System. The Colorado River Basin is divided into 25 reaches with inflow points, diversion points, and reservoirs. Inflows are used to model headwaters, return flows, and gains or losses. Diversions are used to model demands on the system. Eighty-five years of natural flow data and salt contents for 29 inflow points are the hydrologic data supplied. The model is operated on a monthly time basis.

CALIFORNIA MODELS

There are two main water projects in the state of California, the State Water Project (SWP) and the CVP (Central Valley Project). Both projects remove water from the Sacramento-San Joaquin Delta (DELTA) for use in the southern part of the state, so the projects are worked in cooperation. The SWP is operated by the California Department of Water Resources and the CVP is operated by the USBR. Each has endeavored to model their project separately. This has resulted in numerous models to simulate these two projects.

The first attempt to develop a model involved modifying HEC-3 to handle the complex water sharing agreement between the two agencies at the DELTA. The model assumed a constant demand for the CVP and simulated water available to control structures on the SWP (Chung and Helweg 1985). The model used rule curves and storage zones to model reservoir operations. Demands were given and the model satisfied these demands by using mass balance with no consideration of priority. Storage zones were established comparable to the method used by Sigvaldason (1976) in the ACRES model. A special subroutine was written to model the complex DELTA sharing agreement between the two agencies.

In March 1983, the contract that the SWP had for low-cost power expired (Sabet, et al. 1985). This forced the SWP to optimize its power producing capabilities. The operators of the system developed a program to simulate water deliveries made by the SWP. Network Flow Programming (NFP) was then applied to optimize the release of these deliveries for peak power production and to minimize power use.

NFP consists of representing a river as a system of nodes and links, similar to a piping network. Once the system of nodes and links are established, the user assigns a value or cost to each link. The program then tries to minimize the total cost for the system. Several NFP programs were developed by the Texas Water Board during the late 1970's and early 1980's. NFP requires an enormous amount of data preparation before a simulation is ran.

The operation of the SWP and CVP were incorporated into a single model, the California Department of Water Resources Simulation (DWRSIM) model (Barnes and Chung 1986). This model is based on HEC-3 with many changes incorporated into the program to account for the uniqueness of the system. The model is designed to operate the combined SWP-CVP system on a monthly time basis. The model simulates water supply, recreation, instream-flows, and hydro-electric power generation. The system provides a minimum flow to maintain water quality at the DELTA. The model is structured so that any configuration of reservoirs, diversions, power generating plants, pumping plants, and conveyance facilities can be modeled. Changes in the configuration of the system are accomplished by altering the input data files.

Inflows to the system can come from two sources, historic hydrology from 1922-1978, or a stochastic hydrology model which produces synthetic streamflows.

Reservoirs are defined by elevation versus storage, elevation versus surface area, outlet capacity, and power plant characteristics for power generation. In addition, reservoirs operate to satisfy downstream demands at selected control points. Storage zones are established for each reservoir to assist in assigning priorities for releases. Diversion amounts can remain fixed for each month or vary according to the flow in the system and return flows are restored to the system over a twelve month period. Available water and reservoir storage are accounted for by applying mass-balance to the system.

Chung, et al. (1989) describe how NFP was applied to the DWRSIM model, and Bridgeman, et al. (1988) describe how NFP was applied to the ACRES model. NFP results in more efficient models capable of assigning priorities to system operations. Updating a model for NFP consist of representing the systems by a series of nodes and arcs with "costs" assigned to the arcs. The data preparation is enormous for NFP systems.

In addition to the models discussed above, there are models for simulating or optimizing parts of the SWP. These models may try to model one function of the SWP, or try to improve upon one of the models presented. However, the models presented provide a good representation for modeling the complete CVP and SWP projects.

NETWORK FLOW PROGRAMMING MODELS

Another model that uses NFP is the Central Resource Allocation Model developed for the City of Boulder, Colorado (Brendecke, et al. 1989). The water supply system for the City of Boulder receives water from Boulder Creek and two transbasin diversions. The object of the model is to evaluate additions to the system considering two population projections for the year 2040. The model uses historical data for water rights and water use in the Boulder Creek Basin and the hydrology of Boulder Creek for the years 1950-1985. A model of the Colorado River Basin water rights and hydrology estimates the yield from transbasin diversions to the City of Boulder. The final step of the model is to check the ability of the water treatment facilities and the distribution network to handle the estimated flow capacity needed to supply population projections for the year 2040.

The modeling of water rights and streamflow section of the program estimates the flow in Boulder Creek by applying a mass-balance to the system. Costs are then assigned to the various water rights based on priority, and storage facilities are assigned costs based on priority and desired operating procedures. The model then minimizes total cost for the system. Streamflows are also computed in the system to assess the possibility of exchanges with other water right holders. Once again, assigning costs to each link is very tedious.

The next example of NFP is MODSIM3 which is based on the NFP models developed by the Texas Water Board (Labadie, Bode, and Pineda 1986). The City of Fort Collins, Colorado was used to demonstrate MODSIM3 as a water supply model. The model provides monthly or weekly management guidelines for the entire water supply system. It considers formal water rights and informal water exchanges to formulate an optimal operation plan. Inputs to the program include unregulated stream flow, reservoirs and evaporation losses, demands and water rights, conveyance and seepage losses, return flows, imported water, and transbasin diversions. The model is operated similar to the other NFP models discussed and has the same problems.

WATER ACCOUNTING MODELS

The first true water rights model that accounted for the priority of downstream diversions in reservoir operation was the North Platte River Management Model (NPRMM) developed by the USBR (Wei 1977). With the NPRMM, diversions are recognized as having a priority date that must be satisfied. However, the NPRMM is primarily a reservoir simulation model that accounts for water ownership and evaporation at each reservoir. Only large diversion projects are considered when determining system demand and priorities are assigned by the user. The model is very specific to the North Platte River since a complex set of reservoir operating rules are programmed into the model. Instead of having inflows to the North Platte, the model assigns reach gains based on return flows and inflows from other streams.

In spite of its limitations, the NPRMM introduced some interesting concepts in river management. The NPRMM was the first to store water in the reservoir farthest upstream. The belief is that water can be released from that reservoir to meet any downstream demand. Second, the NPRMM sets an operating range for reservoir storage by declaring a minimum reservoir level and an absolute minimum reservoir level. Finally, water stored in reservoirs is assigned to a project indicating that the project owns the water in stored in the reservoir.

A model that uses many of the concepts introduced by the NPRMM is HYDROSS, also developed by the USBR (1991). HYDROSS could be considered a more improved general version of the NPRMM. HYDROSS is primarily a program for assessing the feasibility of new storage facilities on a river system. Natural stream flows are calculated for each control point on the system, and priorities are assigned to diversions and instream flow rights based on their priority date. Flows to the reservoirs are accomplished after all other rights have been met. The operation of reservoirs are quite complex, considering pool elevation, downstream channel capacity, tailwater elevation, and outlet works capacity to determine releases. Releases can be made for power generation, to meet downstream demands, and to provide flood control. Complex relationships for area-capacity-elevation, flow versus tailwater depth, generator efficiency versus storage and tailwater depth, and seepage versus pool level are possible. Return flows enter the system over a twelve month schedule, and evaporation is removed from the reservoirs. Once all the data is input to the model, an accounting of the water available is performed. However, the programs method of calculating a diversion amount is not practical and assigning priorities is very tedious.

The Texas Water Commission has developed a surface water availability model called TAMUWRAP (Texas A&M University Water Resources Allocation Program) (Wurbs and Walls 1989). TAMUWRAP is a generalized computer model for simulating surface water management under a prior appropriation water rights system. Water systems are represented by the following components; control points, basin hydrology, reservoirs, and water rights. Control points are specified to indicate the location of streamflow data, reservoirs, and water rights. The basin hydrology consists of streamflows and reservoir evaporation rates at each control point. A storage capacity versus water surface area relationship is input for each reservoir for use in evaporation computations. Water rights are represented by the following data: control point location, annual diversion amount, reservoir storage capacity, priority number, type of use, and return flow factor (all but the control point location may be set to zero). The model distributes the annual diversion amount over the twelve months of the year, and the priority number is the date that the water right was adjudicated.

From Wurbs and Walls (1986):

> "The water balance computations proceed by month and, within each month, by water right on a priority basis. The water right diversion amount is diverted as long as streamflow or reservoir storage not yet appropriated by senior water rights, is still available. A shortage occurs if sufficient streamflow and/or storage are not available to supply the water right that month. Permitted reservoir capacity is filled to the extent allowed by available streamflow. Reservoir evaporation is computed by multiplying the computed average water surface area during the month by an inputted net evaporation rate. Return flows are computed as a fraction of diversions. An accounting is maintained of storage levels in each reservoir and streamflow still available at each control point."

The output from the model is used as input into a reservoir optimization model such as HEC-3 or HEC-5.

WIRSOS MODEL

The last model to be discussed is the WIRSOS model (WIRSOS 1985). The WIRSOS model is similar to the TAMUWRAP model. Control points, basin hydrology, reservoirs, and water rights are inputs to the system. However, WIRSOS has a provision for allowing instream flow rights to be modeled and natural water rights are separate from project rights. Project rights are rights that receive all or part of their allocated water from reservoir storage. WIRSOS allows the user to assign inflows to the river system at the headwaters of the river. The model then distributes that water downstream through the system.

The WIRSOS model starts by retrieving information from the input data files. The information read is the run data for the simulation, the station data, the delay tables, the reservoir data, and the evaporation data. Next, the model reads in the first two years of runoff data and calculates the non-project releases from the reservoirs. The water accounting is then handled in the main body of the program.

The main body of the model consists of seven functions performed by separate sections of the program. First, the instream flow rights, the reservoir rights, and the diversion rights are read from their data files, and the priority dates for each right is compared. The program is then directed to the proper section to handle the most senior right, then the next most senior right until all rights have been addressed. Six of the sections perform essentially the same function; each section compares the amount of water needed to satisfy the right with the amount of water available. In the case of project rights, the water right is also subject to the availability of water in the reservoir and the flow capacity out of the reservoir. The final section of the main body distributes return flows from diversions back to the river system over a twelve month period. After the main body of the program, evaporation is removed from reservoir storage, and the final reports of water allocation are written.

CHAPTER III

MODIFICATIONS TO WIRSOS

INTRODUCTION

Meena (1993) compared the WIRSOS planning model with the USBR's HYDROSS model. The comparison revealed that WIRSOS was not as good at modeling reservoir operations as HYDROSS. This was an expected result, since HYDROSS was written to determine the availability of water for reservoirs.

The algorithms of HYDROSS were analyzed in detail for this study to determine how HYDROSS performs reservoir operations. Once sufficient data on HYDROSS and its operation were gathered, its best features (from Wyoming's perspective) were incorporated into WIRSOS. In addition to improving reservoir operation by WIRSOS, the post-model operation of restructuring output files was incorporated into the model and critical data files are checked for proper arrangement for model operation.

Reservoir operations performed by WIRSOS corresponded to Wyoming State Water Law. However, most large reservoirs in Wyoming are not owned or operated by the state, and even fewer are operated exactly according to state law. One of the reasons for modifying WIRSOS was to more precisely simulate the amount of water available in a river system.

Subroutines were written or modified to:

- Allow for a minimum release through the outlet works of a reservoir;
- Allow excess water to be stored in a reservoir after all rights have been satisfied;
- Allow water stored in a reservoir at the end of the water year, to be allocated to a reservoir right;
- Allow a project water right to be tied to a reservoir right;
- Allow a project water right to call on two reservoir;
- Allow for off-channel storage reservoirs; and
- Allow water exchanges between the river and project rights not physically downstream from their associated reservoir.

How these tasks were accomplished is discussed in detail in the following sections.

OPTIONAL INPUT

When modifying a program, its former users should be able to utilize the new version with only minor modifications to existing data bases; therefore, all modifications to WIRSOS are optional, not mandatory, for model operation. However, the inclusion of off-channel storage reservoirs in the WIRSOS model has necessitated the addition of one line to each reservoir data record. At the start of the program (when the user is prompted to name the input files, name the output files, name the run, declare the number of stations where runoff data is input, and declare whether reservoirs are to be modeled), the user will be able to choose which options are used during the current simulation. Except for the reservoir data files, model users will not have to update existing data bases to run the modified program. They also do not need two versions of the program to run old and new models. The model user will have to modify existing data bases if they want existing models to benefit from the modifications made to WIRSOS through this study.

OUTPUT

To be most acceptable to the model users, two WIRSOS output files are modified into an easier to understand format. This was accomplished by executing a program that reads these two files, modifies the data format, and prints the information to a new file. This restructuring of output files is now done internally by WIRSOS reducing the number of post model operations performed by the user. The result is easier model operation.

DATA CHECKING

WIRSOS is sensitive to the order in which data is entered into the model. The diversion rights, instream flow rights, junior project rights, and reservoir rights data files must be sorted by priority data and station number. For these data bases, the oldest right is entered first then the second oldest right. This is repeated until the most junior right has been entered into the data file. When two rights have the same priority date, the right furthest upstream is entered first. The station data file is also sorted by station number, the station furthest upstream is entered first then the next furthest. This is repeated until the most downstream station has been entered. Because the way these files are arranged is critical to model operation, the model checks the arrangement of these files prior to processing the information contained in the files.

RESERVOIR OPERATION

The principal difference between WIRSOS and HYDROSS is the method in which priority dates assigned to reservoirs and the amount of water a reservoir can divert. In WIRSOS, each reservoir has a priority date and the maximum amount of water that can be stored by a reservoir in a year is assigned similar to other water rights. HYDROSS also assigns a priority to reservoirs. The priority assigned to each reservoir is after all other water rights have been satisfied. In addition, the amount that is diverted to a reservoir in HYDROSS is any water not used by diversions or

Instream Flow Rights (IFR) during the current month. The first method simulates how Wyoming State Water Law declares a reservoir should be operated; and the second represents how reservoirs are often operated. WIRSOS had to allow both the diversion of water to a reservoir to satisfy the legal water right and the storage of extreme water in reservoirs after Wyoming State Water Law had been satisfied.

WIRSOS is already equipped to handle legal diversions to reservoirs and a new subroutine was added to divert an extreme supply of water (water not needed by any other right on the river system) to reservoir storage. The subroutine is called after all water rights, including IFR's, are satisfied. The program then checks the flow at each reservoir and diverts any available water to storage. The subroutine does not comply with Wyoming State Water Law. It allows a reservoir to fill more than once during a water year and to retain more than its maximum permitted right during that year. This modification was necessary to allow modeling of reservoir operations on Wyoming rivers where this type of procedure can occur.

A drawback to allowing excess water to be stored in reservoirs is that a river can theoretically be drained, negatively impacting tourism, sport fishing, and public opinion. Therefore, the USBR and other reservoir operators often release a minimum amount of water through the outlet works to augment the flow below a reservoir. This water in general is used to preserve the esthetics of the river and support fish populations.

To model a reservoir, one of the new input variables to WIRSOS is a minimum release from a reservoir. This release can be any combination of project water, natural water, and IFR's. If the minimum release is not met and the release is less than the flow into the reservoir, the reservoir releases water to maintain the minimum release. The water is counted as a non-project release. The model user can set this value for each reservoir a lower minimum release can be set to reduce the amount of water released from storage.

OFF-CHANNEL STORAGE RESERVOIRS

A reservoir that is not on the main stream is considered an off-channel storage reservoir. For an off-channel storage reservoir, water is diverted to the reservoir at a station upstream from the reservoir station. This water is returned at another station downstream from the diversion station or to a station another stream. The size of the diversion becomes the limiting factor in deciding how much water is diverted to an off-channel storage reservoir. WIRSOS did not allow off-channel storage reservoirs; it returned water diverted to a reservoir at the point of diversion. To model an off-channel storage reservoir, the user had to use false diversions and elaborate delay tables (a very difficult and often inaccurate procedure). As an improvement to the operation of WIRSOS, the amount of water diverted to an off-channel storage reservoir is limited to the size of the diversion canal.

RESERVOIR RIGHTS

One area that caused WIRSOS users considerable frustration was the method used to allocate water on a multiple right reservoir. Water stored in a reservoir in a year and held in storage until the next year was not assigned to a reservoir right. This allowed an early priority reservoir right to fill its full allocation every year. In actuality, the water stored by a given reservoir right, minus the water used from that right during the year, is the amount of water that is allocated to the reservoir right for the next year. This method was programmed into WIRSOS to assign water stored in a reservoir to a reservoir right. The output from WIRSOS was also modified to give an account of the status of the rights in each reservoir at the end of every year and to give the model user an idea of what was the limiting factor for reservoir storage at the reservoir right's priority date.

Allocating stored water to each reservoir right created an additional problem with WIRSOS. Project water rights in WIRSOS were assigned to a reservoir not a reservoir right. Therefore, if there was water in the reservoir, the project water right is receives this water regardless of ownership. WIRSOS has been programmed to allow the user to assign a project water right to a reservoir right. If project water rights are not tied to a reservoir right, WIRSOS will subtract water release from a reservoir from the reservoir rights. The water is first removed from the oldest reservoir right and then from the junior reservoir rights.

A related problem was that a project water right was only allowed to call on one reservoir to fulfill its requirement. However, there are instances where a project water right is allowed to call on more than one reservoir to satisfy its demand. To meet this requirement, WIRSOS has been modified to allow a project water right to call on two reservoirs.

WATER EXCHANGES

A water exchange occurs when a project right is not physically downstream from its associated reservoir. A water exchange is when a project right removes water from the stream or river and that water is replaced by water from the associated reservoir. WIRSOS has been modified to allow water exchanges for senior project rights.

CONCLUSION

The WIRSOS model was developed to assess the water rights disputes on the Wind River which passes through the Wind River Indian Reservation. Since this dispute was a legal battle, the model was structured to approximate Wyoming water law. The program provided a model for water planning. With the modifications made in this study to WIRSOS, it can became an excellent tool for accurate water planning on any stream system in Wyoming.

CHAPTER IV

DESCRIPTION OF WIRSOS

This chapter contains a description of how the WIRSOS model operates and how the model functions were separated into individual subroutines. Each section of the program is discussed and the function each subroutine performs is described.

INTRODUCTION

SEPARATION INTO SUBROUTINES

When this project started, WIRSOS consisted of a single 3200 line program with two small subroutines and two functions. The 3200 line format, combined with the complex nature of the model, made the program difficult to understand. For example, if a section of the program was changed, great care had to be taken to ensure that the whole program was not affected; and foresight was necessary to confirm that the proper section was being changed. In short, the program was difficult to understand and even more difficult to modify.

The problem was reduced by dividing the program into separate, interactive subroutines that perform the functions of the original program. The purpose of each section of the program is easier to understand, and the effect of changing a section of the program (on the rest of the model) is easier to determine. The model is more efficient since it no longer has to search the entire program to find the section it needs. Instead it calls the proper subroutine for each function performed. Splitting the program into subroutines also allowed the programmer to become familiar with all aspects of the WIRSOS model before altering the program. The modular format created by the subroutines also facilitated the addition of new subroutines when enhancing the model.

The original program was separated into a main program and twenty-six subroutines. With the two subroutines and two functions defined by the original program, the model had thirty-one separate modules. Later, twenty additional subroutines were developed to enhance the program, so that the final model now consists of fifty-one separate modules. The function of each module will be discussed in this chapter, and the discussion of the modules will combine to describe how the WIRSOS model operates.

While separating the original program into subroutines and in all further development, attempts were made to reduce the number of GO TO statements by replacing them with IF-THEN-ELSE statements. Additional documentation (comment lines) were also placed in the Fortran code to make the program easier to understand. To make the program easier to modify, the lines of code between a DO statement and its CONTINUE statement were indented to make it easier to recognize a DO-loop. Lines of code between an IF statement and its ENDIF or ELSE statements were also indented to separate an IF block from the rest of the program. This should make the program easier to understand when making any modifications in the future.

PREVIOUS MODIFICATIONS

WIRSOS has been in use for over ten years, and during that time, the program has been modified to fit the different needs of the various modelers. It was necessary to study these modifications and attempt to incorporate some of the better changes into the current model. As discussed above, considerable work was done to separate WIRSOS into subroutines. Whenever possible, the modifications to WIRSOS were added as a separate subroutine. If the modification was not significant enough to form its own subroutine, existing subroutines that handle the same information were altered to include the modification. Every change made by previous model users was not incorporated into the current model. When two or more model users made similar changes, the changes were combined to form a single modification, or the modification that performed the function best was added.

CURRENT MODIFICATIONS

All modifications to WIRSOS were developed as separate subroutines whenever possible. A considerable effort was made to not change existing input data file formats, or alter them as little as possible.

WIRSOS5.FC

WIRSOS5.FC declares all the variables that are used by the model. Variables that are arrays, that are integers but do not begin with a proper integer character, that are real but do not begin with a proper real character, that represent character sequences, and that are transferred between the main program and the subroutines are declared in the WIRSOS5.FC. The variables that are transferred between the main program and the subroutines are declared in common statements. This allows the computer to assign a location in its memory to these variables, and to access that location from any subroutine that has the same common statement. The common statement saves computer memory and reduces the size of the call statements needed to access a subroutine. WIRSOS5.FC is then included in the main program and most of the subroutines. By including WIRSOS5.FC, the size of each subroutine is reduced.

WIRSOS5

INTRODUCTION

WIRSOS5 is the main program; it is the framework from which all functions are performed. It directs the flow of the program and keeps track of where the model is at any time.

All the variables that are passed between the subprograms are initialized in WIRSOS5.FC. The number of days in a month are assigned for each month and entered into an array. The names of the months are also entered into an array and the value used to convert cfs to acre-feet is set.

After the variables are initialized, the program enters an interactive user interface. The user interface allows the model user to declare the input file names, the output file names, and the run control information for the model. The main program then opens the output files used to store information produced during a model run. The information written to the output files is discussed later.

The program is written in FORTRAN 77 and has been compiled with Microsoft Fortran Version 5.0. All system calls are compatible with this compiler.

TIME AND DATE

The original WIRSOS program contained a function that read the time and date from the computer operating system. This function was system dependent and did not work after the program was written for use on personal computers (PC). hen operating WIRSOS, some model users have modified the original PC version of the program to call the time and date from their computer operating system. Once the time and date were read, they were written to the output files with the rest of the header information. The current model calls the time and date, and writes that information to the output files.

A header is written to the output files declaring the time, the date, and what information each file contains. The block of programming that writes the header, the time, and the date to the output files is located in HEAD. HEAD is called by the main program after the output files are opened.

INPUT DATA

The subroutines that read data from the data input files are called by the main program. The data in these files are used, but not changed by the model. The subroutines called include DELAY, STATION, RESDATA, and RESAREA, each of which will be defined later in this chapter. The subroutine that determines the percent of the permitted reservoir volume that each reservoir right controls (RRBALNC) is also called. The main program then opens the large data files associated with instream flow, diversion, and reservoir rights.

The default names for the WIRSOS input data files begin with the prefix 'Inp' followed by the file number. The file numbers are: '1' for the stations data; '2' for runoff data; '3' for instream flow rights data; '4' for diversion rights data; '7' for delay tables data; '14' for area versus capacity data; '15' for reservoir data; '16' for reservoir rights data; '17' for junior project rights data; '20' for efficiency table data; and '21' for run control information for the interactive user interface.

The principal operation of the model begins when the subroutine FLOW is called. FLOW reads the runoff data from the runoff data file (Inp 2). Runoff data must be entered for each year simulated by the model and two years of data are active in the model during a simulation year. The model needs two years of data to be able to add return flows to the system for up to twelve months after the time of diversion. FLOW reads the second year of data and allows the main program to transfer the second year data to the first year array. The subroutine FLOW can then be called again without erasing the first years data.
After each year is simulated the model goes to the beginning of the next year and transfers the data in the second year array to the first year array before reading the next year of runoff data. Therefore, a year of flow data is not inadvertently erased before the it is used by the model. Before WIRSOS was rewritten, the program had

separate sections to read in the first and second years of runoff data.

DATA CHECKING

After the input data has been entered and the rights data bases opened, the diversion rights, instream flow rights, junior project rights, reservoir rights, and station data files are checked to determine if the data was entered in the proper sequence. The station data file is checked to make sure upstream stations were entered before downstream stations. The others data files were checked to make sure senior rights were entered before junior rights and that upstream rights are entered before downstream rights for rights that have the same priority date. These files are checked by PRICHK, IFRCHK, RSVCHK, and STACHK. In addition, the junior project rights file is checked by JPRCHK to make sure all the JPR's are physically downstream from their associated reservoir.

MODEL PROGRESS REPORTING

The main program next enters the loop where the primary functions are performed. The program preforms water accounting for a river system on a monthly basis, hence the main loop counts from one to twelve to correspond to the months of the year. At the start of each month of simulation (the beginning of the main loop), the program writes the month and year of model operation to the screen. In this manner the modeler knows how far the model has progressed during the current simulation. Figure 1 is a simplified flow chart of the WIRSOS model (See Appendix A for a list of where more complete information can be obtained).

MAIN LOOP

After the loop is initiated, the program writes a header to the main output file, sets water available for diversion equal to the flow in the river, sets the number of rights not met to zero, and calls RESRLSE. RESRLSE calculates the power and non-project releases from reservoirs. The main program then calls the subroutines ISFLWR, RESRITE, and DIVERS to read the rights from the instream flow, reservoir, and diversion files. After the rights are read, the main program compares their priority dates. Priority is established according to when a right was filed. The right with the earliest filing date has the highest priority. In the case of two rights with the same priority date, diversions have the highest priority, reservoirs have the second highest priority, and instream flows have the lowest priority. If two rights are the same type of right and have the same priority date, the upstream right will be processed first.

Depending on which right is in priority, the main program calls one of three subroutines to assess the availability of water to satisfy the right. These subroutines are ISFLWA, RESADJS, and DIVEVAL. In the case of project rights that call on a reservoir to satisfy all or part of their demand, the main program can call JPREVAL, JPRIVER, and PROJRITE depending on the status of the reservoir and the type of project right. Each program that makes a diversion through a canal calls RETFLW to calculate the flows returning to the river system from a diversion. The model continues processing water rights until all rights have been properly addressed. Figure 2 is a simplified flow chart for the main loop of the program.

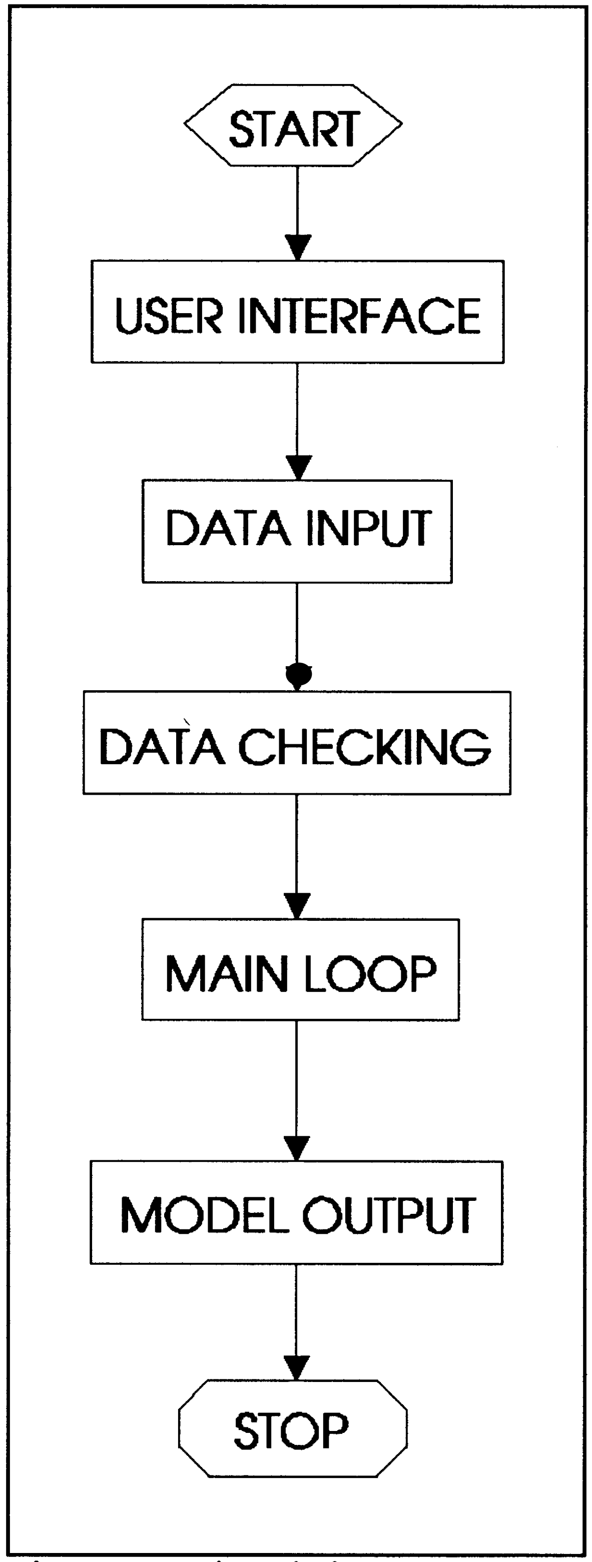

Figure 1 Simplified flow chart for WIRSOS5.

Depending on the options the model user selects, the program can call RESMIN to determine the release required to

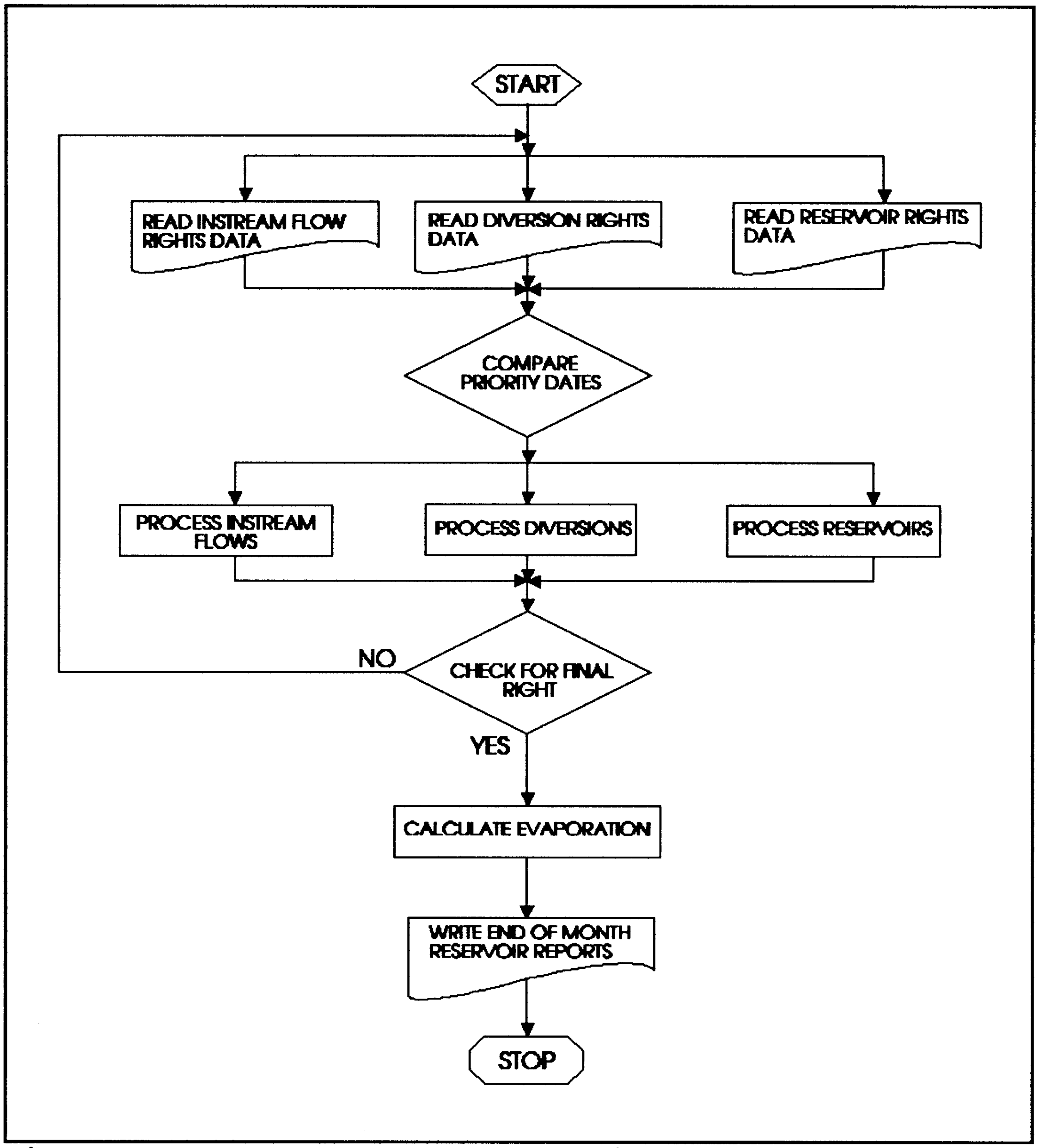

Figure 2 Flow chart of WIRSOS5 main loop.

meet the minimum release from a reservoir, MINRSRL to release water from a reservoir to meet the minimum release, and OVERSUP to store an extreme supply of water in the reservoirs. Before leaving the main loop, the main program calls EVAPR to calculate an average evaporation from the reservoirs for that month and EOM to write the end of the month reservoir data to temporary output files.

MODEL REPORTING

After the main loop, the main program calls EOY to write the end of the year flow data to the output files. If the final year or data was processed, the program calls TWELVE to sort the rights called out files (Tape 12 and Tape 13) by permit number, month, and year, EOR to write the end of the model run reservoir status to an output file (Tape 18), and TOR to write end of model run reservoir rights status to an output file (Tape 19). (When a right is called out, it means that the demand for that right was not satisfied.) After TOR, the model run is over and the program stops.

OUTPUT FILES

Tape 5

Tape 5 is an "echo" of the information provided by the model user during the interactive user interface. This file can later be used as the batch file (Inp 21) that contains the information needed to run the model.

Tape 6

Tape 6 provides an "echo" of the Input data files if the Input data is being "debugged". The information "echoed" is the delay tables, the efficiency tables, the station data, the reservoir data, the reservoir area data, the "virgin" flow data, and the diversion data if JPRIVER is called.

<u>Tape 8</u>

Tape 8 contains the "virgin" flow in cfs at every station. The flows are reported by month for every year simulated.

<u>Tape 9</u>

Tape 9 contains the flow in cfs at every station after all the diversions have been made. The flows are reported by month for every year simulated.

<u>Tape 10</u>

Tape 10 contains the water available in cfs for diversion at every station after all the diversions have been made. The flows are reported by month for every year simulated.

<u>Tape 11</u>

The information reported to Tape 11 is more complicated than the information reported to the other output files. Tape 11 accepts a label declaring the type of right called out and whether the right was fully or partially called out. Table I lists the labels and offers a brief description of what each label indicates. After the label, the station, the permit, the priority date, the percent called out, and the station name where the right is located are written to Tape 11. Finally, there are several messages that can be written to Tape 11 depending on the reason the right was called out. Table II lists the call-out messages, the variables used by the messages, and the situation for which each message would be written to Tape 11.

Tapes 12 and 13

The information written to Tapes 12 and 13 includes: the number of months in the simulation since the program started; the permit number of the right called out (diversion rights for Tape 12 and instream flow rights for Tape 13); the percent of the right called out (calculated by dividing the amount not met by the amount required); and the station where the right is located.

Tapes 12A and 13A

Tapes 12A and 13A are Tapes 12 and 13 sorted by permit number, month, and year.

HEAD

HEAD is a subroutine that writes the header, time, and date for each run to the output files. These files include Tape 8, Tape 9, Tape 10, Tape 11, Tape 18, Tape 19, Tape 12A, and Tape 13A.

REPEATED PROGRAMS

Repeated programs are subroutines that are called from other subroutines, or functions that are performed more than once.

Table I. Tape 11 Call-Out Labels

LABEL	EXPLANATION
IFR NOT MET	INSTREAM FLOW RIGHT NOT MET FROM RIVER
NO DIVERSION	NO DIVERSION POSSIBLE FROM RIVER
PART DIVERSN	DIVERSION PARTIALLY MET FROM RIVER
NO PROJ RIV	NO SENIOR PROJECT RIGHT POSSIBLE FROM THE RIVER DURING A SPILL CONDITION
PART PROJ RIV	PARTIAL SENIOR PROJECT RIGHT FROM THE RIVER DURING A SPILL CONDITION
NO PROJ RES	NO SENIOR PROJECT RIGHT POSSIBLE FROM RESERVOIR
PART PROJ RES	PARTIAL SENIOR PROJECT RIGHT FROM RESERVOIR STORAGE
NO RES STOR	NO STORAGE IN THE RESERVOIR FROM THE RIVER
PART RES STOR	PARTIAL STORAGE FOR THE RESERVOIR RIGHT
NO JPR NOSP	NO JUNIOR PROJECT RIGHT MET FROM RESERVOIR WHEN THE RESERVOIR IS NOT SPILLING
PART JPR NOSP	JUNIOR PROJECT RIGHT PARTIALLY MET FROM STORAGE WHEN THE RESERVOIR IS NOT SPILLING
NO JPR RIV	JUNIOR PROJECT RIGHT NOT MET FROM THE RIVER WHEN THE RESERVOIR IS NOT SPILLING
PART JPR RIV	JUNIOR PROJECT RIGHT PARTIALLY MET FROM RIVER WHEN THE RESERVOIR IS NOT SPILLING
NO PJ 2ND RES	NO SENIOR PROJECT RIGHT POSSIBLE FROM THE SECOND RESERVOIR
PT PJ 2ND RES	PARTIAL SENIOR PROJECT RIGHT POSSIBLE FROM THE SECOND RESERVOIR
NO 2ND JPR NS	NO JUNIOR PROJECT RIGHT MET FROM SECOND RESERVOIR WHEN THE RESERVOIR IS NOT SPILLING
PT 2ND JPR NS	PARTIAL JUNIOR PROJECT RIGHT FROM SECOND RESERVOIR WHEN THE RESERVOIR IS NOT SPILLING
WT X LIM PJ 1	A WATER EXCHANGE WITH A PROJECT RIGHT NOT PHYSICALLY DOWNSTREAM FROM ITS RESERVOIR WAS LIMITED
WT X LIM PJ 2	A WATER EXCHANGE WITH A PROJECT RIGHT NOT PHYSICALLY DOWNSTREAM FROM ITS SECOND RESERVOIR WAS LIMITED

Table II. Tape 11 Call-Out Messages

[AMOUNT] REQ [AMOUNT] AVAIL AT STA [STATION]

WHEN A RIGHT IS PARTIALLY SATISFIED WITH FLOW FROM THE RIVER, IT DECLARES THE AMOUNT REQUIRED BY THE RIGHT AND THE AMOUNT AVAILABLE FOR DIVERSION AND THE STATION AT WHICH THE FLOW IS AVAILABLE.

[AMOUNT] REQ [AMOUNT] AVAIL AT RES [RESERVOIR]

WHEN A PROJECT RIGHT IS PARTIALLY SATISFIED BY RESERVOIR STORAGE, IT DECLARES THE AMOUNT REQUIRED BY THE PROJECT RIGHT AND THE AMOUNT AVAILABLE FROM THE RESERVOIR AND THE CODE THAT IDENTIFIES THE RESERVOIR CALLED.

[AMOUNT] REQ [AMOUNT] RSRIT AVAIL [RESERVOIR]-[RIGHT]

WHEN A PROJECT RIGHT IS PARTIALLY SATISFIED BY RESERVOIR RIGHT STORAGE, IT DECLARES THE AMOUNT REQUIRED BY THE PROJECT RIGHT AND THE AMOUNT AVAILABLE FROM THE RESERVOIR RIGHT AND THE CODE THAT IDENTIFIES THE RESERVOIR AND THE RESERVOIR RIGHT CALLED.

[AMOUNT] REQ OUTFLOW [AMOUNT] OUTFLOW RES [RESERVOIR]

WHEN THE REMAINING OUTLET CAPACITY OF A RESERVOIR IS LESS THAN THE AMOUNT REQUIRED TO SATISFY A PROJECT RIGHT, IT DECLARES THE AMOUNT REQUIRED THE FLOW AVAILABLE AND THE CODE THAT IDENTIFIES THE RESERVOIR CALLED.

[AMOUNT] REQ OUTFLOW AT MAX - RES [RESERVOIR]

WHEN THE FLOW CAPACITY OF A RESERVOIR IS AT ITS MAXIMUM AND THE RESERVOIR CAN NOT RELEASE WATER TO SATISFY A PROJECT RIGHT, IT DECLARES THE AMOUNT REQUIRED AND THE CODE THAT IDENTIFIES THE RESERVOIR CALLED.

SEN DS [RIGHT] NOT FULLY MET AT [STATION]

WHEN A SENIOR DOWNSTREAM RIGHT HAS NOT BEEN FULLY SATISFIED, DECLARES THE TYPE OF RIGHT (DIV FOR DIVERSION, IFR FOR INSTREAM FLOW, RES FOR RESERVOIRS) AND THE STATION WHERE THE SENIOR RIGHT IS LOCATED.

FLOW ADJUSTMENTS

Adjusting or checking the flow at every downstream station is done by several sections of the program. To adjust the flow at every station downstream, a program checks the stream order of the next station in the station array. Figure 3 demonstrates station numbers and stream orders. If the stream orders equals the current stream order, the flow in the stream flow is adjusted. Furthermore, if the stream order from the next station is equal to the current stream order minus one, the stream order is set equal to the current stream order minus one and the flow is adjusted at the downstream station. Eventually, the stream order equals one, which is the order for the main stream in the model, and the flow will only be adjusted on the main branch. In this manner, WIRSOS only adjusts the flow at stations physically downstream from the current station (i.e. those with equal or lesser stream orders).

CHECK

CHECK is a subroutine that was written to check if any downstream rights have not been fully satisfied. CHECK is called from RESADJS, DIVEVAL, JPRIVER, and PROJRITE. If a senior instream flow, diversion, or reservoir right was not satisfied, the subroutine checks to see if the senior right is downstream from the current right. If the senior right is downstream, a message is written to Tape 11 and a flag is set. When CHECK returns control of the program to the calling subroutine, that subroutine checks the flag set by CHECK. If the flag is true, the calling subroutine increments the counter for rights called out, enters the station for the current right into an array, and exits to

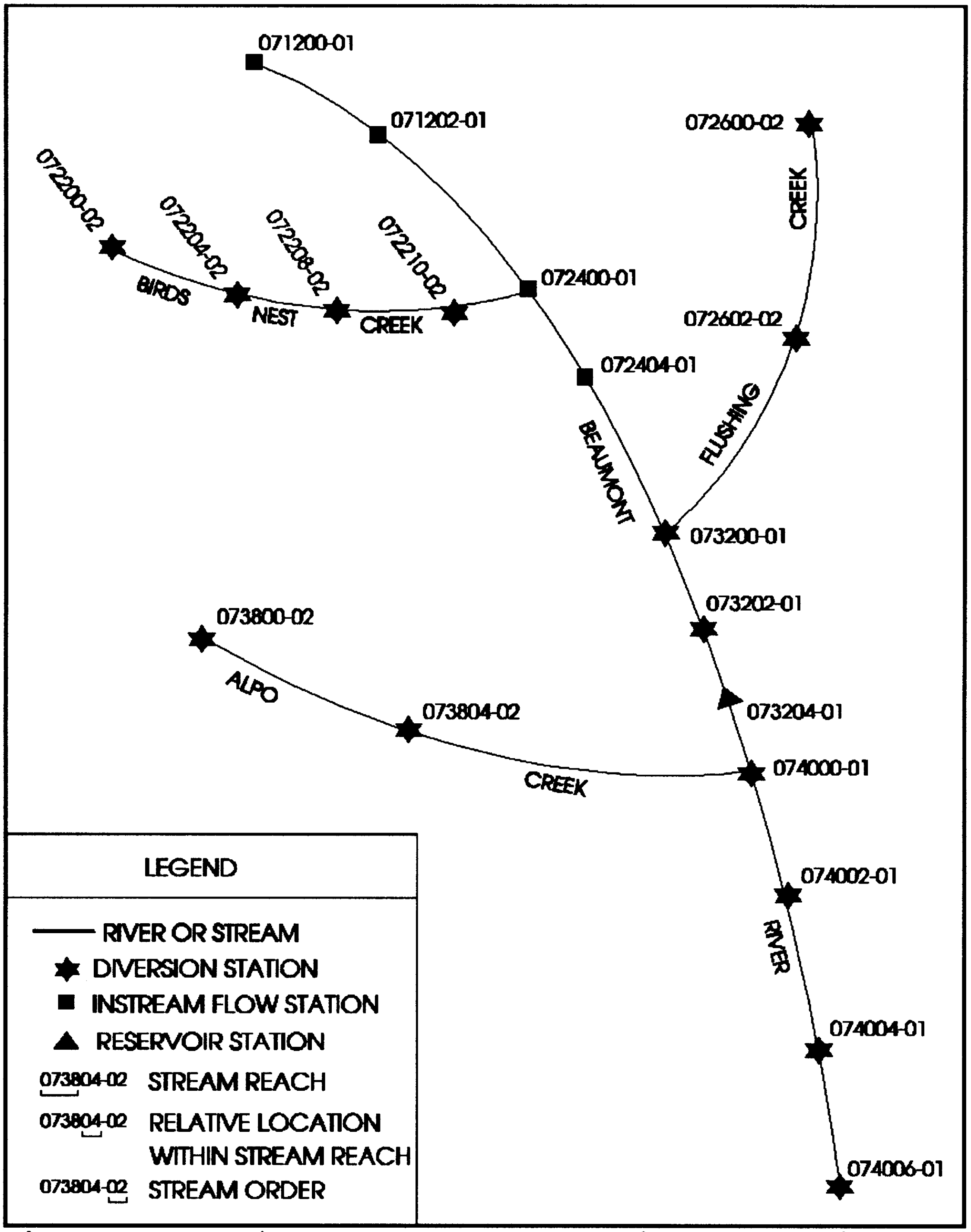

Figure 3 Stataions and stream order river schematic.

the main program since the current right cannot be processed.

RETMON

RETMON is called when a right cannot be met due to water availability. For a given diversion amount, RETMON calculates the return flow and adds that return flow to the available water for the current month. The subroutine then tries to satisfy the requested diversion. If the diversion still can not be met, the amount required is reduced to the smallest amount available downstream from the diversion station. The return flow for this amount is then calculated and added to the water available downstream. The program then tries again to satisfy the diversion. The process is repeated until the amount requested can be satisfied. In this manner, a diversion has the right to its own return flow. Table III shows this process for an imaginary right.

The station where availability controls the diversion amount is defined as the control station to be passed to the calling subroutine and RETMON returns control to the calling program.

RETFLW

RETFLW distributes the return flow according to the type of return flow delay used. The subroutine finds the station and stream order in the station array, and then finds the delay table for the current return flow code. The return flow code indicates how the return flow table is used. The values in the return flow table are the

Table III Example diversion utilizing return flows.

EXAMPLE OF CUMPUTING "NORMAL DIVERSION" AMOUNT
WHEN RIVER FLOW IS GREATER THAN AVAILABLE FLOW *

DIFFER	DIVER (PERMIT #N1961)	AVWRET
		0.0487
0.1513	2.0000	1.8487
0.1312	1.7987	1.6675
0.1113	1.6175	1.5045
0.0968	1.4545	1.3578
0.0821	1.3078	1.2257
0.0689	1.1757	1.1068
0.0570	1.0568	0.9998
0.0463	0.9498	0.9035
0.0366	0.8538	0.8169
0.0280	0.7669	0.7389
0.0202	0.6889	0.6687
0.0130	0.6187	0.6055
0.0068	0.5555	0.5487
0.0012	0.4987	0.4975
-0.0040	0.4475	0.4515

PERMIT #N1961 CAN DIVERT 0.4475 CFS.

DIFFER = DIVER - AVWRET
AVWRET = (AVAIL + RET)
DIVER = DIVERSION AMOUNT, INITIALLY SET TO 2.0 CFS FOR N1961 RIGHT.
AVAIL = MINIMUM DOWNSTREAM FLOW AVAILABLE FOR DEVERSION.
EFFICIENCY = 10%
RET = DIVER * AMOUNT RETURNED IN CURRENT MONTH (100% *(1 - EFFICIENCY)
RET (INITIAL) = (2.0)*(0.9) = 1.8 CFS

* ALSO APPLIES TO OFF-CHANNEL STORAGE RESERVOIRS

percentages of the total return flow that will be distributed each month. The twelve values in each table must sum to equal 100.

A delay code of 0-50 indicates that the return flow is returned to the stream system for the next twelve months starting with the current month as depicted in Figure 4a. The remaining a delay code (51-99) indicate that the first value is the percentage to be returned for the month of January, and the second value is the percentage returned for the month of February. Every column corresponds to a month as depicted in Figure 4b, and the program reads the value for the current month in determining the return flow for that month. The program returns the proper percentage of the return flow for the next eleven months. The return flows are returned to stations downstream from the diversion station, this can include stations on another branch of the river or to a station on another diversion ditch. Figure 5 shows a water right schematic with return flow depicted.

DELAY NUMBER	1	2	3	4	5	6	7	8	9	10	11	12	
11	59	27	4	2	1	1	1	1	1	1	1	1	= 100%

(a)

Figure 4a Non-month specific delay table.

DELAY NUMBER	JAN	FEB	MAR	APR	MAY	JUNE	JULY	AUG	SEPT	OCT	NOV	DEC	
51	1	1	1	1	1	1	13	35	32	11	2	1	= 100%

(b)

Figure 4b Month specific delay table.

RESCHK

There are three factors that can limit the amount of water that a project right can draw from a reservoir. These are the amount of water available from the reservoir, the flow capacity of the reservoir, and the amount of water

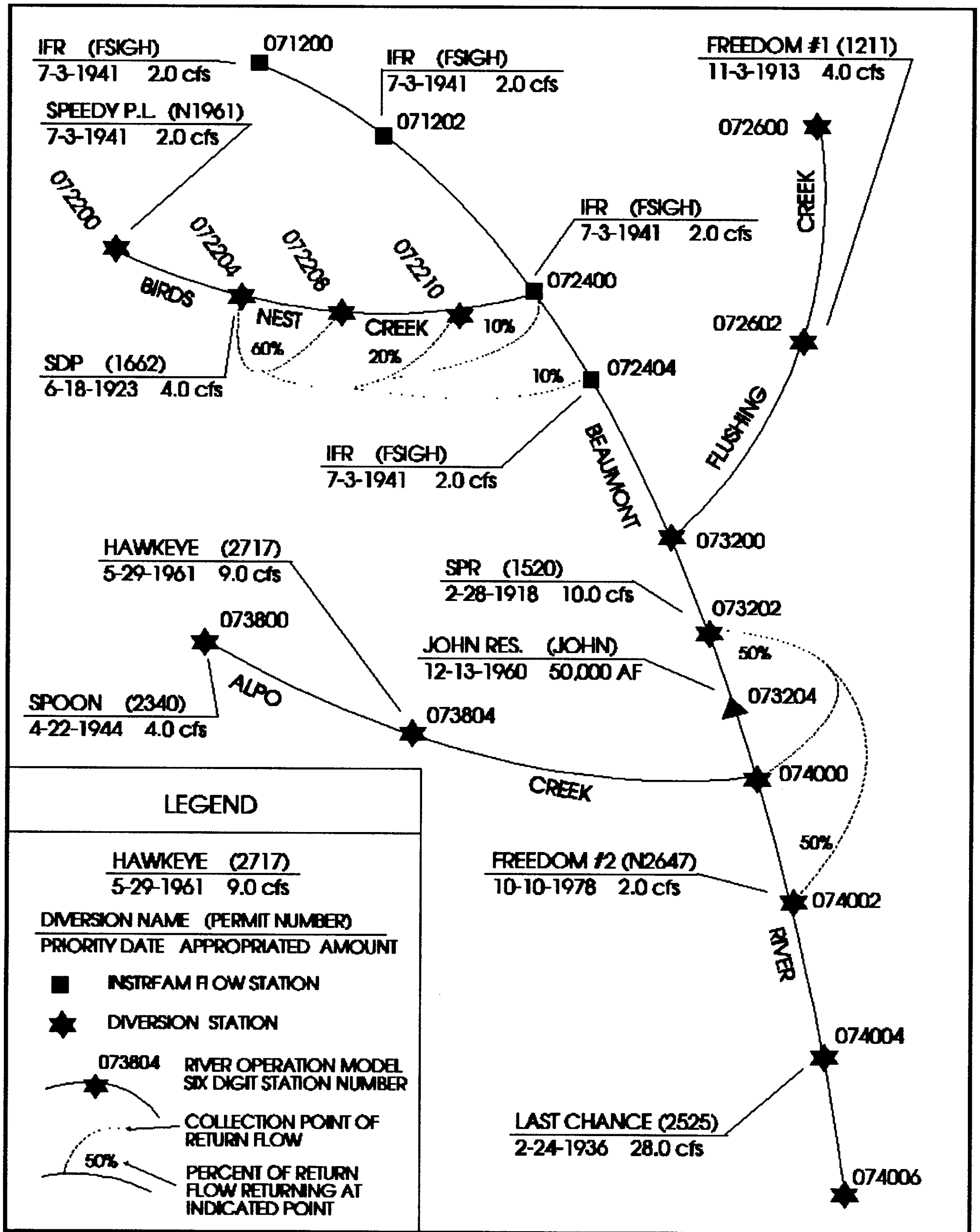

Figure 5 Water rights schematic.

available from a particular reservoir right if the project right is tied to a particular reservoir right. If a project right is tied to a particular reservoir right, that becomes the controlling factor, not the amount available from reservoir storage. RESCHK determines which of the three is the controlling factor and if that factor will limit the amount of water the project right is allowed to withdraw. If the project right is called out, RESCHK writes a message to Tape 11 declaring the type of right called out and the factor that limited the reservoir release. The program also sets a flag that is read by the calling program. This flag indicates to the calling program whether the right was called out, and if it was called out, what reservoir release factor limited the right.

LIMAVL

LIMAVL writes a message to Tape 11 if a diversion, instream flow, or reservoir right is called out because of a lack of water available for diversion. The calling program passes the label declaring the type of right not satisfied, the diversion station, the permit number, the priority date, the percent called out, the diversion station name, the amount requested, the amount available, and the controlling station (the station where the availability is the least) to LIMAVL. Given this information LIMAVL writes a call out message to Tape 11.

PAGE11

PAGE11 writes the column headers to the output file Tape 11.

CVMGM

CVMGM is a function called by the subroutine RESRLSE. CVMGM returns the lesser of two values. The remaining reservoir capacity and the maximum vacant capacity declared in the reservoir data file are the two factors that can limit the amount of water a reservoir can store in a given year. These two values and the quantity of the vacant capacity subtracted from the remaining capacity are passed as arguments to CVMGM. CVMGM initially sets the amount that can be stored for the water year to the vacant capacity. If the vacant capacity subtracted from the remaining capacity is less than zero, the amount that can stored for the water year is set to the remaining capacity.

CVMGT

CVMGT is a function called by RIVDIS and AVLDIS. CVMGT returns one of two values depending on whether the logical argument passed to is true or false. The flow or the availability in the river at a station, the amount to be removed or added at that station, and a logical argument are passed to CVMGT. CVMGT checks the logical argument. If it is true, the current station is physically downstream from the releasing or diverting station. The water is then added or removed from the current station. Otherwise the current station is not downstream from the releasing or diverting station and no adjustment is necessary.

<u>RIVDIS</u>

RIVDIS adds/removes any water discharged/diverted to/from the river flow array. The calling program passes the starting station number, the ending station number, the stream order of the discharge/diversion station, and the amount added/removed to/from the stream. The program then checks to find all downstream stations between the starting station and the ending station, and adjusts the flow at those stations.

<u>AVLDIS</u>

AVLDIS adds/removes any water discharged/diverted to/from an availability array. The calling program passes the availability array, the starting station number, the ending station number, the stream order of the discharge/diversion station, and the amount added/removed to/from the stream. The program then checks to find all downstream stations between the starting station and the ending station, and adjusts the availability at those stations.

<u>RRDIS</u>

If a project right is not tied to a reservoir right, RRDIS removes the water released to satisfy the project right from the most senior reservoir right. However, if the most senior right does not have enough water left to satisfy the project right, RRDIS removes what it can from the most senior right and the rest is removed from the next most senior right. This is repeated until the water has been removed from the rights or until all (up to four) of the reservoir right accounts associated with a reservoir are empty.

<u>SCNDRESV</u>

This subroutine tries to satisfy the remainder of a project right from a second reservoir. The program calculates the water available from reservoir storage, the flow available from the reservoir, and the water available from a reservoir right if the project right is tied to a particular right. RESCHK is then called to determine if the remaining project right can be met. After RESCHK, the amount of the right that was satisfied is determined, the remaining right is set, any water released is added to the project releases for that reservoir, the water released is removed from current reservoir storage, the reservoir rights are adjusted (either according to the right called or by RRDIS), and the water is distributed to all stations between the reservoir and the project right diversion station by RIVDIS. This subroutine is similar to JPREVAL.

USER INTERFACE

With the original WIRSOS model, every Input and output file had a distinct name; and any deviation from these names would prevent the model from running. In addition, the run control input file contained the name or header that described each run, the number of runoff stations, and whether reservoirs were to be modeled. The information in the run control input file was changed for each separate model run. To make the model more flexible, a previous model user created an interactive user interface for the model. This interface is composed of four sections: the interactive user interface run control; the input data file names; the output file names; and the model run information. In addition to adding the interface to WIRSOS, each section was developed into its own subroutine. As more modifications were completed, the amount of information entered from the user interactive interface increased. Figure 6 is a simplified flow chart of the interactive user interface.

Two new features were added to the existing interface. First, the user can create a file (Inp 21) that contains all the user interface information. The model can read the information from this file and the data is not entered from the screen. By answering a series of questions, the user can specify what information is to be read from the file and what information is entered from the screen.

Second, the information requested by the user interface is written to a file (Tape 5). This file can be used to run the model in batch mode.

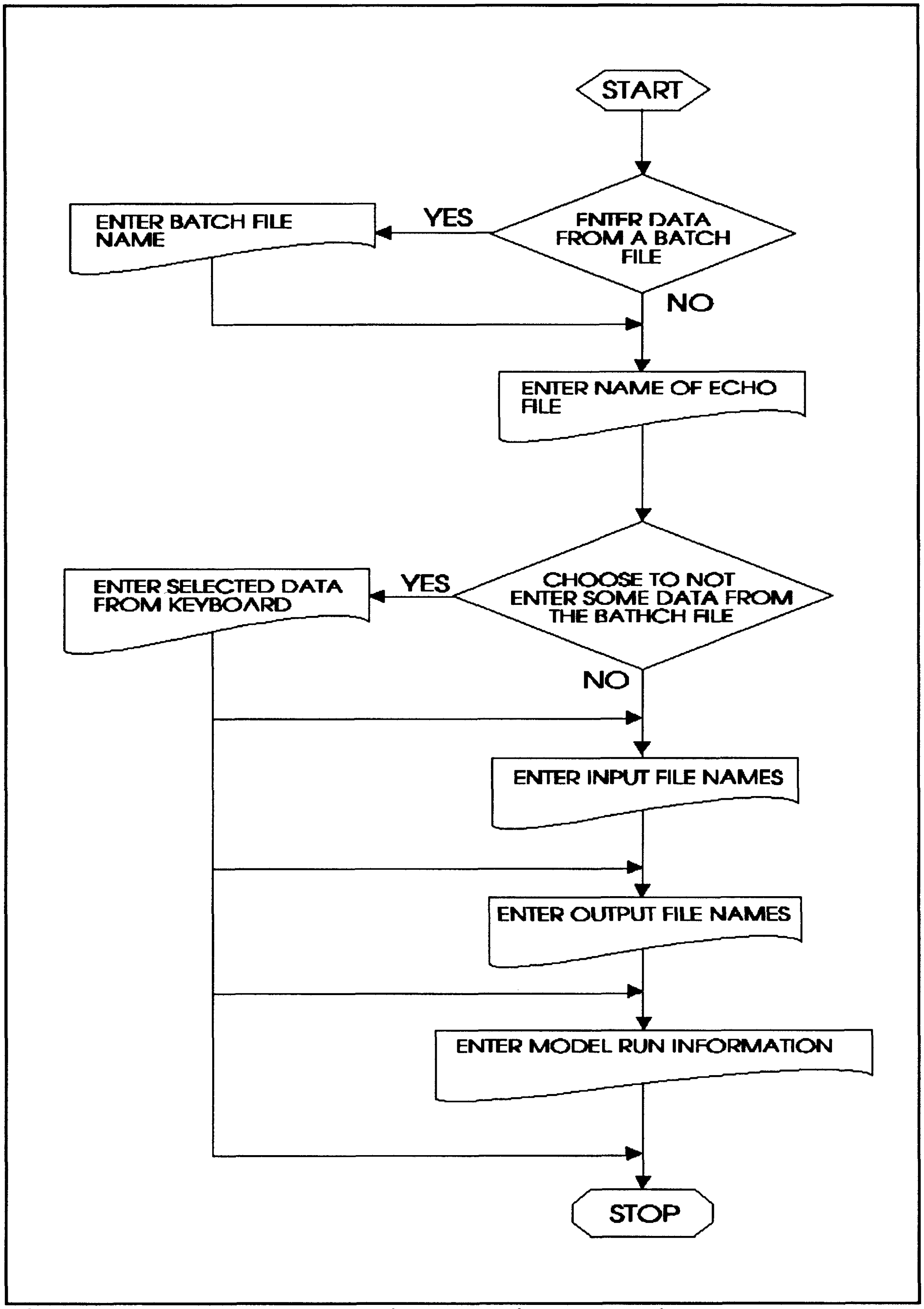

Figure 6 Flow chart of interactive user interface.

An option common to each interactive input section is that a screen is then presented that displays the information that was just entered. The user is given the option of changing any incorrect names. The interface was modified to allow the model user to choose the data that is wrong and to change that data. Before this modification, the model user had to re-enter all the data in a section. The model user can also choose to re-enter all the data by choosing zero. In addition, the modeler must answer all queries with on of the requested responses or the model will re-ask the question.

<u>HEADER</u>

HEADER is the subroutine that controls the interactive user interface. The user is directed to chose how they want to enter the run control data and what to name the echo file (Tape 5). If the model is being run in batch mode HEADER reads the input file names, the output file names, and the run control information from the batch file (Inp 21). The subroutine then directs the modeler to decide which information read from the batch file is used in the current simulation. HEADER next calls the subroutines INPUT, OUTPUT, and MDLRUN. The last option given to the model user is whether to build the data debugging file (Tape 6). If the debugging file is built, the run control information is written to this file.

<u>INPUT and OUTPUT</u>

The default file names for the input data files and the output data files are given in Table IV and Table V. Before the interactive user interface, the input and output data files had to be in the same directory as the WIRSOS program for the model to operate.

The interactive user interface now allows the model user to call the data files anything, and to place those files in any directory on any drive. If the model user is entering the input or output data file names from the screen, they are prompted to enter the names of the input or output data files. However, the model can leave the input or output files named according to the 'Inp' and 'Tape' scheme shown in Figure 7 and Figure 8 since they are the default names used in the program. The input file names are read by the subroutine INPUT and the output file names are read by the subroutine OUTPUT.

EFFICIENCY TABLES. While reading the input file names, the modeler is given the option to enter a file name for the efficiency tables. If an efficiency file is read, a flag is set that tells the rest of the program to use monthly efficiencies.

RIGHTS CALLED OUT MODIFICATION. The modeler is given the option to modify the diversion rights called out file (Tape 12) and the instream flow rights called out file (Tape 13) during the output file name section of the interactive user interface. If the files are to be modified, the modeler is asked for the name of the files created (Tape 12a and Tape 13a by default). Later, after the model has simulated each year of operation, the main program calls the subroutine TWELVE that converts the output file format.

Table IV Default input file names.

INP1	Station data.
INP2	Runoff data.
INP3	Instream flow data.
INP4	Diversion data.
INP7	Delay tables.
INP14	Area versus Capacity data.
INP15	Reservoir data.
INP16	Reservoir rights data.
INP17	Junior project rights data.
INP20	Efficiency tables.

MDLRUN

The interactive user interface reads the header that described the model run, the number of stations where runoff data is entered, the first year to be simulated, whether reservoirs were to be modeled during the simulation, whether the model is to be run on a calendar versus a water year, whether to reset the reservoir rights to zero at the beginning of the water year or to the previous years reservoir rights account, and whether to store an extreme supply of water in the reservoirs. What the user interface does is allow the modeler to enter this information from the screen, or to modify the information if it has been read from a file.

Table V Default output file names.

TAPE8	"Virgin" river flows.
TAPE9	Final river flows.
TAPE10	Water availability in the river.
TAPE11	Rights "called-out" report.
TAPE12	Diversion rights called-out.
TAPE13	Instream flow rights called-out.
TAPE18	Final reservoir report.
TAPE19	Final reservoir rights report.
TAPE12A	Diversion rights called-out sorted by permit and year.
TAPE13A	Instream flow rights called-out sorted by permit and year.

FIRST YEAR OF SIMULATION. Another option that has been added is the ability to define the first year of model simulation. If the user knows that the runoff data represents data beginning in 1945, 1945 is entered as the first year of the model run. The model then prints the year of operation to the output files. Before this option was added, the program printed year 1, year 2, year 3, ..., year N to the output files. If the user does not know the first year of model operation, a value of 1 should be entered to indicate to the program that it is the first year of data. The first year of model simulation is part of the model run information section of the interactive user interface.

CALENDAR YEAR. A problem created when the program was modified by several previous model users, was the type of year used to operate the model. The original program operated on a water year which begins in October. The first month of data in the input files corresponds to October (i.e. October is the first month of the year). All year-to-date storage or diversion amounts are reset to zero and all reservoir flags are reset to false at the first of October. However, in a few of the later versions of WIRSOS, the model was modified to operate on a calendar year (January is the first month of the year). All year-to-date storage, diversion amounts, and reservoir flags are still reset at the first of October, but the first month in the data files correspond to January. This creates a problem since the runoff in January is not the same as October and the diversion amounts are different for January and October. Also, when the program uses the number of days in a month to convert acre-feet to cfs, the flow values will be changed since every month does not have the same number of days.

To eliminate the need to convert all the calendar year files to a water year, the modeler is given the option of choosing the type of year to be used by the model. This option is part of the model run information presented with the interactive user interface. If a modeler wants to run the model on a calendar year, they choose this option at the start of a model run. The days in a month and the month names arrays are then set to represent a calendar year. Otherwise, the two arrays are set to represent a water year.

READING MODEL RUN INPUT DATA

DELAY

The subroutine DELAY opens the delay table Input file (Inp 7) and reads the delay codes and the monthly return flow tables from that file. There can be up to 99 return flow delay types in the delay file, and each delay has a table of twelve values that correspond to the twelve months of the year. Each value is the percentage of the return flow, from a diversion, that is returned to the river system at its return station for a month of the year. The values entered as return flow percentages must sum to 100%.

EFFNCY

Efficiency indicates how much of the water diverted from the river system is used by the water right holder. The portion not used is returned to the system by the subroutine RETFLW as discussed previously. The standard method to determine efficiencies is to enter a single value that is the average efficiency of the water rights. Figure 7 is an example of how crop consumptive use could be used to calculate efficiency and Table VI shows a distribution of losses and return flows.

A few agencies require efficiencies to be determined for each month that the right diverts water. The subroutine EFFNCY was written to read a table of monthly efficiency values from an Input file (Inp 20). This file is organized like the delay table file and the first value read is the efficiency code, and the following twelve values are the monthly efficiency values. There can be up to 99 efficiency tables read from the monthly efficiency file.

To use the monthly efficiency table, the Input data file formats in the diversion data bases do not have to be changed. When the model reads an efficiency value for a diversion, it reads it as a real number. If monthly efficiencies are being used, the program converts the efficiency value to an integer efficiency code. The efficiency code is then checked to see if it corresponds to

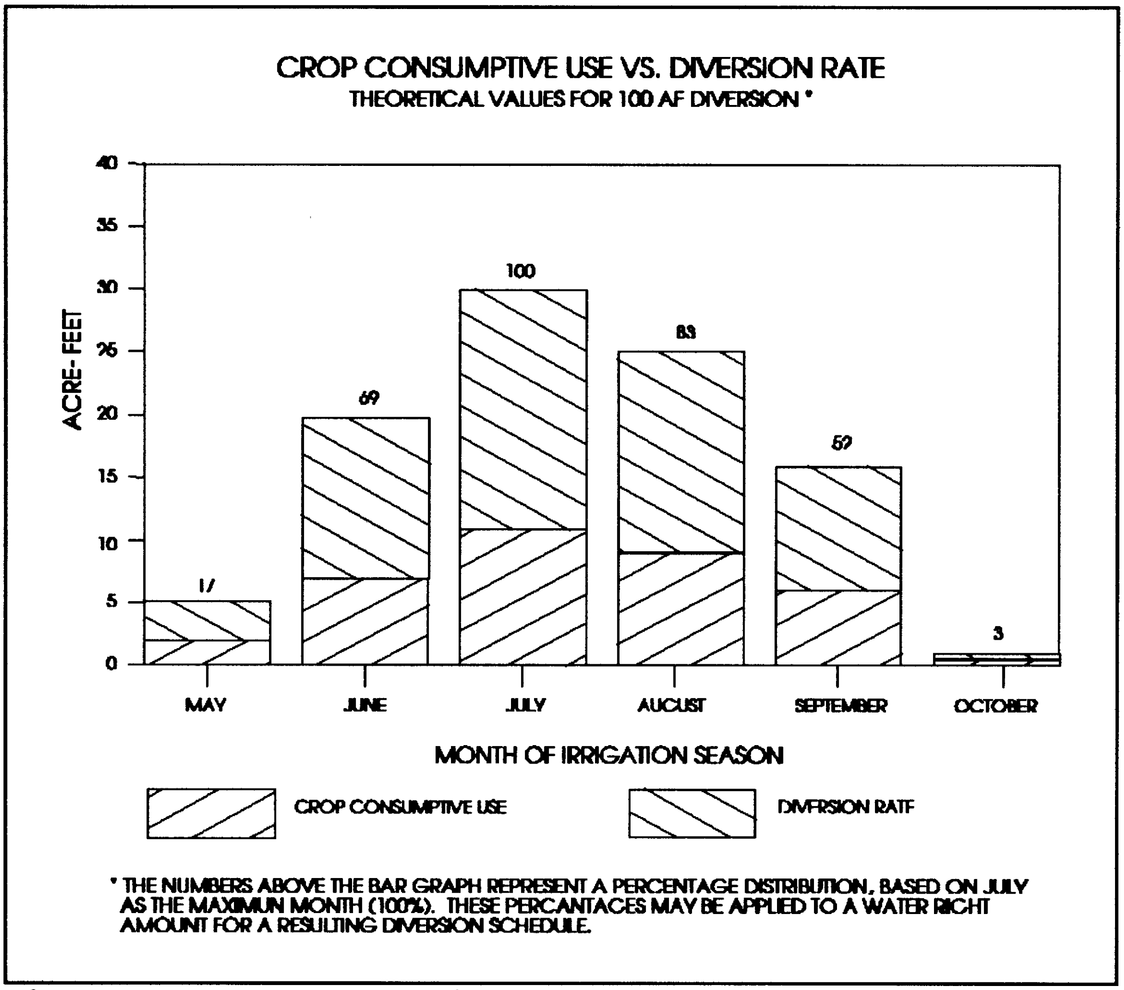

Figure 7 Crop consumptive use vs. diversion rate.

one of the efficiency codes read from the monthly efficiency file. If the value does not correspond to an efficiency code, the program operation is terminated. If the value does correspond to an efficiency code, the program uses the efficiency value that corresponds to the current month to determine the total amount of water returning to the river from the current month's diversion. If monthly efficiencies are not being used, the program uses the efficiency value read to determine the total

DISTRIBUTION OF LOSSES AND RETURN FLOWS

FUNCTION	LOSS EXPRESSED AS A PERCENTAGE OF: HEADGATE DIVERSION	LOSS EXPRESSED AS A PERCENTAGE OF: INDICATED FUNCTION	DISRTIBUTION OF NON-RECOVERABLE LOSSES AND RECOVERABLE RETURN FLOW FOR AN ASSUMED HEADGATE DIVERSION OF 100.00 ACRE FEET — NON-RECOVERABLE: PERCENT	NON-RECOVERABLE: AMOUNT ACRE FT.	RECOVERABLE: PERCENT	RECOVERABLE: AMOUNT ACRE FT.	TOTAL: PERCENT	TOTAL: AMOUNT ACRE FT.
CONVEYANCE	25							
SURFACE		25	30	1.88	70	4.38	100	6.25
SEEPAGE		75	5	0.94	95	17.81	100	18.75
SUB-TOTAL				2.81		22.19		25.00
EVAPOTRANSIPIRATION	38	100	100	38.00	0	0.00	100	38.00
ON FARM APPLICATION	37							
SURFACE		50	15	2.78	85	15.73	100	18.50
PERCOLATION		50	5	0.93	95	17.58	100	18.50
SUB-TOTAL				3.70		33.30		37.00
TOTAL LOSSES				4.65		20.10		24.75
SURFACE				1.86		35.39		37.25
SEEPAGE & PERCOLATION				38.00		0.00		38.00
GRAND TOTAL OF LOSSES				44.51		55.49		100.00

DITCH HEADGATE EFFICIENCY = EVAPOTRANS./TOTAL DIVERSION = 38%
FARM HEADGATE EFFICIENCY = EVAPOTRANS./FARM HG DELIVERY = 51%

Table VI Distribution of losses and return flows.

amount of water returning to the system from the current month's diversion. No modification of the diversion data bases is necessary, except to create a table of monthly efficiency values if monthly efficiencies are used.

STATION

STATION is a subroutine that reads the station numbers, stream orders, and station names/descriptions from the station input file (Inp 1). There can be up to 1000 stations in the station input file. After all the stations have been read into the arrays for station, stream order, and station name, the subroutine sets the number of stations equal to the number of stations read from the input file and each station is assigned a number. This number is used to locate a station or stream order in the station array.

RRBALNC

RRBALNC reads the reservoir rights from the reservoir rights file (Inp 16), and sums the volumes permitted by each right for the reservoir. The volume permitted by each individual right is then divided by total permitted volume. This gives the percent of the total permitted volume that each reservoir right controls. The number of rights associated with each reservoir, the volume controlled by each right, and the total number of reservoir rights are determined in RRBALNC.

RESDATA

RESDATA is a subroutine that reads the reservoir data from the reservoir data file (Inp 15). Each reservoir in this file requires three lines of data. The information read from line one includes the reservoir name, the reservoir station, the reservoir code, the flag that indicates the reservoir is an off-channel storage facility, the flag that indicates whether the reservoir is to be modeled, the minimum reservoir storage volume, the maximum reservoir storage volume, the maximum primary outlet capacity, the initial storage volume, the evaporation rates for twelve months, and the maximum storage that can be diverted during a water year. The information read from line two includes the non-project releases for twelve months, the power release goal month, the power release goal volume, the percent of the beginning volume to assign to each reservoir right, the minimum release from a reservoir, the off-channel storage reservoir diversion station, the off-channel storage reservoir diversion efficiency, the number of off-channel storage reservoir diversion return flow stations, and the off-channel storage reservoir diversion canal capacity. Line three contains the return flow stations, the percent returned at each station, and the delay code of the delay table used at each station.

The non-project releases are the releases made from the reservoir for each month. They are a percent of the water currently available for release from the reservoir. The power release goal volume is the reservoir volume remaining after the power releases are made, and the month set for the program to reach the goal volume is the power goal month.

After reading the data from the Input file, RESDATA sets all flags and initial values on key variables for each reservoir. The flags are the junior project rights processed with the reservoir, the reservoir rights met, the one-fill, and the spill flags. The key variables that are set to zero are the power release for each month, the power required for each month, the index of reservoir rights, the reservoir right year-to-date storage, the monthly storage, the extreme monthly storage, and the total year-to-date storage. The current storage is set to the initial volume. RESDATA then finds the reservoir station in the station array and sets the stream order at the reservoir equal to the stream order of that station. The subroutine finally sets the number of reservoirs equal to the number of the active reservoir with the highest reservoir code.

There are two options for setting the reservoir rights year-to-date during the interactive user interface. First, the model user can choose to set the reservoir rights year-to-date to zero. The reservoir rights year-to-date will then be set to zero at the start of each water year for the rest of the simulation. The second option is to set the reservoir rights year-to-date equal to the previous years reservoir right account.

If the model user chooses to set the reservoir rights year-to-date equal to the previous years account, there are two options for dividing the initial storage among the reservoir rights. First, the modeler can enter up to four values that correspond to the percent of the initial reservoir volume to be assigned to each reservoir right, the sum of these four values must equal 100%. If these values are entered, the program multiplies the value times the beginning volume to establish the initial reservoir rights year-to-date and the initial reservoir rights account. Second, the initial volume can be divided between the reservoir rights according to the percent of the permitted volume that each reservoir right controls (as determined by RRBALNC). This is done by setting the four values for dividing initial storage to zero in the reservoir data input file.

OFF-CHANNEL STORAGE RESERVOIRS. Off-channel storage reservoirs were not handled by the original WIRSOS model. To model an off-channel reservoir, the user had to use diversions and return flows to simulate reservoir operation. To simplify this procedure, the model was modified to allow off-channel storage reservoirs and the subroutine RESDATA was modified to read off-channel reservoir data. RESDATA reads the off-channel storage data from the third line and the end of the second line in the reservoir data input file. RESDATA checks to see if monthly diversion efficiencies are being used during the current run. If monthly diversion efficiencies are being used, the off-channel diversion canal efficiency is converted to an integer efficiency code and the efficiency tables are searched for this value. RESDATA also finds the off-channel storage diversion station for in the station array and sets the stream order at the off-channel diversion station equal to the stream order of that station.

RESAREA

RESAREA reads the area versus capacity information from the reservoir area-capacity input file (Inp 14). The subroutine reads the reservoir code for the reservoir to which the area-capacity information is to be applied. RESAREA then reads the range, the upper capacity limit, the equation type, and the three coefficients used in the area-capacity equations. The range is the number of area-capacity relationships for the reservoir (up to three), and the upper capacity limit is the storage volume in the reservoir at which the program switches to another area-capacity relationship for calculating surface area. The model has a choice of five equations for calculating the surface area of a reservoir based on its current storage capacity, and the equation type directs the program as to which equation to use. The three coefficients are constants that are used in the equations to determine surface area for a given reservoir storage.

FLOW

FLOW reads the runoff data from the runoff data input file (Inp 2) and stores that information in the second year of the river array. As discussed previously, the information is read into the second year array so that the first year of data is not erased, and a separate subroutine is not necessary to read in the initial year of runoff data. Data is then transferred to the first year array as necessary for model operation. The information read is the station where the runoff or flow enters the system and each month's runoff at that station. The subroutine then converts the runoff from acre-feet to cubic feet per second (cfs) and finds the runoff station in the station array. Once the runoff station is found, the stream order of the runoff station is determined. After obtaining the stream order, FLOW adds the runoff to every station downstream from the runoff station. When the data from all the runoff stations has been read and the water distributed through the system, the subroutine writes the year's "virgin" flow in the river to the initial flow report (Tape 8). Figure 8 shows how runoff is distributed through a river system by WIRSOS.

DATA CHECKING

JPRCHK

JPRCHK checks the rights in the junior project rights input data file to determine if all rights are physically downstream from their associated reservoir. The subroutine reads a single junior project right and checks if that right is downstream from its reservoir. A right is downstream if its station number is greater than the reservoirs station number and the stream order of the right is less than or equal to the stream order at the reservoir station. JPRCHK also checks if the right is downstream from the second reservoir if the right calls on two reservoirs. If a junior project right is not downstream from its reservoir an error message is printed and the program stops.

PRICHK, IFRCHK, and RSVCHK

These subroutines determine if the rights in the junior project rights, diversion, instream flow, and reservoir rights data input files have been entered in the proper sequence. These subroutines read two rights from the same data base and compares their priority dates to determine if the first right is older than the second right. They also checks if two rights have the same

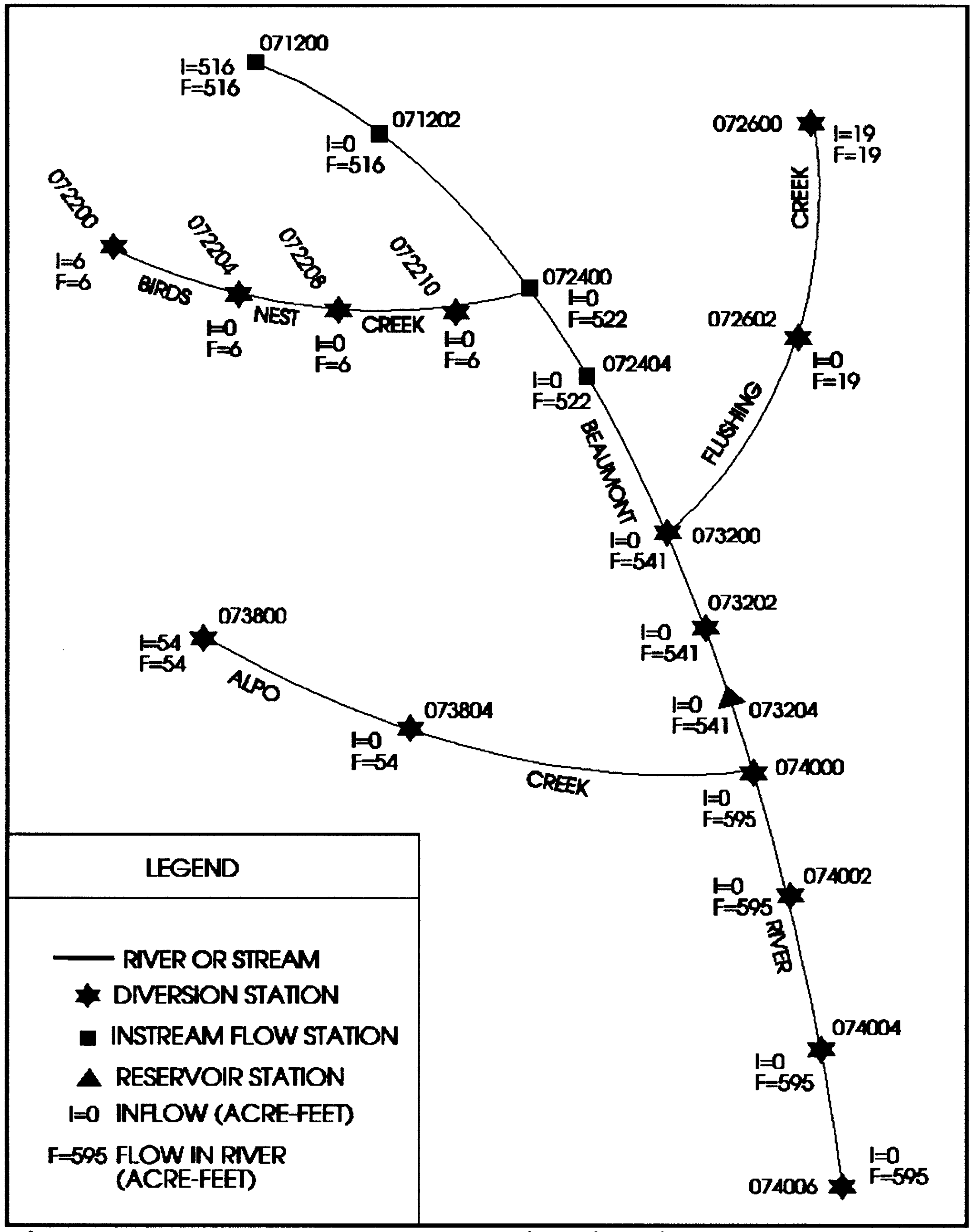

Figure 8 Example stream flow distribution.

priority date. If two rights have the same priority date, the subroutines determine if the upstream right is first in the data base. If the two rights read are fine, the next operation is to read the next right and compare it with the second right already read into the subroutine. If the rights read are not entered in the proper sequence, the subroutines write an error message to the screen and the program stops. PRICHK checks the junior project and diversion data files, IFRCHK checks the instream flow data file, and RSVCHK checks the reservoir rights data file.

STACHK

STACHK determines if the stations were entered correctly in the station data file. Station are entered from the lowest station number (the station furthest upstream) to the highest station number (the station furthest downstream). If the stations are not entered in this order, an error message is printed to the screen and the program operation stops.

MAIN BODY OF THE PROGRAM

RESRLSE

RESRLSE calculates the power and non-project releases from each reservoir. If it is the beginning of the water year, the subroutine also resets the reservoir fill flag and the reservoir spill flag; clears the reservoir rights year-to-date and the total year-to-date storage accounts; and the reservoir rights flag is reset to 'NO'; the reservoir rights index is set to zero or to the previous water years reservoir right's account (depending on the option chosen during the interactive user interface. If the reservoir is being modeled in the current model run, the non-project and power releases are calculated. The power release required for the current month is compared with the outlet flow capacity of the reservoir and the water available from reservoir storage. The water released for power is then equal to the lesser of the following parameters: the power release requested, the outlet flow capacity of the reservoir, or the water available from reservoir storage. The water released for power is then subtracted from reservoir storage and the outlet flow capacity of the reservoir is reduced.

Non-project releases are calculated (non-project releases are a percentage of the available reservoir volume). The non-project releases are then compared with the outlet flow capacity of the reservoir and the available reservoir volume. The lesser of the non-project release requested, the outlet flow capacity of the reservoir, and the available reservoir volume is the amount released to the river system. The power and non-project releases are removed from the current reservoir storage and the amount release is added to the river array at all downstream stations downstream by RIVDIS and AVLDIS. The water released is also removed from the reservoir rights accounts. The amount removed from each reservoir right account is obtained by multiplying the release by the reservoir rights account divided by the current storage. Therefore, the percent of the current reservoir storage that each reservoir right controls is the percent of the release that is removed from each right.

ISFLWR

ISFLWR reads the instream flow rights from the instream flow rights data input file (Inp 3). The information read is the instream flow station, the instream flow permit number, the instream flow priority date, and the instream flow required for each month of the year. For the program to be able to compare the priority dates of each type of right, the priority date is converted from a month-day-year to a year-month-day format. This conversion is done in ISFLWR. If the last right has been read from the instream flow data file, ISFLWR sets the instream flow flag to true. This flag signals the main program not to process any more instream flow rights. The priority date is set to 99999999 after the last right is read. This causes the instream flow rights to have the lowest priority possible in the system so no other right will be injured.

RESRITE

RESRITE reads the reservoir rights from the reservoir rights data input file (Inp 16). The information read is the reservoir station, the reservoir permit number, the reservoir priority date, the reservoir code, the storage volume permitted, and whether it is the last right associated with this reservoir. The subroutine then converts the priority date from a month-day-year to a year-month-day format to make it possible to compare the date with the priority date from the instream flow and diversion rights. If the last right has been read from the reservoir right data file, RESRITE sets the reservoir rights flag to true. This flag signals the main program not to process any more reservoir rights. The priority date is set to 99999999 after the last right is read. This causes the reservoir rights to have the lowest priority possible in the system so no other right will be injured..

DIVERS

DIVERS reads the diversion rights from the diversion rights data input file (Inp 4). The information read is the diversion flag, the reservoir right from which the diversion draws, the diversion station, the diversion type, the diversion efficiency, the diversion permit number, the diversion priority date, the number of return flow stations, the diversion amount required for each month of the year, the return stations, the percent of the total return amount to return at that station, the return delay code, the flag that indicates whether water is drawn from a second reservoir, the second reservoirs code number, and the reservoir right from which to draw the water. The program then checks to see if monthly diversion efficiencies are being used. If monthly diversion efficiencies are being used, the diversion efficiency is converted to an integer efficiency code and the efficiency tables are searched for this value. If the efficiency code is not found in the efficiency tables an error message is written to the screen and program operation is stopped. The subroutine then converts the priority date from a month-day-year to a year-month-day format to make it possible to compare the date with the priority dates from the instream flow and reservoir rights. If the last right has been read from the diversion data file, DIVER sets the diversion flag to true. This flag signals the main program not to process any more diversion rights. The priority date is set to 99999999 after the last right is read. This causes the diversion rights to have the lowest priority possible in the system so no other right will be injured.

<u>ISFLWA</u>

If the amount of instream flow required is greater than zero, ISFLWA tries to satisfy the instream flow right. The instream flow station is first located in the station array. The flow required is then adjusted by subtracting the difference between the river flow and the water available from the flow required to satisfy the right. The water available in the river at that station is checked. If there is not enough water available to satisfy the instream flow right, the available water is assigned to the instream flow right, messages are written to rights called out files (Tape 11 and Tape 13), and the available water at the station is set to zero. However, if there is enough water available to satisfy the instream flow right, the water required is removed from the water available at the instream flow station.

RESADJS

RESADJS sets the diversion station equal to the reservoir station (or the reservoir diversion station for off-channel storage reservoir) and designates the stream order to that station as the stream order for the current reservoir. Next, the reservoir rights index (the counter of the number of reservoir rights processed for the current reservoir this month) is incremented by one and the code for the current reservoir right is set equal to the reservoir right index. If the reservoir has been filled once during the current water year, the subroutine returns control of the model to the main program. Otherwise, the subroutine calculates the remaining right. This calculation consists of subtracting the reservoir right's year-to-date storage (the amount already stored by the reservoir right in the current water year) from the reservoir right (the amount the law allows a reservoir right to store in a water year). The remaining right is then compared with the remaining vacant capacity (maximum reservoir storage allowed in a year minus the amount stored in the reservoir during the current water year) and the remaining reservoir capacity (the maximum reservoir capacity minus the water currently stored in the reservoir). The lesser of these three is the amount that can be diverted to the reservoir during the current month. The program also checks to see if the diversion capacity for an off-channel storage reservoir will limit the amount stored in the reservoir during the current month.

If the amount the reservoir is allowed to divert is greater than zero, RESADJS checks to see if any senior downstream diversion, instream flow, or reservoir rights were not satisfied. If any senior downstream rights were not satisfied, CHECK writes a message to Tape 11 stating that the current right was not met because a senior downstream right was not satisfied. If all senior downstream rights were satisfied, the subroutine finds the station downstream from the reservoir with the least amount of water available for diversion. This becomes the amount of water that can be stored in the reservoir during the current month.

Off-channel storage reservoirs have the right to the return flows from their diversion canals. If there is not enough water available to satisfy an off-channel storage right, RETMON is called to add the return flows from the diversion canal to the river downstream from the diversion station.

If the amount the reservoir is allowed to divert is greater than the water available for diversion (including no water available for diversion), the available water is stored in the reservoir, a message is written to Tape 11, and a flag (read by CHECK) indicating that a reservoir right was not completely satisfied is set.

The subroutine also sets a flag for the parameter that limited storage by the reservoir right. The flag consists of a code for the limiting parameter. The codes are shown in Table VII.

Table VII Factor that limited reservoir storage codes.

CODE	LIMITING FACTOR
1	The reservoir has previously been filled during the current water year.
2	The remaining reservoir right has been satisfied.
3	The remaining vacant capacity is less than the remaining right.
4	The remaining reservoir capacity is less than the remaining right.
5	The off-channel storage diversion canal capacity is less than the remaining right.
6	A senior downstream right has been called out.
7	The flow available for diversion is less than the remaining right.

If the reservoir was filled, a spill condition exists for the remainder of the month and a flag is set indicating that the reservoir was filled this water year. The amount of water stored in the reservoir during the month is added to the current storage, the year-to-date storage, the monthly storage, the reservoir right account, and the reservoir right year-to-date storage arrays. RESADJS next checks to see if the reservoir right was satisfied during the current month and flag is set to signal this condition to the main program. The amount of water stored in the reservoir is then removed from the water available for diversion downstream from the reservoir station (RIVDIS and AVLDIS). Additionally, for an off-channel storage reservoir, RETFLW is called to add the return flows from the diversion canal to the river downstream from the diversion station.

JPREVAL

JPREVAL is the part of the WIRSOS model that reads junior project rights and tries to satisfy those rights with water from their associated reservoir. Junior project rights are only processed by JPREVAL if a spill condition exists at the reservoir (spill conditions are processed by the subroutine PROJRITE). If the last reservoir right associated with the reservoir was processed, JPREVAL reads the first right associated with the current reservoir (each project right is evaluated separately according to priority date).

Similar to other sections of the model, the subroutine must find the junior project right station in the station array. The diversion efficiency is also converted to an integer efficiency code and found in the efficiency table if monthly efficiencies are being used. The water available from reservoir storage is calculated by subtracting the minimum reservoir volume from the current reservoir storage. The flow capacity available from the reservoir is calculated by subtracting the flow in the river below the reservoir station from the maximum flow capacity of the outlet works. If a junior project right is associated with a reservoir right, the amount available in that reservoir right's account is the limiting diversion amount. If either the remaining flow capacity or the water available from the reservoir (or reservoir right) restrict the reservoir's ability to satisfy the junior project right, a message will be written to Tape 11 by RESCHK. If the junior project right is allowed to draw on two reservoirs, JPREVAL calls SCNDRESV to satisfy the unsatisfied portion of the right. Any water released from the reservoir for the junior project right is added to the flow in the river between the reservoir station and the junior project right station by RIVDIS and the amount released is added to the project releases for that reservoir. JPREVAL calls RETFLW to return the water (diverted to the right but not consumed) to the river system.

JPRIVER

JPRIVER evaluates junior project rights read as regular diversions. Junior project rights have a priority date junior to the most junior reservoir right on the reservoir with which they are associated. If a junior project right was not completely satisfied by its reservoir, JPRIVER tries to satisfy the remainder of the junior right with natural flow from the river. JPRIVER first finds the diversion station in the station array. After the station is found, the subroutine checks to see if there is any water available for diversion at the diversion station. If no water is available for diversion, a message is written to Tape 11 by LIMAVL. If there is water available for diversion, the program calls CHECK to see if any senior downstream diversion, instream flow, or reservoir rights were not satisfied. If a senior downstream water right was not satisfied, a message is written to Tape 11. Assuming that there is water available for diversion at the diversion station and all senior downstream rights were satisfied, the maximum amount of water available at each station downstream from the diversion station is determined. The minimum water available downstream becomes the amount available for diversion at the current station. The subroutine checks to see if the amount available for diversion is greater than zero (writing a message to Tape 11 if it is not) and removes the amount diverted from the water available downstream (RIVDIS and AVLDIS). If the right remains unsatisfied a message is written to Tape 11 by LIMAVL. The return flows are calculated and returned to the system by the subroutine RETFLW.

DIVEVAL

DIVEVAL processes "normal" diversions. "Normal" diversions are not associated with a reservoir and receive their water from direct flow. The subroutine performs the same function as described under the subroutine JPRIVER with one exception. The exception is that DIVEVAL calls RETMON to add the return flow for the current month to the river. RETMON is called if there is not enough flow available downstream to satisfy the right. DIVEVAL then rechecks to see if there is enough water downstream to satisfy the right. Otherwise, the two subroutines are the same. RETFLW is called to return excess water to the river.

PROJRITE

PROJRITE processes senior project rights and junior project rights treated as senior water rights when a reservoir is full and spilling water. PROJRITE starts by trying to fill the junior project right from the available flow in the river. The procedure followed is the same as that described under the subroutine JPRIVER. PROJRITE then calls DIVCHK to determine if a water exchange is necessary and the size of the water exchange. PROJRITE then tries to fill the remaining right from its reservoir as described under the subroutine JPREVAL. RETFLW is called to return excess water to the river.

DIVCHK

DIVCHK determines if a project right entered in the diversion data base is physically downstream from its associated reservoir. If the right is not downstream from its reservoir, DIVCHK calculates a water exchange between the river and the reservoir. The subroutine determines the water available between the diversion station and the first station downstream from the reservoir and the diversion station. The water available is the maximum amount that the right can divert. The amount diverted by the right is released from the reservoir to replace the water removed from the stream. DIVCHK also does this for rights calling on two reservoirs.

RESMIN

RESMIN takes the minimum release from a reservoir input by the modeler and compares it to the flow in the river at the station above the reservoir station. Whichever of these is the least, is the flow that must be released from the reservoir. RESMIN then subtracts the flow in the river at the station below the reservoir station from this amount to obtain the amount of water that must be released from reservoir storage to satisfy the minimum release from the reservoir.

MINRSRL

Once RESMIN has determined the amount of water needed to satisfy the minimum release requirement from a reservoir, MINRSRL checks to see if the water is available from reservoir storage. If the water is available (or partially available), the water released is removed from current storage, the water is removed from the reservoir right accounts according to the percent of the current storage each reservoir right controls, the water released is added to the non-project reservoir releases, and the water is distributed to all stations downstream from the reservoir station by RIVDIS.

OVERSUP

OVERSUP determines if there is an extreme supply of water for the current month and what reservoirs are capable of storing that water. The program initializes three temporary arrays: one for the reservoir codes, one for the active reservoir flag, and one for the availability. The program then removes any inactive reservoirs from consideration and sorts the active reservoirs by station number (the reservoir furthest upstream is considered fist). OVERSUP checks each reservoir to see how much reservoir capacity it has and how much water is available to be stored. The amount available for storage is removed from the temporary availability array and added to the total amount available for storage. During this process any reservoir that is not capable of storing water is flagged inactive in the temporary activity array.

Inactive reservoirs are removed and the active reservoirs are again sorted by station number. To determine the amount stored in each reservoir the total storage is divided by the number of active reservoirs. The program then tries to store that amount in each active reservoir. A reservoir may not be able to store its allotted amount depending on the remaining reservoir capacity and the water available for storage. The remaining diversion capacity can also limit further reservoir storage for off-channel storage reservoirs. If any reservoir can not take its storage, that reservoir is flagged as inactive. Any water not stored by a reservoir remains in the total amount to be stored. The water stored is added to the current storage, the year-to-date storage, the monthly storage, the reservoir rights year-to-date, the reservoir right's account, the flag of reservoir rights met, the spill flag, the fill flag, and the extreme monthly storage are adjusted. Water is added to the reservoir rights accounts according to the amount of permitted reservoir storage that each reservoir right controls defined in RRBALNC. If the reservoir is an off-channel storage facility, the return flows are calculated and returned to the system. If there is any water remaining in the total amount to be stored, the program returns to the section that removes any inactive reservoirs and sorts the active reservoirs. This is repeated up to the total number of reservoirs.

EVAPR

For each active reservoir the user can define three storage ranges in the reservoir area input file (Inp 14). For each storage range, there is a different area versus capacity relationship. EVAPR sorts the number of ranges defined by the model user, and for each range it calls the subroutine EVAPSUB. After the areas for the initial storage and the final storage for the month are calculated by EVAPSUB, the two reservoir areas are averaged and used to calculate the evaporation from the reservoir for that month. The evaporation from a reservoir is removed from the current storage in that reservoir. The evaporation is removed from the reservoir rights account according to the amount of the current storage that each reservoir right controls.

EVAPSUB

EVAPSUB is one of the subroutines that was developed with the original program. EVAPSUB selects one of five equations that define an area versus capacity relationship. The equations are selected according to the equation type specified for the given storage range in the reservoir area-capacity curve data file. EVAPSUB returns an area for a given storage volume. The area is used to calculate evaporation.

EOM

EOM is called after each month of river system simulation. For each active reservoir, the subroutine writes the reservoir name, the monthly reservoir storage, the flow in the river at the reservoir station, the water required for power releases, the actual water released for power, the water required for non-project releases, the actual water released for non-project demands, the water released for project rights, the evaporation from the reservoir, the extreme supply stored during the month, and the current storage to a temporary output file (Tape 22). For each active reservoir, the subroutine also writes the reservoir number, the reservoir right number, the end-of-month reservoir right's account, the year-to-date reservoir right met, and a code (set in RESADJS) that indicates what limited reservoir storage at the reservoir rights priority date to a temporary output file (Tape 23).

The variables that track monthly project releases and monthly storage are reset to zero. The flag that indicates that a reservoir is spilling is reset and the flag that indicates that junior project rights have been processed with the reservoir is also reset.

The subroutine calculates the water required for power releases. Before calculating the power required and the power release, EOM checks to see if a power goal month has been set. If one has not been set, the power required and the power release are set to zero. However, if a goal month has been set, the subroutine sets the power required and the power release for the months remaining between the current month and the goal month. The power release goal is set by subtracting the power goal volume from the current storage in the reservoir. If the goal month is less than or equal to the current month, the goal month is reset to the same month as the goal month but for the next year. The number of months to reach the goal volume is calculated by counting the number of months between the current month and the goal month. The power release and the power required are computed by dividing the power release goal by the number of months until the power release goal month, provided that the power release goal is greater than zero.

END-OF-YEAR OUTPUT

EOY

The subroutine EOY writes to two output files at the end of each year of river system simulation after all rights have been addressed. The first output file (Tape 9) records the water flowing (cfs) in the river system at every station for each month and the total flow (acre-feet) at every station for the year. The second output file (Tape 10) records the water available (cfs) in the river system at every station for each month and the total flow (acre-feet) available at every station for the year.

END-OF-RUN OUTPUT

TWELVE

When WIRSOS is run it creates two output files (Tape 12 and Tape 13). These files contain a list of the diversion and instream flow rights called out. Each right called out during a month is written to these files along with the percent called out and the station where the right is located. These files are then sorted by month, year, and permit number and written to a new file to make them easier to analyze. The restructuring of these output files was not performed by WIRSOS, but was done by a program that was executed after a model run. For simpler model operation and to consolidate model operations, a subroutine that performs this function was written.

TWELVE reads one month of data from the output file. The percent called out is then stored in an array location corresponding to the current month. Each permit is assigned its own array to contain the percent called out each month. Another month of data is then read from the output file. The program checks to see if any of the rights were called out during the months previously read. If the right was previously read, the percent called out is stored in the array address corresponding to the current month and one of the duplicate rights is deleted. Otherwise, the percent called out is stored in the array for the new right. This is repeated until every right that was called out during the current year is read and checked for repetition. The program then sorts the rights according to an alpha-numeric priority. The lowest right number or right character is stored first in the rights called-out array. Once the rights are sorted, a header containing the year and the months of the year is written to the new output file. The permit numbers, the percent called out each month, and the station where the right is located are written to an output file. The process is repeated until every year of the model simulation is processed.

EOR

This subroutine is executed at the end of a model run after all the years have been simulated. EOR reads the information from the file written to during the subroutine EOM (Tape 22) and rewrites the data to another output file (Tape 18). In the process, the information is sorted by reservoir, month, and year. Tape 18 consists of a report of the status of each active reservoir at the end of each month for every year simulated by the model. The subroutine also creates headers and captions for the output file, so that the information is easier to understand.

<u>TOR</u>

This subroutine is executed at the end of a model run after all the years have been simulated. TOR reads the reservoir right information from the file written to during the subroutine EOM (Tape 23) and rewrites the data to another output file (Tape 19). In the process, the information is sorted by reservoir, reservoir right, month, and year. Tape 19 consists of a report of the status of each active reservoir and the rights associated with the reservoir at the end of each month for every year simulated by the model. The subroutine also creates headers and captions for the output file, so that the information is easier to understand.

CHAPTER V

TESTING THE MODEL

This chapter describes the procedures used to test the WIRSOS program after modifications to the model were finished. Testing of the model at incremental stages of the project was also done and will be discussed as it is presented. All modifications to the model and the testing of each modification is discussed.

INTRODUCTION

The WIRSOS model was quite large before any work was done to enhance its capabilities, and each new feature increased the size of the model. The current model has approximately twice the code of the original program model. The increased size of the program has increased its complexity. Unfortunately, it also means an increased chance for error. In an attempt to limit the number of errors, the program was partially "debugged" each time the program was changed. After the model was completed, a data base was modified and every aspect of the completed model was tested.

The features tested were: an initial run to create a comparison file; initial reservoir rights year-to-date distributed by the user; initial reservoir rights year-to-date distributed by the program; minimum release from a reservoir; off-channel storage reservoir; project rights tied to a reservoir right; project rights allowed to withdraw water from two reservoirs; water exchanges for project rights not downstream from their associated reservoir; extreme monthly storage; checking the water rights and station input files; a combination of all options; and a combination of all options with the full data base.

One problem encountered while testing the model, was that there is no way to "check the answer". Instead of checking for a "correct" answer, the model was to drive the model to extreme conditions. The output was then checked to see if it reflected the correct direction expected for the given condition. The final action of the testing plan was to create data bases and output for the current model, so that any future modifications made to the program would have a data base and its corresponding output files to compare to the output files produced by the modified program.

INCREMENTAL TESTING

The size of the model created the problem of how to handle the output files, so testing the model had to be addressed before the program was divided into subroutines.

First, the data base used to test the model contained thirteen years of runoff data. The output data files created from thirteen years of runoff data were too large to edit or compare. Therefore, the runoff data base was trimmed to two years of data. Two years of runoff data still produced output files were still too large to efficiently compare the results of separate model runs. In response to this problem, a small program was written that reads the lines of two output files and compares the files line by line. If the files differed in any way, the program would detect the differences. Before making any changes to WIRSOS, the latest version was used to create output files for comparison with output files created by the modified version.

After reviewing the program and studying the output files, it became apparent that Tape 11 (list of rights called out) was the major output file that needed to be compared. Tape 11 was chosen because it is written to by virtually every major section of the model and any changes would show up in this file. After each alteration or addition to the program, the model was recompiled and a new set of output files was produced. The comparison program was then ran on the original Tape 11 output file and the new Tape 11 output file to easily identify any differences between the two files. If there were any differences, the error that caused the difference was located and eliminated. If the difference was not caused by an error, the reason for the change was isolated and it determined if the difference was a desired response. If the difference between the two Tape 11 files was the result of a change in the program and not an error, the original output files were replaced with the new output files. The model was tested in this manner after each modification to WIRSOS.

One problem with the incremental testing, was its inability to test any of the new features added to the program. However, it did verify that the current version of WIRSOS (WIRSOS5) is capable of reproducing the output from the original model. To test the new features, new data bases were created and each function had to be tested separately.

DATA BASES

The original diversion rights data base used in the initial phases of this project was not sorted by priority date (as required by WIRSOS). This data base, the junior project rights, the instream flow rights, and the reservoir rights data bases were sorted by priority date, station number, and permit number. Additionally, the reservoir data file (inp 15), the diversion rights file (inp 4), and the junior project rights file (inp 17) were modified to accept data compatible with WIRSOS5. The modification of these data bases is discussed in the section where testing of each added feature is described. An input file for the efficiency tables (inp 20) was also created.

INITIAL RUN

In addition to testing the added features, WIRSOS5 was ran with all the run options disabled. This produced an initial set of sample output files to compare to files created testing the model. For this initial run, efficiency tables were used and Tapes 12 and 13 were sorted at the end of the model run. Reservoir rights year-to-date were set to zero at the start of the model run and at the beginning of each water year. The initial run was also performed with the sorted diversion data files.

RESERVOIR RIGHTS DIVISION BY MODELER

One of the features added to WIRSOS, was the ability to associate storage in a reservoir with a particular reservoir right. With this feature, the reservoir rights year-to-date (the amount stored by a reservoir right during the current water year) are set equal to the reservoir rights account at the beginning of each water year. The reservoir rights account is all the water stored and removed from a reservoir right during the course of the year. As part of this option, the modeler is given two choices for dividing the initial reservoir volume (volume of the reservoir at the start of the model run) between the reservoir rights. The two choices are initial reservoir rights year-to-date distributed by the modeler and initial reservoir rights year-to-date distributed by the program.

Initial reservoir rights year-to-date distributed by the modeler involves inputting values in the reservoir data file. These values indicate how to divide the initial reservoir volume between the reservoir rights. The modeler can input up to four values (corresponding to the four rights possible for each reservoir). These values should sum to 100%. In this manner, the modeler can chose to assign a percentage of the initial reservoir volume to each right. These values are entered (as a percentage value with no decimal points) in the reservoir data file after the power release goal volume.

A new output file was created to report the reservoir rights accounts for each month of the simulation (Tape 19). In addition to reporting reservoir rights account, the new output file reports reservoir rights year-to-date and the factor that limited reservoir storage at the reservoir right's priority date. This file was used to test the reservoir rights distributed by the modeler option of the WIRSOS5 program.

The data base used to test the model has only one reservoir with multiple reservoir rights (Upper Sunshine). For this test, Upper Sunshine was assumed to store an initial volume of 10,000 acre-feet. The program was directed to assign 65% of this initial volume to the most senior right and 35% to the junior right. A check of Tape 19 after the model run indicated that 3500 acre-feet had been assigned to the junior right. This approach proved conclusively that this option is functioning properly.

RESERVOIR RIGHTS DIVISION BY PROGRAM

The second way to distribute reservoir volume between reservoir rights is to allow the model to distribute the volume between the rights. To distribute the initial reservoir volume in this manner, the modeler sets the reservoir right distribution values in the reservoir data file to zero. The reservoir rights year-to-date are also set equal to the previous water year's reservoir rights account at the start of each water year.

When reading the reservoir data at the beginning of a model run, the program assigns the initial reservoir volume according to the percent of the total permitted volume that each reservoir right controls. For Upper Sunshine, the junior reservoir right controls 6.66% of the permitted reservoir volume. When the model was run, 666 acre-feet of water were assigned the junior reservoir right. This proved that the option was working correctly since 666 is exactly 6.66% of 10,000.

At the beginning of each water year, the reservoir rights year-to-date for the junior right equalled to the reservoir right's account of that right at the end of the previous water year. It was concluded that the model was distributing reservoir rights correctly, and that the reservoir right accounts were functioning properly.

MINIMUM RELEASE FROM A RESERVOIR

Upper Sunshine was again selected to test this model option. A value can be entered in the reservoir data file after the initial storage distribution values for the minimum release from a reservoir. If no minimum release is desired, the minimum release value should be set to zero. The minimum release from a reservoir is either the flow to meet the minimum release value or the flow entering the reservoir at its upstream station. The flow entering the reservoir at its upstream station is used if the flow entering is less than the minimum release value. The flow released is the minimum flow value (or the entering flow) minus the flow below the reservoir. The water released from the reservoir is distributed between the reservoir rights according to the amount of water, stored in the reservoir, that each reservoir right controls. Thus, if 10% of the water stored in a reservoir is controlled by a reservoir right, 10% of the water released will be taken from that right.

To test this option, the minimum release from Upper Sunshine was set at ten cfs. After the model run, the flow (at the river station below the reservoir) was reported in Tape 18 as 10 cfs. The option works correctly.

OFF-CHANNEL STORAGE

To allow off-channel storage reservoirs in WIRSOS, an additional line of data had to be created for each reservoir in the reservoir data file. This extra line contains the ten return flow stations, percent returned at each station, and a return flow delay table number to be used at each station. However, the off-channel storage reservoir diversion station, the reservoir diversion efficiency, the number of return flow stations for the reservoir diversion, and the reservoir diversion canal capacity are located on the second line of data following the minimum release value. The program is directed to model a reservoir as an off-channel storage facility by setting a flag in line one of the reservoir data file.

To test off-channel storage, the reservoir diversion canal capacity was set at ten cfs for Lower Sunshine reservoir. The intent was for the diversion canal capacity to limit reservoir storage. The canal capacity would be the limiting factor reported to Tape 19. This was the case after the model run. Therefore, it is assumed that the off-channel storage option is working correctly.

PROJECT RIGHTS TIED TO A RESERVOIR RIGHT

In order for a project right to be associated with a particular reservoir right, the diversion and junior project rights data bases were modified. Actually, no columns were added to the data bases, but the function of the second column in the first line was changed for each right. In the diversion data base, the second column was used to indicate whether the right was a "normal" diversion or project right. However, this is no longer necessary since the variable DIVTYP performs this function. The second column of the first line of each diversion right is used to indicate whether the right is tied to a particular reservoir right (if the diversion right is a project right). Hence, a value of 0 indicates the right is not tied to a particular reservoir right, and values of 1-4 indicate which reservoir right the project right is associated.

In the junior project rights data base, the second column in the first line of each right previously performed no function. Therefore, making the second column indicate which reservoir right the junior project right was associated did not change the program. In the junior project rights file, a value of 1-4 indicate which reservoir right the junior project right is associated. Any other character indicates the junior project right is not associated with a particular reservoir right.

With this in mind, all project and junior project rights that draw on the Upper Sunshine reservoir were forced to draw on a the junior reservoir right. This was accomplished by putting a '2' in the second column of the first row of each project and junior project right associated with Upper Sunshine. When the model was tested, the project rights released from Upper Sunshine (as reported by Tape 18) decreased. The reservoir released water normally for the first three month when all releases stopped. In addition, Tape 19 indicated that the second right on Upper Sunshine was depleted after the third month of operation. The conclusion reached was that this option was functioning correctly.

PROJECT RIGHT TIED TO TWO RESERVOIRS

Associating a project right with two reservoirs was not as easy as associating a project right to a particular reservoir right. To allow this option to happen, the diversion and junior project rights files had to be modified. In each file, the information for this option is entered at the end of the second line of data (columns 113-116). The information consists of: a flag that indicates that the project right is allowed to withdraw water from a second reservoir, the reservoir from which the water is to be withdrawn, and the reservoir right to which the project right is tied. The flag is either '0' for no second reservoir or '1' to draw from a second reservoir. The reservoir is the reservoir code just like DIVTYP and RESNUM in the diversion and junior project rights data bases, and the reservoir right code is the same as the diversion data base ('0' for no particular reservoir right or '1-4' for a particular reservoir right).

Upper Sunshine was chosen as the test reservoir for this model option. Upper Sunshine was chosen because its releases enter the main branches of the Greybull river below Lower Sunshine reservoir. All rights that were physically below both Sunshine reservoirs, could withdraw first on Upper Sunshine and then on Lower Sunshine if the right remained unsatisfied. To test this option work, the river system was stressed beyond "normal" operating limits. To do this, a "dummy" right was added to the diversion and junior project rights data bases. This "dummy" right tried to withdraw over 100 cfs from the river system. While trying this option, the project rights were not associated with a particular reservoir right. When the model ran, Tape 18 showed large (greater than 100 cfs) releases for project rights from Upper Sunshine until that reservoir was empty, and then larger than normal releases for project rights were made from Lower Sunshine until it fell below its minimum volume. In conclusion, this model option appeared to be working correctly.

WATER EXCHANGES

A water exchange occurs when a project right is not physically downstream from its associated reservoir. When this happens, the project right receives its demand from flow in the river. This flow is then replaced by water from the reservoir. To test this option, a right was entered above Lower Sunshine reservoir. This right withdrew water from Lower Sunshine and its demand was 100 cfs per month. When the model was ran, large releases were made from Lower Sunshine and the right was written to the rights called out file (Tape 11). In addition, statements that reported model progress were used to write messages to screen. These messages traced the model progress as the dummy right was processed. It was concluded that the model was working.

EXTREME MONTHLY STORAGE

Extreme monthly storage which allows extreme excess water in the system to be stored out of priority did not require the alteration of any data bases, but did require considerable effort to program into the model. Once it was part of the WIRSOS model, it seemed to operate quite well. The extreme monthly storage option is chosen in the model run information section of the interactive user interface. When this option was initiated, more water was stored in the reservoirs (Tape 18) after their year-to-date rights were met (Tape 19). No water was stored if water available downstream, senior downstream rights, or off-channel diversion capacity limited the amount of water stored by a right during a reservoirs priority date (Tape 19). Tape 11 showed that fewer rights were called out during this simulation, which was expected since there would be more water stored in the reservoirs. In short, the option appeared to be working correctly.

DATA CHECKING

The subroutines that were developed to check the diversion rights, junior project rights, instream flow rights, reservoir rights, and station data files were also tested. For every aspect of a data file that is checked, a data record was deliberately entered incorrectly to test the model. Each time the proper error message was printed to the screen and program operation was stopped.

COMBINED MODEL

As a final check on the model operation, all the options and revised data bases were combined. The model was then tested using these data bases with all options functional. Everything that was checked after this run would take too much space to describe; therefore, it is sufficient to say that everything was operating as expected. In addition to Tape 18 and Tape 19, every output file was visually inspected for any out of place characters or anomalies. Tape 11 was inspected to check all callout messages written to that file to determine whether all aspects of the improved program were functioning. Based on this information, it was concluded that the model was functioning properly.

COMBINED MODEL WITH FULL DATA BASE

At this point, the combined model with the full thirteen years of runoff data was ran. The purpose of running this model was to try and catch any errors overlooked during the previous testing. The model ran without problems.

DATA BASES AND KNOWN OUTPUT

To create the data base and known output files for the current model, the full thirteen years of runoff data were chosen. In addition, no project rights were tied to a particular reservoir right, the off-channel diversion capacity was set at 100 cfs, the minimum release was set at 0.5 cfs, all project rights below both Upper and Lower Sunshine were allowed to call on both reservoirs, reservoir rights year-to-date were set to the previous years reservoir rights account, the beginning volume was divided 35% to the junior right and 65% to the senior right, and the reservoirs were allowed to store any extreme supply of water in the system.

CONCLUSIONS

When testing the model, the first run to test each option often failed. Therefore, much effort was made to track and eliminate the errors discovered. In the end the options worked as expected, and from the output there is a reasonable level of confidence that the model is working properly.

CHAPTER VI

SUMMARY AND RECOMMENDATIONS

This chapter contains a summary of the modifications to the WIRSOS program, recommendations for efficient model operation, and considerations for future modification.

SUMMARY

The original WIRSOS model had many good features that allowed it to model water rights within the state of Wyoming. WIRSOS also had weaknesses, one of which was the modeling reservoirs. It was the purpose of this thesis to improve the operation of the WIRSOS model in several aspects including reservoir operations. To achieve this objective, reservoir storage, project rights handling, reservoir rights handling, and releases from a reservoir were modified in the model. These modifications are an improvement to the operation of WIRSOS where reservoirs are concerned. In addition to modifying WIRSOS, the program was separated into subroutines and the previous modifications were incorporated into the current model. An improved modular program for river system operation simulation was created which allows for ease of change where special problems arise in modeling a river system.

After WIRSOS was separated into subroutines, previous modifications and additions to the model were incorporated into the program. These improvements affected every subroutine in the model to some degree. A few subroutines remained relatively unchanged; however, a few were altered significantly. In addition to altering existing subroutines, twenty additional subroutines were added to enhance WIRSOS. Eight of these subroutines were previous modifications that were partially or totally altered or expanded to handle more information and made into subroutines. The other twelve are totally new subroutines.

The first eight subroutines are: RRDIS which removes water, released from a reservoir, from the reservoir rights accounts; HEADER which controls the operations of the interactive user interface; INPUT which reads in the input data file names and locations; OUTPUT which reads in the output file names and locations; MDLRUN which reads in the model run information; HEAD which writes the header, the time, and the date for each run to the output files; EFFNCY which reads in the monthly efficiency tables; and TWELVE which sorts Tapes 12 and 13 by permit number, month, and year before writing the information to another output file. The last twelve are: PRICHK, RSVCHK, and IFRCHK which check the rights data files to determine if the rights were entered in the proper sequence; STACHK which checks the station data file to determine if the stations were entered in the proper order; JPRCHK which checks if any rights were entered that are not downstream from their associated reservoir; DIVCHK which checks and computes water exchanges; SCNDRESV which processes a project rights request for water from a second reservoir; RRBALNC which determines the percent of the total permitted reservoir volume that each reservoir right controls and the number of reservoir rights associated with a reservoir; RESMIN which determines the amount of water needed from a reservoir to satisfy the minimum release requirement from a reservoir; MINRSRL which releases the water needed to satisfy the minimum release from a reservoir; OVERSUP which stores an extreme supply of water in the river system in the reservoirs on that system; and TOR which reads the information written to temporary file Tape 23 by EOM and sorts that information by right, month, and year before writing the information to Tape 19. Finally, the last file added was WIRSOS5.FC which contains the variables declared and used in WIRSOS5.

Though the model runs well, there are a few items that should be kept in mind during its operation. First, each reservoir must have a station above and below the reservoir station, so that the minimum flow below a reservoir can be calculated. Second, any junior project right that calls on two reservoirs must be physically downstream from both reservoirs. Third, an integer value greater than zero must be entered for the first year of model operation. Finally, any existing model that is to be ran with the current version of the program must have an extra line added for each reservoir in the reservoir data file. This is because the program now tries to read an extra line of data for an off-channel reservoir. The model could have been set up to read this line only if a reservoir was an off-channel storage reservoir, but the data base manager for the input data base files would not support the format. Also, only ten of the fifty reservoirs allowed by the model can be off-channel storage reservoirs.

The model created when WIRSOS was modified is a very powerful tool for modelling water rights in the state of Wyoming. Furthermore, this tool should be capable of meeting the states needs into the future. As water becomes a more scarce commodity, each state needs the best tools possible to utilize their water to the best of their ability.

RECOMMENDATIONS

Two areas are possible candidates for future improvements to the WIRSOS model. These are 1) how area is calculated for the reservoir evaporation section of the model; and 2) equalization of water held in reservoirs.

The area of a reservoir is currently calculated from the storage of the reservoir, and the area is calculated for the beginning and the end of each month. The two areas are then averaged to get an average area for the month for calculating evaporation. There are many ways in which this operation could be improved; however, most are too complicated for the simple nature of WIRSOS. Therefore, it was decided not to change this function at this time.

In the case of a project right being allowed to draw water from two reservoirs, the first reservoir must be empty before water is drawn from the second reservoir. In actual practice, water would be drawn evenly from both reservoirs, and the program should reflect this actuality. However, such a change would require considerable restructuring of WIRSOS and its input files. Therefore, it was decided not to implement this function.

SELECTED REFERENCES

Anderson, H. N. and Shaw, D. B. (1987), "Satellite Data for Adjudications in Idaho", James, L. G. and English, M. J. (ed.), Irrigation Systems for the 21st Century, Proceedings of Irrigation and Drainage Division ASCE, pp. 258-262

Anibal A., Wright, J. R., and Houck, M. H. (1990),"Bayesian Inferencing Applied to Real-Time Reservoir Operations", J. Water Resources Planning and Management, vol. 116, no. 1, pp. 38-51

Barnes, G. W. and Chung, F. I. (1986), "Operational Planning for California Water System", J. Water Resources Planning and Management, vol. 112, no. 1, pp. 71-86

Brendecke, C. M., DeOreo, W. B., Payton, E. A., and Rozaklis L. T. (1989), "Network Models of Water Rights and System Operations", J. Water Resources Planning and Management, vol. 115, no. 5, pp. 684-696

Bridgeman, S. G., Norrie, D. J. W., Cook, H. J., and Kitchen, B. (1988), "Computerized Decision-Guidance System for Management of the Trent River Multireservoir System", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 97-105

Brosz, D. J. and Jacobs, J. J. (1980), "Wyoming Water Resources", AE 80-90, Division of Agricultural Economics, Laramie, Wyoming: University of Wyoming.

Changchit, C. and Terrell, M. P. (1989), "CCGP Model for Multiobjective Reservoir Systems", J. Water Resources Planning and Management, vol. 115, no. 5, pp. 658-670

Chung, F. I., Archer, M. C., and DeVries, J. J. (1989), "Network Flow Algorithm Applied to California Aqueduct Simulation", J. Water Resources Planning and Management, vol. 115, no. 2, pp. 131-147

Chung, I. and Helweg, O. (1985), "Modeling the California State Water Project", J. Water Resources Planning and Management, vol. 111, no. 1, pp. 83-97

Coe, J. Q. and Rankin, A. W. (1988), "California's Adaptable Model for Operations Planning for the State Water Project", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 119-130

Courtney, B. M. and Whitlock, A. W. (1988), "Computer Support for TVA's Reservoir System Operations: Software", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 51-56

Crower, K., Diamond, E., Ford, J., and Witt, R. H. (1988), "Computerized Resource Planning for the Central Valley Project California", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 97-105

Diaz, G. E. and Fontane, D. G. (1989), "Hydropower Optimization via Sequential Quadratic Programming", J. Water Resources Planning and Management, vol. 115, no. 6, pp. 715-734

Eichert, B. S. and Franke, C. (1988), "Multi-Purpose, Multi-Reservoir Simulation on a PC", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 313-327

Fisher, G. T. (1989), "Georaphic Information System/ Watershed Model Interface", Ports, M. A. (ed.), Hydraulic Engineering, Proceedings of the 1989 National Conference on Hydraulic Engineering, pp. 851-856

Flug, M. and Ahmed, J. (1990), "Prioritizing Flow Alternatives for Social Objectives", J. Water Resources Planning and Management, vol. 116, no. 5, pp. 610-624

Flug, M., Fontane, D. G., and Ghoneim, G. A. (1990), "Modeling to Generate Recreational Alternatives", J. Water Resources Planning and Management, vol. 116, no. 5, pp. 625-638

Ford, D. T. (199), "Reservoir Storage Reallocation Analysis with PC", J. Water Resources Planning and Management, vol. 116, no. 3, pp. 402-416

Grainey, M. S. (1987), "Managing Water Resources Using Geographic Information Systems as a Tool", James, L. G. and English, M. J. (ed.), Irrigation Systems for the 21st Century, Proceedings of Irrigation and Drainage Division ASCE, pp. 693-699

HYDROSS version 4.3 (1991), "HYDROSS Users Manual", Information Resources Division, Bureau of Reclamation, Billings, MT

Johnson, L. E. (1991), "GIS Visualization Techniques in Surface Water Resource Planning", Shane, R. M. (ed.), Hydraulic Engineering, Proceedings of the 1991 Conference, pp. 955-960

Kuo, J., Hsu, N., Chu, W., Wan, S., and Lin, Y. (1990), "Real-Time Operation of Tanshui River Reservoirs", J. Water Resources Planning and Management, vol. 116, no. 3, pp. 349-361

Labadie, J. W., Bode, D. A., and Pineda, A. M. (1986), "Network Model for Decision-Support in Municipal Raw Water Supply", Water Resources Bulletin, vol. 22, no. 6, pp. 927-940

Loganathan, G. V. and Bhattacharya, D. (1990), "Programming Techniques for Optimal Reservoir Operations", J. Water Resources Planning and Management, vol. 116, no. 6, pp. 820-838

Loucks, D. P. (1976), "Surface Water Quantity Management Models", in Biswas, A. K. (ed.), Systems Approach to Water Management, New York, New York: McGraw-Hill, Inc., pp. 156-161.

Meena, J. M. (1993), "Water Allocation Model on the Green River", Masters Thesis, University of Wyoming, Laramie, pp. 7-16

Mohammadi, B. (1988), "Colorado River Microcomputer Model", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 328-340

Rockwood, D. M. (1982), "Theory and Practice of the SSARR Model as Related to Analyzing and Forecasting the Response of Hydrologic Systems", Singh, V. P. (ed.), Applied Modeling in Catchment Hydrology, Proceeding of the International Symposium on Rainfall-Runoff Modeling, pp. 87-106

Rockwood, D. M., Speers, D. D., and Davis, E. (1988), "Operational Forecasting in Complex River Basins with SSARR-Micro Versus Mainframe Computer", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 302-312

Sabet, M. H., Coe, J. Q., Ramirez, H. M., and Ford D. T. (1985), "Optimal Operation of California Aqueduct", J. Water Resources Planning and Management, vol. 111, no. 2, pp. 222-237

Schuster, R. J. (1989), "The Colorado River Simulation System", Ports, M. A. (ed.), Hydraulic Engineering, Proceedings of the 1989 National Conference on Hydraulic Engineering, pp. 473-478

Shane, R. M. and Gilbert, K. C. (1982), "TVA Hydro Scheduling Model: Practical Aspects", J. Water Resources Planning and Management, vol. 108, no. WR1, pp. 1-20

Shane, R. M., Waffel, H. D., Parsly, J. A., and Goranflo H. M. (1988), "TVA Weekly Scheduling Model Application Experience", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 426-438

Sigvaldason, O. T. (1976), "A Simulation Model for Operating a Multipurpose Multireservoir System", Water Resources Research, vol. 12, no. 2, pp. 263-278

Staschus, K., Johnson, S. A., Stedinger, J. R., and Tejada-Guibert, A. (1988), "Computer Simulation of CVP Power Production for Integration with PG&E's Power System", Labadie, J. W., Brazil, I. C., and Johnson, L. E. (ed.), Computerized Decision Support Systems for Water Managers, Proceedings of the Third Water Resources Operations Management Workshop, pp. 106-118

Tejada-Guibert, A., Stedinger, J. R., and Staschus, K. (1990) "Optimization of Value of CVP's Hydropower Production", J. Water Resources Planning and Management, vol. 116, no. 1, pp. 52-70

Weghorst, P. A., Cunningham, J., and Mortazavi, B. (1991), "Geographic Information Systems in Water Resources Management", Shane, R. M. (ed.), Hydraulic Engineering, Proceedings of the 1991 Conference, pp. 889-893

Wei, T. C. (1977), "North Platte River Operational Options Study With of Without Enlarged Seminoe Reservoir using a Simulation Model", Water Resources Research Institute, University of Wyoming, Laramie, Wyoming

WIRSOS (1992), "Wind River WIRSOS Users Manual", State Engineers Office, Cheyenne, Wyoming

WIRSOS (1985), "WIRSOS Users Manual", Water Research Center, University of Wyoming, Laramie, Wyoming

Wurbs, R. A. (1991), "Optimization of Multiple-Purpose Reservoir System Operations: A Review of Modeling and Analysis Approaches", Research Document 34, U.S. Army Corps of Engineers, Hydrologic Engineering Center, Davis, CA.

Wurbs, R. A. and Walls, W. B. (1989), "Water Rights Modeling and Analysis", J. Water Resources Planning and Management, vol. 115, no. 4, pp. 416-430

Yeh, W. W. (1985), "Reservoir Management and Operation Models: A State-of-the-art review", Water Resource Research, vol. 21, no. 12, pp. 1797-1818

APPENDIX A

The following appendices can be found with the WIRSOS5 reference manual: WIRSOS5 program listing, WIRSOS5 flowcharts, WIRSOS5 variable list, WIRSOS5 sample input files, and WIRSOS5 sample output files. These appendices are contained on 5 1/4" diskettes to save printing costs. The WIRSOS5 reference manual or these files can be obtained from the Wyoming State Engineers Office in Cheyenne, WY; or from the Wyoming Water Resource Center in Laramie, Wy.