SUPPLEMENTS TO THE HEC-1 HYDROLOGIC MODEL,

USING THE MONTE CARLO METHOD AND

ARTIFICIAL NEURAL NETWORK
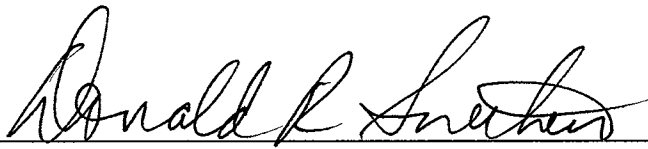
By

BORIS DANILO HERNANDEZ

Bachelor of Science
San Diego State University
San Diego, California
1992

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1996

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
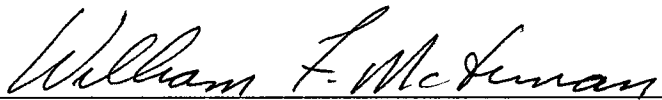DOCTOR OF PHILOSOPHY
December, 2001

SUPPLEMENTS TO THE HEC-1 HYDROLOGIC MODEL,

USING THE MONTE CARLO METHOD AND
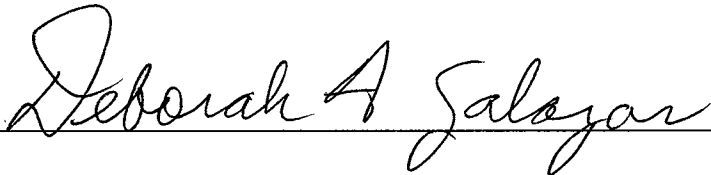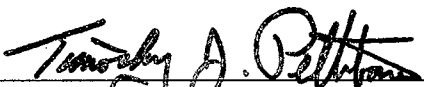
ARTIFICIAL NEURAL NETWORK

Thesis Approved:

_____
Thesis Adviser

_____

_____

_____

_____
Dean of the Graduate College

PREFACE

This research study was conducted to provide two methods of enhanced flood forecasting to improve the one-dimensional flood forecasting tools used by the U.S. Army Corps of Engineers since 1967. The one-dimensional HEC-1 software is a relatively simple tool used for forecasting, however, due to the one-dimensional aspect of this software, it presents some serious limitations. A typical basin has many factors and characteristics that influence the rainfall-runoff process. Infiltration rates, evapo-transpiration rates, evaporation rates, ponding, rainfall concentration, and many other spatial, temporal, and terrain parameters are difficult to model and yet are essential to the runoff process. The Monte Carlo method was used in this report to give all possible combinations of the few HEC-1 input parameters to represent the rainfall-runoff potential. The Artificial Neural Network method was also studied to develop a relationship between Doppler rainfall data and the runoff potential of a basin. Both of these methods were used as additional aids to the HEC-1 software.

Chapter 3 describes the Monte Carlo method in which a detailed study of the rainfall-runoff process for the years 1997, 1998, and 1999 was conducted. Chapter 4 describes the Artificial Neural Network in which a relationship between Doppler rainfall data and effective rainfall runoff was developed. Both these methods were used to study the Glover Basin located in southeastern Oklahoma. This basin has been difficult to forecast in a timely and accurate manner. By the time that the HEC-1 model is calibrated to the observed real hydrograph produced by the storm event, the reservoir control

personnel may have had to implement in-appropriate actions the conditions. These tools have been used as an enhancement to better model the effective-rainfall runoff of this basin.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

PROBLEM STATEMENT

Flooding is one of the worst and most costly natural disasters that Americans face today. Between 1986 and 1995 the United States incurred an average annual loss of $3.5 billion dollars due to flooding.(USACE, 1996)[42] The 1993 floods that occurred in the midwest and eastern portions of the United States cost more than $16.4 billion dollars. "The Great Flood of 1993" was the costliest flood event within the last 15 years.(USACE, 1996) [42] Table 1.1, taken from the U.S. Army Corps of Engineers 1996 Annual Report, presents a summation of flood damages in U.S. dollars between 1986 and 1995.

TOTAL FLOOD DAMAGES SUFFERED									
FISCAL YEARS (FY), IN MILLIONS OF US DOLLARS									
FY 86	FY 87	FY 88	FY 89	FY 90	FY 91	FY 92	FY 93	FY 94	FY 95
6,007	1,444	225	1,081	1,636	1,699	763	16,370	5,111	3,546

Table 1.1: Total flood damages in United States between 1986 and 1995. (USACE, 1996) [42]

Flooding has always been and continues to be a major problem around the world. Flooding has always occurred in the U.S., but the devastation associated with this problem worsened when settlers began migrating westward. Settlers built their homes

near rivers and streams for easy access to potable water. Soon the settlers had to endure

hardships due to the rising water. At first, levees were constructed to try to prevent

flooding near the major waterways. In the 1930's, the U.S. Army Corps of Engineers

oversaw the construction of very tall levees along the lower Mississippi River. However,

these levees were still not high enough to contain the devastating floods that occurred the

following years. These storm events caused an enormous amount of damage and a large

number of deaths. The Corps of Engineers was, therefore, originally more involved with

flood relief than with flood prevention. As a result of these devastating events, Congress

mandated the U.S. Army Corps of Engineers along with other federal agencies to study

and implement actions to try to minimize flooding along the major waterways. Federal

officials concluded that flood control was and should be a major responsibility of the

government. Consequently, in 1936 the U.S. Congress authorized the "Flood Control

Act." (USACE 1996)(41). This act directed several federal government agencies to find

solutions that would minimize the destruction and death toll caused by these natural

disasters. Investigations and studies subsequently recommended the building of

reservoirs with the primary purpose and major benefit of flood control. Since the 1936

Act, signed by then President Franklin D. Roosevelt, over three hundred multipurpose

reservoirs have been built with the main function of flood control. Local communities

and wildlife have also greatly benefited from the construction of these reservoirs

(USACE 1996) (41). At that time, the environmental impact of building man-made lakes

was not of much concern. There were little or no restrictions on where and how to build

these reservoirs.

Since the construction of large reservoirs, the U.S. Army Corps of Engineers and the Bureau of Reclamation have been required to regulate and control water releases to prevent flooding. However, the task has been complicated due to the fact that local municipalities, the Environmental Protection Agency (EPA), the National Wildlife Service, the Forest Service and other local and federal agencies pay for a certain amount of water storage for their general use (USACE 1996)[41]. Reservoir control regulators were soon faced with the problem of constantly monitoring lake levels and balancing water storage needs to appease all interested parties and at the same time maintaining some free-board for flood prevention. As an example, local communities pay for water storage so that there can be enough water to last through the summer periods when rainfall amount is minimal. The EPA, which is mostly interested in water quality, pays for storage so that water releases can be made during periods when water quality is poor. The EPA has mandated these releases through Congress for environmental protection reasons (Tiemann, Mary 1996). There are many more reasons for water storage, but all these controls placed on reservoir usage and storage have created the need for technical people to predict the amount of runoff that a certain storm will produce in order to regulate the amount of release from the reservoirs. Proper and timely forecasts of potential effective rainfall runoff can substantially reduce the possibility of severe flooding both upstream and downstream of a reservoir. With the highly interconnected system of rivers and reservoirs, forecasting of runoff is even more critical. Communications with the reservoir control teams and data acquisition personnel for precipitation data are very important in flood forecasting. The release of water from one reservoir has tremendous effects on the reservoirs and communities located downstream.

In a potentially devastating storm event, forecasters and regulators from several states, work around the clock trying to balance the system of rivers and lakes to prevent or minimize potential costly flood damages.

In the 1960's, the U.S. Army Corps of Engineers office located at Davis, California became the center of expertise on flood control. This Corps office known as the Hydrologic Engineering Center (HEC) was responsible for studying and developing methods to predict effective runoff from rainfall events. In 1967, Leo R. Beard and other members of the HEC developed the first computerized version of HEC-1 for the VAX machine, (USACE 1969) (44). HEC-1 is a compilation of various Fortran subroutines for predicting the various components of a hydrologic rainfall event. Since 1967 various revisions have been made to HEC-1, but the major subroutines for predicting runoff have remained the same.

Leo R. Beard and his staff reduced a 3-dimensional problem to a 1-dimensional solution with the HEC-1 software. Therefore, the task of predicting a flood with a single model is not feasible for large basins due to software limitations. The model uses overall average values for the entire basin which is not accurate. HEC-1 tries to combine temporal, spatial, and terrain variations into a singular average model parameter. With this method, it is almost impossible to use one model for every storm event. Forecasters predicting the results of a storm event have to be very familiar with the basin they will be modeling. This task requires years of experience. This is difficult for a young engineer to comprehend and to model without years of experience. It is also difficult for an experienced forecaster to understand the ever-changing dynamics of a basin. Land usage,

4

devastating fires, severe drought, and changing weather patterns are a few of the parameters a forecaster has to contend with.

The Tulsa District, U.S. Army Corps of Engineers is in charge of overseeing, regulating, and maintaining over sixty lakes and reservoirs around Oklahoma, Texas, Kansas, and Arkansas. Due to the recent downsizing of government personnel, the Tulsa District is currently involved in finding new and better methods for studying all aspects of hydrology and hydraulics for projects within the District. Detailed lake sediment surveys are being conducted to evaluate the loss of storage potential due to sediment loads. Electronic stream flow measuring devices are being used to study river flow and to understand the ever-changing dynamics of the river. Lake discharges through dam gates are being calibrated with the use of an acoustic discharge current profiler (ADCP). Within a year (2002) all the Corps of Engineers Districts including Tulsa District will be equipped with a multi-million dollar data gathering system specifically for hydrologic runoff forecasting. The "Water Control Data System" (WCDS) has been under development for several years by personnel at HEC under contract with all the Districts. As a result of developing the WCDS, HEC has converted the HEC-1 program from a DOS command software into a graphical user interface Windows program. HEC has recently concentrated on accurate and real time data gathering routines from gage data stations, also known as USGS gages. Data such as gage precipitation, temperature, solar radiation, evaporation, river-water surface elevation, and wind velocity and direction data are anticipated to be collected with the WCDS system in a more timely and consistent manner.

Although faster computers, larger disk storage devices, and more data are available, predicting a proper and timely forecast is still difficult and time consuming. This crucial mission aids a reservoir control regulator in determining when and how much water to release from a lake without impacting cities and towns along the waterways. Knowing the peak flow that a storm will produce hours in advance can potentially prevent devastating losses. Flood warning can be announced in advance if it is inevitable that a flood will occur. Local city officials can be notified of the impending danger and the approximate peak flow arrival time. Even though a flood disaster can never be eliminated, the costly effects can be reduced drastically with the aid of a proper forecasting method.

This study will examine a basin that has been difficult to model within an acceptable time increment. Doppler data will be used for the rainfall instead of the gage station at the mouth of the basin. This work will attempt to improve the flood forecasting methods currently in practice.

OBJECTIVES

The following objectives were identified for this study:

1) Establish a basin in Oklahoma that can be used for this study. The Glover Basin was chosen for this thesis because the basin has been difficult to model in a proper timely manner with current modeling techniques. Due to the basins large size of 325.25 square miles it has been difficult to determine appropriate average values for the HEC-1 model to produce a computed hydrograph which closely matches the observed hydrograph for any storm event before the peak flow

reaches the outlet of the basin. The new methods used by this report will be evaluated by Tulsa District in order to implement similar methods to other models.

2) Create a model that will be used with the HEC-1 software to aid a forecaster in predicting parameters for the HEC-1. Since HEC-1 or the new HEC-1 program called "Watershed Modeling System", WMS, will continue to be the standard for the Corps of Engineers for forecasting, this research will implement the new methods around this software model.

3) Determine a method that can better predict the peak flows quicker than current methods. Two methods of predicting some of the ever-changing parameters for the HEC-1 model will be used. The Monte Carlo method and Artificial Neural Networks (ANN) will be used in this thesis to determine if they can improve current methods forecasting river discharge and provide results to reservoir regulators so that they can determine appropriate reservoirs actions before the peak discharge of the storm is encountered.

4) Use Doppler data rather than gage data for the rainfall input. Doppler data will be a critical part of this study when using ANN to model effective rainfall amounts. Since HEC-1 is a one-dimensional model, an average basin rainfall amount will be used for the actual HEC-1 model for the Monte Carlo method. The discrete data precipitation from the Doppler will be used for ANN method.

5) Use ANN to predict several hours of effective runoff from any storm event without the use of HEC-1. This will give the forecaster a target to achieve for their HEC-1 model simulation.

GLOVER BASIN

The Glover Basin is located in the southeastern portion of Oklahoma in McCurtain County, figure 1.1. The population of McCurtain County is approximately 35,000 as of 1999, (U.S. Bureau of Census, 1999). Glover Creek is the main river channel that runs through the basin. The water from Glover Creek eventually joins Little River, which runs downstream towards Horatio Reservoir located about 10 miles east of the Oklahoma-Arkansas border, figure 1.2. The basin is composed of mixed forestland, cropland, and pastureland. The upper portion of the basin is steep evergreen forestland managed by the U.S. Forest Service. This basin is a sub-basin of an even larger forecast model, the Horatio Basin Forecast model. This basin was chosen as the model for this study due to the difficulty of modeling the effective runoff during a storm event. Several conventional models have been developed since 1991 to try to predict runoff at the mouth of the basin, but like other forecast models, one model cannot predict the runoff for every storm. Several detailed studies have been conducted to build a HEC-1 model. Several alternative methods offered by the HEC-1 program have also been tried. Due to the complexity of this basin the model parameters cannot be predicted accurately and a timely forecast is not easily achievable.

LIMITATIONS

Three years of hourly radar data have been collected for this area of Oklahoma. Only one gage station is located in the Glover Basin. The gage station is located at the mouth of the basin. Some of the gage data are missing due to either malfunctioning gage equipment or with the computer data storage equipment. Historical HEC-1 calibrated

8

McCurtain County

Oklahoma Counties

N
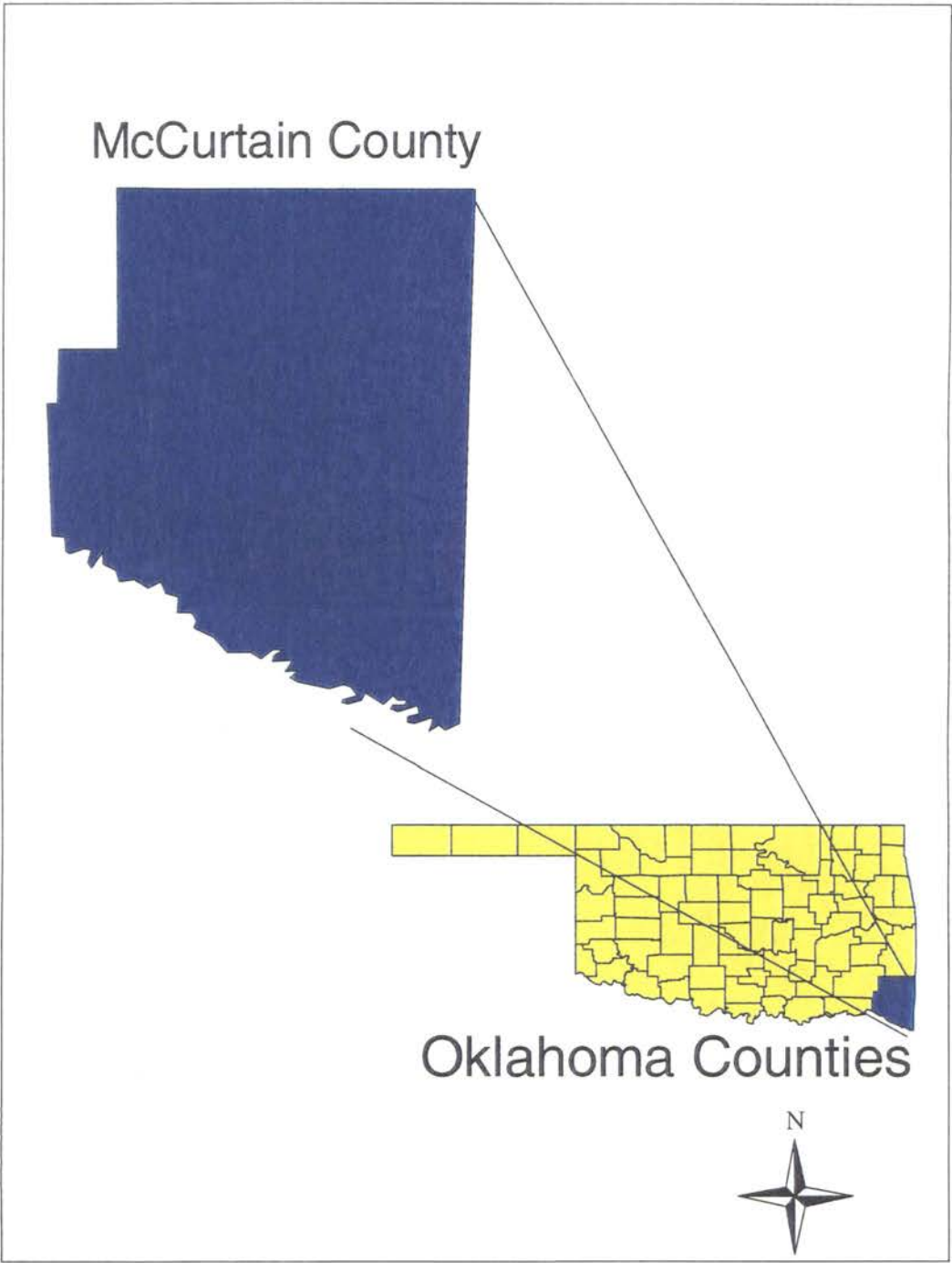
Figure 1.1. McCurtain County located in the southeastern portion of Oklahoma.
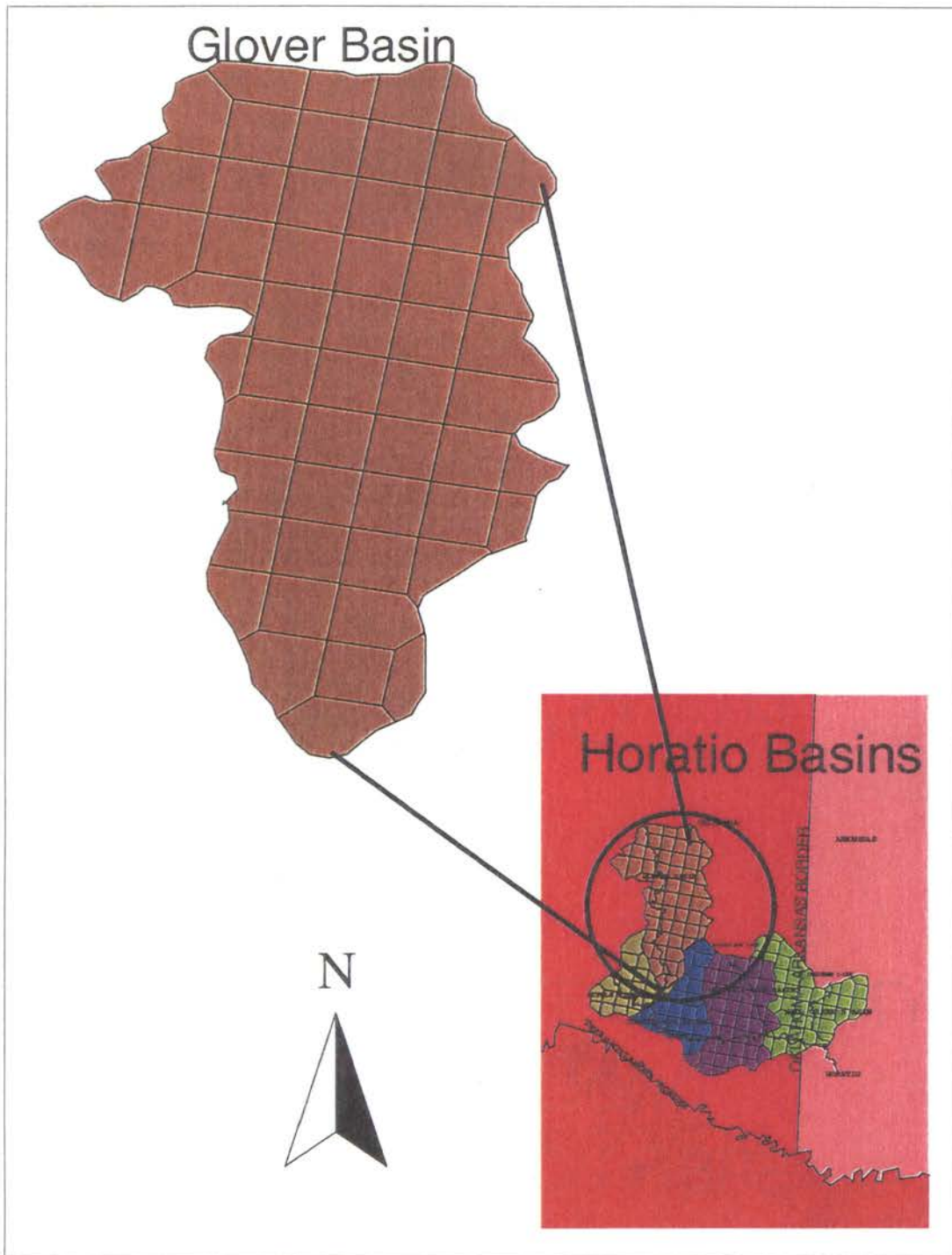
Figure 1.2. Location of Glover Basin within the Horatio Basin Model.

models for storm events during these three years were not available. After a rainfall

event has expired, the final forecast models used are discarded and not used again.

CHAPTER 2

USING DIGITAL DATA AND GIS TO BUILD A HEC-1 MODEL

INTRODUCTION

In early 1996, after Vice President Albert Gore's summit meeting for conservation of our natural resources, Congress mandated all federal agencies to reduce paperwork as a means of conserving our natural resources and provide all pertinent data digitally (EnviroSense 1996). This required that all agencies of the federal government provide contracts, bids, survey data, pertinent land data, and all other data in a digital or electronic format rather than paper format. At first, it was difficult to determine the standards for providing the data. It was also a requirement that very few federal employees knew how to implement. The requirement was not only to provide the data, but to provide the "meta-data" as well. Meta-data was a new term that meant more work for the agencies. Meta-data is a term used for providing a detailed description about the digital data being distributed at a cost or free. This includes information as to who or what company collected the data, the dates the data was collected, the coordinate system in which the data was collected, a point of contact for obtaining the digital data, the quality of the data, and any other description pertinent to the data. Agencies began to fulfill the requirement and provide their own data and standards along with some meta-data about the data they were disseminating. With this mandated order of reducing paper documents and the ever-exploding Internet/Intranet technology, digital data of all types was slowly becoming accessible to individuals with a desktop computer or a workstation

capable of accessing the Internet. Since 1999, two national meetings have been held at

the University of Texas to address the problem of standardizing the format for providing

map or land-based digital data. Among some of the attendees were members of the

USCE, USGS, HEC, EPA, National Data Clearing House, Environmental Systems

Research Institute Inc. (ESRI), the Waterways Experiment Station, and various other

federal, and local (Texas and Oklahoma) municipal engineers and technicians. The most

significant problem with agencies providing their own standards is that it has become

very difficult to overlay pertinent data from several sources due to different projections or

formats. Debates and articles are ongoing on how to code digital data. Each federal

agency and private entity have their own justifiable reason for developing data in a

certain format. Therefore, there is a redundancy of data collection because of

inconsistent and incompatible formats.

PREVIOUS METHODS OF BUILDING A HEC-1 MODEL

Before 1999 building a hydrologic model for the HEC-1 was time consuming and

tedious. At first, several 1:250,000 scale paper topographical maps were used to get a

general idea of the basin being studied. Once the basin had been delineated on the

1:250,000 scale maps, better scale topographic quadrangle maps were obtained from the

United States Geological Survey (USGS). The 1:250,000 scale maps with fifty-foot

contour intervals were not and still are not adequate enough to obtain accurate parameters

for the HEC-1 model. The basin outlined was then transferred to the 1:24,000 scale map

which have ten-foot contour intervals for better accuracy. Once the basin was outlined

on the 1:24,000 maps someone had to manually planimeter the basin to obtain the area

13

and length of each stream. This task took several days or even weeks. Due to the inevitable human factor each map had to be planimetered three to four times. This ensured that any human error would be minimized. After each map was planimetered, an average area for each map was obtained and then summed to obtain the total area for the basin of study. Similar tasks were used to obtain some of the other basin parameters required for the HEC-1 model.

In the mid 1990's, the USGS began to produce digital elevation maps (DEMs) for Oklahoma. Initially, the USGS began to develop DEMs for the 1:250,000 scale topographic maps. Later they worked on developing them for the 1:24,000 scale maps. At present time, DEMs for a major portion of Oklahoma are available. The USGS has made this data available to those with a computer and Internet access free of charge. Due to the availability of these DEMs through the Internet, commercial software vendors began to produce computer macros or computer scripts that could read and utilize these DEM's. Bently, producer of Microstation®, then Environmental Systems Research Institute (ESRI), producer of ArcView® and ArcInfo®, developed scripts for their graphical packages for reading and manipulating DEMs. As a result of these packages and data, basin delineation can be done quicker and more accurately than with previous methods. Due to the tremendous downsizing and reduced funding, hydrologic and hydraulic engineers are being forced to conduct studies with this data and software packages.

AVAILABLE DATA

The USGS has currently produced approximately 90% of the 1:250,000 scale DEMs for the entire United States. However, these 1:250,000 DEMs are not accurate enough to use to develop the HEC-1 model. The USGS is continuing to develop digital elevation models (DEM) at a scale of 1:24,000 for the entire United States. A major portion of Oklahoma has been developed at this scale. Because the Glover Basin is a large basin, a total of 14 DEM data files of 1:24,000 scale are necessary to encompass the area. The USGS has also taken the responsibility of producing digital line graphs (DLG) of the major streams and rivers for the entire U.S. The data produced by the USGS is available free from their web site for downloading at http://www.usgs.gov.

The EPA, in their interest for creating a better environmental analysis systems tool, have gathered an enormous amount of data pertaining to watershed and water quality based studies. They have compiled available data for each region of the U.S. They have developed a GIS tool called "Better Assessment Science Integrating Point and Nonpoint Sources" (BASINS) to support engineers and scientists in analysis of environmental issues (USEPA, 2000). Included in this data set are the State Soil and Geographic Database (STATSGO) data that was used for this study (USDA 1994). The EPA has converted all their data from paper to digital format and made it available to the public at no cost through EPA's web site at http://www.epa/ost/basins.

DEVELOPING BASIC HEC-1 PARAMETERS USING GIS

Both ArcView® and ArcInfo® software programs developed by ESRI were used to develop the basic parameters used for the HEC-1 model. The DEM data obtained from

the USGS website were imported into the ArcInfo® software to develop a grid of elevation data. Before the DEM data could be used, all the USGS data for this project had to be converted into one projection and one system of units. The code used by the USGS to develop these DEMs was obtained and modified so that all the data could be converted to the Universal Transverse Mercator Projection(UTM) zone 15 and all elevation data to feet. Since most of the data for this project was published as UTM zone 15, the DEM data provided by the USGS in decimal degrees was converted to UTM zone 15 as well. Later the modified data was converted into a 100 ft. grid cell for the purpose of modeling instead of the typical metric grid cell. The data provided by the USGS incorporate several different coordinate systems. The latitude and longitude measurements were provided in either decimal degrees or metric units while the elevation measurements were provided in US customary units, feet.

With the aid of two hydrologic researchers from the University of Texas at Austin, ESRI developed several subroutines for their software to delineate basins and to obtain stream data from the USGS DEMs. The Glover Basin was delineated using the aforementioned subroutines (Figure 2.1) and compared to the basin previously created from the 1991 Glover Basin study conducted by the Tulsa District COE. The difference between them was minimal at the 1:24,000 scale, but the difference was much greater at the 1:250,000 scale. In the 1:24,000 scale topographic maps, missing data was encountered at the adjoining edges of each quad area. Figure 2.2 shows the location of black squares and rectangles indicating that there is no data for that portion of the grid. Therefore, linear interpolation method was used to fill the gaps of each quad. Filling in of missing data was required so that the software could delineate the basin correctly
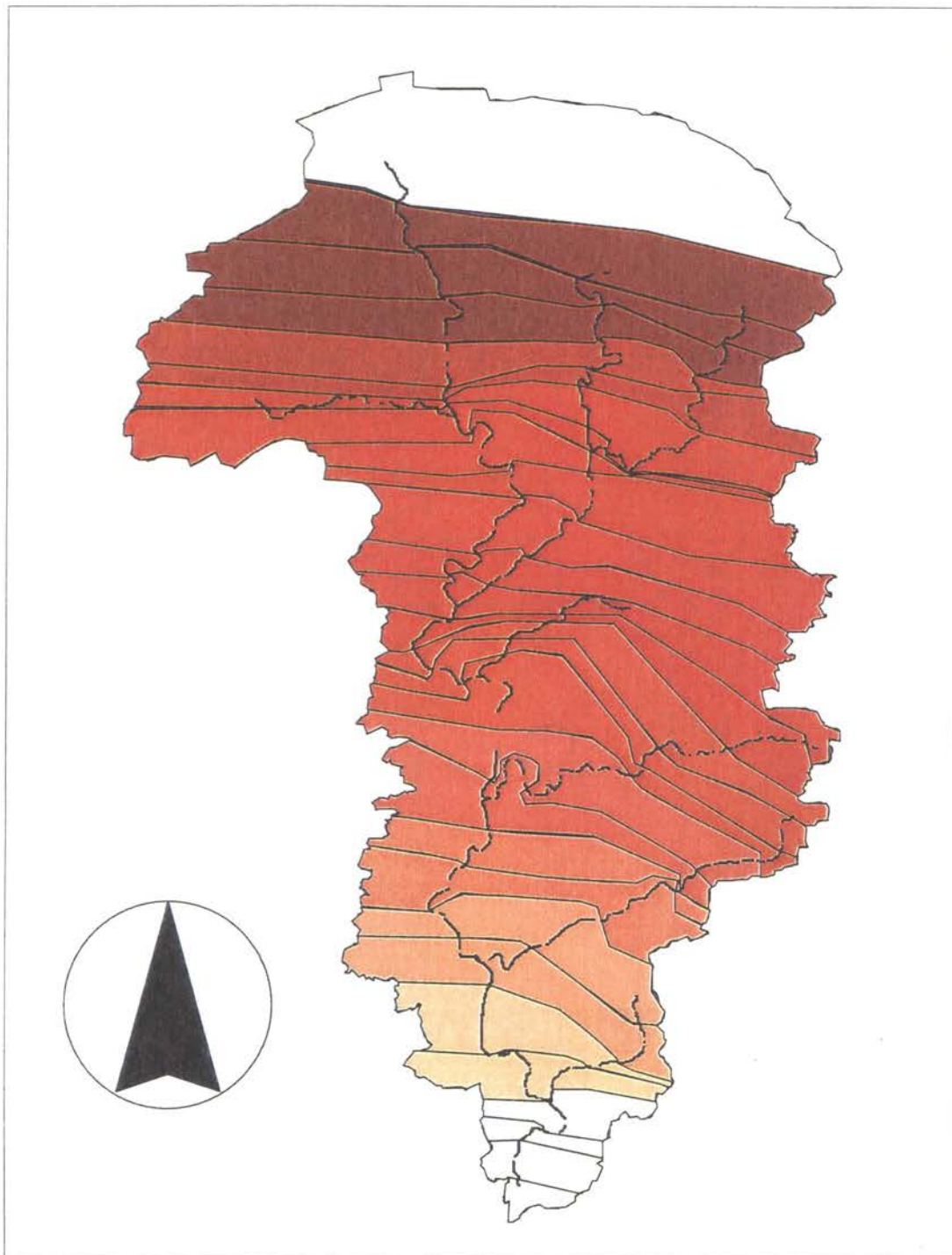
16

Figure 2.1. Glover Basin delineated using USGS 1:24,000 DEM.
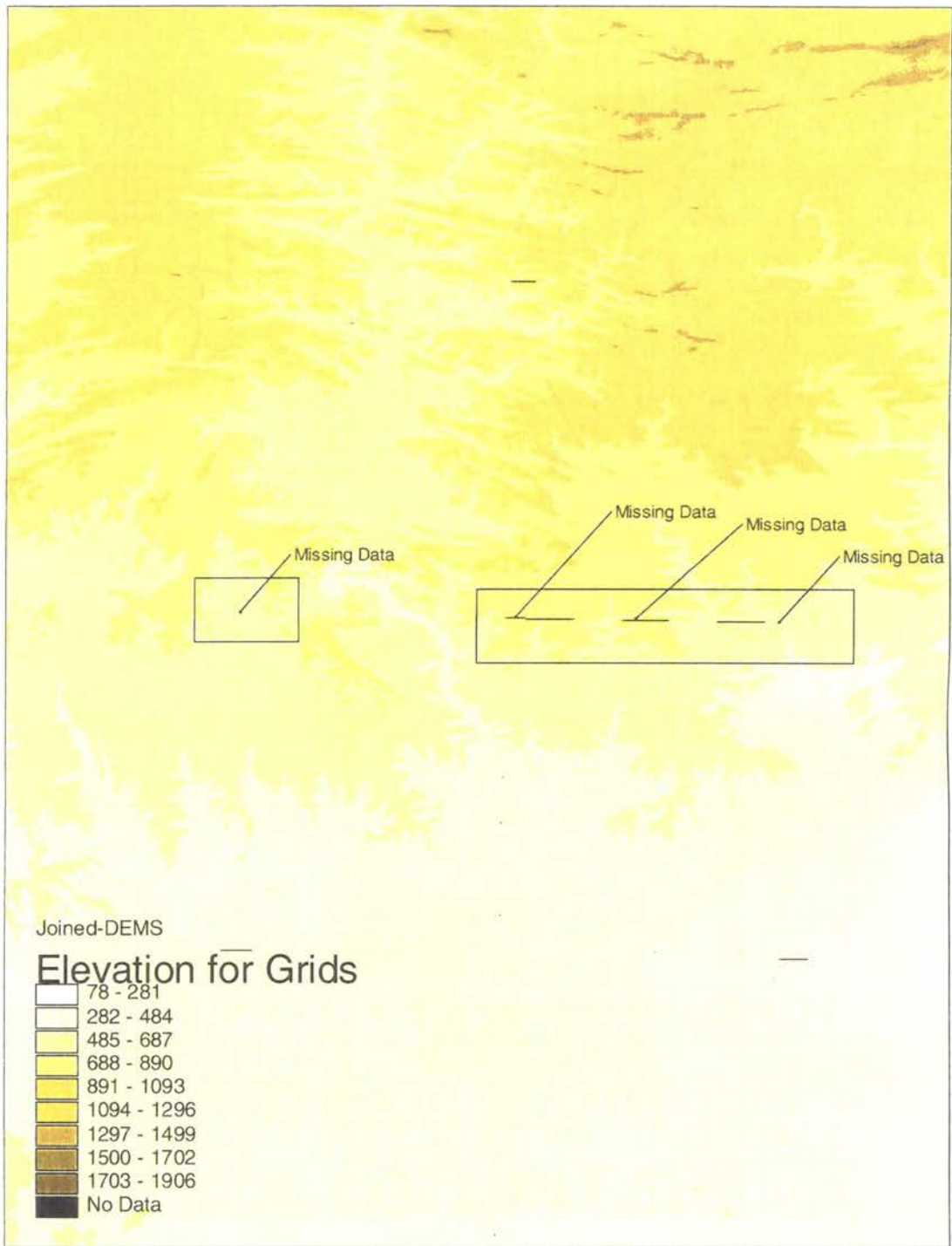
Figure 2.2. Joined USGS 1:24,000 DEMs showing the missing data at the joining edges. The black lines and dots are points with no data.

otherwise the software would terminate the basin delineation at the first occurrence of

missing data. Further studies conducted by Maidment and Olivera , discovered that these

DEMs had to be "filled" for better and proper delineation of the streams (21). Although,

the DEM data were constructed with data points spaced every 100 feet apart, Maidment

and Olivera determined that some elevation points in the grid showed a much lower or a

much higher elevation than realistically possible. To correct for this discrepancy in the

data, they worked with ESRI to develop a method of correcting the grid to a "most

probable" elevation grid. Their method of correcting the grid was to eliminate the high

and low values and replace them with an average value from the surrounding data.

Judgment by the engineer or researcher has to be made to determine the correct tolerance

for the maximum range in high and low elevation. After the elevation grid is corrected, a

flow grid and a stream grid can be created for the basin of interest. Figure 2.3 shows

segmented lines on the grid that represent the major streams, computed by the software

based on the corrected and manipulated DEMs. Figure 2.4 shows a grid of streams

showing all the minute paths that water will take to finally flow toward the mouth of the

basin. Figure 2.4 is a very detailed grid of streams for the basin. Several iterations of the

grid elevation correction routines were required in order to better match the streams

obtained with the software to that of the actual paper quad maps or previous detailed

study.

After the basin and stream delineation was obtained, the EPA data was projected

to fit the Glover Basin. Land use and soil type data was graphically depicted for the

Glover Basin using the ArcView® software. Figure 2.5 is a graphical representation of

land use for Glover Basin. A numbered coding system is used by EPA to determine if the
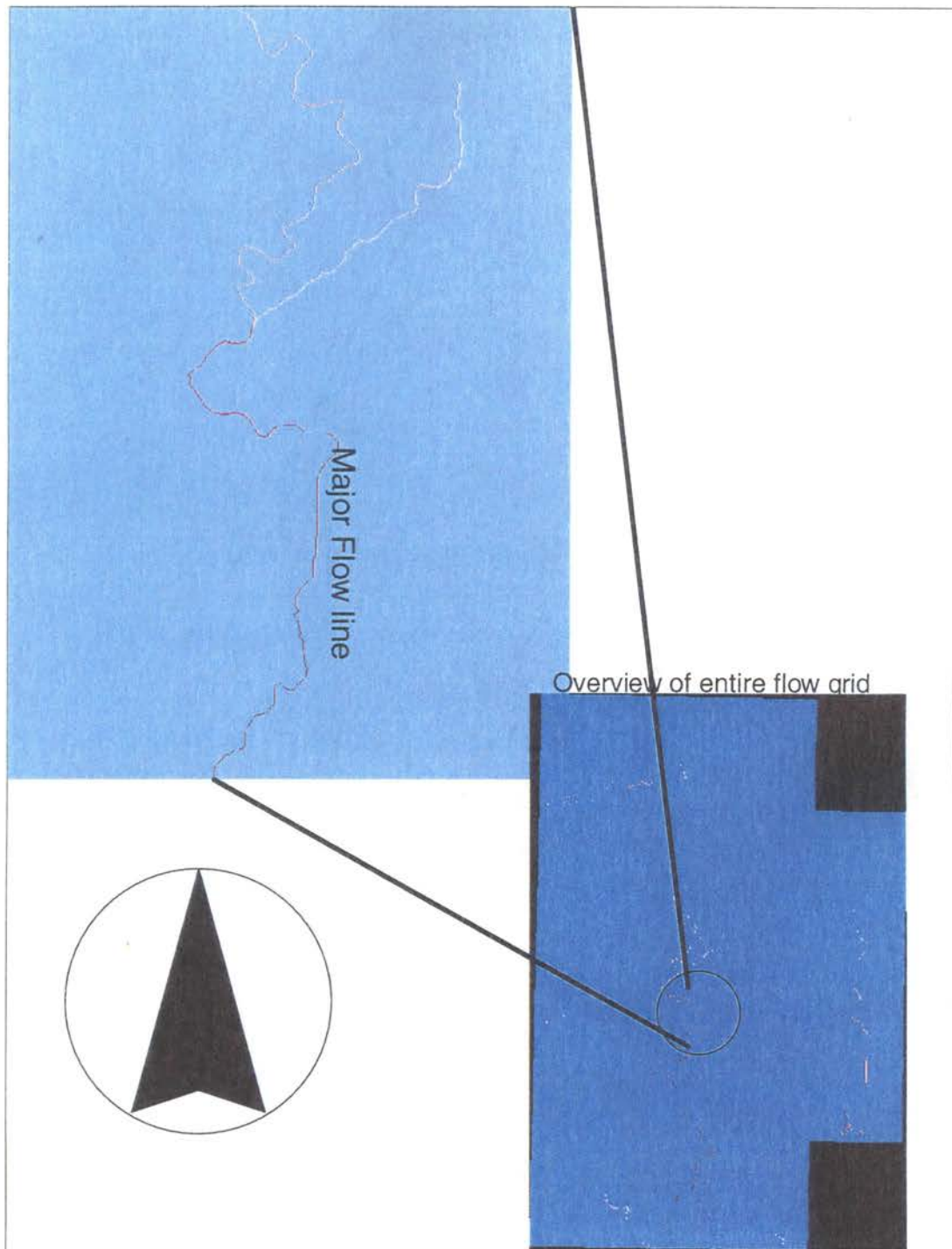
Figure 2.3. A flow accumulation grid shows the major streams for the area. A Closer look at this grid shows a continual segmented line. The overview shows Separate line segments for the stream which is just a visual misrepresentation.
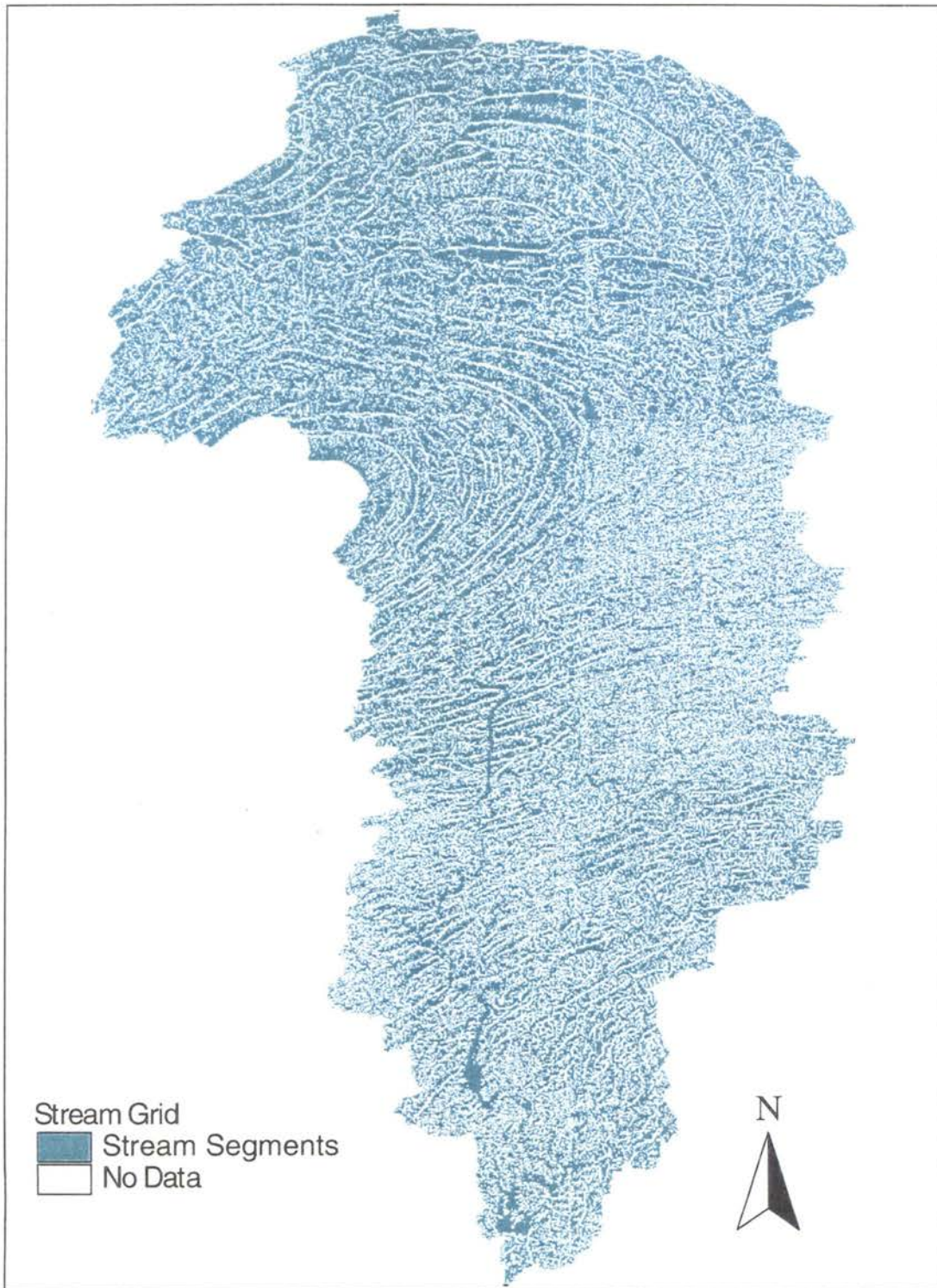
Figure 2.4. Stream grid of Glover Basin. The light blue color are line segments For all the streams that ArcView computed from the DEMs.
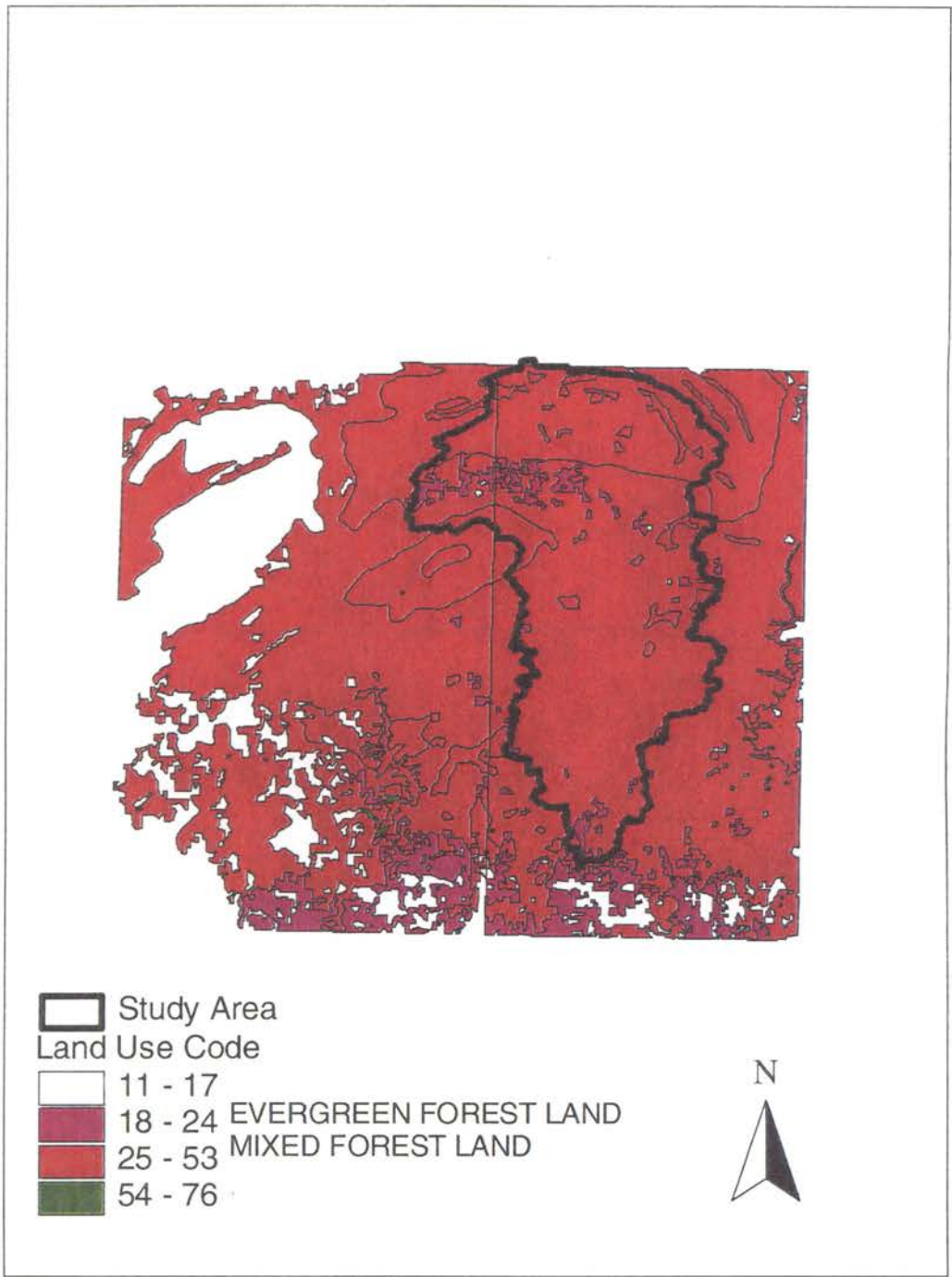
Figure 2.5. Land use data obtained from EPA for the Glover B asin. The data Indicates that the Glover Basin is mostly a mixed forest land, with some Evergreen scattered throughout the basin.

land is used for agriculture, business, or urban development. Figure 2.6 shows graphically the soil type in the Glover Basin. These data were used by the BASINS software to determine maximum and minimum infiltration rates for the basin of interest. These data were used for obtaining the range of possible infiltration rates for the HEC-1 model. These parameters will be discussed in the next several sections.

TIME-AREA CURVE

The HEC-1 software has several methods for calculating basin runoff. These methods fall into one of two types: (1) hydrologic and (2) hydraulic. Funding and project time mostly determines the type of modeling to conduct. As a result, very few models are done using the hydraulic method. The hydrologic model is based on the unit-hydrograph technique. Three methods for developing a synthetic unit hydrograph are available in HEC-1. The Clark Unit Hydrograph method, the Snyders Unit Hydrograph method, and the Soil Convservation Service (SCS) Dimensionless Unit Hydrograph method are available to the user. Internally the Synders method is transformed to the Clark method by HEC-1 software. The SCS method is the least used method by forecasters in the Tulsa District or by southwest Division which includes the Fort Worth District, Little Rock District, and the Galveston District. Since most of the models in the Tulsa District are run using the Snyders method, this study will try to improve on this method of forecasting. Three parameters, the time of concentration, storage coefficient, and a time-area curve are required for the Snyders method.

Time of concentration, as described by Victor M. Ponce, is the time that it takes a drop of water to travel from the farthest point in a channel of a basin to the outlet of a basin or stream junction. Many studies have been conducted trying to determine this
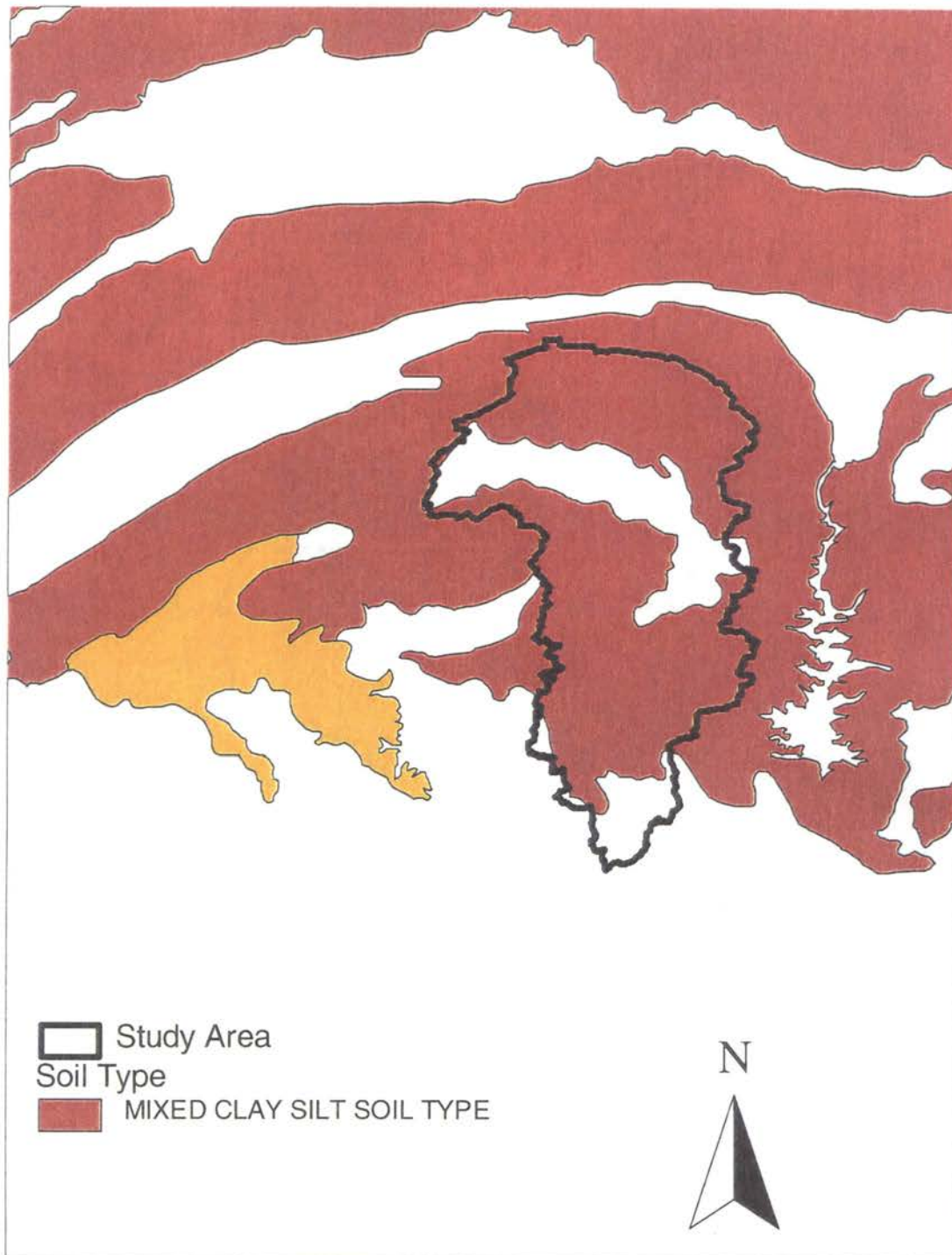
Figure 2.6. Soil type data obtained from EPA's website. The Glover Basin is composed of clay and silts. There are some areas of the basin where there are no data on the soil type, however it will be assumed that the entire basin is of the same material.

parameter. The time-area curve is an extension of the time of concentration. The time-area curve essentially transforms an effective rainfall hyetograph into an outflow hydrograph. HEC has also studied this parameter for several basins and has recommended the use of an empirical equation when no other data is available (USACE 1969) [42]. When a time-area curve is not supplied, HEC-1 uses the following equation to develop the curve:

$$AI = 1.414T^{1.5} \qquad 0 <= T < 0.5 \qquad (2\text{-}1)$$

$$1 - AI = 1.414(1\text{-}T)^{1.5} \qquad 0.5 < T < 1$$

where;

AI is the cumulative area as a fraction of total sub-basin area and

T is the fraction of time of concentration.

The major importance of the time-area curve is that it can account for temporal rainfall intensity giving more weight to the location of heavier rainfall.

The development of a time-area curve is a simple concept but difficult to construct, especially for large basins. The basin's time of concentration is divided into a number of equal time intervals. This time interval is usually 1-hour intervals for forecasting purposes. ESRI briefly discussed the issue of the time of concentration in their recently published book, "Modeling Our World." However, they did not discuss or demonstrate how to use ArcInfo® or ArcView® to build a time-area curve. In building the time-area curve, the basin is divided based on the 1-hour interval time of concentration. The area of the basin that produces a certain time-interval is then calculated. This value is then entered into the HEC-1 model. Hathaway's formula for

25

"tc" was used for this study to determine the time of concentration. The formula and terms are defined as follows:

$$tc = (0.606)*(L*n)^\wedge(0.467)/(S^\wedge 0.234)$$ (2-2)

where,

tc is time of concentration in minutes

L is length of watercourse in feet

n is roughness factor

S is slope in feet rise per foot length.

BUILDING THE TIME-AREA TABLE

In cooperation with the University of Texas, ESRI wrote a routine for developing a slope grid using DEM data (ESRI 1999). Using ESRI's ArcInfo® Macro Language 'AML', a routine to calculate slope grid was developed for the Glover Basin, see figure 2.7. A slope grid is a grid that contains slope information for each cell. For this study, the cell is approximately 100 ft by 100 ft. The average slope for each cell is calculated by the software and the information is presented graphically. Included in the newest ArcInfo® software is the ability to better manipulate gridded data. This method was used to obtain the data needed to build the time-area curve. The streams for the Glover basin were separated into separate coverages. A conditional statement was used to subtract or eliminate all the slope data not pertaining to an stream. After the gridded data for the stream and slope were obtained, they were converted into shape files for visually manipulating the data to obtain the time of concentration for each stream-line segment.
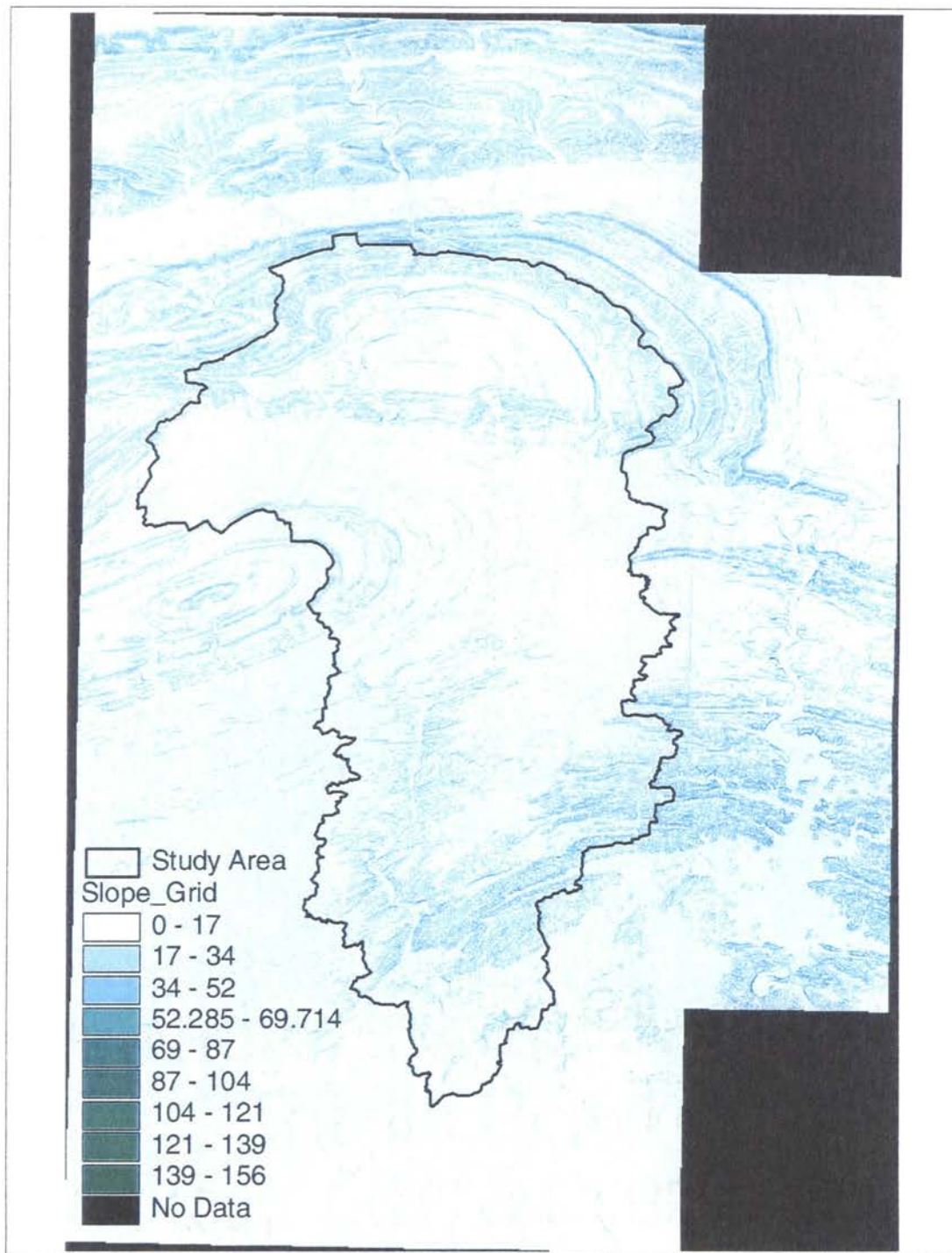
Figure 2.7. Slope grid for the Glover Basin. This figure shows a color coded grid of the slope for each 100ft by 100ft cell size. This information was used with the stream segments to calculate the time-area curve.

Several database tables containing the length of each line-segment for the streams, and the slope data for each cell were obtained. In converting the ArcInfo® files to an ArcView® shape files, the software did not produce files that could be manipulated easily. The stream lines were broken into straight line reaches, while the slope was broken by the 100ft cell size. By using ArcView® Avenue ®scripting language, a program was developed to match the slope cells to each stream line. Hathaway's formula was then used to obtain the time of concentration for each stream line by averaging the slope for each stream line segment. A value of 0.06 was used for the n value in the formula. This roughness coefficient was obtained from previous studies conducted for the Horatio Basin. Tables A.1 through A.16 show the tables used for the calculation of the time-area curve. Micrsoft Excel was used to implement Hathways 'tc' formula to the tables. Tables A.1 through A.10 can be found in appendix A of this report. Figures 2.8 show the final breakdown of the time-area curve for the Glover Basin and Figure 2.9 shows a graph of the time-area curve.

SUMMARY

Building a hydrologic model from available data is becoming easier and faster with the new GIS software and readily available data. In 1991 it took several technicians weeks or even months to build a HEC-1 model. Now it takes approximately 8 hours to build a functional HEC-1 model. Instead of concentrating on digitizing a basin or plenimeter a basin by hand, the engineer is mostly focusing on converting all digital data to one consistent projection and unit of measure. Software vendors such as ESRI are currently tackling the problem of trying to merge the data automatically into one
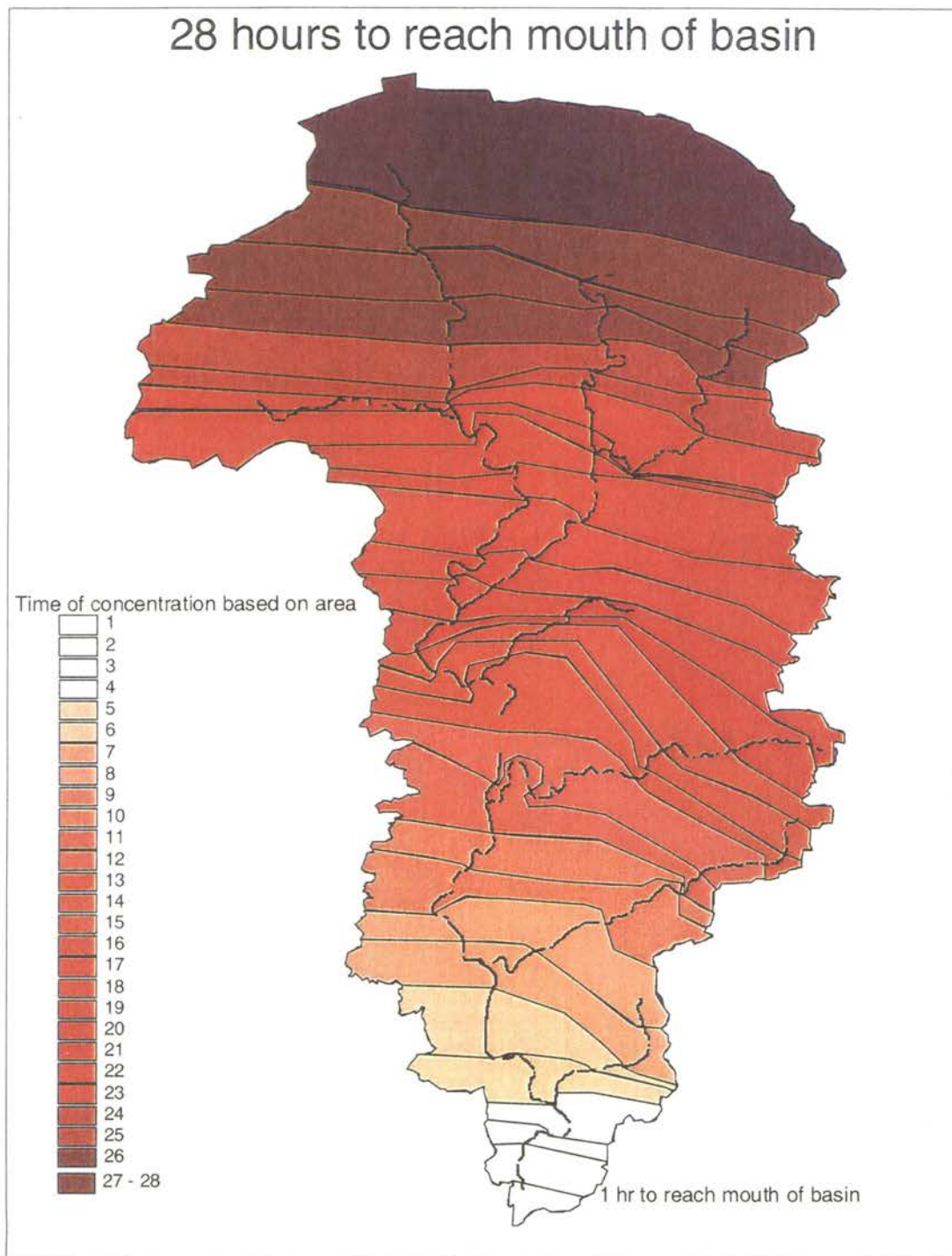
Figure 2.8. Time-area graphical display for the Glover Basin. The different segments of the basin are divided into one-hour increments starting at 1 hour at the mouth of the basin to 28 hours at the top of the basin. This indicates that any rainfall within a segment of the basin will have a time of concentration between the lower value and the higher value of the segment.
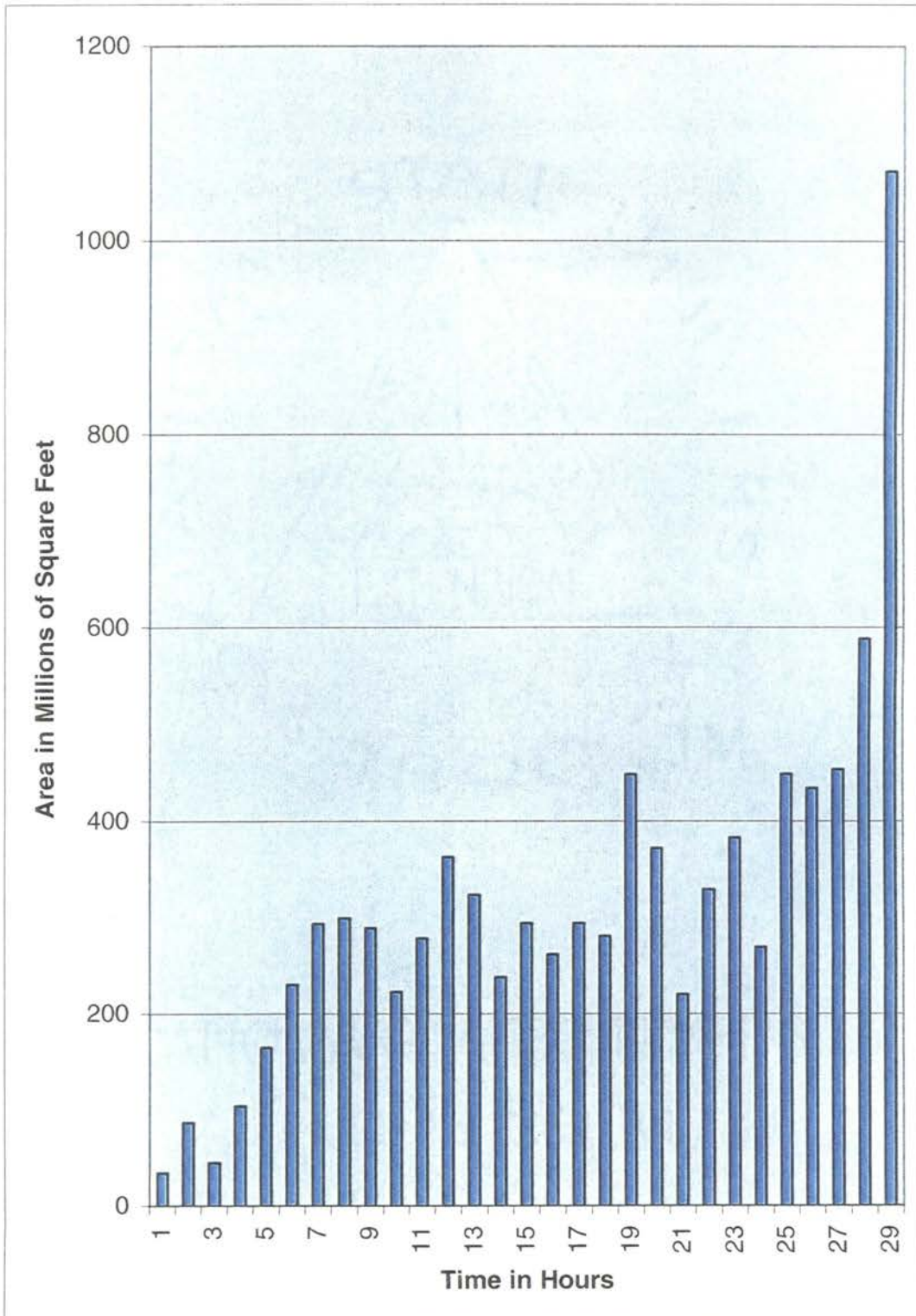
Figure 2.9. Time-area curve chart for the Glover Basin.

projection and linear unit. ArcInfo® 8.0 currently re-projects imported data automatically to match the existing working project. Blue Marble® has also written a software tool just for converting data from one projection to another. The Corps of Engineers has produced a conversion utility for point data, and members of the USGS have also written several ArcView® scripts for converting their DEM data into other projections.

The parameters for Glover Basin were created using the new GIS software and scripts and techniques described by Olivera and Maidment (1996). This new method of building the HEC-1 model generated a basin area of 325.25 square miles. The hand planimetered 1991 HEC-1 model generated a basin area of 296 square miles. The difference can be attributed to human error with the planimeter. The 1991 HEC-1 model used the general equation for the time-area curve due to the complexity and tedious work of producing the curve for the basin. With ArcInfo® and ArcView®, a time-area curve was created in a matter of hours. These new techniques are tremendous time savers for an engineer, especially with the downsizing of the government infrastructure. Currently Ph.D. students at the University of Texas at Austin and at Brigham Young University are conducting research on making this process easier and more accurate. Eventually, it will be possible to create a surface water model using the Internet by combining all pertinent data on a single web page.

CHAPTER 3

A PRELIMINARY APPROACH TO FLOOD FORECASTING

INTRODUCTION

Hydrology is the study of the waters of the earth. It focuses on the various

interactions that the water cycle goes through as it travels from the atmosphere to the

earth and eventually traveling back towards the ocean again. The non-ending,

reoccurring hydrologic cycle is shown in Figure 3.1. Hydrologists and engineers are

continually studying this cycle to gain better understanding of the various interactions of

this physical process. Climate conditions, soil type, terrain extent, and vegetation types

are some of the parameters being studied as well as their interactions with one another.

These parameters are constantly varying both in space and time. This dynamic situation

makes any model very difficult to simulate real world conditions. Flood forecasters try to

predict or interpret the space-time variability of these parameters in general, to better

estimate the effective runoff of specific storms of great magnitude in an effort to prevent

costly disasters or human losses.

The physical model adopted by the U.S. Army Corps of Engineers for forecasting has

been the HEC-1 model. As pointed out by Woolhiser(1996) "We must keep in mind that

all models are simplifications or abstractions of reality and all models are to some extent

wrong. In fact, if they aren't simpler in some sense than the real-world object, they aren't

useful!"
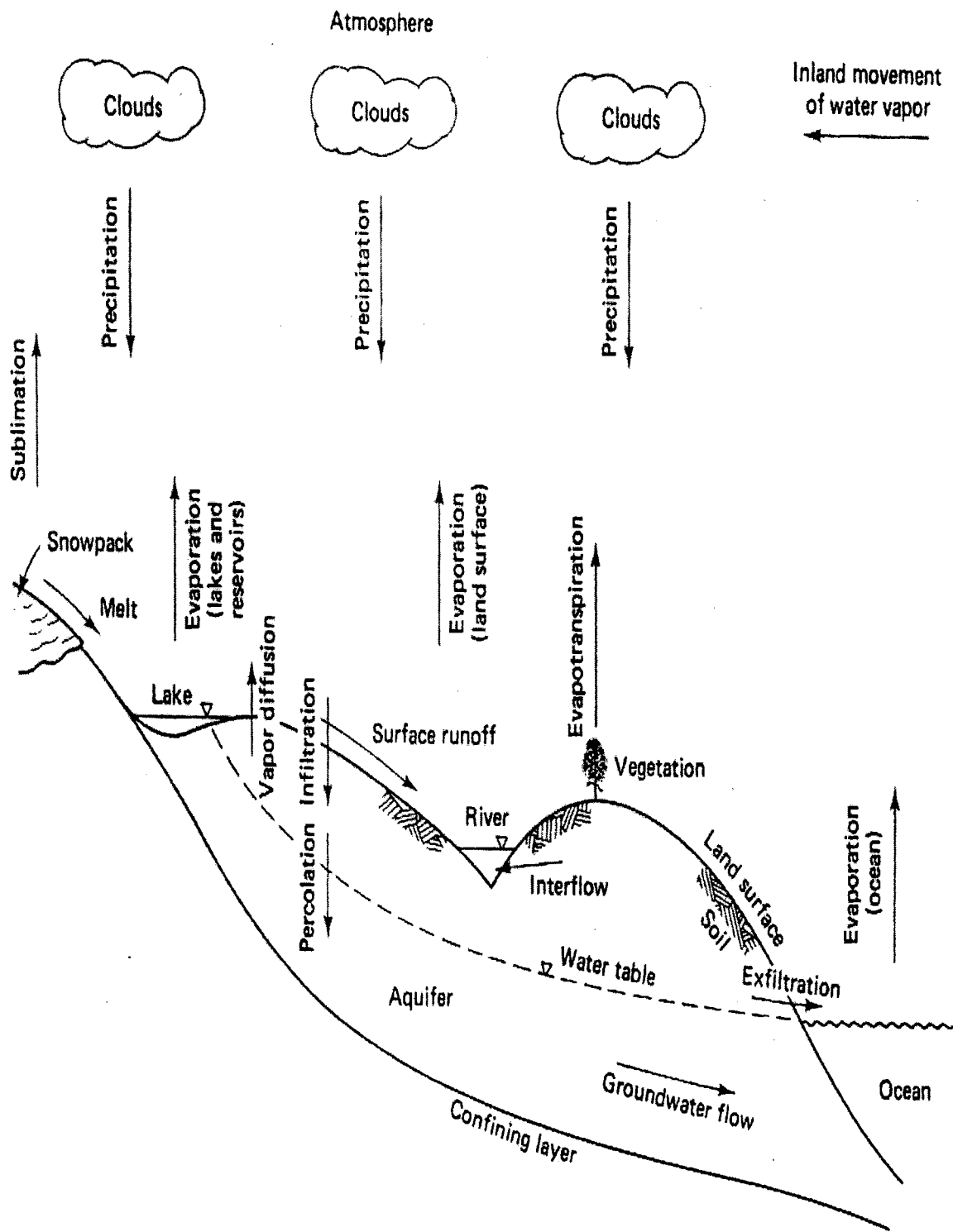
Figure 3.1. Typical process of the hydrologic cycle. (Ponce 1989)

33

There are many distributed physical based models being studied at present time by notable researchers. These models have immense potential and utility in, "forecasting the effects of land-use change; forecasting the movement of pollutants and sediment; and forecasting the hydrological response of ungaged catchments." (Beven, 1985) These models also have a disadvantage because they require a lot of data, a lot of computer usage, a lot of computer memory, and a lot of time to produce a good quality model. However, there is not much improvement gained with these types of models as compared with the typical HEC-1 numerical models at this time.

In this study some aspects of the distributed model have been combined with the lumped physical model to generate a better forecast in a more reasonable time period than currently done. Since the Corps of Engineers will continue to require the use of HEC-1, it has been necessary to base this study around the HEC-1 software.

STATEMENT OF THE PROBLEM

Predicting the flood peak and the time that the peak flood will occur is one of the major tasks of flood forecasting. Knowing this information can cause reservoir control regulators to initiate actions in order to prevent or minimize the flooding impact on communities both upstream and downstream of a reservoir. Prediction of these two variables (time to peak, and peak discharge) is difficult with each storm event. In a storm event, a forecaster can begin comparing the output of his model to that of the observed runoff data after the rising limb of the hydrograph has taken some shape. Many times the observed hydrograph begins to take a shape at least 8 to 16 hours after the storm has elapsed. After this time lapse the forecaster can begin comparing the true hydrograph

with that of the computed hydrograph. The forecaster can then begin adjusting some of

the HEC-1 parameters to try to match the observed results. Smith et al. (1994) showed

that it is difficult to predict peak discharges in controlled experiments. Smith et al. found

that under nearly identical situations of a controlled experiment, on an impervious surface

area there was a 4% difference in peak discharge. He concluded that determining peak

discharge of a natural system is even more difficult. In 1994, a sprinkling plot

experiment conducted at the Bernardino Plot 164 at Walnut Gulch in Arizona, Smith et

al. (1994) found that peak discharges for nearly identical initial conditions, varied by as

much as 35%. This illustrates the difficulties a forecaster has in determining the peak

discharge as well as the predicting the time the peak discharge will occur.

BACKGROUND DATA

The HEC-1 model for the Glover Basin was built using available digital data. The

data were imported into ArcInfo® and ArcView® to obtain the basin's basic parameters.

EPA's BASINS data was imported into ArcView® to obtain the minimum and maximum

hourly infiltration soil rates. The final HEC-1 model used for this research is shown in

table 3.1. The minimum, maximum, and average dynamic parameter values are shown in

table 3.2

One USGS precipitation-recording and stage-recording gage were located at the

mouth of the Glover Basin. The precipitation data from this gage were not used because

it had several periods of missing data during the years of 1997 through 1999. The

Doppler data collected by the National Weather Center at Tulsa was used for

precipitation. This Doppler data wer extracted for the Glover Basin using several

```
ID     Glover Basin HEC-1 Model
ID     Snyder Unit Hydrograph Method with a time-area data. Jan 2001.
ID     Precip Data obtained from Radar and a basin average rainfall computed.
ID     Basin average rainfall amounts stored incrementally from Jan-Dec in DSS
IT   60 12MAY99    0100     54
IO    1      2
OU
KK   GLOV GLOVER BASIN
ZR=QO B=GLOV C=FLOW-LOC CUM F=OBS
BA325.25
* BF 848     -.35    1.1
BF  2137     -.25    1.10
PB     0
ZR=PI B=PRECIP C=HRAP F=OBS
US  18.9    0.94
UA     0    1.22    4.32    5.92    9.63   15.52   23.76   34.29      45   55.35
UA 63.33    73.3   86.31   97.90  106.43  116.96  126.33  136.85  146.90  162.98
UA176.31  184.20  195.99  209.70  219.36  235.45  251.01  267.30  288.38  325.25
* LU .74      -1
LU  0.47    0.11
ZW B=GLOV C=FLOW-LOC CUM F=CALC
ZZ
```

Table 3.1. Basic HEC-1 input file format for the Glover Basin.

Parameter	Tp	Cp	Strtl	Cnstl
Min	9.53	0.38	0	0
Max	24.5	0.96	1.55	0.11
Ave	15.76909	0.691818	0.534545	0.041364

BASINS Permeability values

Min	0.027
Max	4.4
Ave	2.2135

Table 3.2. Minimum, Maximum, and Average values for the dynamic parameters used for the HEC-1 model for the Glover Basin. These values obtained from manual calibrations of the HEC-1 models with historical storm events.

programs written by the Corps of Engineers at Tulsa. A software package called CEnvic32, produced by Nombas Inc. was used to create 'C' type scripts called Cmm scripts for this project. The Cmm scripts were similar to coding in the typical 'C' language format. The major advantage of using scripts as compared to executable programs is that the scripts can easily be edited with Notepad and run without compiling the code every time a change is made. Several Cmm scripts were written to automate the process of converting the hourly Doppler data into point data for ArcView®. Another Cmm script was written to convert the point data into an average basin rainfall amount for the HEC-1 model. These later data were then converted into a Data Storage System (DSS) database format for easier HEC-1 manipulation. DSS is a database produced and purchased specifically for the U.S. Army Corps of Engineers for maintaining rainfall and gage data. Appendix B shows the written code for the script files.

Doppler data of Stage III type have only been around for the last three to four years. Stage I and Stage II data have been collected much longer than Stage III data, but these data were used mostly for experimental purposes or graphical representation of the rainfall event. Stage I and Stage II data are not reliable and are not very useful for mathetical models, especially for this research project. Stage III data are Doppler data that have been calibrated, adjusted and warped to fit the data from the physical gage stations around Oklahoma. The River Forecast Center in Tulsa has used several programs developed by the U.S. Army Corps of Engineers in Tulsa to warp and adjust Doppler data to more reasonable rainfall data. The rainfall pattern collected by the Doppler remains the same but the amount of rainfall, as seen by the instrument, is adjusted upward or downward to match the data obtained by the gages. The Corps of

Engineers will be looking at this process in further detail in the near future. As is often

the case, this is little data to determine any significant pattern between rainfall event and

peak flow and time to peak.

MANUAL CALIBRATIONS

Doppler precipitation data and USGS (Corps of Engineers owned) gage discharge

data from January 1997 through December 1999 were evaluated for this project. Twenty-

two storm events of this time frame were used for model calibration. Of this time period,

only four storm events produced peak discharge greater than 25,000 cfs. The twenty-two

models were calibrated to the observed hydrographs for their respective time periods by

changing several parameters of the HEC-1 model. The final computed hydrographs

closely matched the hydrographs obtained by the USGS gage. Peak discharge is a

parameter picked from historical gage data. The other four parameters used in the HEC-1

model, time-to-peak, % of base, initial losses, and constant losses were obtained from

model calibrations. Table 3.3 shows the results of the model calibrations for each storm

event.

MONTE CARLO SIMULATION

The Monte Carlo simulation approach is a method of selecting and inputting

randomly generated values from statistically defined distributions into a deterministic

model (McTernan, Daniels, and Garner, 1990). The approach taken for this research was

to use the maximum and minimum values obtained from historical data, and then use a

File Name	Tp	Cp	strtl	cnstl	ObPeak	CalPeak
F0697	18.18	0.38	0	0	2241	2289
F1297	22.94	0.74	0.41	0.04	2017	2073
*F2097	**11.39**	**0.96**	**0.74**	**0.01**	**47652**	**44696**
F2697	15.78	0.5	0.03	0.04	3990	4237
A0497	9.91	0.72	1.55	0.08	14987	14694
A2597	24.5	0.71	0.86	0.05	2778	2931
N0497	11.45	0.88	1.01	0.1	14715	14708
N1297	24.5	0.7	0.3	0.03	2366	2365
D0297	15.44	0.91	0.45	0.11	4189	4191
D0797	19.07	0.5	0.52	0.05	3308	3354
*J0498	**9.53**	**0.89**	**0.02**	**0**	**31938**	**30476**
F1098	12.24	0.54	0.76	0.08	9938	9834
*O0598	**11.62**	**0.87**	**0.57**	**0.03**	**41801**	**41273**
D0498	17.8	0.79	0.5	0.07	7410	7096
D1298	19.13	0.65	0.38	0.02	3858	3798
J2999	19.44	0.38	0.21	0.01	3530	3648
F0599	14.14	0.7	0.33	0.05	4353	4350
F1199	21.84	0.51	0.17	0	1379	1359
M1299	12.08	0.59	0.82	0.03	13167	13099
MY0499	10.94	0.52	0.75	0.01	8868	8768
MY1099	13.95	0.84	1.21	0.09	7148	6657
*MY1199	**11.05**	**0.94**	**0.17**	**0.01**	**26573**	**25880**

Table 3.3 Model calibrations of twenty-two storm events from January 1997 Through December 1999. The six parameters listed are time-to-peak (Tp), % of base (Cp), initial losses (strtl), constant losses (cnstl), observed gage peak (ObPeak), and calculated peak (CalPeak). The rows with an * indicate storm events which produced peak discharge greater than 25,000 cfs.

Monte Carlo approach with the HEC-1 model for computation of the peak flow and the time to peak (Mc Ternan, Daniels, and Garner, 1990). All the pertinent data for this simulation have been collected and put into a DSS database for each year, 1997, 1998, and 1999. The HEC-1 program was modified for this research to output the five parameters into a separate file.

A relatively simple program was written in Python 2.0 that created an input file for the HEC-1 software while at the same time randomizing the desired parameters between the limits established previously. This program then executes the HEC-1 software and retrieves the peak results for the desired parameters and appends them into a separate file. A total of 2000 simulations were done for each storm event. The number of simulations was chosen at random and is large enough to include all possible combinations of the randomized parameters. Some software manufacturers or programmers have calculated an error value that is minimized as more iterations are completed. This is normally done so that the processes can be stopped at a point where the error value does not change. However, with the newer powerful computer, it takes less than thirty seconds to do 2000 iterations of the model so processing efficiency is less of a problem. The code for Python is shown in Appendix C of this report.

RESULTS

This research concentrated on the four storm events that produced peak flows greater than 25,000 cfs. These flows were selected since the purpose of flood forecasting is to prevent flooding when high flows occur. The smaller flows typically do not cause any concern for flooding hazard. At first all four parameters, the time to peak, the

percent of base, the initial loss, and the constant loss, of the HEC-1 were randomized to obtain all possible peak flows from all possible combinations that can also represent various conditions for the basin. Then one parameter, each in turn, was fixed while the other three parameters were randomized. The results of the Monte Carlo Simulations were plotted on a Log Pearson Type III plot. The Log Pearson Type III distribution fit the data obtained by the Monte Carlo method for all the scenarios studied in this section.

The results of randomizing the four parameters for the HEC-1 model showed a wide range of flows for the four storms studied. A forecaster is concerned with the upper end of the graph that shows the higher peak flows for the storm event. The results of randomizing the four parameters showed that the peak flow during these storm events, fall between the 5 and 0.5 percent exceedence probability interval as shown in figures 3.2 through 3.5.

Figure 3.2 is a plot of the resulting output data from the Monte Carlo Simulation for the storm event that occurred on February 20,1997. A HEC-1 model similar to that of table 3.2 was done for the 1997 storm event and the Python program was used to do the Monte Carlo Simulation runs to produce the various outcomes. As shown in the figure, a large range of peak flows were produced from the Monte Carlo model. The storm event may produce peak floods between 7,000 cfs to 50,000 cfs for the various basin and simulation conditions. The observed value that this storm produced was 47,652 cfs which is at the upper end of the plot. Not knowing the conditions of the basin at present time in this study, the engineer may use the upper end value as the worst case senario for forecasting the peak flow which will be produced. Typically, the hydrologic engineer or the forecaster has some knowledge of the basin conditions while trying to predict the
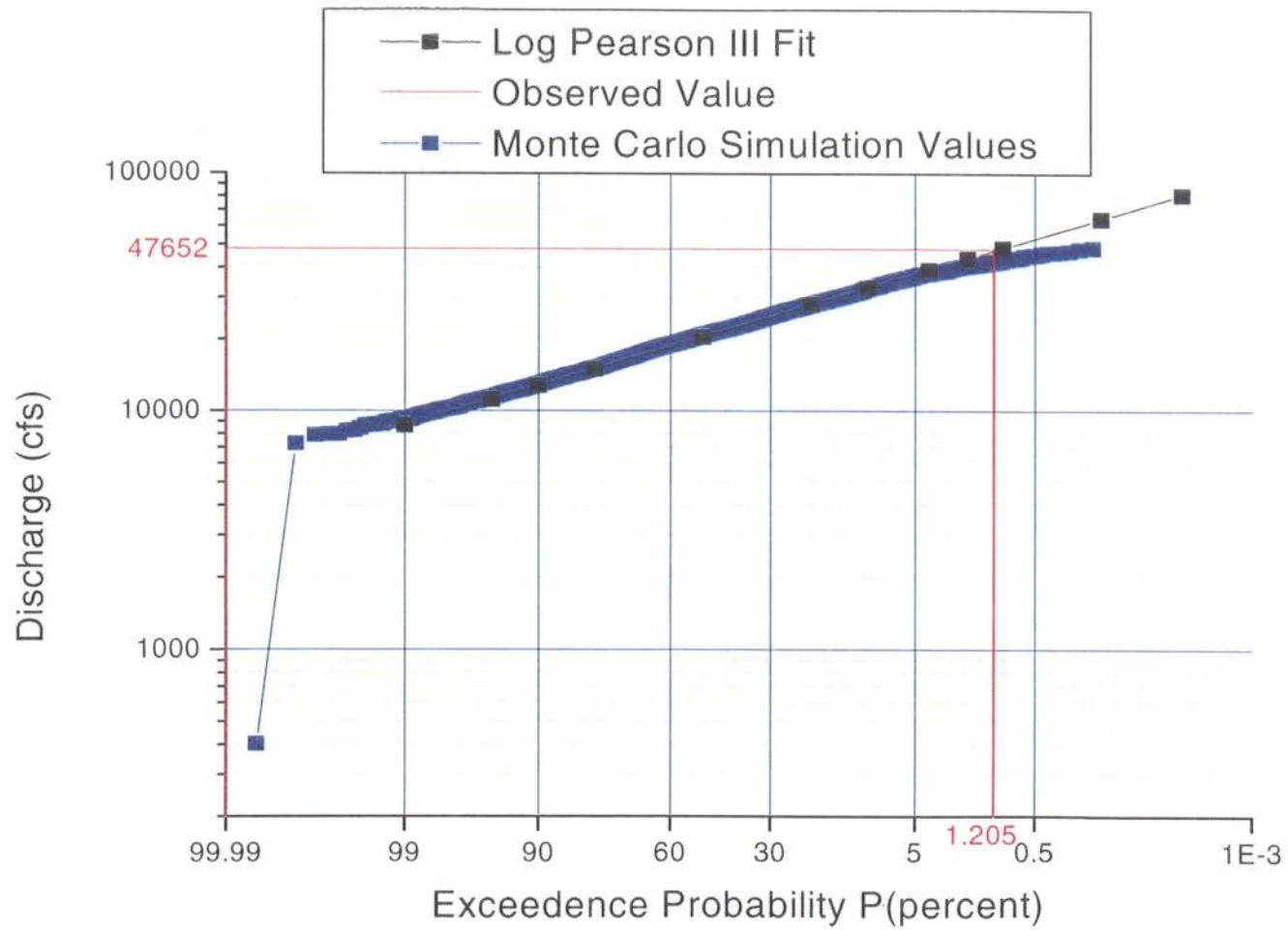
Figure 3.2. February 20, 1997 Storm Event Results. Observed Value of 47,652 cfs, P = 1.2.
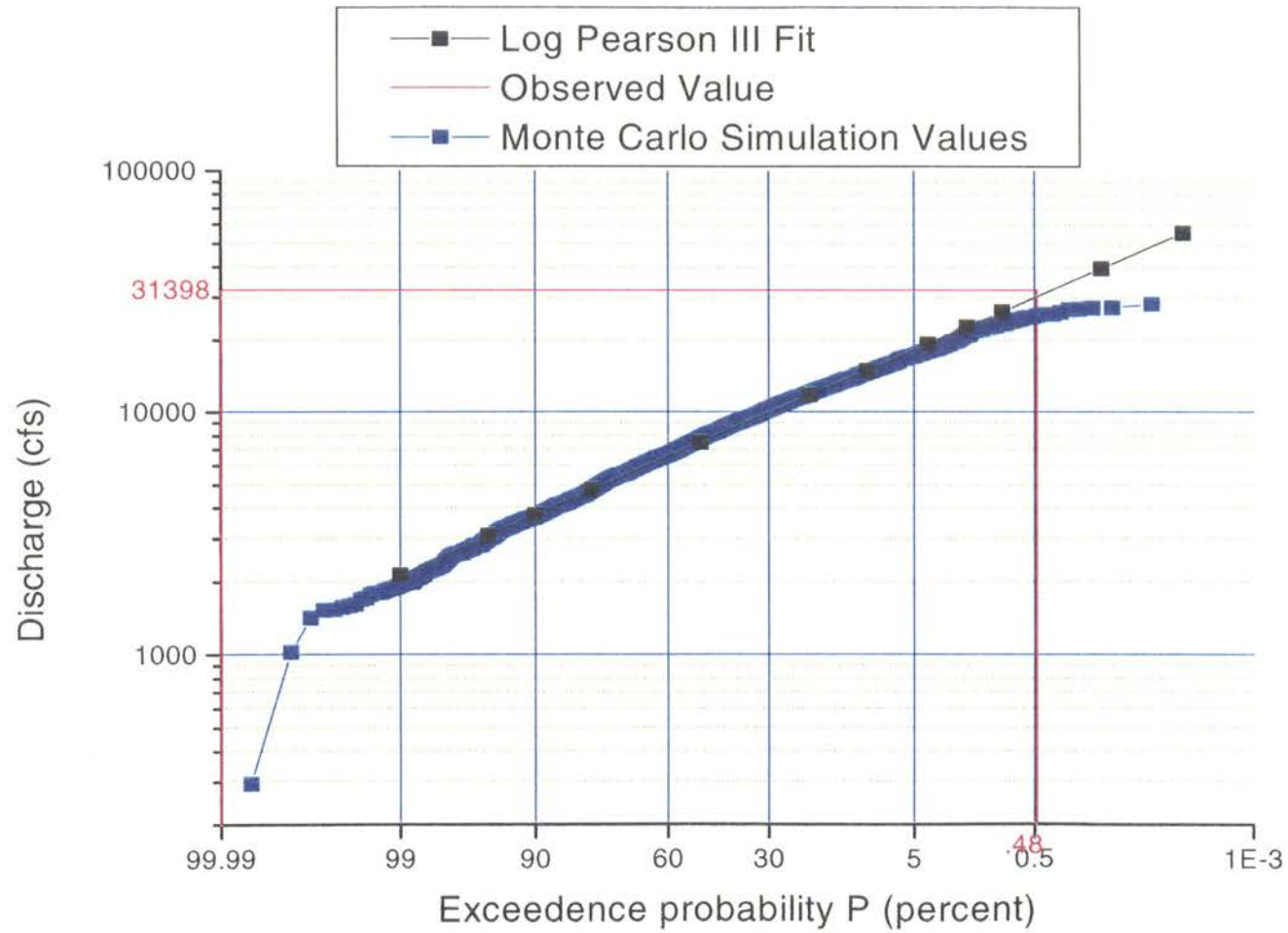
Figure 3.3. January 4, 1998 Storm Event Results. Observed value of 31,398 cfs, P = .48.

Figure 3.4. October 5, 1998 Storm Event Results. Observed value of 41,801 cfs, P = .997.
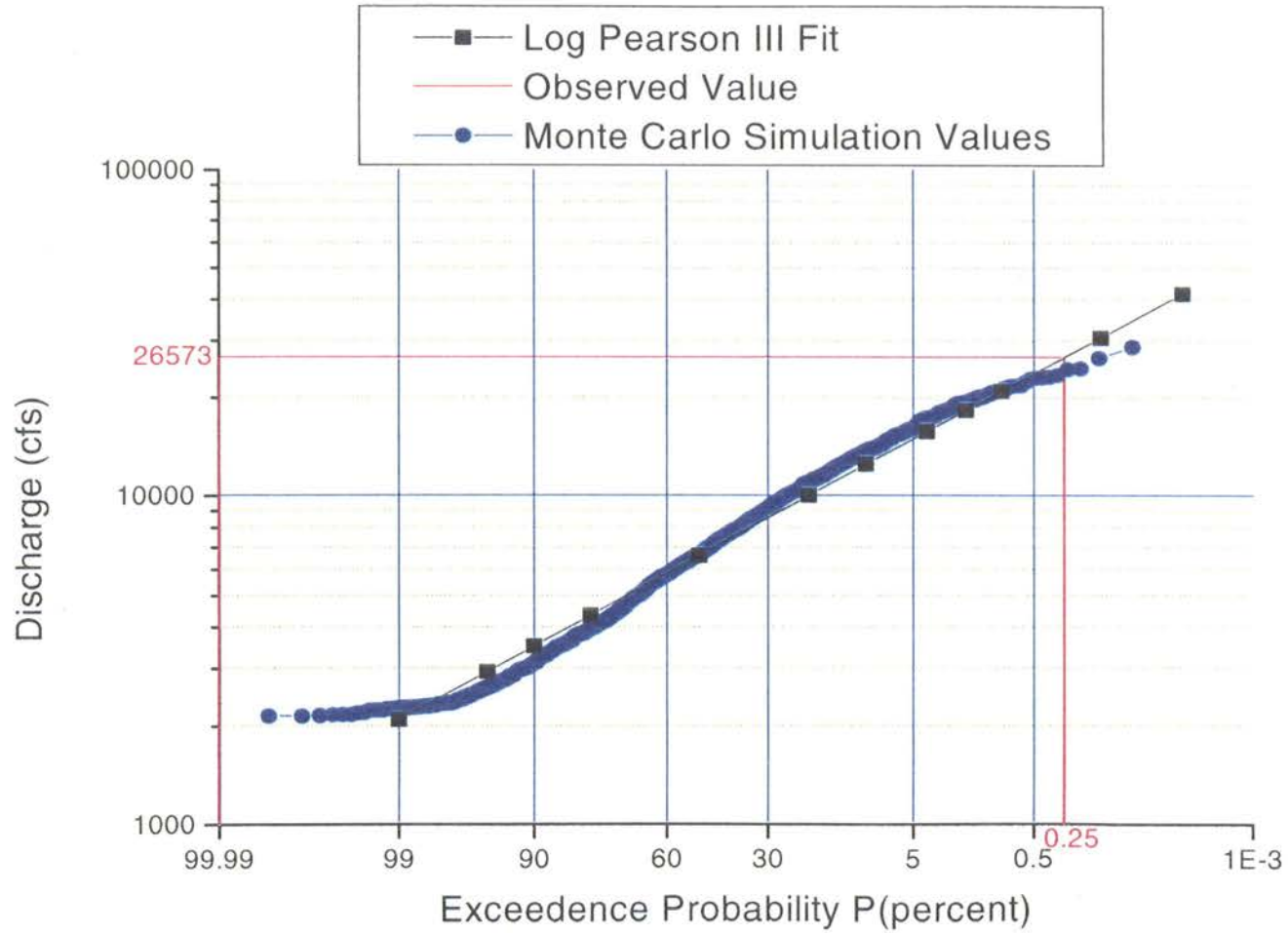
Figure 3.5. May 11, 1999 Storm Event Results. Observed value of 26,573 cfs, P = 0.25.

effective runoff during the storm. Knowledge of previous rainfall events can aid the engineer in estimating the approximate exceedence probability of the runoff. By communicating with the reservoir control personnel, it can be determined if the system (lake(s), and river(s)) can accommodate this large flow at the appropriate time without causing flooding damages. If it cannot, preparations can be made in advance to handle this large flow event.

Figure 3.3 is similar to figure 3.2 but represents the storm event that occurred on January 4, 1998. This storm event also occurred in the winter when there is minimal vegetative land cover. Knowing that there is minimal vegetative land cover during this time period decreases the evapo-transpiration losses, thus increasing the total effective runoff from the storm. This knowledge of little vegetative cover gives the engineer a better understanding for determining a more accurate range of the exceedence probability. The Monte Carlo Simulation model produced peak discharge values in the range of 1,000 cfs to 40,000 cfs. The observed peak flow that actually occurred for this storm event was 31,398 cfs. The observed value for this event plots in the upper portion of the Log Pearson plot. A lower peak flow value of 100 cfs was also computed by the Monte Carlo simulation but the trend as seen in the plot is greater than 1000 cfs.

Similarly, Figure 3.4 is a plot of the resulting output data from the Monte Carlo Simulation model for the storm event of October 5, 1998. This storm event produced a peak discharge of 41,801 cfs. The storm event occurred in the fall when the vegetative cover is still in bloom causing the roughness coefficient of the channel to be higher and the evapo-transpiration losses through the vegetation are increased. The engineer can then assume that the losses will be higher and that the effective runoff will be diffused

even greater causing the peak flow value to be lower than expected. The Log Pearson III distribution is a perfect fit to the data as seen in the figure. The plot also shows that the storm can produce a peak flow between 5,000 cfs up to 43,000 cfs depending on the conditions of the basin.

Figure 3.5 is a simulation for the storm event of May 11, 1999. This storm event occurred near the end of the spring season. The vegetative cover can be assumed to be in full bloom at this time. Typically, the late springs of Oklahoma are somewhat humid with few storm events occurring. The heavy vegetation, along with the semi-dry soil conditions, can cause the initial and constant abstraction values used in the HEC-1 model to be in the upper end of the distribution. The heavy vegetation along the channel and the terrain will also cause the runoff to be diffused or lagged for a longer period of time. Therefore the peak flow will occur much later after the storm even has subsided and the peak flow will be decreased due to the losses. The peak observed discharge value of this storm event (26,573 cfs) also plotted at the upper end of the Log Pearson Type III distribution developed from the Monte Carlo simulation.

As shown in the various figures, (figures 3.2 thru 3.5) the Log Pearson Type III distribution fits the data well. The output of this model can be modified to also include the output of the four HEC-1 parameter values for all the peak flows. This can aid an experienced forecaster in narrowing the range of peak flows given a smaller range of values. If the forecaster is confident to a degree of one or more of the values, time to peak, percent of base, initial losses or constant losses, a better prediction can be made. An experienced forecaster can eventually determine a good relationship between the conditions of the basin at the time a storm event occurs and the exceedence probability

that the storm will produce. This method gives the forecaster a realistic range of peak flows that a storm can produce based on the basin's conditions. Typically a hydrologic engineer conducting a forecast knows the approximate conditions of the basin. If previous rainfall events have occurred in the basin within the last day, it can be assumed with some degree of confidence that the soil conditions are still moist thus reducing the loss rates and increasing the peak flow that the storm will produce. The engineer can then expect that the exceedence probability be in the lower range of the plot.

Even with few Stage III data available for this research, the results showed that a hydrologic engineer can use the Monte Carlo method as a preliminary analysis of the storm's effective runoff. The forecaster or reservoir control engineer should initially view the five percent exceedence as the maximum peak flow that a storm can produce depending on the basin's condition. If it is known that the ground is on the wet side, to nearly saturated, the engineer should anticipate peak flows closer to the 0.5 percent section. It is cautioned against initially presenting the peak flow for any storm event near the 0.5 percent exceedence without greater knowledge of the basin's conditions due to the costs involved in declaring a flood potential in any community. This can also impact storage potential for future usage by releasing more stored water then necessary. The engineer doing the forecast should take into account past climatic conditions before selecting a predicted peak flow value. These simulations show the peak flows that can possibly occur and alternative flood prevention scenarios can be viewed during the initial forecasting conditions.

Beard (1962) recommends against using the statistical skew coefficient for a frequency study on a single storm event. He suggested that a zero skew coefficient

should be used for hydrological studies unless there are historical data of 100 years or more that show a radical departure from usual values of skew. The fitted line used for this study with a skewness of zero matched the data obtained from the Monte Carlo Simulation. A fitted line of this type is recommended for future usage because it gives a better representation of the peak flows near the high end as compared to just using the calculated results.

The results of fixing one of the variables while randomizing the other three variables showed that each variable has a tremendous effect on the resulting peak flows. Figures 3.6 through 3.21 shows the peak flows obtained by running several Monte Carlo Simulations and fixing one variable for each of the aforementioned storm events. The data of the Monte Carlo Simulation for randomizing all four parameters of the HEC-1 model was also plotted in each of the graphs for comparison. Figures 3.6, 3.7, 3.8, and 3.9 are plots showing the results of fixing the time-to-peak for the February 20, 1997, January 4, 1998, October 5, 1998, and May 11, 1999 storm event respectively. The data from Figures 3.2, 3.3, 3.4 and 3.5 were also included in their respective plots as well. By fixing the time-to-peak values for each storm event to the minimum value, produced consistent higher peak values than the peak values produced by randomizing this parameter as well. This is expected since the HEC-1 model uses this information to attenuate the flow values. That is, the precipitation produced by the storm is routed downstream at a fast rate of travel with little dissipation of the flow. The opposite is also true. A greater time-to-peak value produces lower peak flow values. This means that the time-to-peak of the flow, which the storm produced, is slowed down considerably or diffused more widely, thus forcing the program to dissipate the flow and lower the peak

49

Figure 3.6. February 20, 1997 Storm Event. Results of the Monte Carlo Simulation with fixed Time to Peak values at 9.53 hours, 15.77 hours and 24.5 hours. Results show the effects of fixing the Time to Peak values.
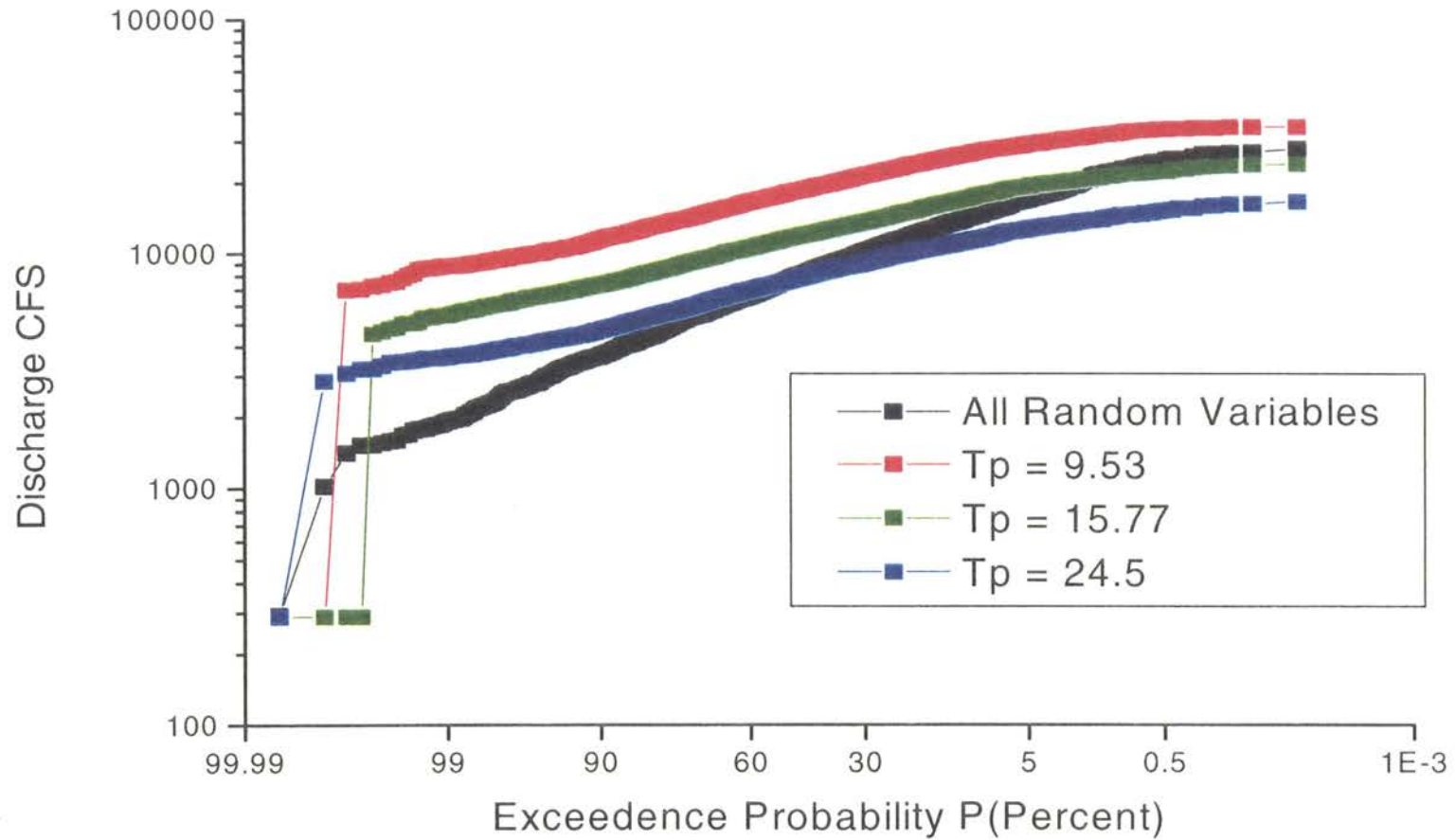
Figure 3.7. January 4, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Time to Peak values at 9.53 hours, 15.77 hours and 24.5 hours. Results show the effects of fixing the Time to Peak values.
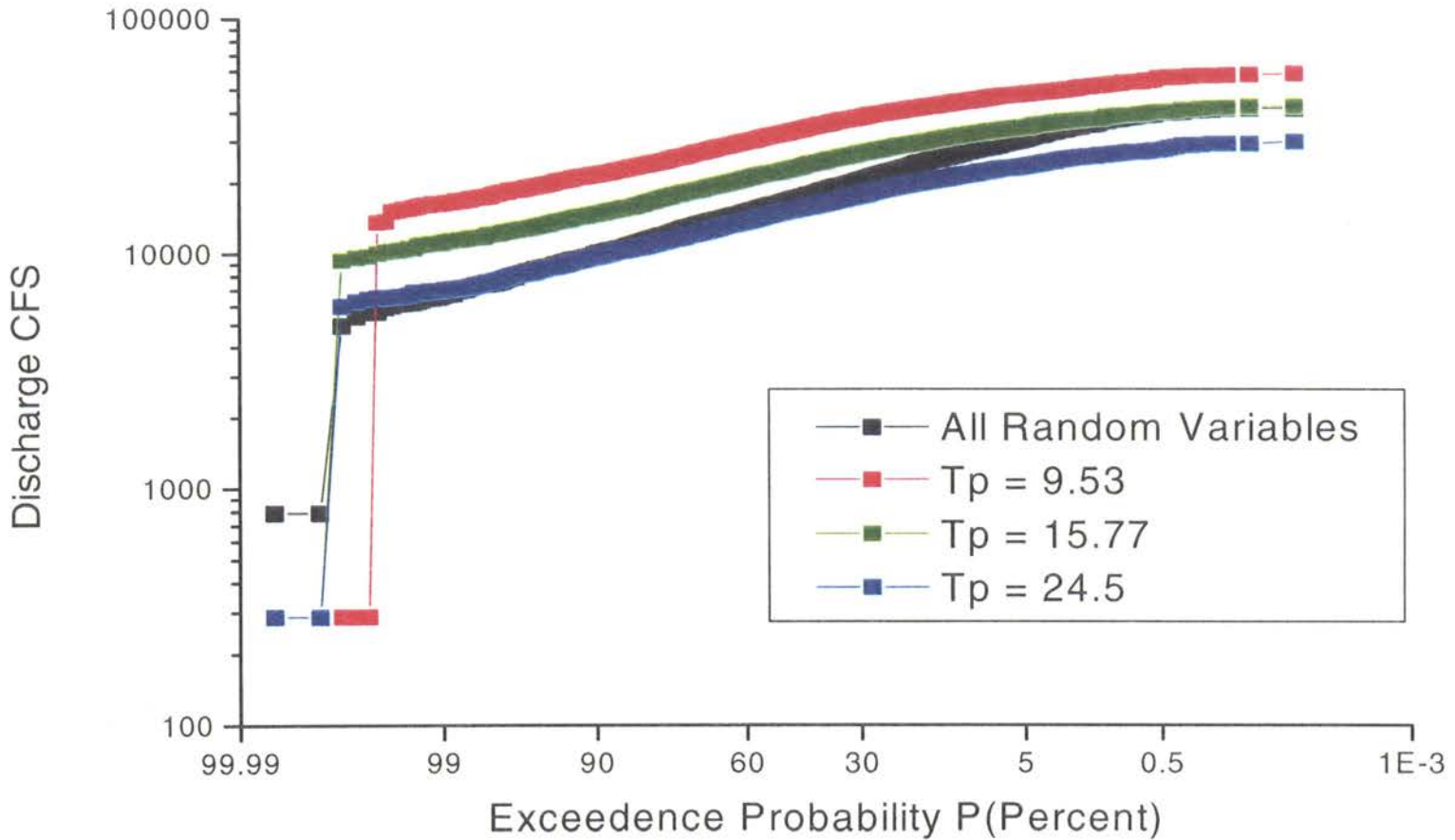
Figure 3.8. October 5, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Time to Peak values at 9.53 hours, 15.77 hours and 24.5 hours. Results show the effects of fixing the Time to Peak values.

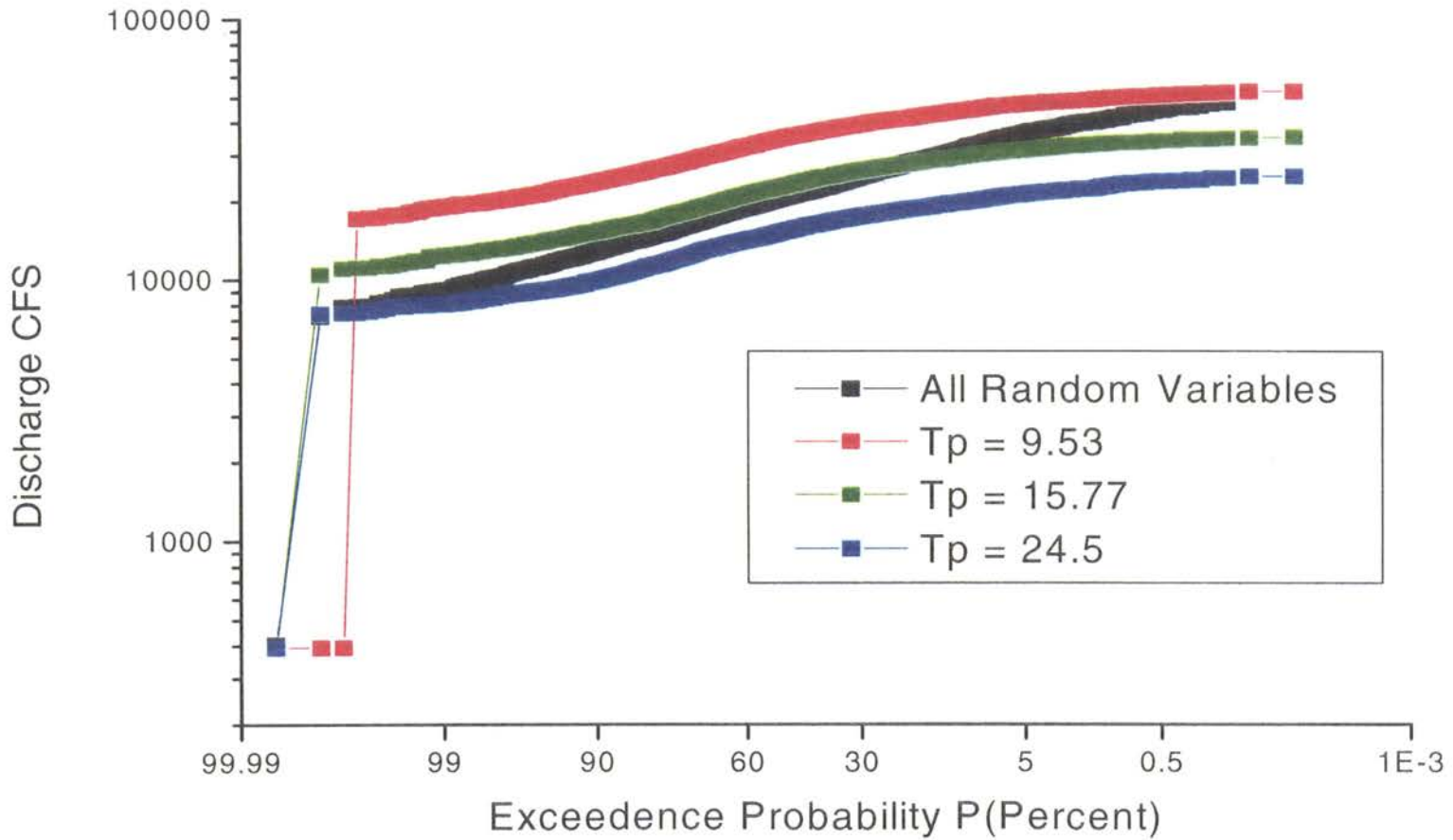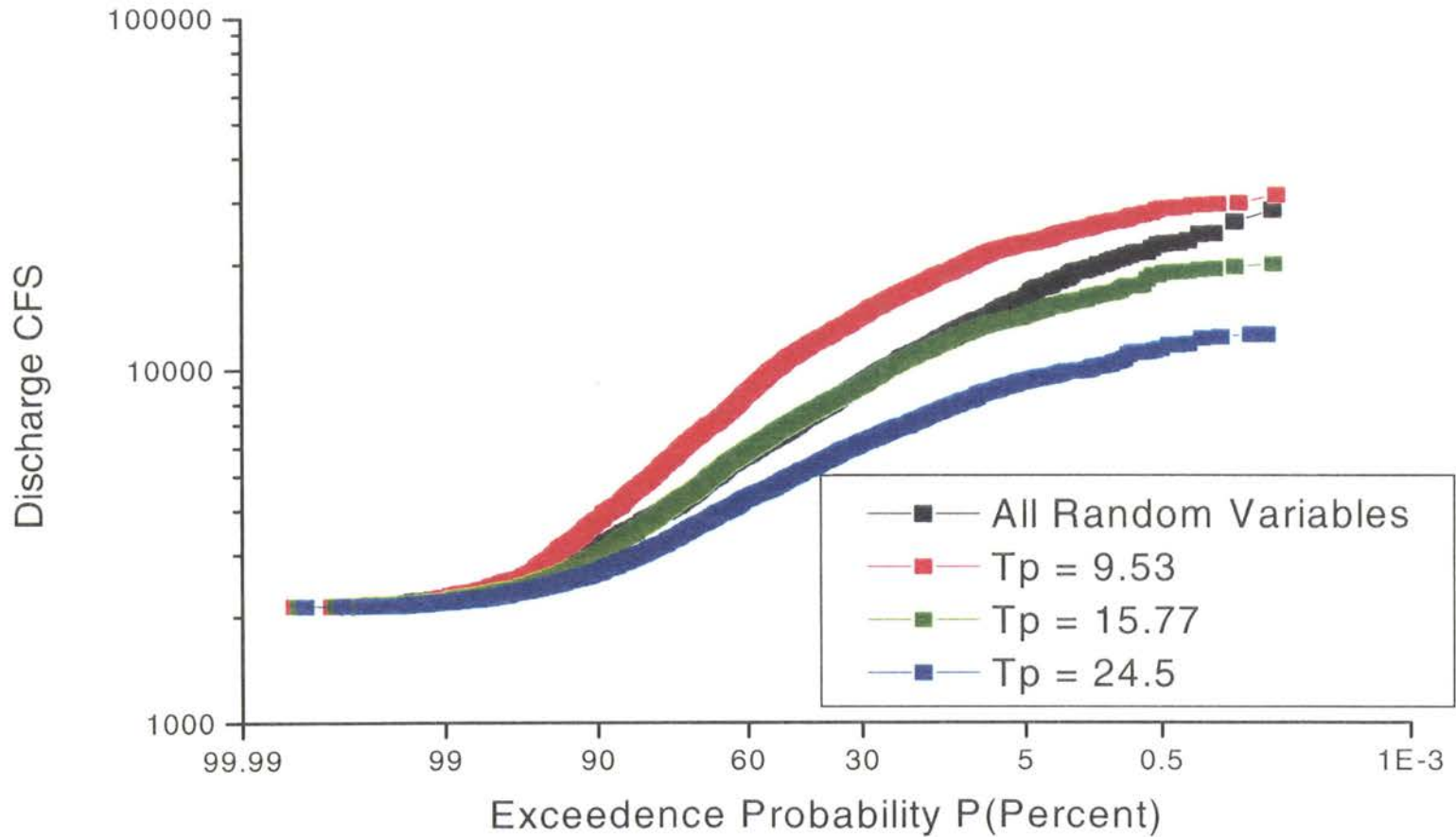**Figure 3.9. May 11, 1999 Storm Event. Results of the Monte Carlo Simulation with fixed Time to Peak values at 9.53 hours, 15.77 hours and 24.5 hours. Results show the effects of fixing the Time to Peak values.**
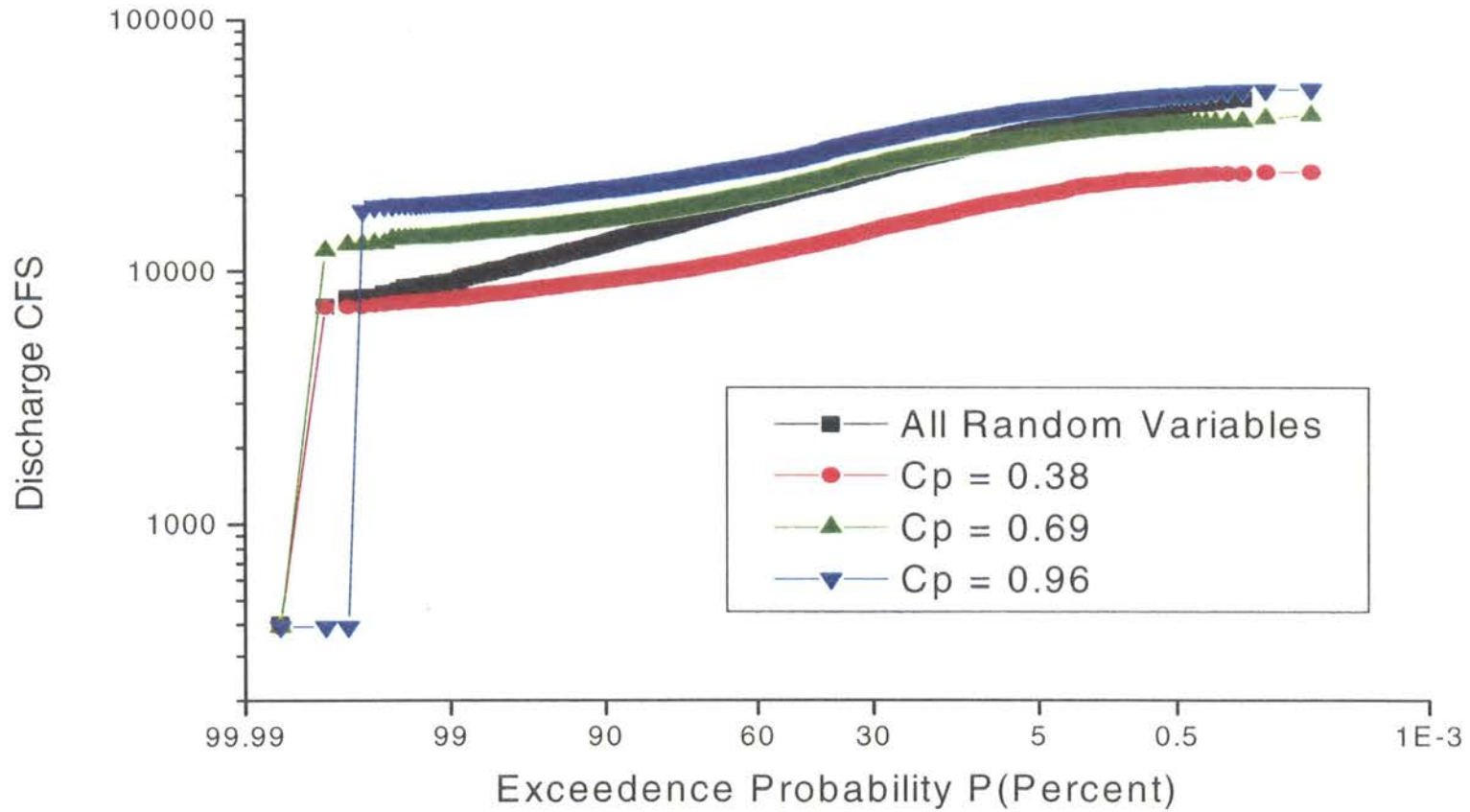
Figure 3.10. February 20, 1997 Storm Event. Results of the Monte Carlo Simulation with fixed Cp values at 0.38%, 0.69%, and 0.96%. Results show the effects of fixing the Cp values.
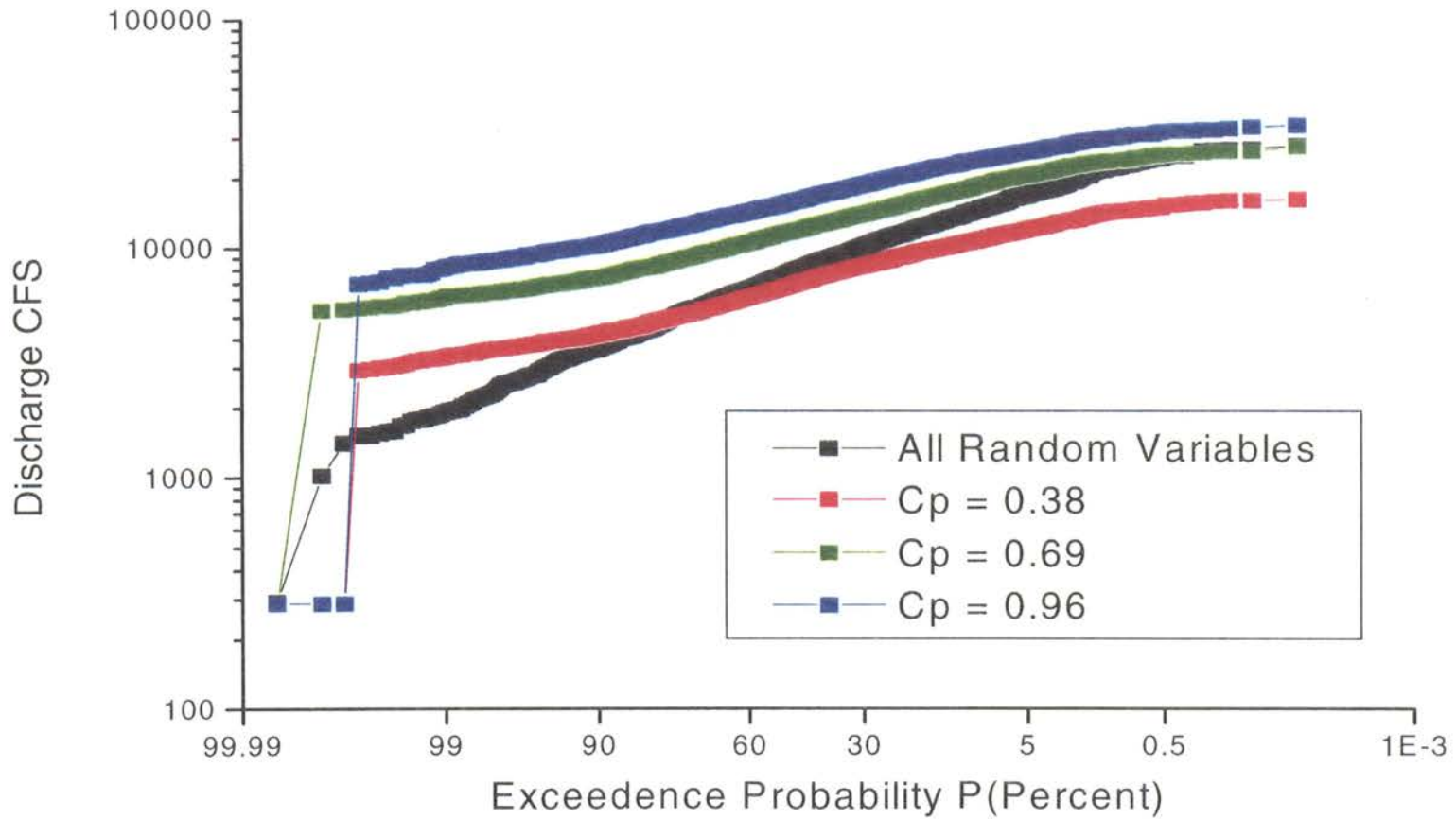
Figure 3.11. January 4, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Cp values at 0.38%, 0.69% and 0.96%. Results show the effects of fixing the Cp values.
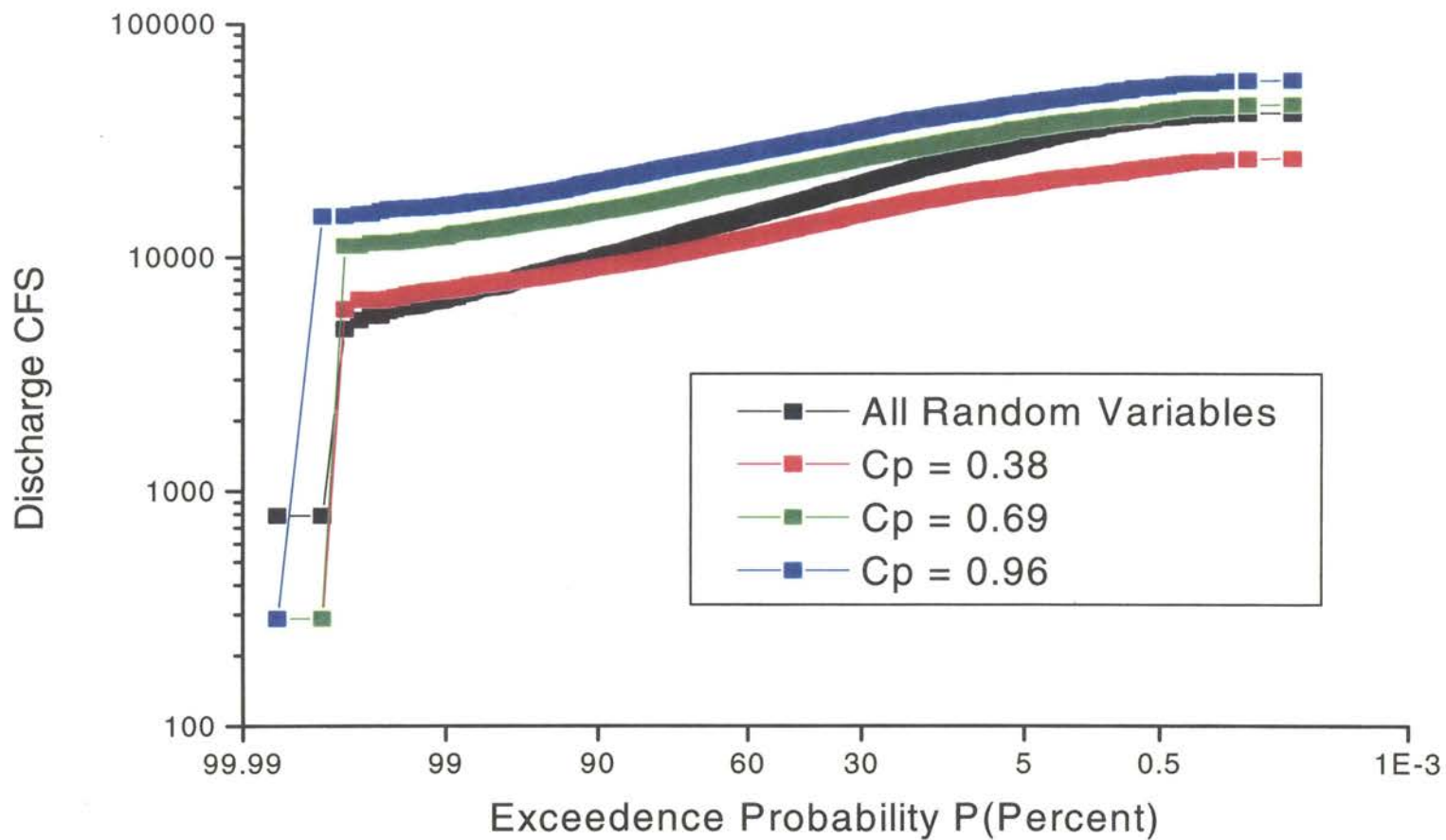
Figure 3.12. October 5, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Cp values at 0.38%, 0.69% and 0.96%. Results show the effects of fixing the Cp values.
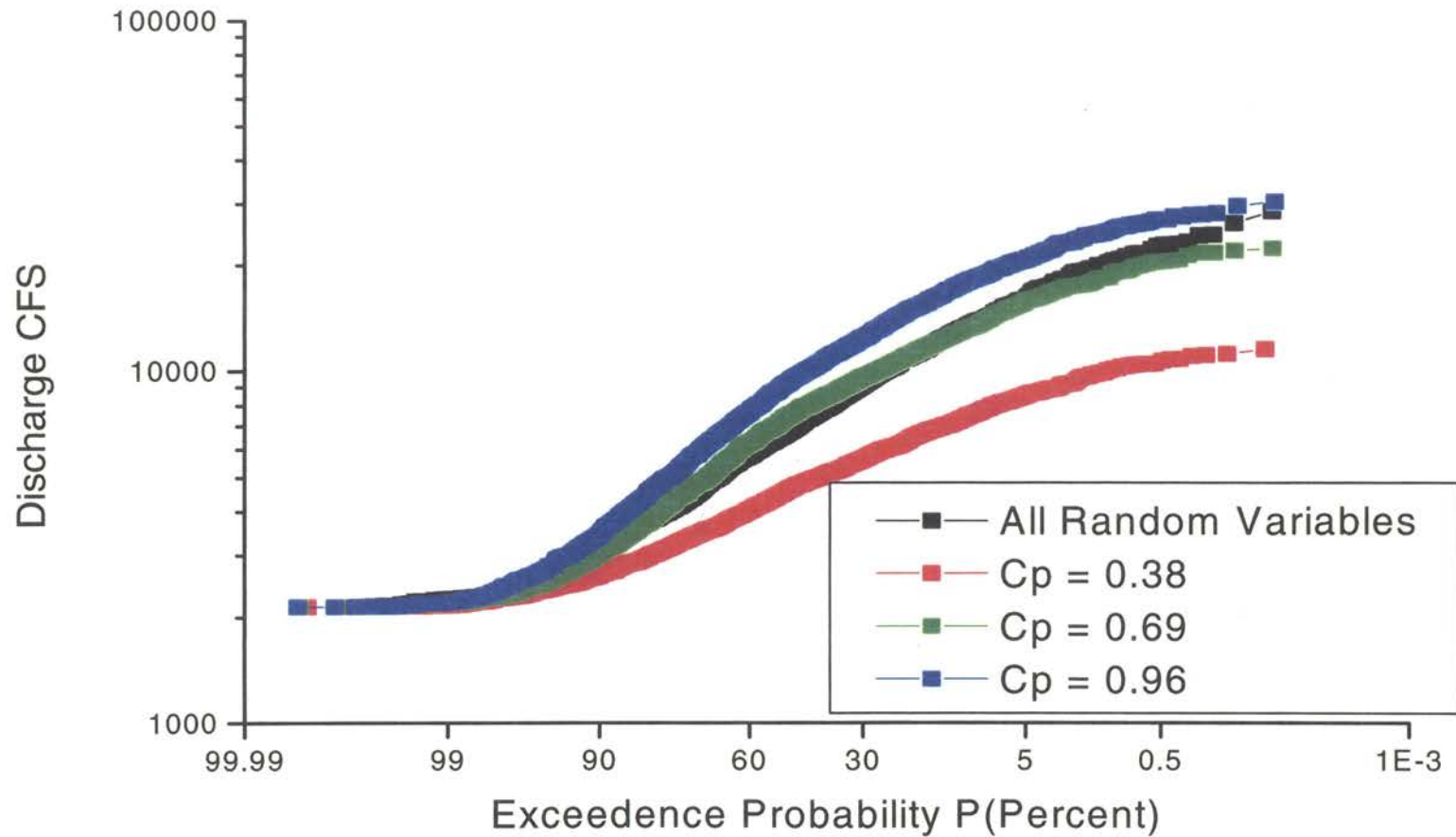
Figure 3.13. May 11, 1999 Storm Event. Results of the Monte Carlo Simulation with fixed Cp values at 0.38%, 0.69%, and 0.96%. Results show the effects of fixing the Cp values.

Figure 3.14. February 20, 1997 Storm Event. Results of the Monte Carlo Simulation with fixed Strtl values at 0 inches/hour, 0.53 inches/hour and 1.55 inches/hour. Results show the effects of fixing the Strtl values.
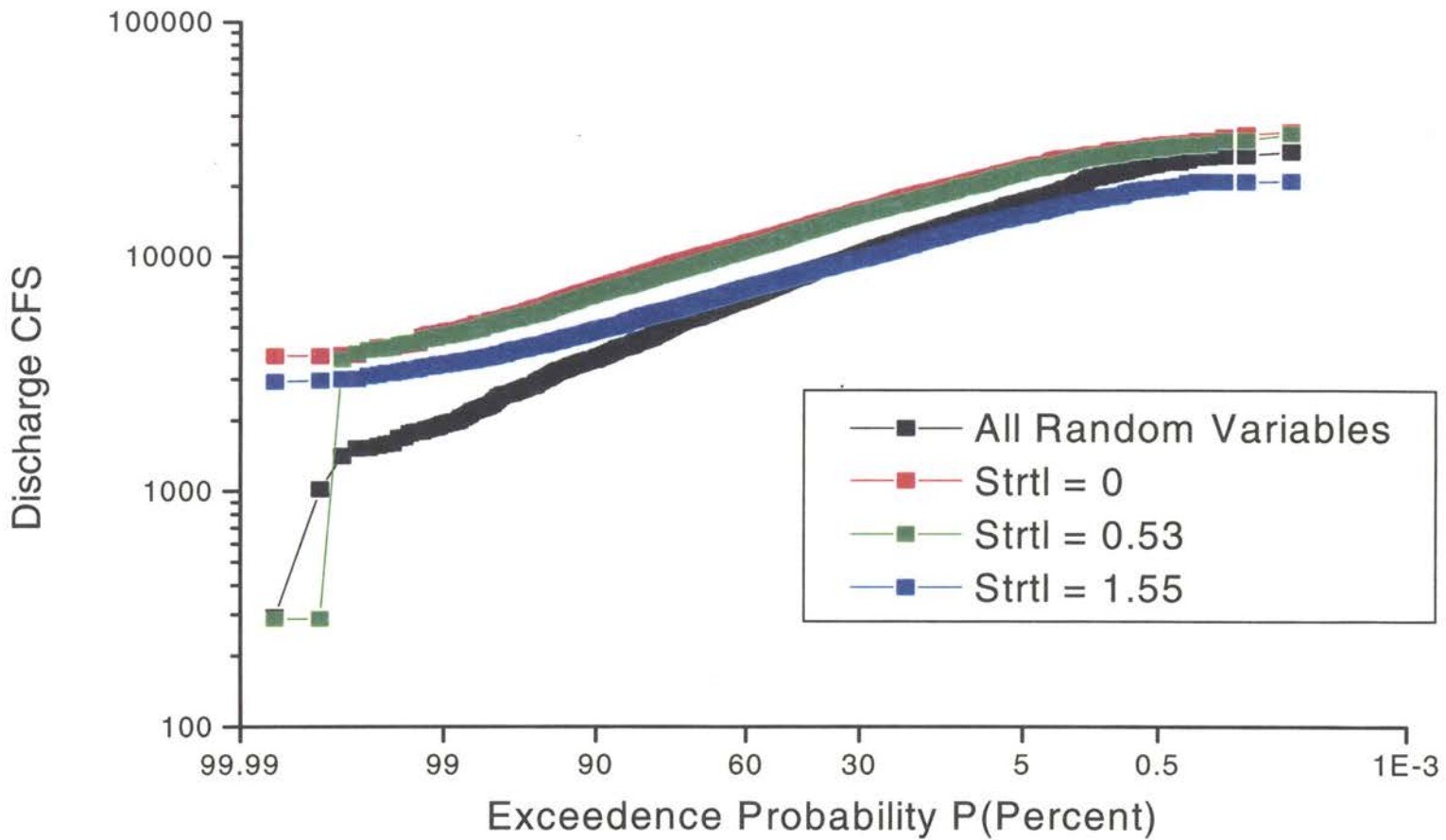
Figure 3.15. January 4, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Strtl values at 0 inches/hour, 0.53 inches/hour and 1.55 inches/hour. Results show the effects of fixing the Strtl values.
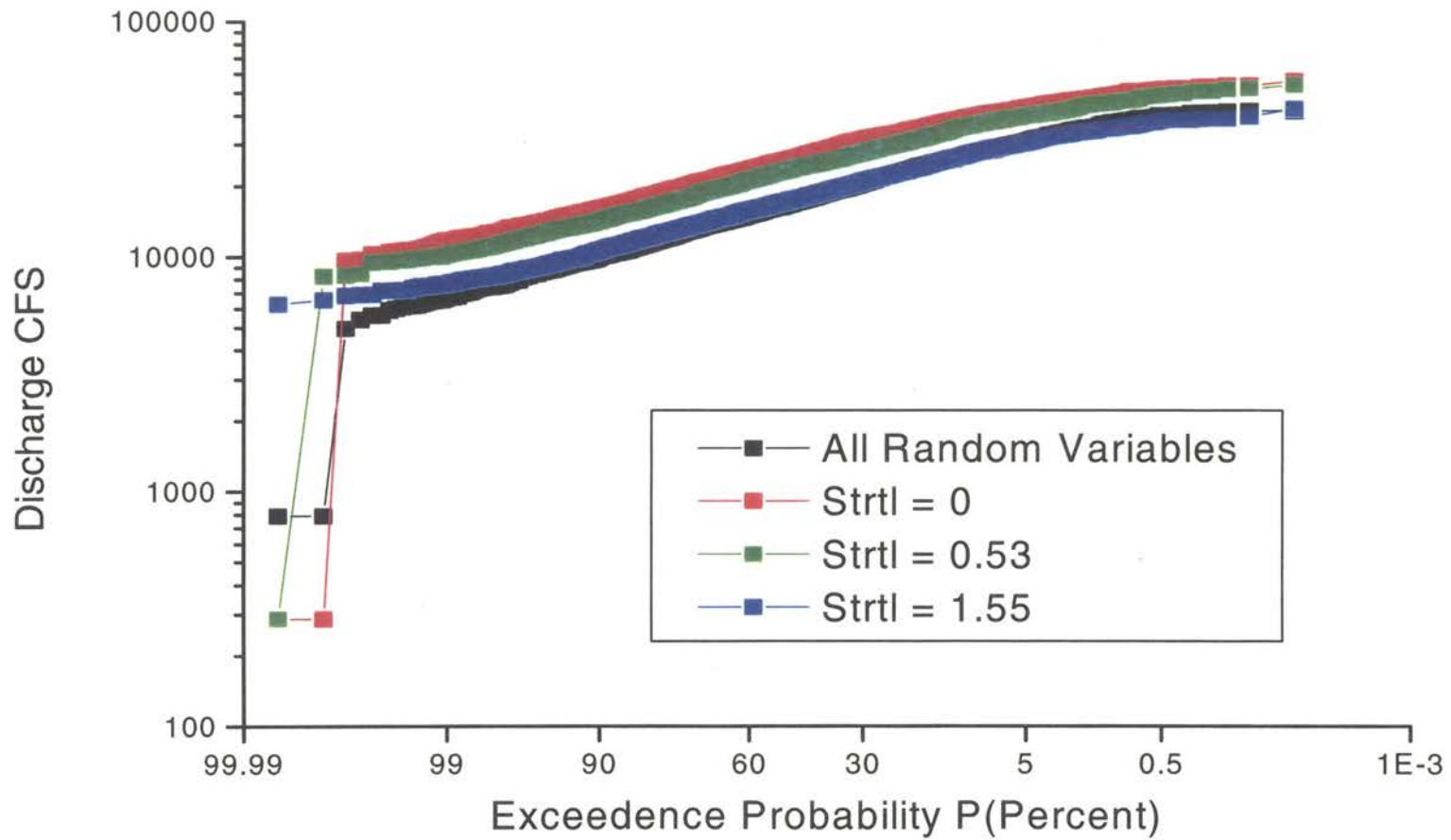
Figure 3.16. October 5, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Strtl values at 0 inches/hour, 0.53 inches/hour and 1.55 inches/hour. Results show the effects of fixing the Strtl values.
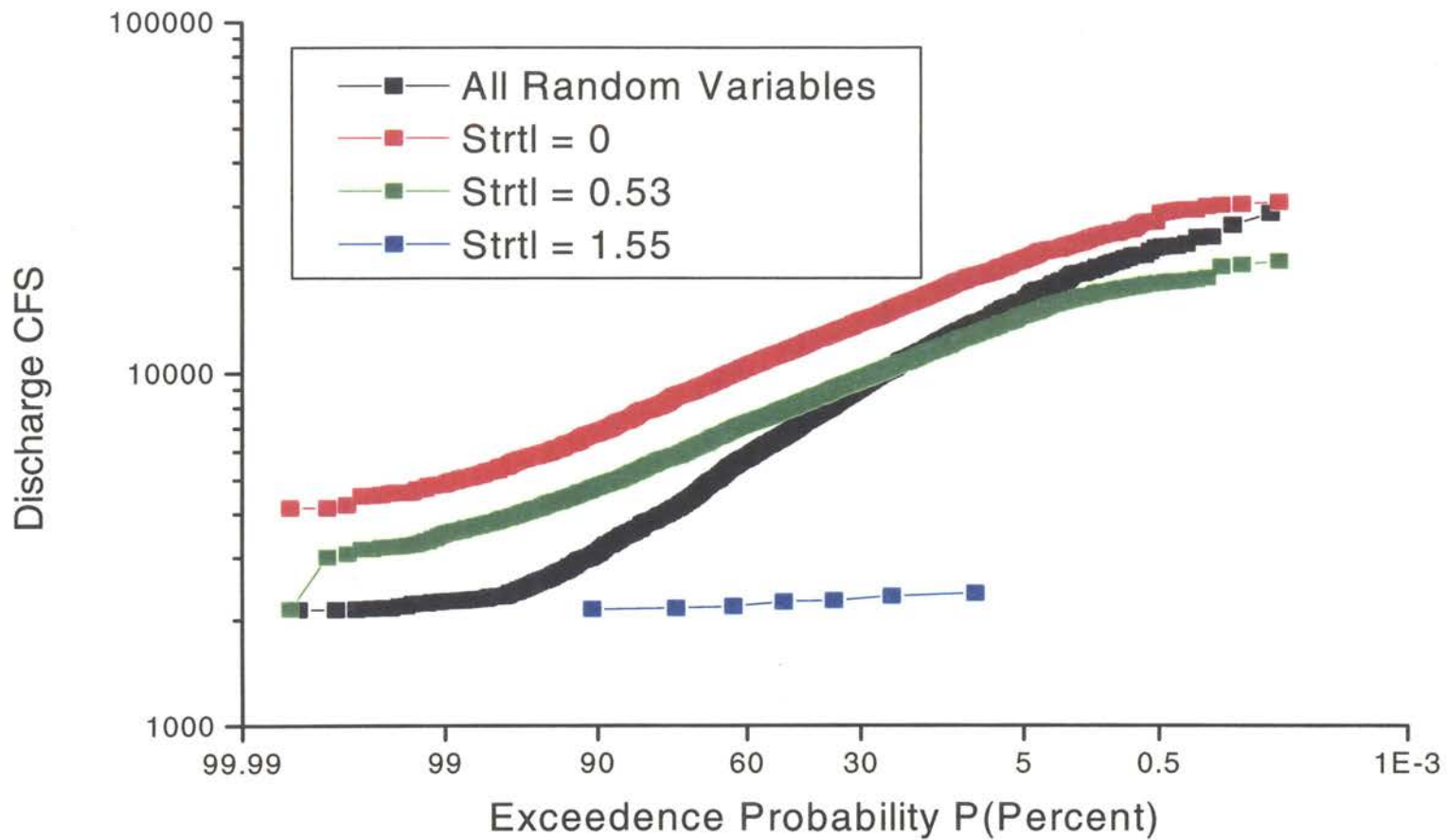
Figure 3.17. May 11, 1999 Storm Event. Results of the Monte Carlo Simulation with fixed Strtl values at 0 inches/hour, 0.53 inches/hour and 1.55 inches/hour. Results show the effects of fixing the Strtl values.
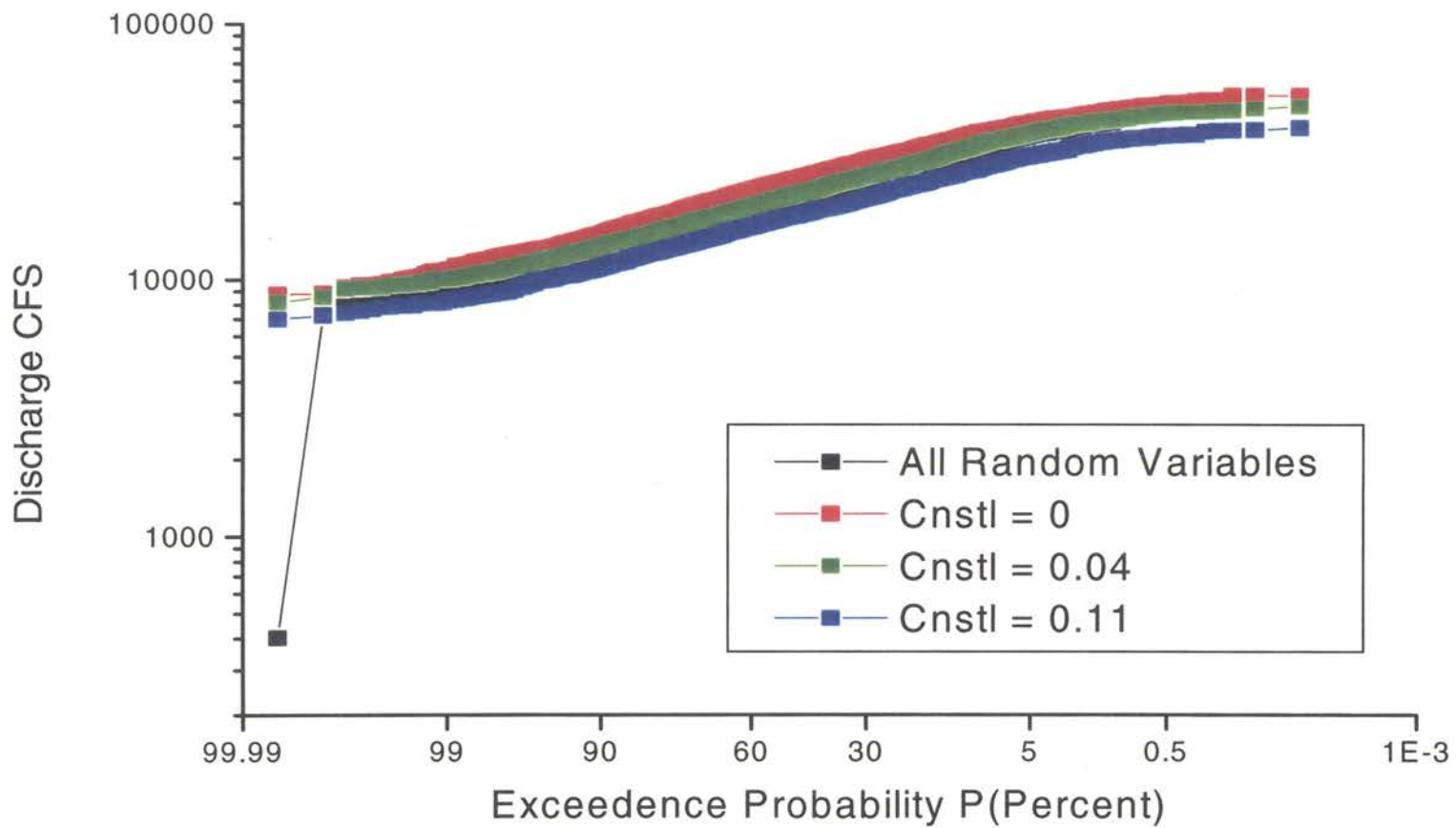
Figure 3.18. February 20, 1997 Storm Event. Results of the Monte Carlo Simulation with fixed Cnstl values at 0 inches/hour, 0.04 inches/hour and 0.11 inches/hour. Results show the effects of fixing the Strtl values.
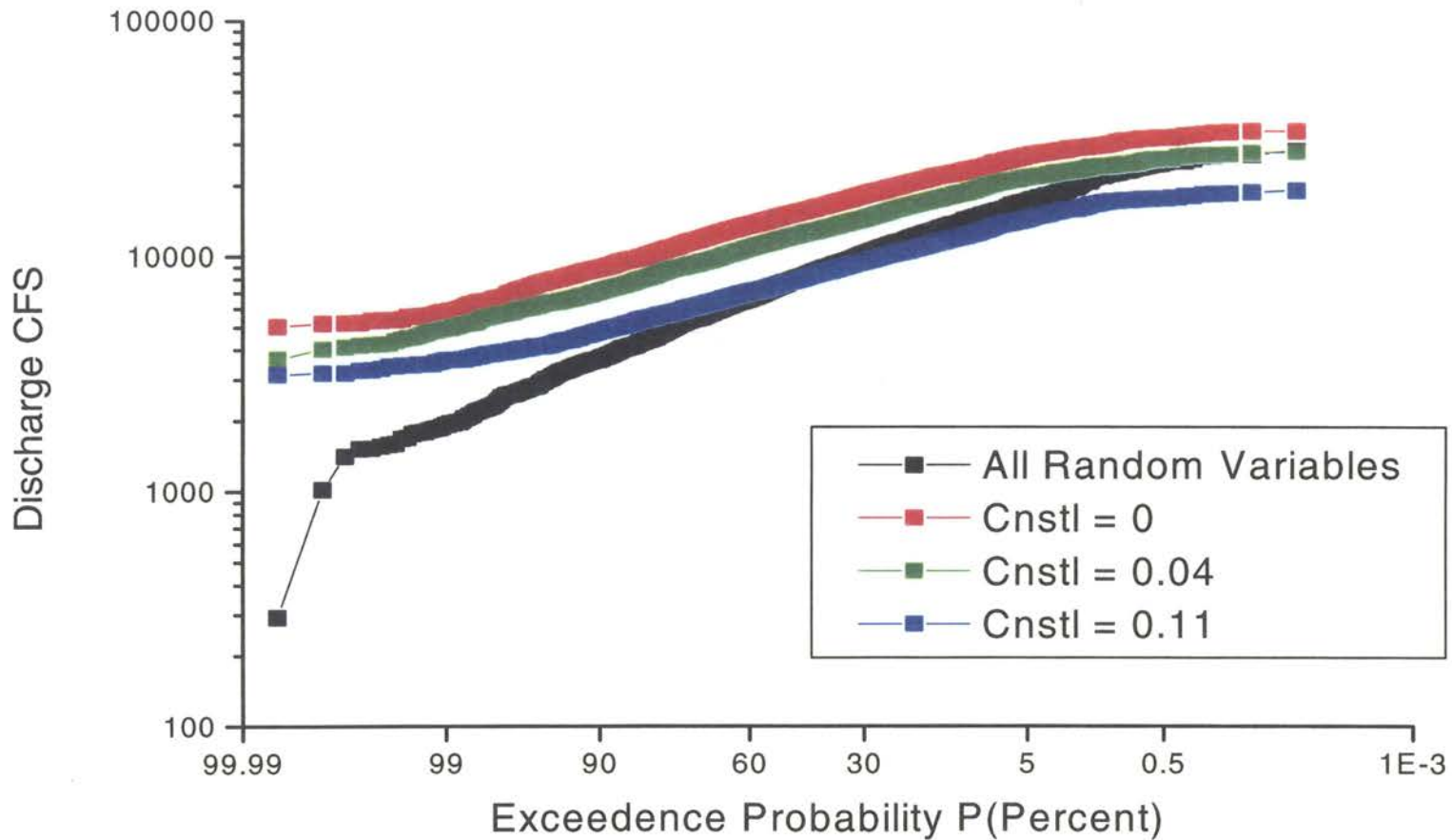
Figure 3.19. January 4, 1998 Storm Event. Results of the Monte Carlo Simulations with fixed Cnstl values at 0 inches/hour, 0.04 inches/hour and 0.11 inches/hour. Results show the effects of fixing the Strtl values.
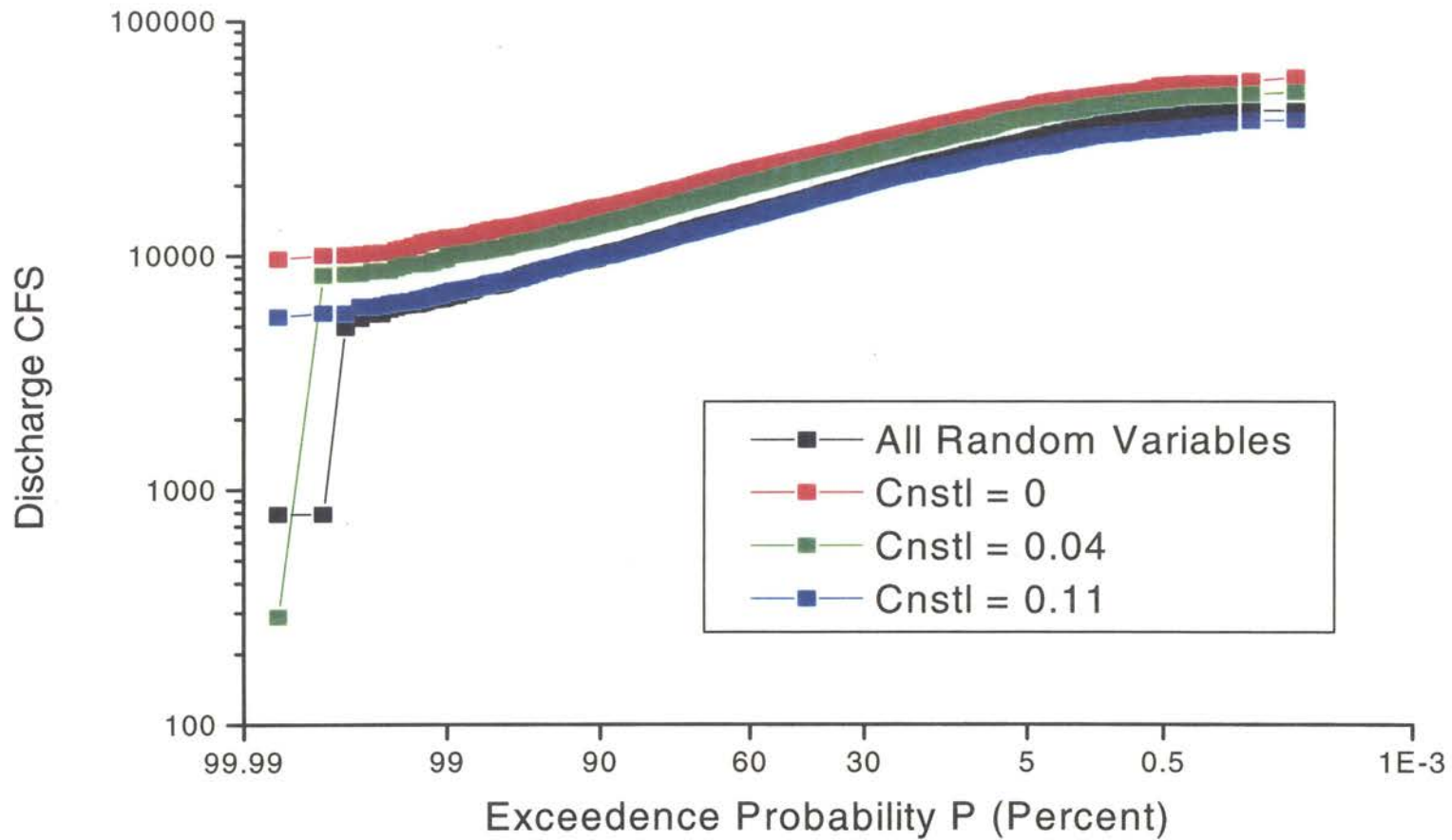
Figure 3.20. October 5, 1998 Storm Event. Results of the Monte Carlo Simulation with fixed Cnstl values at 0 inches/hour, 0.04 inches/hour and 0.11 inches/hour. Results show the effects of fixing the Strtl values.
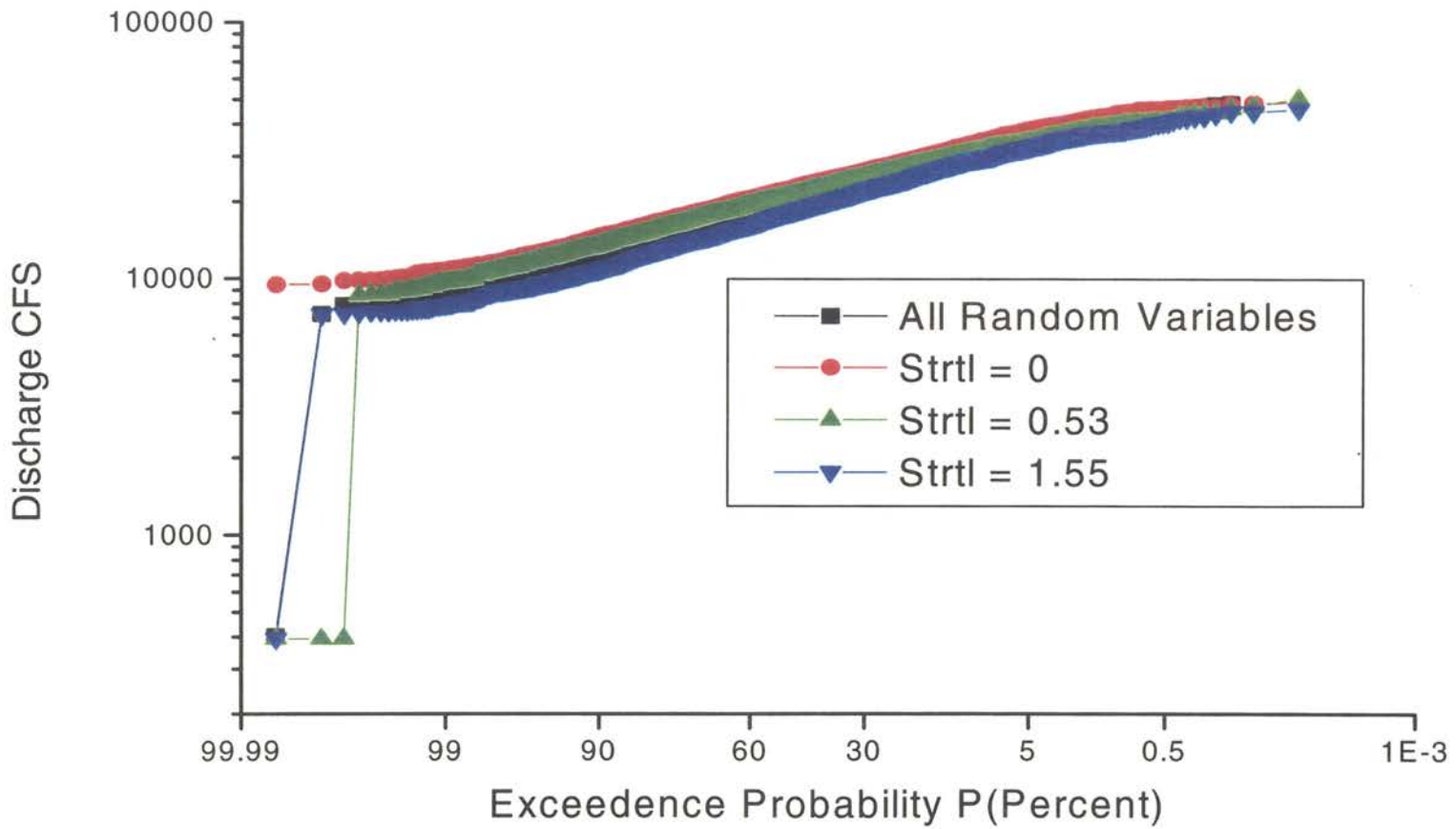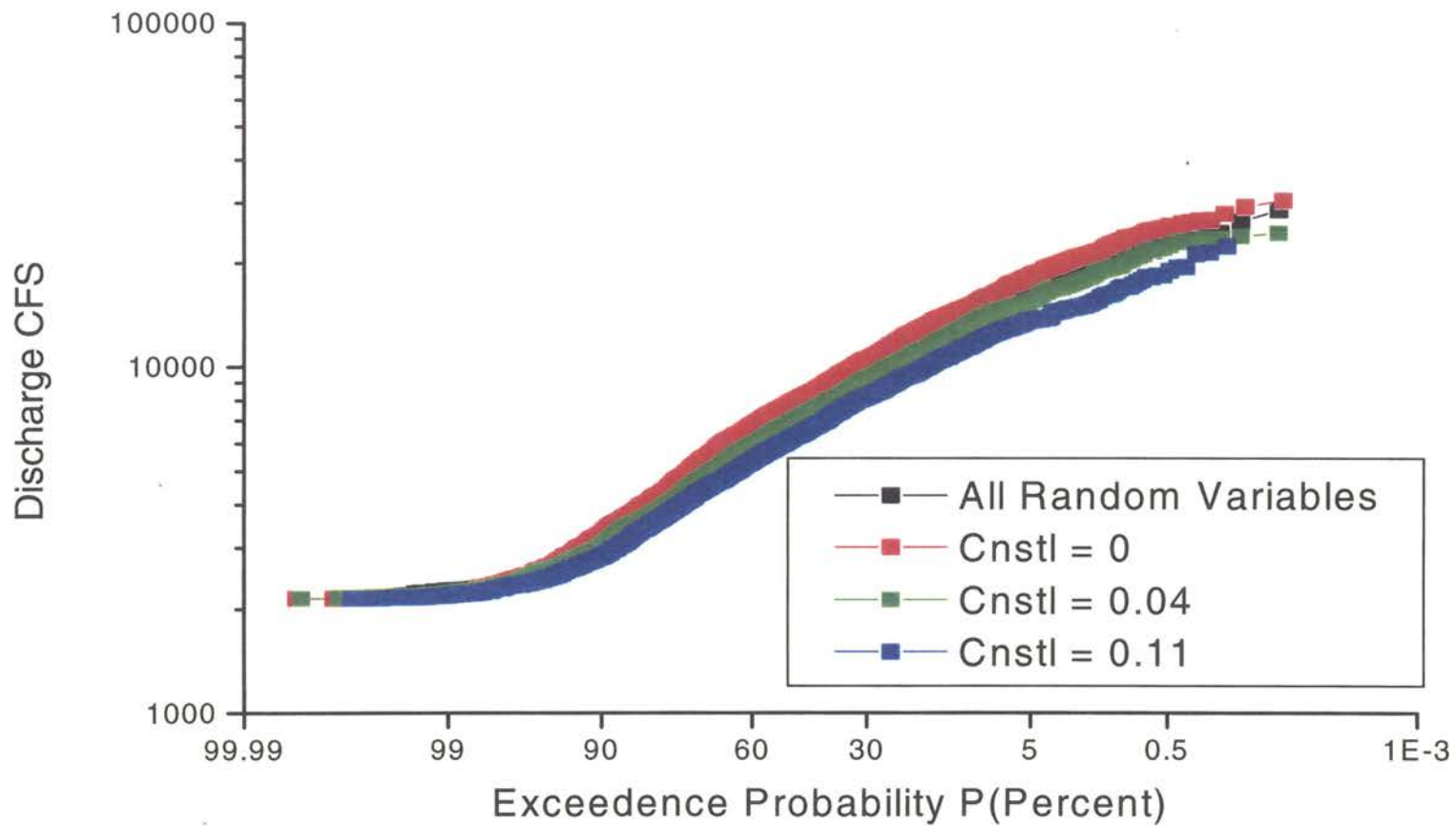
Figure 3.21. May 11, 1999 Storm Event. Results of the Monte Carlo Simulation with fixed Cnstl values at 0 inches/hour, 0.04 inches/hour and 0.11 inches/hour. Results show the effects of fixing the Strtl values.

value. The median time-to-peak value produces peak flows that are in between the higher peak flows and the lower peak flows. These plots clearly show the effects that the time-to-peak values have on the overall storm event. This value can sometimes be easily estimated from the rising hydrograph for a storm event that does not have separate rainfall patterns or events where there are no gaps between rainfall events. These types of singe storm events are rare. The typical storm event produces several hydrographs in which the time-to-peak is difficult to estimate. The minimum, median, and maximum time-to-peak values used in the model were obtained from historical calibration models mentioned previously.

Figures 3.10, 3.11, 3.12 and 3.13 are plots showing the results of the Monte Carlo Simulations for the same four storm events of February 20, 1997, January 4, 1998, October 5, 1998, and May 11, 1999 with fixed time-to-base values (Cp). The Cp parameter is used by the HEC-1 program to determine the lower hydrograph width for the storm event. In a unit-hydrograph, the Cp value determines the base width of the plot. This has great influence on the maximum peak discharge that the runoff will produce. The Cp parameter is a Snyders coefficient which represents the percentage that the real hydrograph matches the synthetic unit Snyders hydrograph. A value closer to 1 or 100% produces a slender hydrograph while a value closer to 0 produces a wide based hydrograph. The wider the hydrograph, the lower the peak flow will be. The more slender the base of the hydrograph, the greater the peak flow value. The results of the Monte Carlo Simulation are consistent with the results described. A Cp value of 0.38 produced lower peak flow values for the four figures as expected. A Cp value of 0.96 produce higher peak flow values for the four figures as well. The data from Figures 3.2,

3.3, 3.4 and 3.5 were plotted along with these data. The results of randomizing the values show peak flow values lower than the maximum values. These results also showed peak flow values lower than the minimum values obtained from fixing the Cp values. The lower computed peak flows may be influenced by the one of the other three input variables to the HEC-1 as well. As an example, having lower Cp values with high infiltration rates lowers the computed peak flow. These plots show the influence of the Cp values over the peak flow rates.

Figures 3.14, 3.15, 3.16, and 3.17 shows the plotted data for the same four storm events as mentioned previously, but with the initial loss parameter fixed to either the maximum, the minimum, or the median historical values obtained previously. This parameter used by the HEC-1 software incrementally sums the precipitation data on an hourly basis until this set value is reached. The software then uses the rest of the precipitation data to compute the effective runoff and the peak discharge value. The larger the value for this parameter, the lower the computed peak discharge will be for any storm event. A value of 0 indicates that the soil is saturated and that no more water is able to infiltrate the ground. Soil infiltration is one the largest contributors to the total initial losses. Evapo-transpiration, evaporation, ponding, and storm drains are other factors to consider when trying to determine the correct value for initial losses. The four figures show that the smaller the initial losses value, the larger the peak flow values will be. An infiltration value of 1.55 inches per hour is appropriate, based on previous studies, especially in Oklahoma and Texas, after a long hot summer.

Figures 3.18, 3.19, 3.20, and 3.21 show a plot of the Monte Carlo Simulation results by fixing the constant loss value (Cnstl) for the four storm events studied in this

chapter. This parameter is used by the HEC-1 model to subtract a constant loss every hour from the sum of the incremental precipitation data obtained after the initial losses have been accounted. These four figures show similar results as the previous four figures. The lower the "Cnstl" value, the higher the peak flow, and likewise, the higher the "Cnstl" value, the lower the peak flow. The "Cnstl" is used to indicate the loss of precipitation from either soil, evapo-transpiration, evaporation, and/or ponding.

SUMMARY

The Monte Carlo Simulation technique is a great tool for showing the range of peak discharges that a storm can produce. An experienced forecaster can determine the level of exceedence that a storm can produce and therefore take proper actions to minimize flooding potentials. Since the HEC-1 software was modified for this project to output all four values for each run, the engineer can cross-reference the computed peak discharge with the time-to-peak, percent of base, initial loss and constant loss values. This method of forecasting can be easily developed without much effort to the forecaster. Once the software has been developed to properly implement this method, the forecaster can better anticipate any flooding problems that may occur. Alternative effective runoff scenarios can be viewed by the reservoir control personnel before implementing any costly actions.

HEC has tried to make the HEC-1 software easy to use for an experienced hydrologist. But with the model's simplicity, it is difficult to accurately account for all the conditions necessary to produce the effective runoff of any storm event. The HEC-1 program uses few parameters to define the characteristics of a basin. This makes the

software easier to manage, but difficult to match any particular storm hydrograph. A method of trial and error is used by forecasters to try to determine the best parameter values that will produce the correct hydrograph for the storm event being modeled. Each storm event, due to the dynamic characteristics of a basin, requires different sets of values for each parameter. Antecedent moisture conditions affects the values of the initial losses as well as the constant losses. These values in return affect the time to peak and the time to base as well.

A more complicated software model would require more parameters to adjust, more data to be gathered, and more time to run. HEC and other government agencies have tried more complicated models with little success in the area of forecasting. The failure of these models is mainly due to time constraints in producing an accurate, and timely forecast.

CHAPTER 4

LEARNING HYDROLOGY WITH ARTIFICIAL NEURAL NETWORKS

INTRODUCTION

Mankind has long been fascinated with how the brain learns and processes information. The brain, either from wildlife, domesticated animal, or from a human is a very complex organ. Psychologists and doctors have been trying to understand the process of learning, understanding, and recognition for the last century. One of the most studied features of learning is that of learning a language, more specifically the study of how children learn their language. A child processes visual inputs, auditory inputs, and stimulation inputs to learn the language spoken to them. This, coupled with the physiological growth of the brain, makes learning possible for humans. Over the last 70 years, man's fascination with understanding and reproducing functions of learning has increased significantly. However, Jeff Elmans describes two main problems of understanding the brain process. First, the brain does not work on a list-like basis, and second, the learning behavior is somewhat unpredictable.

The 1900's showed the most progress in discovering the structure of the brain along with understanding algorithms and attempting the replication of the brain's procedural functions. In 1911 Ramon y Cajal studied the brain as a biological organ. He was one of the first pioneers to discover and introduce the functions of the brain as neurons with electrical conduits for passing and receiving information from neurons to neurons. A neuron is comprised of dendrites that receive input from many other neurons. A cell body is a node in the neuron to which the dendrites and the axom are attached.

The axom is the conduit of the neuron that transmits the input signals to the synaptic

terminals. The synaptic terminals are responsible for transmitting all processed signals to

other neurons. Figure 4.1 shows a typical neuron with all of the components labeled.

Simon Haykin describes the brain as having more than 10 billion neurons and over 60

trillion synaptic terminals. The brain is a very energy efficient machine compared to

today's powerful computer, however, it is much slower at doing operations than a typical

computer of today's generation. Simon Haykin estimates that the brain is at least in the

order of 10^{-6} seconds slower than today's computers. Today's fastest computers can do

operations in the order of nanoseconds which is 10^{-9} seconds. Since the algorithms for

learning are not very sophisticated for computers, the brain is still much more powerful

than any computer in existence today. One of the problems with computers that Haykin's

describes is that it is not efficient at processing information in parallel, but it is a singular

connectionist processor. Unlike the brain, computers are not able to process several

objects at the same time, but must process each object one at a time. The closest form of

parallel computing in existence to today is a machine with a maximum of two processors.

However there are very few software or firmware programs that take advantage of these

types of computer configurations. The current market still uses single processing

machines due to the expense of having duel processor machines and the lack of improved

performance of these dual processors.

One of the most prominent early researchers in the field of artificial neural

networks was a mathematical logician in the mid 1930's named Alan Turing. Alan

Turing introduced the first abstract machine which computer scientists and engineers
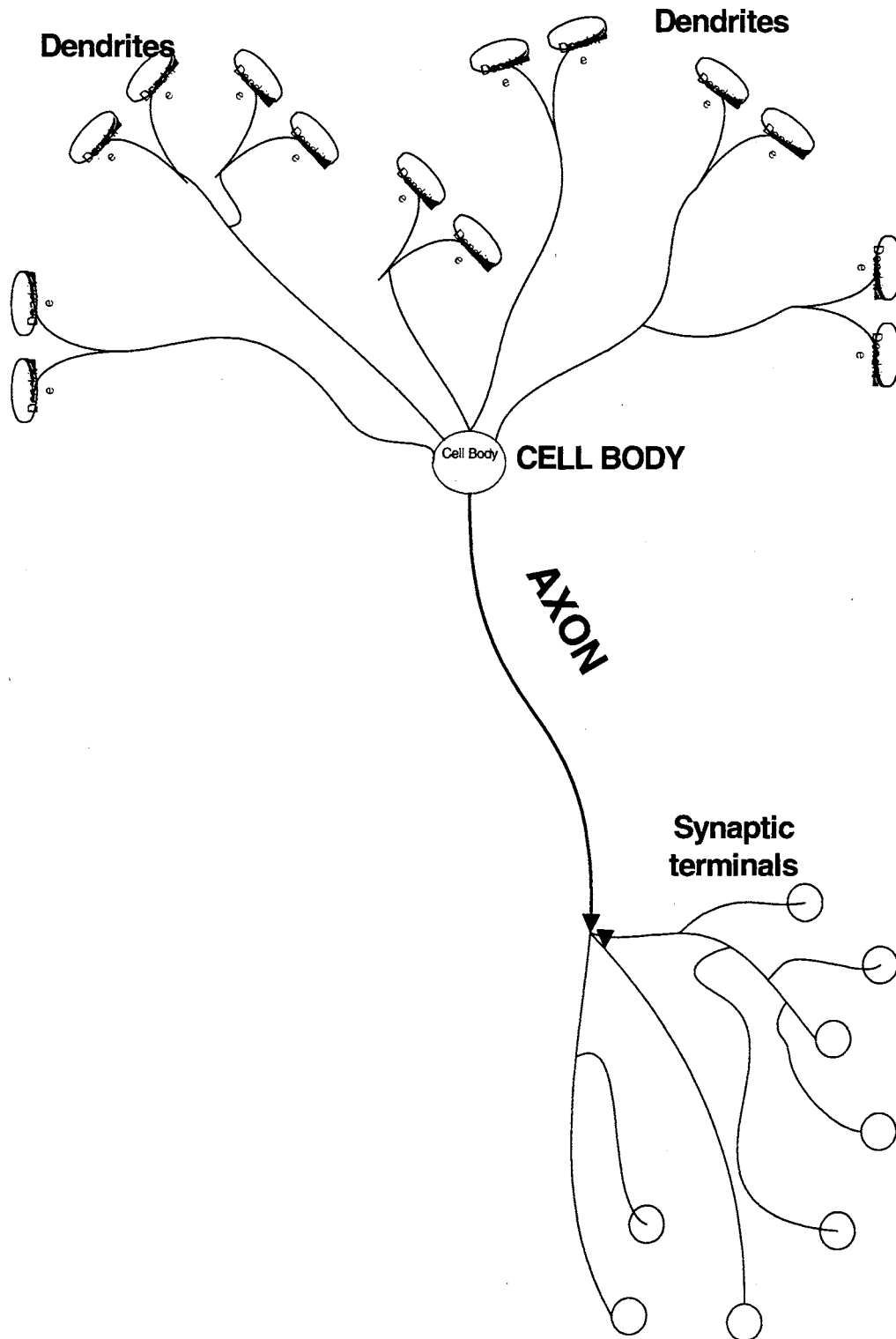
Figure 4.1. A typical Neuron comprised of dendrites, cell body, axon, and synaptic terminals.(Mehortra ,Mohan, and Ranka, 2000)

use today as a model for forming algorithms that solve problems. The first hypothetical machine was a set of rudimentary redundant rules used to store, fetch and retrieve symbols on a serial type device, such as an infinite tape. His machine was hypothesized as a set of list-like commands such that anyone person could read and understand the step-by-step processes of a machine used for storing, erasing, and retrieving data. These types of procedural steps were one of the first algorithms used to solve a simple problem that is used in many computer programs. These abstract machines that are known today as a set of algorithms for solving problems have been named Turing Machines in honor of Alan Turing. Alan Turing introduced solutions to problems that are simple to solve, but redundant in processing. He also provided a means that researchers can begin to use for eventually solving complex real world problems. Computer scientists and programmers are introduced early in their learning that there are many problems that do not have solutions. Therefore, the tasks of these programmers are to find solutions to these real world problems. Turing eventually theorized that a machine can solve simple and complex problems. This machine has been termed Turing's Universal Machine. By taking this idea further, Turing suggested that a universal machine can be formulated to solve problems that other machines can solve. One can then take parts of these machines to begin to solve other more complex problems and create other machines. Some mathematicians and programmers consider the brain as a universal machine that uses many algorithms to solve many problems (Mehrotra 2000). Therefore, there must be a universal machine that can solve problems that this universal machine (the brain) can solve. Researchers within the last four to five decades have been able to take methods that the brain uses to solve problems to create another machine.

Figure 4.2 shows a schematic of a simple Turing machine that can learn a

language of a 0's and 1's such as {0, 1}*. The notation shown – {0,1}*, represents a

function that can be used for repeated patterns of 0s and 1's. These patterns can be a set

such as (01, 0101, 010101, 010101...}. Turing machines may be thought of as a

complicated Finite Automaton (FA) machine or a Push Down Automaton (PDA)

machine that can be used for learning a finite set of states. The FA and PDA machines

have been introduced by computer scholars as machines that can recognize very simple

language patterns. This can be a language of 1's and 0's or a language of letters. A

simple example of a language of letters can be a set of a's and b's with redundant

patterns. For example a language can be a set such as:

{ab, abab, ababab, abababab...} or

{abc, abccba, abccbaabc, abccbaabccba..}.

This set, which is a subset of the English alphabet, repeats itself indefinitely. As shown

in Figure 4.3, Turing Machines have input node(s), intermediate node(s) and an output

node(s). The input is either accepted by the node or rejected by the node thus turning the

node on or off respectively. The intermediate nodes can be visited several times as

necessary to obtain the pattern required. This type of machine is, therefore, constructed

to produce different output signals based on its inputs. Some mathematicians, such as

John Martin, have described this type of machine as being able to learn a set of language

of 1's and 0's or a language containing a subset of some known alphabet. Figure 4.4 is a

Turing machine that can learn a finite set of a palindrome (PAL). A palindrome is a set

of strings that are equal to their reverse string. In other words, it is a string where the first

and last letters are the same. Then the next letter (the second letter) and the letter next to
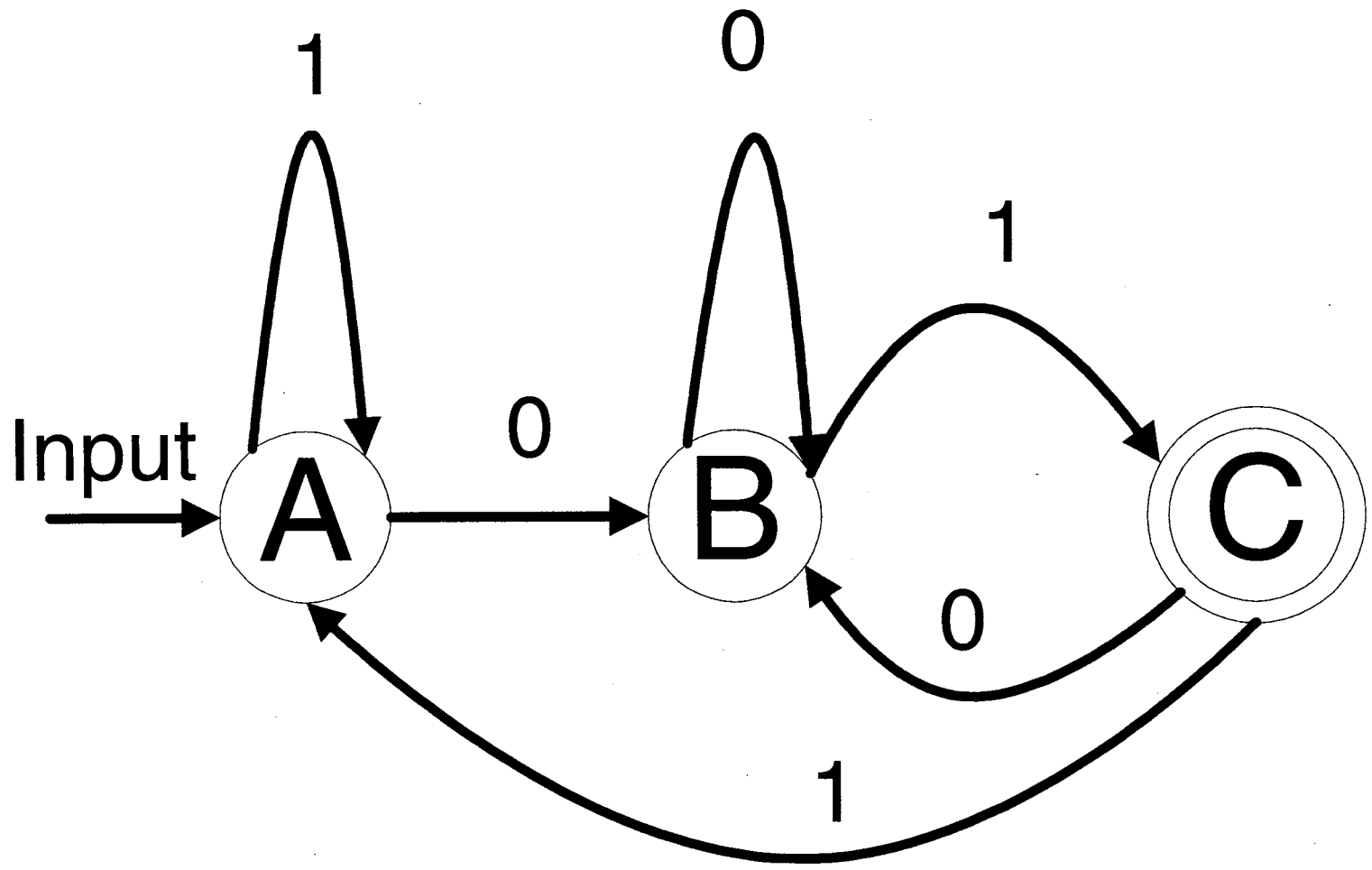
Figure 4.2. Schematic of a simple Turing Machine recognizing a language of (0, 1)* (Martin, 1991).
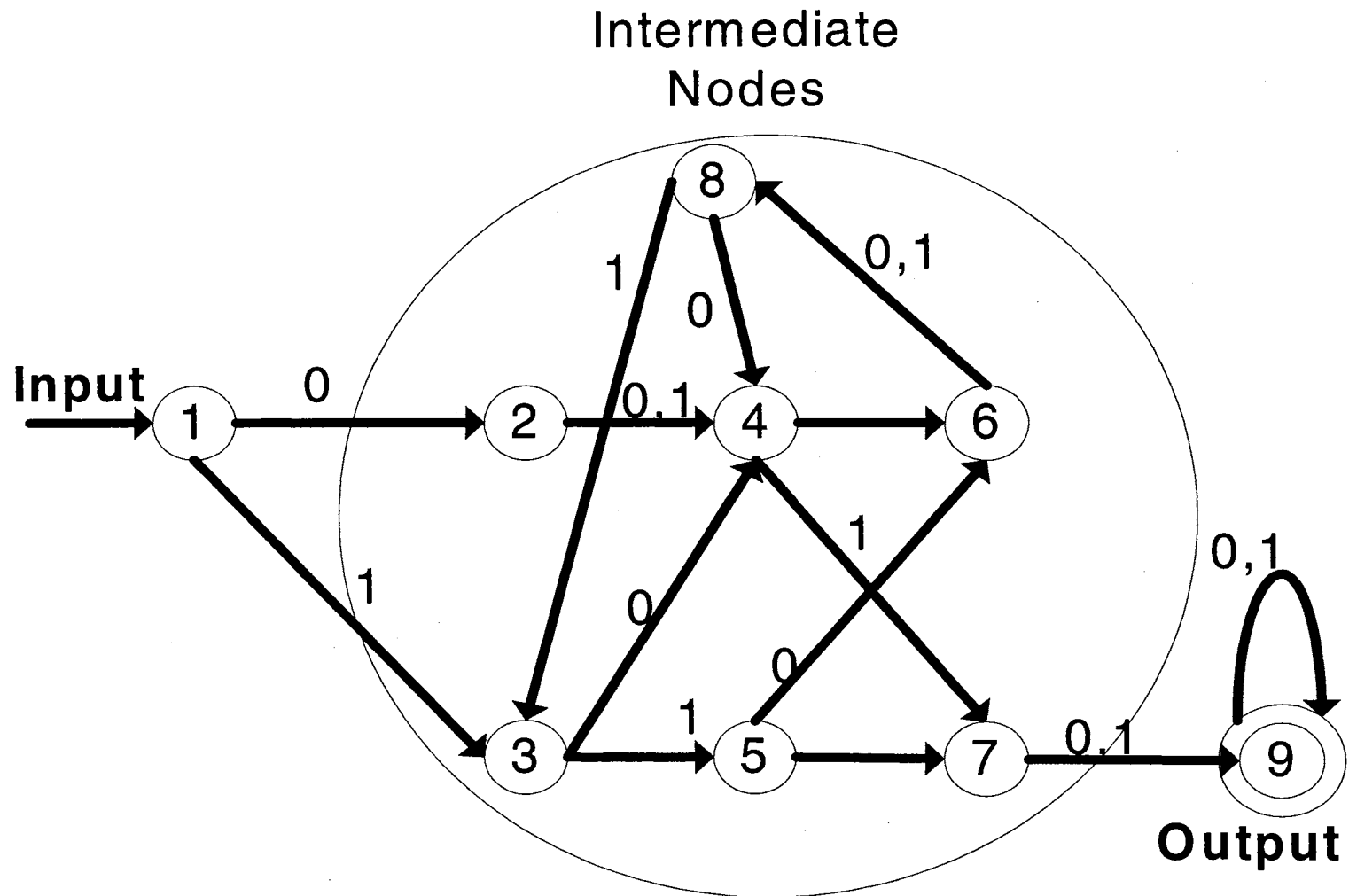
Figure 4.3. An example of a Turing Machine Schematic. (Martin, 1991)
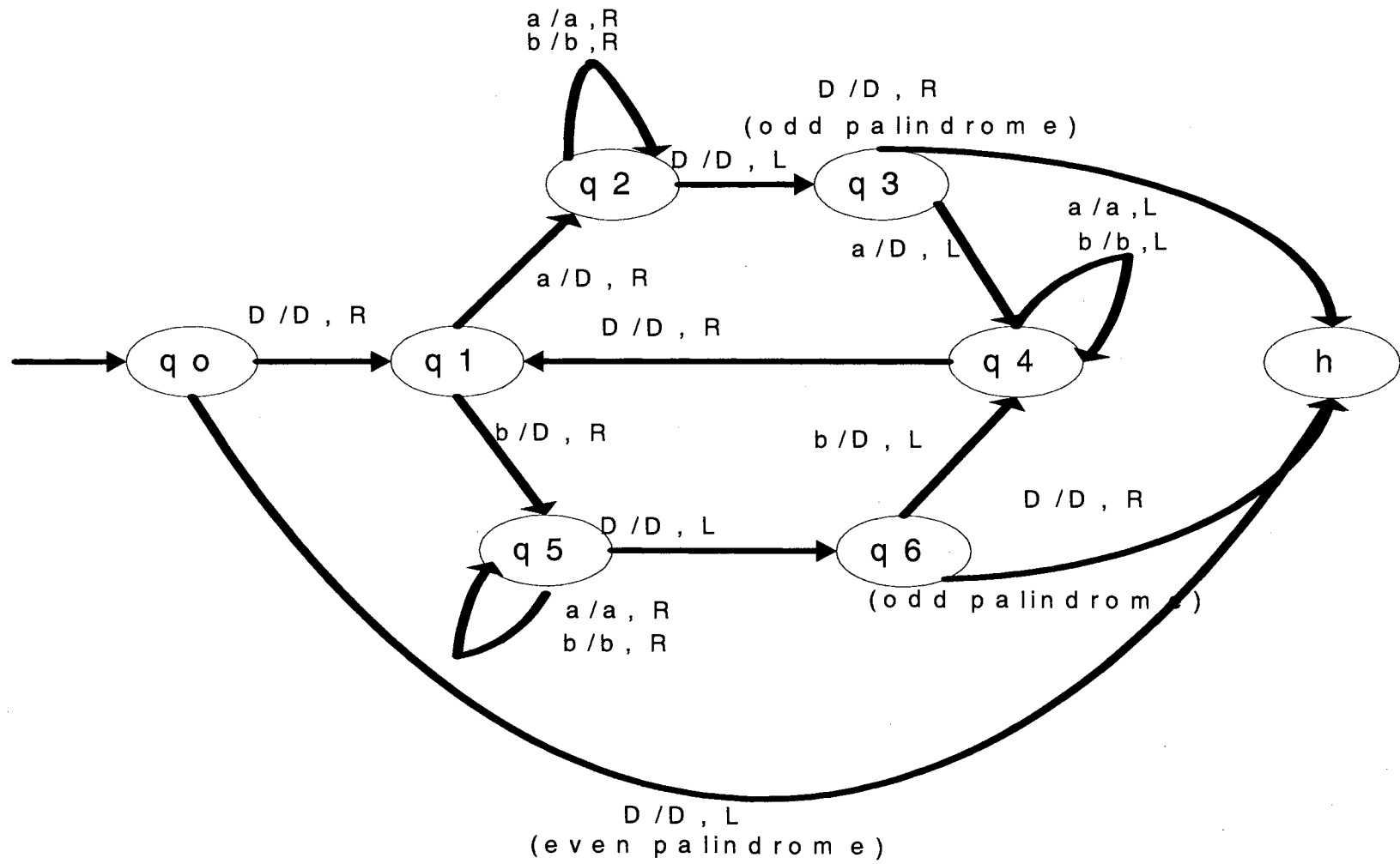
Figure 4.4. Turing machine that can learn a palindrome. (Martin, 1991)

the last letter are equivalent and so forth. As one can imagine, Turing machines can

easily become very complicated. Simple routines of solving problems can increase

exponentially with little effort as the example of the palindrome shown in Figure 4.4.

In 1949, Hebb studied both Turing's thesis as well as Ramon y Cajal's research of

neurons to begin his research on neural networks. The outcome of Hebb's studies

provided the basic learning rules that began the work for writing algorithms for artificial

neural networks. Kishan Mehrotra states that Hebb's learning rule is as follows:

"repeated activation of one neuron by another, across a particular synapse,

increases its conductance."

In a more general form, Hebb's rule is of the following form:

$$\Delta w_{ij} = g(a_j(t), t_j)h(o_i(t), w_{ij}) \tag{4.1}$$

Where:
w_{ij} = weight of the link from unit I to unit j,
$aj(t)$ = activation of unit j in step t,
tj = teaching input, in general the desired output of unit j,
$oi(t)$ = output of unit i at time t.   (Zell et al. 1998)

Many more researchers added more rules and hypotheses for neural networks. However,

the largest and most powerful computers of the 1960's were still not able to handle the

tremendous computations and data required by the artificial neural network (ANN)

functions. The scientific field of ANN's began to flourish tremendously in the 1980's

when affordable personal computers began to emerge. Currently the study of ANN's

have improved dramatically, but it is still in its infancy with much more learning to be

done.

ARTIFICIAL NEURAL NETWORK MODELS

Many dedicated researchers interested in understanding and furthering the work that Hebb began have influenced present day neural network models. Researchers such as Rosenblatt, Widrow and Hoff, Hubel, Taylor, Cowan, and Minsky to name a few have made great contributions to current algorithms used in many of today's ANN models. ANN models, due to its simplicity in usage, have slowly begun to become useful instruments in the scientific and engineering field.

A very simple and rudimentary definition of present ANN models is that it is a pattern recognition model. The ANN model is comprised of networks with input and output data and a hidden layer of nodes which forms the connectivity between the input and output layers. The network examines the relationship between the input and output data and formulates a function or a relationship between the data. The function or functions between the input and output data is the learning that the network accomplishes. This type of learning has been described as a function that a neuron in the brain accomplishes. One can also imagine the hidden nodes as a Turing machine able to learn the language necessary to translate the input parameters to come up with the output parameters. However, the algorithms required to learn these patterns are much more complicated as those described in the previous section.

A basic ANN model requires a set of input parameters and its corresponding output parameters mainly used for training the model as shown in Figure 4.5. Most of the current models require two to three sets of learning data for a model. The first set of data is used to train the model. The second set of data is used for testing the resulting functions of the training set, (Mehrotra, Mohan, and Ranka, 2000). Some use this second

79
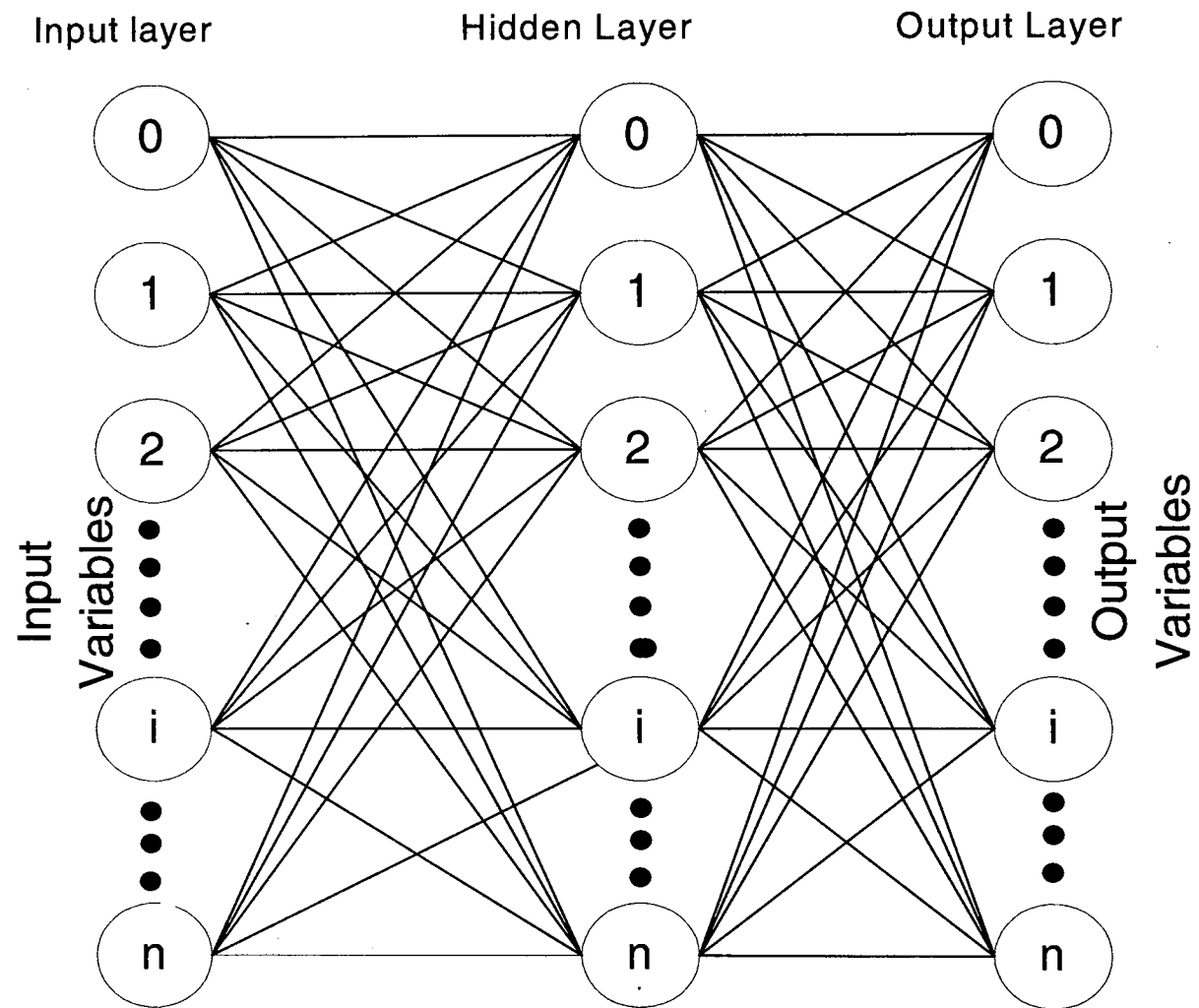
Figure 4.5. A typical ANN network model. (Mehortra ,Mohan, and Ranka, 2000)

set to validate the training model as well. Others use a third set of data for validating the training set. Once the learning process of the network is completed, the neural network model can be used for obtaining results from other similar input data. Some have described this type of algorithm capablility of modeling very complex non-linear relationships.

The hidden nodes mentioned previously can be described as a "black box" where you feed it some values and the black box, through previous data learning sets, adjusts the data and sends the corresponding output values. This black box is the major component of an ANN model. This box or hidden layer is composed of hidden nodes. Each hidden node, therefore, learns a function based on all the inputs taken together. The combination of hidden nodes is described as the hidden network layer in the ANN model. This layer is where the network learns the relation or algorithm for the model. Figure 4.6 shows a simple explanation of a hidden node. The hidden node can be thought of as the following function:

$$F(I) = \Sigma(\text{inputs X weights}) \tag{4.2}$$

Various algorithms are used to derive these weighted values, used in the neural network. Vemuri and Rogers, 1994, have shown and described the successfulness of this method for modeling complex nonlinear input-output relationships in various fields of engineering and science.

LEARNING THE RAINFALL-RUNOFF PROCESS

The rainfall-runoff event in a basin is a dynamic, non-linear process that is very difficult to describe and simulate using rudimentary mathematical models. The

Input values

Hidden
Node

Output values

$$F(I) =$$
Input1 * weight1+
input2 * weight2 +
input3 * weight3
+...
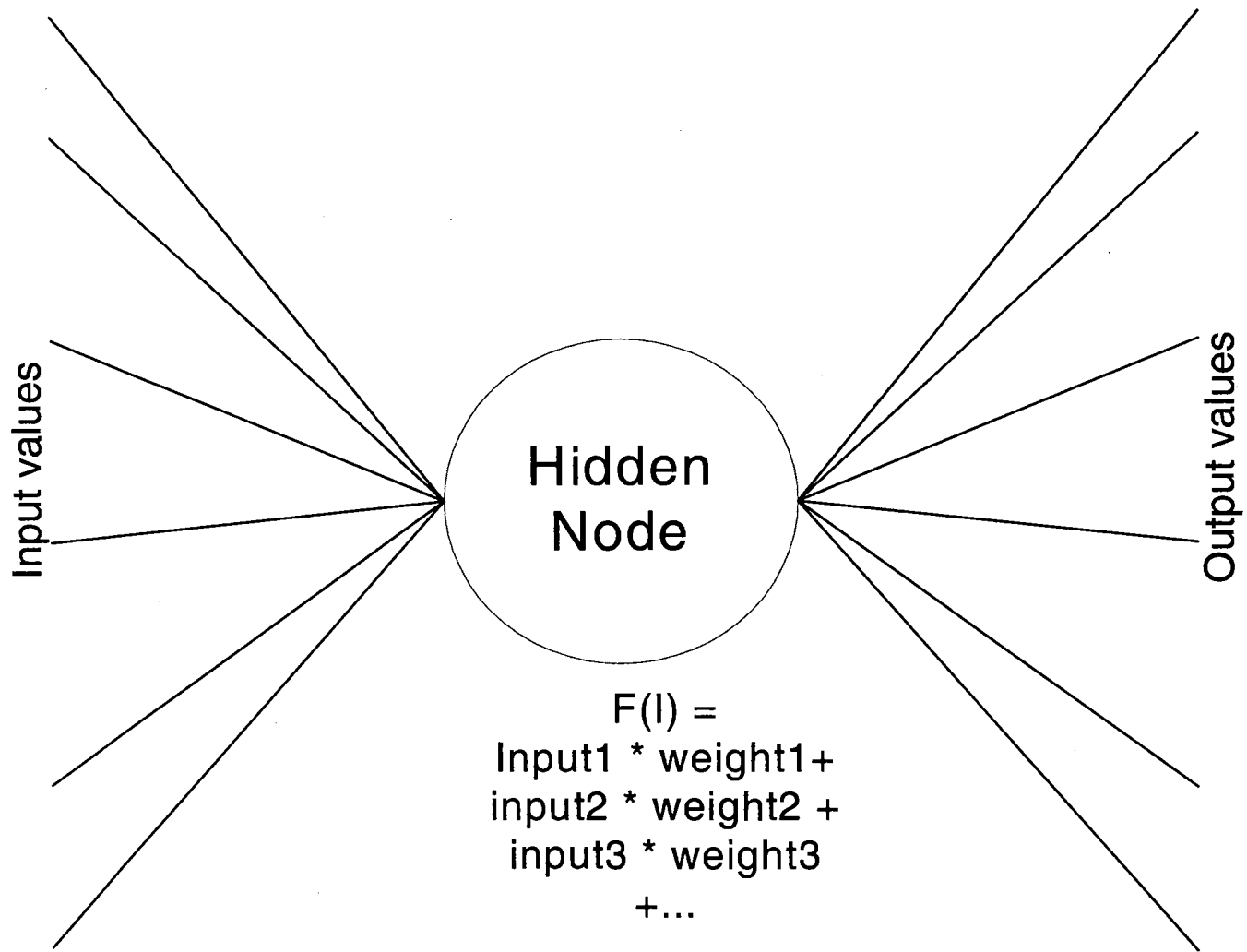
Figure 4.6. Hidden node adjusts the input values through some weight functions.

Hydrologic Engineering Center (HEC) located at Davis, California, has reduced this problem into a rudimentary one-dimensional problem for simplicity and for manageable model simulation. They have attempted more sophisticated modeling techniques and have achieved some, success but at the expense of a more complicated modeling solutions and enormous data requirements. The costs associated with a finite element analysis, which HEC has tried in the past, makes these types of comprehensive projects too costly and time consuming for timely forecasting. The approach for the remainder of this study is to use available data and use the neural network approach to provide the forecaster with results which can be used for analyzing the rainfall-runoff process in a more timely manner and without much difficulty in usage. The remainder of this chapter describes the process of building an ANN model for forecasting hourly water discharge for the Glover Basin located in southeastern Oklahoma. This section and the remaining sections also show that an ANN model can be used as an aid and not necessarily as a replacement for the more acceptable approach of using the HEC-1 modeling scheme.

The Glover Basin is a 325.25 square mile basin with many dynamic features constantly changing. The ground water level fluctuates every season, the roughness of the channel and of the basin used for computing the retarding process of runoff changes drastically from very high roughness to very low roughness, depending on the vegetative growth stage. The basin has very steep slopes at the upper one-third portion of the basin and very gentle slopes at the lower end of the basin. The upper portion of the basin is a protected wild life refuge which means that very little land change occurs at the hand of human intervention. However, the lower portion of the basin contains several communities that are adding more paved roads, buildings, and parking lots which change

the infiltration rate of this portion of the basin. Several modeling techniques used to model this basin have produced little success. The ANN modeling technique was, therefore, used to determine if a pattern of rainfall to runoff can be learned.

Presently the U.S. Army Corps of Engineers has two gage stations located within the basin. One station is located at the mouth of the basin and the other station is stationed at a half way point in the river between the mouth and the most upper portion of the basin. The upper gage station collects rainfall data while the lower gage collects both rainfall data as well as river stage data. Since the task of an ANN model is to learn the pattern between the rainfall and the runoff, it was decided not to use the rainfall data directly from either of these gage stations. Rainfall over this basin can occur over the whole basin or over certain regions of the basin, it seemed appropriate to use spatially distributed rainfall data without averaging the data over the basin. Therefore, Stage III radar data was used for the part of the input to the ANN model. Stage III data uses gage data indirectly for calibration of the radar data collected.

Stage III data over this basin were available since late 1996. Stage III data are rainfall data that have been calibrated with local mid-west gage data using algorithms developed by the Tulsa District U.S. Army Corps of Engineers. Stage I and Stage II radar data are available over this region for years prior to 1996 but it was recommended that data not be used since they were mainly collected in the learning stages of radar data collection. Stage III rainfall data are collected on a 4km X 4km grid cell as shown in

Figure 4.7. Due to several months of missing stage data for the year 1997 it was determined not to use the any data from 1997 in this section of the study. The radar and stage data from 1998 and 1999 were used for this study.

The standard back propagation modeling technique was used for this research. This network model is based on a feed-forward neural net, with a logistic activation function. This is the most common type of neural network in use and in most cases the most appropriate method to use (Zell el al. 1998). In a feed-forward, network a connection may exist only from a node in one layer to a node in another forward layer as shown in Figure 4.8. But a connection may not exist between nodes in the same layer. In this feed forward network the hidden nodes perform a two-fold function. First the hidden nodes compute a value based on all the coming information. Second they transform the value using a nonlinear activation function into an outgoing signal in which the receiving node conducts the same two processes. This process is repeated until the last receiving node is an output node. Zell et al. (1998) describes the value calculated by the receiving node as the following relation:

$$S_{jk}(t) = \Sigma W_{i,j} \, O_j(t) - \Theta_j \tag{4.3}$$

where $W_{i,j}$ is the weight associated with the connection from the input node and the receiving node. Node i being the input node and j being the receiving node. $O_i(t)$ is the input value of the input node and Θ_j is a threshold value that is different for each node. The activation function described by Zell et al (1998) is of the following form:

$$f(S_{jk}(t) = \frac{1}{1 + \exp[-S_{jk}(t)]}, \, f(S_{jk}(t) \in [0{:}1]. \tag{4.4}$$
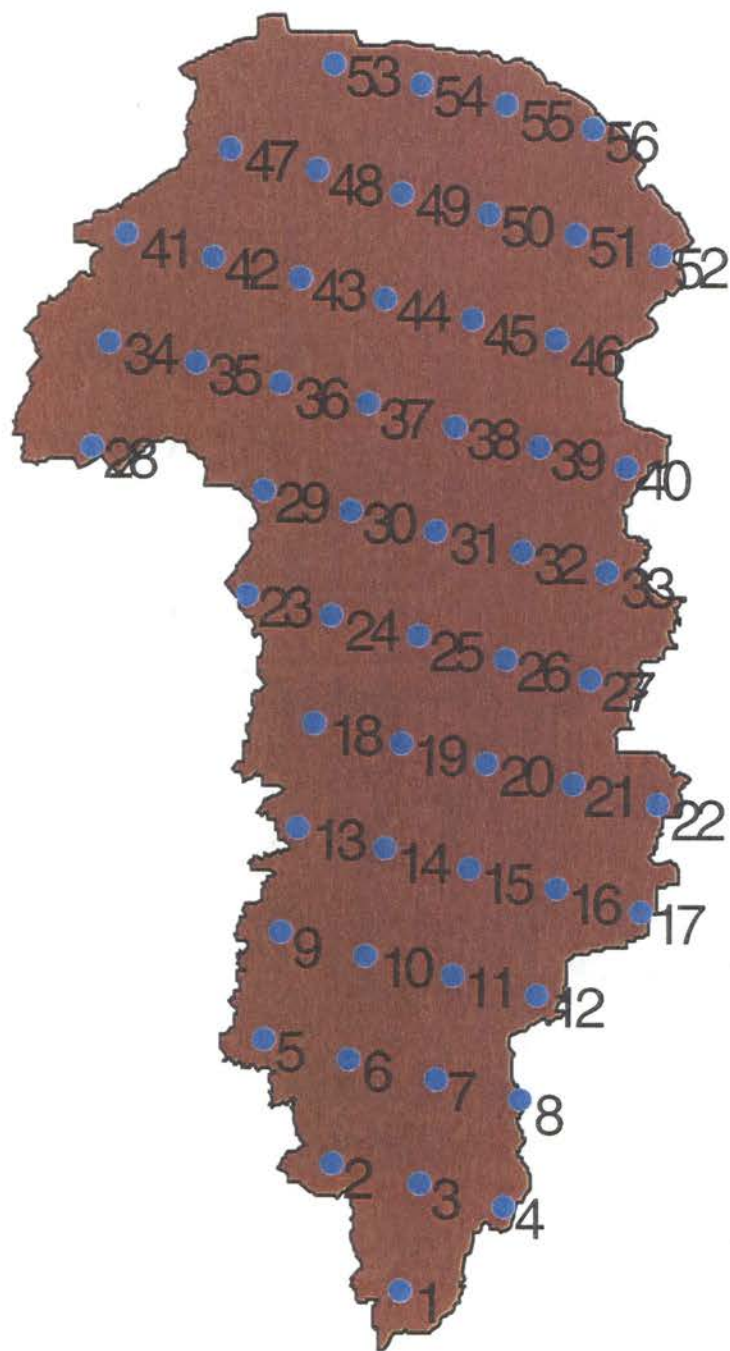
Figure 4.7. Stage III radar points for the Glover Basin. Graph plotted with ArcView®.
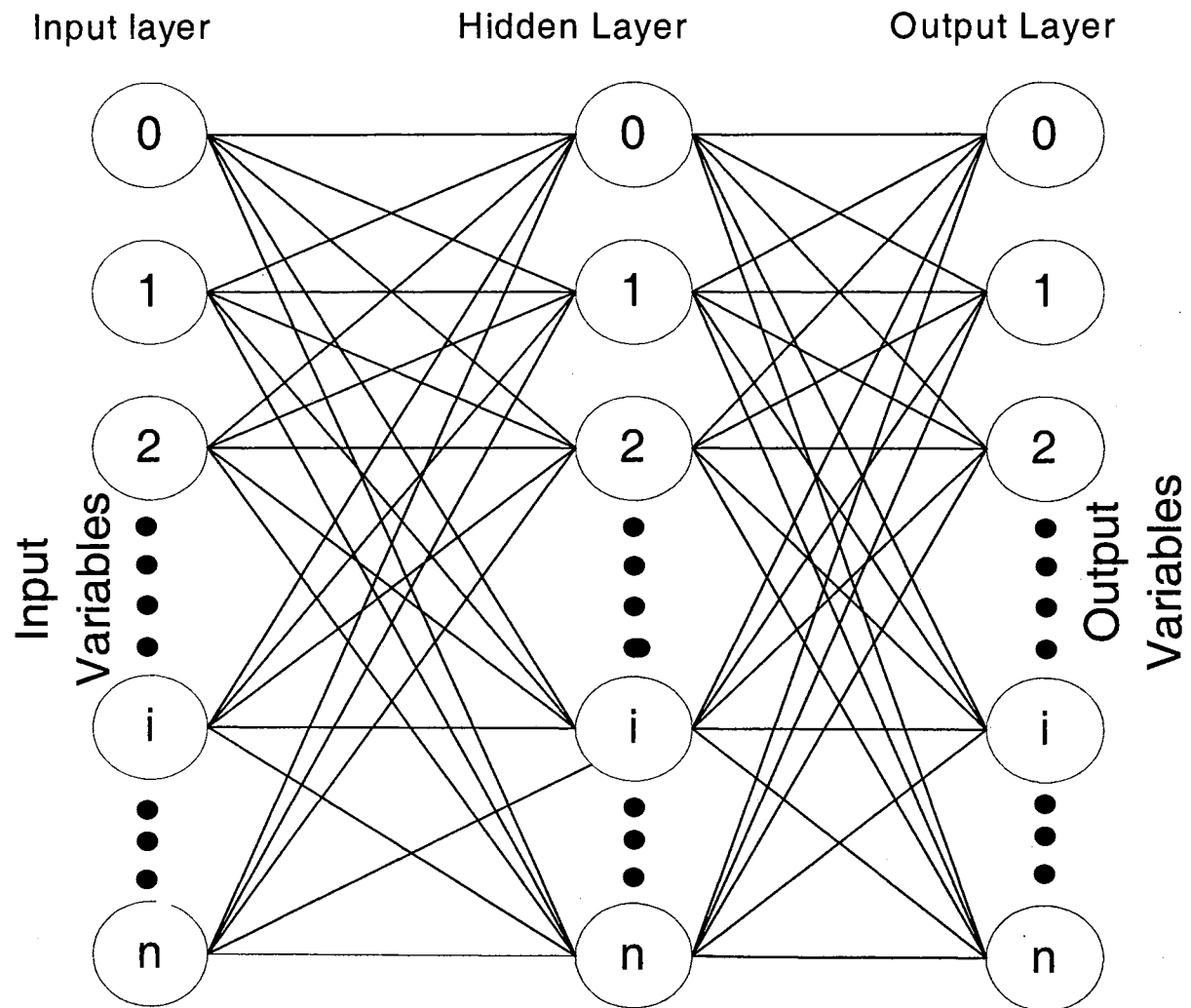
Figure 4.8. Network connections are from the input nodes to the hidden nodes, and from the hidden nodes to the output nodes. (Mehortra ,Mohan, and Ranka, 2000)

This outgoing logistic signal is then fed to a node in subsequent layers and the process of these two relationships continues until the logistic signal reaches an output node.

Stuttgart Neural Network Simulator (SNNS) software developed by the University of Stuttgart was used for implementation of this neural network model. The back propagation learning algorithm, which is the most commonly used algorithm, was used for this research. The learning or training of the network in this software package is done by comparing the output layer with the teaching-input real data. The square of the error of the model with regard to the real data is used to compute the necessary changes of the weight links. Since no teaching inputs are available for hidden layers a formula is used to compute the changes necessary for the inner weights. Therefore, the errors are propagated backwards. The change in the weights for the hidden nodes is calculated using the following formula:

$$\Delta W_{i,j} = -\eta(dE/dW_{i,j}), \tag{4.5}$$

where

η is the learning rate,

dE is the error of real data vs. output data ,

and $W_{i,j}$ is the weight of the node.

For this study the learning rate used was in the range from 0.5 to 0.01 time steps.

RAINFALL-RUNOFF ANN MODEL

In this work, four ANN models were developed to predict the discharge at the mouth of Glover Basin. Each model was used to forecast the discharge of the river at different time intervals. The first model was used to predict the discharge at one hour in

advance, the second model was used to predict the discharge three hours in advance, the third model was used to predict the discharge six hours in advance, and the last model was used to predict the discharge nine hours in advance. The input data for each model differed slightly depending on the forecast to be made. The database at the Tulsa District Corps of Engineers and at the National Weather Service was researched for hourly data that could be used for this work. The only data available was Stage III data. Hourly evaporation data, temperature data, wind velocity data, and solar radiation data are destroyed at the last day of each month to maintain database space for rainfall data. Thus, Stage III radar data and previous hourly discharge data were the only data used for these models. A column for seasons was also used in the input data to the models.

Before the rainfall data and discharge data could be used in the network models, the data had to be normalized between 0 and 1. Hourly rainfall data for the years of 1998 and 1999 were normalized using the Log Pearson type III distribution. Since a zero rainfall value was encountered numerous times in the data and the zero value does not contribute any value to a forecast, it was eliminated from the data set used for the distribution function. The Log Pearson type III distribution was used since it could give a linear interpolation of rainfall events greater than the available data contained. Fifty-six nodes from the radar data were used for the Glover Basin study. These nodes encompassed all the data necessary to conduct the network study for this basin. All non-zero values from each node for each hour for the two years were used to obtain the Log Pearson Type III distribution graph shown in Figure 4.9. The data from 1997 was not used, since the corresponding discharge hourly values for several months were missing from the data due to malfunctioning equipment. There were very few discharge data
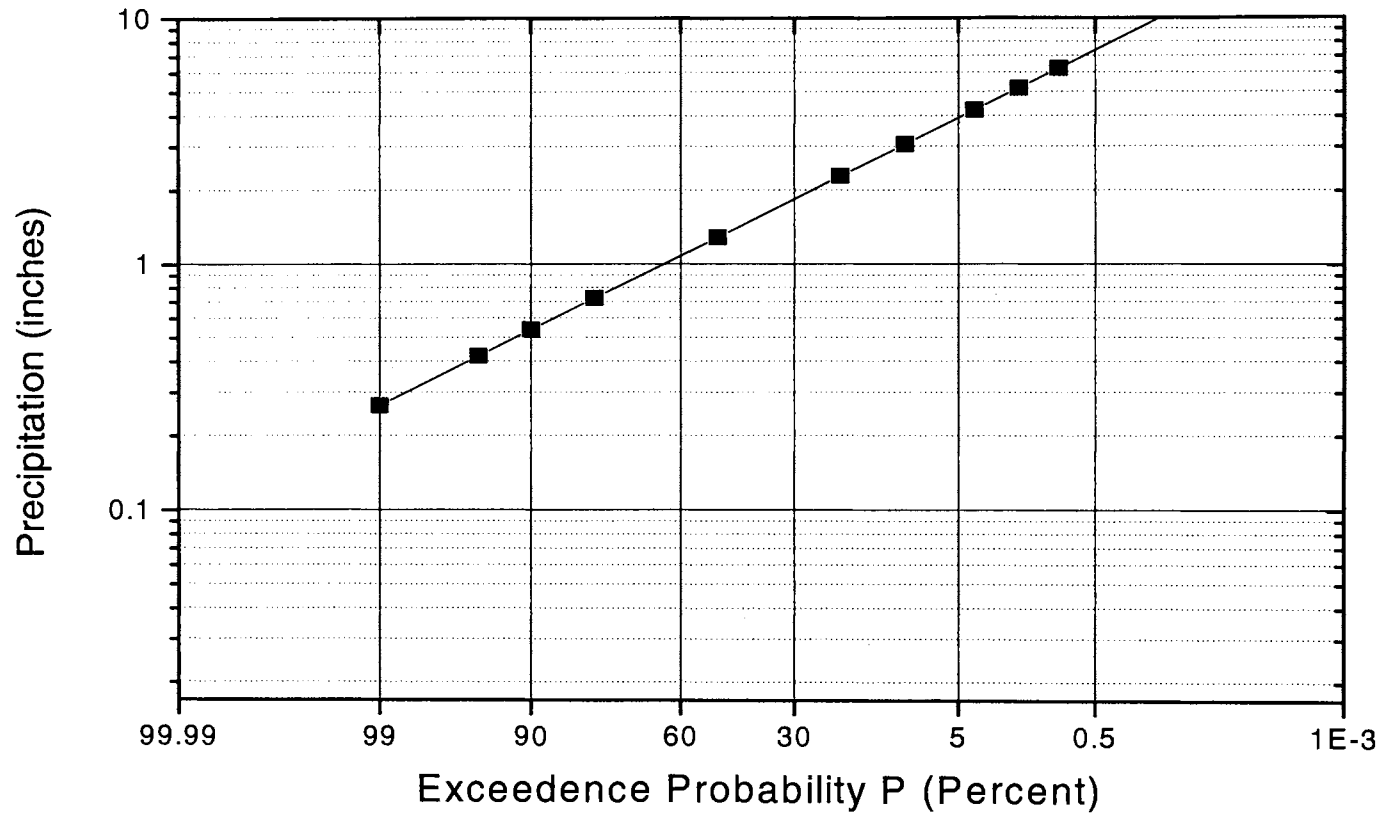
Figure 4.9. Log Pearson Type III distribution plot for the precipitation data.

missing from the database for the years 1998 and 1999. The rainfall data corresponding

to this missing discharge data had to be eliminated from the data set. The hourly

discharge data for the years of 1998 and 1999 was also normalized between 0 and 1 using

the Log Pearson Type III distribution. Figure 4.10 shows the results of the distribution

for the discharge data. The Log Pearson Type III distribution function was also used to

normalize the discharge data since it could also produce a linear interpolation of

discharge greater than the given data set.

The input data for the training network consists of columns for winter, spring,

summer, fall, and 56 columns for the 56 nodes of rainfall data from radar, and several

columns for previous hourly discharge data, depending on the type of model used. Since

the input data has to be between and including 0 and 1, the convention used for the

seasons was to use a 1 for the corresponding season and 0 otherwise. A typical input data

format is shown in figure 4.11. The complete data for each model consisted of the

maximum of 6762 rows. Appendix C contains all the data used for the input for the four

models. The three-hour, six- hour, and nine-hour models contained less data than the

one-hour data but was still in the range of 6,000 data sets. In neural network

terminology, each row of input and its corresponding output data is called a pattern.

Therefore the models have more than 6,000 patterns from which the network can learn

from. The final selection of the structure model was done by trial and error by

optimizing the number of nodes to perform the best pattern recognition. Kishan Mehrotra

suggests that the maximum number of hidden nodes for a typical model be less than or

equal to 15 nodes per layer. Research done by various other scientists on other projects

indicated that one layer for the hidden nodes should be sufficient for this study
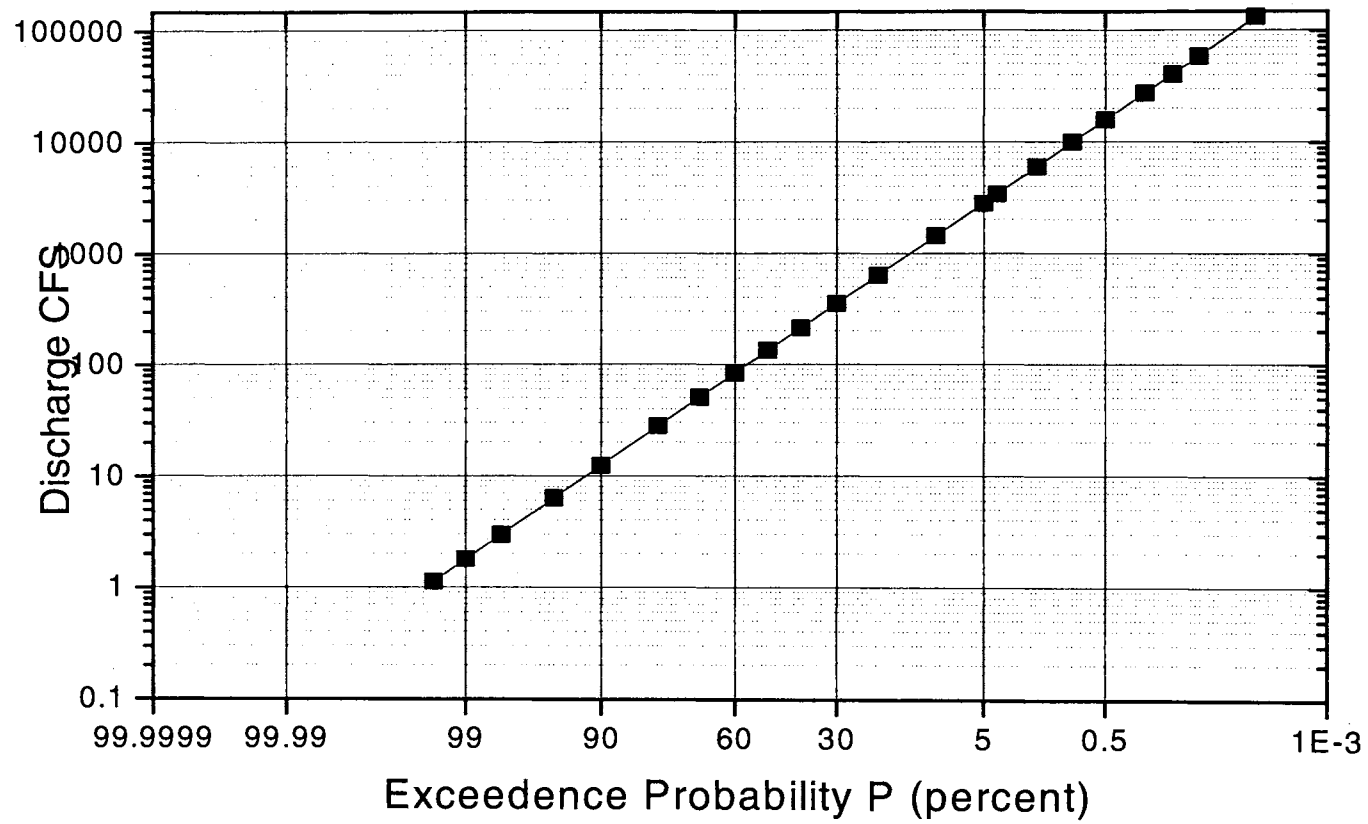
Figure 4.10. Log Pearson Type III distribution plot for the discharge data.

SNNS　　　pattern　　definition file　　　V1.4
generated at　　　　Tue　　Dec　　　　　　3 20:36:50　　　1991

No.　　　of　　　　patterns :　　　　　7289
No.　　　of　　　　input　units　　:　　　　　　59
No.　　　of　　　　output　units　　:　　　　　　1

#					Node-0	Node-1	Node-2	Node
#	Winter	Spring	Summer	Fall	Jan	98	740	3
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0

Figure 4.11. Typical input data format for the Stuttgart Neural Network model. (Zell, 1998).

(Campolo, Soldati, and Andreussi 1999). The 1998 data were used for training and the 1999 data were used for testing. The training and testing was done for each model separately until the mean square error of the testing data was at its minimum. Once the mean square error reached a minimum, the training was stopped. Even though the mean square error for the training data kept continuously getting smaller, the mean square error for the testing data reached its minimum and began to get larger. This effect of having the mean square error begin to get larger once the minimum is reached is called network over-training. One set of training and testing was done for a particular number of nodes, the next iteration was done with one less node in the hidden layer structure. This process was continued until the network contained one node in the hidden layer. The best model for each of the four scenarios was chosen by looking at the model with the smallest mean square error in the testing data. The results that produced the smallest mean square error corresponds to the optimum number of hidden nodes that will produce the best results for the testing and training set. Figure 4.12 thru Figure 4.15 shows the mean square error of the data vs. the number of nodes. The results shown in the figures indicate that the testing error is much greater than the training error. These results are typical of an ANN model. The model with the minimum mean square error is the optimum model that will give the best results thereafter (Zell 1998). Figure 4.12 showed the lowest mean square error for the testing set of all four models. A value of less than 0.001 was obtained for a hidden layer with four to eleven nodes. Figure 4.13 showed good results of the testing set as well, even though the lower mean square error value was greater than 0.001. Figures 4.14 and 4.15 increasingly show poorer results for the testing set. The weighted values of the hidden nodes obtained for the training set do not respond well to the testing data for

Figure 4.12 – Mean square error vs. Number of hidden nodes for the One-Hour Forecast.
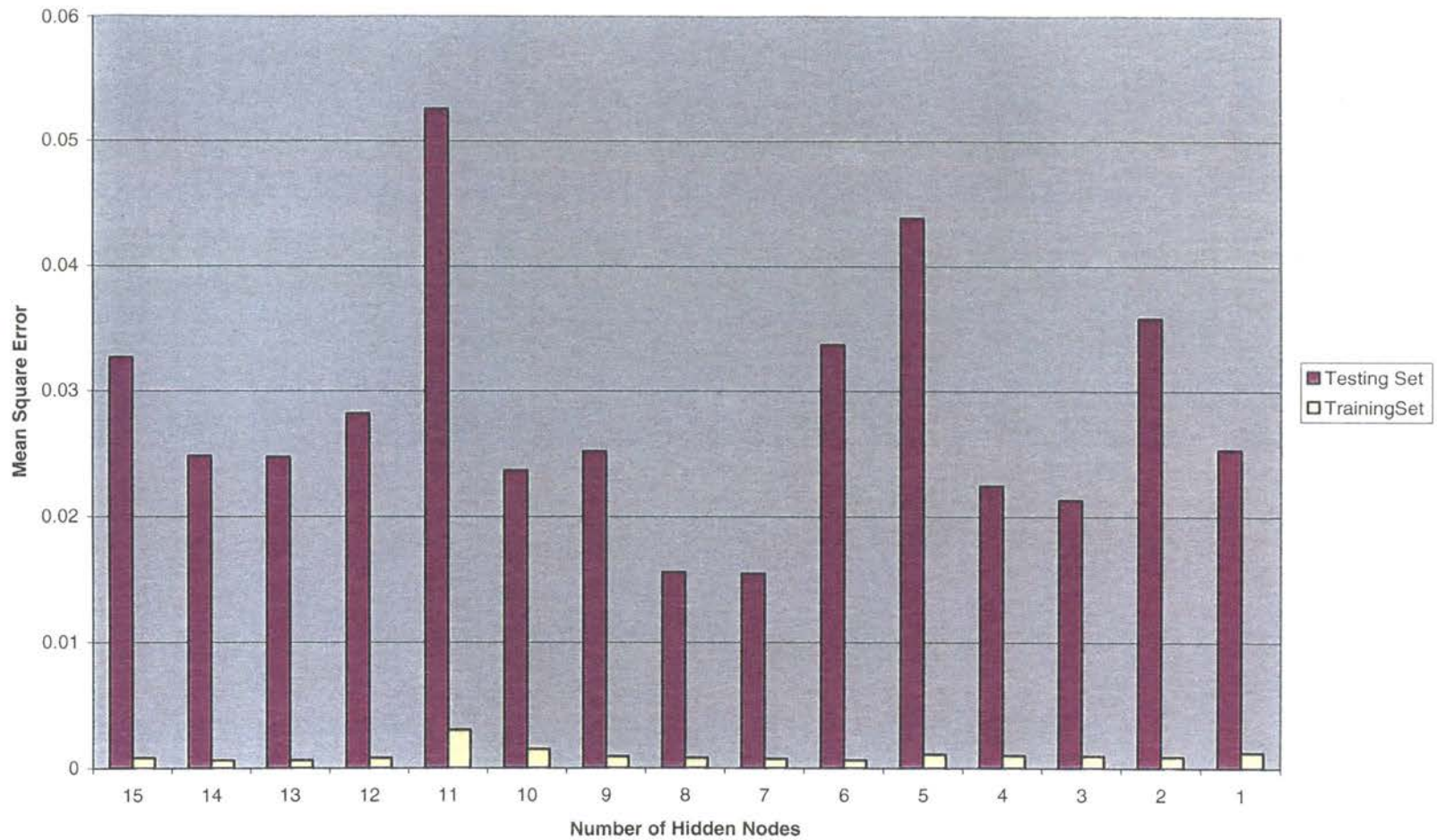
Figure 4.13 – Mean square error vs. Number of hidden nodes for the Three-Hour Forecast
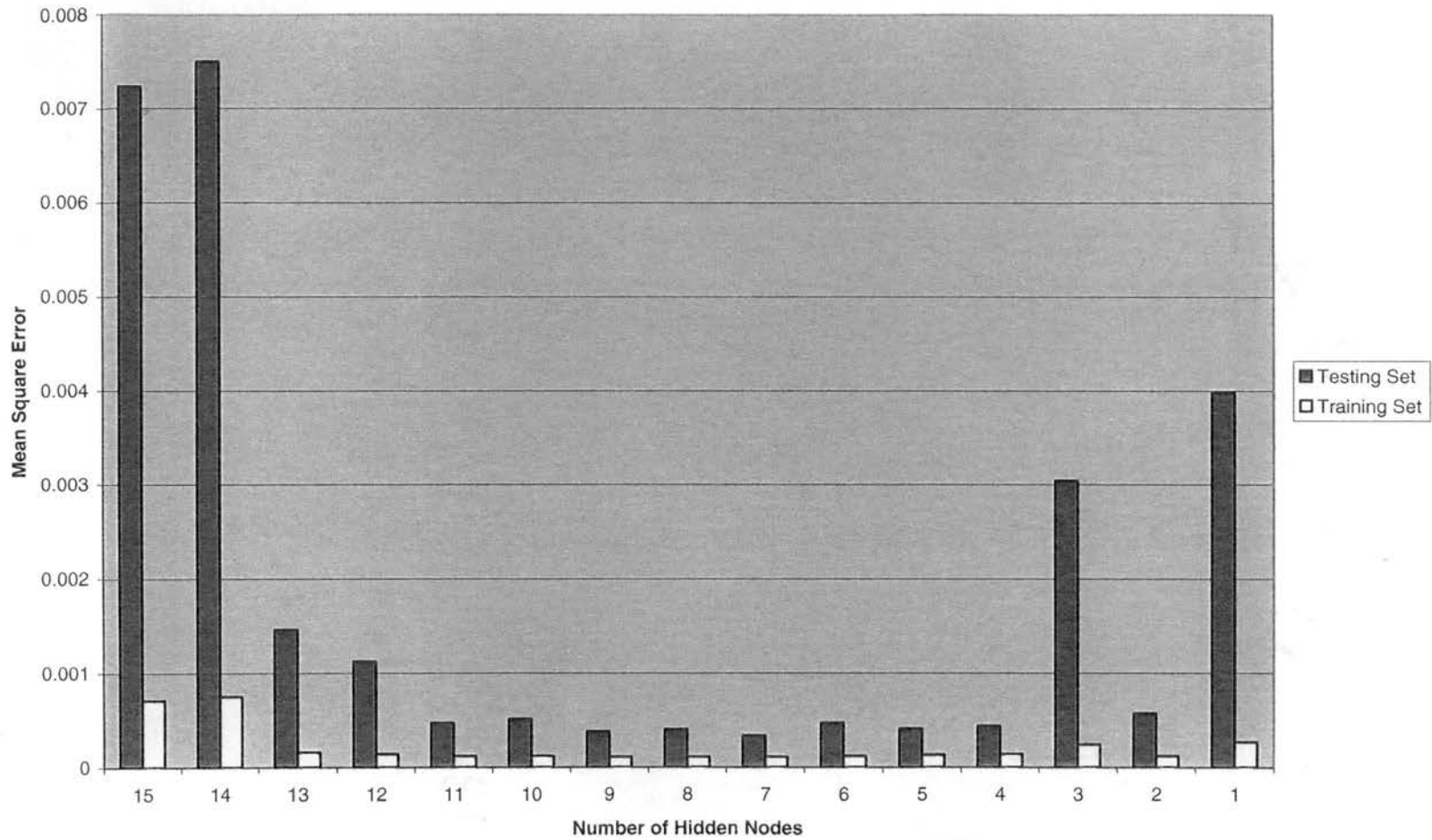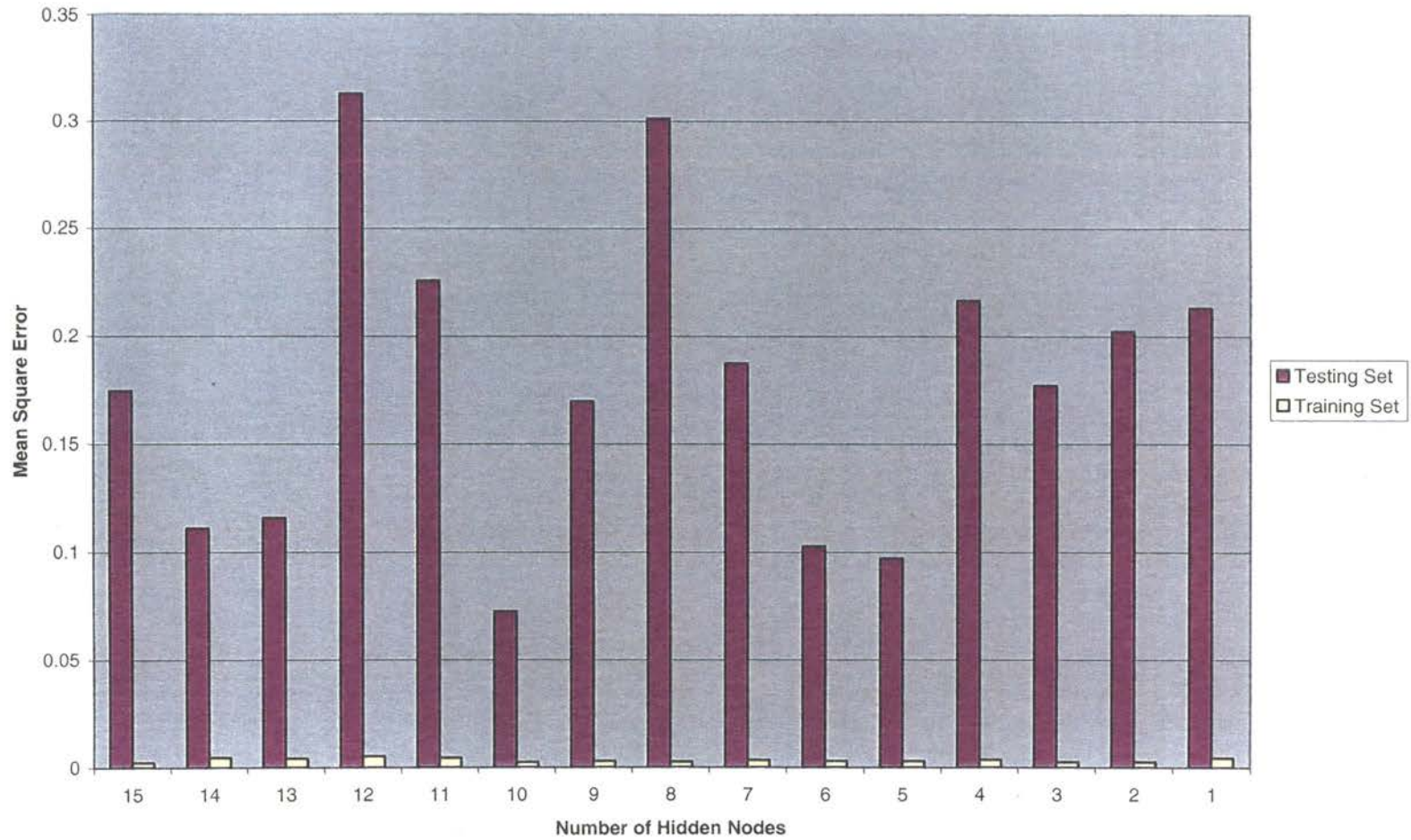
Figure 4.14 – Mean square error vs. Number of hidden nodes for the Six-Hour Forecast.
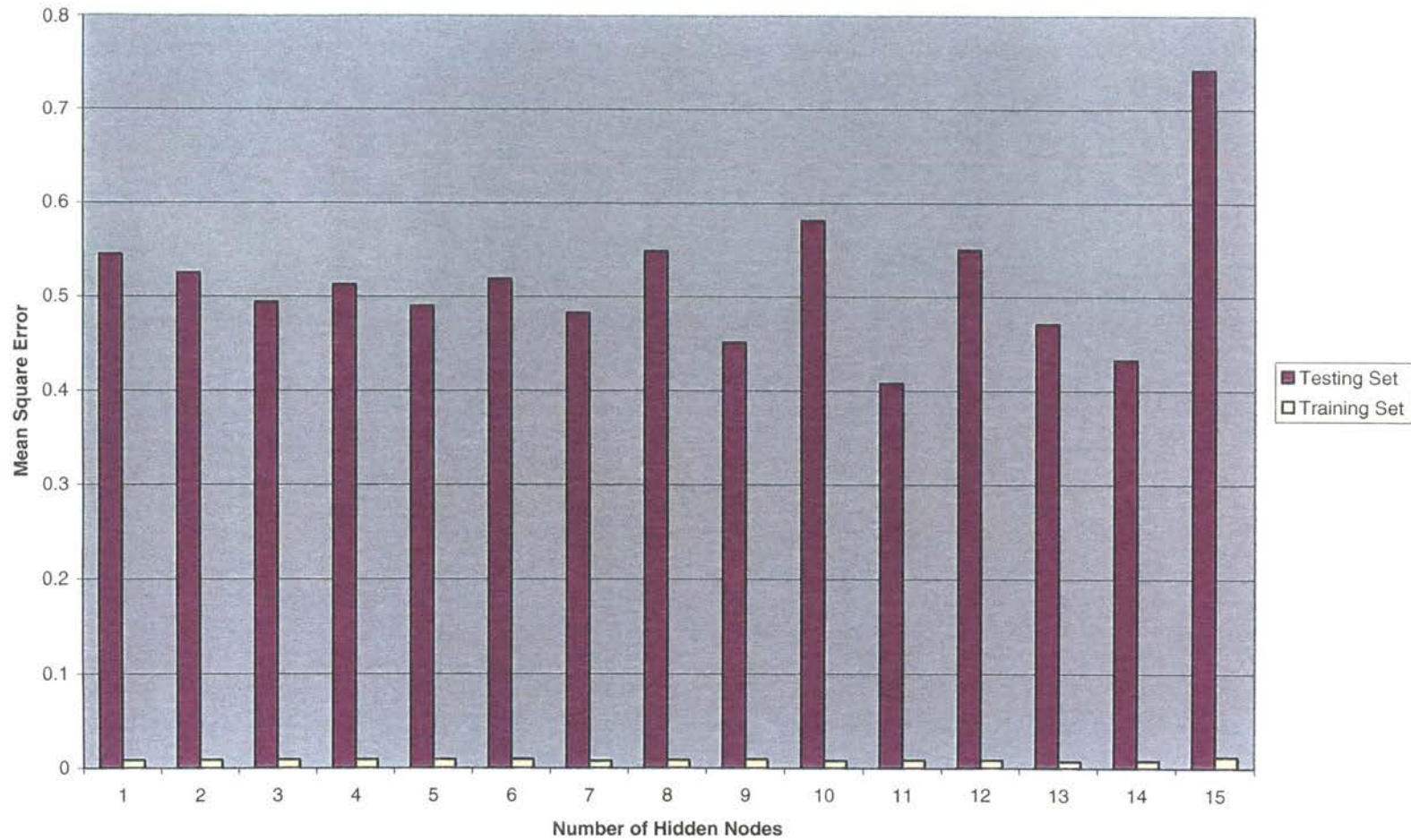
Figure 4.15 – Mean square error vs. Number of hidden nodes for the Nine-Hour Forecast.

the six-hour and the nine-hour testing set. This indicates that the network requires more data to better find a proper pattern. These results indicate that the longer the required forecast, the more difficult it is for the ANN model to learn the future characteristics of the basin.

RESULTS OF THE ONE-HOUR FORECAST

From the results shown in the previous figures, the one-hour forecast prediction had the smallest mean square error for the testing data as compared to the other three. This, however, is just one parameter to examine in determining the adequacy of the model. Campolo suggests looking at several other parameters to determine the adequacy of the model. The root mean square error, the mean square error, the percent of patterns exceeding the root mean square error, and the model efficiency are the four parameters recommended for determining the model adequacy. The first three parameters are common statistical parameters that can be solved easily. The fourth parameter is one that Campolo formulated by using the mean value of the data, the real output data and the computed output data. For this study the model efficiency formula (R2) is of the following form:

$$R2 = \Sigma(Qi - Qm)^2 / \Sigma(Qi - Qc)^2 \qquad (4.6)$$
Where

Qi is the observed discharge data used for training,

Qm is the mean discharge data and Qc is the computed discharge data

from the network model.

Table 4.1 shows the results of the four parameters used to determine the model adequacy for the one-hour forecast. The best model configuration for the one-hour forecast model

as shown in Figure 4.12 is the model with seven hidden nodes. This model structure was

used for the final model of the one-hour forecasting analysis.

Table 4.1. Model Adequacy Error Data

RMSE	0.018
MSE	0.00011
$N_{RMSE, PERCENT}$	3.86
$R^2_{, PERCENT}$	99.95

As seen in the table, the error values are very small. A root mean square error of 0.018

for the normalized data represents an error of 1,830 cubic feet per second in the real data

values. The percentage of the data that exceeded the RMSE value were calculated from

the data. The N_{RMSE} value indicates that 96.14% of the computed flows were smaller

than the RMSE and the efficiency (R^2) of the model was very good with a value of

99.95%. According to the values presented in table 4.1, the model predictions are very

good. A plot of the normalized observed discharge values vs. the normalized computed

discharge values is shown figure 4.16. The computed discharge values as shown in the

scatter plot in figure 4.16 closely approximates the optimum required line. This is a very

good indication of the network's ability to compute discharge values based on the rainfall

input. A plot of the comparison of the observed normalized discharge and the computed

normalized discharge data vs. time was done for the entire year of 1998 as shown in

Figure 4.17. Overall, as seen in the figure, the computed data closely fit the observed

data well. There were very few computed data sets that were not as close to the

observed data as expected. As indicated in Table 4.1 a little under 4% of the computed
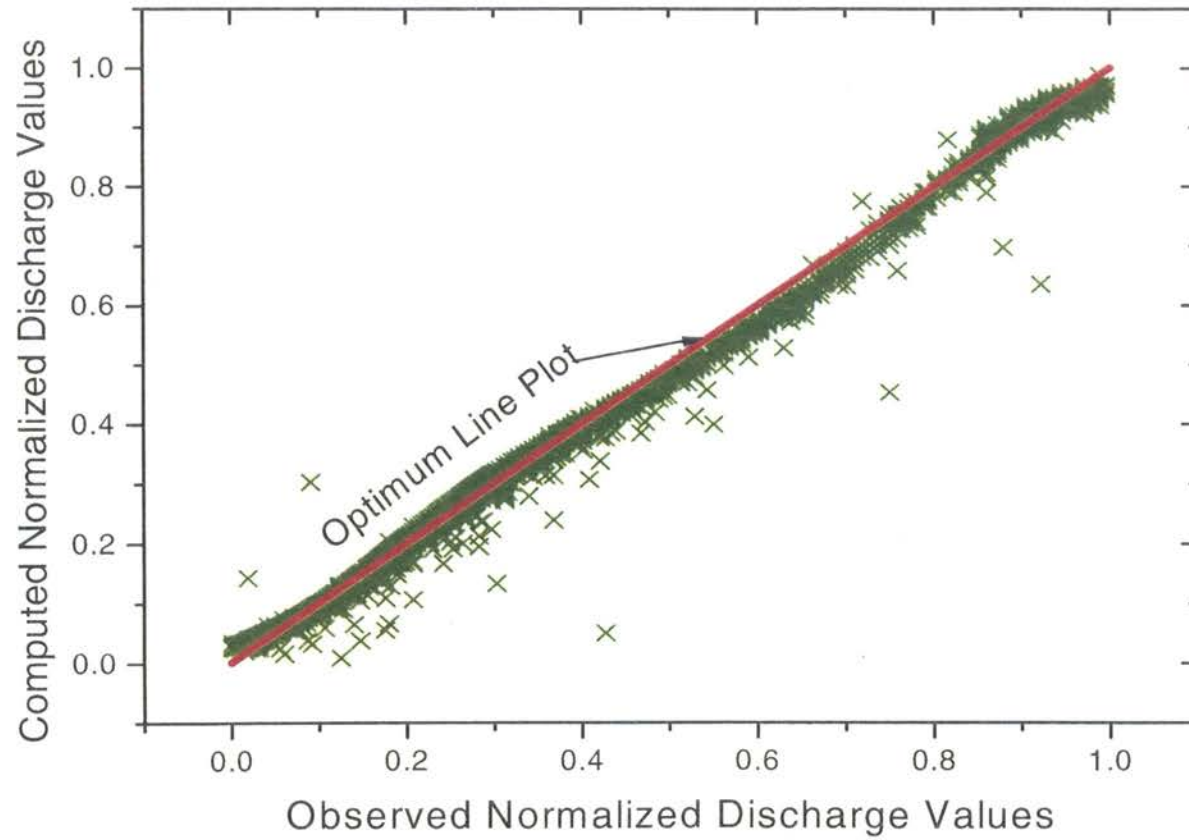
Figure 4.16. Discharge comparison between observed values and computed values.
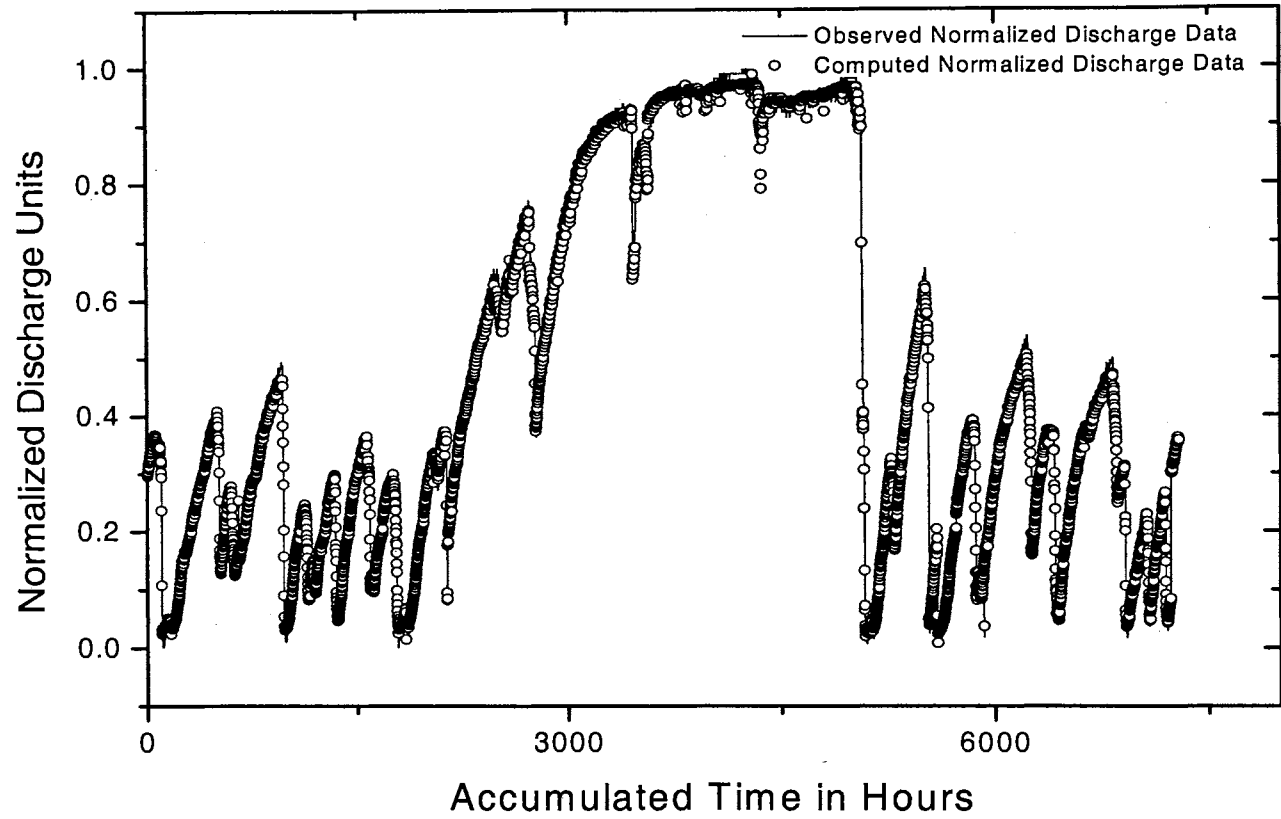
Figure 4.17. Comparison of observed and computed discharge values vs. incremental time in hours.

data were outside the range of the RMSE. The computed data was converted back to real discharge values of cubic feet per second to compare the results with the real observed data. Figure 4.18 shows the real computed discharge data vs. the observed discharge data. As seen, the results were very good. Figure 4.18 shows the computed and the observed data for the entire year of 1998. There were four peaks that the computed data did not match those of the observed data. This might be due to the lack of training data available for these conditions that produced these higher peaks. A closer view of a portion of the same graph reveals that the computed vs. observed discharge data are very good. Figure 4.19 shows graphically a portion of the results of the analysis. The results do not exactly match the observed data in some circumstances, but the error is minimal and this data can still be used for forecasting the discharge one hour ahead.

RESULTS OF LONGER FORECASTING TIMES

The three, six, and nine hour forecasting ANN models included three-hours of previous hourly discharge data for the input data set. This was done so that the network may learn the initial conditions of the basin based on the current river discharge conditions. From an analytical perspective, the lower the previous flow values in the river the more porous the soil in the basin becomes. Since the discharge of the Glover Basin is not controlled by any structure, it was determined that a relationship between the river discharge and the potential for rainfall losses could be established by the network. Thus, the lower the discharge rate at the mouth of the basin, the higher the losses or abstraction. The lower discharge rate, the dryer the basin's condition is becoming.
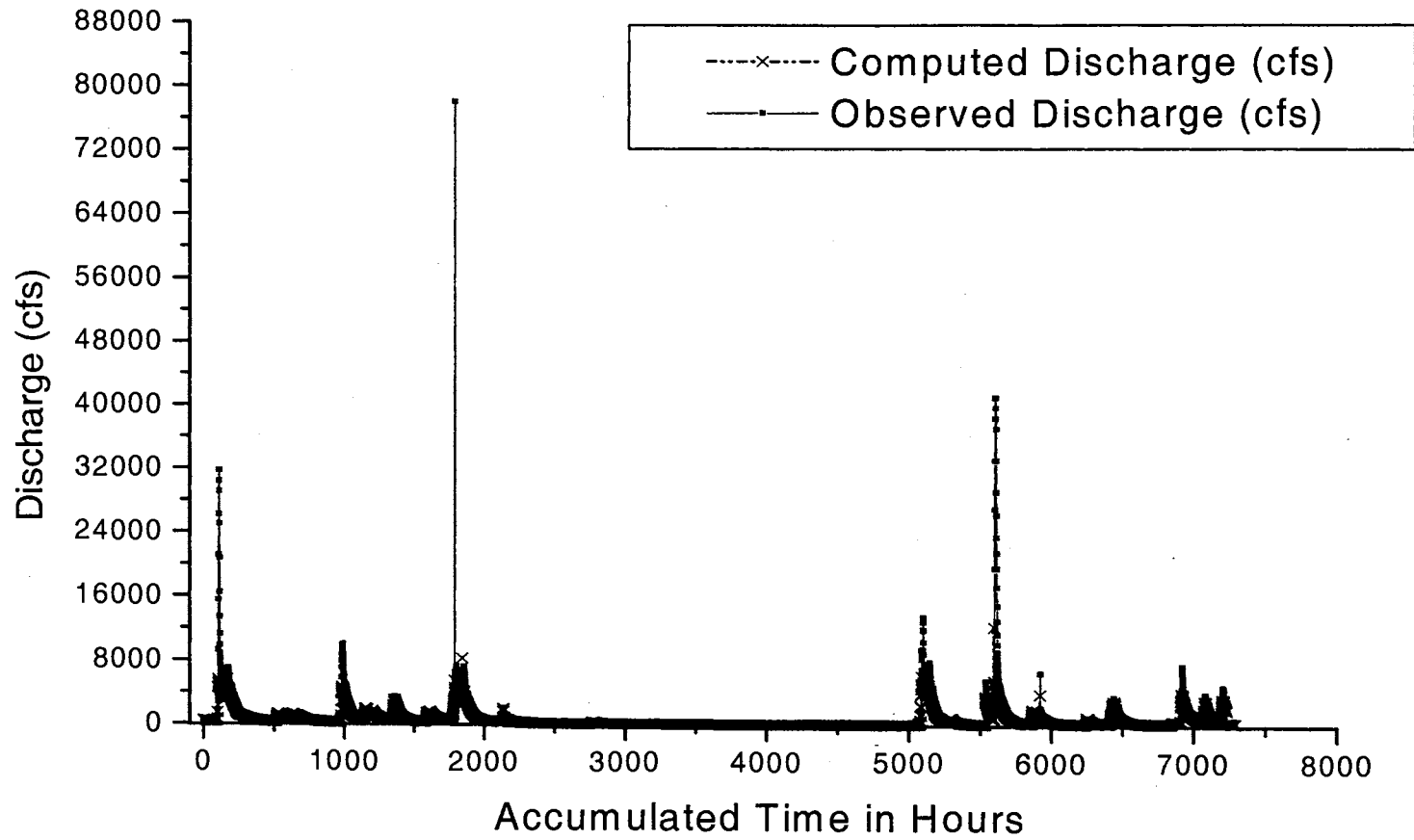
Figure 4.18. Comparison of the computed and observe discharge values after converting them from normalized units to cfs.
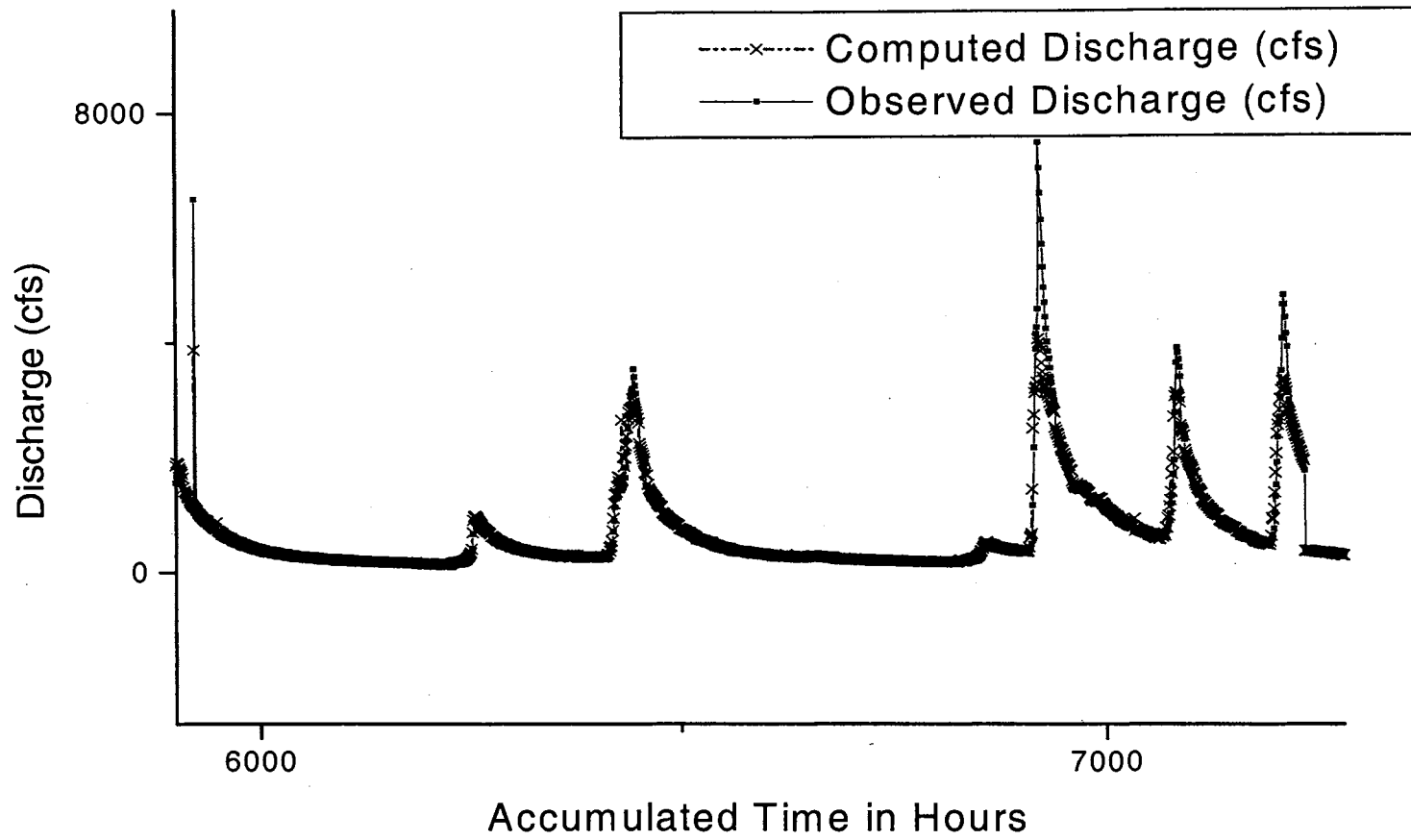
Figure 4.19. Closer view of the computed discharge data vs the observed discharge data.

Likewise, the higher the discharge rate, the lower the loss rate. Therefore, it was

determined that three hours of previous discharge data would be very beneficial for the

network to determine a forecast greater than one-hour. The results as shown in Table 4.2,

show that the RMSE increases with further incremental time forecasting, but the model

adequacy is still very good in the

Table 4.2. Three Hour, Six hour, and Nine Hour Error Results.

Error Parameters	Three Hour	Six Hour	Nine Hour
RMSE	0.0646	0.1894	0.3069
MSE	0.01547	0.07239	0.52491
$N_{RMSE, PERCENT}$	19.88	36.38	30.41
$R^2_{, PERCENT}$	98.81	99.83	99.91

range greater than 90%. A plot of each of the model results shows increasing scatter

from the three hour forecast to the nine hour forecast. Figures 4.20 thru 4.22 show that

the model's ability to predict the runoff gets increasingly poorer the further in time the

forecast is needed. The three-hour forecast is still a good forecast with a very low

RMSE value. The six and nine hour forecast diminished even further. Figure 4.20 is a

scatter plot of the three hour normalized discharge data vs. the normalized observed data.

The computed data for the three-hour model indicates that the network is able to predict

the discharge near the optimum line which is good. Figure 4.21 shows that the network is

not able to predict the discharge data as well as the one-hour or the three-hour models.

Figure 4.22 shows a decrease in network performance as expected since the RMSE value

is much greater than the other three models. The scatter for the nine-hour model starts at

a far greater error interval from the optimum line. By further comparing the data, as
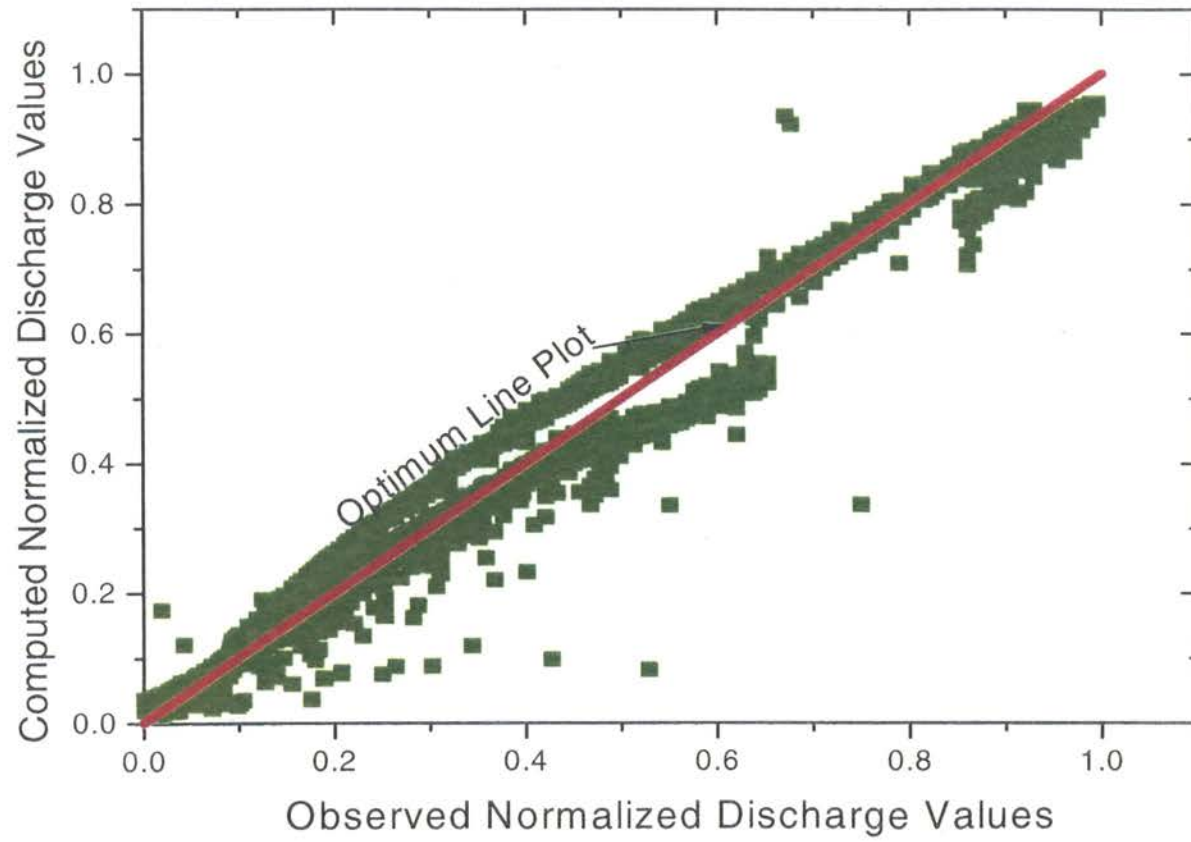
Figure 4.20. Three hour observed vs. computed normalized plot.
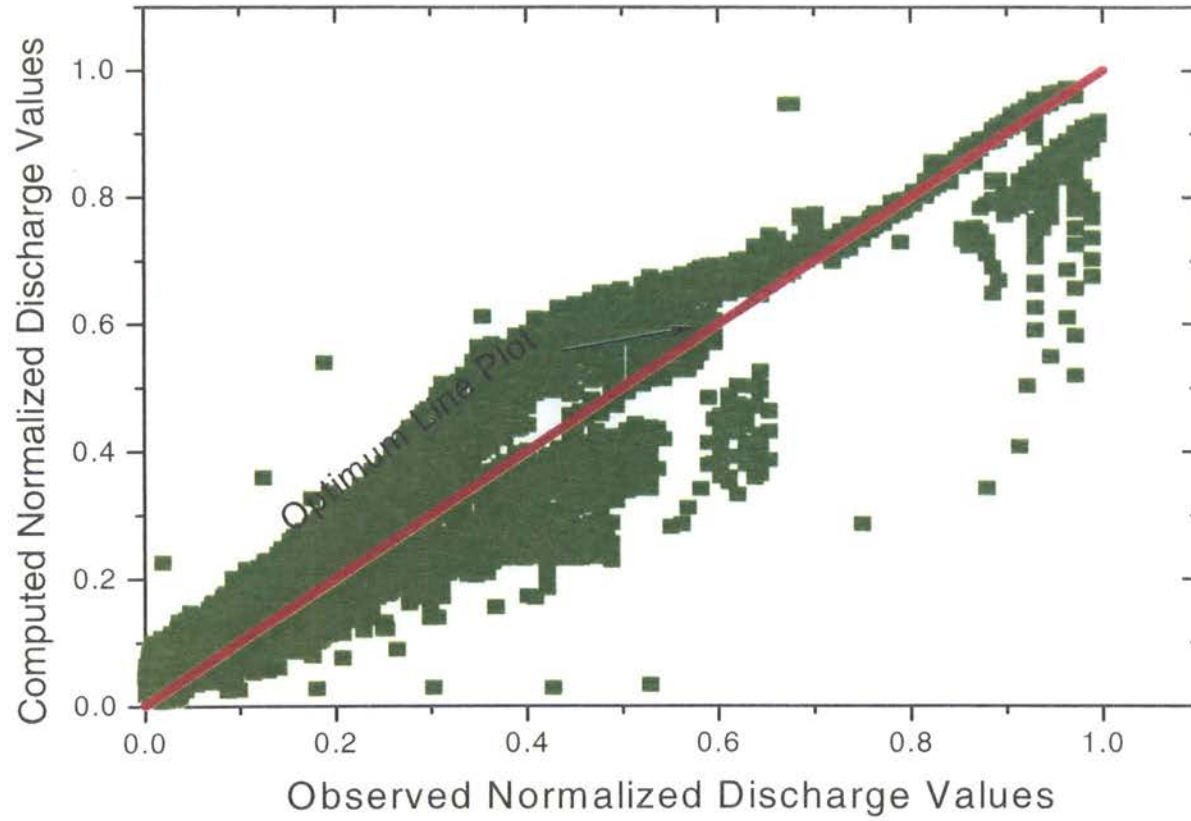
Figure 4.21. Six hour observed vs. computed normalized plot.
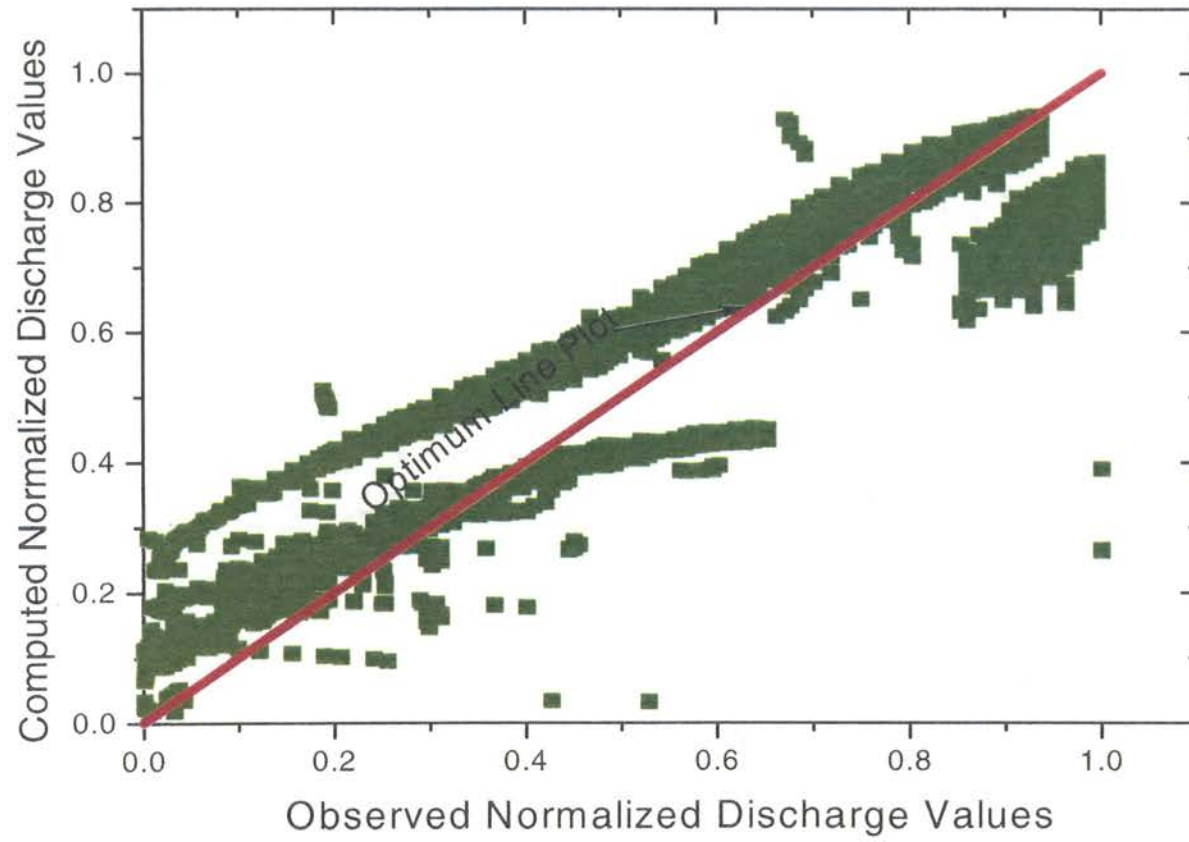
Figure 4.22. Nine hour observed vs. computed normalized plot.
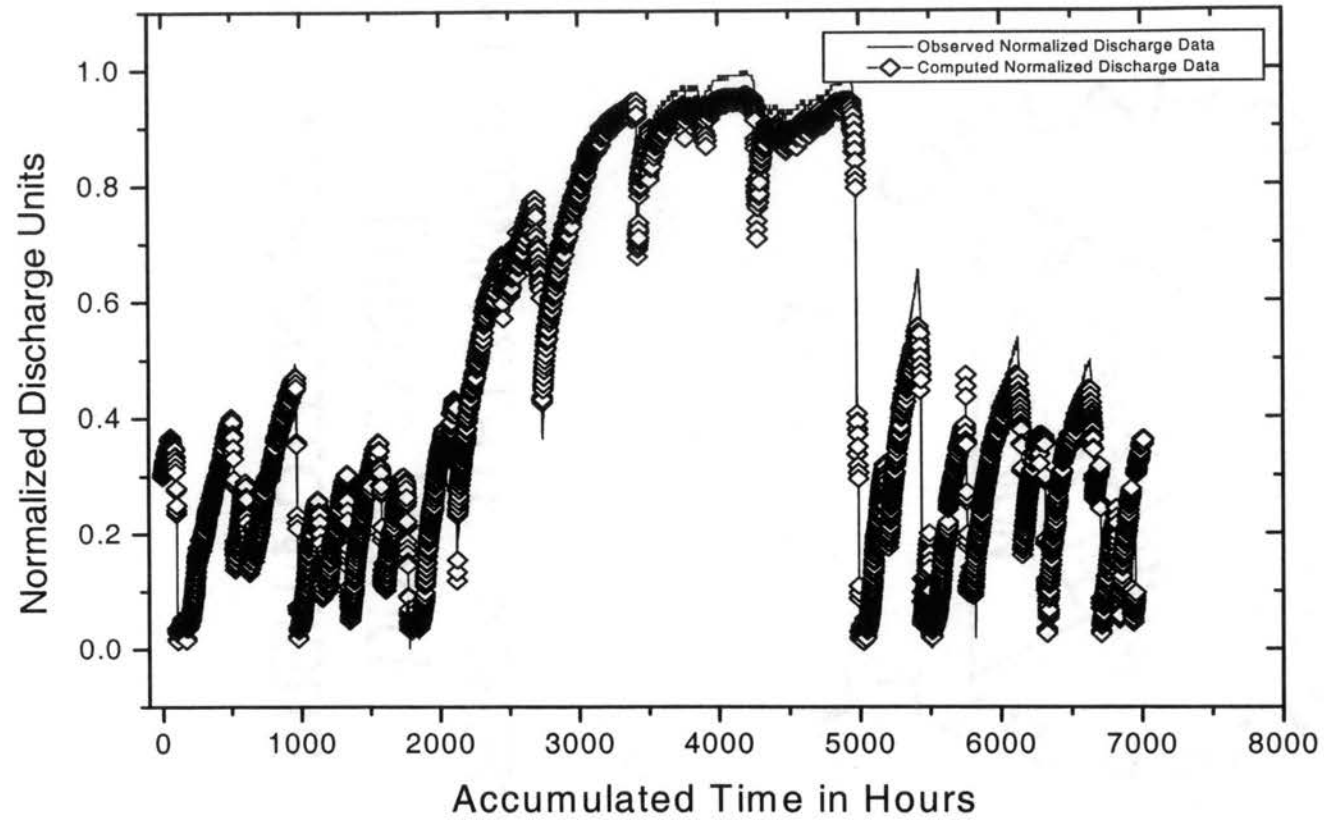
Figure 4.23. Three hour plot of the observed normalized discharge and computed normalized discharge vs. time.
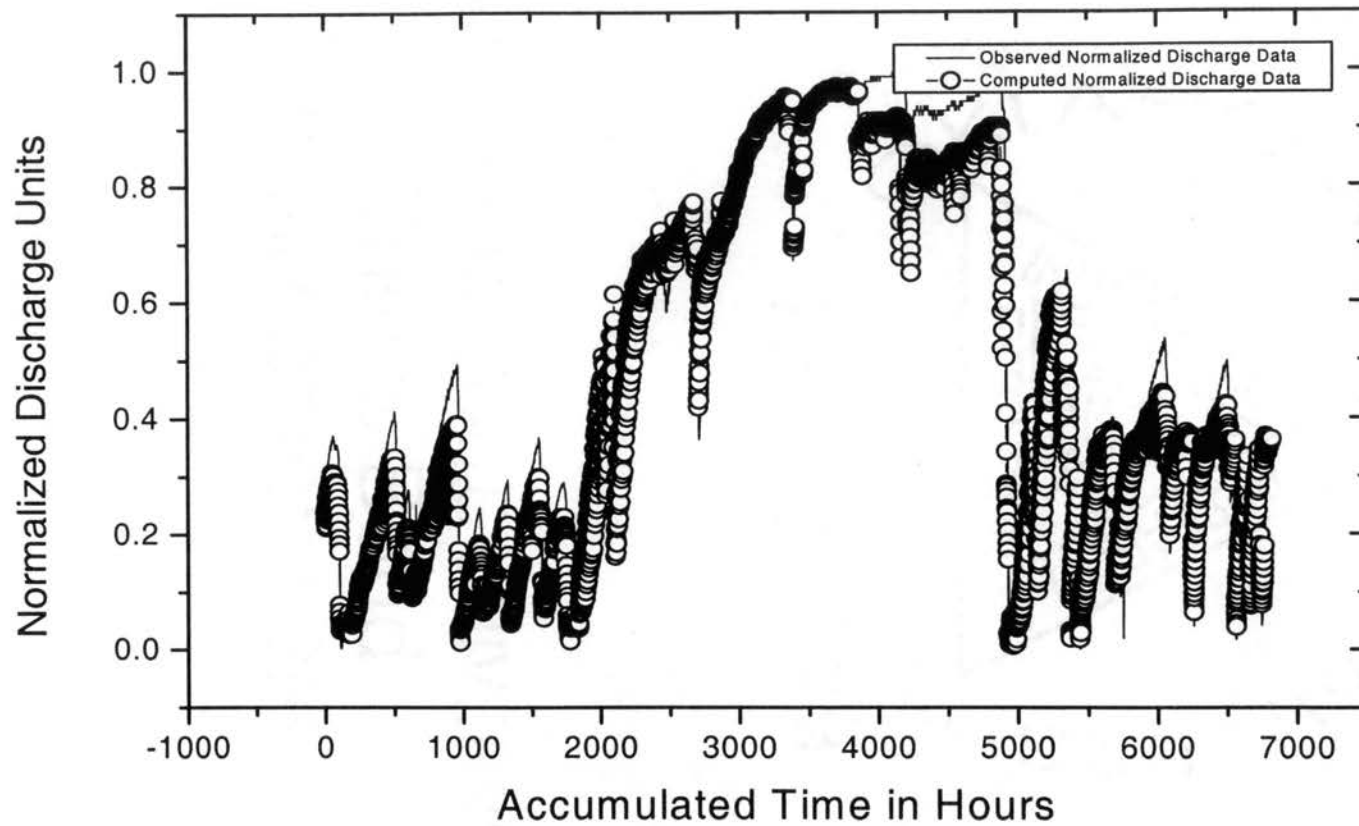
Figure 4.24. Six hour plot of the observed normalized discharge and computed normalized discharge vs. time.
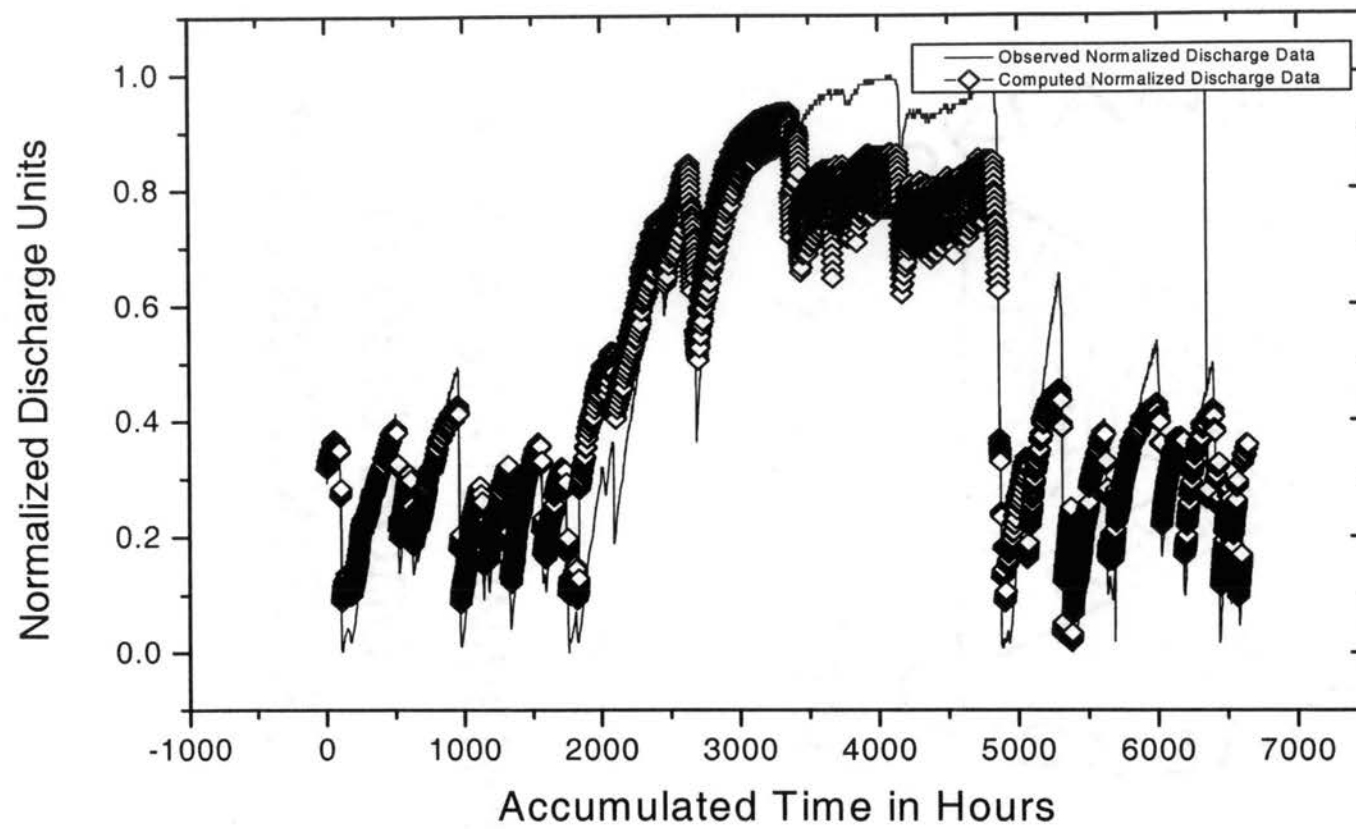
Figure 4.25. Nine hour plot of the observed normalized discharge and computed normalized discharge vs. time.

shown in figures 4.23 thru 4.25, it can be seen that the model's performance is not good the further out in time the prediction is made. Figure 4.23 shows good results for the three-hour forecast, but the six-hour and the nine-hour forecast, as shown in figure 4.24 and figure 4.25, diminishes rapidly. These aforementioned graphs also show that the further out in time a forecast is needed, the poorer the predictability of the network. The main reason is that like many mathematical models it is difficult to determine a basin's saturation level at any one point in time. It is also difficult to determine the basin's response to rainfall until the entire basin is fully saturated. The rainfall-runoff transformation is a non-linear event in which many factors contribute. The results of the one-hour and three-hour models presented show that the network can form a reasonable pattern relationship between the input and output. However, these models still show some difficulty in predicting a rapidly rising hydrograph as shown in Figure 4.18. This may be due to the lack of training data or the lack of input data required by the network. Adding more historical discharge data to the model, rather than just three hours of data, may increase the network's ability to do a better forecast since this one parameter does not fully describe the basin's infiltration rate or other appropriate basin losses.

SUMMARY

The neural network performed very well at predicting the one-hour and the three-hour discharge forecast. The RMSE for these two models were very low which is a good indication the model is able to learn the relationship between the input and the output data provided. The six-hour and the nine-hour network models steadily decreased in performance. A value of 0.189 and 0.3069 for the RMSE for the six-hour and the nine-

113

hour indicate a gradually poor network model. The number of computed data values for these two models showed data that were greater than their perspective RMSE value. The N_{RMSE} increased to over 30% for these two models.

From a formal analytical method of studying this problem, it can be determined that the rainfall data alone are not enough to form a relationship between all possible rainfall losses and the optimal effective rainfall. Even though spatially distributed total rainfall data was provided to the network, it is not enough data for the network to understand all the dynamics that occurs in a basin. The neural network requires data such as hourly infiltration rates, evaporation rates, and temperatures, which are not yet available to better understand all the possible combinations of rainfall-runoff pattern for a basin.

Overall, the neural network performed well in predicting a one-hour and a three-hour forecast with the exception of a rapidly rising hydrograph. There were approximately four peak discharge values that did not match well in the one-hour plot. These four peak values were among the 4% in the one-hour forecast and among the 19.88% in the three-hour forecast that exceeded the RMSE results. These two models can be an aid to the hydrologic engineer in determining a reasonable hydrograph trend that a storm may produce. Determining a forecast of the runoff three hours in advance can help the engineer in determining the appropriate parameter values to use in the HEC-1 model. The engineer can use the discharge data produced by the network as observed data for the HEC-1 model. The network's ease of use makes it a viable tool for engineers to begin using.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

INTRODUCTION

The main focus of this research was to find new or alternative methods to help hydrologic engineers tasked to conduct effective runoff forecasts from storm events in a more effective manner. Hydrologic engineers, working mostly for the U.S. government, have to produce good results of their rainfall forecast hours in advance before the peak discharge caused by a particular storm event reaches the reservoir located downstream of the basin. The time-frame of a good forecast, which is not specific to any basin or storm event, must be well in advance so that a reservoir control engineer can take proper actions of water release to prevent or minimize flooding upstream of the reservoir as well as downstream of the reservoir.

Foreseeing future water discharge caused by a particular storm event can drastically aid engineers in implementing actions to lower the threat of overflowing river waters. A hydrologic engineer working on river forecast is mostly concerned with predicting the outcome of a storm. The engineer has to learn to predict how much effective runoff a storm will produce, understand the ever-changing dynamics of the basin that is being studied, determine the effects of temporal climatologically changes, and understand the effects on a basin due to human intervention. The difficulty associated with forecasting is also ever changing due to more demanding controls on structures for allocating water storage for various other uses.

SUMMARY

Since the U.S. Army Corps of Engineers will continue to use the one-dimensional HEC-1 model for flood forecasting, this research focused on implementing new tools that can be easily used as an aid to the hydrologic engineer conducting a forecast. The HEC-1 model is a good tool for hydrologic engineers due to the ease of use for someone familiar with it. However, this software model has its difficulty in computing discharge levels comparable to the observed discharge levels for various storm events. The model's parameters are not consistent or static with each storm event. Each storm event has its own hydrologic parameters that are constantly changing.

Researches have shown that even with a controlled experiment, the final outcome of a hydrologic study may vary significantly (Smith, Woolhiser, and Simanton, 1994). Infiltration rates have been shown to vary significantly from day to day or even hour to hour. Smith et al. (1994) has shown that discharge values in a controlled experiment can vary as much as 35% due to the dynamic changes of a basin. This illustrates the difficulties of determining appropriate parameter values for the HEC-1 software model when conducting a forecast. It is even more complicated when trying to determine one value for each of the model's parameters when the basin studied is over 300 square miles. Each square mile of the basin responds differently to a rainfall event. Constant losses and initial losses vary throughout the basin. The job of the hydrologic engineer is to determine a single value that represents an average value for the basin as a whole.

This research concentrated on two distinct approaches to aid a hydrologic engineer conducting a forecast of storm events. The first method introduced was the Monte Carlo method and the second method was the Artificial Neural Network method.

Both methods showed good results that can be used as an aid to the forecaster. After the study was completed it was determined that both methods are good tools for the hydrologic engineer. The Monte Carlo methods can give an overall view of all possible peak discharges that a storm event may produce, while the results from the Artificial Neural Network can be used to calibrate the HEC-1 forecast model to pin-point a more precise value of peak discharge for the storm.

The Monte Carlo method required some modifications to the HEC-1 software. The modifications done to the HEC-1 software were to produce the output in a different format that is normally done by the software. Historical data were required for this method. Historical data were used to calibrate the model to determine the range of values that the basin had encountered previously. The model's range of values were obtained by calibrating the HEC-1 model to several storm events that occurred between the years 1997 through 1999. Many storm events from small to large were used for calibrating the model and obtaining the minimum and maximum parameter values. Each of the parameters for the HEC-1 model was randomized between the maximum and minimum historical values using a typical linear distribution. Each parameter was independently randomized meaning that there were no relationships between the parameters. The results of the Monte Carlo method produced a range of peak discharge values for a storm event for all probable average basin parameters. The Monte Carlo method produced results within 30 seconds after the start of the initial run. The results were then plotted on a Log Pearson Type III distribution for visualizing purposes. This gives the hydrologic engineer the maximum probable peak discharge that a storm can produce. Actions can then be implemented to determine if the system as a whole can accommodate the larges

peak discharge computed by the Monte Carlo method. Plan of actions can then be implemented hours in advance if the hydrologic engineer is certain that the storm will produce a large discharge.

The Log Pearson Type III distribution fit the results of the Monte Carlo simulation well as shown in figures 3.2 through 3.5. The four storm events studied in chapter 3 produced discharge values that plotted on the 5% or less probability of exceedence. The hydrologic engineer conducting a forecast can use this method fairly quick and easy. This method gives the engineer a wide range of possible flows for all the various average parameter values for the basin as soon as the precipitation subsides. In the tradition method of conducting a forecast, the engineer has to wait several hours before he can even begin calibrating his model to results of a particular storm event. In a large basin such as the one studied, the effective runoff may take several hour to reach the downstream mouth of the basin. It may take several hours after seeing the initial observed values of the storm before the engineer has enough data to begin his model calibration in an effort to determine the peak flow and the time the peak flow will reach the mouth of the basin. The Monte Carlo method gives the engineer preliminary results of the storm event right after the rainfall stops. This method gives the engineer initial values hours in advance even before the effective runoff begins to reach the mouth of the basin.

The ANN model is new to the scientific and engineering community. It has only begun to spark interest in the last 25 years. The ANN model is easy to use without knowing very much about it. It can be described as a "black box" in which the inner computations of the model are not known by the engineer. The ANN model's major

strength is that it can do non-linear computations fairly well (Zell et al. 1998). It is able to recognize a pattern between the input data and the output data without much effort on the engineer's or scientist's part. The user mainly gives the ANN model a set of input and output data from which the model can learn. The model then determines the appropriate relationship algorithm between the input and output data. The algorithm produces weighted values that can be used to predict the outcome of various other similar inputs.

Chapter 4 described the use of the ANN modeling technique used to conduct a continuous forecast of discharge values for four models. Stage III type radar data were used as the precipitation input for all the models. This data gave a precipitation values on a 100ft by 100ft square grid for the entire basin. Previous hourly discharge values were also entered into the model as input data. The discharge data were given so that the network models could establish relationships between discharge and basin conditions. Both the one-hour and the three-hour network models showed good model adequacy as shown in tables 4.1 and 4.2. A continuous forecast for the entire year of 1998 was done with all available data. The model efficiency for all four models produced results above 90% and the RMSE were small indicating good model performance. However, the six-hour forecast and the nine-hour forecast showed a larger scatter of the results than the other two models indicating that the network could not predict a good pattern for these two models.

RECOMMENDATIONS

This research concentrated on developing several methods of forecasting which are easy and quick to use without much added burden to the hydrologic engineer. The engineer is still required to have a strong background in hydrology and hydraulics to implement these methods. The following are recommendations for future research for this topic:

1. The HEC-1 model still uses lumped sum rainfall data for forecasting purposes for the entire basin. A model that can use spatially distributed radar data would produce better results. Even though the time-of-concentration curve was developed for the basin, this curve still limits the model's ability to produce good results.

2. The forecast models done for storm events should be archived for further future analysis. The models should be archived based on season, due to the different basin conditions. The maximum and minimum values for each season should be used in a Monte Carlo method rather than the maximum and minimum values for all storm events. This would narrow the range of values used for computing the discharge data.

3. All gage data should be archived on a database. At present time only rainfall precipitation and discharge data are archived for each gage. Hourly temperature, hourly evaporation, hour wind velocity and vector data, and hourly solar radiation data are destroyed after one month to conserve disk space.

4. Both the Monte Carlo method as well as the ANN method should be used in conjunction as an aid to the hydrologic engineer. The results of the Monte Carlo

method should be used as an overall view of all possible peak flow discharges that the storm event can produce. An engineer familiar to the basin can also narrow the range for the random variable to obtain a smaller range of possible outcomes from the storm event. The ANN model should than be used to begin calibrating the HEC-1 model. By using both these methods, an engineer can better predict the effective runoff produced by a storm event well in advance so that actions can be taken to prevent or minimize flooding.

REFERENCES

1. Beard, Leo R., "Statistical Methods in Hydrology," U.S. Army Corps of Engineers 1962.

2. Bentley Systems, Incorporated., "MicroStation Software Version 05.07.01.14," Bentley 1998.

3. Beven, K.J., "Distributed models in Hydrological Forecasting," edited by M.G. Anderson and T.P. Burt, pp. 405-435, John Wiley, New York, 1985.

4. Blue Marble Geographics, "Geographic Translator," Software Package, 2000.

5. Cajal, S. R, "English Translation: Histology of the Nervous Systeom of Man and Vertebrates," (translated by N. Swanson and L.W. Swanson), New York: Oxfor University Press, 1995.

6. Campolo, Marina, Soldati, Alfredo, and Andreussi, Paolo, "Forecasting River Flow Rate During Low-Flow Periods Using Neural Networks," American Geophysical Union, 1999.

7. Corporation for National Research Initiatives, "Python Software Version 2.1," CNRI, 2000.

8. Cowan, J.D. , "Neural Networks: The Early Days," NIPS, 2:828-842, 1990.

9. Dearden, R, Friedman, N., and Russell, S., "Speech Recognition with Dynamic Bayesian Networks," In Proc. AAAI-98, Madison Wisconsin: AAAI Press 1998.

10. Elmans, Jeff, "Generalization, Simple Recurrent Networks, and the Emergence of Structure," Proceeding of the 20th Annual Conference of the Cognitive Science Society, 1998.

11. Environmental Systems Research Institute Inc., (ESRI), "Modeling Our World," ESRI Press 1999.

12. EnviroSence, "Executive Order 12873 – Federal Acquisition, Recycling, and Waste Prevention", EnviroSence, 1996.

13. Haykin, Simon, "Neural Networks," Macmillan College Publishing Company, Inc. 1994.

14. Hebb, D., "The Organization of Behavior," John Wiley, New York, 1949.

15. Hellweger, Ferdi, "BALANCE-Map Based Surface Water Quality Model," Thesis paper 1997.

16. Honavar, Vasant, "Artificial Intelligence and Neural Networks: Steps Toward Principled Integration," Academic Press, Inc. 1994.

17. Hsu, Kuo-lin, Gupta, Hoshin, Vijai, and Sorooshian, Soroshh, "Artificial Neural Network Modeling of the Rainfall-Runoff Process," Water Resources Research, Vol. 31, No. 10, Pages 2517-2530, October 1995.

18. Hubel, D. H. and Viesel, T. N., "Receptive Fields, Binocular Interaction and Functional Architecture in a Cat's Visual Cortex," Journal of Physiology, 160:106-154, 1962.

19. Jones, N. L., Wright, S.G., and Maidment, D. R., "Watershed Delieation with Triangle-Based Terrain Models," Journal of Hydraulic Engineering, 116(10), 1232-1251, 1990.

20. Maidment, David, "Delineation of Watersheds and Streams," University of Texas at Austin, Spring 1995.

21.   Martin, John C., "Introduction to Languages and the Theory of Computation," McGraw-Hill, Inc. 1991.

22.   McTernan, William F., Daniels, Barry T., Garner, Tina V., "A Preliminary Approach to Determining the Probability of Non-Point Source Contamination of Shallow Aquifers by Agricultural Chemicals," Proceeding from 'Risk Assessment for Groundwater Pollution Control,' ASCE 1990.

23.   Mehrotra, Kishan, Mohan, Chilukuri, Ranka, Sanjay, "Elements of Artificial Neural Networks," The Massachusetts Institute of Technology Press, 2000.

24.   Microcal Software Inc., "Microcal Origin Software Version 5.0." Microcal Sofware 1997.

25.   Minsky, M. L. and S. A. Papert, "Perceptrons," MIT Press, Cambridge, 1969.

26.   Neil, Fraser, "Introduction to Neural Networks," Albyn House, Scotland, IV September 1998.

27.   Nombas Inc., "Cenvic32 Cmm Software," Nombas 1996.

28.   Olivera, Francisco and Maidment, David, "Runoff computation Using Spatially Distributed Terrain Parameters," Proceeding of ASCE-North American Water and Environment Congress 1996, Aneheim, California, June 22-28, 1996.

29.   Oliver, Francisco, and Midment, David, "CRWR-PrePro: An ArcView Preprocessor for Hydrologic, Hydraulic and Environmental Modeling," Center for Research in Water Resources, University of Texas at Austin 1999.

30.   Olivera, Francisco, and Midment David, "GIS Tools for HMS Modeling Support," ESRI 1999.

31. Ponce, Victor M., "Engineering Hydrology Principles and Practices," Hall, Inc., 1989.

32. Reed, Seann M., and Maidment David, "Map-Based Surface Water Simulation," Center for Research in Water Resources, University of Texas at Austin, November 1996.

33. Rosenblatt, F., "Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms," Spartan, New York, 1961.

34. Simon, Haykin, "Neural Networks," MacMillan College Publishing Company, Inc. 1994.

35. Smith, R. E., Goodrich, D. R., Woolhiser, D. A., and Simanton, J.R., "Physically Based Hydrologic Modeling, 2, Is the Concept Realistic?" Water Resources Research, Vol. 30, Pages 851-854, 1994.

36. Stergiou, Christos, and Siganos, Dimitrios, "Neural Networks," http://www.doc.ic.ac.uk/nd/surprice_96/vol4/cs11/reporth.html, 1999.

37. Taylor, W.K., "Electrical Simulation of Some Nervous System Functional Activities," Information Science, Pages 314-328, 1956.

38. Tiemann, Mary, "Safe Drinking Water Act: Overseeing Implementation," The National Council For Science and Environment, 1997.

39. Turing, A.M., "On Computable Numbers with an Application to the Entscheidunsproblem," Proceeding of London Mathematical Society, 2, No. 42, 1936.

40. U.S. Army Corps of Engineers, "Hydrologic Analysis of Ungaged Watersheds Using HEC-1," U.S. Army Corps of Engineers, 1982.

41.   U.S. Army Corps of Engineers, "A History of the USACE Civil Works Program," U.S. Army Corps of Engineers, 1996.

42.   U.S. Army Corps of Engineers, "U.S. Army Corps of Engineers Annual Flood Damage Report to Congress for Fiscal Year 1995," U.S. Army Corps of Engineers, 1996.

43.   U.S. Army Corps of Engineers, "Hydrologic Frequency Analysis," Hydrologic Engineering Center- Corps of Engineers, 1975.

44.   U.S. Army Corps of Engineers, "HEC-1 Flood Hydrograph Package, Users Manual," U.S. Army Corps of Engineers, 1969.

45.   U.S. Army Corps of Engineers, "Hydrologic Engineering in Planning," U.S. Army Corps of Engineers, 1981.

46.   U.S. Army Corps of Engineers," Statistical Methods in Hydrology," U.S. Army Corps of Engineers, 1965.

47.   U.S. Army Corps of Engineers, "The Financial Cost of Water Resources Development: An Overview," U.S. Army Corps of Engineers, 1955.

48.   U.S. Army Corps of Engineers, "Water Control Management, Engineering Regulation 1110-2-240," U.S. Army Corps of Engineers, 1955.

49.   United States Census Bureau, "County Population Estimates for July 1, 1999 and Population Change: July 1, 1998 to July1, 1999,: United States Census Bureau, 1999.

50.   Unites States Department of Agriculture, "State Soil Geographic (STATSGO) Data Base," United States Department of Agriculture, 1994.

51.   United States Environmental Protection Agency, "Better Assessment Science
        Integrating point and Nonpoint Sources, Version 3," United States
        Environmental Protection Agency, 2000.

52.   U.S. Department of the Interior Geological Survey, "Guidelines for
        Determining Flood Flow Frequency," U.S. Department of the Interior
        Geological Survey, 1982.

53.   Vemuri V. M, and Rogers, R. D. "Artificial Neural networks Forecasting
        Time Series," IEEE Computer, Soc. Press, Los Alimos, Calif., 1994.

54.   Widrow, B, "Adaptive Sample-data Systems – A Statistical Theory of
        Adaptation," WESCON Convention Record: Part 4, Pages 74-85, 1959.

55.   Woolhiser, D.A., "Search for Physically Based Runoff Model – A Hdryologic
        El Dorado?," Journal of Hydrologic Engineering, Pages 122-129, 1996.

56.   Zell, Andreas, Mamier, Gunter, Vogt, Michael, Mache, Neils, Hubner, Ralf,
        Doring, Sven, Herrmann, Kai-Uwe, Soyez, Tobias, Schmalzl, Michael,
        Sommer, Tilman, Hatzigeorgiou, Artemis, Posselt, Dietmar, Schreiner,
        Tobias, Kett, Bernward, Clemente, Gianfranco, Wieland, Jens, Gatter, Jurgen,
        " Stuttgart Neural Network Simulator," University of Stuttgart Press, 1998.

APPENDIX

APPENDIX A

Tables A-1 through A-.10 attached in this appendix and stored in the 3.5" floppy

disks are the tables used for calculating the time of concentration for Chapter 2. The

tables were created through Arc/Info and Arc/View. Most of the columns such as

SLOPE_ID, FNODE, TNODE, LPOLY1, LPOLY2, STREAM1_CO, AND STR_NO1

are used in conjunction with the software for visual representation. Columns

GRIDCODE, AND LENTGH2 were queried with an Arc/View script for filling the last

column TC_Hours. The values under column GRIDCODE are the slope values in

percent for each stream length. Due to the software limitations, the slope grid and the

stream shape file did not match exactly. The slope grid was broken into smaller sections.

The script looked at the STR_NO1 and the SREAM1_CO to match all the slope values

with the stream value. An average of the slope values was computed for each stream

length. This gave all the required values to calculate TC. Below is a sample of the

computations the script made for calculating Tc. As can be seen from this small

computation, that this would be a tedious task if it were done by hand. With the script

written for Arc/View it took a matter of minutes to compute all the Tc for each stream

length.

Example Calculation of Tc from Table 2. Using ID number 1 – 6 as a sample.

ID	GRIDCODE	LENGTH2	STREAM1_CO
1	3	376.71	2915
2	5	376.71	2915
3	1	376.71	2915
4	3	376.71	2915
5	2	376.71	2915
6	0	376.71	2915

Average slope   2.33% or 0.02333

Using Haraway's formula Equation 2-2 Tc can be calculated.

Tc = (0.606)*(3761.71*0.06)^(0.467)/(0.0233)^0.234

Tc = 18.34 minutes which is equal to 0.305 hours.

The previous Tc in table 2.1 was 28.83. Summing the new Tc for the first part of the table with 28.83 gives 28.93 hours which is the value on table 2.1.

APPENDIX B

The attached pages in this appendix shows the code of the CMM script used to extract the data from the Doppler files specifically for the Glover Basin.

gz.cmm   Script used for extracting the compressed NexRad government zipped files to uncompressed format.

ncg.cmm   Script used to convert the uncompressed NexRad Stage II files to ASCII format.

wtav2.cmm   Script used to extract precipitation data from the ASCII file specifically for the Glover Basin.

MonteCarlo.py Python script code to run the Monte Carlo Simulations used in Chapter 3.

```
//---------------------------gz.cmm---------------------------------
//Program to unzip the NexRad NetCDF files from a government zipped
format.  It unzips a month worth of hourly files
#include <cgi.lib>
#include <findfile.lib>
#include <time_ext.lib>

g_outfile_name = tmpnam();

g_outfile = NULL;

g_images_dir = "d:\\horatios2\\apr97";

g_referer = NULL;

g_chmon = {"Jan", "Feb", "Mar", "Apr", "May", "Jun",
           "Jul", "Aug", "Sep", "Oct", "Nov", "Dec" };




//====================================================================

void main(argc, argv)
{

    getfiles(num_dates);




}
//====================================================================
getfiles(num_dates)
{

    strcpy(fileout,g_images_dir);
    strcat(fileout,"\\gz.bat");

    filein=fopen(fileout,"w");
    //date1[0]="";
    selection = 0;
    num_dates=0;
    sprintf(filemask, "%s\\*gz", g_images_dir);
    writetime = 0;
    handle = findfirst(filemask, fileinfo)
    if (handle != -1)
    {
        do
        {
            if (strlen(fileinfo.name) != 15) continue;
            {
                if (fileinfo.write> writetime)writetime =
fileinfo.write;
                printf("%s\n", fileinfo.name);
                fprintf(filein,"gzip.exe -d %s\n",fileinfo.name);
                i = num_dates++;
                //strncpy(date1[i], fileinfo.name, 15);
```

132

```
                    //if (date1[0] && !strcmp(date1[0], date1[i]))selection
= i;

            }
        } while (!findnext(handle, fileinfo));

        // findclose(handle);
        fclose(filein);

    }
}
//====================================================================
```

```
//------------------------ncg.cmm---------------------------------
// Program to convert the NexRad data stage III files from NetCDF
format to ASCII format.
// It converts all 720 or more files from a monthly directy into
separate 720 ASCII foremat files.
#include <cgi.lib>
#include <findfile.lib>
#include <time_ext.lib>

g_outfile_name = tmpnam();

g_outfile = NULL;

g_images_dir = "c:\\horatios\\jan99";

g_referer = NULL;

g_chmon = {"Jan", "Feb", "Mar", "Apr", "May", "Jun",
            "Jul", "Aug", "Sep", "Oct", "Nov", "Dec" };



//======================================================================

void main(argc, argv)
{

    getfiles(num_dates);




}
//======================================================================
getfiles(num_dates)
{

    strcpy(fileout,g_images_dir);
    strcat(fileout,"\\nc.bat");
    filein=fopen(fileout,"w");
    //date1[0]="";
    selection = 0;
    num_dates=0;
    sprintf(filemask, "%s\\*nc", g_images_dir);
    writetime = 0;
    handle = findfirst(filemask, fileinfo)
    if (handle != -1)
    {
        do
        {
            if (strlen(fileinfo.name) != 12) continue;
            {
                if (fileinfo.write> writetime)writetime =
fileinfo.write;
                strncpy(newfile, fileinfo.name,9);
                strcat(newfile, ".coe");
                strncpy(newnc, fileinfo.name, 9);
```

```
                strcat(newnc,".txt");
                printf("%s\n", fileinfo.name);
                printf("%s\n", newfile);
                fprintf(filein,"s3rfc2coe.exe -i %s -f 1 -o %s -
c\n",fileinfo.name, newfile);
                fprintf(filein,"ncgetprecip.exe -i %s -f 1 -o %s -c1
628,296 -c2 635,307\n", newfile, newnc);
                i = num_dates++;
                //strncpy(date1[i], fileinfo.name, 15);

                //if (date1[0] && !strcmp(date1[0], date1[i]))selection
= i;


            }
        } while (!findnext(handle, fileinfo));

        // findclose(handle);
        fclose(filein);


    }
}
//=====================================================================
```

```
//-----------------------------wtav2.cmm----------------------
//cmm script for extracting the precip data for the glover basin from
720 hourly files per month.
#include <cgi.lib>
#include <findfile.lib>
#include <time_ext.lib>

g_outfile_name = tmpnam();

g_outfile = NULL;

g_images_dir = "c:\\horatios\\jan99";
g_outfile = "c:\\horatios\\output";

g_referer = NULL;

g_chmon = {"Jan", "Feb", "Mar", "Apr", "May", "Jun",
           "Jul", "Aug", "Sep", "Oct", "Nov", "Dec" };

struct b
{
     int x, y;
     float area;
}

struct c
{
        int x,y;
        float precip, pa, summ;
}


//======================================================================

void main(argc, argv)
{

     getfiles(num_dates);




}
//======================================================================
getfiles(num_dates)
{
     b[0].x = 340453;
     b[0].y = 945340;
     b[0].area = 7.0355;
     b[1].x = 340722;
     b[1].y = 945546;
     b[1].area = 7.6767;
     b[2].x = 340660;
     b[2].y = 945312;
     b[2].area = 6.1251;
     b[3].x = 340637;
     b[3].y = 945039;
```

```
b[3].area = 4.1236;
b[4].x = 340952;
b[4].y = 945752;
b[4].area = 4.8811;
b[5].x = 340929;
b[5].y = 945518;
b[5].area = 6.1300;
b[6].x = 340907;
b[6].y = 945245;
b[6].area = 7.4232;
b[7].x = 341159;
b[7].y = 945725;
b[7].area = 5.0525;
b[8].x = 341136;
b[8].y = 945451;
b[8].area = 6.1340;
b[9].x = 341114;
b[9].y = 945217;
b[9].area = 6.1330;
b[10].x = 341051;
b[10].y = 944944;
b[10].area = 4.3622;
b[11].x = 341406;
b[11].y = 945657;
b[11].area = 5.6474;
b[12].x = 341343;
b[12].y = 945424;
b[12].area = 6.1380;
b[13].x = 341321;
b[13].y = 945150;
b[13].area = 6.1370;
b[14].x = 341258;
b[14].y = 944916;
b[14].area = 6.1362;
b[15].x = 341235;
b[15].y = 944643;
b[15].area = 3.1288;
b[16].x = 341613;
b[16].y = 945630;
b[16].area = 7.1222;
b[17].x = 341550;
b[17].y = 945356;
b[17].area = 6.1420;
b[18].x = 341528;
b[18].y = 945123;
b[18].area = 6.1411;
b[19].x = 341505;
b[19].y = 944849;
b[19].area = 6.1401;
b[20].x = 341442;
b[20].y = 944615;
b[20].area = 3.4219;
b[21].x = 341843;
b[21].y = 945837;
b[21].area = 2.9691;
b[22].x = 341820;
b[22].y = 945603;
```

```
b[22].area = 6.1470;
b[23].x = 341758;
b[23].y = 945329;
b[23].area = 6.1460;
b[24].x = 341735;
b[24].y = 945055;
b[24].area = 6.1451;
b[25].x = 341712;
b[25].y = 944821;
b[25].area = 8.3624;
b[26].x = 342050;
b[26].y = 945810;
b[26].area = 4.7251;
b[27].x = 342027;
b[27].y = 945536;
b[27].area = 6.1510;
b[28].x = 342005;
b[28].y = 945301;
b[28].area = 6.1501;
b[29].x = 341942;
b[29].y = 945027;
b[29].area = 6.1491;
b[30].x = 341919;
b[30].y = 944753;
b[30].area = 6.9558;
b[31].x = 342342;
b[31].y = 950251;
b[31].area = 9.4847;
b[32].x = 342320;
b[32].y = 950017;
b[32].area = 8.4683;
b[33].X = 342257;
b[33].y = 945742;
b[33].area = 6.1559;
b[34].x = 342235;
b[34].y = 945508;
b[34].area = 6.1549;
b[35].x = 342212;
b[35].y = 945234;
b[35].area = 6.1541;
b[36].x = 342149;
b[36].y = 944960;
b[36].area = 6.1531;
b[37].x = 342126;
b[37].y = 944726;
b[37].area = 4.2150;
b[38].x = 342550;
b[38].y = 950224;
b[38].area = 2.9956;
b[39].x = 342527;
b[39].y = 945949;
b[39].area = 6.1608;
b[40].x = 342505;
b[40].y = 945715;
b[40].area = 6.1599;
b[41].x = 342442;
b[41].y = 945441;
```

```
b[41].area = 6.1591;
b[42].x = 342419;
b[42].y = 945206;
b[42].area = 6.1580;
b[43].x = 342357;
b[43].y = 944932;
b[43].area = 6.1571;
b[44].x = 342334;
b[44].y = 944658;
b[44].area = 3.3581;
b[45].x = 342735;
b[45].y = 945922;
b[45].area = 6.9875;
b[46].x = 342712;
b[46].y = 945648;
b[46].area = 6.1639;
b[47].X = 342649;
b[47].y = 945413;
b[47].area = 6.1630;
b[48].x = 342627;
b[48].y = 945139;
b[48].area = 6.1622;
b[49].x = 342604;
b[49].y = 944904;
b[49].area = 6.1612;
b[50].x = 342541;
b[50].y = 944630;
b[50].area = 4.2638;
b[51].x = 342919;
b[51].y = 945620;
b[51].area = 5.3477;
b[52].x = 342857;
b[52].y = 945346;
b[52].area = 4.5712;
b[53].x = 342834;
b[53].y = 945111;
b[53].area = 6.1809;
b[54].x = 342811;
b[54].y = 944837;
b[54].area = 4.0138;


strcpy(fileout,g_images_dir);
strcat(fileout,"\\tst.bat");
strcpy(wtave, g_outfile);
strcat(wtave,"\\wave00.txt");


filein=fopen(fileout,"w");
//date1[0]="";
selection = 0;
num_dates=0;
accwav = 0;
sprintf(filemask, "%s\\*txt", g_images_dir);
writetime = 0;
handle = findfirst(filemask, fileinfo)
if (handle != -1)
```

```c
    {
        do
        {
            if (strlen(fileinfo.name) != 12) continue;
            {
                if (fileinfo.write> writetime)writetime =
fileinfo.write;
                printf("%s\n", fileinfo.name);
                txt_file = fopen(fileinfo.name,"r");
                fgets(line,280, txt_file);
                fgets(line, 280, txt_file);
                fgets(line, 280, txt_file);
                fgets(line, 280, txt_file);
                fgets(line, 280, txt_file);
                summarea = 0;
                precipsum = 0;

                while (fgets(line, 280, txt_file) != NULL)
                {
                    sscanf(line, "%d, %d %d %d %f", id1, id2, x1,
y1, pre);

                    //printf ("%d, %d %d %d %f\n",id1, id2, x1, y1,
pre);

                    for (j = 0; j < 55; j++)
                    {
                        //printf ("%d, %d %d %d %f %d\n",id1,
id2, x1, y1, pre, j);

                        if ((b[j].x == x1) && (y1 == b[j].y))
                        {
                            printf("%d %d %d %d %f\n
j=%d\n", b[j].x, b[j].y, x1, y1, pre,j);
                            summarea = summarea +
b[j].area;

                            precipsum = b[j].area * pre;


                        }
                    }

                }
                opwave = fopen(wtave, "a+");
                wtav = precipsum/summarea;
                accwav = accwav + wtav;
                printf ("%f is the weighted aver", wtav);
                fprintf (opwave, "%4.2f\n", accwav);
                fclose(opwave);
                printf ("this is the end");
                fclose(txt_file);
                //for (k=0; k<2; k++)printf("%d\n", b[k].x);
                //fprintf(filein,"gzip.exe -d %s\n",fileinfo.name);
                i = num_dates++;
                //strncpy(date1[i], fileinfo.name, 15);

                //if (date1[0] && !strcmp(date1[0], date1[i]))selection
= i;

        }
```

```
        } while (!findnext(handle, fileinfo));

        // findclose(handle);
        fclose(filein);

    }
}
//=====================================================================
```

```
//---------------------MonteCarlo.py----------------------------------
// Python code to run a Monte Carlo Simulation with the HEC-1 software
import sys, os, random, whrandom
TabularPageTemplate = """ID      Glover Basin HEC-1 Model
ID      Snyder Unit Hydrograph Method with a time-area data. Jan 2001.
ID      Precip Data obtained from Radar and a basin average rainfall
computed.
ID      Basin average rainfall amounts stored incrementally from Jan-Dec
in DSS
IT    60 20FEB97    0200       150
IO    1        2
OU
KK   GLOV GLOVER BASIN
ZR=QO B=GLOV C=FLOW-LOC CUM F=OBS
BA325.25
* BF 500     -.35     1.1
BF    500    -.25     1.10
PB      0
ZR=PI B=PRECIP C=HRAP F=OBS
US%6.4s%8.4s
UA      0    1.22     4.32     5.92     9.63    15.52    23.76    34.29
45    55.35
UA 63.33      73.3    86.31    97.90   106.43   116.96   126.33   136.85
146.90  162.98
UA176.31   184.20   195.99   209.70   219.36   235.45   251.01   267.30
288.38  325.25
* LU .74        -1
LU%6.4s%8.4s
ZW B=GLOV C=FLOW-LOC CUM F=CALC
ZZ
"""
#        Tp                Cp              Strtl           Cnstl
# |-------------|  |-------------|  |-------------|  |------------|
# Min   Ave    Max   Min Ave   Max   Min  Ave   Max   Min Ave    Max
# 9.53 15.77  24.5  0.38 0.69  0.96  0      0.53  1.55  0   0.04   0.11
hec1 = "c:/thesis/hec1.exe i=c:/thesis/f2a.dat o=c:/thesis/junk.out
dss=c:/thesis/glov97al.dss mnt=c:/thesis/out.mnt"
bsn1 = ''
temp = ''
for x in range(1, 1001):
    generator = whrandom.whrandom()
    tp = generator.random()*10+9
    cp = generator.random()+0.3
    st = generator.random()*2
    cn = generator.random()
    TabularPageData = (tp, cp, st, cn)
    f = open("c:/thesis/f2a.dat","w")
    f.write(TabularPageTemplate % TabularPageData)
    f.close()
    os.system(hec1)
    f1 = open("c:/thesis/out.mnt","r")
    bsn1 = bsn1 + f1.readline()
    f1.close()
    f2 = open("c:/thesis/bsn1.mnt","w")
    f2.write(bsn1)
    f2.close()
```

APPENDIX C

The included floppy disk contains the input data used for the training and testing set used for the ANN study of Chapter 4.

VITA

Boris D. Hernandez

Candidate for the Degree of

Doctor of Philosophy

Thesis: SUPPLEMENTS TO THE HEC-1 HYDROLOGIC MODEL,
USING THE MONTE CARLO METHOD AND
ARTIFICIAL NEURAL NETWORK

Major Field: Civil Engineering

Biographical:

Personal Data: Born in Guatemala City, Guatemala, On March 4, 1964.

Education: Graduated from Lompoc High School, Lompoc, California in May
1982; received Bachelor of Science degree in Civil Engineering from
San Diego State University, San Diego, California in May 1992; received
Master of Science in Civil Engineering from Oklahoma State University,
Stillwater, Oklahoma in December 1996. Completed the requirements for
the Doctor of Philosophy degree with a major in Civil Engineering at
Oklahoma State University in December 2001.

Experience: Worked for the City of National City located in San Diego,
California assisting the city engineer with road construction and
traffic related items. Employed by the U.S. Army Corps of
Engineers in 1992 to present working in the hydrology and
Hydraulics branch.

Professional Memberships: Society of American Military Engineers.