MIT

CONSTRAINED STOCHASTIC CLIMATE SIMULATION

by

David Carleton Curtis
and
Peter S. Eagleson

DEPARTMENT
OF
CIVIL
ENGINEERING

RALPH M. PARSONS LABORATORY

HYDROLOGY AND WATER RESOURCE SYSTEMS

Report Number 274

SCHOOL OF ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Cambridge, Massachusetts 02139

Prepared with the Support of the National Oceanic and Atmospheric Administration, the National Weather Service, and the National Science Foundation
Under Grant Nos. ATM-7812327 and ATM-8114723

May 1982

R82-25

CONSTRAINED STOCHASTIC CLIMATE SIMULATION

by

David Carleton Curtis

and

Peter S. Eagleson

RALPH M. PARSONS LABORATORY

HYDROLOGY AND WATER RESOURCE SYSTEMS

Report Number 274

Prepared with the Support of the National Oceanic and Atmospheric Administration, the National Weather Service, and the National Science Foundation

MAY 1982

R82-27

NOTICE

Readers interested in using the Computer Programs associated with this report should see the separate document of this laboratory:

CONSTRAINED STOCHASTIC CLIMATE SIMULATION:
COMPUTER PROGRAM AND USERS MANUAL

DAVID CARLETON CURTIS

Report Number 276

ABSTRACT

A stochastic, multivariate, hydrometeorological data generation algorithm is presented. Hourly values of precipitation, cloud cover, shortwave radiation, longwave radiation, temperature, dewpoint, wind speed, and wind direction are jointly generated for the two-meter level. The procedure is designed to provide coherent sets of input data for models of various land surface processes. The model's flexibility and economy allow the study of land surface responses to different atmospheric forcings.

Generated data plots, model output statistics, and generated mean diurnal curves are compared to observations for the months of January and July at two sites, Boston, Massachusetts and Dodge City, Kansas. Data representing three "climates", normal, wet, and temperature-biased were generated and applied to a detailed model of the land surface. The resulting energy fluxes across the land-atmosphere interface are reviewed and the differences are noted.

ACKNOWLEDGEMENTS

This work was performed with the support of the National Oceanic and Atmospheric Administration, the National Weather Service, and the National Science Foundation.

Professor P. S. Eagleson, my advisor for this project, was a constant source of encouragement and discipline. His exacting standards were an ever-present charge to improve, advance, and refine this effort.

Dr. Rafael Bras, Dr. Eric Adams, and Dr. Peter Stone of M.I.T. and Dr. Michael Hudlow and Dr. John Schaake of the National Weather Service are commended for many helpful comments and criticisms during the course of this project. Mr. Charles A. Smith, Hydrologist-in-Charge at the Northeast River Forecast Center, has been particularly supportive of my efforts over the past two years. Without Mr. Smith's support, completion of this project would have been considerably more difficult.

I will always be indebted to Mr. Joseph Bryan who, in the summer of 1979, searched through 15 years of professional effects to find an unpublished report of his that later formed the foundation for the temperature component described in this report.

The efforts of Chris Milly who developed the Land Surface Model used for the applications described in Chapter 12 of this report are deeply appreciated. By setting up and running the Land Surface Model with output from the CSCS Model, Chris provided an important contribution to this report.

LIST OF FIGURES

LIST OF FIGURES

LIST OF TABLES

NOTATION

C_H	bulk transfer coefficient for sensible heat
C_M	bulk transfer coefficient for momentum
C_W	bulk transfer coefficient for water vapor
D	Julian day
E	water vapor flux
G	heat flux into the ground
H	sensible heat flux
I_c	clear sky shortwave radiation
I_o	insolation
I_s'	total direct and diffuse shortwave radiation
K_H	eddy transfer coefficient for heat
K_M	eddy transfer coefficient for momentum
K_W	eddy transfer coefficient for water vapor
K(t)	radiation attenuation factor
L	Monin-Obukhov length
LE	turbulent latent heat diffusion into the atmosphere
M_0	mean fairweather cloud cover
N	cloud cover
$\overline{N}_a$	mean cloud cover for all inter-storm periods
$\overline{N}_t$	intermediate mean cloud cover during inter-storm periods
P_a	atmospheric pressure
P(t)	cloud cover transition function
$P_o(t)$	observed cloud cover transition

NOTATION

R	time of local sunrise
R_a	longwave radiation
R_g	total reflectivity of the ground
Ri	Richardson number
$(Ri)_B$	bulk Richardson number
R_n	net all wave radiation
R^*	"fairweather" region
S	time of local sunset
T	cloud transition period
$T(t)$	temperature
$T_d(t)$	dewpoint temperature
$\tilde{T}(t)$	deterministic temperature component
$T'(t)$	stochastic component of temperature
$\overline{T}_d$	mean hourly dewpoint temperature
$\tilde{T}_d(t)$	deterministic component of dewpoint temperature
$T_d'(t)$	dewpoint deviations
$T_g(t)$	ground surface temperature
$\overline{W}_d$	mean wind direction
$X_i(t)$	predictors in temperature regression equation
$Y(t)$	hourly temperature change
W_{bo}	solar constant
$W_d(t)$	wind direction

NOTATION

$\overline{W}_s$	mean wind speed
$W_s(t)$	wind speed
W_*	friction velocity
a_i	regression coefficients
a_1	molecular scattering factor
a'	mean atmospheric transmission coefficient for cloudless, dust-free, moist air after scattering only
b_i	coefficients of differential equation for temperature
c_p	specific heat of air
d	total dust depletion
d_a	depletion coefficient of the direct solar beam by dust absorption
d_i	regression coefficients for dewpoint temperature
d_0	zero displacement plane
d_s	depletion coefficient of the direct solar beam by dust absorption
e_o	atmospheric vapor pressure
e_s	saturation vapor pressure
f	relative humidity
g	acceleration of gravity
h	storm depth
k	von Karmon constant
m	relative thickness of the air mass

NOTATION

m_p	elevation adjusted optical air mass
$m(t)$	cloud cover deviations
n	turbidity factor
q_h	specific humidity
$q(t)$	longwave radiation
r	ratio of actual earth-sun distance to mean earth-sun distance
$r(t)$	$ds(t)/dt$
$s(t)$	sine of solar altitude
t_b	time between storms
t_0	arbitrary initial time
t_r	storm duration
w	mean monthly precipitable water
z	elevation
Φ_H	empirically determined adjustment factor for sensible heat profile
Φ_M	empirically determined adjustment factor for wind profile
Φ_t	random input for wind direction
Φ_W	empirically determined adjustment factor for water vapor profile
α	angle of radiation
β	(mean time between storms)$^{-1}$
γ	cloud cover decay coefficient - approaching storms

NOTATION

γ_ε	skew coefficient of ε_t
γ_s	skew coefficient of wind speed
δ	(mean storm duration)$^{-1}$
δ	declination of the sun
ε_a	effective atmospheric emittance
ε_t	random deviate
ζ	cloud cover decay coefficient - receding storms
ζ_t	standard normal deviate
η	(mean storm depth)$^{-1}$
θ_d	mean monthly surface dewpoint
ξ	is -1 for West longitude is +1 for East longitude
ρ_a	atmospheric density
ρ_d	lag-1 serial correlation coefficient of wind direction
$\rho_d(1)$	lag-1 serial correlation coefficient for dewpoint temperatures
ρ_s	lag-1 serial correlation coefficient of wind speed
$\rho_m(\tau)$	cloud cover serial correlation function
$\rho_{T'}$	lag-1 serial correlation coefficient for stochastic component of temperature
$\rho_{T'_{do}}(1)$	lag-1 serial correlation coefficient of dewpoint deviations
σ	Stefan-Boltzman constant
σ_d	standard deviation of hourly dewpoint temperature

NOTATION

σ_d	standard deviation of wind direction
σ_m^2	variance of the fairweather cloud cover
σ_s	standard deviation of wind speed
$\sigma_{T'}$	standard deviation of stochastic component of temperature
$\sigma_{T'_{do}}$	standard deviation of dewpoint deviations
τ	shear stress, hour angle of sun, lag in a serial correlation function, dummy variable of integration
τ_o	shear stress of surface
ϕ	local latitude
ψ_t	standard normal deviate

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Overview

Motivation for the research outlined in the following report is the growing need to provide high resolution hydrometeorological data for various computer simulation models of the physical processes taking place near the land surface. Subjects for such modelling include the transfer of heat and moisture across the land-atmosphere interface, plant growth, plant disease propogation, insect infestation, irrigation management, and crop forecasting. Each of these modelling efforts is becoming more sophisticated as our knowledge of the individual processes grows. Many of the processes are related, and efforts to couple related models are being made to study larger and more comprehensive land surface systems.

Data requirements of these studies include: precipitation, radiation, cloud cover, temperature, humidity, wind, etc. For many models, data at hourly intervals is highly desirable. This time resolution may be necessary when studying diurnal effects.

Previously, researchers had only historical observations from which to draw a statistically coherent set of input data. While it is true that observed data are the only data where all of the variable interactions survive intact, a researcher using such data is limited to a given set of statistics. If, for instance, a researcher wants to study the effect of a fundamental change in the statistical parameters of one input variable on a land surface process, there exists no rational way to modify the other inter-related input variables whose statistics would naturally be changed by the shift. For example, if the number of storms was to be increased, how would cloudiness, temperature, and incoming shortwave radiation be adjusted to accomodate the change?

The physical linkages between the variables that reflect the flow of heat, moisture, and momentum across the land-atmosphere interface are complex. Figure 1.1 provides some insight into the nature of these inter-relationships. It is clear that simple scaling of one variable would not be sufficient to realistically study the system-wide responses. A more sophisticated adjustment procedure is required.

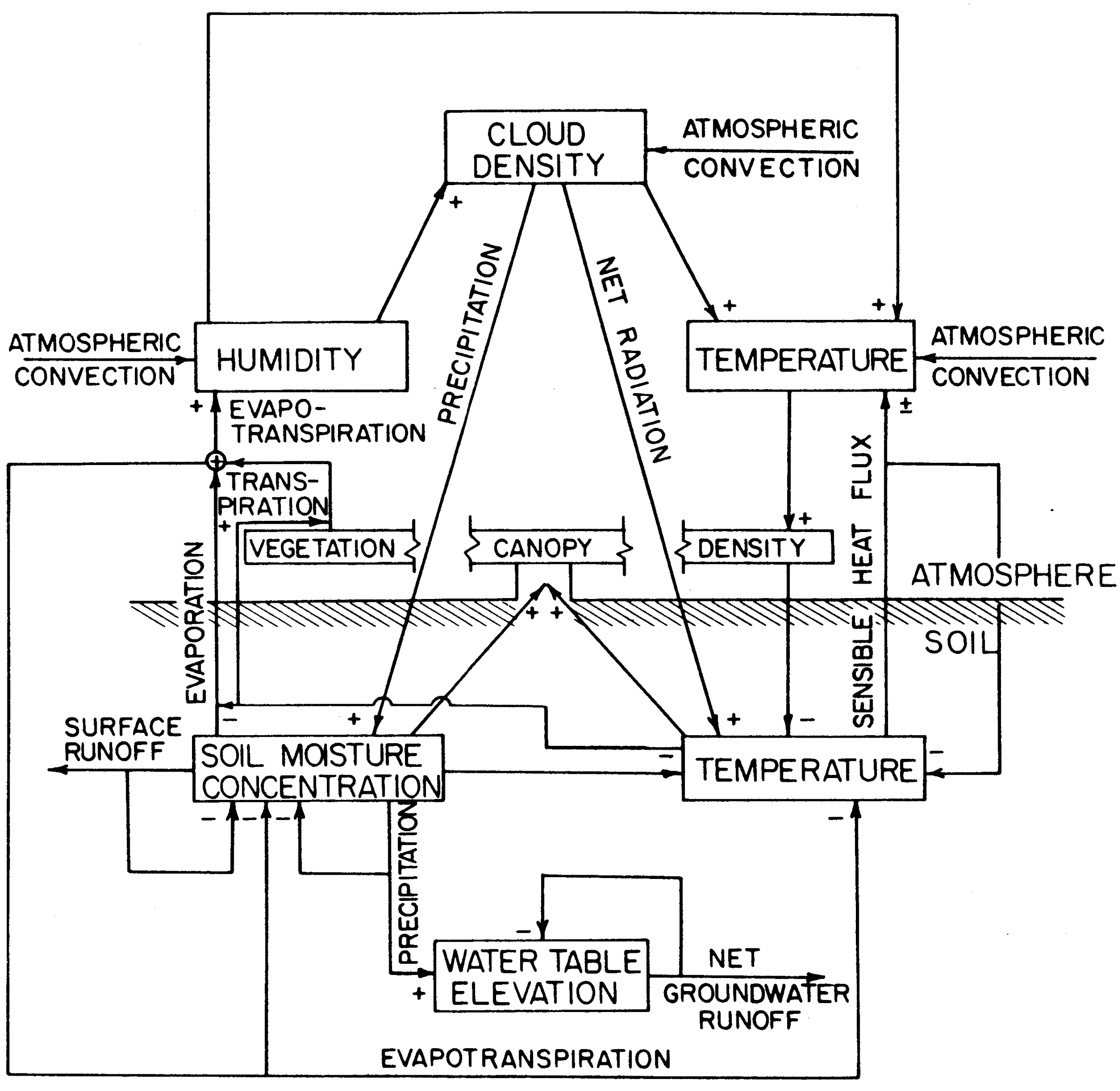


Figure 1.1 Land-Atmosphere Interconnections

Data could possibly be generated by existing computer models of planetary weather dynamics. Changing boundary conditions would produce a number of different weather scenarios which would provide the appropriate data. However, for most cases, the computer costs of this approach are still prohibitive.

Another approach would be to create data using multivariate stochastic generation techniques. However, severe non-stationarities, discontinuities, and unusual data distributions inhibit the application of multivariate techniques as they have traditionally been applied in hydrology.

Because of these problems, very few researchers have successfully developed algorithms to stochastically generate several weather variables simultaneously. Those that exist make some extreme simplifying assumptions, smooth the data, are applicable only at three or four specified times per day, and in general, are quite inflexible.

1.2 Multivariate Climate Data Generation: Previous Work

Kim (1976) generated time series of precipitation and temperature for use in snowmelt forecasting. However, he was able to show that, in his case, temperature and precipitation were statistically independent, greatly simplifying the problem.

Jones et.al. (1970) formulated an algorithm to generate rainfall, daily average temperature, and daily evaporation of water. The functional relationships among his weather variables can be summarized as

Rainfall = f(time of year, previous rainfall)

Temperature = f(time of year, current rainfall)

Evaporation = f(time of year, current rainfall, previous rainfall)

The approach of Jones et.al. was to analyze the historical data and use fitted high order polynomials to predict probability distribution parameters (e.g. means and variances) for each variable as a function of the week of the year. Polynomial equations were obtained based on the occurrence or non-occurrence of rainfall. For example, one equation predicting mean daily temperature was developed for dry days and another equation was developed for wet days. A similar approach was used to calculate the standard deviation of daily temperatures. The stochastic nature of daily temperature was then simulated by sampling from a normal distribution of temperatures having the derived mean and standard deviation for that particular day.

The approach of Jones et.al. considers only the day-to-day variations of the primary variables. Time variations of much less than one day are needed.

Ahmed (1974) developed a program to generate rainfall, ambient temperature, air humidity, short and longwave radiation, and wind speed to use in a dynamic simulation of crop behavior. The weather variable inter-relationships as specified by Ahmed were

Rainfall	=	f(location, probability of rainfall for current day)
Radiation	=	f(location, time of day, time of year, rainfall for the day, clear _or_ cloudy conditions)
Wind speed	=	f(location, time of day, time of year)
Temperature	=	f(location, time of day, time of year, rainfall for the day, clear _or_ cloudy conditions)
Air humidity	=	f(location, rainfall for the day, air temperature)

The description of Ahmed's functional relationships make this algorithm appear quite attractive, but his formulation and execution of them do not have the desired resolution and flexibility.

Rainfall is generated on a daily basis. No consideration is given to storm duration and hence to storm intensity.

Cloud cover, which is one of the most important ingredients in determining the surface energy balance, was treated by Ahmed as a binary variable. That is, cloud conditions were assumed to be either fully overcast or clear, nothing in between.

Ambient temperature was computed by generating weekly means. Empirical equations were used to convert weekly means to temperatures at 8:00 AM, 12:00 Noon, and 4:00 PM for each day of the week.

Two simplifying assumptions were also used in Ahmed's temperature formulation: 1) ambient temperature decreases in direct proportion to the amount of rainfall, and 2) the probability of clear or cloudy sky on any day was assumed proportional to the rainfall probability of that day.

Nicks (1975) developed a model to generate values for daily rainfall, daily minimum and maximum temperatures, and daily solar radiation. Rainfall was generated by a Markov chain process. The temperature and radiation data were generated individually by lag-1 Markov processes conditioned by current and preceding wet or dry days.

Probably the best effort thus far to jointly generate a set of meteorological data was presented by Richardson (1981). Richardson developed a procedure to generate daily precipitation, maximum temperature, minimum temperature, and solar radiation. Precipitation was generated independently using a Markov chain. Daily max/min temperatures and daily radiation data were generated using a multivariate model with means and standard deviations conditioned on the occurrence of wet or dry days. In this manner, Richardson was able to preserve the inter-relationships among the four variables.

For most of the models reviewed, time resolution was on the order of one day. No multivariate hydrometeorological data generation algorithms with time resolution as low as one hour have been found in the literature.

1.3 Constrained Stochastic Climate Simulation

The result of the current research is a computer model to stochastically generate ten hydrometeorological variables with hourly resolution. Included in the variable set are

1.	time between storms	6.	longwave radiation
2.	storm duration	7.	temperature
3.	storm depth	8.	dewpoint temperature
4.	cloud cover	9.	wind speed
5.	shortwave radiation	10.	wind direction

The general approach in constructing the model was to develop a set of stochastic elements that could be coupled and thus constrained by deterministic relationships in order to preserve as much of the important cross-correlations as possible. At the same time, the individual stochastic elements were designed to provide time series whose statistical properties approximate historical values.

To accomplish this task, several major hurdles had to be overcome. The two most important dealt with the generation of hourly cloud cover and the generation of hourly temperature.

Hourly cloud cover is a highly non-stationary variable. The first and second moment properties are obviously quite different during an intra-storm period than during an inter-storm period. A model was required that constrained cloud cover during storm events, provided for the proper transition into and out of storm periods, and permitted the occurrence of total cloud cover during an inter-storm period.

A technique was developed that allows the generation of a time series whose mean and variance at a given point in time are allowed to vary in a controlled fashion. This technique is an essential ingredient in providing much of the desired coordination between precipitation occurrences, cloud cover, short and longwave radiation, and temperature. It allows the "ripple" effects that would result from a change in precipitation statistics to be felt throughout the generated data set.

Hourly temperature also exhibits pronounced non-stationarities, both diurnally and seasonally. To attack this problem, a new methodology is used that is based on an expansion of ideas presented in an unpublished report by Bryan (1964). The technique generates hourly temperatures as a function of the previous hourly temperatures, short and longwave radiation, wind speed, and wind direction. (Provision was made to include a link to ground temperatures as well). Stochasticity is introduced by cloud cover as it affects short and longwave radiation and by superimposing a serially correlated series of random deviations on the calculated temperature.

The resolution of the cloud cover and temperature problems formed the framework that allowed the remaining elements to be knitted together to form a rational model. The model has been named Constrained Stochastic Climate Simulation (CSCS).

Chapters 2 - 8 present the theoretical development for each component of the CSCS model. Parameter estimation is discussed in Chapter 9. The results of four data generation experiments using the CSCS model appear in Chapter 10. The model has been tested for two time periods of the year, January and July. These two months were chosen because they correspond to a common procedure of January-July comparisons in the climate-modelling literature and because they represent two significanly different weather regimes.

Two different geographical locations were tested: 1) Boston, Massachusetts, and 2) Dodge City, Kansas. Coastal and continental climatic regimes are represented respectively by these locations.

Output from the CSCS model was also used as input to a detailed model of the land surface (Milly, 1982) to show its applicability to studies of land surface response to various meteorological forcings. (Chapter 12).

The CSCS model generates data that is representative of the 2-meter level. The land surface model used in Chapter 12 requires data at the surface or zero-meter level. Chapter 11 describes how the CSCS model is linked to the land surface model through the surface boundary layer. Stable, unstable, and neutrally stable atmospheric conditions are accounted for in establishing the various flux profiles.

In this project, the generated atmospheric data were used to directly force the land surface model. Feedbacks from the land surface model to the atmosphere are not explicitly accounted for, although the potential for coupling is built into the CSCS model.

By not accounting for the feedback mechanisms in this application, the CSCS-land surface system is in effect an "island" model. This means that the data representing the 2-meter atmospheric level at a point are unaffected by the local land surface conditions. The natural analogy for this situation would be a small island whose land surface processes were being forced by a meteorological data set that derived its properties from the areas surrounding the island.

Perhaps the most attractive feature of the CSCS model is its efficiency. On a DEC-10 time-share computer system, twelve months of hourly data can be generated in less than one CPU minute. Overall, the CSCS model should be an effective, flexible, and cost efficient tool to use in a wide variety of studies that require large amounts of hydrometeorologic data.

Chapter 2

PRECIPITATION MODEL

2.1 Introduction

Many stochastic precipitation models have been devised over the years to serve a variety of needs. The character of these models ranges from the simple to the complex. Each model attempts to satisfy certain statistical properties that are observable in a historical data base and are important to a particular application. Most of the precipitation models used in hydrologic applications, including those used in the multivariate weather data generators discussed in Chapter 1, describe the occurrence of daily precipitation. Kavvas and Delleur (1975) and Nicks (1975) provide good surveys of stochastic models of precipitation that appear in the literature.

Generally, these models describe the precipitation phenomenon in two stages. First, some sort of determination is made to decide if a wet or dry period has occurred. Second, if a wet period has occurred, the amount of precipitation for the period is computed.

For the current application, a precipitation model is needed that can yield data with hourly resolution, yet not overburden the project computationally. One model that satisfies these requirements is an alternating renewal process for independent, alternating wet and dry periods used by Grayman and Eagleson (1969).

2.2 Grayman-Eagleson Precipitation Model

Grayman and Eagleson found that a respectable sequence of synthetic rainfall data could be created by modelling the times between storms, t_b, storm durations, t_r, and the total storm depths, h. Detailed investigations of observed storm sequences by Grayman and Eagleson showed that storm durations and times between storms could be treated as independent events, but that storm depths were highly dependent on storm durations. Grayman and Eagleson also found that times between storms and storm durations could often be described as being exponentially distributed. Storm depths were found to follow a gamma distribution when conditioned by storm duration. Thus, the precipitation model can be expressed by successive sampling from the probability density functions (pdf) described by the following equations

Time between storms - pdf

$$f(t_b) = \beta e^{-\beta t_b} \quad , \quad t_b \geq 0 \tag{2.2-1}$$

where β = (mean time between storms)$^{-1}$

Storm duration - pdf

$$f(t_r) = \delta e^{-\delta t_r} \quad , \quad t_b \geq 0 \qquad (2.2\text{-}2)$$

where $\delta = (\text{mean storm duration})^{-1}$

Storm depth given storm duration - conditional pdf

$$f(\eta h|\delta t_r) = \frac{\eta(\eta h)^{\delta t_r - 1} e^{-\eta h}}{\Gamma(\delta t_r)} \quad , \quad h \geq 0 \qquad (2.2\text{-}3)$$

where $\eta = (\text{mean storm depth})^{-1}$

The solution procedure is as follows. At some initial time, say t_o, generate a time between storms, t_b. Once t_b is known, the period (t_o, t_o+t_b) is considered dry with the hourly precipitation set equal to zero. Next, when time, t, reaches $t_o + t_b$, the storm duration, t_r, is selected. The period $(t_o + t_b,\ t_o + t_b + t_r)$ is then considered wet. Using the value just computed for t_r, a storm depth is selected from the distribution described by Equation 2.2-3. When time reaches $t_o + t_b + t_r$, the process is repeated to determine the next storm sequence.

Presently, a uniform precipitation rate is assumed. Later versions of the CSCS model could easily contain an algorithm to provide variable intrastorm precipitation rates. But for now, hourly precipitation is found by dividing storm depth, h, by storm duration, t_r.

2.3 Data Generation

The data generation technique used for the exponential distributions of Equations 2.2-1 and 2.2-2 is straightforward and is described in Appendix A.

Generation of gamma distributed variates is not as easy. Direct selection of a gamma variate is complicated by the fact that the gamma probability density function cannot be analytically inverted. Therefore, indirect methods are required.

If the parameters of the gamma distribution are integer, a gamma variate can be determined by summing variates chosen from exponential distributions. However, the parameters of Equation 2.2-3 will generally be non-integer.

The method used by Grayman and Eagleson (1969) to generate a gamma variate, ηh, involved a mixture of techniques depending on the value of the product δt_r. Basically, the authors used a method of summing exponentially distributed variates when $\delta t_r \geq 1$ and a numerical integration technique when $0 \leq \delta t_r \leq 1$. The reason for using a different technique when $\delta t_r \geq 1$ results from the fact that for $0 \leq \delta t_r < 1$, the peak of the gamma distribution is located at $\eta h = 0$, but its magnitude is undefined. The situation where t_r is less than one occurs often, meaning that the numerical integration procedure is used frequently. A more efficient procedure to generate gamma variates is desired.

Curtis (1978) investigated three alternative techniques to generate gamma distributed variates. The first technique considered was a purely numerical technique used by Thom (1968) to generate direct and inverse tables of the gamma distribution. The second technique considered was an acceptance-rejection technique developed by Curtis (1978) that followed procedures outlined in Abramowitz and Stegun (1970). The third technique considered was another acceptance-rejection method presented by Fishman (1973).

Fishman's approach was by far the most efficient and worked for both integer and non-integer distribution parameters. The solution procedure for the Fishman technique is given in Appendix A.

2.4 Summary

With the implementation of the Fishman technique to generate gamma variates, a very efficient precipitation generator results. One big computational advantage is that this precipitation model yields hourly values, yet is only run aperiodically. In each dry-wet cycle, the precipitation model is "turned on" only two times. First at t_o, a time between storms is selected. Second, at $t_o + t_b$, a storm duration and a storm depth are computed. The rest of the time, the only computation that occurs is a simple check to determine if a new time between storms or a new storm duration is required. If no new variate is required the entire generation scheme is skipped. This contrasts with other methods, such as Markov Chain techniques, that require a solution of the generating scheme at each time step.

Another advantage of this particular precipitation model results from the generation of the time between storms, t_b. By knowing the times that storms begin, (and end for that matter), explicit and continuous coordination between the precipitation model and other CSCS components such as cloud cover, temperature, solar radiation is possible.

Previous investigators who have attempted to develop multivariate meteorological data generators have all recognized this coordination problem as manifested by the differences between meteorological variables on dry days as opposed to wet days. Different sets of equations had to be developed as "special cases" depending on whether a particular day was wet or dry. As will become clear in later chapters, the information provided by the precipitation model allows the development of a generalized set of equations that operate for all times, wet or dry.

CHAPTER 3

CLOUD COVER MODEL

3.1 Introduction

The evolution of cloud cover plays a critically important role in the flux of heat and moisture at the land surface. Energy balances are greatly affected as cloud cover continuously alters the transmission and reflection of radiant energy. Of course, cloud cover is also associated with precipitation inputs to the land surface moisture balance. Yet, cloud cover as a stochastic process has received very little treatment in the hydrologic literature.

Where studies have been performed, (Gringorten, 1971 and 1966; Fox and Rubin, 1965; Chargnon and Huff, 1957) cloud cover has been treated independently of other meteorologic processes. Developers of the various multivariate climate data generators discussed in Chapter 1 circumvented this issue by modelling net solar radiation, temperature, etc. The only time the effect of cloud cover was even implied in these works was through the development of separate sets of generating equations for wet days and for dry days. The lower temperatures and solar radiation levels on wet days implied the presence of more cloud cover than on dry days.

When interest is in the association of cloud cover and precipitation, the underlying modelling philosophy has been to follow the mechanics observed in the atmosphere. That is, clouds must be present prior to establishing the quantity of precipitation. However, as many meteorologists will say, one of their most difficult tasks is to predict total precipitation amounts when presented with a given atmospheric situation having precipitation potential.

In the following sections, a new approach will be used to model cloud cover as a stochastic process. The new technique overcomes many of the difficulties previous researchers have encountered when jointly generating meteorological data. It allows the establishment of the essential relationships between the meteorological variables of interest.

3.2 General Description

Cloud cover, $N(t)$, is a process that is bounded by 0 (clear sky) and 1 (overcast). Cloud conditions between these two extremes are reported in tenths. Thus, the observed cloud cover data set includes 0., .1, .2, ...8, .9, and 1.0.

Since the precipitation model divides time into two states, an inter-storm period and an intra-storm period, it seems reasonable to use some of this information to constrain the cloud cover model to conform to a certain set of conditions. One obvious condition that can be imposed immediately is that during an intra-storm period (i.e. $(t_0+t_b, t_0+t_b+t_r)$) cloud cover is total (i.e., $N(t) = 1.0$). This leaves only the inter-storm period within which to generate cloud cover.

To develop cloud cover during an inter-storm period, first consider $N(t)$ as a random process. Next, consider the expectation of $N(t)$ conditioned on the time between storms, t_b (i.e. $E(N(t)|t_b)$). If the process, $N(t)$, is examined near the beginning or near the end of an inter-storm period, $E(N(t)|t_b)$ would be close to 1.0. Whereas, if the process is examined near the middle of the inter-storm period, $E(N(t)|t_b)$ would usually be quite different from 1.0. Obviously, $N(t)$ is non-stationary.

The nature of the precipitation model discussed in Chapter 2 presents an interesting feature to the development of a cloud cover model. Generally, in simulation problems, only the past states of the system are known. The only thing known about the future is implied from the assumption that the statistical properties of future responses of the physical process being modelled will be identical to those observed in the past. In this problem, however, one future state is always known. Since the time until the next storm is part of the output of the precipitation model, the state $N(t_0+t_b) = 1.0$ is always known in addition to the past history of the system states.

The cloud cover process as defined here is very similar to the classic Dirichlet problem in mathematics. There a differential equation is constructed to describe a process that occurs within a bounded region. The solution is known initially and the solution at the boundary is known for all time, t, of interest. A solution is desired within the specified region.

The development of the cloud cover model will follow along the lines that are used to solve boundary value problems in differential equations. The proposed procedure is to acknowledge and analyze the properties of the function at the boundaries, infer the existence of properties of the function on the interior of the region, and select one of a possible set of solutions that satisfies the prescribed interior and boundary conditions.

3.3 Interior and Boundary Conditions

Boundary conditions of the inter-storm cloud cover process occur at the end of the previous storm event and at the beginning of the next event. At these times $N(t) = 1.0$. Overcast conditions (i.e. $N(t) = 1.0$) will not be precluded from inter-storm periods. However, no rainfall will be associated with the inter-storm overcast conditions.

From a statistical point of view, it is important to determine the moment properties of the process at the boundaries. The first moment, or the conditional expectation of $N(t)$ with respect to t_b at the end of the previous storm is

$$E(N(t_0)|t_b) = 1.0 \qquad (3.3\text{-}1)$$

since $N(t_0)$ is completely deterministic. Similarly, at $t=t_0+t_b$

$$E(N(t_0+t_b)|t_b) = 1.0 \qquad (3.3\text{-}2)$$

The second moment or conditional variance at the boundaries will be

$$VAR(N(t_0)|t_b) = 0 \qquad (3.3\text{-}3)$$

and

$$VAR(N(t_0+t_b)|t_b) = 0 \quad (3.3\text{-}4)$$

since the process is completely deterministic at the boundaries.

In the interior of the inter-storm region, imagine that the given t_b is long enough that there exists a subregion, R*, loosely centered around the midpoint of the inter-storm period in which the process N(t) can be assumed stationary. Thus, the first and second moment properties of N(t) when $t \varepsilon R^*$ are

$$E(N(t)|t_b) \equiv E(N(t)) = M_0 \quad (3.3\text{-}5)$$

and

$$VAR(N(t)|t_b) \equiv VAR(N(t)) = \sigma_m^{\ 2} \quad (3.3\text{-}6)$$

This implies the existence of a "fairweather" cloud cover process that is relatively unaffected by approaching or receding precipitation-producing systems.

Now that the existence of specific first and second moment properties of the process at the boundaries has been established and the existence of first and second moment properties in a sufficiently large interior region has been inferred, it is further suggested that there exists a smooth transition of moment properties from the boundaries to the interior region.

There may exist a whole set of solutions satisfying the established or inferred boundary and interior conditions. It is not the purpose here to find all or even a part of the set of possible solutions. It is sufficient to find just one that works.

3.4 Solution Development

One candidate solution is the function

$$N(t) = M_0 + (1-M_0)(1-P(t)) + m(t)P(t) \qquad (3.4\text{-}1)$$

where M_0 is the "fairweather" mean value of $N(t)$, $P(t)$ is the transition function, $m(t)$ is the stationary sequence of correlated deviations with $E(m(t)) = 0$, $VAR(m(t)) = \sigma_m^2$ and serial correlation function $\rho_m(\tau)$, where τ is lag.

Since by definition, M_0, $E(m(t))$, and $VAR(m(t))$ are not functions of time, the properties of the transition function must induce Equation 3.4-1 to meet the required boundary and interior conditions. At the boundaries, $N(t)$ becomes

$$N(t_0) = N(t_0+t_b) = 1 \qquad (3.4\text{-}2)$$

By inspection of Equation 3.4-1 with $N(t) = 1$, the following is required of $P(t)$

$$P(t_0) = P(t_0+t_b) = 0 \qquad (3.4\text{-}3)$$

Before proceeding further, the first and second moments at the boundaries of the process defined by Equation 3.4-1 will be verified. The conditional expected value of $N(t)$ is

$$E(N(t)|t_b) = E(M_0+(1-M_0)(1-P(t)) + m(t)P(t)) \quad (3.4-4)$$

For more detail refer to Appendix B. Completion of the operations indicated in Equation 3.4-4 leads to the expression for the <u>time varying</u> conditional expectation of cloud cover.

$$E(N(t)|t_b) = M_0 + (1-M_0)(1-P(t)) \quad (3.4-5)$$

Substitution of Equation 3.4-3 into Equation 3.4-5 at t_0 and t_0+t_b yields

$$E(N(t_0)|t_b) = E(N(t_0+t_b)|t_b) = 1 \quad (3.4-6)$$

as required by Equations 3.3-1 and 3.3-2.

Equation 3.3-5 specifies the requirement for $E(N(t)|t_b)$ when $t\varepsilon R^*$. Substitution of Equation 3.3-5 into Equation 3.4-5 gives

$$M_0 = M_0 + (1-M_0)(1-P(t)) \quad (3.4-7)$$

or

$$(1-M_0)(1-P(t)) = 0 \quad (3.4-8)$$

In order to have a meaningful solution, Equation 3.4-8 requires that

$$(1-P(t)) = 0 \quad , t \varepsilon R^* \tag{3.4-9}$$

or

$$P(t) = 1 \quad , t \varepsilon R^* \tag{3.4-10}$$

Thus, a second condition has been inferred for P(t).

The second moment property of Equation 3.4-1 is found by

$$VAR(N(t)|t_b) = E(N^2(t)|t_b) - E^2(N(t)|t_b) \tag{3.4-11}$$

Again the reader is referred to Appendix B for the details of evaluating Equation 3.4-11. Evaluation of Equation 3.4-11 leads to

$$VAR(N(t)|t_b) = \sigma_m^{\ 2} P^2(t) \tag{3.4-12}$$

To verify Equation 3.4-12 at the boundaries, substitute Equation 3.4-3 into Equation 3.4-12. Thus,

$$VAR(N(t_0)|t_b) = VAR(N(t_0+t_b)|t_b) = 0 \tag{3.4-13}$$

as required by Equations 3.3-3 and 3.3-4.

For the interior region, Equation 3.4-10 can be substituted into Equation 3.4-12 to show that

$$VAR(N(t)|t_b)_{t\varepsilon R^*} = \sigma_m^{\;2} \qquad (3.4\text{-}14)$$

as required by Equation 3.3-6.

It has now been demonstrated that Equation 3.4-1 can be a desirable solution to the cloud cover problem if the transition function P(t) has the following properties

$$P(t_0) = P(t_0+t_b) = 0 \qquad (3.4\text{-}15)$$

$$P(t) = 1 \text{ when } t\varepsilon R^* \qquad (3.\text{-}4\text{-}16)$$

One such function that satisfies the conditions of Equations 3.4-15 and 3.4-16 has the form

$$P(t) = (1 - e^{-\zeta(t-t_0)})(1 - e^{-\gamma(t_0+t_b-t)}) \qquad (3.4\text{-}17)$$

where ζ, γ are decay coefficients controlling the transition rates from the boundaries to R*. ζ would apply to receding storms and γ would apply to approaching storms. These transition rates could be different values, but for convenience, γ and ζ are assumed equal. Thus

$$P(t) = (1 - e^{-\gamma(t-t_0)})(1 - e^{-\gamma(t_0+t_b-t)}) \qquad (3.4\text{-}18)$$

To verify that Equation 3.4-18 satisfies the conditions set forth by Equations 3.4-14 and 3.4-15, the function is evaluated at t_0, t_0+t_b, and $t \varepsilon R^*$.

At t_0

$$P(t_0) = (1 - e^{-\gamma(t_0-t_0)})(1 - e^{-\gamma(t_0+t_b-t_0)})$$

$$= (1-1)(1 - e^{-\gamma(t_b)})$$

$$P(t_0) = 0 \tag{3.4-19}$$

At t_0+t_b

$$P(t_0+t_b) = (1 - e^{-\gamma(t_0+t_b-t_0)})(1 - e^{-\gamma(t_0+t_b-t_0-t_b)})$$

$$= (1 - e^{-\gamma t_b})(1-1)$$

$$P(t_0+t_b) = 0 \tag{3.4-20}$$

Finally, when $t \varepsilon R^*$

$$\lim_{t_b \to \infty} P(t) = 1 \tag{3.4-21}$$

Equation 3.4-21 suggests that the condition of Equation 3.4-16 is met only in the limit as $t_b \to \infty$. However, this is not a problem since, for all reasonable values of γ, P(t) will reach a value close to 1.0, say 0.99, sufficiently soon to permit practical application of the function. The value chosen for γ will be discussed in Chapter 9.

Another feature of the function N(t), that is shown in more detail in Appendix B, is the serial correlation function. The auto-correlation function of the cloud cover process defined by Equation 3.4-1 is

$$\rho_N(\tau) = \rho_m(\tau) \tag{3.4-22}$$

where $\rho_m(\tau)$ is the serial correlation function of the correlated random process, m(t). So, while the mean and variance of the cloud cover are controlled or modulated by the time varying function, P(t), the serial correlation function is unaffected.

3.5 Stationary Deviations Process

The stationary deviations process, m(t), is taken to be a simple first order Markov process defined by

$$m(t) = \rho_m(1)m(t-1) + \eta(t)\sqrt{1-\rho_m^2(1)} \tag{3.5-1}$$

where

$\rho_m(1)$ = lag-1 correlation coefficient

$\eta(t)$ = random deviate with

$E(\eta(t)) = 0$

$VAR(\eta(t)) = \sigma_m^{\ 2}$

In order for Equation 3.5-1 to be an appropriate model for the process, the auto-correlation structure of the natural process must follow

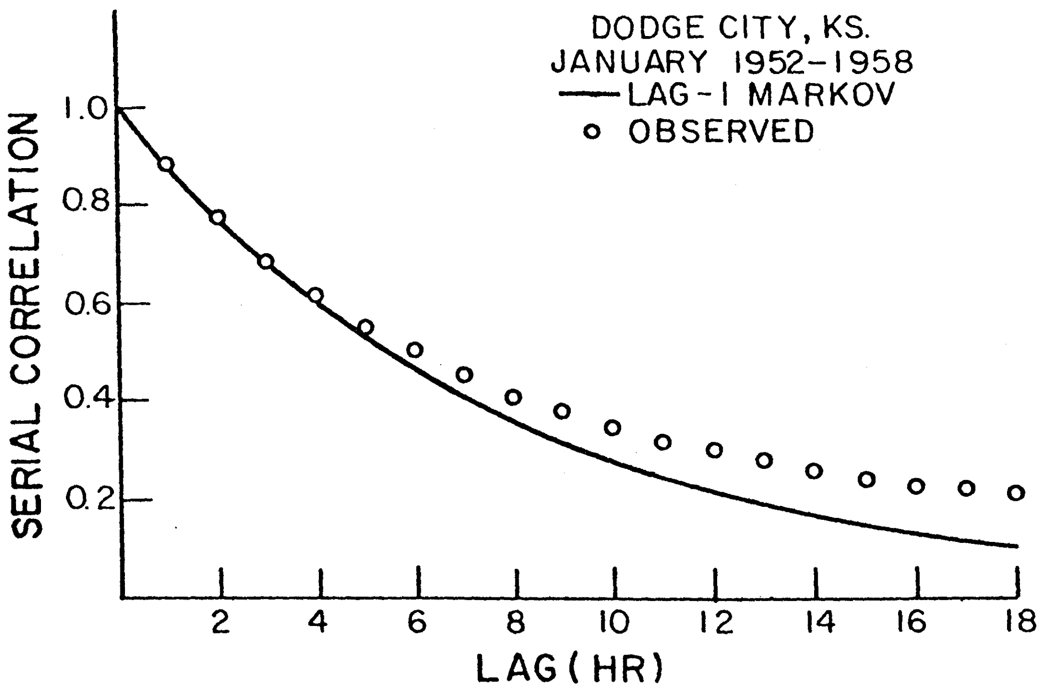
$$\rho_N(\tau) = \rho_N{}^{\tau}(1) \tag{3.5-2}$$

It turns out that the observed data used in this study follows Equation 3.5-2 sufficiently well to warrant the use of Equation 3.5-1 in the cloud cover model (See Figures 3.1-3.2).

3.6 Summary

A cloud cover model has been developed that satisfies a prescribed set of requirements during both inter-storm and intra-storm periods. A continuous transition from one set of conditions to the next is provided. The first and second moment properties of the cloud cover process are allowed to vary in a controlled fashion, while the auto-correlation structure is not affected by the transition function.

The process is capable of producing values that are less than zero or greater than one. Model output will, however, be constrained to $0 \leq N(t) \leq 1$. Actually, the fact



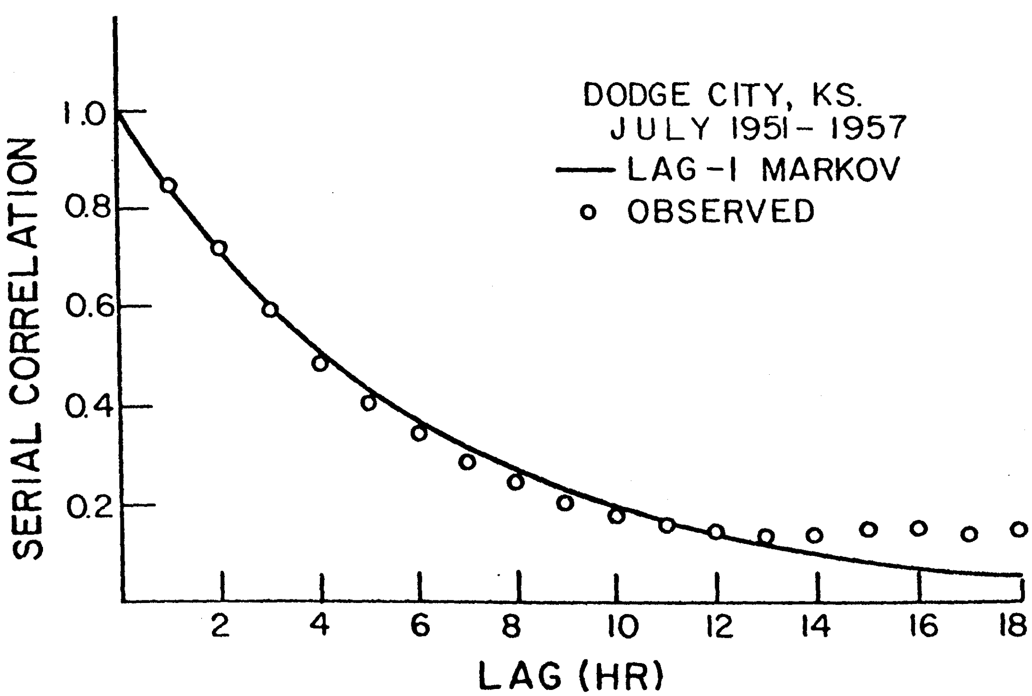
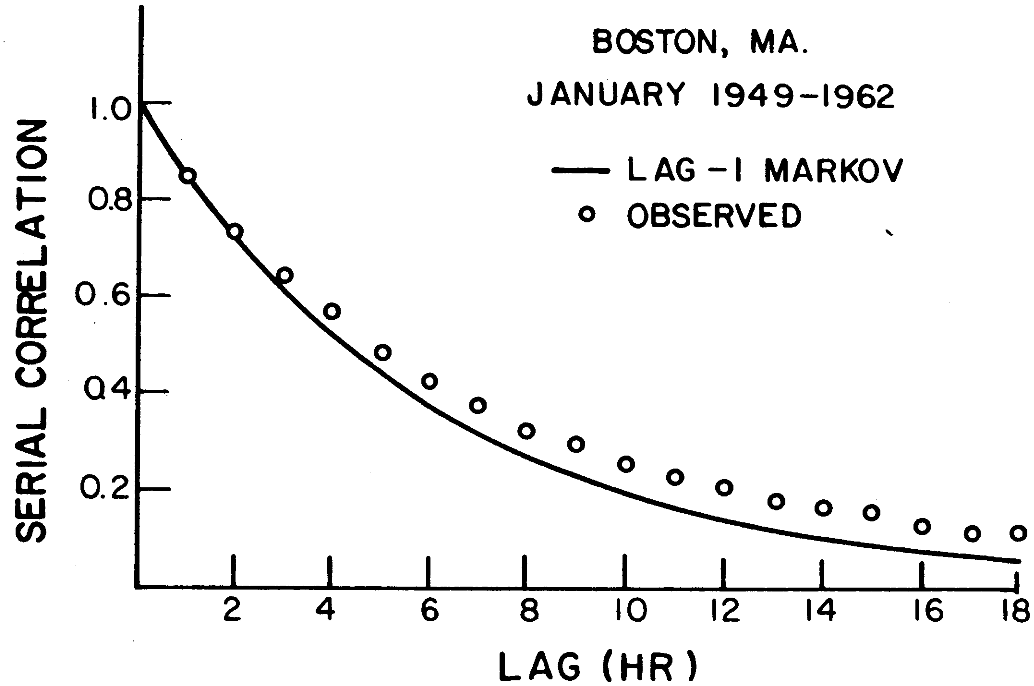


Figure 3.1 Fairweather Cloud Cover Serial Correlation (Dodge City, KS)



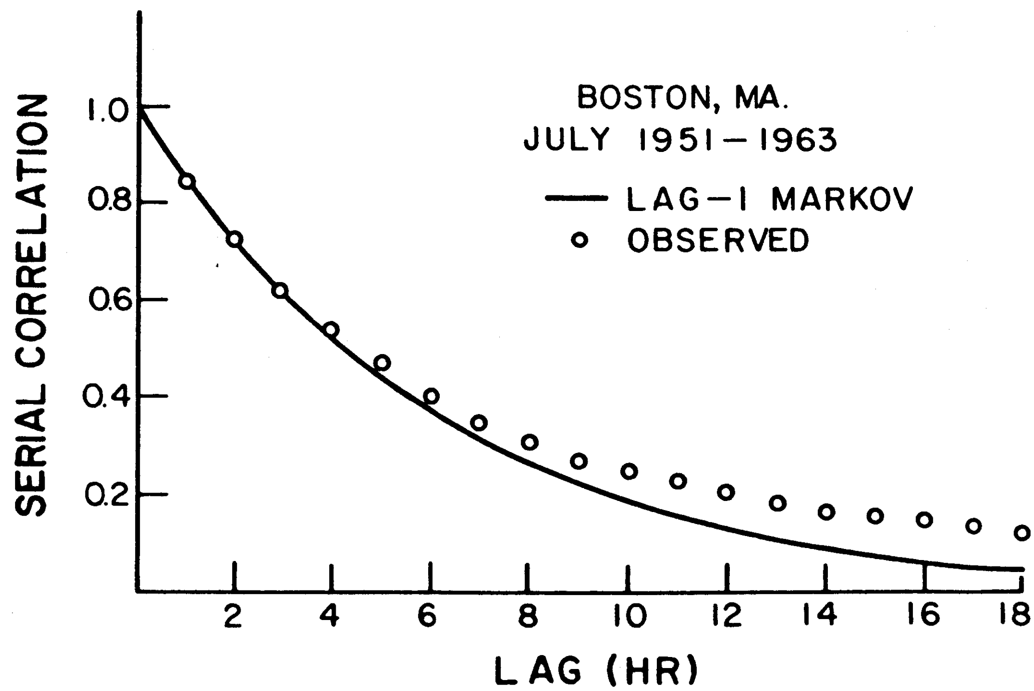


Figure 3.2 Fairweather Cloud Cover Serial Correlation (Boston, MA)

that the model described by Equation 3.4-1 can generate values outside the valid range for $N(t)$ is an advantage. It mimics the real atmosphere in the sense that the real atmosphere can assume a range of conditions with a clear sky, as well as with a totally cloudy sky.

Cloud cover viewed by a weather observer is just the manifestation of a set of atmospheric conditions that allows the formation of clouds. A clear sky is not just one atmospheric state, but a whole continuum of states "below" the cloud formation threshold. The atmosphere may be just below the cloud formation threshold or it may be well below the threshold and require the completion of a series of evolutionary atmospheric processes in order to form clouds again.

Similarly, overcast sky is not one state, but a continuum of states beyond the point where the sky is totally obscured. Total cloud cover may exist as a single very thin layer, a single very thick layer, or multiple layers of variable thickness and cover. A series of events must occur at the various atmospheric levels to cause the clouds to break up again.

Parameter estimation for the cloud cover model will be discussed in Chapter 9.

Chapter 4

SHORTWAVE RADIATION MODEL

4.1 Introduction

One of the most important variables in the surface energy balance is, of course, solar or shortwave radiation. Solar input is highly variable and nonstationary, both daily and seasonally. The shortwave radiation model proposed in the following sections will be used to generate hourly values of solar input at any time of the year.

Since, for all practical purposes, the sun radiates its energy at a constant rate, much of the variation in the amount of radiant energy actually intercepted by the earth can be described by the mechanics of earth's rotation about its axis and by its orbital path about the sun. The equations describing the earth's motion are well known and straightforward.

The real difficulty lies in the description of what happens to the shortwave radiation as it passes through the earth's atmosphere on its way to the surface. A multitude of particulate and molecular atmospheric constituents scatter, reflect, and absorb radiant energy. Analytical evaluation of these effects is all but impossible. Fortunately, a number of empirical relationships have evolved through observation and experimentation that allow estimates of radiation finally reaching the earth's surface.

4.2 Shortwave Radiation

As mentioned previously, the sun radiates energy at a nearly constant rate. The average intensity of solar radiation received on a plane unit area normal to the incident radiation at the outer limit of the earth's atmosphere is called the solar constant. A commonly used value for the solar constant, W_{bo}, (Eagleson, 1970) is:

$$W_{bo} = 2.0 \text{ cal}\cdot\text{cm}^{-2}\cdot\text{min}^{-1} \tag{4.2-1}$$

The portion of W_{bo} incident on a horizontal surface is generally of more interest and is referred to as insolation, I_o.

$$I_o = \frac{W_{bo}}{r^2}\sin\alpha \tag{4.2-2}$$

The solar altitude or angle of radiation, α, with the horizontal is given by

$$\sin\alpha = \sin\delta\sin\phi + \cos\delta\cos\phi\cos\tau \tag{4.2-3}$$

where δ is the declination of the sun, ϕ is the local latitude, and τ is the hour angle of the sun. The variable r is the ratio of actual earth-sun distance to mean earth-sun distance and is given by (TVA, 1972)

$$r = 1.0 + 0.017 \cos\left[\frac{2\pi}{365}(186-D)\right] \quad (4.2\text{-}4)$$

where D is the Julian day (i.e. $1 \leq D \leq 365$ or 366).

The sun's declination varies throughout the year and from year to year. Hence, declination values are usually published in tabular form (List, 1963). However, an approximation formula that is sufficiently accurate for heat transfer computations is available (TVA, 1972). Thus

$$\delta \simeq \frac{23.45\pi}{180} \cos\left[\frac{2\pi}{365}(172-D)\right] \quad (4.2\text{-}5)$$

The local hour angle, τ, can be computed from

$$\tau = ST + 12 - DTSL + ET \quad (4.2\text{-}6)$$

when the sun is east of the observer's meridian and from

$$\tau = ST - 12 - DTSL + ET \quad (4.2\text{-}7)$$

when the sun is west of the observer's meridian. The variables in Equations 4.2-6 and 4.2-7 are defined as

ST = standard time in the time zone of the observer in hours counted from midnight (e.g. $0{:}00 \leq ST \leq 23{:}59$).

DTSL = time difference between local and standard meridian in hours

$= \frac{\xi}{15}(LSM-LLM)$

where ξ is -1 for WEST longitude, ξ is for +1 for

EAST longitude, LSM is the longitude of the standard meridian and LLM is the longitude of the observer's meridian.

ET = difference between true solar time and mean solar time in hours. (Usually neglected for heat transfer computations . ET = 0 here).

The total radiation for a given period, $\Delta t = t_2 - t_1$, can be found by substituting Equation 4.2-3 into Equation 4.2-2 and integrating.

$$\Delta t^{I_o} = \int_{t_1}^{t_2} I_o dt = \int_{t_1}^{t_2} \frac{W_{bo}}{r^2} (\sin\delta\sin\phi + \cos\delta\cos\phi\cos\tau)dt \quad (4.2\text{-}8)$$

$$\Delta t^{I_o} = \frac{W_{bo}}{r^2} \left\{ \int_{t_1}^{t_2} \sin\delta\sin\phi dt + \int_{t_1}^{t_2} \cos\delta\cos\phi\cos\tau dt \right\} \quad (4.2\text{-}9)$$

In the evaluation of the first integral on the right-hand side of Equation 4.2-9, δ and ϕ are considered constant over the interval. Thus,

$$\int_{t_1}^{t_2} \sin\delta\sin\phi dt = \sin\delta\sin\phi(t_2 - t_1) \quad (4.2\text{-}10)$$

In the second integral on the right-hand side of Equation 4.2-9, δ and ϕ are again held constant, but τ is a function of time, t.

By introducing the change of variables

$$\tau = \frac{2\pi}{24}\,t \tag{4.2-11}$$

to transform hours to radians, the second integral becomes

$$\int_{t_1}^{t_1} \cos\delta\cos\phi\cos\tau\,dt = \cos\delta\cos\phi \int_{t_1}^{t_2} \cos\left(\frac{24\tau}{2\pi}\right) d\tau \tag{4.2-12}$$

$$\int_{t_1}^{t_2} \cos\delta\cos\phi\cos\tau\,dt = \frac{12}{\pi}\cos\delta\cos\phi(\sin(\tau_2) - \sin(\tau_1)) \tag{4.2-13}$$

Now by substituting Equation 4.2-10 and Equation 4.2-13 into Equation 4.2-9, the total hourly isolation is computed as

$$_{\Delta t}I_o = \frac{W_{bo}}{r^2}\left[(t_2 - t_1)\sin\delta\sin\phi + \frac{12}{\pi}\cos\delta\cos\phi(\sin(\tau_2) - \sin(\tau_1))\right] \tag{4.2-14}$$

the hour angle τ should fall in the range $0 \leq \tau \leq 2\pi$. However, when t is near noon standard time, discrepancies may arise due to the non-synchronization with true solar noon. Thus, if $\tau < 0$ as computed by Equation 4.2-11, just add 2π. Similarly, if $\tau > 2\pi$ from Equation 4.2-11, subtract 2π.

Sunrise and sunset are assumed to occur at $\alpha=0$. Obstructions near the horizon and refraction considerations are ignored.

4.3 Clear Sky Shortwave Radiation

Eagleson (1970) quotes the following equation for the attenuation of the radiation spectrum under clear skies, based on the monochromatic arguments of Beer's Law.

$$\frac{I_c}{I_o} = \exp(-na_1m) \qquad (4.3\text{-}1)$$

where I_c is clear sky radiation, a_1 is a molecular scattering factor (a_1 = 0.128 - 0.054 log m), m is the relative thickness of the air mass (m = cosecα), and n is a turbidity factor (2.0 for clear air, 5.0 for smoggy urban air).

TVA (1972) considers that attenuation relationships of the form of Equation 4.3-1 to be valid only for monochromatic radiation and can therefore be considered only as an approximation when used to compute the attenuation of the total spectral solar radiation flux. However, its simplicity is attractive. For the current version of the CSCS model, Equa-4.3-1 is used.

However, it is prudent at this point to present an alternative to Equation 4.3-1 that should be considered in future versions of the CSCS model. Atmospheric transmission of the solar beam is a function of a number of variables including dust, moisture, elevation, ground cover, solar altitude, etc. Referring to TVA (1972), a method used by Klein (1948) incorporates these elements. For clear sky solar radiation

$$\frac{I_c}{I_o} = \frac{a' + 0.5(1-a'-d) - 0.5d_a}{1 - 0.5R_g(1-a'+d_s)} \tag{4.3-2}$$

$$a' = e^{-(0.465+0.134w)(0.129+0.171e^{-0.880m_p})m_p} \tag{4.3-3}$$

$$w = e^{(-0.981+0.0341\theta_d)} \tag{4.3-4}$$

$$m_p = m((288 - 0.0065z)/288)^{5.256} \tag{4.3-5}$$

$$m = (\sin\alpha + 0.1500\ (\alpha + 3.885)^{-1.253})^{-1} \tag{4.3-6}$$

$$d = d_s + d_a \tag{4.3-7}$$

where a' is the mean atmospheric transmission coefficient for cloudless, dust-free, moist air after scattering only, w is the mean monthly precipitable water content in cm, θ_d is the mean monthly surface dewpoint, in ^oF, measured at the 2m-level, m is the optical air mass, dimensionless, m_p is the elevation or pressure adjusted optical air mass, dimensionless, z is the elevation in meters, α is the solar altitude in degrees, d is the total dust depletion, d_s is the depletion coefficient of the direct solar beam by dust absorption, and R_g is the total reflectivity of the ground. Some of the coefficients that appear in the preceding equations may vary with location and time of year. TVA (1972) provides brief summaries of coefficients at different locations and refers to studies providing more comprehensive lists (e.g. Kimball, 1927, 1928, 1929; Fritz, 1949; Bolrenga, 1964; Reitan, 1960, etc.)

4.4 Cloudy Sky Shortwave Radiation

The presence of clouds will further reduce the amount of shortwave radiation reaching the earth's surface. The amount of additional attenuation depends not only on the cloud cover but cloud type, thickness and elevation.

The U. S. Army Corps of Engineers (1956) gives the following relationship to estimate the impact of cloud cover.

$$\frac{I_s'}{I_c} = 1 - (1-K)N \qquad (4.4\text{-}1)$$

where I_s' is the total direct and diffuse shortwave radiation, N is the fraction of sky obscurred by clouds, and K is a coefficient to account for altitude considerations.

$$K = 0.18 + 0.0853(10^{-3})z \qquad (4.4\text{-}2)$$

where z is the cloud base altitude in meters.

Prior to Equation 4.4-1, all equations in Chapter 4 have been deterministic. With the introduction of N and K, the stochastic element has now entered the solar radiation generation process. Cloud cover, N, was discussed in Chapter 3.

The stochastic generation of K is not particularly easy. Any relationships that might logically be expected to exist between K and N are difficult to identify, due to the way data for z are reported. Cloud base altitude is only reported when $N > 0.50$. For $N < 0.50$, z is reported as "unlimited ceiling".

The scale on which z is reported also varies with altitude. For example, z may be reported in 30 to 150m (100-500 ft.) intervals when z is small and 1500-3000m (5,000-10,000 ft.) intervals when z is large. To avoid the problems with establishing K, an alternative attenuation function is desired that is a function of N alone.

TVA (1972) reports that the relationship

$$\frac{I_s'}{I_c} = 1.0 - 0.65N^2 \qquad (4.4\text{-}3)$$

provides reasonable results. Under certain kinds of cloud cover, Equation 4.4-3 can give values for attenuation that are too high. As $N \to 1$ for high thin cloudiness, more radiant energy passes through than Equation 4.4-3 would indicate. To help alleviate this problem, total opaque cloud cover is used instead of total cloud cover. Opaque cloud cover data are also reported at first-order stations where total cloud cover is recorded and it gives a more accurate indicator of the current cloud deck's ability to attenuate solar energy.

4.5 Summary

A procedure for generating hourly values of shortwave radiation has been developed that uses predominantly deterministic techniques to establish "potential radiation". Stochasticity enters through the introduction of generated cloud covers that were discussed in Chapter 3. Seasonal and diurnal variations are handled through the equations describing the earth's motions about the sun and its own axis.

Perhaps one of the most important features presented thus far is that the depressed values of solar input observed on cloudy days are now accounted for. Since the cloud cover model is "synchronized" with the precipitation model, the shortwave generation model automatically follows in step. Furthermore, an infinite variety of radiation inputs are possible, even on a day with precipitation. For example, the precipitation may occur at night, clouds clear away, and maximum solar input is observed for the day. Or cloudiness and precipitation may last all day and a minimum solar input is generated. Any combination in between is also possible. This feature is one of the significant elements that is missing from the models in the current literature.

Chapter 5

LONGWAVE RADIATION MODEL

5.1 Introduction

Atmospheric constituents are heated by conduction, convection and radiation. These elements in turn emit what is known as atmospheric or longwave radiation. The incoming longwave radiation is another significant element of the land surface energy balance that must be simulated.

5.2 Longwave Radiation with Clear Skies

The temperature, density, and depth of atmospheric water vapor, carbon dioxide, and ozone largely determine the amount of longwave radiation at the land surface. The major source of variability in the total atmospheric emittance is associated with the emission of water vapor in the 8-14μm spectral window. (Idso, 1981)

Since atmospheric radiation is a function of the full depth of the atmosphere, and since routine soundings of atmospheric properties are not generally available, many researchers have attempted to estimate longwave radiation using parameters that can be measured at the land surface. The two most commonly used parameters are the atmospheric vapor pressure and air temperature, both measured at the 2m-level.

The effective emittance of a cloudless atmosphere is generally expressed as

$$\varepsilon_a = \frac{R_a}{\sigma T^4} \qquad (5.2\text{-}1)$$

where ε_a is the effective emittance, R_a is the longwave radiation of all wavelengths, σ is the Stefan-Boltzman constant ($0.826(10^{-10})$cal cm^{-2}min^{-1} $^oK^4$), and T is the 2m air temperature in ^oK.

Brunt (1932) and Angstrom (1915, 1936) developed equations for estimating ε_a based on atmospheric vapor pressure alone. Brunt's equation is of the form

$$\varepsilon_a = a + b(e_o)^{\frac{1}{2}}$$

and Angstrom reported

$$\varepsilon_a = \alpha - \beta 10^{-\gamma e}. \qquad (5.2\text{-}3)$$

where a, b, α, γ, and β are empirical constants.

Formulations that depend only on temperature include those of Swinbank (1963) and Idso and Jackson (1969). Swinbank developed

$$\varepsilon_a = \delta T^2 \qquad (5.2\text{-}4)$$

and Idso and Jackson used

$$\varepsilon_a = 1 - ce^{-d(273-T)^2} \tag{5.2-5}$$

where δ, c, and d are empirical constants.

Idso, in cooperation with several other researchers, led a number of investigations into the nature of atmospheric radiation through the 1970's. This work culminated in a 1981 publication which presented a new equation for full spectrum thermal radiation. The new equation takes into account both atmospheric water vapor and temperature. The new equation was developed to follow the body of evidence that links longwave radiation to the binding energies of certain hydrogen bonds. Idso's latest approach takes the form (Idso, 1981)

$$\varepsilon_a = 0.70 + 5.95(10^{-5})e_o e^{(1500/T)} \tag{5.2-6}$$

where e_o is in mb and T is in ^{o}K. Idso developed the model using data that ranged from 245^{o}K to 325^{o}K for T and from 3mb to 28mb for e_o.

To stochastically generate values representing longwave radiation, models to generate temperature and vapor pressure are required. The temperature generation scheme will be discussed in a later chapter. Vapor pressure will be computed as a function of the 2m dewpoint and the 2m temperature. Dewpoint will be a generated variable and will also be covered in a later chapter.

5.3 Atmospheric Water Vapor Pressure

As mentioned earlier, dewpoint and temperature will be used to compute vapor pressure as required by Equation 5.2-6. The path from dewpoint to vapor pressure is not particularly direct. Several steps are taken.

First, the saturation vapor pressure, e_s, is computed using an approximation formula found in Rasmussen (1979)

$$e_s = C_o+C_1T+C_2T^2+C_3T^3+C_4T^4+C_5T^5 \qquad (5.3\text{-}1)$$

where e_s is in mb and T is in ^oC. The coefficients of Equation 5.3-1 were given as

$$\begin{aligned} C_o &= 6.0689226 \\ C_1 &= 4.4358312(10^{-1}) \\ C_2 &= 1.4590816(10^{-2}) \\ C_3 &= 2.7619554(10^{-4}) \\ C_4 &= 2.9952590(10^{-6}) \\ C_5 &= 1.4398885(10^{-8}) \end{aligned} \qquad (5.3\text{-}2)$$

Equation 5.3-1 was indicated to be valid over the range -50^oC to $+50^oC$.

A more computationally efficient form of Equation 5.3-1 was actually used. Equation 5.3-1 can be rewritten as

$$e_s = C_o + T(C_1+T(C_2+T(C_3+T(C_4+TC_5)))) \qquad (5.3\text{-}3)$$

Equation 5.3-3 requires approximately half the effort to evaluate than does Equation 5.3-1.

The second step is to evaluate the relative humidity. Linsley, et.al. (1975) provide the following approximation

$$f \simeq \left(\frac{112 - 0.1T - T_d}{112 + 0.9T}\right) \qquad (5.3\text{-}4)$$

where f is the relative humidity, T is temperature in ^oC, and T_d is the dewpoint temperature in ^oC. For the range of -25^oC to $+45^oC$, Equation 5.3-4 approximates relative humidity to within 0.6 percent.

Relative humidity can be defined as

$$f = \frac{e_o}{e_s} \qquad (5.3\text{-}5)$$

Since f and e_s in Equation 5.3-5 are known, the remaining step is to solve Equation 5.3-5 for e_o and compute the vapor pressure needed by Equation 5.2-6.

5.4 Longwave Radiation with Cloudy Skies

The presence of clouds will increase longwave radiation due to the energy emitted by water and ice particles at the base of the clouds. Cloud type, temperature, and extent all have an impact on the total additional contribution. One correction factor found by TVA (1972) to work reasonably well for a variety of conditions is

$$K = (1 + 0.17N^2) \tag{5.4-1}$$

where N is cloud cover. Applying Equation 5.4-1 and Equation 5.2-6 to Equation 5.2-1 yields the final relationship used to generate longwave radiation.

$$R_a = (0.70 + 5.95(10^{-5})e_o e^{(1500/T)})(1 + 0.17N^2)\sigma T^4 \tag{5.4-2}$$

5.5 Summary

A generating scheme for longwave radiation has been developed using the latest results of Idso (1981) to determine the atmospheric emissivity. Stochastically generated temperatures and dewpoints are used to "drive" the longwave generator.

Chapter 6

TEMPERATURE MODEL

6.1 Introduction

In recent years, several researchers have attempted to generate temperatures stochastically. In some fashion, each investigator had to deal with the diurnal and seasonal cycles that appear in the data. These cycles account for much of the variability in observed temperature.

Because the periodicities are so evident, Fourier or harmonic techniques have often been used to generate temperatures. Kim (1976) and Song et.al. (1973) are two examples. Kim used Fourier techniques to generate an independent trace of daily temperatures for input to a snowmelt forecast model. Song et.al. developed a model to generate daily air temperatures and water temperatures for streams in the Missouri River Basin. Song et.al. proposed that air and water temperatures could be considered to contain a deterministic part and a stochastic part.

$$AT_i = \overline{AT}_i + AT_i' \qquad (6.1\text{-}1)$$

$$WT_i = \overline{WT}_i + WT_i'$$

where AT_i and WT_i are the respective average daily air temperature and the average daily water temperature on the ith day; $\overline{AT}_i$ and $\overline{WT}_i$ are the deterministic components; and AT_i' and WT_i' are the stochastic components.

The deterministic components, $\overline{AT}_i$ and $\overline{WT}_i$, were taken to have the general form

$$T_i = A + B\sin\frac{2\pi i}{365} + C\cos\frac{2\pi i}{365} \qquad (6.1\text{-}3)$$

where the coefficients A, B,and C were derived through regression analysis.

The stochastic components, AT_i' and WT_i' are not purely random. Serial and cross-correlations exist. Therefore, Song et.al. proposed that the water temperature departures be written as a function of the air temperature departures.

$$WT_i' = DAT_i' + \delta_i \qquad (6.1\text{-}4)$$

where δ is a random number with zero mean. Substituting Equation 6.1-4 into Equation 6.1-2 to get a temperature model (albeit for water instead of air) that enables the output to be correlated with a second time series.

$$WT_i = a + b\sin\frac{2\pi i}{365} + c\cos\frac{2\pi i}{365} + dAT_i' + \delta_i \qquad (6.1\text{-}5)$$

The coefficients a, b, c and d are evaluated through regression analysis.

Other researchers have created temperature generation models that essentially depend on techniques yielding weakly stationary processes (e.g. Markov lag-1). Seasonal variation is introduced by using different parameter sets for different times of the year. (Jones et.al., 1972; Ahmed, 1974; Nicks, 1975; Richardson, 1979, 1981). With the exception of Ahmed's model, all of these models generate daily temperatures (either mean or max-min) that are conditioned on the occurrence of wet or dry days. This approach attempts to account for the fact that on wet days temperatures tend to be lower than on dry days.

Nicks (1975), for example, generated daily maximum and minimum temperatures using a Markov lag-1 process. Four different sets of parameters were developed depending upon the current wet/dry sequence. Parameter sets were developed for a wet day following a wet day, a wet day following a dry day, a dry day following a wet day, and a dry day following a dry day.

Richardson (1979, 1981) used a similar approach but also considered "maximum temperature, minimum temperature, and solar radiation to be a continuous multivariate stochastic process". Richardson then used a multivariate generating approach (Yevjevich, 1972) that was conditioned by the current day's wet or dry state.

Ahmed (1974) also conditioned temperature by the current day's wet/dry state, but used a somewhat different approach. Ahmed was studying water-use efficiency in crop production systems and needed temperatures for time scales shorter than one day. Instead of continuously generating temperatures throughout the day, Ahmed simplified the problem by developing a set of equations designed to yield air temperature at three specific times each day.

At 8:00 a.m.:

$$T = \overline{T} - 3.0 \pm 1.5P_p - 0.5h \qquad (6.1\text{-}6)$$

At 12:00 noon:

$$T = \overline{T} + 2.0 \pm 1.5P_p - 0.5h \qquad (6.1\text{-}7)$$

At 4:00 p.m.:

$$T = \overline{T} + 1.0 \pm 1.5P_p - 0.5h \qquad (6.1\text{-}8)$$

where T is the air temperature in ^{o}C, $\overline{T}$ is the average temperature for the day in ^{o}C, P_p is the precipitation probability, and h is the amount of precipitation in cm. The + or - sign depends on the clear or cloudy conditions of the sky (i.e. a binary switch).

All of the approaches seen thus far eliminate the problem of diurnal variation by dealing with longer time scales or, as in Ahmed's case, develop an empirical set of equations for each time of interest. In effect, Ahmed's approach uses a daily time scale as well, since each equation is based on data from only one particular time of day. This is really no different than a max-min approach.

The literature on stochastic generation of temperatures at time scales of less than a day is quite limited. Perhaps that in itself is a statement of the difficulty of the problem. The literature certainly indicates that the need is there (Jones et.al. 1972; Nicks, 1975; Ahmed, 1974; Mishoe, 1978; Jones and Smerage, 1978, Baker, 1981) but the solution is not.

Only one relevant paper was found that approaches the problem of stochastic generation of temperatures at the hourly level. Hansen and Driscoll (1977) developed a mathematical model for the generation of hourly temperatures. They were able to develop a model of the periodic course of mean hourly temperatures using the first, 365th, 730th, and 1095th harmonics which correspond to the annual, daily, 12 hour and 8 hour variations.

$$T_t = \overline{T} + (A_1 \sin((360/N)t) + B_1 \cos((360/N)t))$$
$$+ (A_{365} \sin((360/N)365t) + B_{365} \cos((360/N)365t))$$
$$+ A_{730} \sin((360/N)730t) + B_{730} \cos((360/N)730t)$$
$$+ A_{1095} \sin((360/N)1095t) + B_{1095} \sin((360/N)1095t)) \tag{6.1-9}$$

where T_t is the temperature at hour t, $\overline{T}$ is the mean annual hourly temperature, A_i and B_i are amplitude coefficients, and N is the number of observations in the fundamental period.

To simulate the irregular and aperiodic variations of hourly temperatures, Hanson and Driscoll superimposed a sequence of serially correlated standard normal deviates upon the temperatures generated by Equation 6.1-9. A lag-1 Markov process was used.

For some reason, however, Hanson and Driscoll chose not to try to estimate what the variance of the superimposed set of deviations ought to be. Rather, the sequence was assumed to have a variance of one which caused, as the authors acknowledged, the overall model variance to be lower than the observed.

Unfortunately, none of the models discussed so far have both the refinement in the time scale and the necessary flexibility to rationally include the effects of other variables (e.g. cloud cover) on a continuous basis. A new approach must be defined.

6.2 Bryan's Temperature Forecast Model

In 1967 Gerrity published a report describing a physical-numerical model for the prediction of synoptic-scale low cloudiness. The model was designed to permit the investigation of the significance of certain boundary-layer processes for the development of horizontally extensive areas of low cloudiness. The model required temperature inputs at the lower boundary, the 2-m level. Gerrity chose an empirical method developed by Bryan (unpublished,1964) to estimate the temporal variation of the air temperature attributed to the divergence of radiative heat flux and the divergence of eddy heat flux. Bryan's method uses the equation

$$\frac{dT(t)}{dt} = b_o - b_1T(t) + b_2s(t) + b_3r(t) \qquad (6.2\text{-}1)$$

where T(t) is temperature, t is time in hours after local midnight.

$$s(t) = \sin\delta\sin\phi - \cos\delta\cos\phi\cos\frac{\pi t}{12} \quad (R\leq t\leq S) \qquad (6.2\text{-}2)$$

$$s(t) = 0 \text{ (otherwise)}$$

$$r(t) = \frac{ds(t)}{dt} = \frac{\pi}{12}\cos\delta\cos\phi\sin\frac{\pi t}{12} \quad (R\leq t\leq 12) \qquad (6.2\text{-}3)$$

$$r(t) = 0 \quad \text{(otherwise)}$$

and δ is the solar declination, ϕ is the local latitude, R is the local time of sunrise and S is the local time of sunset. Equation 6.2-1 gives the temperature change as a function of the current temperature and solar input as represented by the two terms s(t) and r(t). The solar input is then represented by the sine of the solar altitude. (This is especially interesting, since the relationship for the sine of the solar altitude also appears in the shortwave radiation model of Chapter 4. The possibility thus presents itself for possible linkage of the shortwave radiation model with a method for computing temperatures.)

Equation 6.2-1 can be integrated by using the integrating factor $e^{b_1 t}$. Thus

$$\frac{d}{dt}\left(e^{b_1 t} T(t)\right) = e^{b_1 t}\left(b_o + b_2\, s(t) + b_3 r(t)\right) \quad (6.2\text{-}4)$$

The solution of Equation 6.2-1 is

$$T(t) = T(t')e^{-b_1(t-t')} + e^{-b_1 t} F(t,t') \quad (6.2\text{-}5)$$

$$F(t,t') = b_o\int_{t'}^{t} e^{b_1\tau}d\tau + b_2\int_{t'}^{t} e^{b_1\tau}s(\tau)d\tau + b_3\int_{t'}^{t} e^{b_1\tau}r(\tau)d\tau \quad (6.2\text{-}6)$$

Equation 6.2-5 suggests that temperatures can be calculated for any time, t, if only the initial temperature is known (i.e. $T(t')$). Before Equation 6.2-5 can be evaluated, however, the coefficients b_i must be determined.

The standard method for determining the coefficients that arise from the solution of a differential equation is to apply known boundary or initial conditions and solve for the respective values of the coefficients. Bryan, however, developed a procedure to derive the coefficients by fitting the model to a set of observed data through regression.

The details of Bryan's method can be found in Appendix C. For readability, only the essential elements are presented here.

Equation 6.2-5 can be rewritten in the following form

$$T(t) = e^{-b_1}\left(T(t')e^{-b_1(t-1-t')} + e^{-b_1(t-1)}F(t-1,t')\right) + e^{-b_1 t}F(t,t-1) \tag{6.2-7}$$

The quantity inside the brackets is just $T(t-1)$. Thus Equation 6.2-7 becomes

$$T(t) = e^{-b_1}T(t-1) + e^{-b_1 t}F(t,t-1) \tag{6.2-8}$$

Equation 6.2-8 gives the current temperature based on the conditions an hour earlier at t-1. The hourly temperature change, $Y(t)$, is found by subtracting $T(t-1)$ from both sides of Equation 6.2-8.

$$Y(t) = -(1-e^{-b_1})T(t-1) + e^{-b_1 t}F(t,t-1) \qquad (6.2\text{-}9)$$

Next, substitute the expression for F(t,t-1) into Equation 6.2-9.

$$Y(t) = b_o e^{-b_1 t}\int_{t-1}^{t} e^{b_1 \tau}d\tau - (1-e^{-b_1})T(t-1) + b_2 e^{-b_1 t}\int_{t-1}^{t} e^{b_1 \tau}s(\tau)d\tau$$
$$+ b_3 e^{-b_1 t}\int_{t-1}^{t} e^{b_1 \tau}r(\tau)d\tau \qquad (6.2\text{-}10)$$

Evaluation of the first integral (I_1 for convenience) on the right hand side of Equation 6.2-10 leads to

$$I_1 = \frac{b_o}{b_1}(1 - e^{-b_1}) \qquad (6.2\text{-}11)$$

The last two integrals, I_2 and I_3, on the right hand side are complicated by the exponential term inside the integral. Bryan (1964) indicated that it was sufficient to use the mean value of $e^{b_1 \tau}$ over the integration interval and bring it outside the integral. Thus

$$E(e^{b_1 \tau}) = \frac{1}{b_1}(1 - e^{-b_1})e^{b_1 t} \qquad (6.2\text{-}12)$$

$$I_2 = \frac{b_2}{b_1}(1 - e^{-b_1})\int_{t-1}^{t} s(\tau)d\tau \qquad (6.2\text{-}13)$$

and

$$I_3 = \frac{b_3}{b_1}(1 - e^{-b_1})\int_{t-1}^{t} r(\tau)d\tau \qquad (6.2\text{-}14)$$

Substituting the expressions for I_1, I_2, and I_3 back into Equation 6.2-10 yields

$$Y(t) = \frac{b_o}{b_1}(1 - e^{-b_1}) - (1 - e^{-b_1})T(t-1) + \frac{b_2}{b_1}(1 - e^{-b_1})\int_{t-1}^{t} s(\tau)d\tau + \frac{b_3}{b_1}(1 - e^{-b_1})\int_{t-1}^{t} r(\tau)d\tau \qquad (6.2\text{-}15)$$

At this point, it may not be clear that Equation 6.2-15 is of a form that can be utilized to estimate the coefficients by regression. To establish this point, compare Equation 6.2-15 with the following term-by-term

$$Y(t) = a_o + a_1X_1(t) + a_2X_2(t) + a_3X_3(t) \qquad (6.2\text{-}16)$$

For the constants a_i

$$a_o = \frac{b_o}{b_1}(1 - e^{-b_1}) \qquad (6.2\text{-}17)$$

$$a_1 = -(1 - e^{-b_1}) \qquad (6.2\text{-}18)$$

$$a_2 = \frac{b_2}{b_1}(1 - e^{-b_1}) \qquad (6.2\text{-}19)$$

$$a_3 = \frac{b_3}{b_1}(1 - e^{-b_1}) \qquad (6.2\text{-}20)$$

For the predictors, $X_i(t)$

$$X_1(t) = T(t-1) \tag{6.2-21}$$

$$X_2(t) = \int_{t-1}^{t} s(\tau)d\tau \tag{6.2-22}$$

$$X_3(t) = \int_{t-1}^{t} r(\tau)d\tau \tag{6.2-23}$$

Once the a_i's have been determined by regression, the b_i's can easily be found since the set of Equations 6.2-17 through 6.2-20 is a system of four equations in four unknowns. Therefore, the b_i's are determined as

$$b_1 = -\ln(a_1 + 1) \tag{6.2-24}$$

and

$$b_i = -\frac{b_1}{a_1} a_i \ , \quad i = 0,2,3 \tag{6.2-25}$$

Now standard regression techniques can be used on the observed data set of hourly temperature changes to establish the b_i's. Once the b_i's are established, Equation 6.2-5 can be used to forecast temperature given only the initial temperature $T(t')$.

Since $s(t)$ and $r(t)$ operate only during certain portions of the day, the equations for both $Y(t)$ and $T(t)$ will have different forms depending upon the time of day. These different forms and the details of their development appear in Appendix C.

Bryan's temperature model presents some interesting possibilities. First, as was noted earlier, a direct linkage is evident between Bryan's temperature model and the shortwave radiation model through the joint use of the expression for the sine of the solar altitude. This allows the temperature model to continuously respond to the temporal variation of the solar signal. In addition, two other parameters in Bryan's approach help account for seasonal variations (i.e. declination,δ) and geographical influences (i.e. latitude,ϕ) on the solar input.

Flexibility is another key element in Bryan's model. Modifications could be made to the original Equation 6.2-1 to help account for the effects of cloud cover, longwave radiation, wind speed, wind direction, ground temperature, etc. If this could be done, then an expanded Bryan model could be used to trace a "deterministic" component of temperature upon which a random component could be superimposed as was done by Hansen and Driscoll (1977). Then an hourly stochastic temperature generator would exist that could be coordinated with other stochastic variables in a multivariate process.

6.3 Stochastic Temperature Generation

An expanded version of Equation 6.2-1 can be written as

$$\frac{d\tilde{T}(t)}{dt} + b_1\tilde{T}(t) = b_o + b_2K(t)s(t) + b_3K(t)r(t)$$
$$+ b_4q(t) + b_5T_g(t) + b_6W_s(t) + b_7W_d(t) \qquad (6.3\text{-}1)$$

where $\tilde{T}(t)$ is the deterministic component; K(t) is the radiation attenuation factor ($K(t) = 1 - 0.65N^2(t)$); N(t) is the cloud cover; q(t) is a longwave radiation estimate; $T_g(t)$ is the ground temperature; $W_s(t)$ is the wind speed; and $W_d(t)$ is the wind direction.

The longwave radiation estimate, q(t) is not the same as the longwave radiation calculated by Equation 5.4-2. Rather, the simpler Swinbank (1963) formulation was used with a cloud cover correction factor (TVA, 1972).

$$q(t) = 0.937(10^{-5})(1 + 0.17N^2(t))\sigma T^6(t) \qquad (6.3\text{-}2)$$

where σ is the Stefan-Boltzman constant, $0.826(10^{-10})$ cal $cm^{-2}min^{-1}{}^oK^{-4}$. One of the main reasons for including the term $b_4q(t)$ in Equation 6.3-1 was to insure that a term responding to the effects of cloud cover was present throughout the entire day. The other two terms that respond to cloud cover are only present during certain portions of the day. The term $b_4q(t)$ will be available all day and should be useful in explaining some of the differences in cooling observed on clear nights as opposed to cloudy nights.

Wind speed and wind direction were added as possible indicators of an advected temperature component. Wind direction, in particular, might give an indication of the sign of the advection (i.e. warming or cooling).

Wind direction is often reported in degrees azimuth measured from the north ($0^{\circ} \leq \text{azimuth} \leq 360^{\circ}$). Inclusion of wind azimuth in Equation 6.3-1 can cause some inconsistencies in parameter estimation. For example, an azimuth report of 360° or 10° physically indicate practically the same property, a northerly flow. However, statistically the two reports would indicate something quite different. The 10° report would be a value that is considered well below the mean value and the 360° report represents a value well above the mean. This problem will most notably affect the serial correlation estimates.

A transformed wind speed is used instead where

$$W_d(t) = \text{azimuth} \quad , \quad (0^{\circ} \leq \text{azimuth} \leq 180^{\circ}) \qquad (6.3\text{-}3)$$

and

$$W_d(t) = |\text{azimuth} - 360^o|, \; (180<\text{azimuth}\leq 360^o) \qquad (6.3\text{-}4)$$

This approach unfortunately filters out east-west influences but the relative impact of the north-south component remains.

To solve Equation 6.3-1, first note that q(t) is a non-linear function of temperature. Since q(t) is really only being used as in index, it is linearized using q(t-1) and bringing it outside the integral. Now the solution to Equation 6.3-1 becomes

$$\tilde{T}(t) = \tilde{T}(t')e^{-b_1(t-t')} + e^{-b_1 t}G(t,t') \qquad (6.3\text{-}5)$$

where

$$G(t,t') = b_o \int_{t'}^{t} e^{b_1\tau}d\tau + b_2 \int_{t'}^{t} e^{b_1\tau}K(\tau)s(\tau)d\tau$$

$$+ b_3 \int_{t'}^{t} e^{b_1\tau}K(\tau)r(\tau)d\tau + b_4\, q(t-1) \int_{t'}^{t} e^{b_1\tau}d\tau$$

$$+ b_5 \int_{t'}^{t} e^{b_1\tau}T_g(\tau)d\tau + b_6 \int_{t'}^{t} e^{b_1\tau}W_s(\tau)d\tau$$

$$+ b_7 \int_{t'}^{t} e^{b_1\tau}W_d(\tau)d\tau \qquad (6.3\text{-}6)$$

Parameter estimation can now proceed as was demonstrated in the previous section. The details appear in Appendix D.

The hourly temperature change can now be expressed as

$$
\begin{aligned}
Y(t) = {} & b_o e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} d\tau - (1 - e^{-b_1}) \tilde{T}(t-1) \\
& + b_2 e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} K(\tau) s(\tau) d\tau \\
& + b_3 e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} K(\tau) r(\tau) d\tau \\
& + b_4 e^{-b_1 t} q(t-1) \int_{t-1}^{t} e^{b_1 \tau} d\tau \\
& + b_5 e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} T_g(\tau) d\tau \\
& + b_6 e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} W_s(\tau) d\tau \\
& + b_7 e^{-b_1 t} \int_{t-1}^{t} e^{b_1 \tau} W_d(\tau) d\tau
\end{aligned}
\qquad (6.3\text{-}7)
$$

The term $e^{b_1 \tau}$ that appears in the integrals containing $s(\tau)$

and $r(\tau)$ is treated by using the mean value argument shown in the previous section (see Equation 6.2-12). The integration interval is short enough that the values $K(\tau)$, $T_g(\tau)$, $W_s(\tau)$, and $W_d(\tau)$ can be evaluated at time t and brought outside the respective integrals.

The regression formula for Y(t) is now

$$Y(t) = a_o + a_1X_1(t) + \ldots + a_7X_7(t) \tag{6.3-8}$$

where the coefficients a_i are

$$a_1 = -(1 - e^{b_1}) \tag{6.3-9}$$

$$a_i = -\frac{a_1}{b_1} b_i \quad , \; i = 0,2,3,\ldots,7 \tag{6.3-10}$$

and the predictors $X_i(t)$ are

$$X_1(t) = \tilde{T}(t-1) \tag{6.3-11}$$

$$X_2(t) = K(t) \int_{t-1}^{t} s(\tau)d\tau \tag{6.3-12}$$

$$X_3(t) = K(t) \int_{t-1}^{t} r(\tau)d\tau \tag{6.3-13}$$

$$X_4(t) = q(t-1) \tag{6.3-14}$$

$$X_5(t) = T_g(t) \tag{6.3-15}$$

$$X_6(t) = W_s(t) \tag{6.3-16}$$

$$X_7(t) = W_d(t) \tag{6.3-17}$$

Note that since the temperature at time t is the variable being computed, T(t-1) is used in Equation 6.3-14.

Once the a_i's have been estimated, the b_i's are easily found

$$b_1 = -\ln(a_1 + 1) \tag{6.3-18}$$

$$b_i = -\frac{b_1}{a_1} a_i \qquad i = 0,2,3,\ldots,7 \tag{6.3-19}$$

Now Equation 6.3-5 can be used to estimate the "deterministic" component of hourly temperatures.

The b_i's are developed for each period of interest. In the current application, observed hourly values of temperature change, opaque cloud cover, wind speed, and wind direction for a particular month were used to estimate the b_i's. Ground temperature data were not available. Thus, b_5 was set to 0.0.

Equation 6.3-5 is applied each day to compute temperatures at t = 0 (midnight), 1, 2, ..., 23. The initial temperature, $\tilde{T}(t')$, for the period is the 11:00 p.m. (t=23) temperature for the previous day.

The "deterministic" component is essentially the expected temperature given the set of predictor values. All of the temperature variability is not explained by the model. To represent the random element, a serially correlated set of random variates will be added to the "deterministic" trace. Thus, the hourly temperature, $T(t)$,

$$T(t) = \tilde{T}(t) + T'(t) \qquad (6.3\text{-}20)$$

where $\tilde{T}(t)$ is the "deterministic" element and $T'(t)$ is the random element.

The random element is defined as

$$T_o'(t) = T_o(t) - \tilde{T}(t) \qquad (6.3\text{-}21)$$

where $T_o'(t)$ is the observed deviation, $T_o(t)$ is the observed temperature and $\tilde{T}(t)$ is the deterministic component. The deviations are assumed to be approximated by a lag-1 Markov process.

$$T'(t) = \rho_{T'} T'(t-1) + \zeta_t \sigma_{T'} \sqrt{1-\rho_{T'}^2} \qquad (6.3\text{-}22)$$

where $\rho_{T'}$ is the lag-1 serial correlation, ζ_t is the standard normal deviate, and $\sigma_{T'}$ is the standard deviation.

6.4 Summary

The stochastic temperature model generates hourly temperatures as a function of the time of day, time of year, latitude, longitude, cloud cover, wind speed, longwave radiation, shortwave radiation, ground temperature, and wind direction. Also, because the precipitation model in effect "drives" the cloud cover generation, the temperature output is appropriately affected by the occurrence of precipitation. These features make the proposed stochastic temperature algorithm the keystone in the framework of the CSCS approach.

Chapter 7

WIND MODEL

7.1 Introduction

The wind component of the CSCS model is composed of two parts, wind speed and wind direction. Wind speeds are required as input to flux computations of the land-air interface. Wind speeds may also quantify, somewhat, advection processes for the temperature model. Wind direction is required as an advection indicator for the temperature model as described in Chapter 6.

For the most part, the cross-correlation coefficients between wind speed, wind direction, and the other variables in the CSCS model are relatively low, generally less than 0.35 (see Tables 7.1-7.4). Therefore, for this version of the CSCS model, both wind speed and wind direction are treated as independent lag-1 Markov processes.

7.2 Wind Speed

The frequency distributions of wind speeds tend to be positively skewed. A variety of probability distributions with this property have been applied to wind speeds. Among them are the Planck, Rayleigh, gamma and the Weibull. (Hennessey, 1977; Justus et.al., 1977; Sherlock, 1951). The

Weibull appears to be the most popular.

It is apparent then, that not only must the mean and variance of the generated data be reproduced, but the generated data should be skewed as well. One often-used approach in hydrology to generate skewed serially correlated data is the Thomas-Fiering method (Haan, 1977).

The equation for a lag-1 Markov process can be written

$$W_s(t) = \overline{W}_s + \rho_s(W_s(t-1) - \overline{W}_s) + \varepsilon_t \sigma_s \sqrt{1-\rho_s^2} \quad (7.2\text{-}1)$$

where $W_s(t)$ is the hourly wind speed, $\overline{W}_s$ is the mean hourly wind speed, ρ_s is the lag-1 serial correlation coefficient, and σ_s is the wind speed standard deviation. The variable ε_t is random and defined by Thomas and Fiering as

$$\varepsilon_t = \frac{2}{\gamma_\varepsilon}\left(1 + \frac{\gamma_\varepsilon \psi_t}{6} - \frac{\gamma_\varepsilon^2}{36}\right)^3 - \frac{2}{\gamma_\varepsilon} \quad (7.2\text{-}2)$$

where γ_ε is the skew coefficient of ε and ψ_t is a standard normal deviate. The skew coefficient of ε in turn is defined as

$$\gamma_\varepsilon = \frac{(1 - \rho_s^3)\gamma_s}{(1 - \rho_s^2)^{1.5}} \quad (7.2\text{-}3)$$

where γ_s is the skew coefficient determined from the wind speed data.

In some applications, the mean and standard deviation of hourly wind speed may not be independent of the time of day. This can result when surface-generated instabilities promote vertical exchanges. This allows greater momentum transfer from faster moving air aloft and increases surface winds. Since atmospheric stability follows a characteristic diurnal curve, wind speeds may as well. (Oke, 1978).

To approximate this property, the mean and variance in Equation 7.2-1 will be allowed to vary with time. Since there is a relatively smooth transition of the observed hourly means and standard deviations throughout the cycle, the minimum and maximum parameter values are entered with their respective times of occurrence. Parameter values for each hour are then found by linear interpolation.

7.3 Wind Direction

As mentioned previously, wind direction is generated as input to the temperature model as an indicator of advected heating or cooling components. Advection is due to variations in the spatial properties of the atmosphere. When dealing only with point data, however, it is quite difficult to identify the nature of advection, particularly for future time steps. Wind direction appears to be about the only point variable that could indicate advection. This is largely due to the fact that air masses coming to a location from different directions may have characteristically different properties. For instance, winds with a large northerly component may, on the average, bring cooler weather conditions than winds from the south.

The transformed wind direction discussed in Chapter 6 is generated by a lag-1 Markov process.

$$W_d(t) = \overline{W}_d + \rho_d(W_d(t-1) - \overline{W}_d) + \Phi_t\sqrt{1 - \rho_d^2} \qquad (7.3\text{-}1)$$

where $W_d(t)$ is the hourly transformed wind direction, $\overline{W}_d$ is the mean hourly transformed wind direction, and ρ_d is the lag-1 serial correlation coefficient. The variable Φ_t is a random input with zero mean and standard deviation equal to σ_d, the standard deviation of the transformed wind direction.

The distribution of transformed wind direction is, of course, bounded on the left by 0^o and on the right by 180^o. To generate a random variate for Equation 7.3-1, an algorithm was developed that will generate a random variate from an arbitrary frequency histogram. (See Appendix A; Curtis 1978;

Abramowitz and Stegun, 1970). Utilizing the observed frequency histogram of transformed wind direction, a random value, Θ_t, representing wind direction $(\overline{W}_d,\sigma_d)$ is selected. Thus, Φ_t can now be defined as

$$\Phi_t = \Theta_t - \overline{W}_d \tag{7.3-2}$$

to complete the wind direction model.

Table 7.1 Data Correlation Matrix for Dodge City, KS - July 1951 - 1957

TEMP	DEW	CLOUD COVER	WIND SPEED	WIND DIR.
1.00	-0.22	-0.28	0.31	0.26
	1.00	0.23	-0.10	-0.10
		1.00	-0.08	-0.25
			1.00	0.20
				1.00

Table 7.2 Data Correlation Matrix for Dodge City. KS - January 1952 - 1958

TEMP	DEW	CLOUD COVER	WIND SPEED	WIND DIR.
1.00	0.66	-0.10	0.10	0.20
	1.00	0.11	0.08	0.19
		1.00	0.12	-0.03
			1.00	-0.08
				1.00

Table 7.3 Data Correlation Matrix for Boston, MA - July 1951 - 1963

TEMP	DEW	CLOUD COVER	WIND SPEED	WIND DIR.
1.00	0.26	-0.21	0.35	0.18
	1.00	.30	-0.12	0.28
		1.00	-0.05	0.06
			1.00	0.04
				1.00

Table 7.4 Data Correlation Matrix for Boston, MA - January 1949 - 1962

TEMP	DEW	CLOUD COVER	WIND SPEED	WIND DIR.
1.00	0.88	0.33	-0.04	0.36
	1.00	0.48	-0.07	0.28
		1.00	-0.13	0.10
			1.00	-0.08
				1.00

CHAPTER 8

DEWPOINT MODEL

8.1 Introduction

Some measure of atmospheric moisture is required to establish a gradient for moisture transport processes at the land surface. Specific humidity, vapor pressure, relative humidity and dewpoint temperature are all common descriptors of atmospheric moisture content (Eagleson, 1970). Relative humidity and dewpoint data are more generally available since they are measured at National Weather Service first-order stations.

To simulate on an hourly basis, relative humidity appears to be the more difficult due to the strong diurnal variations attributed to temperature (Oke, 1978). Dewpoint, on the other hand, is much more stable during the course of a day (Lorenz, 1978). Therefore, dewpoint temperature is a more likely candidate for simulation.

Ahmed (1974), however, generated air humidity for his multivariate model in the following fashion

$$H_a = H_{as} \cdot H_r \tag{8.1-1}$$

where H_a is the air humidity (i.e. vapor density) in

g/m^3, H_{as} is the saturated air humidity (i.e. saturation vapor density) in g/m^3, and H_r is the relative humidity. Relative humidity for a particular time of day (8:00 AM, 12:00 Noon, 4:00 PM) was computed by linear interpolation between weekly mean values of H_r for the indicated times. H_{as} is a function of temperature and was computed using Murray's adaption of the Goff-Gratch equation (Van Bavel, et.al., 1973). This approach is quite simplistic since any natural stochasticity is filtered out by the use of weekly mean relative humidities. Also, humidities are computed only at three specified times of the day. Higher resolution is required in this study.

Gringorten (1966), in a study simulating the frequency and duration of weather events, suggested that dewpoints could adequately be generated by a lag-1 Markov process. This would be a reasonable approach if the mean hourly dewpoints did not change materially during the course of a day.

8.2 Dewpoint Generation

From the plots of observed hourly dewpoints in Figure 10.13 and Figure 10.15, it is clear that the mean diurnal variation of dewpoint is quite small. The difference between the maximum and minimum hourly dewpoints in Boston, MA was 1.1^{o}C for January and July. For Dodge City, KS, the difference was 2.4^{o}C for January and 1.7^{o}C for July.

It is also apparent from Figure 10.13 and Figure 10.15 that the hourly variation in dewpoint is not random. Rather, the hourly transitions are quite smooth. These variations where they are noticeable, can generally be explained by the short term dynamics at the land-air interface. For example, the pronounced morning minimum in the Dodge City data for January is likely due to the removal of atmospheric moisture near the surface due to frost formation. During the day, rising temperatures cause the moisture to return to the lower atmosphere, elevating the dewpoint again.

During July, the morning rise in dewpoint is probably due to the addition of moisture from evaporating dew. The subsequent dip in dewpoint temperatures in the afternoon is likely the result of instability-generated mixing with dryer air aloft. As the strength of the vertical instability subsides in late afternoon, moisture builds up again in the lowest atmospheric layer and the dewpoints rise.

Tables 7.1 - 7.4 present the lag-0 cross-correlation matrices for the observed data. The generally weak cross-correlations exhibited by the July data indicate that dewpoints could be generated independently.

Since the daily variation of July dewpoints for Boston and Dodge City are small, and since the July dewpoints are only weakly correlated with the other model variables, July dewpoints could be generated independently by a first-order Markov model as suggested by Gringorten (1966). Therefore, the July dewpoints will be generated by

$$T_d(t) = \overline{T}_d + \rho_d(1)(T_d(t-1)-\overline{T}_d) + \psi_t \sigma_d (1-\rho_d^2(1))^{\frac{1}{2}} \qquad (8.2\text{-}1)$$

where $T_d(t)$ is the hourly dewpoint in ^{o}C, $\overline{T}_d$ is the mean hourly dewpoint in ^{o}C, $\rho_d(1)$ is the lag-1 serial correlation coefficient, ψ_t is the standard normal deviate, and σ_d is the standard deviation of hourly dewpoint in ^{o}C.

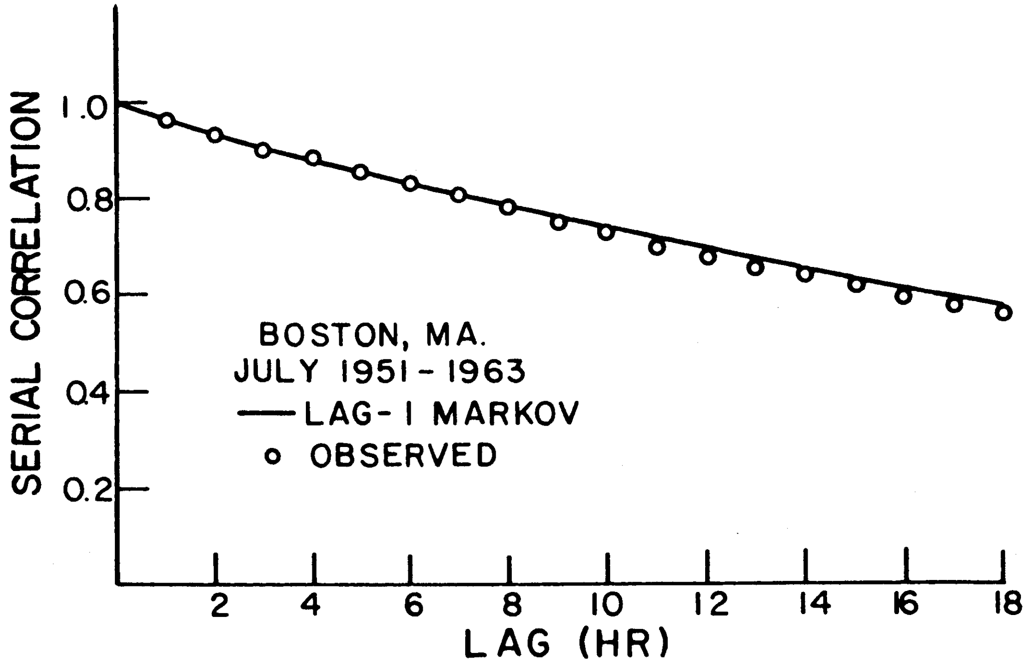
To affirm the choice of a first order Markov process to represent the July dewpoints, the observed serial correlation functions for July hourly dewpoints are plotted in

Figure 8.1 in comparison with the appropriate theoretical curve (i.e. $\rho(\tau) = \rho^{\tau}(1)$). The theoretical curve follows the Boston data very well. For the Dodge City data, the theoretical curve follows the observed data quite well only for the first six to eight hours. Beyond that point the theoretical curve falls faster than the observed. Overall, Equation 8.2-1 seems to be a reasonable choice for July dewpoints.

Since the January dewpoints appear to have a stronger cross-correlation structure with other CSCS model variables, January dewpoints will be assumed to be composed of a "deterministic" component and a random component. This approach follows that established for temperature generation in Chapter 6. The deterministic component, $\tilde{T}_d(t)$, will be estimated by linear regression. Thus

$$\tilde{T}_d(t) = d_0 + d_1\tilde{T}_d(t-1) + d_2T(t) + d_3N(t) + d_4W_s(t) + d_5W_d\,(t) \qquad (8.2\text{-}2)$$

where $\tilde{T}_d(t-1)$ is the previous hourly "deterministic" por-



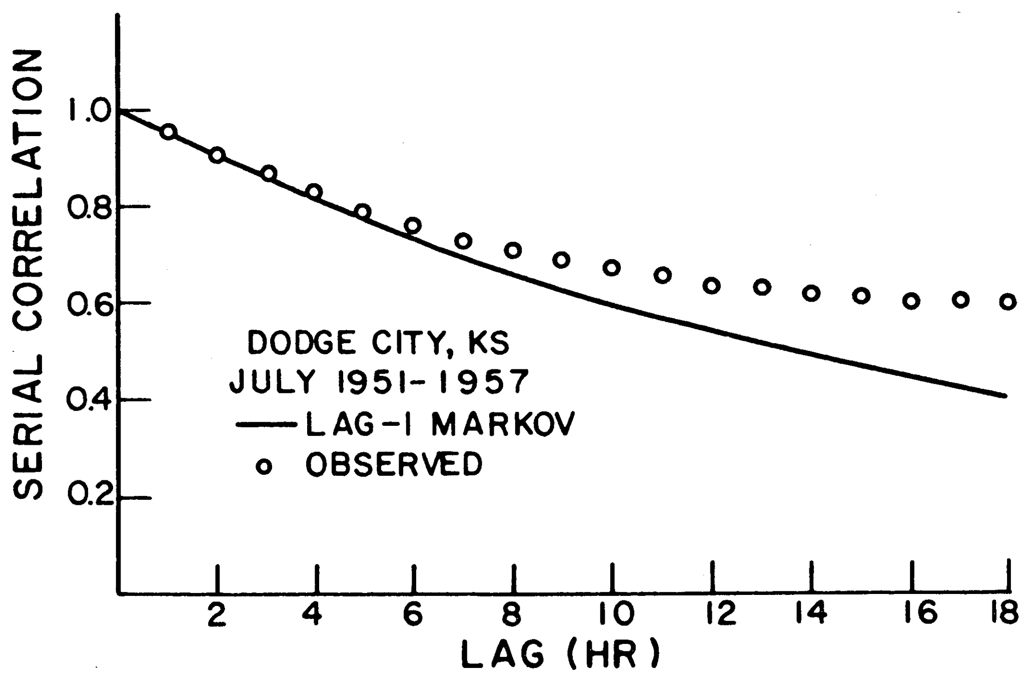


Figure 8.1 Serial Correlation of Hourly Dewpoint

tion in ^{o}C, $T(t)$ is the current temperature in ^{o}C, $N(t)$ is the cloud cover, W_s is the wind speed in ms^{-1}, and W_d is the wind direction in degrees $(0^{o} \leq W_d \leq 180^{o})$. The d_i's are coefficients to be estimated by standard linear regression techniques.

The random component will be treated as a lag-1 Markov process which represents a deviations process defined by

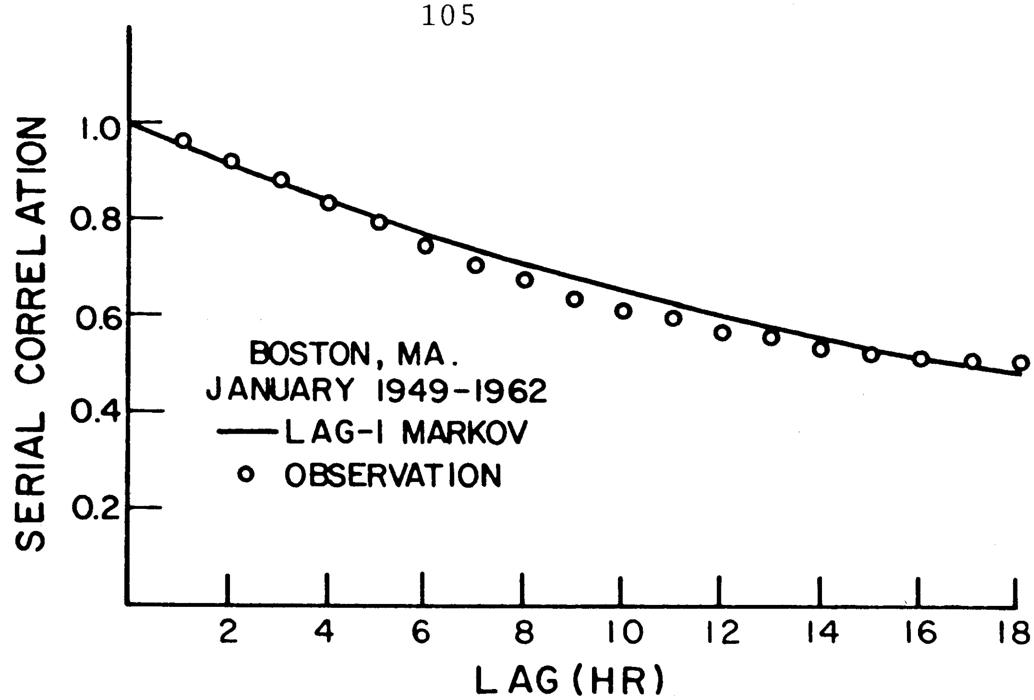
$$T'_{do}(t) = T_{do}(t) - \tilde{T}(t) \qquad (8.2\text{-}3)$$

where $T_{do}(t)$ is the observed dewpoint in ^{o}C, $\tilde{T}(t)$ is the dewpoint in ^{o}C generated by Equation 8.2-3 using observed data as input, and $T'_{do}(t)$ is the observed dewpoint temperature deviation in ^{o}C.

The dewpoint deviations are generated by

$$T'_d(t) = \rho_{T'_{do}}(1)T'_d(t-1) + \psi_t \sigma_{T'_{do}} (1-\rho^2_{T'_{do}}(1))^{\frac{1}{2}} \qquad (8.2\text{-}4)$$

where $\rho_{T'_{do}}(1)$ is the observed lag-1 serial correlation coefficient of the deviations, ψ_t is the standard normal deviate, and $\sigma_{T'_{do}}$ is the standard deviation of the observed deviations in ^{o}C.



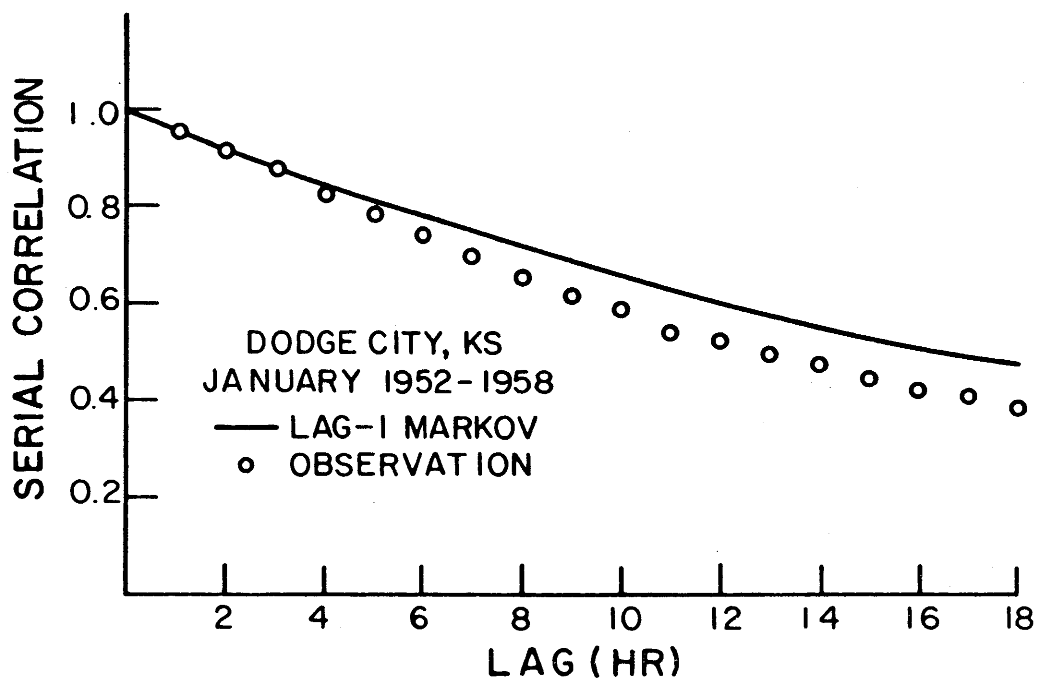


Figure 8.2 Serial Correlation of Hourly Dewpoint Deviations

The January dewpoint model can now be written as

$$T_d(t) = \tilde{T}_d(t) + T_d'(t) \tag{8.2-5}$$

8.3 Skewed Data

The dewpoint data tend to be negatively skewed. For example, the July data were found to have skew coefficients of -0.55 and -0.67 for Boston and Dodge City respectively. To be correct in modelling hourly dewpoints in July, the random deviate ψ_t should be modified according to the Thomas-Fiering approach described in Chapter 7 for wind data. The transformed random variate, ε_t was defined previously as

$$\varepsilon_t = \frac{2}{\gamma_\varepsilon}\left(1 + \frac{\gamma_\varepsilon \psi_t}{6} - \frac{\gamma_\varepsilon{}^2}{36}\right)^3 - \frac{2}{\gamma_\varepsilon} \tag{7.2-2}$$

where

$$\gamma_\varepsilon = \frac{(1 - \rho_d^3(1))}{(1 - \rho_d^2(1))^{1.5}} \gamma_d \tag{7.2-3}$$

where γ_d is the skew coefficient of the observed data.

This approach does not work well for dewpoint generation because the lag-1 serial correlation coefficients for dewpoints are very high. (0.96 for July in Boston and 0.95 for July in Dodge City). To see the problem more clearly, let us look at the modifier of γ_d in Equation 7.2-3 and call it F. Thus

$$F = \frac{(1 - \rho_d^3(1))}{(1 - \rho_d^2(1))^{1.5}} \tag{8.3-1}$$

Examination of Equation 8.3-1 shows that the denominator decays to zero faster than the numerator as ρ_d approaches one. Therefore, as $\rho_d(1) \to 1$, $F \to \infty$. The skew adjustment factor, F, is plotted against lag-1 correlation on Figure 8.3. Generally, when lag-1 correlation is less than about 0.9, there is no problem. But for lag-1 correlation values greater than 0.9, F gets very large. For example, for $\rho(1) = 0.95$, $F \simeq 4.7$.

To see the full impact of such an extreme adjustment factor, we must examine the last term of Equation 8.2-1 using ε_t from Equation 7.2-2 instead of the standard normal deviate ψ_t. On Figure 8.4 the value of $\varepsilon_t \sigma_d (1-\rho_d^2(1))^{\frac{1}{2}}$ is plotted against a wide range of values for ψ_t.

During the course of generating a large number of random standard normal deviates, ψ_t, a few values selected from the tails of the distribution are expected. If, for instance, a large negative value for ψ_t is selected, a very large value for the $\varepsilon_t \sigma_d (1 - \rho_d^2(1))^{\frac{1}{2}}$ term results. In this case, the last term of Equation 8.2-1 so dominates the output that very large and sudden negative shifts of dewpoint occur. From Figure 8.4, it is seen that negative shifts on the order of 10^0 to 14^0C are possible. If two or more large values of ψ_t happen to be generated close in succession, totally unrealistic sequences can be generated. Therefore, the Thomas-Fiering approach was not used for dewpoints. Instead, the process was approximated using normally distributed deviates. Because dewpoints are constrained by temperature, (i.e. $T_d \leq T$), some of the skew is recovered. In future studies, other ways of preserving dewpoint skewness should be examined.

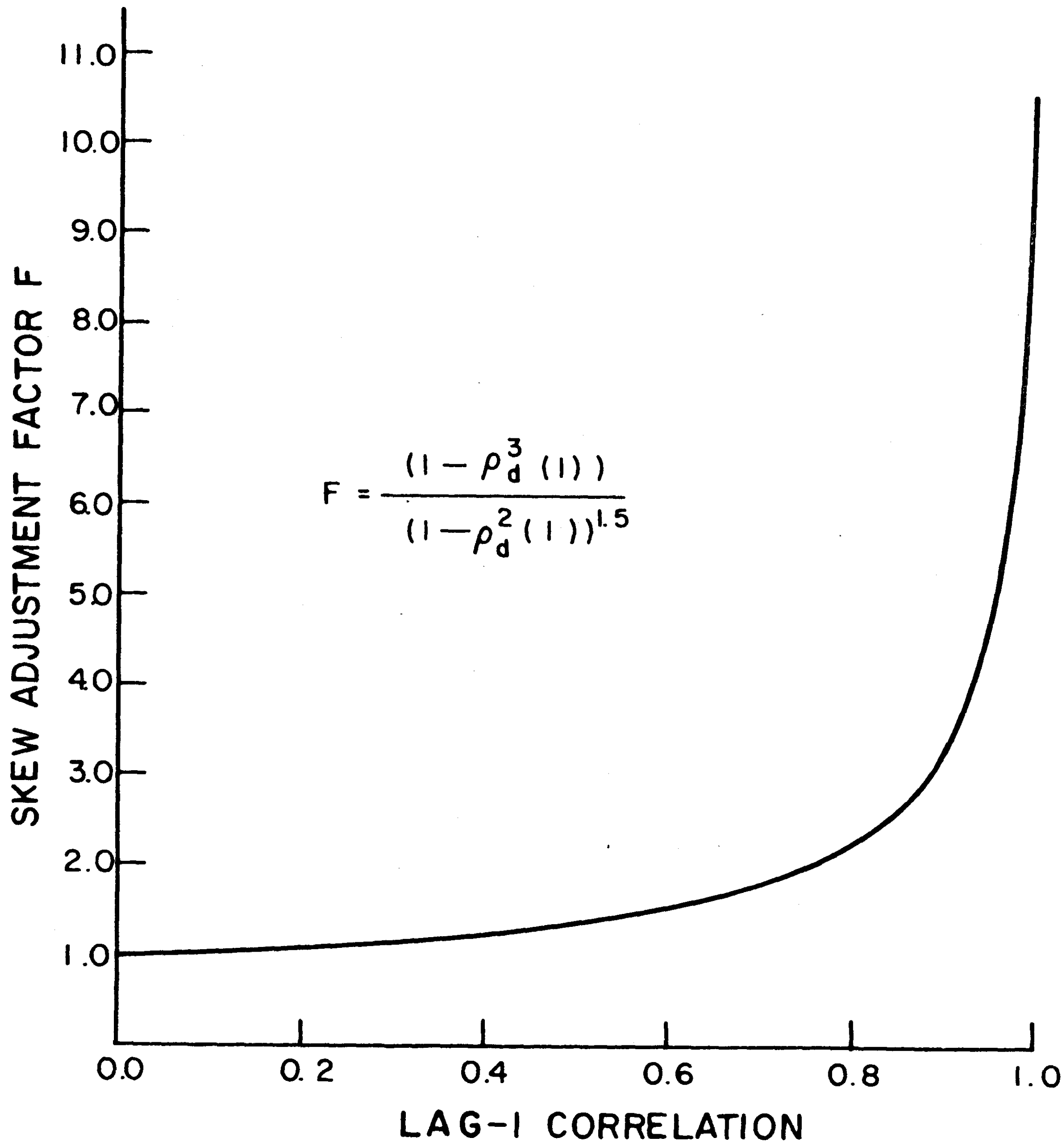


Figure 8.3 Skew Adjustment Factor

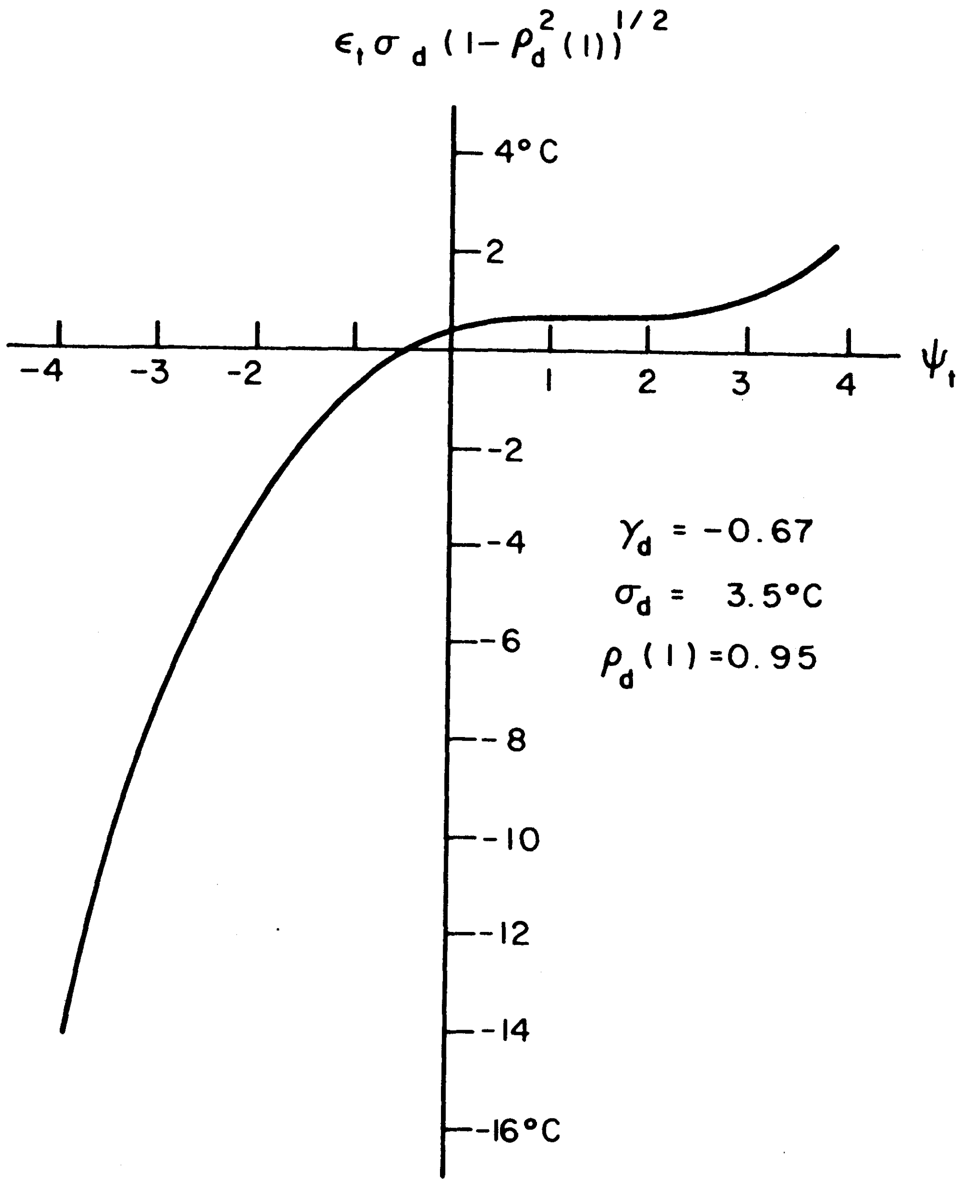


Figure 8.4 Variation of $\varepsilon_t\sigma_d(1 - \rho_d{}^2(1))^{\frac{1}{2}}$ with ψ_t

CHAPTER 9

PARAMETER ESTIMATION

9.1 Introduction

In Chapter 2 through Chapter 8 the individual components of the CSCS model were developed. However, the details of each required parameter estimation were not discussed. Rather, it seems more reasonable to treat the parameter estimation issues in a separate comprehensive chapter. Hopefully, future users of this report will find it more convenient to refer to a single chapter on parameter estimation instead of searching all chapters to seek the necessary information.

In the following sections, the procedures used to identify the parameters used in each component are described. A different set of parameters was derived for each month studied (i.e. January and July).

Hourly observations of rainfall, total opaque cloud cover, wind speed, wind direction, temperature, and dewpoint were obtained from the National Climate Center in Ashville, North Carolina for Boston, MA, Dodge City, KS and Phoenix, AZ. These locations were chosen to represent a variety of climatic and geographic conditions. Unfortunately, the Phoenix records had too many missing observations and the data set was not used in this study. However, adequate records were obtained for January (1949-1962) and July (1951-1963) at Boston and January (1952-1958) and July (1951-1957) at Dodge City.

For each location, data for each January (or July) were stripped from the master data file and combined to create "new" time series containing only January (or July) data. Parameters were then estimated from the January (or July) time series for each location.

9.2 Precipitation

The required parameters for the precipitation components include the mean time between storms, $\bar{t}_b$, in hours, the mean storm duration, $\bar{t}_r$, in hours, and the mean storm depth, $\bar{h}$, in mm. Calculation of the arithmetic mean values is obviously straightforward. The difficulty here lies in the assumptions used in developing the precipitation component, namely that successive storms are treated as independent events and that the times between storms follow an exponential distribution.

During times of precipitation activity, there may occur periods of no recorded precipitation. This is not unusual since a single synoptic scale disturbance can have multiple mesoscale precipitation events imbedded within it. Since the periods of precipitation emanate from systems evolving within a common parent some dependence is expected. As the times between recorded precipitation increases, casuality arguments suggest that this dependence decreases. The key then is to establish some minimum time between recorded precipitation that could be used to discriminate between "independent" storm events.

Restrepo and Eagleson (1982) studied long-term hourly precipitation records for six locations in the continental United States and found minimum times between recorded precipitation required for independence that ranged from 8 to 76 hours. In general, dry climates had high values for this minimum separation interval while humid climates were found to have lower values. Using a procedure outlined by the authors, the minimum separation intervals for Boston, MA and Dodge City, KS would be on the order of 13 hours and 47 hours respectively. Restrepo and Eagleson concede, however, that for precipitation models like the one used here, such a strict requirement on independence is operationally impractical and probably unnecessary.

If these long separation intervals were imposed, long storm durations would result and the storms would contain many periods without precipitation. This would produce unrealistically low average storm intensities. Restrepo and Eagleson (1982) suggest that a shorter separation criterion could be used operationally.

Grace and Eagleson (1967) found that a two hour separation interval was sufficient for identifying separate storms in New England under a sharply limited definition of independence. Using the same criterion Sariahmed and Kisiel (1968) found a three hour separation interval sufficient for an analysis of convective storms in Arizona. For this study, a two hour separation interval was used.

The parameter estimation procedure used in this study defined a storm duration to include the hours with recorded precipitation plus any non-precipitation separation intervals of two hours or less. Once the storms were defined then the appropriate mean storm durations, the mean times between storms, and the mean storm depths were determined by the usual techniques.

9.3 Cloud Cover

Cloud cover, as indicated in Chapter 3, is represented by a modulated non-stationary stochastic process composed of intra and inter-storm sequences. Parameter estimation for cloud cover during intra-storm periods is trivial since total cloud cover is assumed (i.e. $N(t) \equiv 1.0$). For inter-storm periods several parameters must be identified.

In Chapter 3, the existance of a stationary inter-storm "fairweather" cloud cover process was assumed. It was also assumed that the conditional mean and variance of the cloud process follow a smooth transition from their intra-storm values to their inter-storm "fairweather" values. Therefore, parameter estimation for the cloud cover process must include the following: 1) the identification of the appropriate fairweather sequences, 2) the estimation of the mean, variance, lag-1 serial correlation coefficient, and the frequency histogram of the fairweather cloud cover, and, 3) the decay coefficient for the transition period.

For convenience, the cloud cover model is rewritten here as

$$N(t) = M_0 + (1+M_0)(1-P(t)) + m(t)P(t) \qquad (9.3\text{-}1)$$

where M_0 is the fairweather mean cloud cover, $P(t)$ is the transition function, and $m(t)$ is a stationary sequence of serially correlated deviations. $P(t)$ and $m(t)$ are respectively defined as

$$P(t) = (1 - e^{-\gamma(t-t_0)})(1 - e^{-\gamma(t_0+t_b-t)}) \qquad (9.3\text{-}2)$$

$$m(t) = \rho_m(1)m(t-1) + \eta(t)(1-\rho_m{}^2(1))^{\frac{1}{2}} \qquad (9.3\text{-}3)$$

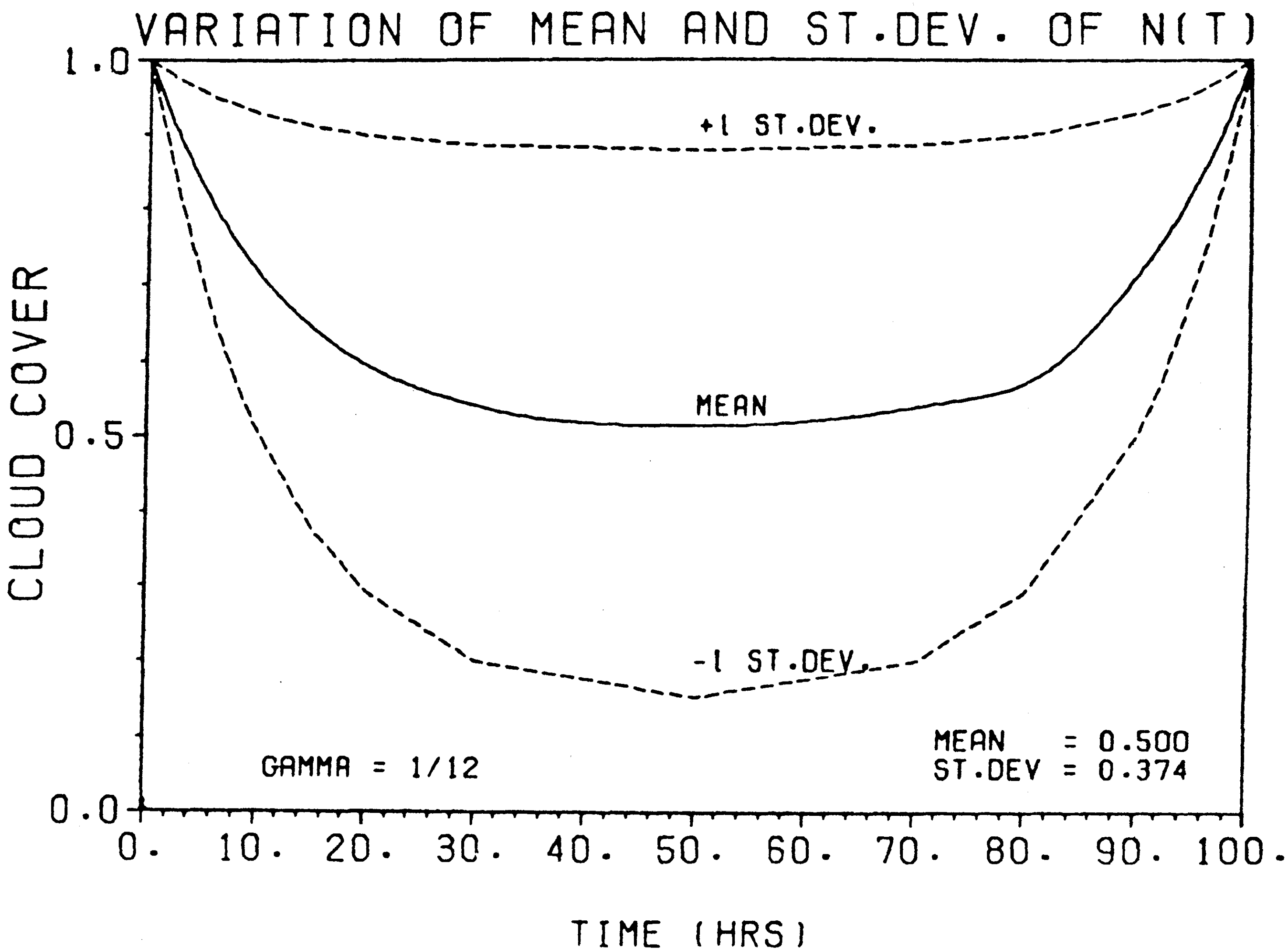


Figure 9.1 Variation of Mean and Standard Deviation of Cloud Cover During an Inter-Storm Period.

where γ is the transition decay coefficient in hr^{-1}, t_0 is the time of beginning of the inter-storm period in hr, t_b is the time between storms in hr, $\rho_m(1)$ is the lag-1 serial correlation coefficient of the fairweather cloud cover, $\eta(t)$ is a zero-mean random deviate with variance, σ_m^2, and σ_m^2 is the variance of the fairweather cloud cover.

The nature of the hypothesized transition of the cloud cover mean and variance is shown in Figure 9.1. In this example where $t_b = 100$ hr, the function describing the mean is U-shaped. The variance is represented by the trace of $\pm$ 1 standard deviation about the mean. The variance narrows to zero at each end and attains its maximum value in the middle as it follows the general curvature of the mean.

The values for the mean and variance that we are looking for are those that represent the stable or fairweather central region during the time between storms. In other words, we are interested in that region described by the bottom of the U-shaped functions shown in Figure 9.1.

To explain the procedure used to identify the fairweather sequences, it is best to again refer to Figure 9.1. Here we have an inter-storm period of 100 hours. If we calculate the mean cloud cover for the entire 100 hour period, we would get a value say $\overline{N}_{100}$. Next, if we eliminate two hours from <u>each</u> end of this 100 hour period and calculate a new mean for the remaining 96 values, we would get $\overline{N}_{96}$ where $\overline{N}_{96} < \overline{N}_{100}$ since some of the highest values of N were eliminated. If we continue to eliminate values at each end, the mean values will continue to decrease, although at a slower rate. When the mean value has stabilized, it is assumed that the fairweather sequence has been identified.

To handle the entire data set, the procedure is to first compute the mean value of cloud cover for all inter-storm periods. Then after successively eliminating values from both ends of the available inter-storm periods, new means are computed. Eventually after some T_r hours have been eliminated, the mean value stabilizes. The value T_r is the length of the transition period. Once T_r is established, the fairweather sequences contained in inter-storm periods of length greater than $2T_r$ are combined in a new time series containing only fairweather values. After the fairweather cloud cover time series has been determined, M_0, $\sigma_m{}^2$, $\rho_m(1)$, and the frequency histogram can be estimated by the traditional methods.

In general, the frequency histograms of the fair-weather cloud covers tend to be U-shaped with spikes at zero and one. Part of the reason for this result is that visual observations of both zero and one actually encompass broader ranges of causative atmospheric conditions than do the other observations (i.e. $N(t) = 0.1, 0.2, \ldots,$ 0.9, etc., see Chapter 3.6). This distortion causes peaks at zero and one that can be two to four times greater than the values obtained for the other levels of cloudiness. As a result the random variate generating scheme described in Appendix A becomes very inefficient.

In addition, the lag-1 Markov model (Equation 3.5-1) used to generate the fairweather cloud cover sequence preserves the first and second moments of the input distribution but does not necessarily preserve the distribution itself. For strongly peaked U-shaped input distributions, the tendancy is to produce output distributions that are more uniform (i.e. lower peaks and higher mid-ranges).

An example of a cloud cover histogram is presented in Table 9.1. Except for zero and one, all elements represent a cloud cover range of 0.10. Because cloud cover observations are bounded by zero and one, histogram elements for zero and one represent a range of only 0.05. To make the histogram a probability mass function, the magnitudes of the histogram elements for zero and one would have to be doubled to get the proper mass contribution for these elements. However, this would compound the peakedness problem discussed earlier.

Another alternative would be to expand the range of the representative histogram elements for zero and one. (Remember that the outcome of the cloud cover process is still constrained to be between zero and one). If these two ranges are expanded such that the resulting histogram elements take on values of the same order as the mid-range values, three positive results occur. First, the data generation efficiency roughly doubles. Second, the output histogram is less distorted and third, the broader causative atmospheric conditions are better represented. An example of the adjusted input histogram is shown in Table 9.2.

The remaining parameter to be estimated for the cloud cover model is the transition decay coefficient, γ. To estimate γ we can use the value found for the length of the transition period, T_r, during the identification of the fairweather sequence.

The transition function P(t), as shown in Equation 9.3-2, is a symmetric function. To examine the transition rate, we need only to look at one side of the function since for analysis purposes we can assume that t_b is large enough to eliminate the influence of the second side of the function.

Table 9.1 Observed Cloud Cover Histogram: July, Boston

RANGE	FREQUENCY (%)
$0.00 \leq N < 0.05$	32
$0.05 \leq N < 0.15$	11
$0.15 \leq N < 0.25$	12
$0.25 \leq N < 0.35$	10
$0.35 \leq N < 0.45$	7
$0.45 \leq N < 0.55$	3
$0.55 \leq N < 0.65$	3
$0.65 \leq N < 0.75$	3
$0.75 \leq N < 0.85$	4
$0.85 \leq N < 0.95$	3
$0.95 \leq N \leq 1.00$	11

Table 9.2 Adjusted Cloud Cover Histogram: July, Boston

RANGE	FREQUENCY (%)
$-0.25 \leq N < -0.15$	10
$-0.15 \leq N < -0.05$	11
$-0.05 \leq N < 0.05$	11
$0.05 \leq N < 0.15$	11
$0.15 \leq N < 0.25$	12
$0.25 \leq N < 0.35$	10
$0.35 \leq N < 0.45$	7
$0.45 \leq N < 0.55$	3
$0.55 \leq N < 0.65$	3
$0.65 \leq N < 0.75$	3
$0.75 \leq N < 0.85$	4
$0.85 \leq N < 0.95$	3
$0.95 \leq N < 1.05$	4
$1.05 \leq N < 1.15$	4
$1.15 \leq N \leq 1.25$	3

Thus, for convenience, the right-hand side of P(t) is ignored and Equation 9.3-2 can be rewritten (after setting the arbitrary initial time t_0 to zero) as

$$P(t) = (1 - e^{-\gamma t}) \tag{9.3-4}$$

According to the criterion established in Chapter 3, P(t) = 1.0 within the fairweather regime. But according to Equation 9.3-4 $P(t) \rightarrow 1.0$ as $t \rightarrow \infty$. This requirement is impractical operationally. However, this problem is overcome by simply choosing a value of P(t) that is sufficiently close to 1.0. Thus, for the present study, the fairweather regime exists for $P(t) \geq 0.99$. This definition of the beginning of the fairweather regime (i.e. when P(t) = 0.99) also implies that the length of the transition period, T_r, is equal to the time it takes P(t) to go from 0.00 to 0.99. Using P(t) = 0.99 and $t = T_r$, Equation 9.3-4 can be written as

$$0.99 = (1 - e^{-\gamma T_r}) \tag{9.3-5}$$

After rearranging and taking the natural logrithm of both sides of Equation 9.3-5, and solving for γ gives

$$\gamma = \frac{1}{T_r} \ln (0.01) \tag{9.3-6}$$

or

$$\gamma = \frac{4.61}{T_r} \tag{9.3-7}$$

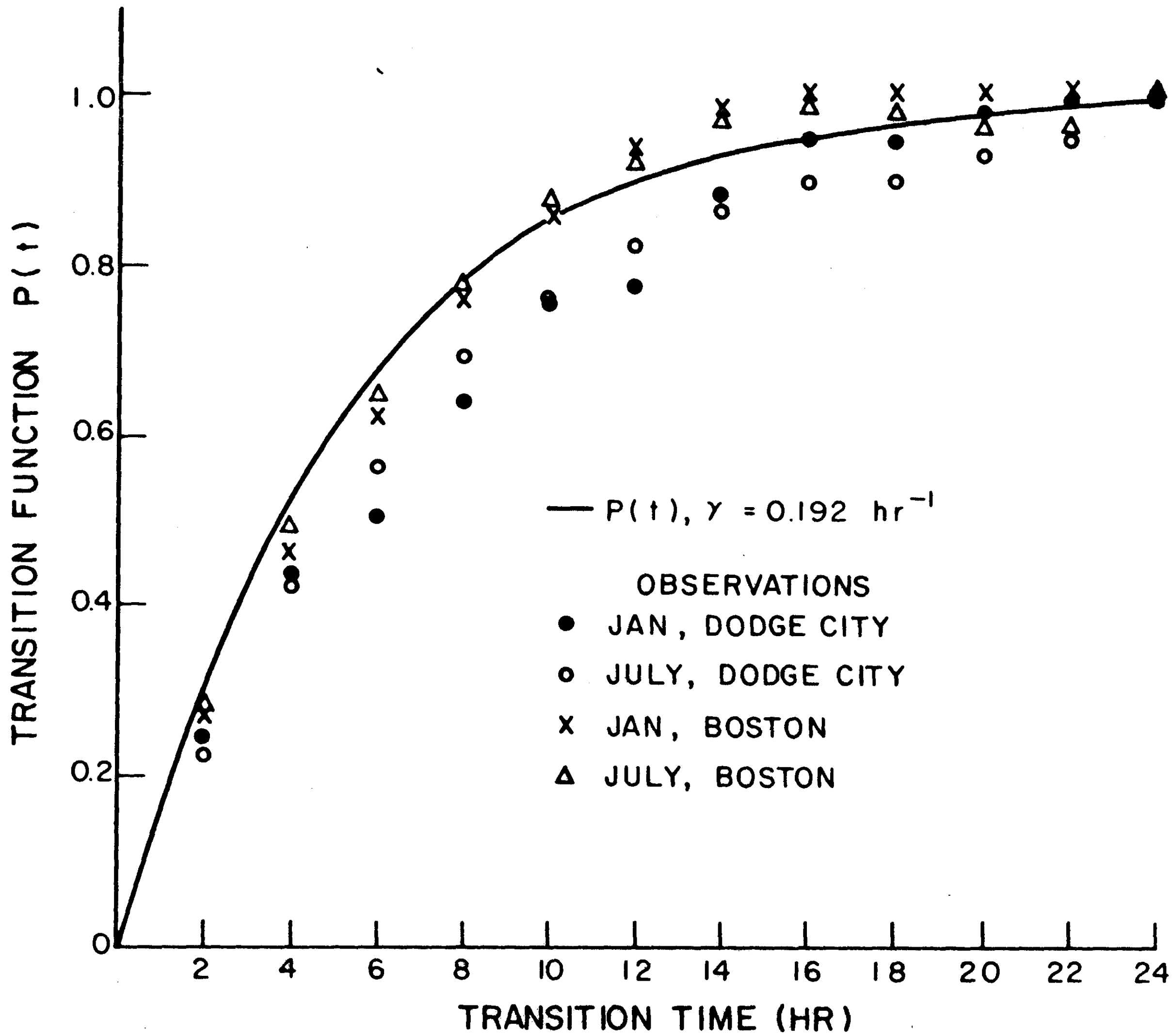


Figure 9.2 Comparison of Observed and Theoretical Transition Function

Table 9.3 Mean Cloud Cover Transition

T (hr)	Dodge City January	Dodge City July	Boston January	Boston July
0	0.367	0.333	0.572	0.432
2	0.354	0.306	0.534	0.403
4	0.344	0.282	0.506	0.379
6	0.340	0.265	0.483	0.362
8	0.333	0.250	0.464	0.348
10	0.327	0.242	0.449	0.338
12	0.326	0.234	0.439	0.332
14	0.320	0.229	0.432	0.327
16	0.317	0.225	0.429	0.325
18	0.317	0.225	0.425	0.326
20	0.315	0.221	0.424	0.328
22	0.314	0.219	0.425	0.328
24	0.314	0.213	0.430	0.324

Thus, by knowing the length of the transition period, T_r, that was used to identify the fairweather regime earlier, the transition decay coefficient can be estimated easily.

Another interesting way to look at the transition is worth noting. The transition can be observed by studying the rate by which the mean cloud cover varies from its value for all inter-storm periods to its fairweather values. In normalized form, the "observed" transition can be expressed by

$$P_0(t) = \frac{\overline{N}_a - \overline{N}_t}{\overline{N}_a - M_0} \tag{9.3-8}$$

where $\overline{N}_a$ is the mean cloud cover for all inter-storm periods. (This corresponds to $\overline{N}_{100}$ in the earlier example), M_0 is the fairweather mean, $\overline{N}_t$ is the mean cloud cover for an intermediate region.

The value of Equation 9.3-8 is that we can now plot observed data to see the smooth transition hypothesized in Chapter 3. Figure 9.2 shows the observed values of $P_0(t)$ for the four data sets used in this study. Based on the observed values for $\overline{N}(t)$ shown in Table 9.3, 24 hours was judged to be a reasonable value for the length of the transition, T_r. This value was used in computing the $P_0(t)$'s shown in Figure 9.2 and in Equation 9.3-7 to determine γ for the hypothesized transition function P(t) which is also plotted in Figure 9.2. The transition function, P(t), represents the overall shape of the observed transitions quite well. However, the theoretical curve appears to fit the Boston observations slightly better than for Dodge City. The Dodge City transitions are slightly slower than Boston's.

9.4 Shortwave and Longwave Radiation

As shown in Chapter 4, shortwave radiation is computed by

$$I_c = I_0 \exp(-na_1 m) \tag{9.4-1}$$

and

$$I_s' = I_c(1 - 0.65N^2) \tag{9.4-2}$$

The variables in Equation 9.4-1 and 9.4-2 have been defined earlier in Chapter 4. The only variable that must be subjectively selected prior to simulation is the turbidity factor, n, which was indicated to vary from about 2.0 for clear air to about 5.0 for smoggy urban air. Because no prior information was available to make anything more than a subjective decision regarding the value of n, its value was set to 2.0 for both Boston and Dodge City.

For longwave radiation, we have

$$R_a = (0.70 + 5.95(10^{-5})e_o e^{(1500/T)})(1 + 0.17N^2)\sigma T^4 \tag{9.4-3}$$

where the principal variables, e_o, T, and N are generated by the CSCS model. No other parameters are required by the longwave component.

9.5 Temperature

The temperature model requires the estimation of several regression coefficients, b_i, for the "deterministic" portion along with the variance and the lag-1 serial correlation coefficient of a superimposed deviations process. Since the methods used to estimate the parameters of the temperature model were an integral part of the model development detailed in Chapter 6 and Appendices C and D, they need not be discussed again here.

9.6 Wind Speed and Wind Direction

Wind speed and wind direction are both generated independently by lag-1 Markov models. The wind speed model requires as input the mean, the variance, the lag-1 serial correlation coefficient and the skew coefficient of the observed wind speeds. The wind direction model requires the mean, the variance, the lag-1 serial correlation coefficient, and the frequency histogram of the observed wind directions. All parameters are estimated by the traditional methods.

9.7 Dewpoint

Two methods have been employed to generate dewpoints depending upon the circumstances. The first method generates dewpoints independently using a lag-1 Markov model and requires the mean, the variance, and the serial correlation coefficient of the observed dewpoints. These parameters are estimated by the usual techniques.

The second option available to generate dewpoints uses a linear regression model with a superimposed deviations process. The coefficients of the regression model are estimated by standard regression methods. The deviations process is again modelled by a lag-1 Markov approach which requires the variance and the lag-1 serial correlation of the observed deviations. The regression model and the method used to determine the observed deviations are discussed in detail in Chapter 8.

9.8 Summary

The parameters required by the CSCS model that are estimated from the observed data are summarized as follows:

Precipitation

- mean time between storms
- mean storm duration
- mean storm depth

Cloud Cover

- fairweather mean
- fairweather variance
- fairweather lag-1 serial correlation
- fairweather frequency histogram
- transition decay coefficient

Temperature

- regression coefficients
- deviations variance
- deviations lag-1 serial correlation

Wind Speed

- mean
- variance
- lag-1 serial correlation
- skew

Wind Direction

- mean
- variance
- lag-1 serial correlation
- frequency histogram

Dewpoint

- mean
- variance
- lag-1 serial correlation

  or

- regression coefficients
- deviations variance
- deviations lag-1 serial correlation

CHAPTER 10
CSCS RESULTS

10.1 Introduction

After estimating the parameters as described in Chapter 9, January and July data sets were generated by the CSCS model for both Dodge City, KS and Boston, MA. Three different aspects of the output will be reviewed. First, plots of the hourly data values generated by the model will be examined to see at least qualitatively that the various output elements are coordinated. Second, model output statistics will be presented to determine how well the observed statistics are reproduced. Third, the mean diurnal curves of generated temperatures and dewpoints will be compared to their observed counterparts.

10.2 Generated Data Plots

Figures 10.1-10.11 each represent three-day segments of the generated data sets. Presentation of hourly plots for the entire simulation period is obviously impractical due to space limitations. The selected three-day segments will be sufficient for demonstration purposes.

Looking first at Figures 10.1-10.2 for January in Dodge City, KS, we have plots of hourly temperature in ^{o}C, hourly dewpoint in ^{o}C, hourly shortwave radiation in langleys (ly), hourly longwave radiation in langleys, hourly cloud cover in tenths, and hourly precipitation in mm. Perhaps the most dominant features of these plots are the obvious diurnal structures of shortwave radiation and temperature.

Beginning with shortwave radiation, the generated hourly values are zero through the night as they should be. At sunrise, solar radiation starts its steady increase to its peak around noon. After the peak at solar noon, shortwave radiation decreases to zero again at sunset.

Shortwave radiation is dramatically affected by the presence of cloud cover. This is seen clearly by comparing the shortwave radiation curves for the two cloudy days (1/19, 1/20) and the mostly sunny day (1/21) in Figure 10.1. The peak solar radiation value on 1/21 was approximately 38 ly when cloud cover was 0.1. This compares to a peak of approximately 14 ly on 1/19 when cloud cover was 1.0. This also represents the 65% reduction of shortwave radiation due to total cloud cover that is dictated by Equation 4.4-3.

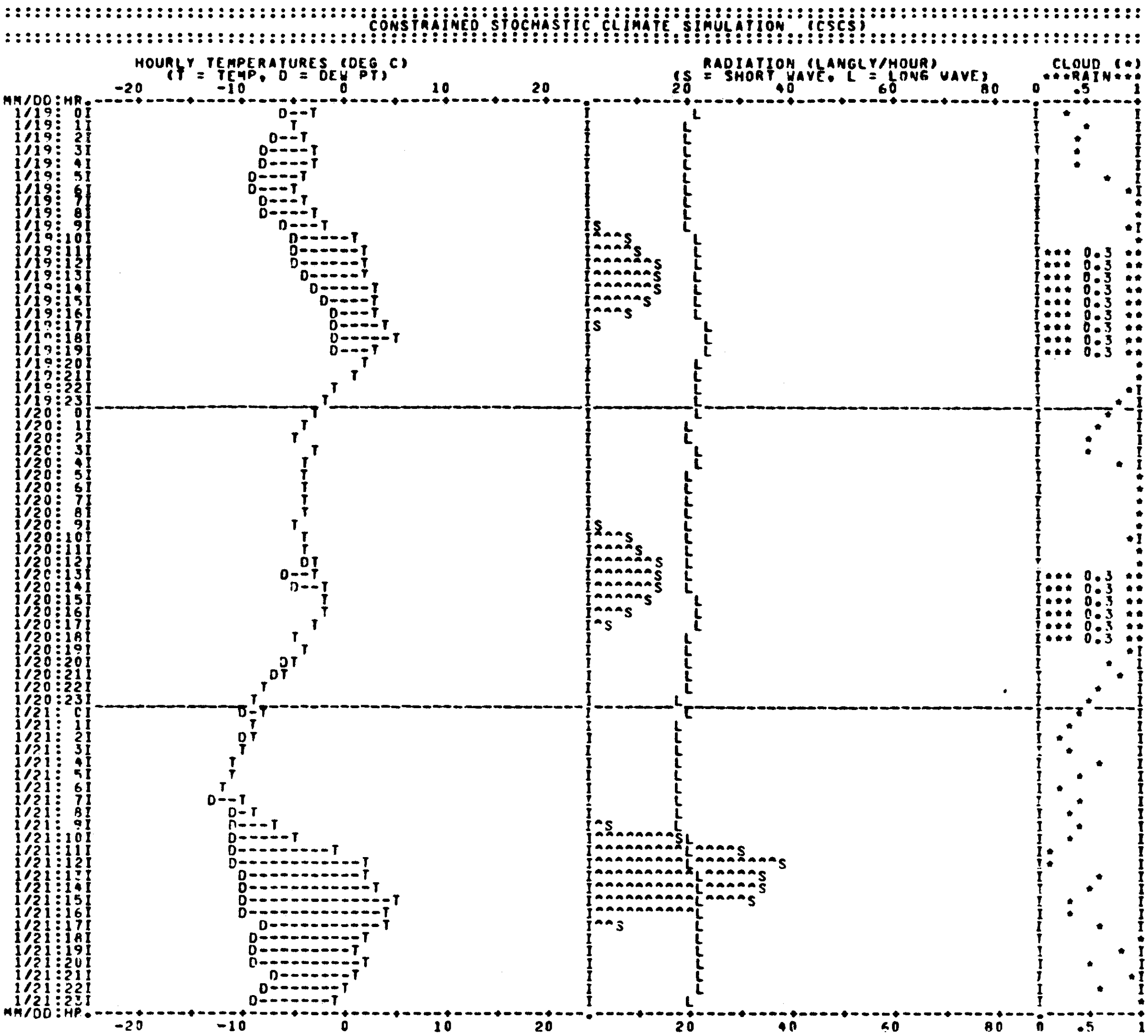


Figure 10.1 CSCS Output: January, Dodge City

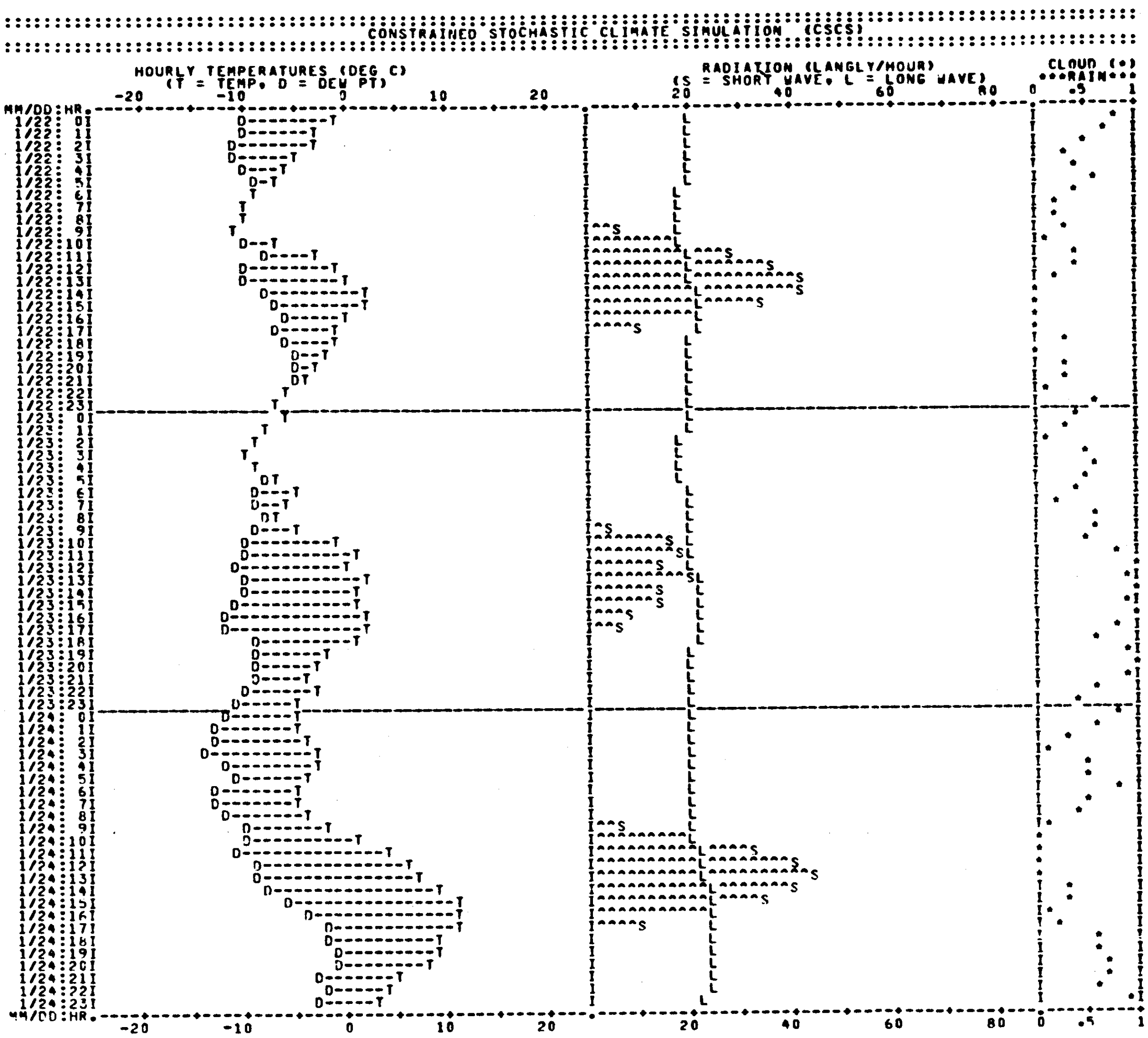


Figure 10.2 CSCS Output: January, Dodge City

The trace of hourly temperature also shows a strong diurnal signature. In general, minimum temperatures occurred in the early morning hours near sunrise and maximum temperatures occurred in mid to late afternoon. However, just as an observed temperature trace can deviate significantly from its characteristic diurnal curve, the CSCS model is capable of generating temperature traces for particular days that lack the characteristic diurnal signature. Witness day 1/20 in Figure 10.1. For the first 16 to 18 hours of this stormy day, the temperature curve stayed relatively flat. This is especially interesting when compared to the temperature curve of day 1/19 which was also stormy. In both cases the radiation inputs were at minimum values yet the temperatures of day 1/19 are substantially higher than on 1/20 and follow a more characteristic curve. This behavior of the CSCS model is explained by the stochastic component in the temperature scheme. On day 1/20, the stochastic components were apparently negative which served to counter the positive influence of the radiation input and to stabilize the temperature. The CSCS model has the capacity to generate a wide range of daily temperature patterns, making for a more natural appearing long-term trace of generated temperatures.

Figures 10.3-10.5 show segments of the generated data for July at Dodge City. Immediately, the increase in generated shortwave radiation over that of January is apparent, not only in magnitude, but in hours of sunshine as well. Peak shortwave radiation values at Dodge City increased from approximately 40 ly hr^{-1} in January to about 86 ly hr^{-1} in July. In addition, the number of hours of significant shortwave radiation (i.e. $I_s' > 1.0$ ly hr^{-1}) increased from about 9 hours in January to about 14 hours in July.

The cloud cover transitions into and out of storm periods can be seen in Figures 10.3 to 10.5. In Figure 10.3, cloud cover increases steadily in anticipation of the first storm on day 7/4. After the first storm, the cloud cover remains high due to the close proximity of a second storm. Once the second storm passes, the cloud deck breaks up and clears for day 7/5 before building again for the approaching storm on day 7/6.

In Figure 10.5, we see a short intense storm preceded by and followed by periods with little cloudiness. It is significant to note here that although a storm occurred on day 7/8 (Figure 10.5), the total shortwave radiation was only slightly reduced. The storm occurred before sunrise

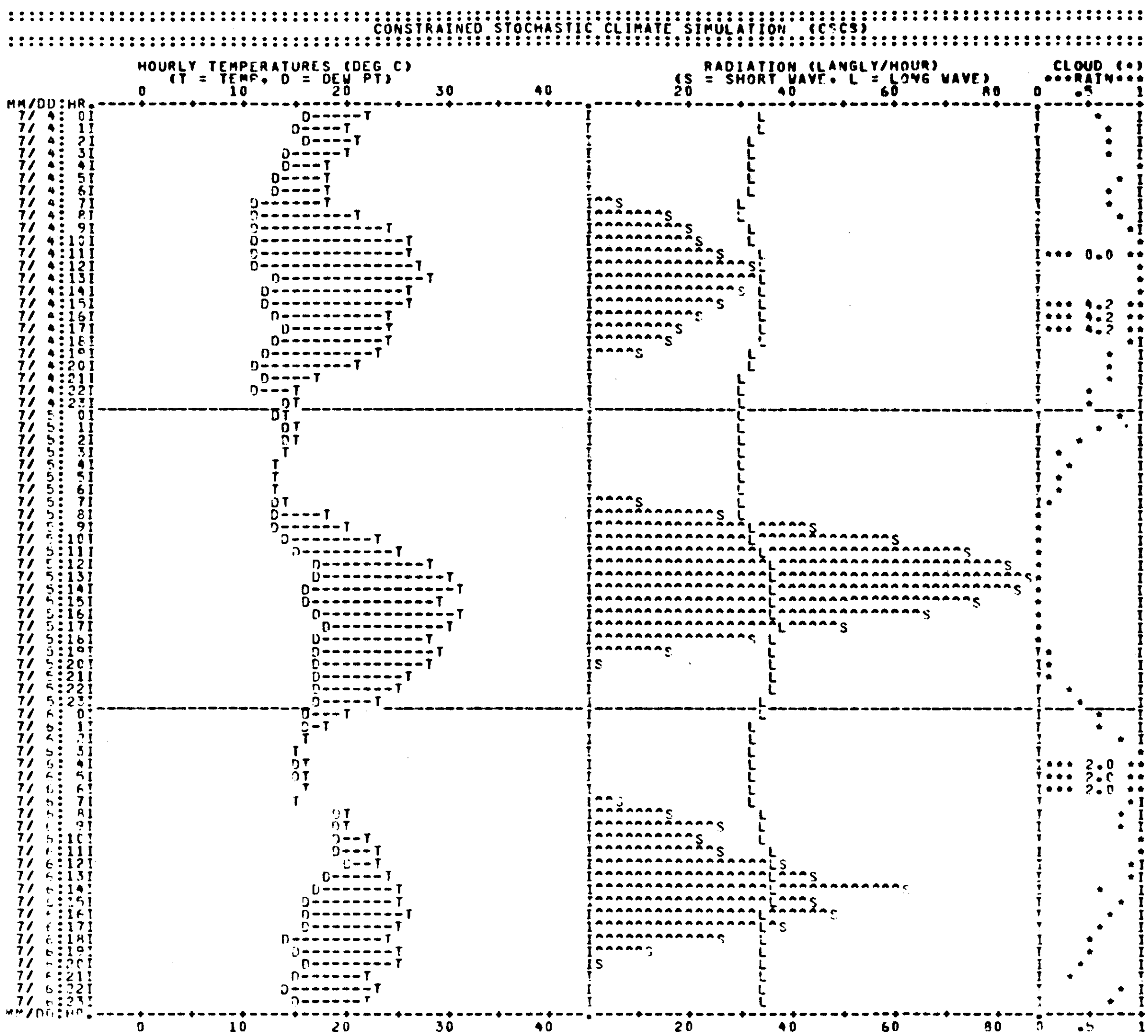


Figure 10.3 CSCS Output: July, Dodge City

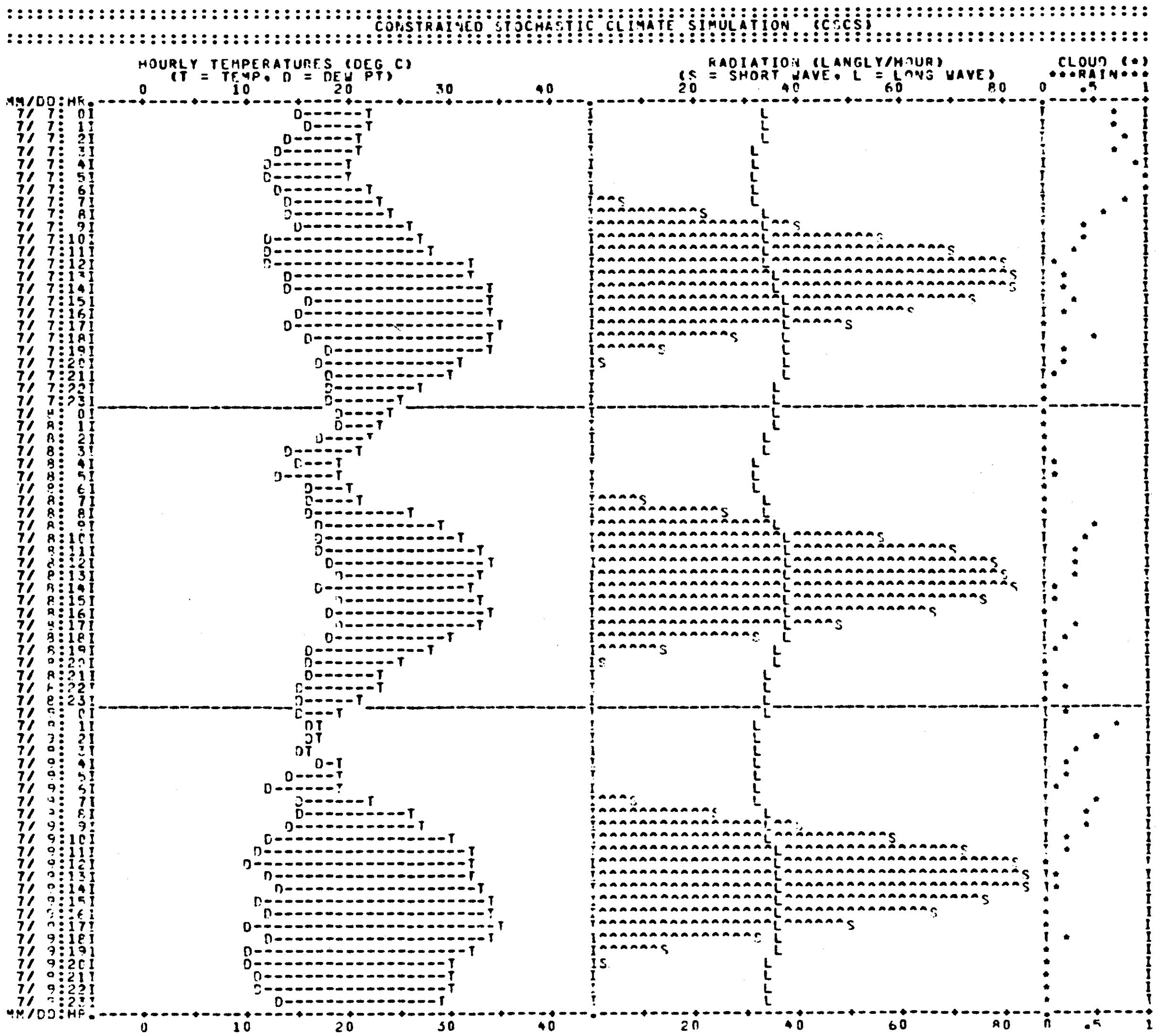


Figure 10.4 CSCS Output: July, Dodge City

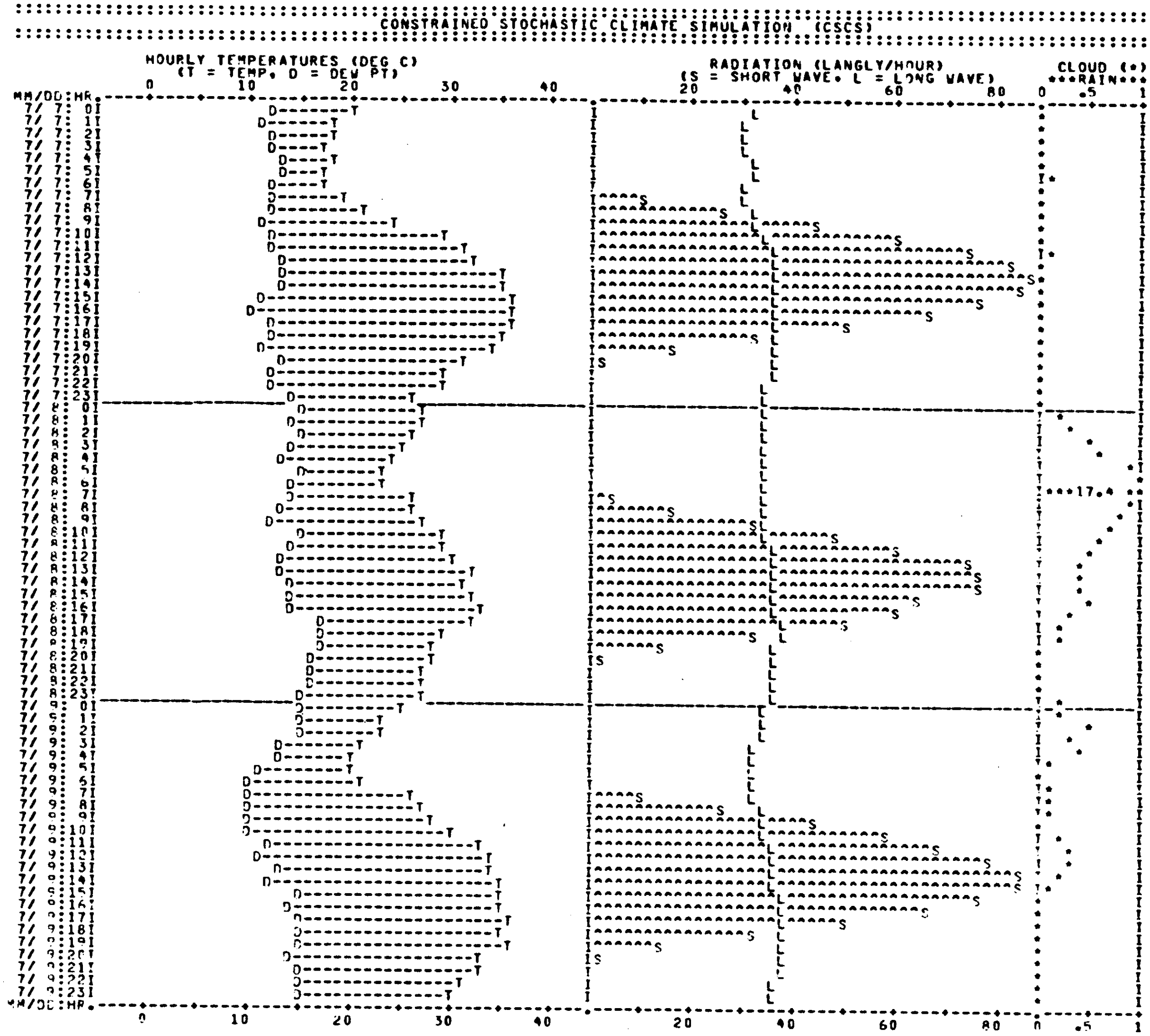


Figure 10.5 CSCS Output: July, Dodge City

and the cloud deck decayed quickly to minimize the impact on shortwave radiation. Contrast this result to that of day 7/4 when the storms occurred during the day and to that of day 7/6 when the storm occurred before sunrise but the cloudiness remained through the day. This behavior of the CSCS model is a significant improvement over previous models that implied specific reductions of shortwave radiation for stormy days regardless of when the precipitation occurred.

Figures 10.6 - 10.8 show segments of data generated for January in Boston, MA. As expected, low values for shortwave radiation are generated. Although the number of hours with significant shortwave radiation is the same as for January in Dodge City, KS, the peak values are slightly lower. Shortwave radiation peaks of about 40 ly hr^{-1} were generated for Dodge City but 36 ly hr^{-1} was the maximum value generated for Boston in January. The reduction is explained by the difference in the latitudes of the two sites since the same atmospheric attenuation parameters were used in both cases. Boston is located at $42^{o}22'$ N while Dodge City is located at $37^{o}46'$ N.

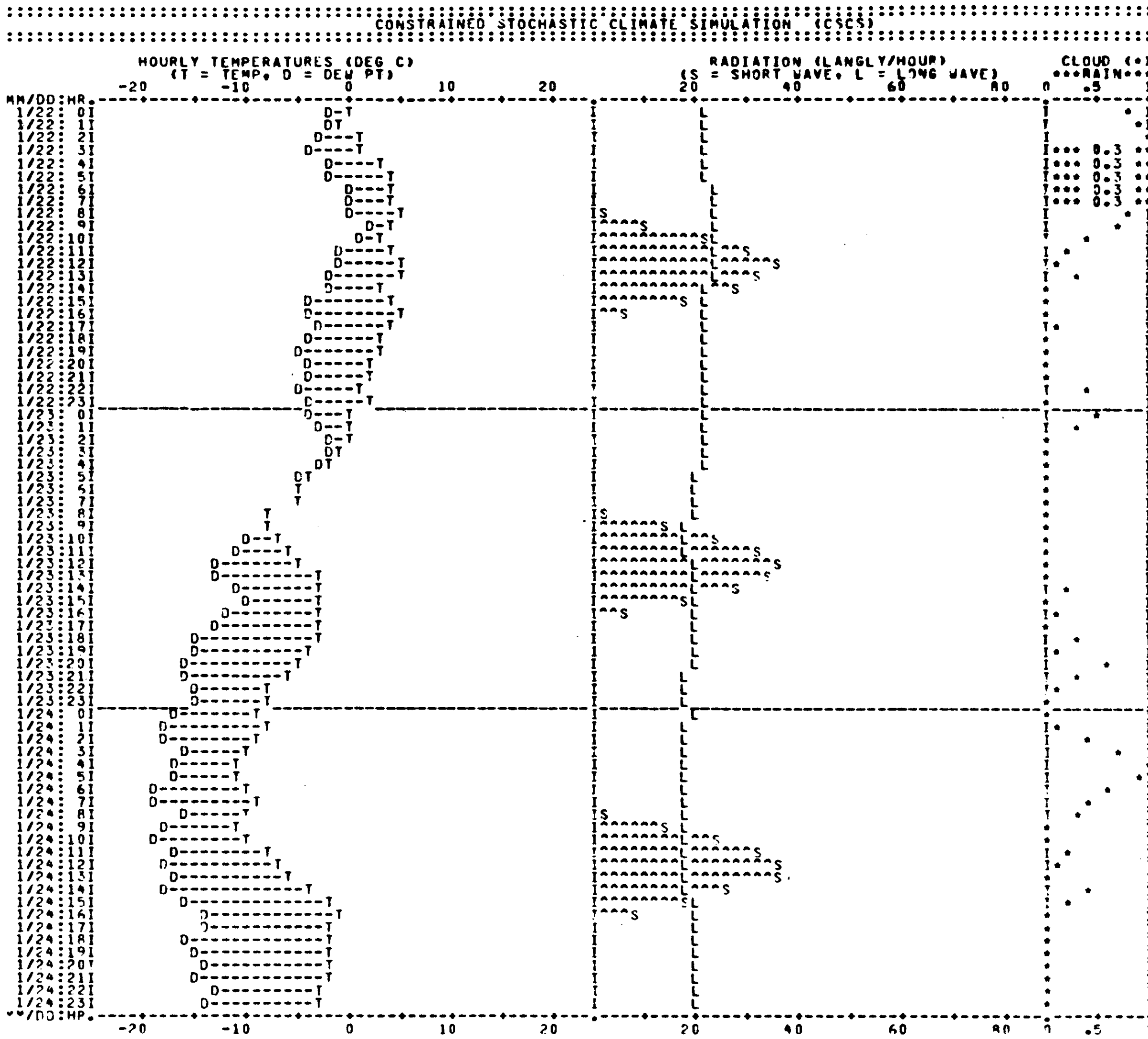


Figure 10.6 CSCS Output: January, Boston

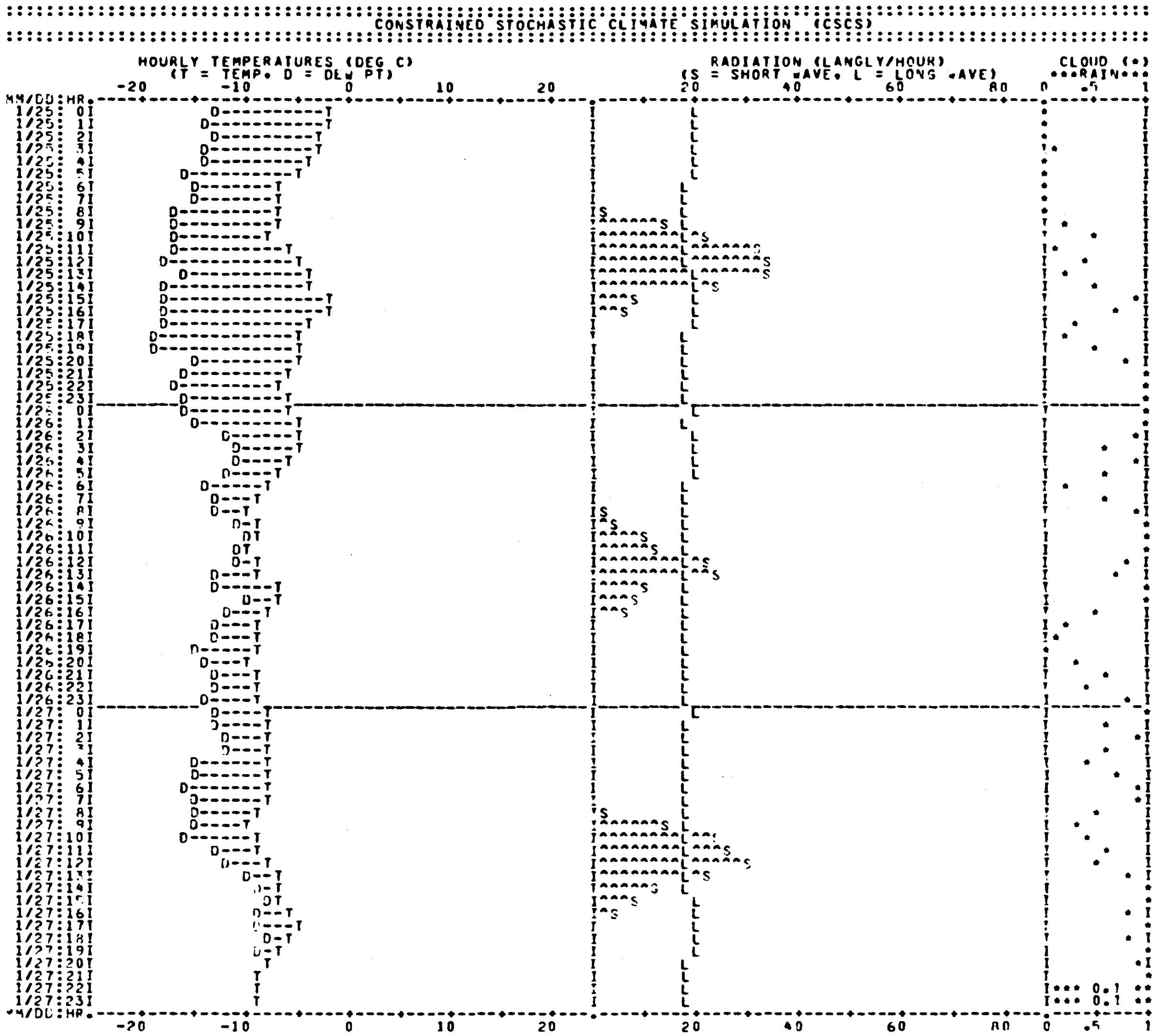


Figure 10.7 CSCS Output: January, Boston

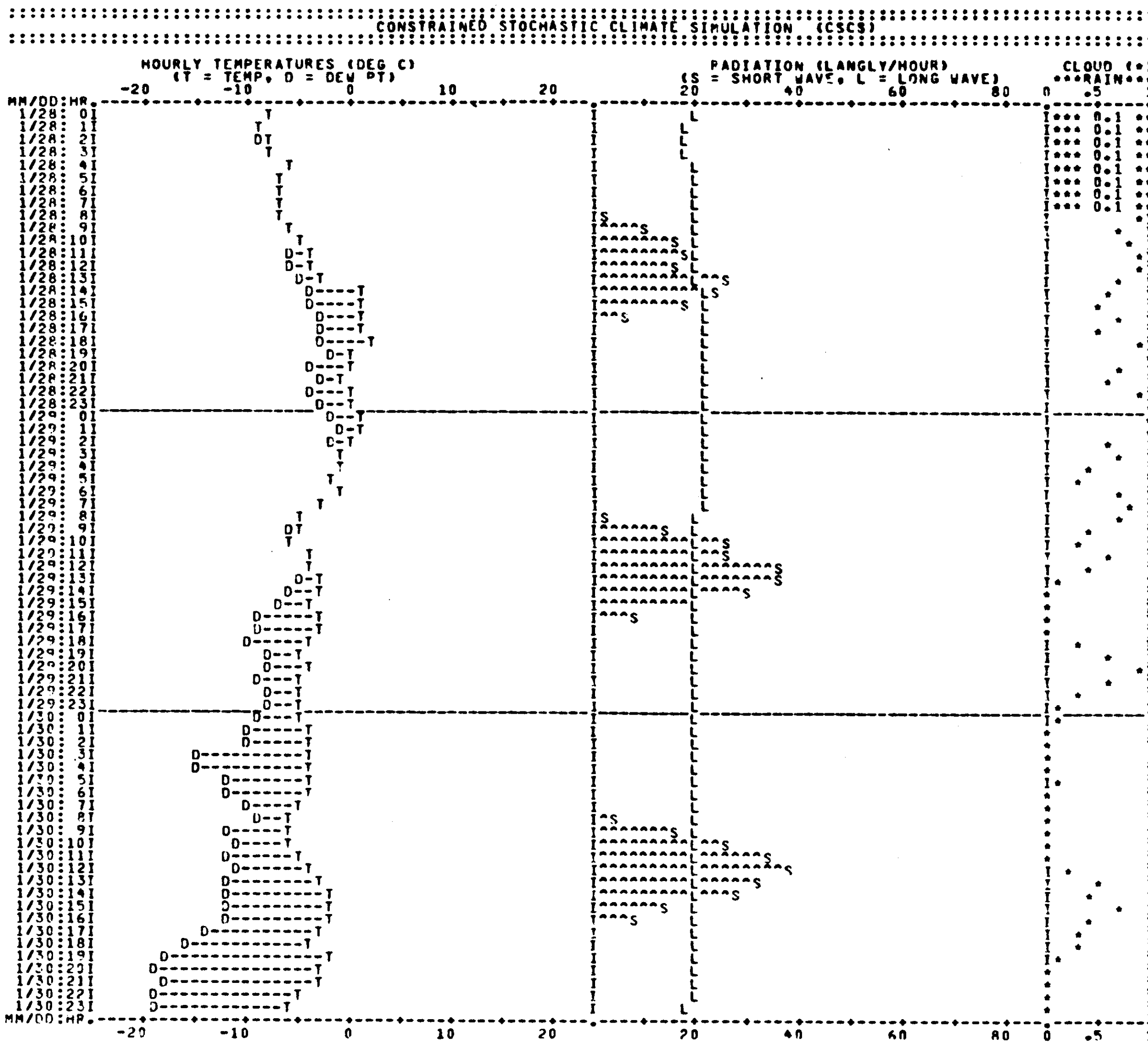


Figure 10.8 CSCS Output: January, Boston

The characteristic diurnal temperature curve is not as strong for January in Boston as it is for the other examples. Looking ahead to Figure 10.12 shows that the difference between the average minimum and maximum hourly temperatures is only about $5^{o}C$ for Boston, compared to about $10^{o}C$ for Dodge City (Figure 10.14).

The temperatures generated by the CSCS model for January in Boston appropriately do not exhibit a strong diurnal signature. This is especially true for days 1/26 - 1/29 in Figures 10.7 and 10.8.

It is also interesting to note the general downward trend from a maximum of $+5^{o}C$ on day 1/22 (Figure 10.6) to temperatures in the -6^{o} to $-3^{o}C$ range on day 1/24. This is consistent with the movement of large synoptic-scale weather systems through the region.

Longwave radiation also shows a general downward trend during the period 1/22 - 1/24. This is the result that should be expected with a general drop in atmospheric temperature and dewpoint.

Figures 10.9 - 10.11 show the segments of data for July in Boston, MA. Again, the notable increases in shortwave radiation and temperature over January levels are evident. Although the diurnal signature of the July temp-

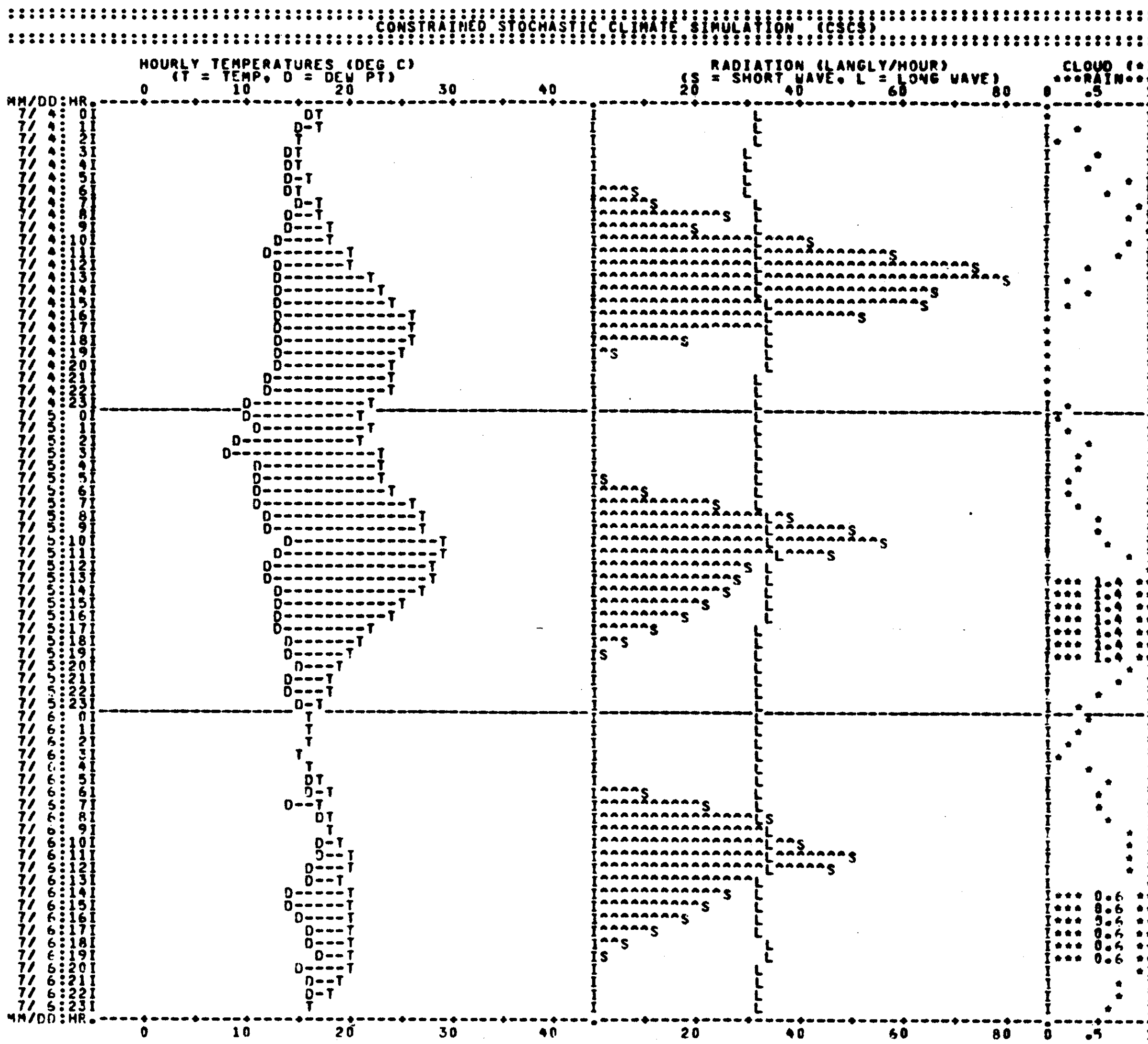


Figure 10.9 CSCS Output: July, Boston

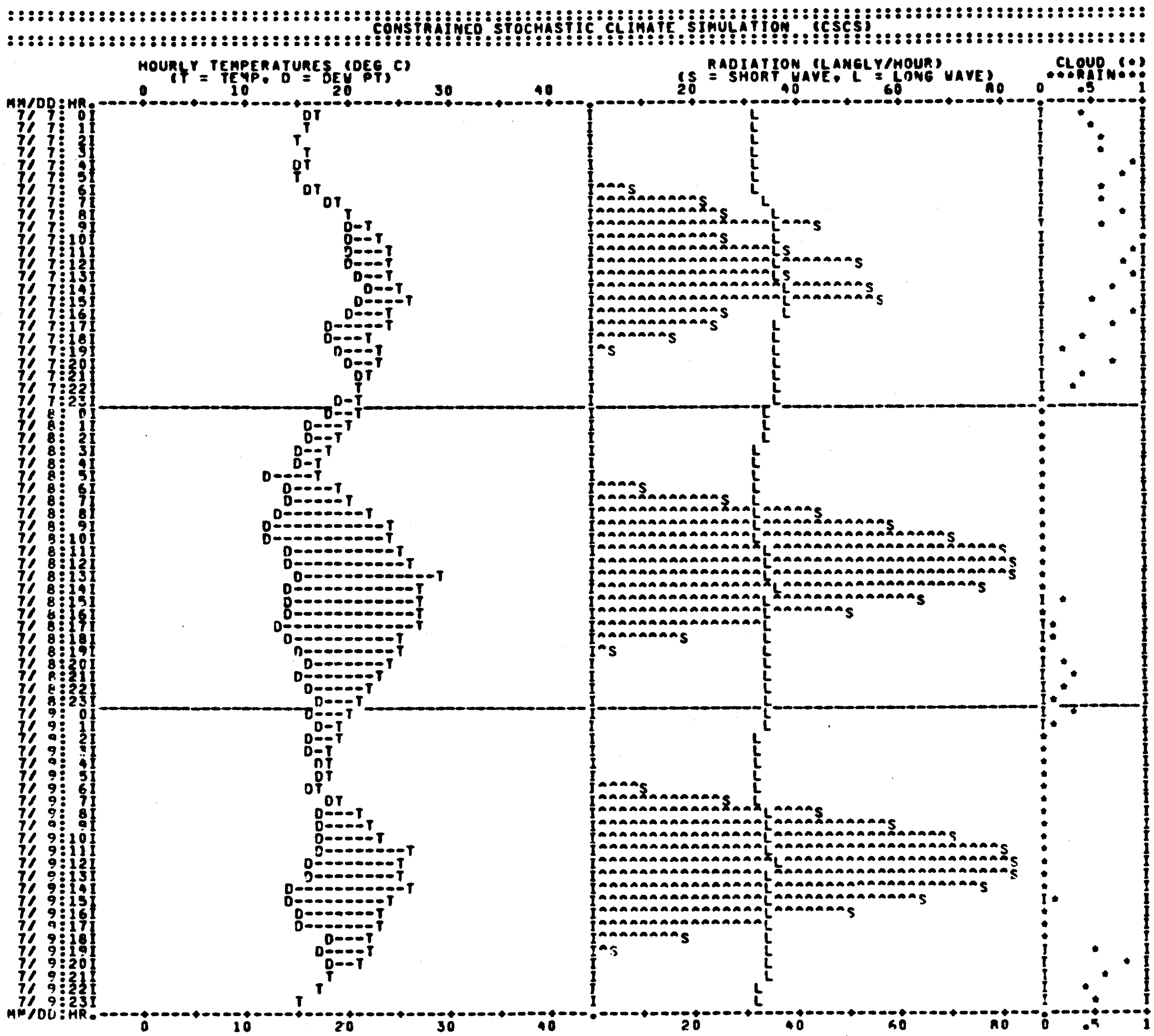


Figure 10.10 CSCS Output: July, Boston

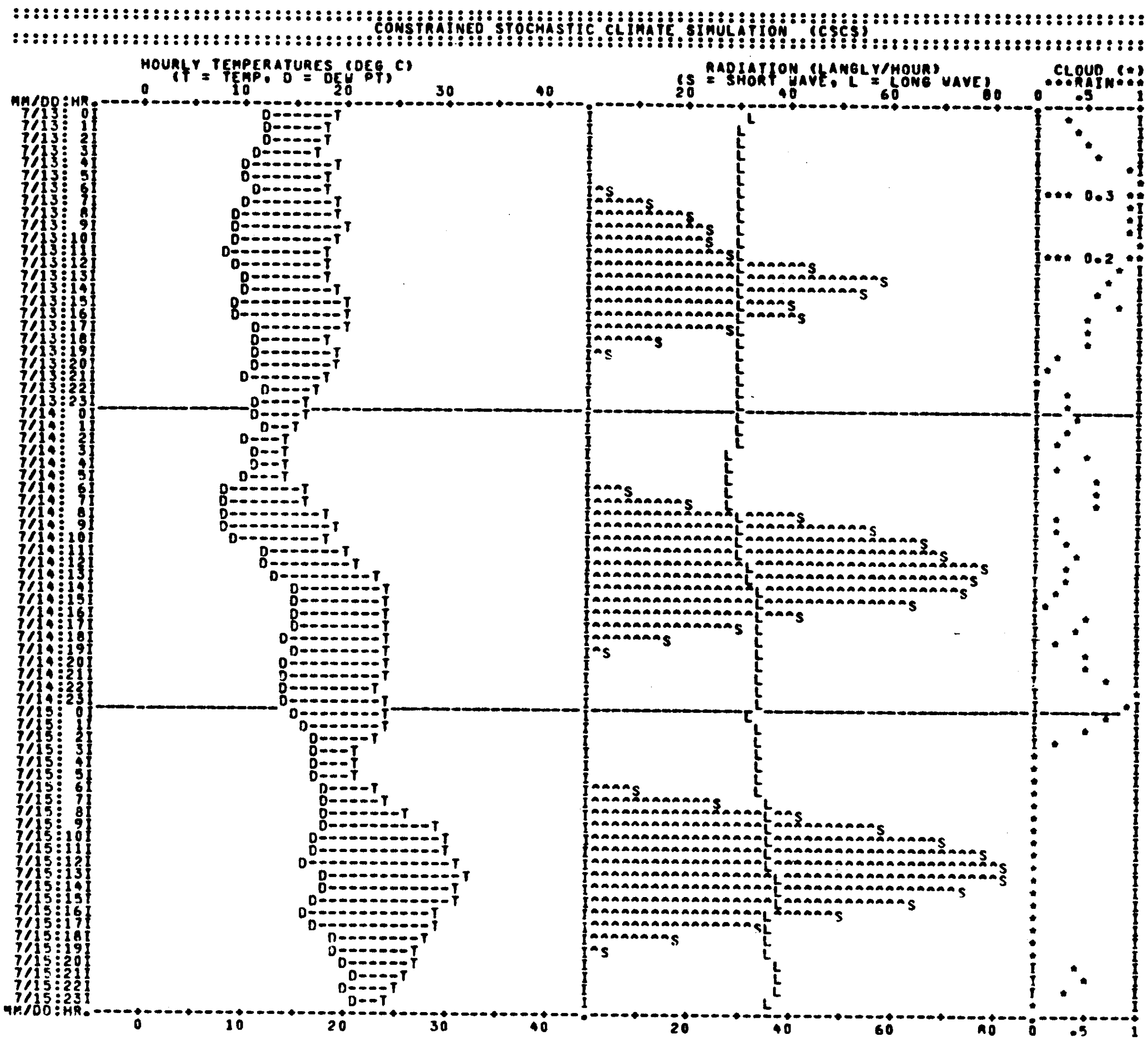


Figure 10.11 CSCS Output: July, Boston

eratures is strong, cloudiness coupled with a negative stochastic element can flatten the temperature curve for short periods of time (see day 7/6, Figure 10.9 and day 7/13, Figure 10.11). Expected downward trends in temperature are also occasionally countered by a positive stochastic component as evidenced by the temperature pattern during the evening hours of day 7/14 (see Figure 10.11).

Although visual examination of various segments of CSCS model output does not constitute a rigorous verification, it does provide a framework for a qualitative interpretation of model component coordination. In this respect, the CSCS model seems to be working properly. That is, cloud cover impacts shortwave radiation, shortwave radiation affects temperature, cloud cover is total during storms, etc. These effects might not be apparent from an analysis of model output statistics alone. The next step is, however, to verify that the model is working well statistically.

10.3 CSCS Model Output Statistics

Tables 10.1 - 10.4 contain the statistics of the model output and the statistics of the observed data for comparison. The generated data sets used in the statistical analysis are each 20 months in length. (i.e. 20 July's, 20 January's, etc.) Thus, 620 days or 14,880 hours of data were generated and analyzed for each experiment.

For temperature, dewpoint, cloud cover, wind speed, and wind direction, the means, standard deviations, and lag-1 serial correlation coefficients were computed. Since the observed skew coefficients were used in the wind speed component, the skew coefficients of the generated wind speeds were also computed. For the precipitation analysis, the mean times between storms, the mean storm durations, and the mean storm depths were computed. Observations of hourly shortwave and longwave radiation were not available for the periods of record used in this study. However, Getz and Nicholas (1979) provide estimates of mean daily shortwave radiation by climatic week based on data for the period 1952-1975. The estimated mean daily shortwave radiation was found from Getz and Nicholas by averaging the radiation values for the climatic weeks that span January and July.

Examination of Tables 10.1 - 10.4 shows that the statistics of the CSCS output compare favorably with the observed statistics in each case. The means and standard deviations of the respective generated temperatures and

Table 10.1 Output Statistics: January, Dodge City, KS

	TMP	DEW	CLD	WSP	WDR
MEAN	0.3°C (0.0)*	-6.5°C (-6.8)	0.41 (0.38)	5.5 m/s (5.5)	90.5° (86.3)
STANDARD DEVIATION	7.3°C (7.4)	5.4°C (5.7)	0.35 (0.41)	2.3 m/s (2.4)	43.9° (59.2)
LAG-1	0.98 (0.98)	0.97 (0.98)	0.87 (0.91)	0.87 (0.86)	0.89 (0.92)
SKEW	--- ---	--- ---	--- ---	0.55 (0.54)	--- ---

PRECIPITATION

	t_b	t_r	h
MEAN	207.7 hr (184.8)	4.9 hr (4.8)	2.2 mm (2.3)

RADIATION

	SWR	LWR
MEAN	190 ly/d (228)	507 ly/d ---

*( ) denotes observed value .

Table 10.2 Output Statistics: July, Dodge City, KS

	TMP	DEW	CLD	WSP	WDR
MEAN	26.3°C (26.8)*	15.5°C (15.4)	0.33 (0.34)	5.3 m/s (5.6)	112.0° (129.5)
STANDARD DEVIATION	5.5°C (5.5)	3.5°C (3.5)	0.32 (0.36)	2.1 m/s (2.3)	41.9° (50.3)
LAG-1	0.96 (0.96)	0.94 (0.95)	0.90 (0.89)	0.77 (0.78)	0.81 (0.84)
SKEW	--- ---	--- ---	--- ---	0.54 (0.51)	--- ---

PRECIPITATION

	t_b	t_r	h
MEAN	63.4 hr (66.9)	2.5 hr (2.5)	6.0 mm (6.1)

RADIATION

	SWR	LWR
MEAN	598 ly/d (626)	826 ly/d ---

*( ) denotes observed value.

Table 10.3 Output Statistics: January, Boston, MA

	TMP	DEW	CLD	WSP	WDR
MEAN	-1.2°C (-0.9)*	-7.4°C (-7.4)	0.58 (0.61)	5.5 m/s (5.7)	82.8° (74.4)
STANDARD DEVIATION	6.7°C (5.9)	7.7°C (8.2)	0.36 (0.44)	2.6 m/s (2.7)	40.3° (49.0)
LAG-1	0.99 (0.99)	0.99 (0.99)	0.88 (0.89)	0.88 (0.88)	0.85 (0.87)
SKEW	--- ---	--- ---	--- ---	0.68 (0.61)	--- ---

PRECIPITATION

	t_b	t_r	h
MEAN	51.1 hr (55.3)	7.2 hr (8.8)	7.0 mm (9.0)

RADIATION

	SWR	LWR
MEAN	126 ly/d (131)	497 ly/d ---

*( ) denotes observed value.

Table 10.4 Output Statistics: July, Boston, MA

	TMP	DEW	CLD	WSP	WDR
MEAN	22.9°C (22.8)*	15.3°C (15.5)	0.42 (0.45)	4.4 m/s (4.4)	101.6° (102.4)
STANDARD DEVIATION	4.5°C (4.3)	3.5 (3.8)	0.35 (0.40)	1.8 (1.8)	39.8 (46.1)
LAG-1	0.97 (0.97)	0.96 (0.97)	0.89 (0.88)	0.82 (0.81)	0.77 (0.78)
SKEW	--- ---	--- ---	--- ---	0.43 (0.45)	--- ---

PRECIPITATION

	t_b	t_r	h
MEAN	66.1 hr (64.5)	4.1 hr (3.9)	7.9 mm (7.2)

RADIATION

	SWR	LWR
MEAN	551 ly/d (479)	797 ly/d ---

*( ) denotes observed value.

dewpoints are almost always within $0.5^{\circ}C$ of the observed values. These results are particularly satisfying, since the temperature component is by far the most complex part of the CSCS model. In essence, the temperature component is the keystone of the CSCS approach since almost all of the other elements in the model influence or interact with the temperature generation algorithm. For the CSCS model to work as a whole, it is most important that the temperature component performs properly.

Statistically, the cloud cover model worked well too. The means of the generated cloud covers were quite close to the observed values. Remember that the final generated cloud covers are a combination of the generated fairweather sequences, the transition periods, and the storm periods. The input parameters for cloud cover generation were the fairweather statistics and the transition decay coefficients. To obtain the proper output statistics, the CSCS model relies on the transition functions into and out of storm periods that were described in Chapter 3 to create the proper evolution of the entire cloud cover process.

To see how well the generated cloud cover mean values evolved from the fairweather mean values, refer to Table 10.5

where the mean observed fairweather cloud cover, the mean observed cloud cover for the entire record, and the mean generated cloud cover are presented. It is apparent that the CSCS is capable of producing an evolutionary cloud cover process whose statistics are quite close to the observed values.

Reviewing the statistics for wind speed in Tables 10.1 - 10.4 shows that the reproduction of the observed statistics by the CSCS model is excellent. However, reproduction of the wind direction statistics is only fair. This is not really unexpected, given the procedure used to represent wind direction in this study (see Chapter 7). To be more correct, wind direction should, at the very least, not be treated independently. However, for the data sets used in this study, wind direction did not appear to be a particularly strong predictor. Therefore, more sophisticated wind direction generation algorithms were not investigated.

The precipitation statistics were also adequately reproduced. The only significant departure was for the mean time between storms for January at Dodge City. However, January in Dodge City is quite dry. Only about 70-75 storms

Table 10.5 Evolution of Mean Cloud Cover

MEAN VALUES

	Fairweather Observed	Total Observed	Total Generated
Boston, MA			
January	0.43	0.61	0.58
July	0.32	0.45	0.42
Dodge City, KA			
January	0.31	0.38	0.41
July	0.21	0.34	0.33

were generated for the 20 month simulation period for January in Dodge City, compared to about 250 storms for the other data sets. With fewer storms to analyze, higher variability in the statistics is expected.

It is difficult to draw many conclusions regarding the shortwave output since the records used by Getz and Nicholas (1979) to obtain the mean daily shortwave radiation cover a much longer period than the data sets used in this study. It is unclear whether any differences noted between observed and generated values could be attributed to modelling deficiencies or to natural statistical variation. Nevertheless, the generated values are near the observed values and the model is making the correct seasonal adjustments.

Observed data were not available for longwave radiation. However, to the extent that the Idso (1981) expression for atmospheric emissivity (see Chapter 5) represents the conditions at Boston, MA and Dodge City, KS, the generated longwave radiation values should be reasonable.

10.4 Diurnal Curves for Temperature and Dewpoint

In the previous section, statistical evidence was presented to suggest that the temperature and dewpoint components of the CSCS model performed well. It is also important that the two temperature components produce the proper diurnal variations. Figures 10.12 - 10.15 show the observed and generated diurnal curves of temperature and dewpoint (i.e. mean hourly values) for January and July at Boston and Dodge City.

Overall, the generated temperature curves compare quite well with the observed values. The generated minimum and maximum temperatures are all within $1^{o}C$ of the observed values and their timing is about right. The only timing discrepancy occurs for the maximum January temperatures at both Boston and Dodge City. The generated mean maximum temperature occurs around 4:00 PM in January. The observed maximums occur near 5:00 PM at Boston and near 3:00 PM at Dodge City. The variation of the two observed January maximums is probably due to the difference between the coastal climate of Boston and the continental climate of Dodge City. Since the timing of all the minimums and the July maximums is quite good, the exact reason for the generated maximums to be an hour off in January is not readily apparent.

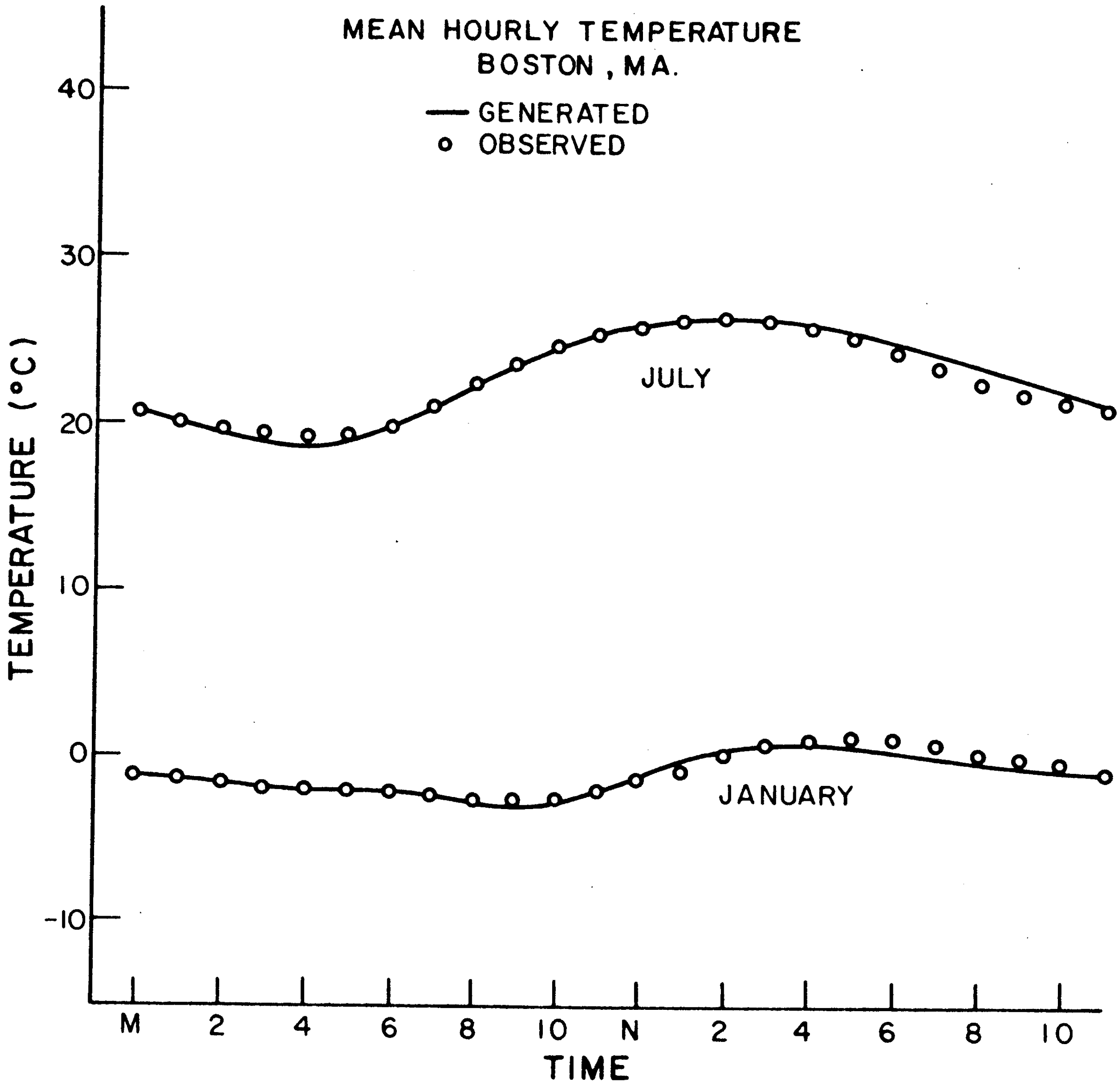


Figure 10.12 Observed and Generated Mean Hourly Temperatures (Boston, MA)

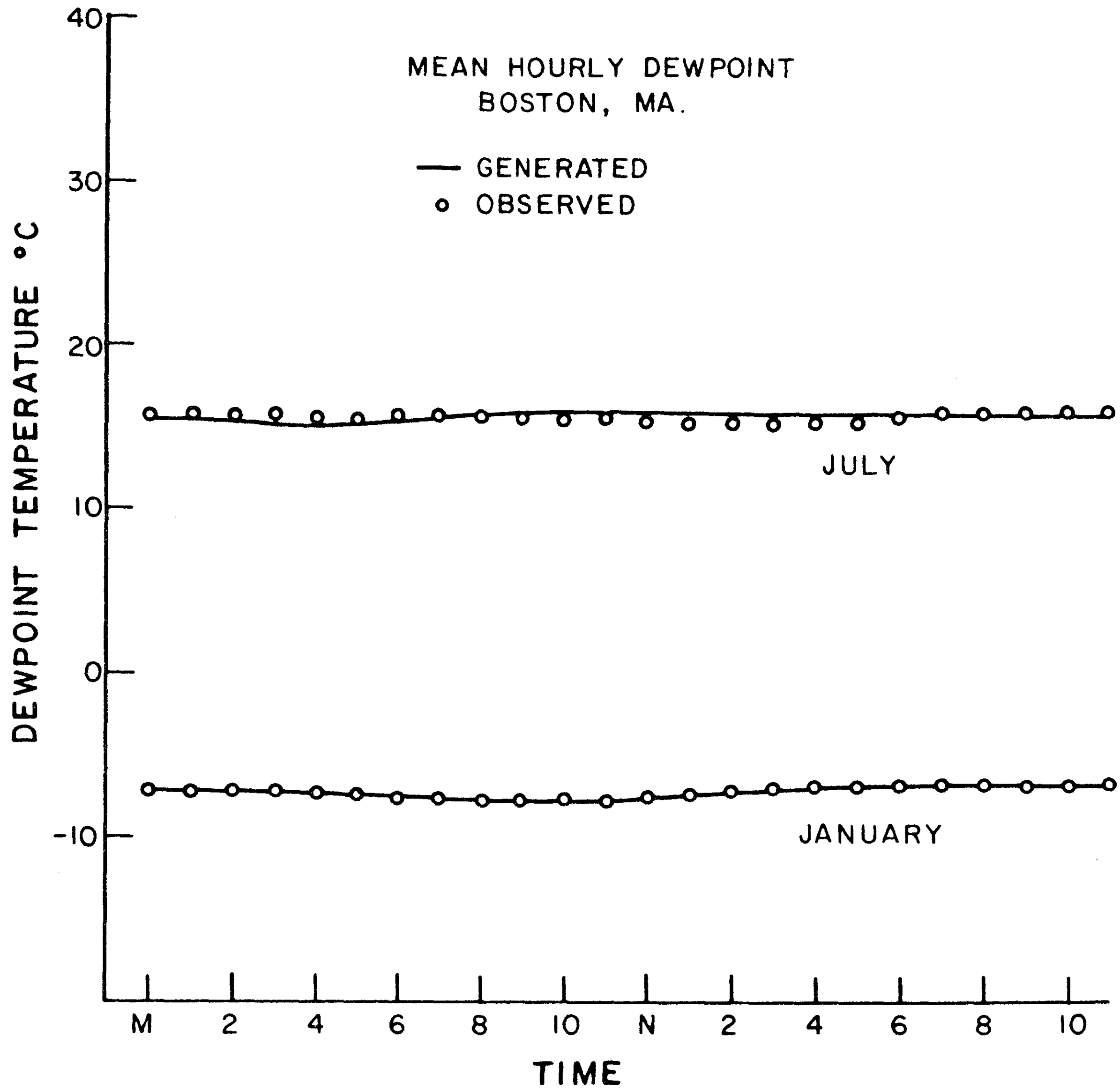


Figure 10.13 Observed and Generated Mean Hourly Dewpoints (Boston, MA)

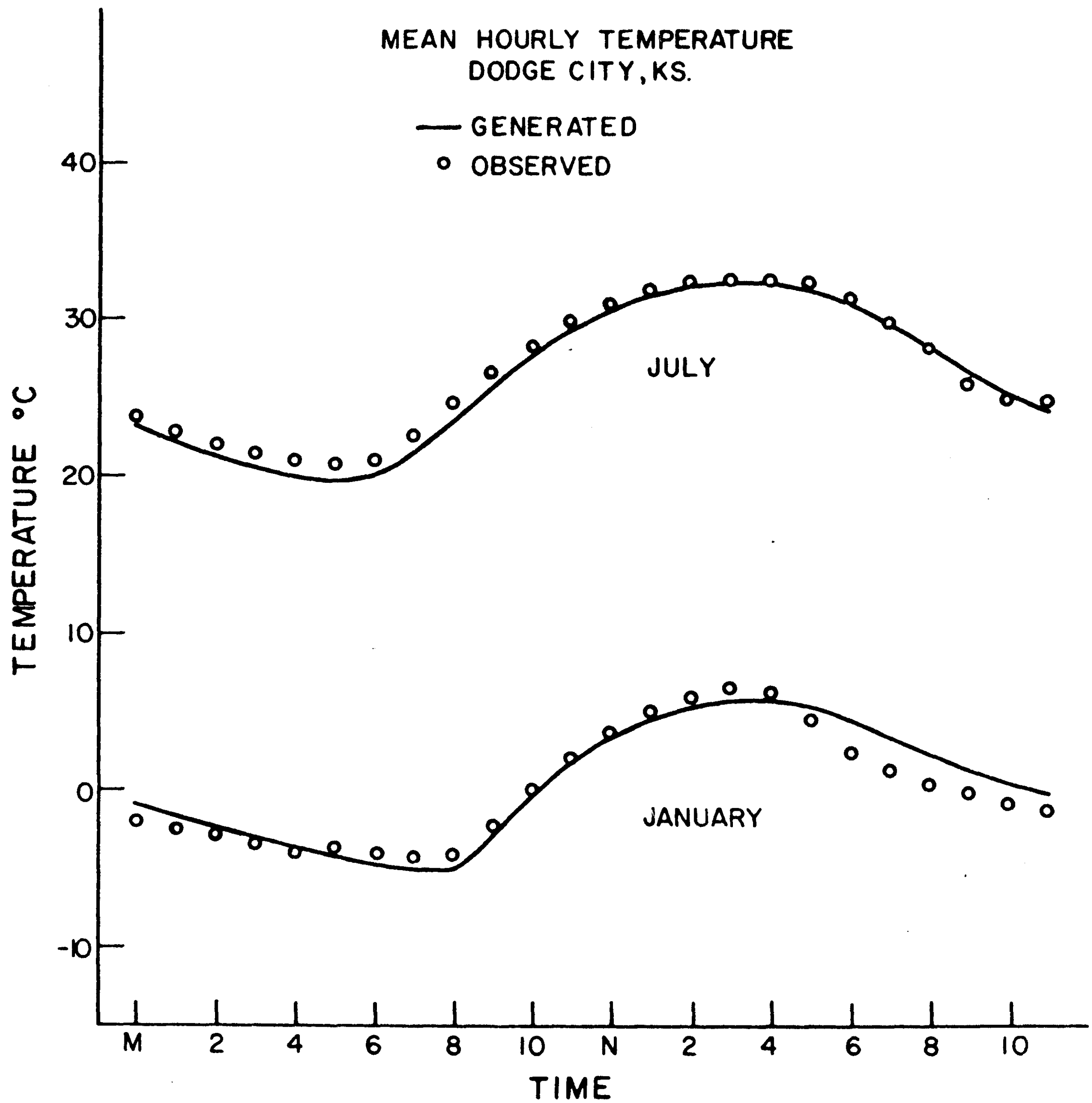


Figure 10.14 Observed and Generated Mean Hourly Temperatures (Dodge City, KS)

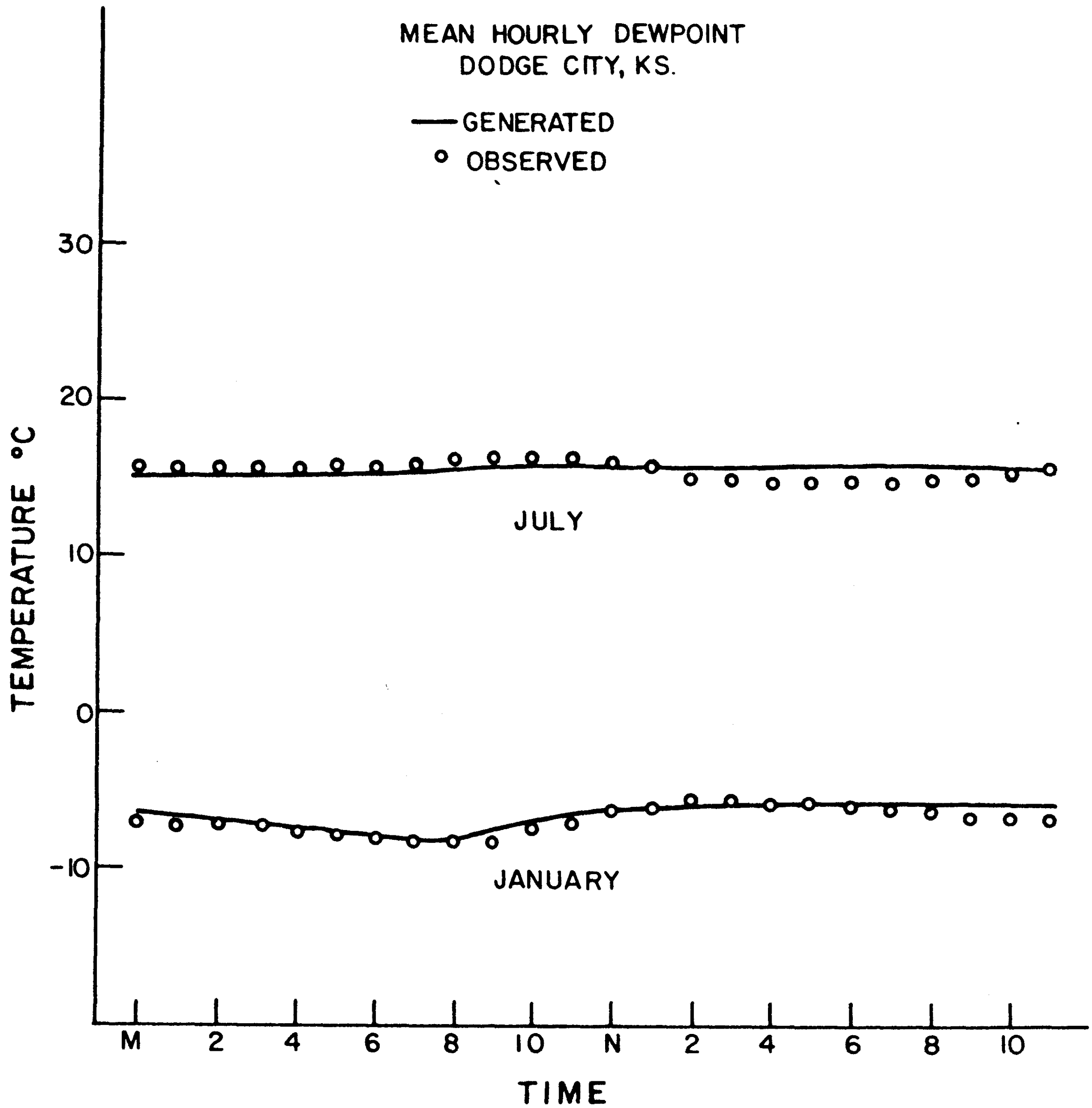


Figure 10.15 Observed and Generated Mean Hourly Dewpoints (Dodge City, KS)

In Boston, the model in its present form does not appear to be accounting for all of the modifying influences of the nearby ocean in the late afternoon. The observed temperatures remain elevated slightly longer in the afternoon before starting the downward trend to the morning minimum. This results in a six-hour period during the evening hours where the model slightly underestimated the temperatures.

Given that the diurnal curve of temperature for January in Boston is so flat ($\sim 5^{o}C$ variation), it is a pleasant surprise that the CSCS model performed as well as it did. Of the four data sets, the January - Boston experiment probably offered the most severe test of the CSCS model's ability to adapt to a variety of climate conditions.

As for the January - Dodge City experiment (Figure 10.14), the observed temperatures in this continental climate drop more sharply in the late afternoon than during the evening and early morning hours. During this period, the temperature model gave a steadier transition for the downward limb of the temperature curve. The exponentially-dominated functions used in the temperature algorithms are not quite able to express the sharp drop observed near sunset in the January - Dodge City experiment.

The CSCS model reproduced the observed diurnal temperature curve for the July experiments quite well. The maximum departure of the generated curve was about $1^{o}C$ for Boston and $1.5^{o}C$ for Dodge City.

Dewpoint temperatures are shown in Figures 10.13 and 10.15. The reader is reminded that an independent stochastic process was used to generate dewpoint temperatures for July and that a regression model was used for January dewpoints.

For July at Boston, (Figure 10.13) the resultant mean generated curve is essentially "flat" as expected and represents the observed dewpoints well. For July at Dodge City, the mean generated curve is again "flat" as expected. However, in the Dodge City observed data there is a subtle wave that is not represented by the stationary lag-1 Markov process. During the forenoon, temperatures rise causing dew to evaporate. This increases the moisture content of the lower atmosphere and elevates the dewpoint temperature. As temperatures continue to rise, more evaporation occurs but by late morning increased instabilities cause mixing with drier air aloft, causing dewpoints to fall.

By early evening, instability decreases again and continued evaporation causes the dewpoints to rise again. To capture this feature, alternative dewpoint generation techniques will have to be explored.

For January, the regression model output represented the observed data well, especially in capturing the morning "dip" in the dewpoint curve. The observed "dip" coincides with the morning temperature minimum. The depressed dewpoints at this time are likely due to moisture driven from the lower atmosphere by frost formation.

Another interesting diurnal curve to review is for the dewpoint depression, defined as the difference between the temperatures and dewpoints. Figures 10.16 and 10.17 present the observed and generated dewpoint depression curves for Boston and Dodge City respectively.

Dewpoint depression is interesting because it is sometimes used as an indicator of the atmosphere's ability to take up moisture. High dewpoint depression values indicate a high capacity to take up moisture. For low dewpoint depression values, the opposite is true. Under the right circumstances then (e.g. with sufficient moisture at the surface), dewpoint depression could also be interpreted as

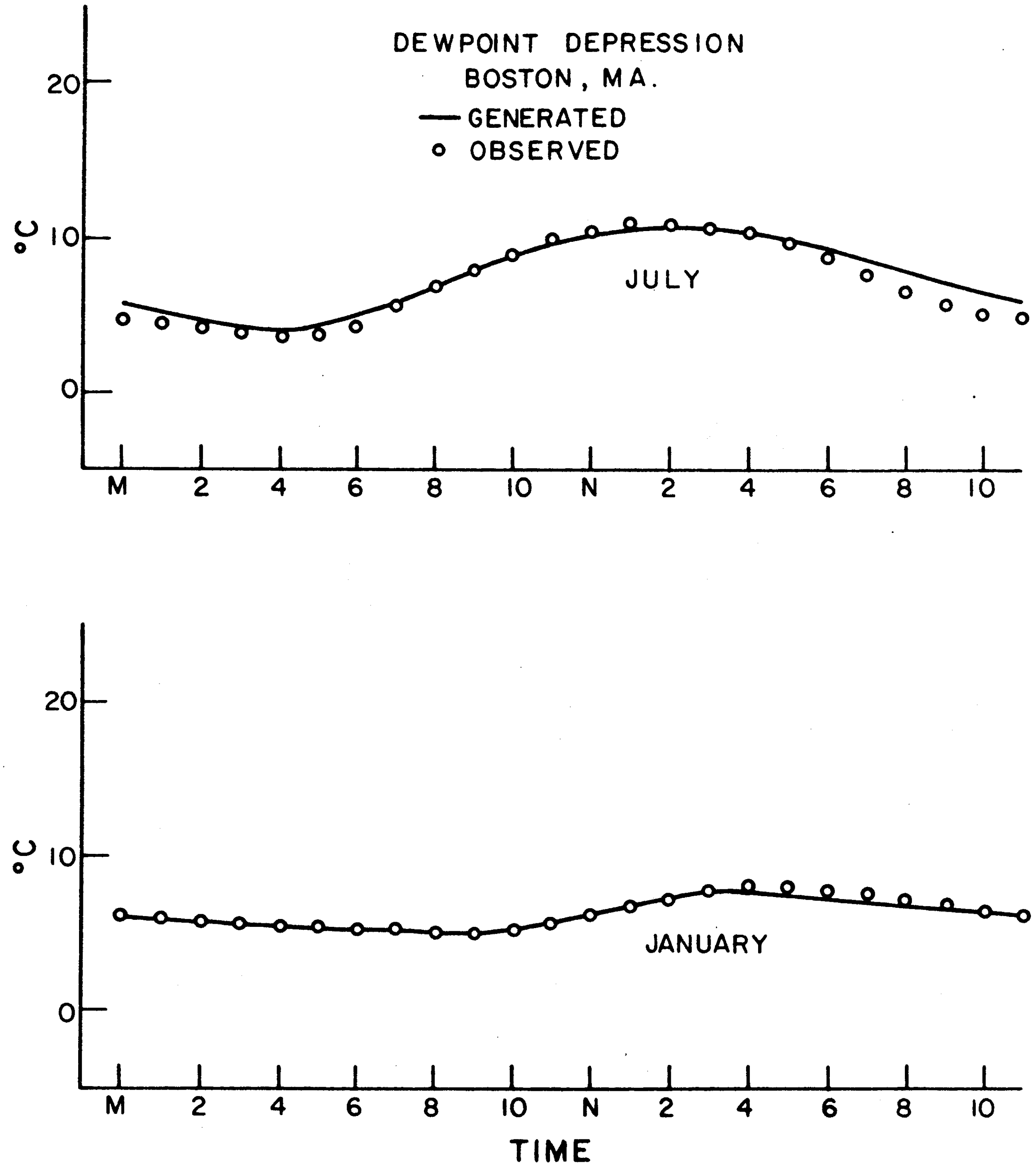


Figure 10.16 Observed and Generated Mean Hourly Dewpoint Depression (Boston, MA)

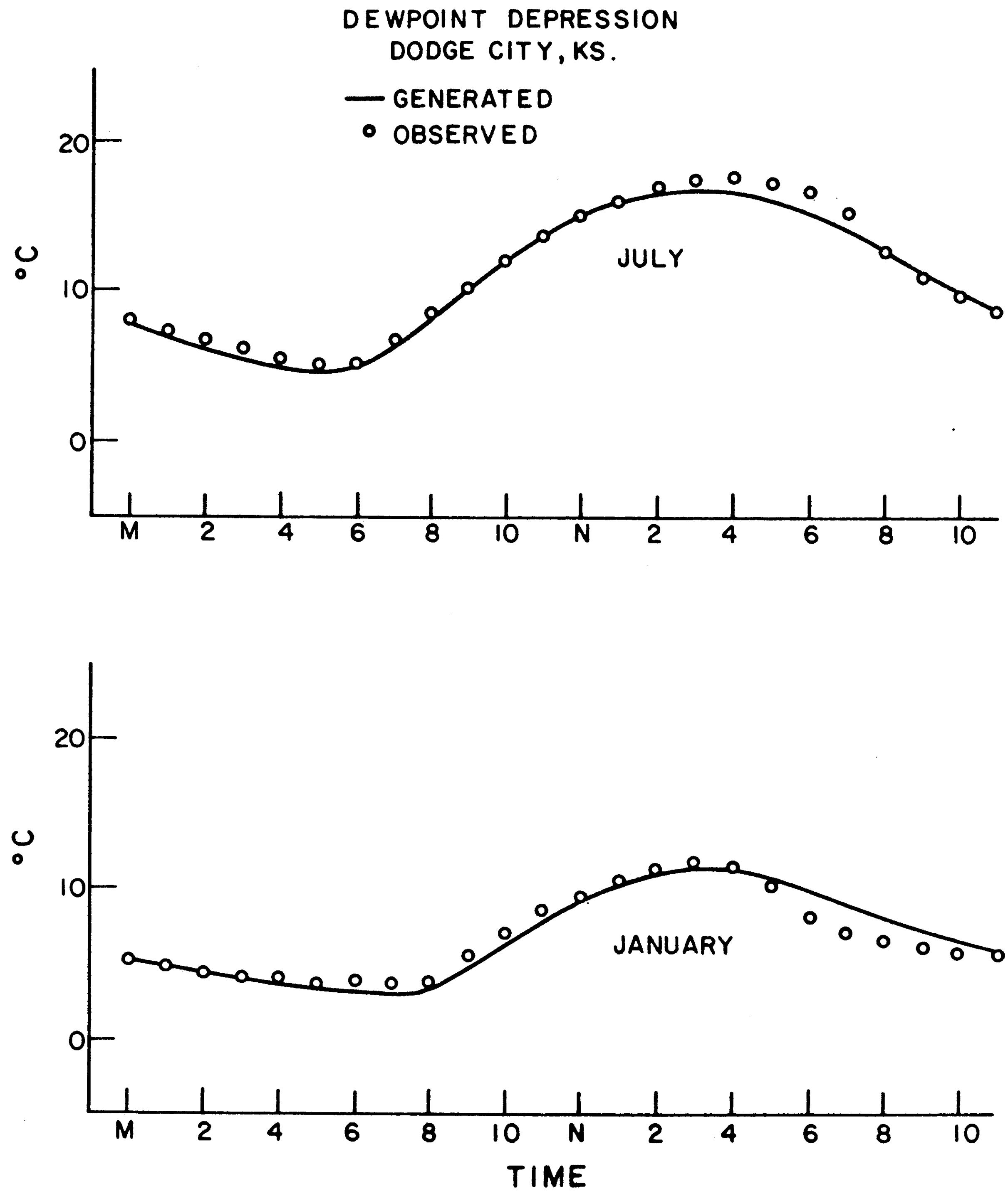


Figure 10.17 Observed and Generated Mean Hourly Dewpoint Depression (Dodge City, KS)

an indicator of surface moisture flux.

Dewpoint depression is not explicitly generated by the CSCS model. It is derived from the output from the temperature and dewpoint components. For the observed and derived dewpoints to compare favorably, the temperature and dewpoint components must be synchronized correctly. In addition, deviations between observed and generated dewpoint depressions can appear more glaring than with either temperature or dewpoint. For example, if a generated temperature and a generated dewpoint differ from their observed values by $1^{\circ}C$, the difference might not be considered significant. However, if the $1^{\circ}C$ differences are opposite in sign, the error in dewpoint depression would be $2^{\circ}C$.

Thus far we have seen that the CSCS model satisfactorily reproduces the desired characteristics of the meteorological data sets. The next step is to examine the target land surface processes that the CSCS output data are designed to force. An application of the CSCS output to a detailed model of the land surface is presented in Chapter 12.

Before we get to the detailed analysis in Chapter 12, it is instructive to make a quick examination of one particular land surface process to see that it is correctly forced by the CSCS model output. Evaporation is perhaps the most important process at the land-atmosphere interface, being the basic mechanism for the restoration of both atmospheric moisture and energy. Solar radiation, temperature, dewpoint, and wind speed all contribute to evaporation. If an estimate of evaporation could be made using these meteorological data, the result would, in essence, be an integration of the joint interactions of the input variables. It is of particular interest to make a comparison of the evaporation estimates computed using the observed meteorological inputs with the estimates computed using the generated CSCS data. In this fashion, we can see to what degree any errors in the CSCS output have an effect on the results of the target process.

Linsley et.al. (1975) present a nomogram solution for the estimation of shallow-lake evaporation as a function of solar radiation, air temperature, dewpoint, and wind movement. Using the mean values of the observed and generated (CSCS) data for July at Boston, MA and Dodge City, KS, evaporation estimates were made with the nomogram of Linsley et.al. The results appear in Table 10.6.

For the Dodge City experiment, the observed and generated evaporation estimates agreed to within 3%. For the Boston experiment, the observed and generated estimates varied by about 9%. The principal source of error in the Boston evaporation estimate stems from the roughly 15% over-estimation of the shortwave radiation input. The shortwave radiation error is likely due to error in the atmosphere attenuation function that was discussed earlier.

10.5 Summary

The results of CSCS model experiments for January and July at Boston, Massachusetts and Dodge City, Kansas have been presented. Hourly data plots, model output statistics, and selected mean diurnal curves were reviewed.

Overall the CSCS model performed well. The results indicate that the CSCS model is capable of generating well coordinated sets of meteorological data with high time resolution (i.e. hourly values). This represents a significant improvement over existing techniques in both the number of variables generated and in the time resolution of the generated data.

Two individual components, cloud cover and temperature, were especially critical to the successful completion of the CSCS model. The modulated non-stationary stochastic process

Table 10.6 Comparison of Shallow Lake Evaporation Estimates for July

		Dodge City		Boston	
Data	Units	OBS	CSCS	OBS	CSCS
Temperature	(^{o}C)	26.8	26.3	22.8	22.9
Dewpoint	(^{o}C)	15.4	15.5	15.5	15.3
Wind	(m/s)	5.6	5.3	4.4	4.4
Shortwave	(ly/d)	626.	598.	479.	551.
Evaporation	(mm/d)	8.1	7.9	5.3	5.8

derived to represent cloud cover enabled the linking of the precipitation, the shortwave radiation, the longwave radiation, the temperature, and the dewpoint regression components with the cloud cover component on an hourly basis. The temperature model enabled the generation of hourly temperatures that were linked to other meteorological variables and that reflected seasonal and geographical changes.

CHAPTER 11

ATMOSPHERIC BOUNDARY LAYER

11.1 Introduction

Vertical transfer of momentum, heat, and moisture between the earth and the free atmosphere occurs through the atmospheric boundary layer. Continuous small scale turbulent fluxes in the boundary layer appear to be the basic mechanism of the exchanges between the atmosphere and the earth. (Bhumralkar, 1979)

Although relatively thin, 10 to 50m (Anderson, 1976), the boundary layer can account for significant atmospheric effects. For example, the boundary layer contains only about 2% of the total atmospheric kinetic energy on an annual basis, yet it contributes up to 25% of the total generation and more than 35% of the total dissipation of atmospheric kinetic energy. (Kung, 1963)

Attempts to quantify earth-atmosphere exchanges have led to a relatively large body of boundary layer literature. General descriptions of turbulent processes of the lower atmosphere can be found in a number of books (e.g., Oke, 1978; Rose, 1966; Priestly, 1959; Sutton, 1953, 1954;

Lumley and Panofsky, 1969).

Two basic approaches to flux estimation commonly appear in the literature. The eddy fluctuation method seeks to describe the instantaneous properties of eddies as they pass a specified level in the boundary layer. Profile or flux-gradient methods infer the flux based on average atmospheric profiles and on the degree of atmospheric stability.

The eddy fluctuation method describes flux using the observation that atmospheric entities exhibit short-term fluctuations about their longer term means. Since the properties contained by an eddy are its density (ρ_e), its vertical velocity (w_v), and the concentration of the atmospheric entity (s), the mean vertical flux density of the entity (S) can be written as (Oke, 1978)

$$S = E\left[(\bar{\rho}_e + \rho_e')(\bar{w}_v + w_v')(\bar{s} + s')\right] \qquad (11.1\text{-}1)$$

where the overbars indicate the mean values and the primes indicate the short-term fluctuations about the means. Expansion of Equation (11.1-1) followed by a term by term evaluation leads to

$$S = E(\rho w_v' s') \qquad (11.1\text{-}2)$$

For the vertical transfer of momentum, sensible heat, and latent heat, Equation 11.1-2 is used to give

$$\tau = -E(\rho_e W_s' w_v') \qquad (11.1\text{-}3)$$

$$Q_H = E(\rho_e c_p w_v' T') \qquad (11.1\text{-}4)$$

$$Q_E = E(\rho_e L_v w_v' q_h') \qquad (11.1\text{-}5)$$

where τ is the shear stress in Pa, ρ_e is the eddy density in kgm^{-3}, W_s' is the horizontal wind speed fluctuation in ms^{-1}, c_p is the specific heat of air in $Jkg^{-1}\,{}^oK^{-1}$, T' is the temperature fluctuation in ^oK, L_v is the latent heat of vaporization in Jkg^{-1}, and q_h' is the specific humidity fluctuation in $kgkg^{-1}$.

The fluctuation terms represent changes in the atmospheric properties over periods on the order of seconds. Data collection for time intervals this short is not routine. In addition, the basic time unit of the CSCS model is one hour. Therefore, eddy fluctuation methods were not used in this study.

In the profile or flux-gradient approach, the flux is generally described by (Oke, 1978)

$$\begin{pmatrix}\text{Flux of}\\ \text{an entity}\end{pmatrix} = \begin{pmatrix}\text{Ability of the medium}\\ \text{to transport the entity}\end{pmatrix} \times \begin{pmatrix}\text{Gradient of}\\ \text{a relevant}\\ \text{property}\end{pmatrix}$$

Through the turbulent surface layer, momentum transfer can be described by

$$\tau = \rho_a K_M \frac{\partial W_s}{\partial z} \qquad (11.1\text{-}6)$$

where ρ_a is the atmospheric density in kgm^{-3}, K_M is the eddy transfer coefficient for momentum in m^2s^{-1}, and z is elevation in m.

For sensible heat flux

$$H = -\rho_a c_p K_H \frac{\partial T}{\partial z} \qquad (11.1\text{-}7)$$

where H is in Wm^{-2}, K_H is the eddy transfer coefficient for heat in m^2s^{-1}, and T is the air temperature in ^oK. Normally, potential temperature is used in Equation 11.1-7. However, in this study, only temperature differences over the lowest 2 meters are of interest. Over this range, potential temperatures and air temperatures are essentially the same. Finally, for water vapor, the turbulent flux transfer can be described by

$$E = -\rho_a K_W \frac{\partial q_h}{\partial z} \qquad (11.1\text{-}8)$$

where E is the water vapor flux in $kgcm^{-2}s^{-1}$, K_W is the eddy transfer coefficient for water vapor in m^2s^{-1}, and q_h is the specific humidity in $kgkg^{-1}$. Equations 11.1-6 to 11.1-8 show that the desired fluxes can be estimated if the appropriate gradient and the associated transfer coefficient are known.

The lower atmosphere is a very active zone with variations in heating and cooling resulting from instantaneous variations of fluxes with height. Over longer periods, such as a half-hour or more, flux variations with height are very small (Oke, 1978). Therefore, the surface layer is often called the layer of constant flux. Practically, this means that estimates of flux at any point in the lowest 50m over a suitable site are assumed equal to their surface values. Atmospheric variables generated for the two meter level by the CSCS model can then be used to help estimate transfers across the land-atmosphere interface.

11.2 Profile Method for Flux Estimation

In a neutrally stable atmosphere, (i.e., one with an adiabatic lapse rate), under fully turbulent conditions, the wind profile is logarithmic and expressed by

$$\frac{W_s}{W_*} = \frac{1}{k} \ln\left(\frac{z - d_o}{z_o}\right) \qquad (11.2\text{-}1)$$

where d_o is the zero displacement plane in meters, z_o is the roughness length in meters, k is the von Karman constant (0.40), and W_* is the friction velocity defined by

$$W_* = (\tau_o/\rho_a)^{\frac{1}{2}} \qquad (11.2\text{-}2)$$

where τ_o is the shear stress at the surface in Pa.

The vertical profile of the horizontal wind speed is found by differentiating Equation 11.2-1 and rearranging to give

$$\frac{\partial W_s}{\partial z} = \frac{W_*}{kz} \qquad (11.2\text{-}3)$$

Remembering that the boundary layer is also assumed to be a layer of constant flux, we can write

$$\tau = \tau_o = \text{constant} \qquad (11.2\text{-}4)$$

Using Equations 11.1-6, 11.2-2, 11.2-3, and 11.2-4, an expression for the eddy transfer coefficient for momentum can be written as

$$K_M = k^2 z^2 \frac{\partial W_s}{\partial z} \qquad (11.2\text{-}5)$$

Equation 11.2-5 shows that the transfer coefficient for momentum is also a function of the vertical gradient of the horizontal wind.

The problem of establishing the transfer coefficients K_H and K_W can be simplified by invoking the "principle of similarity". (Oke, 1978). Under this assumption, an atmospheric eddy can transport any conservative entity with equal facility. Therefore,

$$K_M = K_H = K_W \qquad (11.2\text{-}6)$$

Using Equations 11.2-5 and 11.2-6, a new expression for sensible heat flux can be written as

$$H = -\rho_a c_p k^2 z^2 \frac{\partial W_s}{\partial z} \frac{\partial T}{\partial z} \qquad (11.2\text{-}7)$$

Likewise, an expression for water vapor flux can be written as

$$E = -\rho_a k^2 z^2 \frac{\partial W_s}{\partial z} \frac{\partial q_h}{\partial z} \qquad (11.2\text{-}8)$$

Equation 11.2-8 can also be written in terms of vapor pressure by using an approximation for specific humidity

$$q_h = \frac{0.622 e_o}{P_a} \tag{11.2-9}$$

where e_o is the vapor pressure in mb, P_a is the atmospheric pressure in mb, and the constant, 0.622, is the molecular weight ratio of water vapor to dry air. Substitution of Equation 11.2-9 into Equation 11.2-8 gives

$$E = - \frac{0.622 \rho_a}{P_a} k^2 z^2 \frac{\partial W_s}{\partial z} \frac{\partial e_o}{\partial z} \tag{11.2-10}$$

The equations for τ, H, and E presented so far, are strictly valid for neutral stability only. For stable and unstable conditions, the wind profile is not generally logarithmic. Stable conditions dampen free convection and, using the logarithmic wind profile, cause the fluxes to be overestimated. The opposite is true for unstable conditions.

Monin and Obukhov (1954) have generalized the logarithmic wind profile for all conditions, giving

$$\frac{\partial W_s}{\partial z} = \frac{W_*}{kz} \Phi_M \tag{11.2-11}$$

where Φ_M is an empirically determined adjustment factor that is related to atmospheric stability. Obviously, for neutral conditions, Φ_M is unity.

Similar functions can be defined for the sensible heat and water vapor profiles, giving

$$\frac{\partial T}{\partial z} = - \frac{H}{\rho_a c_p K_H} \Phi_H \qquad (11.2\text{-}12)$$

and

$$\frac{\partial q_h}{\partial z} = - \frac{E}{\rho_a K_W} \Phi_W \qquad (11.2\text{-}13)$$

where Φ_H and Φ_W are the stability related profile adjustment functions. According to Monin and Obukhov (1954), the functions Φ_M, Φ_H, and Φ_W should be functions of a dimensionless height ratio z/L. L is constant with height in the boundary layer and is presented by Anderson (1976) as

$$L = - \frac{W_*^3 c_p \rho_a T}{kgH} \qquad (11.2\text{-}14)$$

where g is the acceleration of gravity in ms^{-2}. The ratio z/L is positive for stable atmospheric conditions, zero for neutral, and negative for an unstable atmosphere. Several studies, conducted under the assumption that the transport mechanisms of conservative entities are similar, and therefore, that their profiles are similar, have resulted in empirical relationships for $\Phi(\frac{z}{L})$. (Dyer, 1967; Dyer and Hicks, 1970; Dyer and Grant, 1978; Businger et. al. 1971; McVehil, 1964; Oke, 1970; Yamamoto and Shimanuki, 1966) Not all researchers agree on the form of the Φ-functions, but the so-called Businger-Dyer formulae are frequently used. For stable conditions, these give

$$\Phi_M = \Phi_H = \Phi_W = 1 + 5\,\frac{z}{L} \qquad (11.2\text{-}15)$$

which implies that $K_M = K_H = K_W$. For stable conditions, the equalities of the eddy transfer coefficients and the Φ-functions are supported by the studies of Saugier and Ripley (1978) and Monji and Businger (1972). For unstable conditions

$$\Phi_M^2 = \Phi_H = \Phi_W = \left(1 - 16\,\frac{z}{L}\right)^{-\frac{1}{2}} \qquad (11.2\text{-}16)$$

The studies of Saugier and Ripley (1978) and Monji and Businger (1972) also provide observational support for Equation 11.2-16.

Since the information required to evaluate L in Equation 11.2-14 is not generally available, some other stability-related procedure to compute z/L from routinely measured data is needed. Richardson (1920) developed a criterion that "reflects the ratio of the consumption of energy by buoyancy forces to the rate of its production by wind shear." (Anderson, 1976)

Anderson gives the gradient form of the Richardson number as

$$Ri = \frac{g(\partial T/\partial z)}{T(\partial W_s/\partial z)^2} \tag{11.2-17}$$

Thus, the Richardson number can be computed from observations of wind speed and temperature. Anderson (1976) also shows that the Φ-functions can be written in terms of the Richardson number. For stable conditions

$$\Phi_M = \Phi_M = \Phi_W = (1-5Ri)^{-1} \tag{11.2-18}$$

and for unstable conditions

$$\Phi_M^2 = \Phi_H = \Phi_W = (1-16Ri)^{-\frac{1}{2}} \tag{11.2-19}$$

Comparison of Equation 11.2-19 with Equation 11.2-16 shows that, in the Businger-Dyer formula for unstable conditions, the height ratio, z/L, is equal to the Richardson number, Ri.

11.3 Computation of Turbulent Transfer Using Measurements at One Level

The CSCS model generates representative data at the

2-meter level only. Therefore, it is not possible to evaluate the various gradients described in the previous section. Similarly, observations of wind, temperature, and humidity are made at one level for most data collection sites. To overcome this problem, the flux equations must be used in their integrated form. If these equations are integrated between z_o and z_a, (assuming $W_s = 0$, $T = T_b$, and $e = e_b$ at the bottom of the boundary layer), the flux equations become

$$\tau = \rho_a C_M W_s^{\,2} \tag{11.3-1}$$

$$H = -\rho_a c_p C_H W_s (T - T_b) \tag{11.3-2}$$

and

$$E = -\frac{0.622\rho_a}{P_a} C_W W_s (e_o - e_b) \tag{11.3-3}$$

where W_s is the 2-meter wind speed in ms^{-1}, T is the 2-meter temperature in ^oK, and e_o is the 2-meter vapor pressure in mb. C_M, C_H, and C_W are the dimensionless transfer coefficients for the integrated flux equations and are called the "bulk" transfer coefficients. Under neutral conditions, and using the similarity assumption, we have

$$(C_H)_N = (C_W)_N = (C_M)_N = \frac{k^2}{\left[\ln\left(\frac{z-d_o}{z_o}\right)\right]^2} \qquad (11.3\text{-}4)$$

where the subscript N denotes neutral conditions.

Deardorff (1968) developed ratios of the bulk transfer coefficients for the general case to their neutral values. For stable conditions where it is assumed $\Phi_M = \Phi_H = \Phi_W$, the ratios can be written as

$$\frac{C_W}{(C_W)_N} = \frac{C_H}{(C_H)_N} = \frac{C_M}{(C_M)_N} = (1.0 - 5(Ri)_B)^2 \qquad (11.3\text{-}5)$$

where $(Ri)_B$ is the bulk Richardson number given by Anderson (1976) as

$$(Ri)_B = \frac{2gz(T-T_b)}{(T+T_b)W_s^2} \qquad (11.3\text{-}6)$$

For unstable conditions, Deardorff (1968) gives

$$\frac{C_M}{(C_M)_N} = \left[1.0 - \frac{(C_M)_N^{\frac{1}{2}}}{k}\left[\ln\left(\frac{1+x^2}{2}\right) + 2\ln\left(\frac{1+x}{2}\right) - 2\tan^{-1}(x) + \frac{\pi}{2}\right]\right]^{-2} \qquad (11.3\text{-}7)$$

and

$$\frac{C_H}{(C_H)_N} = \frac{C_W}{(C_W)_N} = \left(\frac{C_M}{(C_M)_N}\right)^{\frac{1}{2}} \left[1.0 - \frac{2}{k}(C_M)_N^{\frac{1}{2}} \ln\left(\frac{1+x^2}{2}\right)\right]^{-1}$$

(11.3-8)

where

$$x = (1 - 16\,\frac{z}{L})^{\frac{1}{4}} \qquad (11.3\text{-}9)$$

If Equations 11.3-1 and 11.3-2 are substituted into Equation 11.2-14, the Monin-Obukhov length can be written as

$$L = \frac{C_M^{3/2} T W_s^2}{C_H k g (T - T_b)} \qquad (11.3\text{-}10)$$

Dividing the numerator and denominator by $(C_M)_N^{3/2}$ and using Equation 11.3-6 gives the relationship between the height ratio z/L and the bulk Richardson number.

$$\frac{z}{L} = \frac{k C_H/(C_M)_N}{(C_M)_N^{\frac{1}{2}} \left(\frac{C_M}{(C_M)_N}\right)^{3/2}} (Ri)_B \qquad (11.3\text{-}11)$$

By knowing the wind speed, temperature, and vapor pressure at the two meter level and the temperature at the bottom of the boundary layer (i.e., at the land surface), the fluxes can be estimated. The wind speed, temperature, and vapor pressure at the two meter level are generated by the CSCS model. If the temperature, T_b, is available from a model of the land surface, the fluxes across the earth-atmosphere interface can be generated.

11.4 Solution Procedure

For neutral and stable conditions, the bulk transfer coefficients are easily computed. Finding the coefficients for unstable conditions is not quite as straightforward. The coefficients depend on the ratio z/L. But from Equation 11.3-11, it is seen that the coefficients are needed to determine z/L in the first place.

The problem of calculating the transfer coefficients is solved in two phases. First, a table is constructed that relates the ratio z/L to the bulk Richardson number given values for z, z_o, and d_o. Second, during program execution, $(Ri)_B$ is computed from Equation 11.3-6 and z/L is found directly from the table. Once z/L is known, the coefficients are easily found.

CHAPTER 12

LAND SURFACE APPLICATION

12.1 Introduction

To demonstrate the utility of the CSCS model, generated data were used as input to a detailed model of the land surface. The resulting fluxes are plotted here to note any trends that occurred due to different meteorological forcings given identical initial conditions. Also, the mean daily fluxes are presented to show how the partitioning of energy in the surface heat balance changed for each experiment.

12.2 CSCS Generated Data Sets

Three different generated data sets were used. First, the observed statistical parameters found for July in Boston, MA were used to generate a "normal" meteorological data set. The output from the land surface model that results from the "normal" forcing serves as a baseline for comparison with the results from the other experiments.

A second data set was generated that represents a weather scenario which is much wetter than normal. This was accomplished by changing <u>only</u> the input parameters, $\bar{t}_b$, $\bar{t}_r$, and $\bar{h}$ for the precipitation component. The precipitation statistics were estimated from the July 1959 data for Boston, the wettest month in the record.

Finally, a third data set was generated using the observed statistical parameters but adding a constant 2.2°C (4°F) bias to the temperature component. The bias was introduced by adding a constant to the stochastic term in the temperature component represented by Equation 6.3-22.

Due to the rather large computational requirements of the land surface model, the length of simulation was limited to one month for each data set. Table 12.1 presents the statistics obtained from the three CSCS data sets compared to the observed values for the period of record.

Selecting the "normal" data set presented some difficulty. Since the CSCS output is stochastic and since one month is too short a period for statistics to stabilize, it is essentially impossible to generate one month of data with all statistics identical to the historical values. Therefore, several monthly runs were made and the monthly data set whose statistics were judged to most closely represent the historical values was selected as the "normal" data set.

Table 12.1 Data Set Statistics For the Land Surface Application:
July, Boston, MA

SET	TEM ^oC	DEW ^oC	CLD	WSP m/s	WSR deg	SWR ly/d	LWR ly/d	t_b hr	t_r hr	h mm
OBS	22.8	15.5	0.45	4.4	102.4	479	---	64.5	3.9	7.2
NORM	22.7	14.4	0.46	4.6	99.2	522	784	57.3	5.1	6.0
WET	24.4	15.4	0.50	4.3	106.7	534	808	44.4	5.6	18.6
BIAS	25.5	17.7	0.45	4.4	101.5	523	842	45.8	4.1	9.0

For the "wet" data set, the input precipitation parameters were changed from the observed values presented in Table 12.1 to $\bar{t}_b$ = 52.3 hr, $\bar{t}_r$ = 5.3 hr, and $\bar{h}$ = 17.2 mm. Decreasing the time between storms and increasing the storm duration caused the mean cloud cover to increase. In fact, the observed mean cloud cover in July 1959 in Boston was 0.51. This compares with a generated value of 0.50 (see Table 12.1).

It is interesting to note that in spite of the increased mean cloudiness for the "wet" data set, the mean daily shortwave radiation was actually higher than for the "normal" data set. This can occur when, over short periods of time such as one month, the higher levels of cloudiness happened to occur during the night or during times when shortwave radiation is low (e.g. early morning or late afternoon). Existing meteorological data generation algorithms are unable to capture the stochastic feature.

The temperature-biased data set has a mean temperature that is 2.8°C higher than the mean temperature of the "normal" data set. The only other model output variable that is directly influenced by the temperature bias is the longwave radiation. Table 12.1 shows that the longwave radiation is significantly higher for the "biased" data set than for the "normal" data set. The independently generated dewpoints happened to be high for the "biased" data set and also served to drive up the longwave radiation.

12.3 Land Surface Model

The computer model of the land surface used in this study numerically simulates moisture and heat transport in a hysteretic, inhomogeneous porous media (Milly, 1982). In particular, the model is used to represent a vertical column of soil that begins at the land surface and extends downward to a depth of 500 cm.

The atmospheric forcings represented by the CSCS data sets (translated from the 2-meter level to the surface by the boundary layer component described in Chapter 11) define the surface boundary conditions. At the lower boundary, no diffusion of soil moisture or heat is assumed and water leaves the soil column only by gravity drainage, advecting sensible heat with it. Only vertical variations of heat and moisture are considered.

The soil parameters are based on hypothetical silt loam soil. A summary of the soil parameters appears in Table 12.2.

The initial conditions were the same for each experiment. Initially, temperature, matric potential, and volumetric liquid water content were assumed uniformly distributed over the entire soil column. The starting values for these parameters were chosen based upon the mean temperature and mean precipitation for July in Boston as well as upon the properties of the silt loam soil (Milly, 1982). The initial conditions chosen for the current study are:

1. temperature, $T = 22.7^{o}C$
2. matric potential, $\Psi = -1000$ cm
3. volumetric liquid water content, $\theta = 0.233\ cm^3/cm^3$

The output from the land surface model includes plots of the time history of the components of the surface heat balance:

$$R_n - G = H + LE \tag{12.3-1}$$

where R_n is the net all-wave radiation, G is the heat flux into the ground, H is the turbulent sensible heat diffusion into the atmosphere minus the sensible heat carried into the soil by water that infiltrates during precipitation, and LE is the turbulent latent heat diffusion into the atmosphere. All values in Equation 12.3-1 are expressed in langleys/day (ly/day). In addition to the time histories of the surface energy balance components, their average daily values are also available for comparison.

The surface moisture balance equation is written as

$$q_m/\rho_1 = -P + E + \frac{dh_d}{dt} + R_s \tag{12.3-2}$$

where q_m is the upward mass flux of water in $gcm^{-2}d^{-1}$, P is the precipitation rate in cm/d, E is the evaporation rate in cm/d, h_d is depression storage depth in cm, R_s is the surface runoff in cm/d, and ρ_1 is the liquid mass density in g/cm^3. The surface heat balance and the surface moisture balance equations are linked by the evaporation terms, LE and E. Thus, the latent energy term represents an energy form of the evaporation rate which adds another interpretive element to the plots of LE.

12.4 Results

Figures 12.1-12.6 present the 31 day plots of the individual terms in the surface heat balance equation that result from the land surface simulations using the different meteorological data sets (i.e. "normal", "wet", and "biased"). Obviously, some of the fine details in the plots were sacrificed in order to plot all the data. However, the sig-

Table 12.2 Summary of Soil Parameters (ref. Milly, 1982)

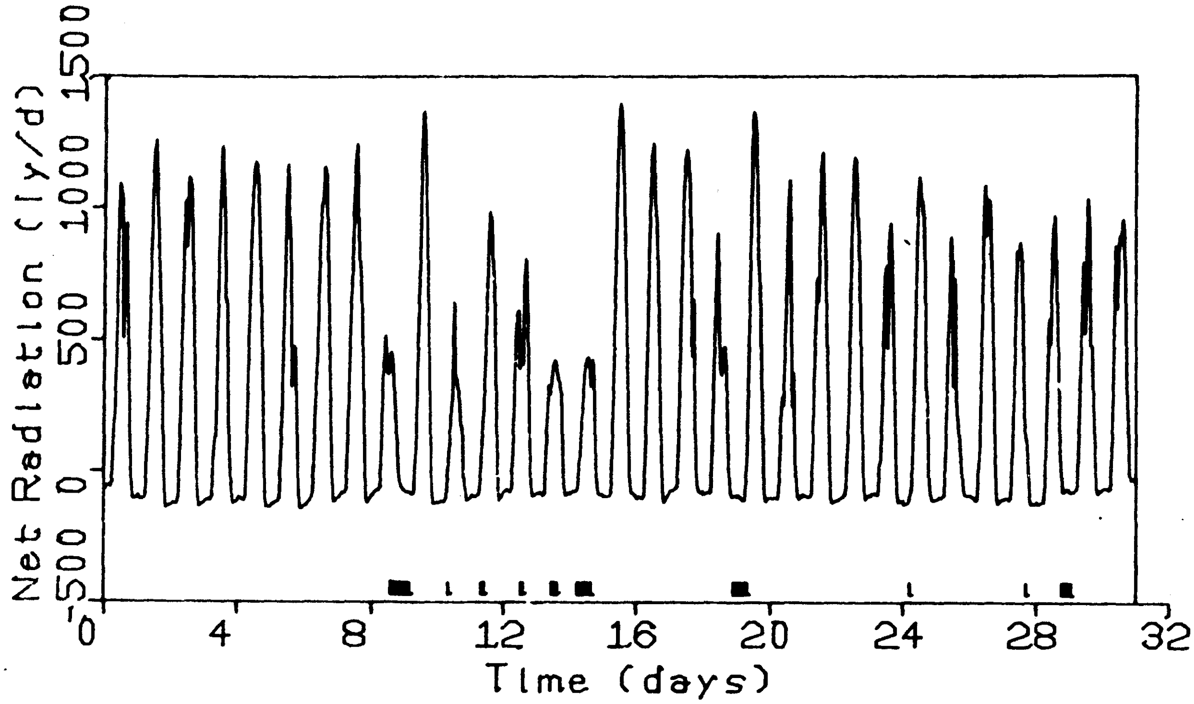
Parameter	Value	Parameter	Value
n	0.46	θ_3	0.16
θ_u	0.414	θ_4	0.33
K_s	10^{-4}cm/s	θ_5	0.05
a	0.210	θ_k	0.11
b	-495.	S	10^5cm^{-1}
c	-0.147	A_d	0.20
d	0.0	A_w	0.10
e	-0.0489	h_{max}	0.5cm

Soil Constituent	i	C_i	λ_i	g_i
Liquid water	1	1.0	$1.37(10^{-3})$	--
Air	2	$3(10^{-4})$	*	*
Quartz	3	0.46	$2.1(10^{-2})$	0.125
Other minerals	4	0.46	$7(10^{-3})$	0.125
Organic matter	5	0.6	$6(10^{-4})$	0.5

* variable - see Milly (1982), Chapter 2

Table 12.2 (continued)

Parameter	Definition
n	soil porosity
θ_u	proportion of medium occupied by water upon rewetting to zero matric potential
K_s	hydraulic conductivity at saturation and temperature T_0
a	fitted coefficient for wetting function
b	fitted coefficient for wetting function
c	fitted coefficient for wetting function
d	fitted coefficient for wetting function
e	fitted coefficient for wetting function
θ_3	volumetric soil fraction of quartz
θ_4	volumetric soil fraction of "other" minerals
θ_5	volumetric soil function of organic matter
θ_k	moisture content at which liquid flow becomes negligible
S	specific surface
A_d	albedo of soil when dry
A_w	albedo of soil when wet
h_{max}	maximum depression storage
C_i	volumetric heat capacity of the i-th constituent
λ_i	thermal conductivity of the i-th constituent
g_i	shape factor of the i-th constituent
T_0	initial temperature of arbitrary reference temperature



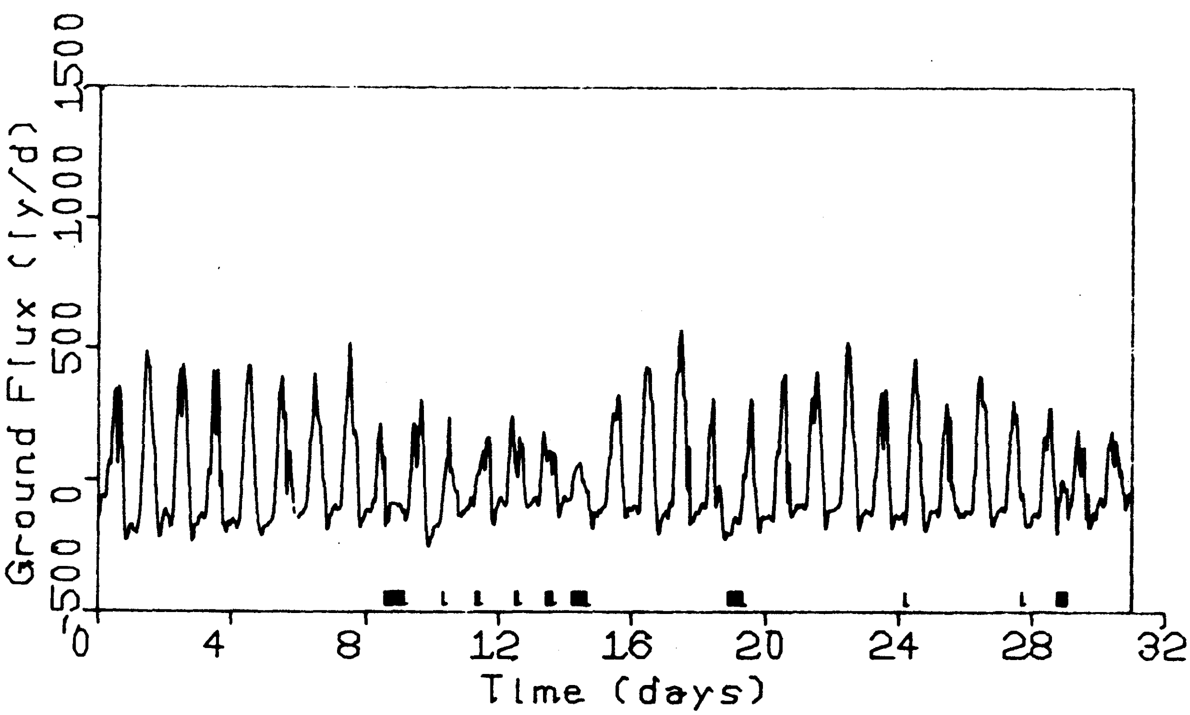


Figure 12.1 Net Radiation and Ground Flux - "normal"

nificant trends can still be examined. All of the data are plotted in units of ly/d. Periods of precipitation are indicated by the "tic" marks just above the time line. The tic marks do not indicate intensity, just the occurrence of precipitation.

The most significant feature of all the plots is the strong diurnal signature. This is obviously due to the radiation input which is dominated by the shortwave component. Figure 12.1 presents the plot of net radiation for the "normal" run. The peak net radiation values represent a positive contribution to the surface heat balance of on the order of 1000 ly/d. At night there is a slight radiational loss as expected. Cloud cover significantly affects net radiation. This is especially clear during the relatively stormy period from day 9 to day 15. The increased cloudiness during the period cut the peak radiational input nearly in half.

Overall the ground flux (Figure 12.1) is the smallest contributor to the heat balance. Although quite variable, the flux away from the surface during the day is very nearly balanced by flux toward the surface at night. During the summer months, such as July, there is a slight positive net ground flux.

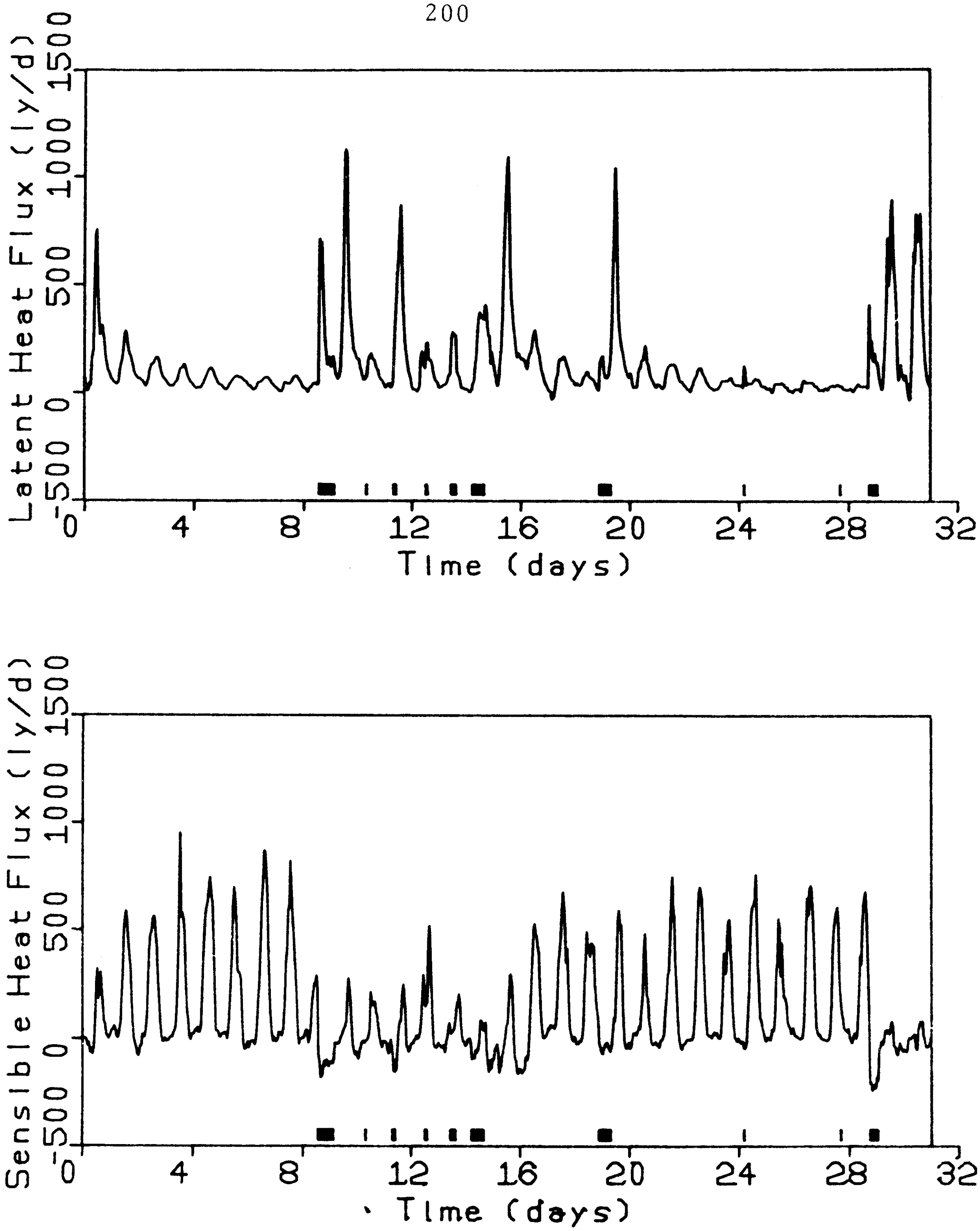


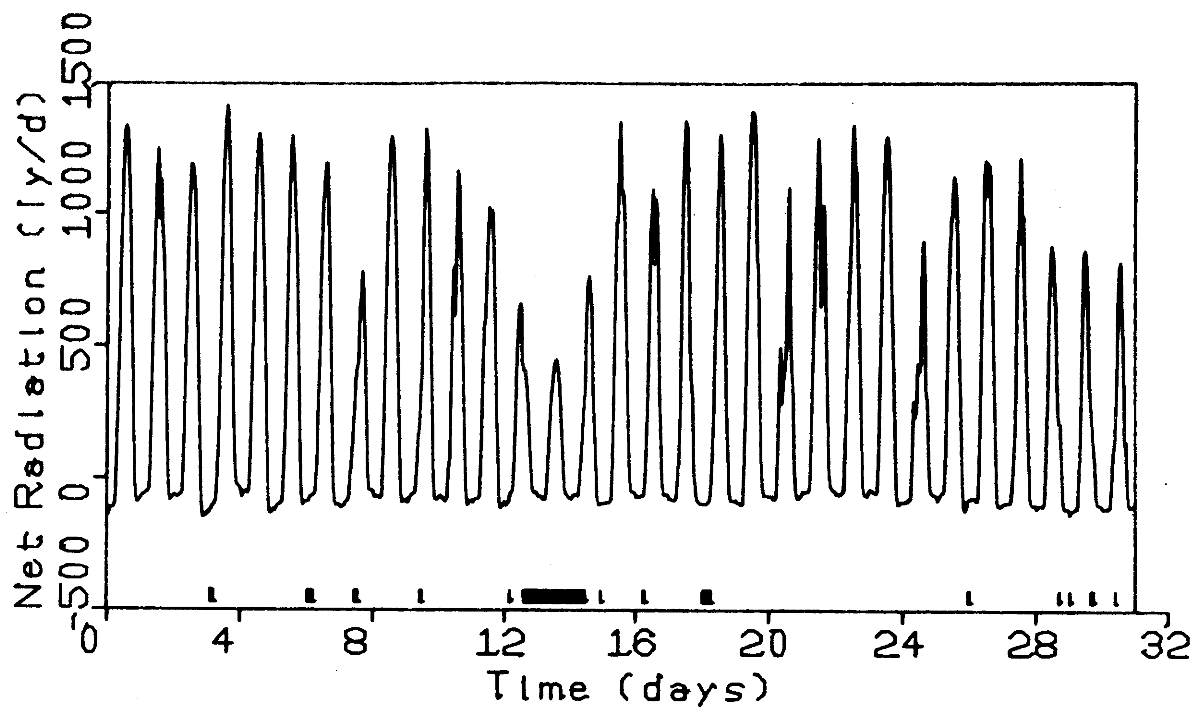
Figure 12.2 Latent and Sensible Heat Flux - "normal"

The latent heat flux for the "normal" experiment is shown in Figure 12.2. A diurnal signature is present in the latent heat flux plot, however, its magnitude depends heavily on the availability of liquid water to evaporate. During the two dry periods (days 1-8 and days 20-28), the latent heat flux steadily decreases as the supply of available liquid water is exhausted. As soon as the available water supply is replenished, the latent heat flux increases sharply again.

The sensible heat flux (Figure 12.2) runs essentially counter to the latent heat flux. As the latent heat flux decreases, the excess heat is transferred to the atmosphere as sensible heat. Once the water is available again to evaporate, the sensible heat flux decreases in response to the increased latent heat flux (see days 9-15 and days 29-31).

Figures 12.3-12.4 present the results of the experiment using the "wet" data set. In this data set, the input short and longwave radiation were higher (as discussed earlier) which is reflected in the net radiation plot.

The biggest change between the results of the experiment using the "normal" data and the experiment using the "wet" data set is evident in Figure 12.4. A much higher



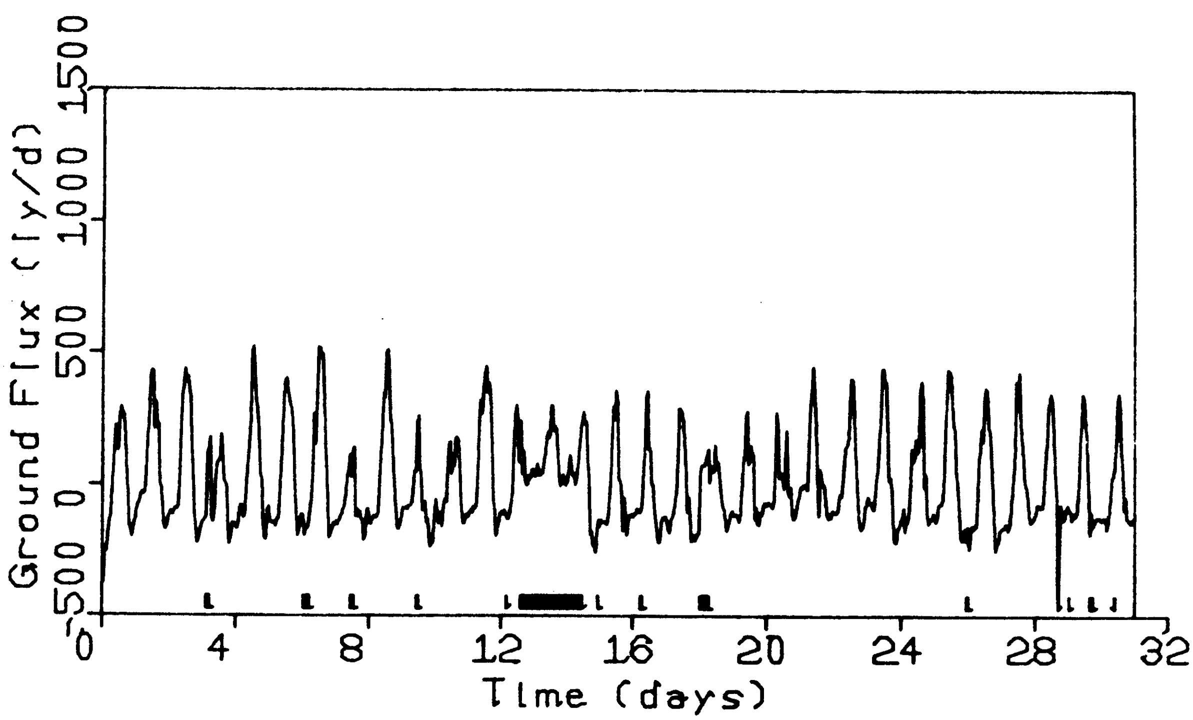


Figure 12.3 Net Radiation and Ground Flux - "wet"

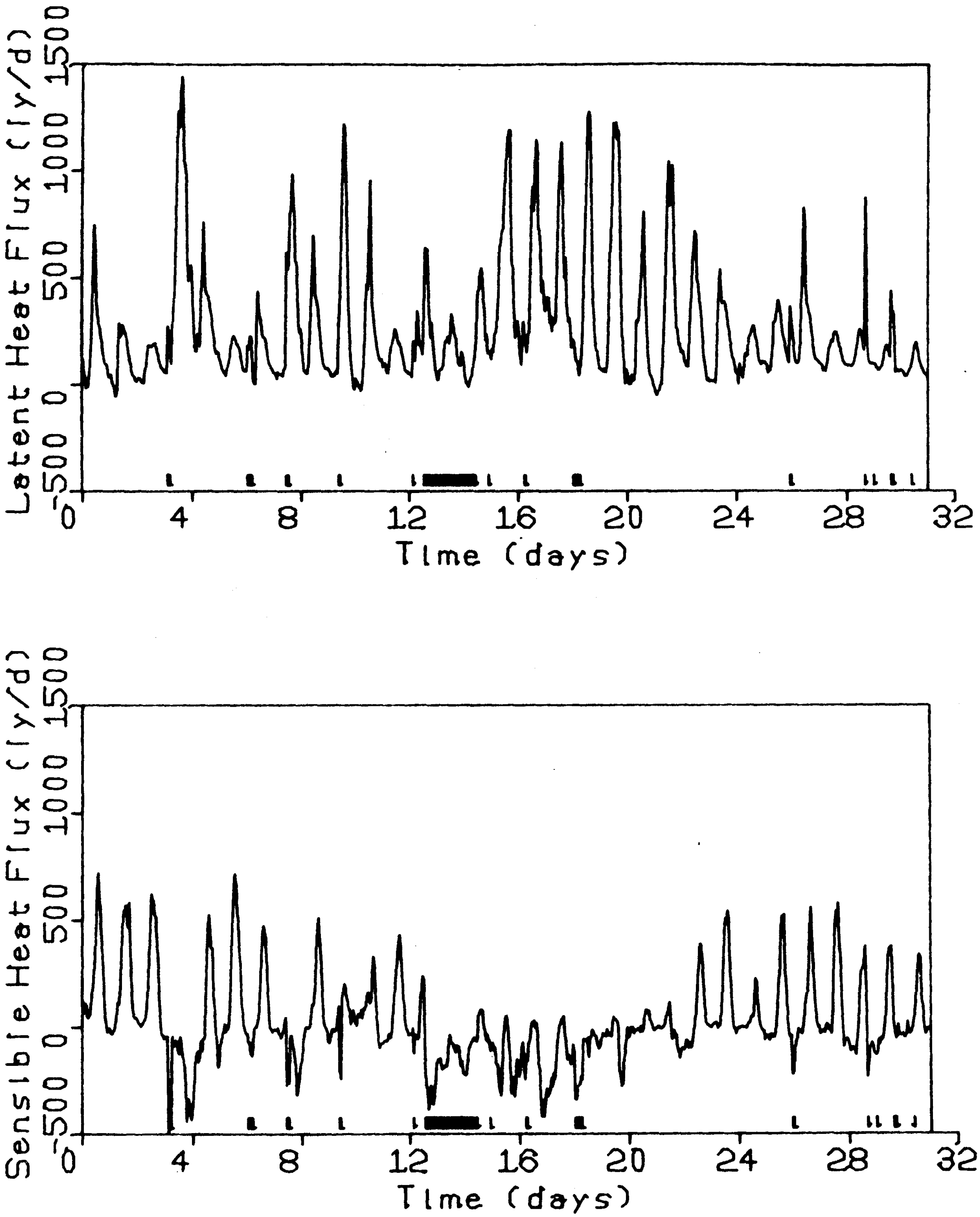
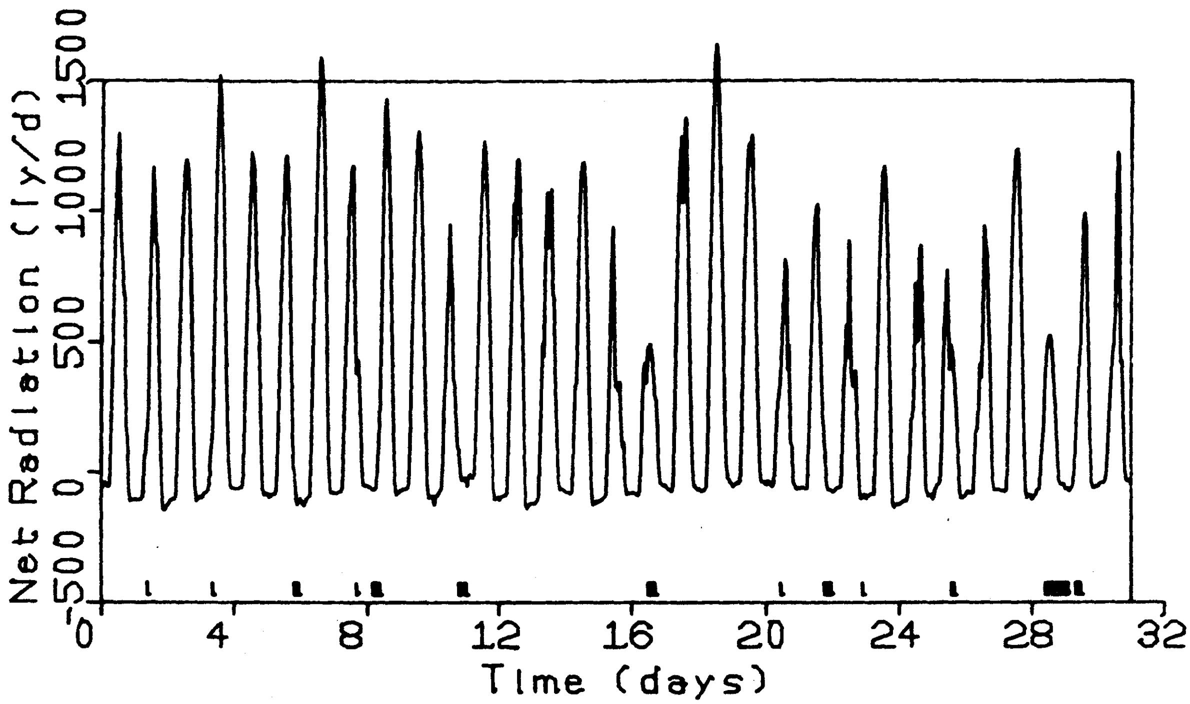


Figure 12.4 Latent and Sensible Heat Flux - "wet"

amount of water was available for evaporation. Thus, high rates of evaporation were sustained throughout the month and sensible heat flux remained at fairly low levels.

Figures 12.5-12.6 present the results from the experiment using the temperature-biased input data. Net radiation levels were even higher for this experiment due to the significant increase in longwave radiation input. This leads to very high peak fluxes of latent heat (Figure 12.6) but the water supply was not able to sustain those rates for very long. Accordingly, the sensible heat fluxes (Figure 12.6) were higher than for the "wet" case (Figure 12.4).

Table 12.3 summarizes the average values for all four terms in the surface heat balance. For the experiment using the "normal" input data, the sensible and latent heat fluxes were portioned almost equally. However in the "wet" experiment, sufficient liquid water was available to allow the latent flux to dominate. In the "bias" experiment, increased radiant energy coupled with a higher than "normal" supply of available water allowed the latent flux to dominate the convective transport but not to the extent of the "wet" case.



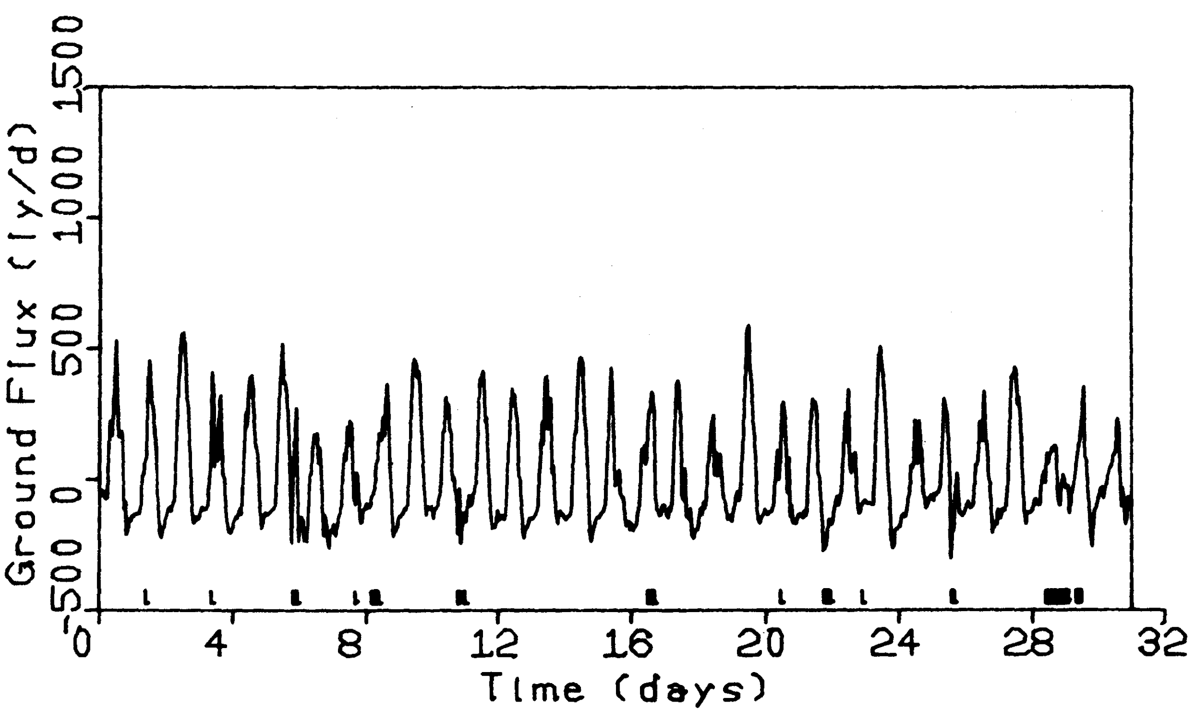


Figure 12.5 Net Radiation and Ground Flux - "temperature-biased"

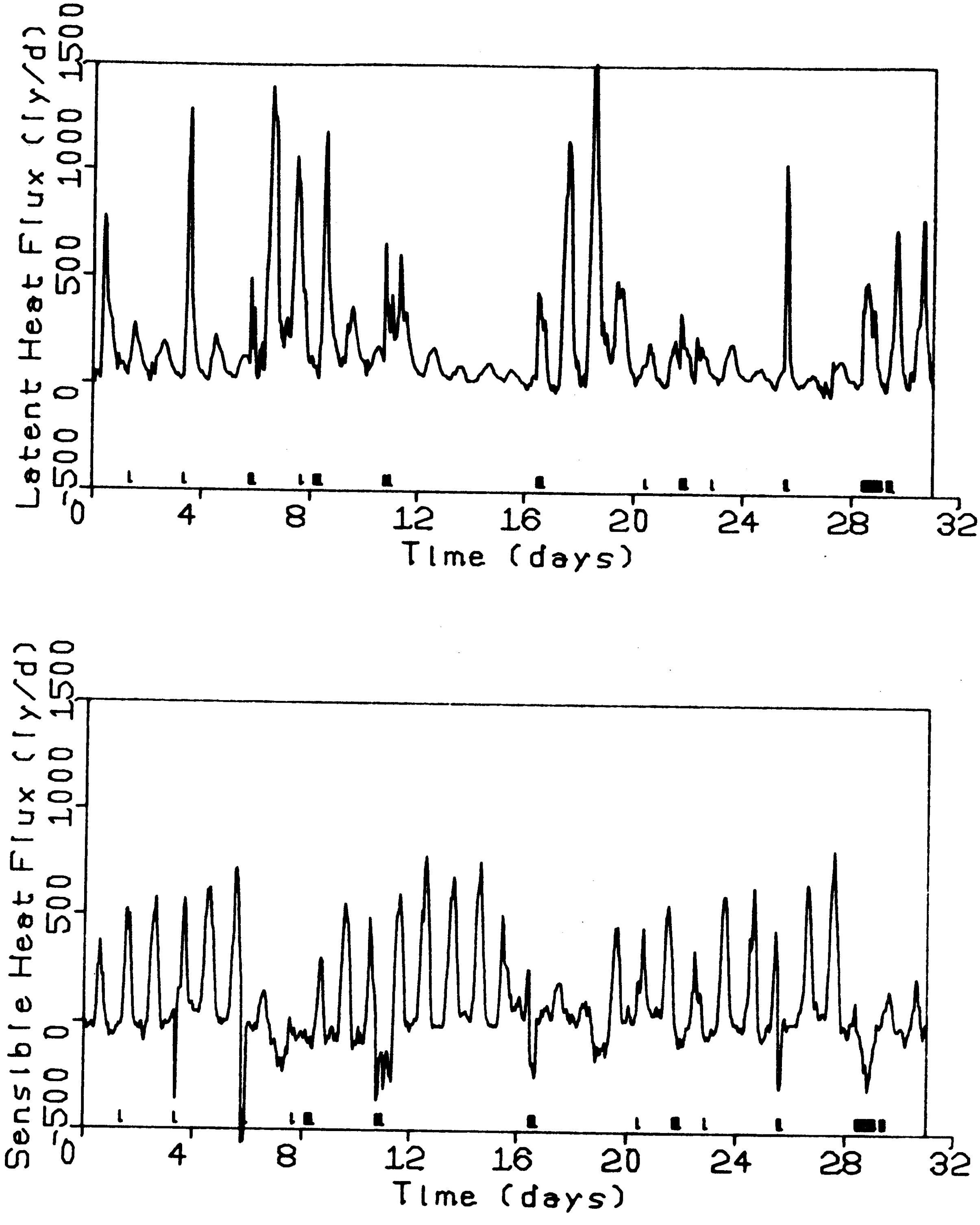


Figure 12.6 Latent and Sensible Heat Flux - "temperature-biased"

Table 12.3 Average Heat Flux For the Land Surface Simulations (All values in ly/d)

	$\overline{R}_n$	$\overline{G}$	$\overline{H}$	$\overline{LE}$
NORMAL	262	6	126	131
WET	302	12	20	270
BIAS	304	12	102	190

NOTE: $R_n - G = H + LE$

12.5 Summary

Three different data sets generated by the CSCS model were used as input to a detailed model of the land surface. In each case, the initial soil column conditions were identical. Thus the differences noted in the resulting surface fluxes were caused by the variations in the input data sets.

The variations in the input data set were in turn caused by varying the input parameters of the CSCS model. This demonstrated the use of the CSCS model to study the response of a land surface to a particular change in a climate or weather scenario. The stochastically generated data set resulting from such experiments will include many of the "ripple" effects that might evolve in a naturally occurring scenario due to the physical coupling of the atmospheric processes.

CHAPTER 13

SUMMARY, RECOMMENDATIONS, AND CONCLUSIONS

13.1 Summary

A computer model representing a new methodology called Constrained Stochastic Climate Simulation has been presented. The CSCS model jointly generates ten meteorological variables with hourly resolution.

Two significant problems were overcome during the development of the CSCS model. As a result, new procedures for the generation of cloud cover and temperature were proposed. These procedures account for the severe non-stationarities in the cloud cover and temperature data and allow the necessary linkages to other CSCS model components.

The CSCS model was tested on four data sets (January and July for Boston, MA and January and July for Dodge City, KS). In each case, hourly output data plots, model output statistics, and mean diurnal curves were examined. The CSCS generated data were shown to represent the historical data well. In addition, estimates of shallow-lake evaporation were made using observed and generated data statistics for July at Boston and Dodge City. This tested the joint use of several CSCS output variables. Again, the results using the CSCS data were satisfactory.

To demonstrate the utility of the CSCS model, three different data sets were generated to use as input to a detailed model of the land surface. Simple changes to the input parameters of the CSCS model were all that were required to create new data sets needed to study how the land surface system responded to different forcings.

13.2 Recommendations

Several recommendations for future work have been discussed in previous chapters. These and several additional recommendations are summarized here.

The precipitation regimes of certain climates exhibit significant diurnal variations. Warm humid climates dominated by late afternoon rain showers illustrate this point. Ways of incorporating this feature into the CSCS need to be explored.

Since the precipitation model "drives" the cloud cover model in the CSCS, diurnal variations in cloud cover due to the precipitation regime will also be accounted for. This "ripple" effect will continue through the CSCS model to the other components linked by cloud cover. (i.e. shortwave radiation, longwave radiation, temperature and dewpoint).

The methodology used to determine an appropriate input probability mass function for the fairweather cloud cover generation algorithm needs to be reviewed. The difficulties in regenerating as well as interpreting the observed frequency histogram were discussed in Chapter 9. Either a more effective way of preserving the strongly U-shaped distribution or a way to quantitatively express the unobservable physical processes needs to be developed.

An alternative shortwave radiation attenuation algorithm was presented in Chapter 4. This method should be implemented in the CSCS model and the results compared with those of the current technique. Both methods need to be compared with more detailed shortwave radiation data than were available for this study. This would help determine whether the use of the more complex alternative is warranted.

The longwave radiation model uses the latest results of Idso (1981). However, his model apparently has been tested at only one site (Phoenix, AZ). Idso's results are promising, but the generality of his model is still open to question. More testing of Idso's approach is needed.

The temperature model has been shown to perform well for two different months, January and July. Although these months represent two climate extremes, the other months should also be tested, particularly the more volatile transition months during Spring and Fall.

In this study, temperature model parameters were estimated for each month. Since the temperature model includes terms that reflect the day of the year, experiments are needed to determine if parameters should be estimated monthly or if parameters could be used that represent longer periods such as a season. If parameters could be developed seasonally, the total parameter estimation chore would be significantly reduced.

Wind speed and wind direction were generated independently in this study. For some locations, the assumption of independence would not be valid. It may be more appropriate to condition wind speeds on wind direction.

In future versions of the CSCS model, wind direction should be generated from its vector component form instead of its azimuth form. By using the x-y components of the wind vector, a continuous bivariate probability distribution function such as the bivariate normal distribution might be used. This should produce a more realistic wind direction specification than currently possible.

July dewpoints were generated independently. This assumption was reasonable for Boston but in Dodge City, stability-related effects in the diurnal dewpoint curve were not reproduced. If it is important to capture this feature, other generation techniques such as the regression model used for January dewpoints should be explored.

In addition to the recommendations relating to the individual components, there is a broader concept that should provide an interesting topic for future research. It relates to the purely stochastic portions of the CSCS model components.

One common way of handling non-stationarities in data that are to be represented by a stochastic generation procedure is to remove the non-stationarities from the data analytically and to treat residuals as a stationary stochastic process. This is essentially the procedure used in the CSCS approach, particularly in regard to cloud cover, temperature, and dewpoint. In the CSCS model these residuals were assumed to be independent. This assumption should be explored more carefully. If significant correlations exist between the residuals, standard multivariate techniques might be used to jointly generate the residuals and thus further improve the coordinated output of the CSCS model.

13.3 Conclusions

The CSCS model is a flexible and efficient tool that can provide high resolution meteorological data to be used in a variety of applications including land surface flux studies, plant disease propogation modelling, insect infestation modelling, irrigation management, and crop forecasting. A variety of possible input weather or climate scenarios could be applied to a system simulation and the outputs could be used to develop probability statements about future events. Various management decisions could be made accordingly.

The flexibility that is inherent in the CSCS model was achieved without great computational cost. This is very desireable since the CSCS model will generally be a tool of the study, not the primary system of interest.

Even for very long simulation periods (e.g. 100 months), the CPU times required on a DEC-10 computer are on the order of minutes. Contrast this to the execution times of the land surface model by Milly (1982) and the model of surface hydroclimatology by Sellers and Lockwood (1981) which are on the order of hours (or days). Thus, the use of the CSCS model in these cases would add an insignificant computational burden to the simulation studies.

REFERENCES

Abramowitz, M. and Stegun, I.A. (editors) Handbook of Mathematical Functions, National Bureau of Standards Applied Mathematics Series No. 55, November 1970.

Ahmed, Jamil, Optimization of Water-Use Efficiency in Crop Production Systems by Dynamic Simulation of Crop Behavior Under Stochastic Regimes, PhD thesis, Texas A&M University, 1974.

Alexander, R.H., Lewis, J.E., Lins, H.F., Jenner, C.B., Outcalt, S.I. and Pease, R.W., Applications of Skylab Data to Land Use and Climatological Analysis Final Report, U. S. Geological Survey, Skylab/EREP, Inventory No. 469.

Anderson, Eric A.,"A Point Energy and Mass Balance Model of a Snow Cover," NOAA Technical Report NWS 19, U. S. Department of Commerce National Weather Service, February 1976.

Angstrom, A., "A Study of the Radiation of the Atmosphere", Smithsonian Miscellaneous Collection, 65(3), 1915.

Angstrom, A., "Effective Radiation During the Second International Polar Year", Medd. Statens Meteorological Hydrology Anst. 6(8), 1936.

Anthony, K.C., "Paper 2: The Background to the Statistical Approach", The Modern Design of Wind-Sensitive Structures, proceedings of the seminar held on 18 June 1970 at the Institution of Civil Engineering, published by Construction Industry Research and Information Association, London, 1971.

Arakawa, A., "Design of the UCLA Atmospheric General Circulation Model", Technical Report No. 7, Department of Meteorology, UCLA, 1972.

Baker, Donald N., "Plant Growth Modelling," Agricultural Engineering, Vol. 62, No. 9, September, 1981.

Barnard, James C., On the Sensitivity of an Atmosphere in Radiative-Convective Equilibrium to Soil Moisture, M.S. thesis, Department of Meteorology, M.I.T., 1977.

Begg, J.E., Bierhuizen, J.F., Lemon, E.R., Misra, D.K., Slatyer, R.O., and Stern, W.R., "Diurnal Energy and Water Exchanges in Bulrush Millet in an Area of High Solar Radiation", Agricultural Meteorology, 1:294-312, 1964.

Bhumralkar, C.M., "Atmospheric Boundary Layer Processes and their Parameterization in Climate Models", Man's Impact on Climate, Wilfried Bach, et.al., Editors, Elsevier Scientific Publishing Company, 1979.

Bingham, C., "Distributions of Weekly Averages of Diurnal Temperature Means and Ranges about Harmonic Curves", Monthly Weather Review, Vol. 89, September 1961.

Bolsenga, S.J., "Daily Sums of Global Radiation for Cloudless Skies", U.S.Army Material Command, Cold Regions Research and Engineering Laboratory, Hanover, NH, Technical Report 160, November 1964.

Bolsenga, S.J., "The Relationship Between Total Atmospheric Water Vapor and Surface Dewpoint on a Mean Daily and Hourly Basis", Journal of Applied Meteorology, June 1965.

Box, George E.P., Jenkins, Gwilym M., Time Series Analysis Forecasting and Control, Holden-Day, 1976.

Bransom, F.A., Miller, R.F. and McQueen, T.S., "Plant Communities and Soil Moisture Relationships near Denver, Colorado," Ecology, Vol. 46, No. 3, pp. 311-319.

Brown, L.R., "The World Food Prospect", Science, 190, 1053-9, 1975.

Brunt, D., "Notes on Radiation in the Atmosphere", Quarterly Journal of the Royal Meteorological Society, 58, 389-418, 1932.

Bryan, Joseph G., Short Range Hour-by-Hour Forecast of Temperature by Projecting the Characteristic Curve with Constants Fitted to Immediately Preceeding Data, unpublished report, The Travelers Research Center, Inc., Hartford, Connecticut, August 1964.

Budyko, M.I., "Teplovoi Balans Zemnoi Poverkhnosti", Gidrometeorologicheskoe, Izdatel'stvo, Leningrad, 1956,(English translation by N. A. Stepanova, 1958, "The Heat Balance of the Earth's Surface". Office of Technical Services, U. S. Department of Commerce, 1958).

Businger, J.A., Wyngaard, J.C., Izumi, Y., Bradley, E.F., "Flux-Profile Relationships in the Atmospheric Surface Layer", Journal of the Atmospheric Sciences, Vol. 28, 1971.

Caughey, T.K., and Stumpf, H.J., "Transient Responses of a Dynamic System under Random Excitation," Journal of Applied Mechanics, December 1961, p 563.

Changnon, S.A., Huff, F.A., Cloud Distribution and Correlation with Precipitation in Illinois, Report of Investigation 33, Illinois State Water Survey, 1957.

Charney, J.G., "A Comparative Study of the Effects of Albedo Change in Drought in Semi-arid Regions", Journal of the Atmospheric Sciences, Vol. 34, No. 9, pp. 1366-1385, September 1977.

Corby, G.A., Gilchrist, A. and Newson, R.L., "A General Circulation Model of the Atmosphere Suitable for Long Period Integrations," Quarterly Journal of the Royal Meteorological Society, Vol. 98, pp. 809-832, 1972.

Crutcher, Harold L., Gamma Distribution Bias and Confidence Limits, U.S.Department of Commerce, NOAA Technical Report EDIS 30, 1978.

Curtis, D.C., A Point Model for the Stochastic Generation of Selected Hydrometeorological Variables, unpublished report, Course 1.66, Massachusetts Institute of Technology, Summer 1978.

Davenport, A.G., "The Spectrum of Horizontal Gustiness Near the Ground in High Winds", Quarterly Journal of the Royal Meteorological Society, April 1961, 87, 194-211.

Davis, J.B., Garner, T.H., "Simulation of Heat and Moisture Movements During Drying of a Two-dimensional Soil Profile", ASAE, 78-4029, 1978.

Deardoff, James W., "Dependence of Air-sea Transfer Coefficients on Bulk Stability," Journal of Geophysical Research, Vol. 73, No. 8, April 15, 1968.

Dyer, A.J., "The Turbulent Transport of Heat and Water Vapor in an Unstable Atmosphere", Quarterly Journal of the Royal Meteorological Society, London, England, 1967.

Dyer, A.J., Hicks, B.B., "Flux-gradient Relationships in the Constant Flux Layer," Quarterly Journal of the Royal Meteorological Society, London, England, Vol. 96, 1970.

Dyer, A.J. and Garratt, J.R., "The Variation of Eddy Fluxes with Height and Fetch in an Unstable Atmosphere," Journal of the Meteorological Society of Japan, Vol. 56, No. 1, 1978.

Eagleson, Peter S., "Climate, Soil and Vegetation 2. The Distribution of Annual Precipitation Derived from Observed Storm Sequences," Water Resources Research, Vol. 14, No. 5, 1978.

Eagleson, P. S., A Dynamic Land Surface Boundary Condition for Climate Models. Research proposal submitted to NSF, MIT, 1978.

Eagleson, Peter S., Dynamic Hydrology, McGraw-Hill, 1970.

Fishman, George S., Concepts and Methods in Discrete Event Digital Simulation, John Wiley & Sons, 1973.

Fox, Martin, Rubin, Herman, A Stochastic Model for the Evolution of Cloud Cover - Estimation of Parameters and Goodness of Fit Based on Boston Data, Air Force Cambridge Research Laboratories, United States Air Force, Bedford, MA, Contract No. AF19(628)-355, Final Report May 7, 1965.

Fritz, S., "Solar Radiation During Cloudless Days," Heating and Ventilating, January 1949.

Gates, W.L. and Schlesinger, M.E., "Numerical Simulation of the January and July Global Climate with a Two-level Atmospheric Model, Journal of the Atmospheric Sciences, Vol. 34 pp 36-76, January 1977.

Getz, Rodger R., Nicholas, Michael M., "Probabilities and Extremes of Solar Radiation by Climatic Week," NOAA Technical Memorandum NWS SR-98, Scientific Services Division, Southern Region, Fort Worth, Texas, January 1979.

Grace, R.A., Eagleson, P.S., "A Model for Generating Synthetic Sequences of Short-time-interval Rainfall Depths," Proc. International Hydrology Symposium, Fort Collins, Colorado, pp 268-276, 1967.

Grayman, Walter M. and Eagleson, Peter S., Streamflow Record Length for Modelling Catchment Dynamics, Technical Report 114, Ralph M. Parsons Laboratory for Water Resources and Hydrodynamics, M.I.T., 1969.

Gringorten, Irving I., "A Stochastic Model of the Frequency and Duration of Weather Events," Journal of Applied Meteorology, Vol. 5, October 1966.

Gringorten, Irving I., "Modelling Conditional Probability," Journal of Applied Meteorology, Vol. 10, p. 646, August 1971.

Haan, Charles T., Statistical Methods in Hydrology, Iowa State University Press, 1977.

Hansen, James E. and Driscoll, Dennis M., "A Mathematical Model for the Generation of Hourly Temperatures", Journal of Applied Meteorology, September 1977.

Harris, R.I., "Paper 3: The Nature of Wind,"The Modern Design of Wind-sensitive Structures. Proceedings of the seminar held on June 18, 1970 at the Institution of Civil Engineers, published by Construction Industry Research and Information Association, London, 1971.

Hildebrand, Francis B., Advanced Calculus for Applications, Prentice Hall, Inc., 1976.

Hill, R.W., and Hanks, R.J., "A Model for Predicting Crop Yields from Climatic Data", ASAE, 78-4030 Transactions of the American Society of Agricultural Engineers, 1978.

Houghton, E.L. and Carruthers, N.B., Wind Forces on Buildings and Structures: an Introduction, John Wiley & Sons, 1976.

Howard, Keith A., and Smith, George T., editors, Climate Variation and its Effects on Our Land and Water, Part C, Geological Survey Climate Plan, USGS circular 776-C, 1978.

Howell, T., Optimization of Grain Sorghum Water Use Efficiency Under High Frequency Irrigation by System Simulation and Stochastic Dynamic Programming, PhD thesis, Agricultural Engineering, Texas A&M University, December 1974.

Hughes, Charles E., Pfleeger, Charles P., and Rose, Lawrence L., Advanced Programming Techniques, John Wiley, 1978.

Idso, S.B. and Jackson, R.D., "Thermal Radiation from the Atmosphere," Journal of Geophysical Research, 74, 5397-5403, 1969.

Idso, S.B. and Blad, B.L., "The Effect of Air Temperature upon Net and Solar Radiation Relations", Journal of Applied Meteorology, 10, 604-605, 1971.

Idso, S.B., "A Set of Equations for Full Spectrum and 8- to 14μm and 10.5- to 12.5-μm Thermal Radiation From Cloudless Skies", Water Resources Research, Vol. 17, No. 2, April 1981.

Jones, J.W. and Smerage, G.H., "Representation of Plant/Crop Physiology," Technical Paper 78-4024, American Society of Agricultural Engineers, 1978.

Kim, Kwonshik, A Stochastic Approach to Snowmelt Runoff Forecasting, PhD thesis, University of Minnesota, March 1976.

Jones, J.W., Colwick, R.F. and Threadgill, E.D., "A Simulated Environmental Model of Temperature Evaporation, Rainfall, and Soil Moisture," Transactions of the American Society of Agricultural Engineers, Vol. 15, No. 2, 1972, pp. 366-372.

Keitz, Edwin, and Berks, Dorothy, Living with Climatic Change, Phase II, Mitre Corporation, MTR-7443, January 1977.

Kimball, H. H., "Measurements of Solar Radiation Intensity and Determination of its Depletion by the Atmosphere," Monthly Weather Review, Vol. 55 (1927), Vol. 56 (1928), Vol. 58 (1930).

Klein, W. H., "Calculation of Solar Radiation and the Solar Heat Load on Man", Journal of Meteorology, Vol. 5, No. 4, 1948.

Kung, E.C., Climatology of Aerodynamic Roughness Parameter and Energy Dissipation in the Planetary Boundary Layer of the Northern Hemisphere, Annual Report, Contract DA-36-039-AMC-00878, Department of Meteorology, University of Wisconsin, 1963.

Lane, L.J., and Osborn, H.B., "Hypotheses on the Seasonal Distribution of Thunderstorm Rainfall in Southeastern Arizona," Floods and Droughts. Proceedings of Second International Symposium on Hydrology, September 11-13, 1972, Fort Collins, Colorado, 1973.

Leith, C.E., "Theoretical Skill of Monte Carlo Forecasts," Monthly Weather Review, 102(6), 1974.

Leith, C.E., "The Standard Error of Time-Average Estimates of Climatic Means," Journal of Applied Meteorology, 12, 1066-1069, 1973.

Linsley, R.K., Kohler, M.A. and Paulhus, J.L.H., Hydrology for Engineers, McGraw-Hill, 1975.

List, R.T., Smithsonian Meteorological Tables, 6th Edition, Washington, D.C., 1963.

Lorenz, Edward N., Personal Communication, Massachusetts Institute of Technology, Cambridge, MA 1978.

Lumley, J.L., and Panofsky, H.A., The Structure of Atmospheric Turbulence, Interscience Publishers, John Wiley & Sons, New York, 1964.

Manabe, S. and Holloway, J. L. Jr., "The Seasonal Variation of the Hydrologic Cycle as Simulated by a Global Model of the Atmosphere," Journal of Geophysical Research, Vol. 80, No. 12, April 1975, pp 1617-1649.

Manabe, S., "Climate and the Ocean Circulation, 1. The Atmospheric Circulation and the Hydrology of the Earth's Surface," Monthly Weather Review, Vol. 97, 1969, pp 739-774.

McVehil, G.E., "Wind and Temperature Profiles Near the Ground in Stable Conditions," Quarterly Journal of the Royal Meteorological Society, London, England, Vol. 90, 1964.

Milly, P. Christopher D., "Moisture and Heat Transport in Hysteretic, Inhomogeneous Porus Media: A Matric Head-based Formulation and A Numerical Model", in press, Water Resources Research, 1982.

Milly, P. Christopher D., Moisture and Heat Fluxes Across the Land Surface: Parameterization for Atmospheric General Circulation Models and the Effects of Spatial Variability, PhD thesis, Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1982.

Mishoe, J. W., "Model Growth and Development of Celery," Technical Paper 78-4032, American Society of Agricultural Engineers, 1978.

Mitchell, J. Murry, Jr., "An Overview of Climatic Variability and its Causal Mechanisms," Quaternary Research, 6, 481-493, 1976.

Monji, N., and Businger, J.A., "Stability Dependence of Temperature, Humidity, and Vertical Wind Velocity Variances in the Atmospheric Surface Layer," Journal of the Meteorological Society of Japan, Vol. 50, No. 2, 1972.

Monin, A.S. and Obukhov, A.M., "Basic Laws of Turbulent Mixing in the Surface Layer of the Atmosphere," Akad. Nauk, SSSR Trudy Geofizicheskii Institut, No. 24, 1954. (Available from Atmospheric Science library, NOAA, Silver Spring, MD).

Myrup, Leonard O., "A Numerical Model of the Urban Heat Island," Journal of Applied Meteorology, Vol. 8, No. 6, December 1969, pp 908-918.

Nicks, Arlin D., Stochastic Generation of Hydrologic Model Inputs, Xerox University Microfilms, 76-3120, 1975.

Oke, T.R., "Turbulent Transport Near the Ground in Stable Conditions," Journal of Applied Meteorology, Vol. 9, 1970.

Oke, T.R., Boundary Layer Climates, Halsted Press, 1978.

Outcult, S.I., "A Numerical Surface Climate Simulator," Geographical Anal III Vol. 4, pp 379-392.

Outcult, S.I., "The Development and Application of a Simple Surface Climate Simulator," Journal of Applied Meteorology, Vol. 11, pp 629-636.

Paulson, C.A., Profiles of Wind Speed Temperature, and Humidity Over the Sea, Science Report, NSF GP-2418, Department of Atmospheric Sciences, University of Washington, 1967.

Pease, R.W., Jenner, C.B. and Lewis, J.E., "The Impact of Land Use on Climate: An Analysis of the Washington-Baltimore Area that Couples Remote Sensing with Numerical Simulation," U.S.G.S. Professional Paper, 1978.

Polowchak, Van M. and Panofsky, H.A., The Spectrum of Daily Temperatures as a Climate Indicator, Vol. 96, September 1968.

Priestly, C.H.B., Turbulent Transfer in the Lower Atmosphere, University of Chicago Press, Chicago, IL, Vol. 130, 1959.

Rao, Ramachandra A., and Chenchayya, B.T., Probabalistic Analysis and Simulation of the Short Time Increment Rainfall Process, Technical Report No. 55, Purdue University, Water Resources Research Center, December 1974.

Rasmussen, L.A., "Notes on the Approximation of Saturation Vapor Pressure," Journal of Applied Meteorology, Vol. 17, pp 1564-1565, October 1978.

Reiton, C.H., "Distribution of Precipitable Water Vapor Over the Continental United States." Bulletin of The American Meteorological Society, Vol. 41, No. 2, 1960.

Restrepo-Posada, Pedro J., and Eagleson, Peter S., "Identification of Independent Rainstorms," Journal of Hydrology, 55, pp 303-319, 1982.

Richardson, C.W., "Stochastic Simulation of Daily Precipitation Temperature, and Solar Radiation," Water Resources Research, Vol. 17, No. 1, pp 182-190, February 1981.

Richardson, L.F., "The Supply of Energy From and To Atmospheric Eddies," Proceedings of the Royal Society of London, Series A, London, England, Vol. 97, 1920.

Rose, C.W., Agricultural Physics, Pergamon Press, London, 230 pp, 1966.

Sariahmed, A. and Kisiel, C.C., "Synthesis of Sequences of Summer Thunderstorm Volumes for the Atterbury Watershed in the Tucson Area," Proc. IAHS Symposium of the Use of Analog and Digital Computers in Hydrology, Vol. 2, pp 439-447, 1968.

Sasamori, Takashi, "A Numerical Study of Atmospheric and Soil Boundary Layers," Journal of the Atmospheric Sciences, Vol. 27, pp 1122-1137, November 1970.

Saugier, B. and Ripley, E.A., "Evaluation of the Aerodynamic Method of Determining Fluxes over Natural Grassland," Quarterly Journal of the Royal Meteorological Society, 104, 1978.

Sellers, Piers J. and Lockwood, John G., "A Numerical Simulation of the Effects of Changing Vegetation Type on Surface Hydroclimatology," Climatic Change, Vol. 3, No. 2, pp 121-136, 1981.

Sellers, W.D., "A New Global Climatic Model," Journal of Applied Meteorology, Vol. 12, No. 2, pp 241-254, March 1973.

Smagorinsky, Joseph, "Global Atmospheric Modelling and the Numerical Simulation of Climate", Climate Modification, Edited by W. N. Hess, Wiley-Interscience, 1974.

Smith, George I., Editor, Climate Variation and its Effects on Our Land and Water, Part A: Earth Science in Climate Research, USGS Circular 776-A, 1978.

Smith, George I., Editor, Climate Variation and its Effects on Our Land and Water, Part B: Current Research by the Geological Survey, USGS Circular, 776-B, 1978.

Somerville, R.C.J., et.al., "The GISS Model of the Global Atmosphere," Journal of the Atmospheric Sciences, Vol. 31, pp 84-117, 1974.

Song, Charles C.S., Pabst, Arthur F., and Bowers, C. Edward, "Stochastic Analysis of Air and Water Temperatures," Journal of the Environmental Engineering Division, Proc. American Society of Civil Engineers, Vol. 99, No. EE6, December 1973.

Swinbank, W.C.,,"Long-Wave Radiation from Clear Skies," Quarterly Journal of the Royal Meteorological Society of London, Vol. 89, July 1963.

Sutton, O.G., Micrometeorology, McGraw-Hill, New York, 333 pp, 1953.

Sutton, O.G., Atmospheric Turbulence, Second Edition, Methuen, London, 111 pp, 1954.

Taylor, R. J., "Small-scale Advection and the Neutral Wind Profile," Journal of Fluid Mechanics, (1962), 13, 529-539.

Tennessee Valley Authority, Heat and Mass Transfer Between a Water Surface and the Atmosphere, Laboratory Report No. 14, April 1972.

Thom, H. C. S., Direct and Inverse Tables of the Gamma Distribution, Environmental Data Service, EDS-2, U. S. Department of Commerce, April 1968.

Wiesner, C. J., Hydrometeorology, Chapmen & Hall, Ltd., London, 1970.

Yamamoto, G. and Shimanuki, A., "Turbulent Transfer in Diabatic Conditions", Journal of the Meteorological Society of Japan, Vol. 44, No. 6, 1966.

Yevjevich, V., "Structural Analysis of Hydrologic Time Series," Hydrology Paper 56, Colorado State University, Fort Collins, Colorado, 1972.

APPENDIX A

RANDOM NUMBER GENERATION

A.1 Introduction

Random numbers drawn from a variety of different distributions are required in the CSCS model. Fortunately, the stochastic behavior of the CSCS components can be generated by transformations of independent random numbers that are uniformly distributed over (0,1) (Fishman, 1973). This is important, since most computer systems have an algorithm for generating random numbers from U(0,1) resident in the system library. By using transformations of U(0,1) to yield random numbers from uniform (U(a,b)), normal, exponential, and gamma distributions, as well as any arbitrary distribution, the generality of the CSCS model is increased. The following sections outline the techniques used to generate the required random numbers for the CSCS model.

A.2 Uniform Distribution, ($a \leq x \leq b$)

The uniform probability distribution of variable, X, is defined by

$$f_X(x) = \begin{cases} \frac{1}{b-a} & , \quad a \leq x \leq b \\ 0 & , \quad \text{elsewhere} \end{cases} \tag{A-1}$$

The cumulative distribution, $F_X(x)$, is defined as

$$F_X(x) = \int_a^x \frac{du}{b-a} = \frac{x-a}{b-a} \qquad \text{(A-2)}$$

$F_X(x)$ can have any value between zero and one. Therefore, when $F_X(x)$ is represented by a random variate U from U(0,1), Equation A-2 becomes

$$U = \frac{x-a}{b-a} \qquad \text{(A-3)}$$

Solving for x gives

$$x = a + (b-a)U \qquad \text{(A-4)}$$

where x is a uniformly distributed number from U(a,b). (Fishman, 1973).

The generation procedure is to simply select U from U(0,1) and use Equation A-4 to generate x from U(a,b).

A.3 Exponential Distribution

The exponential probability distribution function can be written as

$$f_X(x) = \begin{cases} \frac{1}{\beta}e^{-x/\beta}, & 0 \le x \le \infty \\ 0, & x < 0 \end{cases} \qquad \text{(A-5)}$$

The cumulative probability distribution, $F_X(x)$, is

$$F_X(x) = \frac{1}{\beta}\int_0^x e^{-u/\beta}\,du = 1 - e^{-x/\beta} \qquad \text{(A-6)}$$

If $F_X(x)$ is represented by a random number, U, from U(0,1), Equation A-6 can be written as

$$U = 1 - e^{-x/\beta} \tag{A-7}$$

Solving for x gives

$$x = -\beta \ln(1-U) \tag{A-8}$$

Since U is a uniform variate, it's easy to see that (1-U) is also uniform. Therefore, Equation A-8 can be written as

$$\mathbf{x} = -\beta \ln(U) \tag{A-9}$$

The generation procedure is to select U from U(0,1) and use Equation A-9 to obtain the exponentially distributed variate x. (Fishman, 1973).

A.4 Normal Distribution

In the previous sections, the generating technique relied on the invertability of the appropriate cumulative probability distribution. Unfortunately, the cumulative distribution function of the normal is not analytically invertable.

The generating algorithm for normally distributed variates in the CSCS model is based on the direct transformation of uniform variates. (Fishman, 1973). Let U_1 and U_2 be independent variates from U(0,1). Then the variates

$$X_1 = (-2\ln U_1)^{\frac{1}{2}} \cos(2\pi U_2) \tag{A-10}$$

$$X_2 = (-2\ln U_1)^{\frac{1}{2}} \sin(2\pi U_2) \tag{A-11}$$

are independent and each is from a normal distribution with zero mean and unit variance. To demonstrate this, Fishman (1973) indicates that the joint probability distribution function of X_1 and X_2 is

$$f_{X_1,X_2}(x_1,x_2) = Jf_{U_1,U_2}(u_1,u_2) = J \tag{A-12}$$

$$= \frac{1}{2\pi} e^{-(x_1^{2} + x_2^{2})/2} \tag{A-13}$$

where

$$J = \begin{vmatrix} -x_1 e^{-(x_1^{2} + x_2^{2})/2} & -x_2 e^{-(x_1^{2}+x_2^{2})/2} \\ \dfrac{-x_2}{2\pi(x_1^{2}+x_2^{2})} & \dfrac{x_1}{2\pi(x_1^{2}+x_2^{2})} \end{vmatrix} \tag{A-14}$$

The joint distribution in Equation A-13 is that of two independent normal deviates, each with zero mean and unit variance.

The generating procedure is to select U_1 and U_2 from U(0,1) and use either of Equations A-10 and A-11 to yield a normally distributed variate X with zero mean and unit variance. Note that in previous sections, one uniformly distributed variate was selected for each generated random number. Now, two uniformly distributed variates are required for each normally distributed variate. Therefore, an efficiency rating can be defined as the number of "target" variates generated divided by the number of uniformly distributed variates required. Since two uniformly distributed variates are required for each standard normal deviate desired, the generating Equations A-10 and A-11 have an efficiency rating of 50%.

A.5 Gamma Distribution

Consider the variate, X, to be gamma distributed with shape parameter, α, and scale parameter, β. (denoted as $Ga(\alpha,\beta)$). The probability distribution function of the gamma variate, X, is

$$f_X(x) = \begin{cases} \dfrac{1}{\Gamma(\alpha)\beta^{\alpha}} e^{-x/\beta} x^{\alpha-1} & , \quad 0 \le x \le \infty \\ 0 & , \quad x < 0 \end{cases} \qquad \text{(A-15)}$$

Like the cumulative distribution function (cdf) of the normal distribution, the gamma cdf cannot be analytically inverted. Therefore, transformation of uniformly distributed variates will be used to generate gamma distributed random numbers.

Fishman (1973) outlines a technique to generate gamma variates that is valid for both integral and non-integral shape factors. According to Fishman, if X is from Ga(α,β), then X can be considered "to be the sum of k + 1 independent gamma variates, all with scale parameter β, but the first k of which have unit shape parameter and the k+1st has shape parameter $\gamma = \alpha - (\alpha)$." (Note that $k = (\alpha)$ where "( )" denotes "the largest integer in").

The first k independent gamma variates are from Ga(1,β). With unit shape parameter, the gamma distribution reduces to the exponential distribution. Thus, the sum of k independent gamma variates from Ga(1,β) can be expressed as the sum of k independent exponentially distributed variates. Using Equation A-9,

$$X_1 = \sum_{j=1}^{k} (-\beta \ln U_j) \qquad \text{(A-16)}$$

which can also be written as

$$X_1 = -\beta \ln \left(\prod_{j=1}^{k} U_j \right) \qquad \text{(A-17)}$$

where U_j is the jth variate selected from U(0,1).

The k + 1st variate is distributed according to $Ga(\gamma,\beta)$. To obtain the k + 1st variate, let Y and Z be independent variates from a beta distribution, $Be(\gamma,1-\gamma)$, and a gamma distribution, $Ga(1,1)$, respectively. Then, as Fishman (1973) shows, the variate $W = \beta YZ$ is distributed according to $Ga(\gamma,\beta)$. Thus, the gamma variate, X, from $Ga(\alpha,\beta)$ is found by

$$X = -\beta \ln \left(\prod_{j=1}^{k} U_j \right) + \beta YZ \qquad \text{(A-18)}$$

Since Z is exponential with a unit parameter,

$$Z = \ln U_{k+1} \qquad \text{(A-19)}$$

and Equation A-18 becomes

$$X = -\beta \ln \left(\prod_{j=1}^{k} U_j \right) - Y\beta \ln(U_{k+1}) \qquad \text{(A-20)}$$

The remaining task is to select Y from $Be(\gamma,1-\gamma)$.

The probability distribution function for a beta distributed variate with shape and scale parameters a and b respectively is

$$f_Y(y) = \begin{cases} \dfrac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\, y^{a-1}(1-y)^{b-1} & , \ 0 \le x \le 1 \\ 0 & , \ \text{elsewhere} \end{cases} \qquad \text{(A-21)}$$

In general, the parameters a and b will be nonintegral in the CSCS model, therefore, an acceptance-rejection technique for generating Y from Be(a,b) will be used, (Fishman, 1973).

Consider the transformations

$$Y_1 = U_1^{1/a} \tag{A-22}$$

and

$$Y_2 = U_2^{1/b} \tag{A-23}$$

where U_1 and U_2 are independent uniformly distributed variates from U(0,1). If $Y_1 + Y_2 \leq 1$, then Fishman shows that the variate

$$Y = \frac{Y_1}{(Y_1 + Y_2)} \tag{A-23}$$

is distributed according to Be(a,b).

To find the beta variate required by Equation A-20, first find the transformed variates

$$Y_1 = U_1^{1/\gamma} \tag{A-23}$$

$$Y_2 = U_2^{1/(1-\gamma)} \tag{A-24}$$

Next, determine if $Y_1 + Y_2 \leq 1$. If $Y_1 + Y_2 \leq 1$, then "accept" the variates Y_1 and Y_2 and compute the beta variate,

Y, using Equation A-23.

If $Y_1 + Y_2 > 1$, reject the variates Y_1 and Y_2. Select new variates, U_1 and U_2, and repeat the process until a (Y_1, Y_2) pair are accepted to compute Y. Once a valid beta variate, Y, has been identified, Equation A-20 is executed to give the required gamma variate X from $Ga(\alpha,\beta)$.

If η_β is the number of uniformly distributed variates required to generate one beta variate, the total number, η_Γ of uniformly distributed variates required to generate one gamma variate from $Ga(\alpha,\beta)$ is

$$\eta_\Gamma = \eta_\beta + k + 1 \qquad \text{(A-25)}$$

The expected value of η_Γ is then

$$E(\eta_t) = E(\eta_\beta) + E(k) + 1 \qquad \text{(A-26)}$$

Since the number of trials for success in the beta generation procedure follows the geometric distribution, the expected number of uniformly distributed variates required to generate a $Be(a,b)$ variate is

$$E(\eta_\beta) = \frac{2(a+b)\Gamma(a+b)}{ab\Gamma(a)\Gamma(b)} \qquad \text{(A-27)}$$

Substitution of $a = \gamma$ and $b = 1-\gamma$ into Equation A-27 leads to

$$E(\eta_\beta) = \frac{2}{\gamma(1-\gamma)\Gamma(\gamma)\Gamma(1-\gamma)} \tag{A-28}$$

From Hildebrand (1976), the following identity can be used to further simplify Equation A-28

$$\Gamma(\gamma)\Gamma(1-\gamma) = \frac{\pi}{\sin(\pi\gamma)} \tag{A-29}$$

Thus

$$E(\eta_\beta) = \frac{2\sin(\pi\gamma)}{(1-\gamma)\pi} \tag{A-30}$$

Equation A-30 has a maximum when $\gamma = 0.50$. Therefore, the maximum expected value of η_β is approximately 2.5.

Comparison of Equations A-15 and 2.2-3 gives

$$\alpha = \delta t_r \tag{A-31}$$

Since $k = (\alpha)$, then

$$k = (\delta t_r) \tag{A-32}$$

Taking expected values of both sides of Equation A-32 gives

$$E(k) = (\delta E(t_r)) \tag{A-33}$$

However, since t_r is exponential

$$E(t_r) = \delta^{-1} \tag{A-34}$$

and Equation A-33 becomes

$$E(k) = 1 \tag{A-35}$$

Now using the Equations A-30 and A-35, Equation A-26 becomes

$$E(\eta_t) = \frac{2\sin(\pi\gamma)}{\gamma(1-\gamma)\pi} + 2 \tag{A-36}$$

and

$$E(\eta_t)_{max} \simeq 4.5 \tag{A-37}$$

A.6 Arbitrary Distribution

Occasionally, it becomes necessary to generate a random variable from a distribution for which there is no conveniently available mathematical formula. To generate a random variate over a finite domain (a,b), the following steps are used. (Abramowitz and Stegun, 1970).

Let f be the maximum of f(y), the probability distribution function of the variate y. Generate a pair of uniform deviates, U_1 and U_2 from U(0,1). Compute a point $y = a+(b-a)U_1$ in (a,b). If $U_2 \leq f(y)/f$, accept y as the random deviate, otherwise reject the pair (U_1,U_2) and start again. The expected number of uniformly distributed variates, η_t, required to generate the appropriate random deviate is

$$E(\eta_t) = 2(b-a)f \tag{A-38}$$

In the CSCS model where this approach was used, $f(y)$ was approximated by a histogram.

APPENDIX B

STATISTICAL PROPERTIES OF N(t)

B.1 Introduction

The cloud model developed in Chapter 3 was required to have certain statistical properties. These properties were discussed in Chapter 3, but their development is presented here. The cloud cover model has the form

$$N(t) = M_o + (1-M_o)(1-P(t)) + m(t)P(t) \qquad \text{(B-1)}$$

where M_o is the "fairweather" mean value of N(t), P(t) is the storm transition function, m(t) is a serially correlated random sequence with the following characteristics

$$E(m(t)) = 0 \qquad \text{(B-2)}$$

$$VAR(m(t)) = \sigma_m^2 \qquad \text{(B-3)}$$

The sequence, m(t), also has a serial correlation function $\rho_m(\tau)$ where τ is the lag.

B.2 Expected Value of N(t)

The first required property of N(t) is its expected value. More specifically, the expected value of N(t) given the time between storms, t_b, is required. The conditional expected value of N(t) is found by

$$E(N(t)|t_b) = E(M_o + (1-M_o)(1-P(t)) + m(t)P(t)) \tag{B-4}$$

Since M_o is a constant and $P(t)$ is a deterministic function of time, Equation B-4 becomes

$$E(N(t)|t_b) = M_o + (1-M_o)(1-P(t)) + P(t)E(m(t)) \tag{B-5}$$

Substitution of Equation B-2 into Equation B-5 results in the expression for the time varying conditional expected value of cloud cover shown earlier as Equation 3.4-5

$$E(N(t)|t_b) = M_o + (1-M_o)(1-P(t)) \tag{B-6}$$

B.3 Variance of N(t)

The conditional variance of N(t) is defined as

$$VAR(N(t)|t_b) = E((N(t)|t_b - E(N(t)|t_b))^2) \tag{B-7}$$

which can also be written as

$$VAR(N(t)|t_b) = E((N(t)|t_b)^2) - E^2(N(t)|t_b) \tag{B-8}$$

First find $(N(t)|t_b)^2$.

$$(N(t)|t_b)^2 = (M_o + (1-M_o)(1-P(t)) + m(t)P(t))^2 \tag{B-9}$$

$$(N(t)|t_b)^2 = M_o^{\ 2} + 2M_o(1-M_o)(1-P(t))$$
$$+ 2m(t)P(t)M_o + (1-M_o)^2(1-P(t))^2$$
$$+ 2m(t)P(t)(1-M_o)(1-P(t)) \qquad \text{(B-10)}$$
$$+ m^2(t)P^2(t)$$

Taking expected values of both sides of Equation B-10 gives

$$E((N(t)|t_b)^2) = M_o^{\ 2} + 2M_o(1-M_o)(1-P(t))$$
$$+ (1-M_o)^2(1-P(t))^2 + \sigma_m^{\ 2}P^2(t) \qquad \text{(B-11)}$$

Since

$$E(2m(t)P(t)M_o) = 2P(t)M_oE(m(t)) = 0$$

$$E(2m(t)P(t)(1-M_o)(1-P(t))) =$$
$$= 2P(t)(1-M_o)(1-P(t))E(m(t)) = 0$$

and

$$E(m^2(t)P^2(t)) = P^2(t)E(m^2(t)) = P^2(t)\sigma_m^{\ 2}$$

For $E^2(N(t)|t_b)$, Equation B-6 is used to give

$$E^2(N(t)|t_b) = (M_o + (1-M_o)(1-P(t)))^2 \qquad \text{(B-12)}$$

Expansion of Equation B-12 and substitution into Equation B-8 along with Equation B-11 gives

$$VAR(N(t)|t_b) = M_o^2 + 2M_o(1-M_o(1-P(t))$$

$$+ \sigma_m^2 P^2(t) - M_o^2 - 2M_o(1-M_o)(1-P(t))$$

$$- (1-M_o)^2(1-P(t))^2 \qquad \text{(B-13)}$$

Equation B-13 reduces to the expression for the time varying conditional variance of N(t)

$$VAR(N(t)|t_b) = \sigma_m^2 P^2(t) \qquad \text{(B-14)}$$

B.4 Serial Correlation Function (B-15)

The serial correlation function of a time series is found by normalizing the covariance function of the time series. The covariance is defined as

$$COV(N(t),N(t+\tau)) =$$

$$E((N(t)-\mu_N(t))(N(t+\tau)-\mu_N(t+\tau))) \qquad \text{(B-16)}$$

As in previous sections, the process is conditioned by t_b. For ease in writing, the designator "$|t_b$" has been dropped. Also, for convenience

$$\mu_N(t) \equiv E(N(t)|t_b) \qquad \text{(B-17)}$$

Expansion of Equation B-16 leads to

$$COV(N(t),N(t+\tau)) = E(N(t)N(t+\tau))$$

$$-E(N(t)\mu_N(t+\tau)) - E(\mu_N(t)N(t+\tau))$$

$$+E(\mu_N(t)\mu_N(t+\tau)) \qquad \text{(B-18)}$$

Since $\mu_N(t)$ and $\mu_N(t+\tau)$ are deterministic functions of time, the third and fourth terms on the right-hand side of Equation B-18 become respectively

$$E(\mu_N(t)N(t+\tau)) = \mu_N(t)\mu_N(t+\tau) \qquad \text{(B-19)}$$

and

$$E(\mu_N(t)\mu_N(t+\tau)) = \mu_N(t)\mu_N(t+\tau) \qquad \text{(B-20)}$$

Substitution of Equations B-19 and B-20 into Equation B-18 gives

$$COV(N(t),N(t+\tau)) = E(N(t)N(t+\tau)) \qquad \text{(B-21)}$$

$$-\mu_N(t+\tau)E(N(t))$$

The next step is to substitute Equation B-1 evaluated at times t and $t+\tau$ into Equation B-21. This leads to

$$COV(N(t+\tau)) = P(t)P(t+\tau)E(m(t)m(t+\tau)) \qquad \text{(B-22)}$$

The serial correlation function of N(t) is defined as

$$\rho_N(\tau) = \frac{COV(N(t),N(t+\tau))}{\sigma_N(t)\sigma_N(t+\tau)} \qquad \text{(B-23)}$$

where $\sigma_N(t)$ is the standard deviation of the process at time t. The standard deviations are defined as

$$\sigma_N(t) = \sqrt{VAR(N(t)|t_b)} = P(t)\sigma_m \qquad \text{(B-24)}$$

and

$$\sigma_N(t+\tau) = \sqrt{VAR(N(t+\tau)|t_b)} = P(t+\tau)\sigma_m \qquad (B-25)$$

Substitution of Equations B-22, B-24, and B-25 into Equation B-23 yields

$$\rho_N(\tau) = \frac{P(t)P(t+\tau)E(m(t)m(t+\tau))}{P(t)P(t+\tau)\sigma_m^{\ 2}} \qquad (B-26)$$

With the definition

$$E(m(t)m(t+\tau)) = COV(m(t)m(t+\tau)) \qquad (B-27)$$

Equation B-26 can be written as

$$\rho_N(\tau) = \frac{COV(m(t)m(t+\tau))}{\sigma_m^{\ 2}} \qquad (B-28)$$

The right-hand side of Equation B-28 is just the definition of the serial correlation function, $\rho_m(\tau)$, of the random process m(t). Therefore, Equation B-28 reduces to

$$\rho_N(\tau) = \rho_m(\tau) \qquad (B-29)$$

Equation B-29 states that the process N(t), whose mean and variance are modulated in a controlled fashion by P(t), will have a serial correlation function identical to the process m(t).

APPENDIX C

BRYAN'S TEMPERATURE FORECAST MODEL

C.1 Introduction

Since Bryan's 1964 report was unpublished and since the writer knows of no formal presentation of the details of Bryan's technique in the literature, a detailed mathematical description of the approach will be included here.

Bryan's approach is represented by the following equations

$$\frac{d}{dt} T(t) + b_1 T(t) = b_o + b_2 s(t) + b_3 r(t) \qquad \text{(C-1)}$$

$$s(t) = \sin\delta\sin\phi - \cos\delta\cos\phi\cos\frac{\pi}{12}t) \quad , \quad (R \le t \le S) \qquad \text{(C-2)}$$

$$s(t) = 0 \quad , \quad \text{otherwise} \qquad \text{(C-3)}$$

$$r(t) = \frac{\pi}{12}\cos\delta\cos\phi\sin(\frac{\pi}{12}t) \quad , \quad (R \le t \le 12) \qquad \text{(C-4)}$$

$$r(t) = 0 \quad , \quad \text{otherwise} \qquad \text{(C-5)}$$

where T(t) is the temperature at time t, δ is the solar declination, ϕ is the local latitude, R is the local time of sunrise (note the difference between local time and standard time), and S is the local time of sunset.

Equation C-1 can be solved by using the integrating factor $e^{b_1 t}$. Thus

$$\frac{d}{dt}\left[e^{b_1 t}T(t)\right] = e^{b_1 t}\left[b_o + b_2 s(t) + b_3 r(t)\right] \quad \text{(C-6)}$$

and for the interval (t',t)

$$T(t) = T(t')e^{-b_1(t-t')} + e^{-b_1 t}F(t,t') \quad \text{(C-7)}$$

where

$$F(t) = b_o \int_{t'}^{t} e^{b_1 \tau} d\tau + b_2 \int_{t'}^{t} e^{b_1 \tau} s(\tau) d\tau + b_3 \int_{t'}^{t} e^{b_1 \tau} r(\tau) d\tau \quad \text{(C-8)}$$

Equations C-7 and C-8 represent the solution to Equation C-1. Once the coefficients, b_i, are known, a temperature forecast can be made given only the initial temperature T(t').

C.2 Parameter Estimation

Bryan manipulated Equations C-7 and C-8 into a form that leads to a linear regression formula used to estimate the b_i's.

First, note the following identities

$$e^{-b_1 t} = e^{-b_1} e^{-b_1(t-1)} \quad \text{(C-9)}$$

$$e^{-b_1(t-t')} = e^{-b_1(t-1-t')} \quad \text{(C-10)}$$

$$F(t,t') = F(t-1,t') + F(t,t-1) \tag{C-11}$$

Using these identities, Equation C-8 can be rewritten as

$$T(t) = e^{-b_1}\left[T(t')e^{-b_1(t-1-t')} + e^{-b_1(t-1)}F(t-1,t')\right]$$

$$+ e^{-b_1 t}F(t,t-1) \tag{C-12}$$

The quantity inside the brackets is just T(t-1). Therefore,

$$T(t) = e^{-b_1}T(t-1) + e^{-b_1 t}F(t,t-1) \tag{C-13}$$

The hourly temperature change, Y(t), is found by subtracting T(t-1) from both sides of Equation C-13.

$$Y(t) = -(1-e^{-b_1})T(t-1) + e^{-b_1 t}F(t,t-1) \tag{C-14}$$

Substitution for F(t,t-1) leads to

$$Y(t) = b_o e^{-b_1 t}\int_{t-1}^{t} e^{b_1\tau}d\tau - (1-e^{-b_1})T(t-1)$$

$$+ b_2 e^{-b_1 t}\int_{t-1}^{t} e^{b_1\tau}s(\tau)d\tau$$

$$+ b_3 e^{-b_1 t}\int_{t-1}^{t} e^{b_1\tau}r(\tau)d\tau \tag{C-15}$$

Evaluation of the first integral (I_1 for convenience) on the right-hand side of Equation C-15 leads to

$$I_1 = \frac{b_o}{b_1}(1-e^{-b_1}) \qquad \text{(C-16)}$$

The last two integrals, I_2 and I_3, on the right-hand side of Equation C-15 are complicated by the exponential term inside the integral. Bryan indicated that it was sufficient to use the mean value of $e^{b_1\tau}$ and bring it outside the integral. Thus,

$$E\left(e^{b_1\tau}\right) = \frac{1}{b_1}(1-e^{-b_1})e^{b_1 t} \qquad \text{(C-17)}$$

Thus, I_2 and I_3 respectively, become

$$I_2 = \frac{b_2}{b_1}(1-e^{-b_1}) \int_{t-1}^{t} s(\tau)d\tau \qquad \text{(C-18)}$$

and

$$I_3 = \frac{b_3}{b_1}(1-e^{-b_1}) \int_{t-1}^{t} s(\tau)d\tau \qquad \text{(C-19)}$$

Substitution of the expressions for I_1, I_2, and I_3, back into Equation C-15 yields

$$Y(t) = \frac{b_o}{b_1}(1-e^{-b_1}) - (1-e^{-b_1})T(t-1)$$
$$+ \frac{b_2}{b_1}(1-e^{-b_1}) \int_{t-1}^{t} s(\tau)d\tau$$
$$+ \frac{b_3}{b_1}(1-e^{-b_1}) \int_{t-1}^{t} r(\tau)d\tau \qquad \text{(C-20)}$$

Equation C-20 is now in the required regression form from which the b_i's can be estimated. To see this more clearly, compare Equation C-20 term by term with the following

$$Y(t) = a_o + a_1X_1(t) + a_2X_2(t) + a_3X_3(t) \qquad \text{(C-21)}$$

The comparison gives for the coefficients

$$a_o = \frac{b_o}{b_1}(1-e^{-b_1}) \qquad \text{(C-22)}$$
$$a_1 = -(1-e^{-b_1}) \qquad \text{(C-23)}$$
$$a_2 = \frac{b_2}{b_1}(1-e^{-b_1}) \qquad \text{(C-24)}$$
$$a_3 = \frac{b_3}{b_1}(1-e^{-b_1}) \qquad \text{(C-25)}$$

For the predictors

$$X_1(t) = T(t-1) \tag{C-26}$$

$$X_2(t) = \int_{t-1}^{t} s(\tau)d\tau \tag{C-27}$$

$$X_3(t) = \int_{t-1}^{t} r(\tau)d\tau \tag{C-28}$$

Once the a_i's have been determined by regression, the b_i's can easily be found, since the set of Equations C-22 through C-25 is a set of four equations in four unknowns. Therefore, the b_i's can be found from

$$b_1 = -\ln(a_1+1) \tag{C-29}$$

$$b_i = -\frac{b_1}{a_1}a_i \quad , \quad i = 0,2,3 \tag{C-30}$$

Now that the b_i's are established, Equation C-7 can be used to forecast temperatures, given only the initial temperature, $T(t')$.

C.3 Evaluation of Predictors

From the definitions of $s(t)$ and $r(t)$, it is seen that Equation C-20 and, ultimately Equation C-7, will have different forms, depending upon the time of day. The ranges over which each form will be valid are delimited by several "critical" times. These times must be identified in order to coordinate the data observation times which occur at regular intervals according to standard time, and the occurrence of events in the local solar day (e.g. sunrise, sunset, etc.) which vary in time throughout the year. Five critical times are identified: 1) t_o is the value of t in local time corresponding to midnight in standard time, 2) r_s is the value of t which corresponds to the earliest standard hour that does not precede local sunrise, R $(r_s \geq R)$, 3) t_{12} is the value of t at the earliest standard hour that does not precede local noon $(t_{12} \geq 12)$, 4) s_s is the value of t at the earliest standard hour that does not precede local sunset, S $(s_s \geq S)$, and, 5) t_{23} is the value of t corresponding to 11:00 p.m. local standard time.

For all times, t, predictor $X_1(t)$ will equal $T(t-1)$. But the forms of $X_2(t)$ and $X_3(t)$ will change with t. The individual forms of $X_2(t)$ and $X_3(t)$ for each range follow.

<u>Range 1 $t_o \leq t \leq r_s - 1$ for $X_2(t)$</u>

$$X_2(t) = \int_{t_o}^{t} s(\tau)d\tau$$

$$X_2(t) = 0 \tag{C-31}$$

Range 2 $t = r_s$ (first observation hour after sunrise) For $X_2(t)$

$$X_2(t) = \int_{t-1}^{t} s(\tau)d\tau$$

$$= \int_{t-1}^{R} s(\tau)d\tau + \int_{R}^{r_s} s(\tau)d\tau \qquad (C\text{-}32)$$

$$X_2(t) = (r_s - R)\sin\delta\sin\phi - \frac{12}{\pi}\cos\delta\cos\phi(\sin\frac{\pi r_s}{12} - \frac{\pi R}{12}) \qquad (C\text{-}33)$$

Range 3 $r_s+1 \leq t \leq s_s-1$ (daylight hours) for $X_2(t)$

$$X_2(t) = \int_{t-1}^{t} s(\tau)d\tau$$

$$X_2(t) = \sin\delta\sin\phi - \frac{12}{\pi}\cos\delta\cos\phi(\sin\frac{\pi t}{12} - \sin\frac{\pi}{12}(t-1)) \qquad (C\text{-}34)$$

Range 4 $t = s_s$ for $X_2(t)$ (near sunset) for $X_2(t)$

$$X_2(t) = \int_{s_{s-1}}^{s_s} s(\tau)d\tau$$

$$= \int_{s_s-1}^{S} s(\tau)d\tau + \int_{S}^{s_s} s(\tau)d\tau$$

$$X_2(t) = (S-s_s+1)\sin\delta\sin\phi$$
$$- \frac{12}{\pi}\cos\delta\cos\phi(\cos\frac{\pi}{12}S - \cos\frac{\pi}{12}(s_s-1)) \qquad \text{(C-35)}$$

<u>Range 5 $s_s+1 \leq t \leq t_{23}$ (after sunset) for $X_2(t)$</u>

$$X_2(t) = \int_{t-1}^{t} s(\tau)d\tau$$

$$X_2(t) = 0 \qquad \text{(C-36)}$$

Similar ranges exist for $X_3(t)$.

<u>Range 1 $t_o \leq t \leq r_s-1$ (before sunrise) for $X_3(t)$</u>

$$X_3(t) = \int_{t-1}^{t} r(\tau)d\tau$$

$$X_3(t) = 0 \qquad \text{(C-37)}$$

<u>Range 2 $t = r_s$ (near sunrise) for $X_3(t)$</u>

$$X_3(t) = \int_{r_s-1}^{r_s} r(\tau)d\tau$$

$$= \int_{r_s-1}^{R} r(\tau)d\tau + \int_{R}^{r_s} r(\tau)d\tau$$

$$X_3(t) = \cos\delta\cos\phi(\cos\frac{\pi R}{12} - \cos\frac{\pi r_s}{12}) \quad \bullet \qquad (C\text{-}38)$$

Range 3, $r_s + 1 \leq t \leq t_{12} - 1$ (before noon) for $X_3(t)$

$$X_3(t) \int_{t-1}^{t} r(\tau)d\tau$$

$$X_3(t) = \cos\delta\cos\phi(\cos\frac{\pi}{12}(t-1) - \cos\frac{\pi}{12}t) \qquad (C\text{-}39)$$

Range 4, $t = t_{12}$ (near noon) for $X_3(t)$

$$X_3(t) = \int_{t_{12-1}}^{t_{12}} r(\tau)d\tau$$

$$= \int_{t_{12-1}}^{12} r(\tau)d\tau + \int_{12}^{t_{12}} r(\tau)d\tau$$

$$X_3(t) = \cos\delta\cos\phi(\cos\frac{\pi}{12}(t_{12}\ 1) + 1) \qquad (C\text{-}40)$$

Range 5, $t_{12}+1 \leq t \leq t_{23}$ for $X_3(t)$

$$X_3(t) = \int_{t-1}^{t} r(\tau)d\tau$$

$$X_3(t) = 0 \qquad (C\text{-}41)$$

For each hour of the day, the hourly temperature change, Y(t), is computed from the observed data and the predictors $X_i(t)$ are evaluated. Standard linear regression techniques can be used to estimate the coefficients a_i which in turn are used to finally yield the b_i's.

C.4 Evaluation of F(t,t')

As with the predictors $X_i(t)$, the function $F(t,t')$ will have different forms, depending on the time of day. The general solution for $F(t,t')$ will be shown first. Then the individual forms applicable in each range will be developed.

Consider again Equation C-8, where

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1\tau} d\tau + b_2 \int_{t'}^{t} e^{b_1\tau} s(\tau) d\tau + b_3 \int_{t'}^{t} e^{b_1\tau} r(\tau) d\tau \qquad \text{(C-8)}$$

For convenience, let

$$F(t,t') = I_1 + I_2 + I_3 \qquad \text{(C-42)}$$

where

$$I_1 = b_o \int_{t'}^{t} e^{b_1\tau} d\tau \qquad \text{(C-43)}$$

$$I_2 = b_2 \int_{t'}^{t} e^{b_1 \tau} s(\tau) d\tau \qquad \text{(C-44)}$$

$$I_3 = b_3 \int_{t'}^{t} e^{b_1 \tau} r(\tau) d\tau \qquad \text{(C-45)}$$

Evaluation of I_1 is straightforward and can be written directly as

$$I_1 = \frac{b_o}{b_1} (e^{b_1 t} - e^{b_1 t'}) \qquad \text{(C-46)}$$

For I_2, begin by substituting the full expression for $s(\tau)$ inside the integral. Thus,

$$I_2 = b_2 \int_{t'}^{t} e^{b_1 \tau} (\sin\delta\sin\phi - \cos\delta\cos\phi\cos(\tfrac{\pi}{12})) d\tau \qquad \text{(C-47)}$$

The declination δ is actually a function of time and, in a strict sense, ought to be evaluated in the integral. However, the interval (t,t') is sufficiently short so that the variation in δ is ignored. Equation C-47 can now be rewritten as

$$I_2 = b_2 \sin\delta\sin\phi \int_{t'}^{t} e^{b_1 \tau} d\tau - b_2 \cos\delta\cos\phi \int_{t'}^{t} e^{b_1 \tau} \cos(\tfrac{\pi\tau}{12}) d\tau \qquad \text{(C-48)}$$

Completion of the integration finally yields

$$I_2 = \frac{b_2}{b_1} \sin\delta\sin\phi(e^{b_1 t} - e^{b_1 t'})$$

$$- \frac{e^{b_1 t} b_1 b_2 \cos\delta\cos\phi\cos(\frac{\pi t}{12})}{b_1^2 + (\frac{\pi}{12})^2}$$

$$- \frac{e^{b_1 t} b_2 (\frac{\pi}{12})\cos\delta\cos\phi\sin(\frac{\pi t}{12})}{b_1^2 + (\frac{\pi}{12})^2}$$

$$+ \frac{e^{b_1 t'} b_1 b_2 \cos\delta\cos\phi\cos(\frac{\pi t'}{12})}{b_1^2 + (\frac{\pi}{12})^2}$$

$$+ \frac{e^{b_1 t'} b_2 (\frac{\pi}{12})\cos\delta\cos\phi\sin(\frac{\pi t'}{12})}{b_1^2 + (\frac{\pi}{12})^2} \qquad \text{(C-49)}$$

Similarly, for I_3, substitute the full expression for $r(\tau)$ into the integral.

$$I_3 = b_3 \int_{t'}^{t} e^{b_1 \tau} (\frac{\pi}{12} \cos\delta\cos\phi\sin \frac{\pi\tau}{12}) d\tau \qquad \text{(C-50)}$$

Again, the short term variation in δ is ignored. Thus

$$I_3 = b_3(\frac{\pi}{12})\cos\delta\cos\phi \int_{t'}^{t} e^{b_1\tau}\sin(\frac{\pi\tau}{12})d\tau \qquad \text{(C-51)}$$

and

$$I_3 = \frac{\frac{\pi}{12}\, b_1 b_3 \cos\delta\cos\phi}{b_1^{\,2}+(\frac{\pi}{12})^2}\, e^{b_1 t}\sin(\frac{\pi t}{12})$$

$$- \frac{b_3(\frac{\pi}{12})^2\cos\delta\cos\phi}{b_1^{\,2}+(\frac{\pi}{12})^2}\, e^{b_1 t}\cos(\frac{\pi t}{12})$$

$$- \frac{\frac{\pi}{12}\, b_1 b_3 \cos\delta\cos\phi}{b_1^{\,2}+(\frac{\pi}{12})^2}\, e^{b_1 t'}\sin(\frac{\pi t'}{12})$$

$$+ \frac{b_3(\frac{\pi}{12})^2\cos\delta\cos\phi}{b_1^{\,2}+(\frac{\pi}{12})^2}\, e^{b_1 t'}\cos(\frac{\pi t'}{12}) \qquad \text{(C-52)}$$

To simplify the writing of Equations C-49 and C-52, the following definitions are used

$$p = \frac{\pi}{12} \qquad \text{(C-53)}$$

$$K_1 = \frac{b_o}{b_1} \qquad \text{(C-54)}$$

$$K_2 = \frac{b_2\sin\delta\sin\phi}{b_1} \qquad \text{(C-55)}$$

$$K_3 = \frac{b_1 b_2 \cos\delta \cos\phi}{b_1^{\,2} + p^2} \tag{C-56}$$

$$K_4 = \frac{p b_2 \cos\delta \cos\phi}{b_1^{\,2} + p^2} \tag{C-57}$$

$$K_5 = \frac{p^2 b_3 \cos\delta \cos\phi}{b_1^{\,2} + p^2} \tag{C-58}$$

$$K_6 = \frac{p b_1 b_3 \cos\delta \cos\phi}{b_1^{\,2} + p^2} \tag{C-59}$$

Using the definitions in Equations C-46, C-49, and C-52, the general form of $F(t,t')$ can be written as

$$\begin{aligned} F(t,t') = {} & K_1(e^{b_1 t} - e^{b_1 t'}) + K_2(e^{b_1 t} - e^{b_1 t'}) \\ & - (K_3+K_5)e^{b_1 t}\cos(pt) + (K_6-K_4)e^{b_1 t}\sin(pt) \\ & + (K_3+K_5)e^{b_1 t'}\cos(pt') + (K_4-K_6)e^{b_1 t}\sin(pt') \end{aligned} \tag{C-60}$$

Note that $t' = t_o - 1$.

For the range $t_o \leq t \leq R$

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1 \tau} d\tau + b_2 \int_{t'}^{t} e^{b_1 \tau} s(\tau) d\tau$$

$$+ b_3 \int_{t'}^{t} e^{b_1 \tau} r(\tau) d\tau \tag{C-61}$$

In this range s(t) and r(t) are both zero. Thus,

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1 \tau} d\tau$$

$$F(t,t') = K_1 (e^{b_1 t} - e^{b_1 t'}) \tag{C-62}$$

For the range $R \leq t \leq 12$

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1 \tau} d\tau + b_2 \int_{t'}^{t} e^{b_1 \tau} s(\tau) d\tau$$

$$+ b_3 \int_{t'}^{t} e^{b_1 \tau} r(\tau) d\tau \tag{C-63}$$

Equation C-63 can also be written as

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1 \tau} + b_2 \int_{t'}^{R} e^{b_1 \tau} s(\tau) d\tau$$

$$+ b_2 \int_{R}^{t} e^{b_1 \tau} s(\tau) d\tau + b_3 \int_{t'}^{t} e^{b_1 \tau} r(\tau) d\tau$$

$$+ b_3 \int_R^t e^{b_1\tau} r(\tau)d\tau$$

Remembering that prior to sunrise, s(t) and r(t) are zero, carrying out the integration leads to

$$F(t,t') = K_1(e^{b_1 t} - e^{b_1 t'}) + K_2(1-e^{-b_1(t-R)})$$

$$- (K_3+K_5)\cos(pt) + (K_6-K_4)\sin(pt)$$

$$+ (K_3+K_5)e^{-b_1(t-R)}\cos(pR)$$

$$+ (K_4-K_6)e^{-b_1(t-R)}\sin(pR) \qquad \text{(C-64)}$$

For the range $12 \leq t \leq S$

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1\tau} d\tau + b_2 \int_{t'}^{R} e^{b_1\tau} s(\tau)d\tau$$

$$+ b_2 \int_R^t e^{b_1\tau} s(\tau)d\tau$$

$$+ b_3 \int_{t'}^{R} e^{b_1\tau} r(\tau)d\tau + b_3 \int_{R}^{12} e^{b_1\tau} r(\tau)d\tau$$

$$+ b_3 \int_{12}^{t} e^{b_1\tau} r(\tau)d\tau \tag{C-65}$$

Evaluation of Equation C-65 gives

$$F(t,t') = K_1(e^{b_1t} - e^{b_1t'}) + K_2(e^{b_1t} - e^{b_1R})$$

$$- K_3e^{b_1t}\cos(pt) + K_4e^{b_1t}\sin(pt)$$

$$+ K_3e^{b_1R}\cos(pR) + K_4e^{b_1R}\sin(pR)$$

$$- K_6e^{b_1R}\sin(pR) + K_5e^{b_1R}\cos(pR)$$

$$+ K_5e^{12b_1} \tag{C-66}$$

For the range $S \leq t \leq t_{23}$

$$F(t,t') = b_o \int_{t'}^{t} e^{b_1\tau}d\tau + b_2 \int_{t'}^{t} e^{b_1\tau} s(\tau)d\tau$$

$$+ b_2 \int_R^S e^{b_1\tau} s(\tau)d\tau + b_2 \int_S^t e^{b_1\tau} s(\tau)d\tau$$

$$+ b_3 \int_{t'}^R e^{b_1\tau} r(\tau)d\tau + b_3 \int_R^{12} e^{b_1\tau} r(\tau)d\tau$$

$$+ b_3 \int_{12}^t e^{b_1\tau} r(\tau)d\tau \qquad \text{(C-67)}$$

During the evaluation of the integrals in Equation C-67, the following identities prove useful

$$S = 24-R \qquad \text{(C-68)}$$

$$\sin(2\pi-pR) = -\sin(pR) \qquad \text{(C-69)}$$

$$\cos(2\pi-pR) = \cos(pR) \qquad \text{(C-70)}$$

The final form for F(t,t') is now written as

$$F(t,t') = K_1(e^{b_1t}-e^{b_1t'}) + K_2(e^{b_1S}-e^{b_1R})$$

$$- K_3e^{b_1S}\cos(pR) + K_4e^{b_1S}\sin(pR) + K_5e^{12b_1}$$

$$+ K_3e^{b_1R}\cos(pR) + K_4e^{b_1R}\sin(pR)$$

$$- K_6e^{b_1R}\sin(pR) + K^5e^{b_1R}\cos(pR) \qquad \text{(C-71)}$$

The forecast temperatures are now found by substituting the appropriate form of $F(t,t')$ into Equation C-7 and solving for $T(t)$, $t_o \leq t \leq t_{23}$. Note that declination, δ, was assumed constant over the interval (t_o,t_{23}). Thus, variations within a day are ignored. Variations in δ for longer periods cannot be ignored. Therefore, the declination is recomputed for each day in which temperature forecasts are made (see Equation 4.2-5). This accounts for longer term variations in solar input.

APPENDIX D

DETERMINISTIC TEMPERATURE COMPONENT

D.1 Introduction

The deterministic component of the temperature model is represented by

$$\frac{d\tilde{T}(t)}{dt} + b_1\tilde{T}(t) = b_o + b_2K(t)s(t) + b_3K(t)r(t)$$

$$+ b_4q(t) + b_5T_g(t) + b_6W_s(t)$$

$$+ b_7W_d(t) \qquad \text{(D-1)}$$

where $\tilde{T}(t)$ is the deterministic component, $K(t)$ is the solar radiation attenuation factor ($K(t) = 1 - 0.65N^2(t)$), $N(t)$ is the cloud cover, $q(t)$ is a longwave radiation estimate (see Equation 6.3-2), $T_g(t)$ is the ground temperature, $W_s(t)$ is the wind speed, and $W_d(t)$ is the wind direction.

As indicated in Chapter 6, the general solution to Equation D-1 can be written as

$$\tilde{T}(t) = \tilde{T}(t')e^{-b_1(t-t')} + e^{-b_1t}G(t,t') \qquad \text{(D-2)}$$

where

$$G(t,t') = b_o\int_{t'}^{t} e^{b_1\tau}d\tau + b_2\int_{t'}^{t} e^{b_1\tau}K(\tau)s(\tau)d\tau$$

$$+ b_3 \int_{t'}^{t} e^{b_1\tau} K(\tau) r(\tau) d\tau + b_4 q(t-1) \int_{t'}^{t} e^{b_1\tau} d\tau$$

$$+ b_5 \int_{t'}^{t} e^{b_1\tau} T_g(\tau) d\tau + b_6 \int_{t'}^{t} e^{b_1\tau} W_s(\tau) d\tau$$

$$+ b_7 \int_{t'}^{t} e^{b_1\tau} W_d(\tau) d\tau \qquad \text{(D-3)}$$

D.2 Parameter Estimation

The procedure for estimating the coefficients b_i through a regression involving hourly temperature changes, $Y(t)$, has been described in Appendix C. The resulting coefficients a_i are

$$\left.\begin{aligned} a_1 &= -(1-e^{b_1}) \\ a_i &= -\frac{a_1}{b_1} b_i \quad , \quad i = 0,2,3,\ldots,7 \end{aligned}\right\} \qquad \text{(D-4)}$$

The predictors $X_i(t)$ are

$$X_1(t) = \tilde{T}(t-1) \qquad \text{(D-5)}$$

$$X_2(t) = K(t) \int_{t-1}^{t} s(\tau) d\tau \qquad \text{(D-6)}$$

$$X_3(t) = K(t) \int_{t-1}^{t} r(\tau) d\tau \qquad \text{(D-7)}$$

$$X_4(t) = q(t-1) \tag{D-8}$$

$$X_5(t) = T_g(t) \tag{D-9}$$

$$X_6(t) = W_s(t) \tag{D-10}$$

$$X_7(t) = W_d(t) \tag{D-11}$$

The one hour integration interval was considered short enough to allow the variables $K(t)$, $q(t-1)$, $T_g(t)$, $W_s(t)$, and $W_d(t)$ to be brought outside their respective integrals.

Predictors $X_2(t)$ and $X_3(t)$ are used only during selected parts of the day. These times have been defined in Appendix C and will not be discussed again here. The indicated integrations in Equations D-6 and D-7 have also been discussed in Appendix C. The only difference in the final forms of $X_2(t)$ and $X_3(t)$ for the present case is the multiplier $K(t)$. The remaining predictors are used throughout the night and day.

D.3 Evaluation of $G(t,t')$

For convenience, let the seven integrals of Equation

D-3 be written as

$$I_1 = b_o \int_{t'}^{t} e^{b_1\tau} d\tau \qquad \text{(D-12)}$$

$$I_2 = b_2 \int_{t'}^{t} e^{b_1\tau} K(\tau)s(\tau)d\tau \qquad \text{(D-13)}$$

$$I_3 = b_3 \int_{t'}^{t} e^{b_1\tau} K(\tau)r(\tau)d\tau \qquad \text{(D-14)}$$

$$I_4 = b_4 \int_{t'}^{t} e^{b_1\tau} q(\tau)d\tau \qquad \text{(D-15)}$$

$$I_5 = b_5 \int_{t'}^{t} e^{b_1\tau} T_g(\tau)d\tau \qquad \text{(D-16)}$$

$$I_6 = b_6 \int_{t'}^{t} e^{b_1\tau} W_s(\tau)d\tau \qquad \text{(D-17)}$$

$$I_7 = b_7 \int_{t'}^{t} e^{b_1\tau} W_d(\tau)d\tau \qquad \text{(D-18)}$$

The integration indicated for I_1 is straightforward and results in

$$I_1 = \frac{b_o}{b_1}\left(e^{b_1 t} - e^{b_1 t'} \right) \qquad \text{(D-19)}$$

The remaining integrals contain terms such as $K(\tau)$, $q(\tau)$, $T_g(\tau)$, $W_s(\tau)$, and $W_d(\tau)$. Except for reasonably short intervals, treating these terms as constants is not sensible. To deal with integration intervals that are large enough for these variables to vary significantly, the following approach is taken.

Consider I_2, where

$$I_2 = b_2 \int_{t'}^{t} e^{b_1 \tau} K(\tau) s(\tau) d\tau \qquad \text{(D-13)}$$

I_2 can also be written in an equivalent form as

$$I_2 = b_2 \int_{t-1}^{t} e^{b_1 \tau} K(\tau) s(\tau) d\tau$$

$$+ b_2 \int_{t'}^{t-1} e^{b_1 \tau} K(\tau) s(\tau) d\tau \qquad \text{(D-20)}$$

In the first integral on the right-hand side of Equation D-14, the integration interval is short enough such that $K(\tau)$ can be brought outside the integral. Thus

$$I_2 = b_2 K(t) \int_{t-1}^{t} e^{b_1 \tau} s(\tau) d\tau$$

$$+ b_2 \int_{t'}^{t-1} e^{b_1 \tau} K(\tau) s(\tau) d\tau \qquad \text{(D-21)}$$

Now the first integral in Equation D-22 is in the same form as the integrals evaluated in Appendix C (see Equation C-47).

The same argument can be used to successively evaluate the second integral of Equation D-21. Following the procedure hour by hour back to t', a series of the following form results.

$$I_2 = b_2 K(t) \int_{t-1}^{t} e^{b_1 \tau} s(\tau) d\tau$$

$$+ b_2 \sum_{n=1}^{t'-t} K(n) \int_{t'+n-1}^{t'+n} e^{b_1 \tau} s(\tau) d\tau \qquad \text{(D-22)}$$

Since the series of integrals defined in the second term on the right-hand side of Equation D-22 is just the value of I_2 at t-1, the following computational form is used

$$I_2(t) = b_2 K(t) \int_{t-1}^{t} e^{b_1 \tau} s(\tau) d\tau + I_2(t-1) \tag{D-23}$$

Concluding the integration of Equation D-23 yields

$$\begin{aligned} I_2(t) = \hat{K}(t) \Bigg[& \frac{b_2}{b_1} \sin\delta \sin\phi \left(e^{b_1 t} - e^{b_1 (t-1)}\right) \\ & - \frac{e^{b_1 t} b_1 b_2 \cos\delta \cos\phi \cos\left(\frac{\pi t}{12}\right)}{b_1^{\,2} + \left(\frac{\pi}{12}\right)^2} \\ & - \frac{e^{b_1 t} b_2 \left(\frac{\pi}{12}\right) \cos\delta \cos\phi \sin\left(\frac{\pi t}{12}\right)}{b_1^{\,2} + \left(\frac{\pi}{12}\right)^2} \\ & + \frac{e^{b_1 (t-1)} b_1 b_2 \cos\delta \cos\phi \cos\left(\frac{\pi (t-1)}{12}\right)}{b_1^{\,2} + \left(\frac{\pi}{12}\right)^2} \\ & + \frac{e^{b_1 (t-1)} b_2 \left(\frac{\pi}{12}\right) \cos\delta \cos\phi \sin\left(\frac{\pi (t-1)}{12}\right)}{b_1^{\,2} + \left(\frac{\pi}{12}\right)^2} \Bigg] \\ & + I_2(t-1) \end{aligned} \tag{D-24}$$

Using the definitions for K_o, K_1, ..., K_6 defined in Appendix C, Equation D-24 can be written as

$$I_2(t) = K(t)(K_2(e^{b_1 t} - e^{b_1(t-1)}) - K_3 e^{b_1 t}\cos(\frac{\pi t}{12}) - K_4 e^{b_1 t}\sin(\frac{\pi t}{12}) + K_3 e^{b_1(t-1)}\cos(\frac{\pi(t-1)}{12}) + K_4 e^{b_1(t-1)}\sin(\frac{\pi(t-1)}{12})) + I_2(t-1) \qquad \text{(D-25)}$$

Similarly, the remaining integrals, I_i, can be obtained.

$$I_3(t) = K(t)(K_6 e^{b_1 t}\sin(\frac{\pi t}{12}) - K_5 e^{b_1 t}\cos(\frac{\pi t}{12}) - K_6 e^{b_1(t-1)}\sin(\frac{\pi(t-1)}{12}) + K_5 e^{b_1(t-1)}\cos(\frac{\pi(t-1)}{12})) + I_3(t-1) \qquad \text{(D-26)}$$

$$I_4(t) = \frac{b_4}{b_1} q(t-1)(1-e^{-b_1})e^{b_1 t} + I_4(t-1) \qquad \text{(D-27)}$$

$$I_5(t) = \frac{b_5}{b_1} T_g(t)(1-e^{-b_1})e^{b_1 t} + I_5(t-1) \qquad \text{(D-28)}$$

$$I_6(t) = \frac{b_6}{b_1} W_s(t)(1-e^{-b_1})e^{b_1 t} + I_6(t-1) \qquad \text{(D-29)}$$

$$I_7(t) = \frac{b_7}{b_1} W_d(t)(1-e^{-b_1})e^{b_1 t} + I_7(t-1) \qquad \text{(D-30)}$$

The specific form of $G(t,t')$ still depends on the time of day for which the integrals are evaluated. (Note that $t' = t_o-1$).

For Range 1, $t_o \leq t < R$

$$G(t,t') = K_1(e^{b_1 t} - e^{b_1 t'}) + \frac{b_4}{b_1} q(t-1)(1-e^{b_1})e^{b_1 t}$$

$$+ I_4(t-1) + \frac{b_5}{b_1} T_g(t)(1-e^{-b_1})e^{b_1 t} + I_5(t-1)$$

$$+ \frac{b_6}{b_1} W_s(t)(1-e^{-b_1})e^{b_1 t} + I_6(t-1)$$

$$+ \frac{b_7}{b_1} W_d(t)(1-e^{b_1})e^{b_1 t} + I_7(t-1) \qquad \text{(D-31)}$$

Actually, the terms on the right-hand side of Equation D-31 retain the same form throughout the day. For convenience then, the terms on the right-hand side of Equation D-31 will be collectively referred to as $H(t,t')$.

For Range 2, $R \leq t < R+1$

$$G(t,t') = H(t,t') + K(t)\left\{K_2(e^{b_1 t} - e^{b_1 R})\right.$$

$$- K_3 e^{b_1 t} \cos(\frac{\pi t}{12}) - K_4 e^{b_1 t} \sin(\frac{\pi t}{12})$$

$$+ K_3 e^{b_1 R} \cos(\frac{\pi R}{12}) + K_4 e^{b_1 R} \sin(\frac{\pi R}{12}) \Bigg]$$

$$+ K(t) \Bigg[K_6 e^{b_1 t} \sin(\frac{\pi t}{12}) - K_5 e^{b_1 t} \cos(\frac{\pi t}{12})$$

$$- K_6 e^{b_1 R} \sin(\frac{\pi R}{12}) + K_5 e^{b_1 R} \cos(\frac{\pi R}{12}) \Bigg] \qquad \text{(D-32)}$$

For Range 3, $R+1 \leq t < 12$

$$G(t,t') = H(t,t') + K(t) \Bigg[K_2 (e^{b_1 t} - e^{b_1 (t-1)})$$

$$- K_3 e^{b_1 t} \cos(\frac{\pi t}{12}) - K_4 e^{b_1 t} \sin(\frac{\pi t}{12})$$

$$+ K_3 e^{b_1 (t-1)} \cos(\frac{\pi (t-1)}{12}) + K_4 e^{b_1 (t-1)} \sin(\frac{\pi (t-1)}{12}) \Bigg]$$

$$+ I_2(t-1) + K(t) \Bigg[K_6 e^{b_1 t} \sin(\frac{\pi t}{12}) - K_5 e^{b_1 t} \cos(\frac{\pi t}{12})$$

$$- K_6 e^{b_1 (t-1)} \sin(\frac{\pi (t-1)}{12})$$

$$+ K_5 e^{b_1 (t-1)} \cos(\frac{\pi (t-1)}{12}) \Bigg] + I_3(t-1) \qquad \text{(D-33)}$$

For Range 4, $12 \leq t < 12 + 1$

$$G(t,t') = H(t,t') + K(t) \left[K_2(e^{b_1 t} - e^{b_1(t-1)}) \right.$$

$$- K_3 e^{b_1 t} \cos(\tfrac{\pi t}{12}) - K_4 e^{b_1 t} \sin(\tfrac{\pi t}{12})$$

$$\left. + K_3 e^{b_1(t-1)} \cos(\tfrac{\pi(t-1)}{12}) + K_4 e^{b_1(t-1)} \sin(\tfrac{\pi(t-1)}{12}) \right]$$

$$+ I_2(t-1) + K(t) \left[K_5 e^{12 b_1} - K_6 e^{b_1(t-1)} \sin(\tfrac{\pi(t-1)}{12}) \right.$$

$$\left. + K_5 e^{b_1(t-1)} \cos(\tfrac{\pi(t-1)}{12}) \right] + I_3(t+1) \qquad \text{(D-34)}$$

For Range 5, $12 + 1 \leq t < S$

$$G(t,t') = H(t,t') + K(t) \left[K_2(e^{b_1 t} - e^{b_1(t-1)}) \right.$$

$$- K_3 e^{b_1 t} \cos(\tfrac{\pi t}{12}) - K_4 e^{b_1 t} \sin(\tfrac{\pi t}{12})$$

$$\left. + K_3 e^{b_1(t-1)} \cos(\tfrac{\pi(t-1)}{12}) + K_4 e^{b_1(t-1)} \sin(\tfrac{\pi(t-1)}{12}) \right]$$

$$+ I_2(t-1) + I_3(12) \qquad \text{(D-35)}$$

For Range 6, $S \leq t < S + 1$

$$G(t,t') = H(t,t') + K(t)\left\{K_2(e^{b_1 S} - e^{b_1(t-1)}) - K_3 e^{b_1 S}\cos(\frac{\pi S}{12}) - K_4 e^{b_1 S}\sin(\frac{\pi S}{12}) + K_3 e^{b_1(t-1)}\cos(\frac{\pi(t-1)}{12}) + K_4 e^{b_1(t-1)}\sin(\frac{\pi(t-1)}{12})\right\} + I_2(t-1) + I_3(12) \qquad \text{(D-36)}$$

Finally, for Range 7, $S + 1 \leq t < 23$

$$G(t,t') = H(t,t') + I_2(S) + I_3(12) \qquad \text{(D-37)}$$

Now with the appropriate form of $G(t,t')$, Equation D-2 can be used to find the deterministic component, $\tilde{T}(t)$, at any time of day.

APPENDIX E

CSCS PROGRAM LISTING

```
C=====================================================================
C=====================================================================
C
C  CARD INPUT SUMMARY FOR THE CSCS MODEL
C
C
C  CARD  COLUMN   FORMAT   DESCRIPTION
C  ----  ------   ------   -----------
C
C    1    1-80      --      USER INFORMATION CARD USED ONLY TO SEPARATE
C                           CARD GROUPS IN THE DECK (OR FILE)
C
C  2-4    1-80     15A5     THREE TITLE CARDS. THE TEXT ON THESE CARDS
C                           WILL BE PRINTED OUT AT THE BEGINNING OF
C                           THE INPUT DATA SUMMARY.
C
C    5    1-10      10X     SPACE FOR CARD LABEL. NOT READ BY CSCS.
C        11-20      A10     OUTPUT FILE NAME FOR INPUT DATA SUMMARY AND
C                           OUTPUT DATA ANALYSIS. FILE NAME HAS THE
C                           FORM  XXXXXX.YYY
C        21-30      A10     OUTPUT FILE NAME FOR GENERATED DEWPOINT
C                           DEPRESSION.  XXXXXX.YYY
C        31-40      A10     OUTPUT FILE NAME FOR DEBUG INFORMATION
C                           XXXXXX.YYY
C
C    6    1-10      10X     CARD LABEL
C        11-12       I2     INITIAL MONTH  - MM
C        14-15       I2     INITIAL DAY    - DD
C        17-20       I4     INITIAL YEAR   - YYYY
C        22-23       I2     ENDING MONTH   - MM
C        25-26       I2     ENDING DAY     - DD
C        28-31       I4     ENDING YEAR    - YYYY
C        33-35     F3.0     LATITUDE       - DEGREES
C        37-38     F2.0     LATITUDE       - MINUTES
C        40-41     F2.0     LATITUDE       - SECONDS
C        43-45     F3.0     LONGITUDE      - DEGREES
C        47-48     F2.0     LONGITUDE      - MINUTES
C        50-51     F2.0     LONGITUDE      - SECONDS
C        53-62      A10     TIME ZONE (U.S. 'EASTERN', 'CENTRAL',
C                           'MOUNTAIN', OR 'PACIFIC')  LEFT JUSTIFY.
C
C    7    1-80      --      USER INFORMATION CARD
C
C    8    1-10      10X     CARD LABEL
C        11-20    F10.0     MEAN FAIRWEATHER CLOUD COVER
C        21-30    F10.0     ST. DEV. OF FAIRWEATHER CLOUD COVER
C        31-40    F10.0     LAG-1 CORRELATION COEF. OF FAIRWEATHER
C                           CLOUD COVER.
C        41-50    F10.0     CLOUD COVER TRANSITION DECAY COEFFICIENT
C        51-60    F10.0     ATMOSPHERIC TURBIDITY FACTOR
C
C    9    1-10      10X     CARD LABEL
C        11-12       I2     NUMBER OF FAIRWEATHER CLOUD COVER HISTOGRAM
C                           ELEMENTS
C        21-30    F10.0     LOWER BOUND OF FAIRWEATHER CLOUD COVER
```

```
C                                      HISTOGRAM
C         31-40      F10.0             UPPER BOUND OF FAIR-WEATHER CLOUD COVER
C                                      HISTOGRAM
C
C    10    1-10        10X             CARD LABEL
C         11-70     6F10.0             HISTOGRAM ELEMENTS. USE AS MANY CARDS AS
C                                      NEEDED. REMEMBER THAT THE FIRST 10 SPACES
C                                      ON EACH CARD ARE RESERVED FOR THE CARD LABEL
C
C    11    1-10        10X             CARD LABEL
C         11-70     6F10.0             RIGHT HAND COORDINATE OF EACH HISTOGRAM
C                                      ELEMENT FROM LOWEST TO HIGHEST.
C
C    12    1-80         --             USER INFORMATION CARD
C
C    13    1-10        10X             CARD LABEL
C         11-20      F10.0             MEAN TIME BETWEEN STORMS
C         21-30      F10.0             MEAN STORM DURATION
C         31-40      F10.0             MEAN STORM DEPTH
C
C    14    1-80         --             USER INFORMATION CARD
C
C    15    1-10        10X             CARD LABEL
C         11-58     4E12.5             REGRESSION COEFFICIENTS FOR THE DETER-
C                                      MINISTIC COMPONENT OF THE TEMPERATURE
C                                      MODEL (B0-B7). USE TWO CARDS.
C
C    16    1-10        10X             CARD LABEL
C         11-20      F10.0             TEMPERATURE BIAS FOR THE STOCHASTIC
C                                      COMPONENT OF TEMPERATURE
C         21-30      F10.0             ST. DEVIATION FOR THE STOCHASTIC COMPONENT
C                                      OF TEMPERATURE
C         31-40      F10.0             LAG-1 SERIAL CORRELATION COEFFICIENT FOR
C                                      THE STOCHASTIC COMPONENT OF TEMPERATURE
C
C    17    1-80         --             USER INFORMATION CARD
C
C    18    1-10        10X             CARD LABEL
C         11-20      F10.0             MINIMUM HOURLY WIND SPEED
C         21-30      F10.0             TIME OF MINIMUM HOURLY WIND SPEED
C         31-40      F10.0             MAXIMUM HOURLY WIND SPEED
C         41-50      F10.0             TIME OF MAXIMUM HOURLY WIND SPEED
C
C    19    1-10        10X             CARD LABEL
C         11-20      F10.0             MINIMUM HOURLY ST. DEVIATION OF WIND SPEED
C         21-30      F10.0             TIME OF MINIMUM HOURLY ST. DEVIATION OF
C                                      WIND SPEED
C         31-40      F10.0             MAXIMUM HOURLY ST. DEVIATION OF WIND SPEED
C         41-50      F10.0             TIME OF MAXIMUM HOURLY ST. DEVIATION OF
C                                      WIND SPEED
C
C    20    1-10        10X             CARD LABEL
C         11-20      F10.0             WIND SPEED SKEW COEFFICIENT
C         21-30      F10.0             LAG-1 SERIAL CORRELATION COEFFICIENT OF
C                                      WIND SPEED
```

```
C
C      21   1-80         --       USER INFORMATION
C
C      22   1-10         10X      CARD LABEL
C          11-20       F10.0      MEAN TRANSFORMED WIND DIRECTION
C          21-30       F10.0      ST. DEVIATION OF TRANSFORMED WIND DIRECTION
C          31-40       F10.0      LAG-1 SERIAL CORRELATION COEFFICIENT OF
C                                 TRANSFORMED WIND DIRECTION
C
C      23   1-10       F10.0      CARD LABEL
C          11-12          I2      NUMBER OF ELEMENTS IN TRANSFORMED WIND
C                                 DIRECTION HISTOGRAM
C          21-30       F10.0      LOWER BOUND OF HISTOGRAM (USUALLY 0.00)
C          31-40       F10.0      UPPER BOUND OF HISTOGRAM (USUALLY 180.)
C
C      24   1-10         10X      CARD LABEL
C          11-70      6F10.0      HISTOGRAM ELEMENTS. USE AS MANY CARDS AS
C                                 NEEDED
C
C      25   1-10         10X      CARD LABEL
C          11-70      6F10.0      RIGHT HAND COORDINATE OF EACH HISTOGRAM
C                                 ELEMENT, LOWEST TO HIGHEST
C
C      26   1-80         --       USER INFORMATION CARD
C
C      27   1- 5          A5      DEWPOINT MODEL TYPE
C                                 'REGRS' = REGRESSION MODEL
C                                 'INDEP' = INDEPENDENT MODEL
C
C      ***  FOR INDEPENDENT DEWPOINT GENERATION ONLY ***
C
C      28   1-10         10X      CARD LABEL
C          11-20       F10.0      MEAN DEWPOINT TEMPERATURE
C          21-30       F10.0      ST. DEVIATION OF DEWPOINT TEMPERATURE
C          31-40       F10.0      LAG-1 SERIAL CORRELATION COEFFICIENT OF
C                                 DEWPOINT TEMPERATURE
C
C      ***  FOR REGRESSION DEWPOINT GENERATION ONLY  ***
C
C      29   1-10         10X      CARD LABEL
C          11-58      4E12.5      REGRESSION COEFFICIENTS FOR THE DETER-
C                                 MINISTIC COMPONENT OF DEWPOINTS (D0-D5)
C
C
C      30   1-10         10X      CARD LABEL
C          11-20       F10.0      BIAS OF STOCHASTIC COMPONENT OF DEWPOINTS
C          21-30       F10.0      ST. DEVIATION OF STOCHASTIC COMPONENT OF
C                                 DEWPOINTS
C          31-40       F10.0      LAG-1 SERIAL CORRELATION COEFFICIENT OF
C                                 STOCHASTIC COMPONENT OF DEWPOINTS
C
C
C==================================================================
C==================================================================
C
```

```
C
C.....................................................................
C
      PROGRAM CSCS
C
C
C  CONSTRAINED STOCHASTIC CLIMATE SIMULATION
C
C  PROGRAMMER: DAVID C. CURTIS
C              NORTHEAST RIVER FORECAST CENTER
C              705 BLOOMFIELD AVENUE
C              BLOOMFIELD, CT 06002-2478
C
C              TELEPHONE: (203) 244-2520
C
C  THE CSCS MODEL GENERATES HOURLY VALUES OF PRECIPITATION, CLOUD
C  COVER, SHORTWAVE RADIATION, LONGWAVE RADIATION, TEMPERATURE,
C  DEWPOINT, WIND SPEED AND WIND DIRECTION.  THE PROGRAM CODE IS
C  FORTRAN AND HAS BEEN DEVELOPED ON A DEC-10 TIME-SHARE COMPUTER
C  SYSTEM.  STANDARD FORTRAN CODE WAS USED AS MUCH AS POSSIBLE TO
C  AVOID TOO MANY PROBLEMS WHEN TRANSFERRING THIS CODE TO OTHER
C  MACHINES.  HOWEVER SOME MACHINE DEPENDENT CODE IS INEVITABLE,
C  SUCH AS:
C
C           -- 'OPEN' STATEMENTS FOR DATA FILE ACCESS
C           -- 5 CHARACTER WORDS FOR ALPHANUMERIC DATA MANIPULATION
C           -- INPUT/OUTPUT UNIT NUMBERS
C           -- RANDOM NUMBER GENERATION (SEE SUBROUTINE RANDU)
C
C  DATA INPUT AND INTERNAL COMPUTATIONS HAVE BEEN CARRIED OUT IN
C  ENGLISH UNITS.  DATA OUTPUT CAN BE IN ENGLISH OR METRIC UNITS. (SEE
C  THE METRIC CONVERSION SECTION IN THE MAIN PROGRAM) THE PLOT SUB-
C  ROUTINE IS SCALED FOR METRIC OUTPUT.
C
C  THE PROGRAM IS CURRENTLY SET UP FOR GENERATING ANY NUMBER OF SETS
C  OF DATA FOR A PARTICULAR MONTH.  IN OTHER WORDS, 20 JULYS, 35 APRILS,
C  15 JANUARYS ETC. CAN BE GENERATED.  IF THE INPUT PARAMETERS
C  REPRESENT OTHER PERIODS SUCH AS BIMONTHLY, SEASONALLY, ETC., THE
C  DATE COUNTERS MUST BE ADJUSTED ACCORDINGLY (SEE SUBROUTINE DATE1)
C  JULIAN DATES ARE USED INTERNALLY.  THE PROGRAM HAS BEEN FULLY
C  TESTED FOR JANUARY AND JULY ONLY.
C
C  TO ALL USERS:  GOOD LUCK!!!
C
      DIMENSION TITLE(16,3), BCOEF(8), ACOEF(8)
      DIMENSION CCPDF(30), CCORD(30), TTPDF(40), TTORD(40)
      DIMENSION DRPDF(30),DRORD(30)
      DIMENSION DWPDF(30), DWORD(30)
      DIMENSION ZERO(10), SPB(24),SPSD(24)
      DIMENSION RAWSUM(5), XXT(5,5), MEAN(5)
      DIMENSION COVMAT(5,5), CORMAT(5,5)
      DIMENSION TCDATA(24),DWDATA(24),CLDATA(24),WSDATA(24),WDDATA(24)
      DIMENSION TCTITL(5) ,DWTITL(5) ,CLTITL(5) ,WSTITL(5) ,WDTITL(5)
      DIMENSION TCRHO(24) ,DPRHO(24) ,CLRHO(24) ,WSRHO(24) ,WDRHO(24)
      DIMENSION TCHIST(50),DPHIST(50),CLHIST(11),WSHIST(40),WDHIST(9)
```

```
      DIMENSION ASWRS(24),ACLDS(24),ALWRS(24),ATMPS(24),ADEWS(24)
      DIMENSION AWSPS(24),AWDRS(24),ASWRB(24),ACLDB(24),ALWRB(24)
      DIMENSION ATMPB(24),ADEWB(24),AWSPB(24),AWDRB(24)
      DIMENSION ASWRSQ(24),ACLDSQ(24),ALWRSQ(24),ATMPSQ(24),ADEWSQ(24)
      DIMENSION ASWRSD(24),ACLDSD(24),ALWRSD(24),ATMPSD(24),ADEWSD(24)
      DIMENSION AWSPSQ(24),AWDRSQ(24)
      DIMENSION AWSPSD(24),AWDRSD(24)
      DIMENSION DEP(24), PTEXT(16)
C
C
      DOUBLE PRECISION  WRITEF, BUGOFF, OUTPUT, TZONE, TZ(4), DEBUG(7)
      DOUBLE PRECISION  DAFILE, RADTYP, PNFILE
C
C
      REAL KBAR, LW, MEAN
      REAL I0, I1, I2, I3, I4, I5, I6, I7
      REAL LAT(3), LONG(3)
C
C
      INTEGER TCHIST, DPHIST, CLHIST, WSHIST, WDHIST
      INTEGER TCHDIM, DPHDIM, CLHDIM, WSHDIM, WDHDIM
C
C
      EQUIVALENCE (ZERO(1),I0),(ZERO(2),I1),(ZERO(3),I2),(ZERO(4),I3),
     $            (ZERO(5),I4),(ZERO(6),I5),(ZERO(7),I6),(ZERO(8),I7)
C
      COMMON /TITLES/ TITLE
      COMMON /FILES/  WRITEF, OUTPUT, BUGOFF
      COMMON /DATES/  IYR,IMO,IDAY, LYR,LMO,LDAY
      COMMON /LOCATE/ LAT, LONG, TZONE
      COMMON /DBUG/   NBUG,DEBUG
      COMMON /CLOUDS/ CCBAR,CCSD,CCRHO,BETA,GAM
      COMMON /ATMOS/  EN
      COMMON /PDFCLD/ NUMCC,CCPDF,CCORD,CCA,CCB
      COMMON /RAINS/  TDBAR, TRBAR, DBAR
      COMMON /TEMPAR/ TDBIAS, TDSDEV, TDRHO, BCOEF, TEMBAR
      COMMON /PDFTEM/ NUMTT, TTPDF, TTORD, TTA, TTB
      COMMON /WINDSP/ SPBAR1,SPBAR2,SPBT1,SPBT2,SPSDV1,SPSDV2,
     $ .              SPSDT1,SPSDT2,SPSKEW,SPRHO
      COMMON /WINDIR/ DRBAR,DRDEV,DRRHO
      COMMON /PDFDIR/ NUMDR,DRPDF,DRORD,DRA,DRB
      COMMON /DEWONE/ TYPE, ACOEF
      COMMON /DEWTWO/ DWBAR,DWSDEV,DWSKEW,DWRHO
      COMMON /DEWDVS/ DWBIAS, DWDEV, DWDRHO
      COMMON /ORBIT/  PHI,THETAS,THETAL,EP,ET,W
      COMMON /SUN/    DELTA, DTSL, SR,SS
      COMMON /JDATES/ JULDAT, JULREL, JBEGIN, JULEND, JRANGE, NYLPYR
     $ .             ,JSTART, JSTOP,  JREND,  JYEAR
      COMMON /IO/     IN,IS,IB
      COMMON /RAINI/  ITN,ITB
      COMMON /SEED/   ISEED
      COMMON /CLDCOV/ CCLAG1
      COMMON /SEAS/   NSEAS
      COMMON /RTYPE/  RADTYP
      COMMON /STORMS/ STORM
```

```
      COMMON /INTEG/  I0,I1,I2,I3,I4,I5,I6,I7
      COMMON /LINES/  NLINES
      COMMON /VAPORP/ VP
      COMMON /PUNCHD/ PTEXT,PNFILE,IPUNCH
C
      DATA TZ/8HEASTERN  ,8HCENTRAL  ,8HMOUNTAIN  ,8HPACIFIC  /
      DATA TCTITL /5HHOURL, 5HY TEM, 5HPERAT, 5HURE   , 5H       /
      DATA DWTITL /5HHOURL, 5HY DEW, 5HPOINT, 5H      , 5H       /
      DATA CLTITL /5HHOURL, 5HY CLO, 5HUD CO, 5HVER   , 5H       /
      DATA WSTITL /5HHOURL, 5HY WIN, 5HD SPE, 5HED    , 5H       /
      DATA WDTITL /5HHOURL, 5HY WIN, 5HD DIR, 5HECTIO, 5HN      /
      DATA ON /2HON/, OFF /3HOFF /
C
C
C   OUTPUT VARIABLE DEFINITION
C
C   VARIABLE      DIMENSION        DESCRIPTION
C   --------      ---------        -----------
C     SWR           LY/HR          SHORTWAVE RADIATION
C     LW            LY/HR          LONGWAVE RADIATION
C     WDIR          DEGREES        WIND DIRECTION
C     CLD           -----          CLOUD COVER
C
C    ******  ENGLISH UNITS  ******
C
C     RAIN          IN/HR          PRECIPITAION
C     WSP           MI/HR          WIND SPEED
C     TEMP.         DEG F          TEMPERATURE
C     DEW           DEG F          DEWPOINT
C
C    ******  METRIC  UNITS  ******
C
C     RAINM         MM/HR          PRECIPITAION
C     WSPM          M/S            WIND SPEED
C     TEMPM         DEG C          TEMPERATURE
C     DEWM          DEG C          DEWPOINT
C
C
C
C
C   CALL INTERACTIVE INPUT SUBROUTINE TO GET UNIT NUMBER AND
C   DATA FILE INFORMATION NEEDED TO BEGIN OPERATION
C
      CALL INTER (DAFILE,IS,DPLOT,IPL)
      CALL START (ISEED)
C
C===================================================================
C===================================================================
C
C
C   ESTABLISH THE INPUT DATE FILE UNIT NUMBER AND OPEN FILE FOR READ
C
      IN = 21
      OPEN (UNIT=IN,DEVICE='DSK',ACCESS='SEQIN',FILE=DAFILE)
C
```

```
      IF (IPUNCH .LE. 0) GO TO 100
      OPEN (UNIT=IPUNCH,DEVICE='DSK',ACCESS='SEQOUT',FILE=PNFILE)
C
      WRITE (IPUNCH,50) PTEXT
   50 FORMAT (16A5)
C
  100 CONTINUE
C
      IU = 26
C
      OPEN (UNIT=IU,DEVICE='DSK',ACCESS='SEQOUT',FILE=OUTPUT)
C
C
C=====================================================================
C=====================================================================
C
C
C
C   READ INPUT DATA FILE
C
      CALL READF (IN,IS,IB)
C
C
C=====================================================================
C=====================================================================
C
C
C   CONVERT LATITUDE AND LONGITUDE DEGREES:MINUTES:SECONDS TO
C   THEIR DECIMAL EQUIVALENTS.
C
      PHI = DMS(LAT)
      THETAL = DMS(LONG)
C
C   CHECK IF VALID TIME ZONE HAS BEEN REQUESTED
C
      IF (TZONE.NE.TZ(1)) GO TO 200
      THETAS = 75.0*2.0*3.14159/360.0
      GO TO 300
C
  200 IF (TZONE.NE.TZ(2)) GO TO 210
      THETAS =  90.0*2.0*3.14159/360.
      GO TO 300
C
  210 IF (TZONE.NE.TZ(3)) GO TO 220
      THETAS = 105.0*2.0*3.14159/360.0
      GO TO 300
C
  220 IF (TZONE.NE.TZ(4)) GO TO 230
      THETAS = 120.0*2.0*3.14159/360.0
      GO TO 300
C
  230 WRITE (IS,240)
  240 FORMAT (1H1,'TIME ZONE REQUESTED IS NOT VALID'///)
      WRITE (IS,250) TZONE,(TZ(I),I=1,4)
  250 FORMAT (1H0,T10,'REQUESTED TIME ZONE',T36,5H*****,
```

```
     1A8,5H*****/T10,'AVAILABLE TIME ZONES',T36,5H*****,
     2A8,5H*****/T36,5H*****,A8,5H*****/T36,5H*****,
     3A8,5H*****/T36,5H*****,A8,5H*****)
C
C
C=======================================================================
C=======================================================================
C
  300 CONTINUE
C
C
C   INITIALIZE DATE COUNTERS
C
      CALL DATE1
C
C   INITIALIZE RAINFALL MODEL PARAMETERS
C
      CALL RAINST (TB,TR,D,JSINCE,STORM,JHREOS,JHNEXT)
C
C
C
C=======================================================================
C=======================================================================
C
C
C   VARIABLE INITIALIZATION SECTION
C
C     STATISTICAL VARIABLES
C
      NDC    = 0
      NDATA  = 0
      NRDATA  = 0
      TCSUM  = 0.0
      DWSUM  = 0.0
      CLSUM  = 0.0
      WSSUM  = 0.0
      WDSUM  = 0.0
      TRSUM  = 0.0
      TBSUM  = 0.0
      DHSUM  = 0.0
      DRY    = 0.0
      STORMS  = 0.0
      TCSMSQ  = 0.0
      DWSMSQ  = 0.0
      CLSMSQ  = 0.0
      WSSMSQ  = 0.0
      WDSMSQ  = 0.0
      TTSUM3  = 0.0
      DWSUM3  = 0.0
      CLSUM3  = 0.0
      SPSUM3  = 0.0
      WDSUM3  = 0.0
C
      DO 375 IA = 1,24
        TCRHO(IA) = 0.0
```

```
          DPRHO(IA) = 0.0
          CLRHO(IA) = 0.0
          WSRHO(IA) = 0.0
          WDRHO(IA) = 0.0
          ASWRS(IA)  = 0.0
          ASWRSQ(IA) = 0.0
          ACLDS(IA)  = 0.0
          ACLDSQ(IA) = 0.0
          ALWRS(IA)  = 0.0
          ALWRSQ(IA) = 0.0
          ATMPS(IA)  = 0.0
          ATMPSQ(IA) = 0.0
          ADEWS(IA)  = 0.0
          ADEWSQ(IA) = 0.0
          AWSPS(IA)  = 0.0
          AWSPSQ(IA) = 0.0
          AWDRS(IA)  = 0.0
          AWDRSQ(IA) = 0.0
  375 CONTINUE
C
C     HISTOGRAM VARIABLES
C
        TCHDIM = 50
        DPHDIM = 50
        CLHDIM = 11
        WSHDIM = 40
        WDHDIM = 9
        TCDT   = 2.0
        DPDT   = 2.0
        CLDT   = 0.10
        WSDT   = 1.0
        WDDT   = 20.0
        TCBASE = -30.0
        DPBASE = -30.0
        CLBASE = -.05
        WSBASE = 00.0
        SUMSW  = 00.0
        SUMLW  = 00.0
        WDBASE = 00.0
        DO 270 I = 1,TCHDIM
  270   TCHIST(I) = 0
        DO 271 I = 1,DPHDIM
  271   DPHIST(I) = 0
        DO 272 I = 1,CLHDIM
  272   CLHIST(I) = 0
        DO 273 I = 1,WSHDIM
  273   WSHIST(I) = 0
        DO 274 I = 1,WDHDIM
  274   WDHIST(I) = 0
C
C     MISCELLANEOUS VARIABLES
C
        TRACE = OFF
        NLINES = 0
        IDY = 1
```

```
      MLAG = 24
      JHOUR = 0
      NSEAS = 1
      NMAX = 4
      RADTYP = 'CLOUDYSKY'
      EP = -1.00
      BETA = GAM
      ET =  0.00
      W  =  2.00
C
      IF (TRACE .EQ. ON) WRITE (IS,9000)
 9000 FORMAT (' M1')
C
C=====================================================================
C=====================================================================
C
C  SET UP VARIABLE MEAN AND STANDARD DEVIATION ARRAYS FOR
C   WIND SPEED.
C
      ST = 0.0
C
      DO 530 IV = 1,24
C
      CALL VARYX (SPBAR1,SPBAR2,SPBT1,SPBT2,ST,SPB(IV))
      CALL VARYX (SPSDV1,SPSDV2,SPSDT1,SPSDT2,ST,SPSD(IV))
      ST = ST + 1.0
C
  530 CONTINUE
C
C
C=====================================================================
C=====================================================================
C
C
C  BEGIN CYCLES FOR DATA GENERATION
C
C  THE '400' LOOP REPRESENTS THE DAY CYCLE
C
  400 CONTINUE
C
C
C         UPDATE ORBIT PARAMETERS
C
      CALL DECL (JULREL,DELTA,SR,SS)
C
      DTSL = EP*(THETAS-THETAL)*3.81972
C
C
C=====================================================================
C=====================================================================
C
C  STARTING VALUE SELECTION
C
C
C  FOR THE FIRST TIME PERIOD OF EACH MONTH,
```

```
C   GET INITIAL VALUE FOR THE VARIOUS NOISE TERMS BY SELECTING A
C   RANDOM VARIATE FROM THE APPROPRIATE PDF.
C
       IF (IDAY .GT. 1) GO TO 310
       IF (JHOUR.GT. 0) GO TO 310
C
C   GET CLOUD COVER STARTING VALUE
C
       CCLAG1 = ARVA (CCPDF,NUMCC,CCA,CCB,CCORD,NSEAS)
C
       CALL NORMAL (VN)
       TTLAG1 = TDSDEV*VN
C
C   GET WIND SPEED STARTING VALUE
C
       CALL MARGAM (NUMTT,TTPDF,TTORD,TTA,TTB,SPBAR,SPSDEV,0.0,
      $                 SPSKEW,0.0,SPLAG1,WNOISE)
C
C   GET WIND DIRECTION STARTING VALUE
C
       DRLAG1 = ARVA (DRPDF,NUMDR,DRA,DRB,DRORD,NSEAS)
C
C   GET DEWPOINT STARTING VALUE
C
       IF (TYPE .EQ. 'REGRS') GO TO 350
       CALL MARGAM (NUMTT,TTPDF,TTORD,TTA,TTB,DWBAR,DWSDEV,0.0,
      $               DWSKEW,0.0,DWLAG1,DNOISE)
       GO TO 355
  350 CONTINUE
C
       CALL NORMAL (DWX)
       DWLAG1 = 0.85*DWBAR + DWX*DWSDEV
C
       CALL NORMAL(DWX)
       DWDLAG = DWBIAS + DWX*DWDEV
C
  355 CONTINUE
C
C
C   COMPUTE INITIAL TEMPERATURE AND CONSTRAIN DEWPOINT IF NECESSARY
C
       TPR = TEMBAR + TTLAG1
       IF (DWLAG1 .GE. TPR) DWLAG1 = 0.99*TPR
C
       IF (TRACE .EQ. ON) WRITE (IS,9001)
 9001 FORMAT (' M2')
C
  310 CONTINUE
C
C
C==================================================================
C==================================================================
C
C   ESTABLISH THE LAG-1 TEMPERATURES FOR THE TEMPERATURE AND LONG
C   WAVE RADIATION MODELS.
```

```
C
      TEMP1 = TPR
      TMPLAG = TPR
C
C  COMPUTE TODAY'S COEFFICIENTS FOR THE TEMPERATURE MODEL.
C
      CALL TEMPK (DELTA, PHI, BCOEF, TPR,
     $                CO, C1, C2, C3, C4, C5, C6)
C
C  INITIALIZE THE INTEGRATION VARIABLES FOR THE TEMPERATURE MODELS.
C    (SEE THE EQUIVALENCE STATEMENT AT THE BEGINNING OF THE PROGRAM)
C
      DO 320 K = 1,10
      ZERO(K)  = 0.00
  320 CONTINUE
C
C
C======================================================================
C======================================================================
C
C
C  THE '500' LOOP REPRESENTS THE HOUR BY HOUR DATA GENERATION
C
      ST1 = 0.0
C
C
      DO 500 I = 0,23
C
      IF (TRACE .EQ. ON) WRITE (IS,9002)
 9002 FORMAT (' M3')
C
      JP = I + 1
      ST2 = FLOAT(I)
C
C======================================================================
C======================================================================
C
C  RAINFALL SECTION
C
      CALL PCPN (TB,TR,D,STORM,JHOUR,JHREOS,JHNEXT,JSINCE,RAIN)
C
      TSINCE = FLOAT(JSINCE)
C
      IF (TRACE .EQ. ON) WRITE (IS,9003)
 9003 FORMAT (' M4')
C
C======================================================================
C======================================================================
C
C  SHORTWAVE RADIATION SECTION
C
      CALL SOLRAD (JULREL,ST1,ST2,TSINCE,TB,NMAX,CCA,CCB,CCPDF,NUMCC,
     $       CCORD,SWR,CLD,BETA,GAM,CCBAR,CCSD,CCRHO,SEASON)
C
      IF (TRACE .EQ. ON) WRITE (IS,9004)
```

```
 9004 FORMAT (' M5')
C
C=====================================================================
C=====================================================================
C
C  WIND SPEED SECTION
C
      SPBAR = SPB(JP)
      SPSDEV = SPSD(JP)
C
      CALL MARGAM (NUMTT,TTPDF,TTORD,TTA,TTB,SPBAR,SPSDEV,SPRHO,
     $                  SPSKEW,SPLAG1,WSP,WNOISE)
C
      IF (WSP .LT. 0.0) WSP = 0.0
      SPLAG1 = WSP
C
      IF (TRACE .EQ. ON) WRITE (IS,9005)
 9005 FORMAT (' M6')
C
C=====================================================================
C=====================================================================
C
C  WIND DIRECTION SECTION
C
      CALL MARKOV (NUMDR,DRPDF,DRORD,DRA,DRB,DRBAR,DRDEV,DRRHO,
     $                DRLAG1,1,WDIR)
  505 CONTINUE
C
  510 IF (WDIR .GT. 180.0) WDIR = 360.0 - WDIR
      IF (WDIR .GT. 180.0) GO TO 510
  520 IF (WDIR .LT.   0.0) WDIR = ABS(WDIR)
      IF (WDIR .LT.   0.0) GO TO 520
      DRLAG1 = WDIR
C
      IF (TRACE .EQ. ON) WRITE (IS,9006)
 9006 FORMAT (' M7')
C
C=====================================================================
C=====================================================================
C
C  TEMPERATURE SECTION
C
C  COMPUTE THE SHORTWAVE RADIATION ATTENUATION DUE TO CLOUD COVER.
C
      KBAR = 1.00 - 0.65*CLD*CLD
C
C  COMPUTE HOURLY TEMPERATURES
C
C
         CALL TEMPSN (ST2,DTSL,SR,SS,BCOEF,
     $                  C0,C1,C2,C3,C4,C5,C6,CLD,KBAR,CTO,
     $                  WSP,WDIR,TMPLAG,THT,T)
C
C
C  NOTE THAT TEMP1 AND TMPLAG ARE DIFFERENT VARAIBLES!! TMPLAG
```

```
C  DOES NOT HAVE THE DEVIATIONS TERM ADDED IN AND IS USED ONLY
C  IN THE 'REGRESSION' PORTION OF THE TEMPERATURE GENERATION
C  COMPONENT.  TEMP1 IS USED WHEN THE ACTUAL LAG-1 TEMPERATURE
C  IS REQUIRED.
C
      TMPLAG = THT
      TPR = TMPLAG
C
C  ADD THE RANDOM COMPONENT TO THE TEMPERATURE JUST COMPUTED.
C
      CALL NORMAL (ARV)
C
      TDEV = TDBIAS + TDRHO*(TTLAG1 - TDBIAS) +
     $         ARV*TDSDEV*SQRT(1.00 - TDRHO*TDRHO)
C
C
C
      TTLAG1 = TDEV
      TEMP = THT + TDEV
C
C
C
      IF (TRACE .EQ. ON) WRITE (IS,9007)
 9007 FORMAT (' M8')
C
C======================================================================
C======================================================================
C
C  DEWPOINT TEMPERATURE SECTION
C
       IF (TYPE .EQ. 'REGRS') GO TO 560
       IF (TYPE .EQ. 'INDEP') GO TO 570
C
C
         WRITE (IS,80) TYPE
   80    FORMAT (//1X, 'INVALID DEWPOINT MODEL TYPE....',A5)
C
           STOP
C
C
C                                           ***************************
C                                           *** REGRESSION DEWPOINTS ***
C                                           ***************************
C
C
  560 CONTINUE
C
      CALL DEWSIM (ACOEF,DWLAG1,TEMP,CLD,WDIR,WSP,DEWR)
C
C  ADD DEVIATIONS TO GENERATED DEW POINTS
C
      CALL NORMAL(ARV)
      DEWDEV = DWBIAS + DWDRHO*(DWDLAG - DWBIAS) +
     $            ARV*DWDEV*SQRT(1.00 - DWDRHO*DWDRHO)
C
```

```
      DWDL    = DWDLAG
      DWDLAG = DEWDEV
      DEW = DEWR + DEWDEV
C
      IF (DEW .GE. TEMP) DEW = 0.99*TEMP
      DWL = DWLAG1 + DWDL
      DWLAG1 = DEWR
C
      GO TO 580
C
  570 CONTINUE
C
C                                              *****************************
C                                              ** INDEPENDENT DEWPOINTS **
C                                              *****************************
C
C
      CALL NORMAL (DWX)
      DWL = DWLAG1
      DEW = DWBAR + DWRHO*(DWLAG1-DWBAR) + DWX*DWSDEV*SQRT(1.-DWRHO**2)
      DWLAG1 = DEW
C
      IF (DEW .GE. TEMP) DEW = 0.99*TEMP
C
  580 CONTINUE
C
      IF (TRACE .EQ. ON) WRITE (IS,9009)
 9009 FORMAT (' M9')
C
C
C================================================================
C================================================================
C
C  LONGWAVE RADIATION SECTION
C
      CALL LONGWV (TEMP1,TEMP,DEW,DWL,CCLAG1,CLD,LW)
C
      TEMP1 = TEMP
C
C
      IF (TRACE .EQ. ON) WRITE (IS,9008)
 9008 FORMAT (' M10')
C
C================================================================
C================================================================
C
C  METRIC CONVERSION SECTION
C
      TEMPM = (TEMP - 32.00)*(5.00/9.00)
      DEWM  = (DEW  - 32.00)*(5.00/9.00)
      WSPM  =  WSP*0.4470
      RAINM =  RAIN*25.4
C
C  STORE DATA IN THE HOURLY ARRAYS FOR AUTOCORRELATION ANALYSIS
C
```

```
      CLDATA(JP) = CLD
      WSDATA(JP) = WSPM
      WDDATA(JP) = WDIR
      DWDATA(JP) = DEWM
      TCDATA(JP) = TEMPM
C
C===================================================================
C===================================================================
C
C  COMPUTE DEWPOINT DEPRESSION AND OUTPUT FOR LATER ANALYSIS
C
      DEP(JP) = (TEMPM - DEWM)*WSPM
C
      IF ( JP .EQ. 24 ) WRITE (IU,582) DEP
  582 FORMAT (16F5.1/8F5.1)
C
C
C===================================================================
C===================================================================
C
C
C  OUTPUT DATA FOR LAND SURFACE MODEL
C
      IF(IPUNCH.GT.0)CALL PUNCH (IPUNCH, RAINM,VP, WSPM, SWR, LW, TEMPM)
C
C===================================================================
C===================================================================
C
C
      GO TO 506
C
C  DEBUG STATEMENT
C
      WRITE (IS,600)JHOUR,JHNEXT,I,RAINM,CLD,SWR,WSPM,WDIR,TEMPM,LW,DEWM
  600 FORMAT (1H ,I5,4X,I6,3X,I2,3X,F5.2,3X,F4.2,2X,F4.1,2X,F4.0,
     $          2X,F5.0,2X,F5.0,2X,F5.1,2X,F4.0)
  506 CONTINUE
C
      IH = I
C
C
C===================================================================
C===================================================================
C
C  DATA PLOT SECTION
C
      IF (DPLOT .NE. 'Y') GO TO 507
C
      PLOTL = -20.
      IF (IMO.GE.4 .AND. IMO.LE.10) PLOTL = 0.
      PLOTU = PLOTL + 40.
C
      CALL PLOT (IMO,IDY,IH,80,TEMPM,DEWM,SWR,LW,CLD,RAINM,WSPM,WDIR,
     $              STORM,IPL,PLOTL,PLOTU)
C
```

```
      IF (TRACE .EQ. ON) WRITE (IS,9010)
 9010 FORMAT (' M11')
C
C=====================================================================
C=====================================================================
C
  507 CONTINUE
C
      IF(I .EQ. 23) IDY = IDY + 1
      IF (IDY .GT.31) IDY = 1
C
      JHOUR = JHOUR + 1
      ST1 = ST2
      NDATA = NDATA + 1
C
C  UPDATE THE STATISTICAL ANALYSIS
C
      CALL MSTAT (1,TEMPM,DEWM,CLD,WSPM,WDIR,RAWSUM,XXT)
C
      IF (TRACE .EQ. ON) WRITE (IS,9020)
 9020 FORMAT (' 11A')
C
C
C=====================================================================
C=====================================================================
C
C  UPDATE AUTOCORRELATION ANALYSIS
C
C  THE FIRST 24 HOURS OF THE MONTH ARE NEEDED TO FILL UP THE DATA
C  ARRAYS TO BEGIN THE AUTOCORRELATION ANALYSIS.
C
      IF (JHOUR .LE. 24) GO TO 550
        NRDATA = NRDATA + 1
      CALL RAWLAG(MLAG,JP,TCDATA,TCSUM,TCSMSQ,TCSUM3,TCB,TCV,TCK,NRDATA,
     $      TCRHO)
      CALL RAWLAG(MLAG,JP,DWDATA,DWSUM,DWSMSQ,DWSUM3,DPB,DPV,DPK,NRDATA,
     $      DPRHO)
      CALL RAWLAG(MLAG,JP,CLDATA,CLSUM,CLSMSQ,CLSUM3,CLB,CLV,CLK,NRDATA,
     $   .  CLRHO)
      CALL RAWLAG(MLAG,JP,WSDATA,WSSUM,WSSMSQ,WSSUM3,WSB,WSV,WSK,NRDATA,
     $      WSRHO)
      CALL RAWLAG(MLAG,JP,WDDATA,WDSUM,WDSMSQ,WDSUM3,WDB,WDV,WDK,NRDATA,
     $      WDRHO)
C
      IF (TRACE .EQ. ON) WRITE (IS,9021)
 9021 FORMAT (' 11B')
C
  550 CONTINUE
C
C=====================================================================
C=====================================================================
C
C  DIURNAL CURVE SECTION
C
C
```

```
C   COMPUTE MEAN AND STANDARD DEVIATIONS FOR EACH HOUR OF THE DAY.
C   SKEWS ARE NOT COMPUTED.
C
C
       IF ( I .EQ. 0 ) NDC = NDC + 1
C
       CALL STAT (SWR,ASWRS(JP),ASWRSQ(JP),ZZZ,ASWRB(JP),ASWRSD(JP),
     $              -999.0,NDC)
       CALL STAT (CLD,ACLDS(JP),ACLDSQ(JP),ZZZ,ACLDB(JP),ACLDSD(JP),
     $              -999.0,NDC)
       CALL STAT (WSPM,AWSPS(JP),AWSPSQ(JP),ZZZ,AWSPB(JP),AWSPSD(JP),
     $              -999.0,NDC)
       CALL STAT(WDIR,AWDRS(JP),AWDRSQ(JP),ZZZ,AWDRB(JP),AWDRSD(JP),
     $              -999.0,NDC)
       CALL STAT(TEMPM,ATMPS(JP),ATMPSQ(JP),ZZZ,ATMPB(JP),ATMPSD(JP),
     $              -999.0,NDC)
       CALL STAT (DEWM,ADEWS(JP),ADEWSQ(JP),ZZZ,ADEWB(JP),ADEWSD(JP),
     $              -999.0,NDC)
       CALL STAT ( LW,ALWRS(JP),ALWRSQ(JP),ZZZ,ALWRB(JP),ALWRSD(JP),
     $              -999.0,NDC)
C
       IF (TRACE .EQ. ON) WRITE (IS,9022)
 9022  FORMAT (' 11C')
C
C
C=======================================================================
C=======================================================================
C
C
C   UPDATE RAINFALL STATISTICS
C
       CALL RSTAT (TRSUM,TBSUM,DHSUM,TRSBAR,TBSBAR,DHBAR,RAINM,STORMS,
     $               DRY,   STORM)
C
       IF (TRACE .EQ. ON) WRITE (IS,9023)
 9023  FORMAT (' 11D')
C
C
C=======================================================================
C=======================================================================
C
C
C   UPDATE THE HISTOGRAMS
C
         CALL HGRAM (TCHIST,TCHDIM,TEMPM,TCDT,TCBASE)
         CALL HGRAM (DPHIST,DPHDIM,DEWM,DPDT,DPBASE)
         CALL HGRAM (CLHIST,CLHDIM,CLD ,CLDT,CLBASE)
         CALL HGRAM (WSHIST,WSHDIM,WSPM,WSDT,WSBASE)
         CALL HGRAM (WDHIST,WDHDIM,WDIR,WDDT,WDBASE)
C
       IF (TRACE .EQ. ON) WRITE (IS,9011)
 9011  FORMAT (' M12')
C
  500  CONTINUE
C
```

```
C
C=====================================================================
C=====================================================================
C
C
      IF (JULREL .LT. JREND) GO TO 390
C
C  RESET MONTHLY COUNTERS
C
      JHOUR = 0
C
C
C  RESTART STORM SEQUENCE
C
      CALL RAINST (TB,TK,D,JSINCE,STORM,JHREQS,JHNEXT)
C
  390 CONTINUE
C
C  UPDATE THE DAY COUNTERS
C
      CALL DATEM
C
C  CHECK FOR END OF RUN
C
      IF (TRACE .EQ. ON) WRITE (IS,9012)
 9012 FORMAT (' M13')
C
      IF (JULDAT .LE. JULEND) GO TO 400
C
C
C=====================================================================
C=====================================================================
C
C  CALL THE FINAL STATISTICAL ANALYSIS SUBROUTINE
C
      CALL FSTAT (5,RAWSUM,XXT,MEAN,COVMAT,CORMAT,NDATA)
C
C  CALL THE AUTOCORRELATION  SUBROUTINE
C
      CALL AUTOCO (MLAG,TCRHO,TCB,TCV,NRDATA,TCTITL)
      CALL AUTOCO (MLAG,DPRHO,DPB,DPV,NRDATA,DWTITL)
      CALL AUTOCO (MLAG,CLRHO,CLB,CLV,NRDATA,CLTITL)
      CALL AUTOCO (MLAG,WSRHO,WSB,WSV,NRDATA,WSTITL)
      CALL AUTOCO (MLAG,WDRHO,WDB,WDV,NRDATA,WDTITL)
C
      IF (TRACE .EQ. ON) WRITE (IS,9013)
 9013 FORMAT (' M14')
C
C
C=====================================================================
C=====================================================================
C
C
C        OUTPUT RESULTS
C
```

```
C
      WRITE(IS,715) (MEAN(I),I=1,5)
  715 FORMAT(1H1////T28,'MEAN VALUES'/T9,'TCBAR',T21,'DWBAR',
     $ T33,'CLBAR',T45,'WSBAR',T57,'WDBAR'/1X,5F12.2/)
C
      WRITE(IS,720) ((COVMAT(I,J),J=1,5),I=1,5)
  720 FORMAT(1X,T24,'COVARIANCE MATRIX',/(5(1X,5F12.2/)))
C
      WRITE(IS,740) ((CORMAT(I,J),J=1,5),I=1,5)
  740 FORMAT(1X,T24,'CORRELATION MATRIX',/(5(1X,5F12.2/)))
C
      WRITE (IS,745) TCK,DPK,CLK,WSK,WDK
  745 FORMAT (/T25,'SKEW COEFFICIENTS'/T8,'TCSKEW',T20,'DWSKEW',T32,
     $ 'CLSKEW',T44,'WSSKEW',T56,'WDSKEW'/1X,5F12.2)
C
      WRITE (IS,747) TBSBAR, TRSBAR, DHBAR
  747 FORMAT (///T20,'RAINFALL OUTPUT STATISTICS'//
     $  T24,'TB',T32,'TR',T40,'D'//T22,F6.2,T31,F5.2,T37,F5.2///)
C
C
C=======================================================================
C=======================================================================
C
C
C  PRINT HISTOGRAMS OF THE GENERATED DATA
C
      CALL PRINTH (TCHIST,TCHDIM,TCDT,TCBASE,TCTITL,NDATA)
      CALL PRINTH (DPHIST,DPHDIM,DPDT,DPBASE,DWTITL,NDATA)
      CALL PRINTH (CLHIST,CLHDIM,CLDT,CLBASE,CLTITL,NDATA)
      CALL PRINTH (WSHIST,WSHDIM,WSDT,WSBASE,WSTITL,NDATA)
      CALL PRINTH (WDHIST,WDHDIM,WDDT,WDBASE,WDTITL,NDATA)
C
C
C=======================================================================
C=======================================================================
C
C
C  CONVERT VARIANCES TO STANDARD DEVIATIONS.
C
      DO 800 IG = 1,24
C
      ASWRSD(IG) = SQRT(ASWRSD(IG))
      ACLDSD(IG) = SQRT(ACLDSD(IG))
      AWSPSD(IG) = SQRT(AWSPSD(IG))
      AWDRSD(IG) = SQRT(AWDRSD(IG))
      ATMPSD(IG) = SQRT(ATMPSD(IG))
      ADEWSD(IG) = SQRT(ADEWSD(IG))
      ALWRSD(IG) = SQRT(ALWRSD(IG))
C
  800 CONTINUE
C
C
C=======================================================================
C=======================================================================
C
```

```
C
C  COMPUTE TOTAL DAILY SHORT AND LONGWAVE RADIATION
C
      DO 810 IG = 1,24
C
      SUMSW = SUMSW + ASWRB(IG)
      SUMLW = SUMLW + ALWRB(IG)
C
  810 CONTINUE
C
C
C===================================================================
C===================================================================
C
C
C  PRINT HOURLY MEANS AND STANDARD DEVIATIONS.
C
      CALL HOUR (ASWRB,ASWRSD,'SWR')
        WRITE (IS,811) SUMSW
  811   FORMAT (T29,'TOTAL=',T37,F7.2)
      CALL HOUR (ALWRB,ALWRSD,'LWR')
        WRITE (IS,811) SUMLW
      CALL HOUR (ACLDB,ACLDSD,'CLD')
      CALL HOUR (AWSPB,AWSPSD,'WSP')
      CALL HOUR (AWDRB,AWDRSD,'WDR')
      CALL HOUR (ATMPB,ATMPSD,'TMP')
      CALL HOUR (ADEWB,ADEWSD,'DEW')
C
      IF (TRACE .EQ. ON) WRITE (IS,9014)
 9014 FORMAT (' M15')
C
      WRITE (IS,760)
  760 FORMAT (1H1,15(15(5H      )/))
      STOP
      END
C
C..................................................................
C
      SUBROUTINE VARYX (X1,X2,T1,T2,ST,X)
C
C  ROUTINE TO LINEARLY INTERPOLATE A VALUE OF X
C
      RANGE1 = T2 - T1
      RANGE2 = 24.0 - RANGE1
C
      IF (ST .GT. T1) GO TO 100
      X = X2 - (X2 - X1)*((24.0 - T2 + ST)/RANGE2)
      RETURN
C
  100 IF (ST .GT. T2) GO TO 200
      X = X1 + (X2 - X1)*((ST - T1)/RANGE1)
      RETURN
C
  200 CONTINUE
      X = X2 - (X2 - X1)*((ST - T2)/RANGE2)
```

```
C
   10 U1 = RAND(0)
      IF (U1 .LT. 0.00001) GO TO 10
      U2 = RAND(0)
      X = SQRT (-2.0*ALOG(U1))*COS(6.28319*U2)
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE VAPOR (T,TD,E,ES)
C
C
C  ROUTINE TO COMPUTE ATMOSPHERIC VAPOR PRESSURE GIVEN
C  TWO METER TEMPERATURE AND DEWPOINT.
C
C     T     ... TEMPERATURE - DEG C
C     TD    ... DEWPOINT TEMPERATURE - DEG C
C     E     ... VAPOR PRESSURE - MILLIBAR
C     ES    ... SATURATED VAPOR PRESSURE - MILLIBAR
C     C0-C5... COEFFICIENTS IN SAT. VAPOR PRESS. APPROX.
C     R     ... RELATIVE HUMIDITY
C
C
C
      DOUBLE PRECISION C0,C1,C2,C3,C4,C5
C
      DATA C0/6.0689226     /
      DATA C1/4.4358312E-01/
      DATA C2/1.4590816E-02/
      DATA C3/2.7619554E-04/
      DATA C4/2.9952590E-06/
      DATA C5/1.4398885E-08/
C
C
C
C  COMPUTE RELATIVE HUMIDITY
C
      R = ((112. - 0.1*T + TD)/(112. + 0.9*T))**8.0
C
C
C   COMPUTE SATURATION VAPOR PRESSURE
C
      X = C4 + T*C5
      X = C3 + T*X
      X = C2 + T*X
      X = C1 + T*X
      ES= C0 + T*X
C
C
C  COMPUTE ATMOSPHERIC VAPOR PRESSURE
C
      E = R*ES
C
```

```
C
      RETURN
      END
C
C...............................................................
C
      SUBROUTINE INTER (DAFILE,IS,DPLOT,IPL)
C
C  ROUTINE TO READ THE NECCESSARY RUNTIME INFORMATION FROM THE
C  CONSOLE.
C
      DIMENSION PTEXT(16)
C
      COMMON /SEED/  ISEED
      COMMON /PUNCHD/ PTEXT, PNFILE, IPUNCH
C
      DOUBLE PRECISION  DAFILE, PNFILE
C
C
C  SET THE CONSOLE UNIT NUMBER FOR THIS MACHINE.
C
      IC = 5
C
      DAFILE = 'DCCMOD.DAT'
      WRITE (IC,90) DAFILE
   90 FORMAT (//1X,'THE CURRENT DATA FILE IS ',A10/
     $  1X,'DO YOU WISH TO READ A DIFFERENT ONE? (Y/N)')
C
      READ (IC,110) ANS
      IF (ANS .NE. 'Y') GO TO 115
C
C
      WRITE (IC,100)
  100 FORMAT (//1X,'WHAT DATA FILE CONTAINS THE INPUT DATA?'/
     $ 1X,'    ENTER FILE NAME IN THE FORM  XXXXXX.YYY ')
C
      READ (IC,110) DAFILE
  110 FORMAT (A)
C
  115 CONTINUE
C
      WRITE (IC,120)
  120 FORMAT (/1X,'DO YOU WANT TO PRINT THE INPUT SUMMARY TO THE ',
     $ 'CONSOLE? (Y/N)')
C
      READ (IC,130)ANS
  130 FORMAT (A)
C
      IF (ANS .EQ.'Y') IS = IC
C
      WRITE (IC,140)
  140 FORMAT (/1X,'INPUT SEED FOR THE RANDOM NUMBER GENERATOR')
      READ (IC,150) ISEED
  150 FORMAT (I)
C
```

```
C
   10 U1 = RAND(0)
      IF (U1 .LT. 0.00001) GO TO 10
      U2 = RAND(0)
      X = SQRT (-2.0*ALOG(U1))*COS(6.28319*U2)
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE VAPOR (T,TD,E,ES)
C
C
C   ROUTINE TO COMPUTE ATMOSPHERIC VAPOR PRESSURE GIVEN
C   TWO METER TEMPERATURE AND DEWPOINT.
C
C      T     ... TEMPERATURE - DEG C
C      TD    ... DEWPOINT TEMPERATURE - DEG C
C      E     ... VAPOR PRESSURE - MILLIBAR
C      ES    ... SATURATED VAPOR PRESSURE - MILLIBAR
C      C0-C5... COEFFICIENTS IN SAT. VAPOR PRESS. APPROX.
C      R     ... RELATIVE HUMIDITY
C
C
C
      DOUBLE PRECISION C0,C1,C2,C3,C4,C5
C
      DATA C0/6.0689226     /
      DATA C1/4.4358312E-01/
      DATA C2/1.4590816E-02/
      DATA C3/2.7619554E-04/
      DATA C4/2.9952590E-06/
      DATA C5/1.4398885E-08/
C
C
C
C   COMPUTE RELATIVE HUMIDITY
C
      R = ((112. - 0.1*T + TD)/(112. + 0.9*T))**?.0
C
C
C    COMPUTE SATURATION VAPOR PRESSURE
C
      X = C4 + T*C5
      X = C3 + T*X
      X = C2 + T*X
      X = C1 + T*X
      ES= C0 + T*X
C
C
C   COMPUTE ATMOSPHERIC VAPOR PRESSURE
C
      E = R*ES
C
```

```
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE INTER (DAFILE,IS,DPLOT,IPL)
C
C  ROUTINE TO READ THE NECCESSARY RUNTIME INFORMATION FROM THE
C  CONSOLE.
C
      DIMENSION PTEXT(16)
C
      COMMON /SEED/  ISEED
      COMMON /PUNCHD/ PTEXT, PNFILE, IPUNCH
C
      DOUBLE PRECISION  DAFILE, PNFILE
C
C
C  SET THE CONSOLE UNIT NUMBER FOR THIS MACHINE.
C
      IC = 5
C
      DAFILE = 'DCCMOD.DAT'
      WRITE (IC,90) DAFILE
   90 FORMAT (//1X,'THE CURRENT DATA FILE IS ',A10/
     $  1X,'DO YOU WISH TO READ A DIFFERENT ONE? (Y/N)')
C
      READ (IC,110) ANS
      IF (ANS .NE. 'Y') GO TO 115
C
C
      WRITE (IC,100)
  100 FORMAT (//1X,'WHAT DATA FILE CONTAINS THE INPUT DATA?'/
     $ 1X,'    ENTER FILE NAME IN THE FORM  XXXXXX.YYY ')
C
      READ (IC,110) DAFILE
  110 FORMAT (A)
C
  115 CONTINUE
C
      WRITE (IC,120)
  120 FORMAT (/1X,'DO YOU WANT TO PRINT THE INPUT SUMMARY TO THE ',
     $ 'CONSOLE? (Y/N)')
C
      READ (IC,130)ANS
  130 FORMAT (A)
C
      IF (ANS .EQ.'Y') IS = IC
C
      WRITE (IC,140)
  140 FORMAT (/1X,'INPUT SEED FOR THE RANDOM NUMBER GENERATOR')
      READ (IC,150) ISEED
  150 FORMAT (I)
C
```

```
      WRITE (IC,160)
  160 FORMAT (/1X,'DO YOU WANT A PLOT? (Y/N)')
C
      READ (IC,130) DPLOT
      IF (DPLOT .NE. 'Y') GO TO 185
C
      WRITE (IC,170)
  170 FORMAT (/1X,'WHICH PLOT?  1 FOR 6-VARIABLE'/
     $          1X,'             2 FOR 4-VARIABLE'/)
C
      READ (IC,180) IPL
  180 FORMAT (I)
C
C
  185 CONTINUE
C
      IPUNCH = 0
      WRITE (IC,190)
  190 FORMAT (/1X,'CREATE LAND SURFACE MODEL DATA FILE? (Y/N)')
C
      READ (IC,130) ANS
      IF (ANS .NE. 'Y') GO TO 230
C
      IPUNCH = 27
      WRITE (IC,200)
  200 FORMAT (/1X,'ENTER DATA FILE NAME ... XXXXXX.YYY')
C
      READ (IC,110) PNFILE
C
      WRITE (IC,210)
  210 FORMAT (/1X, 'ENTER COMMENTS TO IDENTIFY OUTPUT DATA ',
     $             '(80 CHAR. MAX)')
      READ (IC,220) (PTEXT(I),I=1,16)
  220 FORMAT (16A5)
C
      WRITE (IC,240) IPUNCH,PNFILE,(PTEXT(I),I=1,16)
  240 FORMAT (1X,I5/1X,A10/1X,16A5)
  230 CONTINUE
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE START (ISEED)
C
C         ACTIVATE RAND ISEED TIMES TO PROVIDE A DIFFERENT
C            STARTING POINT IN THE GENERATION OF RANDOM NUMBERS
C            WITH EACH INPUT OF ISEED
C
C
      DO 100 I = 1,ISEED
C
         X = RAND(0)
C
```

```
  100 CONTINUE
C
C
      RETURN
      END
C
C......................................................................
C
      SUBROUTINE READF (IN,IS,IB)
C
C
C   THIS SUBPROGRAM READS THE INPUT DATA FOR THE STOCHASTIC
C   HYDROMETEOROLOGICAL MODEL.
C
C
      DIMENSION TITLE(16,3), BCOEF(8), ACOEF(8)
      DIMENSION CCPDF(30), CCORD(30)
      DIMENSION DRPDF(30), DRORD(30)
C
C
      DOUBLE PRECISION  WRITEF, BUGOFF, TZONE, TZ(4), DEBUG(7)
      DOUBLE PRECISION  OUTPUT
C
C
      REAL LAT(3), LONG(3)
C
      COMMON /TITLES/ TITLE
      COMMON /FILES/  WRITEF, OUTPUT, BUGOFF
      COMMON /DATES/  IYR,IMO,IDAY, LYR,LMO,LDAY
      COMMON /LOCATE/ LAT, LONG, TZONE
      COMMON /DBUG/   NBUG,DEBUG
      COMMON /CLOUDS/ CCBAR,CCSD,CCRHO,BETA,GAM
      COMMON /ATMOS/  EN
      COMMON /PDFCLD/ NUMCC,CCPDF,CCORD,CCA,CCB
      COMMON /RAINS/  TBBAR, TRBAR, DBAR
      COMMON /TEMPAR/ TDBIAS, TDSDEV, TDRHO, BCOEF, TEMPAR
      COMMON /WINDSP/ SPBAR1,SPBAR2,SPRT1,SPRT2,SPSDV1,SPSDV2,
     $                SPSDT1,SPSDT2,SPSKEW,SPRHO
      COMMON /WINDIR/ DRBAR,DRDEV,DRRHO
      COMMON /PDFDIR/ NUMDR,DRPDF,DRORD,DRA,DRB
      COMMON /DEWONE/ TYPE, ACOEF
      COMMON /DEWTWO/ DWBAR,DWSDEV,DWSKEW,DWRHO
      COMMON /DEWDVS/ DWBIAS, DWDEV, DWDRHO
C
C  READ THE GENERAL DATA SECTION
C
C    NOTE: 'DUMMY' READS ARE INSERTED TO READ THE 'CARDS' THAT SEPERATE
C          THE MAJOR SECTIONS OF THE INPUT DATA.  IT IS DESIGNED TO
C          MAKE HANDLING THE DATA DECK EASIER AND AS A MEANS TO MAKE
C          EXAMINATION OF THE DATA DECK EASIER.
C
C
C
      READ (IN,10) DUMMY
   10 FORMAT (A)
```

```
C
C  READ THE TITLE CARDS (3)
C
      DO 15 I = 1,3
         READ (IN,20) (TITLE(J,I),J=1,15)
  20     FORMAT (15A5)
  15     CONTINUE
C
C  READ THE DATA FILE NAMES FOR THE GENERAL OUTPUT AND DEBUG INFO
C
      READ (IN,30) WRITEF, OUTPUT, BUGOFF
  30 FORMAT (10X,3A10)
C
C=====================================================================
C
C OPEN FILES FOR OUTPUT
C
      IB = 22
      OPEN (UNIT=IB,DEVICE='DSK',ACCESS='SEQOUT',FILE=BUGOFF)
      IF (IS .EQ. 5) GO TO 35
      IS = 23
      OPEN (UNIT=IS,DEVICE='DSK',ACCESS='SEQOUT',FILE=WRITEF)
  35 CONTINUE
C
C=====================================================================
C
C  READ DATES, LATITUDE, LONGITUDE, AND TIME ZONE
C
      READ (IN,40) IMO,IDAY,IYR,LMO,LDAY,LYR,
     $      (LAT(I),I=1,3),(LONG(I),I=1,3),TZONE
  40 FORMAT (10X,I2,1X,I2,1X,I4,1X,I2,1X,I2,1X,I4,1X,
     $      2(F3.0,1X,F2.0,1X,F2.0,1X),A10)
C
C  READ DEBUG INFO
C
C   DEBUG INFORMATION CAN BE OUTPUT FROM SEVERAL SUBROUTINES BY SIMPLY
C   READING IN THE APPROPRIATE SUBROUTINE NAME. THESE SUBROUTINES
C   INCLUDE:  TAU, DECL, SOLRAD, CLRSKY, COVER, ARVA, AND TEMPSN.
C   NAMES ARE LEFT JUSTIFIED.
C
C   THIS FEATURE IS CURRENTLY DISABLED.
C
      NBUG = 0
      GO TO 51
C
      READ (IN,50) NBUG,(DEBUG(I),I=1,NBUG)
  50 FORMAT (10X,I2,6A10)
  51 CONTINUE
C
C  READ CLOUD AND RADIATION DATA
C
      READ (IN,10) DUMMY
C
C  READ PARAMETER CARD
C
```

```
      READ (IN,60) CCBAR, CCSD, CCRHO, GAM, EN
   60 FORMAT (10X,6F10.0)
C
      READ (IN,62) NUMCC, CCA, CCB
   62 FORMAT (10X,I2,8X,2F10.0)
C
      READ (IN,64) (CCPDF(I),I=1,NUMCC)
      READ (IN,64) (CCORD(I),I=1,NUMCC)
   64 FORMAT (10X,6F10.0)
C
C   READ RAIN MODEL PARAMETERS
C
      READ (IN,10) DUMMY
      READ (IN,60) TBBAR, TRBAR, OBAR
C
C   READ TEMPERATURE DATA
C
      READ (IN,10) DUMMY
      READ (IN,70) (BCOEF(I),I=1,8)
   70 FORMAT(10X,4E12.5)
      READ (IN,60)  TDBIAS, TDSDEV, TDRHO
C
C   READ WIND SPEED PARAMETER DATA
C
      READ (IN,10) DUMMY
      READ (IN,60) SPBAR1, SPBAR2, SPBT1, SPBT2
      READ (IN,60) SPSDV1, SPSDV2, SPSDT1, SPSDT2
      READ (IN,60) SPSKEW, SPRHO
C
C   READ WIND DIRECTION DATA
C
      READ (IN,10) DUMMY
      READ (IN,60) DRBAR, DRDEV, DRRHO
      READ (IN,62) NUMDR,DRA,DRB
      READ (IN,64) (DRPDF(I),I=1,NUMDR)
      READ (IN,64) (DRORD(I),I=1,NUMDR)
C
C   READ DEWPOINT MODEL PARAMETERS
C
      READ (IN,10) DUMMY
      READ (IN,10) TYPE
C
      IF (TYPE.EQ.'REGRS' .OR. TYPE.EQ.'INDEP') GO TO 100
C
      WRITE (IS,95) TYPE
   95 FORMAT (///,T5,'****** DEWPOINT MODEL TYPE --',A5,'-- IS INVALID.'/
     $            T11,      'ONLY ''INDEP'' OR ''REGRS'' ARE ACCEPTABLE')
      STOP
C
C
  100 IF ( TYPE .EQ. 'INDEP' ) READ (IN,60) DWBAR, DWSDEV,  DWRHO
      IF ( TYPE .EQ. 'REGRS' ) READ (IN,70) (ACOEFF(I),I=1,6)
      IF ( TYPE .EQ. 'REGRS' ) READ (IN,60) DWBIAS, DWDEV, DWDRHO
C
C
```

```
C
C
C
C ::::: INPUT DATA SUMMARY  ::::::::::::::::::::::::::::::::
C
C
C  PRINT GENERAL DATA
C
C
      WRITE (IS,490)
  490 FORMAT (1H1,3(15(5H      )/))
C
      WRITE (IS,491)
  491 FORMAT (1X,79(1H*)/)
C
      WRITE (IS,492)
  492 FORMAT (1X,T20,'CONSTRAINED STOCHASTIC CLIMATE SIMULATION'/
     $            T33,'INPUT SUMMARY'/)
C
      WRITE (IS,491)
C
C
      DO 510 J = 1,3
      WRITE (IS,500) (TITLE(I,J),I=1,15)
  500 FORMAT (1H ,15A5)
  510 CONTINUE
C
      WRITE (IS,491)
C
      IF (IS.EQ.5) WRITEF = 'CONSOLE'
      WRITE (IS,515) WRITEF, OUTPUT, BUGOFF
  515 FORMAT (//1X,T31,'OUTPUT FILE NAMES'//T31,'WRITEF: ',A10/
     $                                    T31,'OUTPUT: ',A10/
     $                                    T31,'BUGOFF: ',A10/)
C
      WRITE (IS,491)
C
      WRITE (IS,520) IMO,IDAY,IYR,LMO,LDAY,LYR
  520 FORMAT (//T12,'BEGINNING DATE ',2X,I2,'/',I2,'/',I4,5X,
     $ 'ENDING DATE ',2X,I2,'/',I2,'/'I4)
C
      WRITE (IS,530) (LAT(I),I=1,3),(LONG(I),I=1,3),TZONE
  530 FORMAT(//1X,'LATITUDE = ',2X,F4.0,F3.0,F3.0,5X,
     $ 'LONGITUDE = ',2X,F4.0,F3.0,F3.0,5X,'TIME ZONE =  ',A10/)
C
      WRITE (IS,491)
C
      IF (NBUG.EQ.0) GO TO 545
      WRITE (IS,540)(DEBUG(I),I=1,NBUG)
  540 FORMAT (//1X,'DEBUG SUBROUTINES - ',7A10)
      WRITE (IS,491)
  545 CONTINUE
C
C
C  PRINT CLOUD AND SKY PARAMETERS
```

```
C
      WRITE (IS,550)
  550 FORMAT (1X,T28,'CLOUD AND SKY PARAMETERS'/)
C
      WRITE (IS,551) CCBAR,CCSD,CCRHO
  551 FORMAT (1X,T24,'FAIRWEATHER CLOUD COVER'/
     $            T31,'MEAN',T42,F6.2/
     $            T31,'ST. DEV.',T42,F6.2/
     $            T31,'LAG-1 COEF. ',T42,F6.2//)
C
      WRITE (IS,552)
  552 FORMAT (1X,T23,'FAIRWEATHER CLOUD COVER HISTOGRAM'/)
C
      CALL PRDIST ( CCPDF, CCORD,  NUMCC )
C
C
C
      WRITE (IS,562) CCA,CCB
  562 FORMAT (//T5,'LEFT BOUND OF HIST. = ',F10.4,5X,
     $              'RIGHT BOUND OF HIST. = ',F10.4)
C
      WRITE (IS,553) GAM, EN
  553 FORMAT (//1X,T23,'CLOUD COVER DECAY COEFFICENT = ',F5.3/
     $              T23,'ATMOSPHERIC TURBIDITY FACTOR = ',F4.1/)
C
      WRITE (IS,491)
      WRITE (IS,490)
      WRITE (IS,491)
      WRITE (IS,492)
      WRITE (IS,491)
C
C  PRINT PRECIPITATION MODEL PARAMETERS
C
      WRITE (IS,565) TBBAR,TRBAR,DBAR
  565 FORMAT (/1X,T25,'PRECIPITATION MODEL PARAMETERS'//
     $            T23,'MEAN TIME BETWEEN STORMS',T50,F7.2/
     $            T23,'MEAN STORM DURATION',T50,F7.2/
     $            T23,'MEAN STORM DEPTH',T50,F7.2/)
C
      WRITE (IS,491)
C
C
C
C  PRINT TEMPERATURE MODEL PARAMETERS
C
      WRITE (IS,570)(BCOEF(I),I=1,8)
  570 FORMAT (//1X,T26,'TEMPERATURE MODEL PARAMETERS'//
     $  T22,'B0',2X,E12.5,4X,'B1',2X,E12.5/
     $  T22,'B2',2X,E12.5,4X,'B3',2X,E12.5/
     $  T22,'B4',2X,E12.5,4X,'B5',2X,E12.5/
     $  T22,'B6',2X,E12.5,4X,'B7',2X,E12.5//)
C
C
      WRITE (IS,580) TDBIAS, TDSDEV, TDRHO
  580 FORMAT (1X, T30,'STOCHASTIC COMPONENT'//
```

```
     $    T28,'TEMPERATURE BIAS',2X,F6.2/
     $    T28,'ST. DEVIATION   ',2X,F6.2/
     $    T28,'LAG-1 COR. COEF.',2X,F6.2/)
C
C
      WRITE (IS,491)
      WRITE (IS,490)
      WRITE (IS,491)
      WRITE (IS,492)
      WRITE (IS,491)
C
C
C  PRINT WIND SPEED MODEL PARAMETERS
C
      WRITE (IS,600) SPBAR1,SPBT1,SPBAR2,SPBT2,
     $               SPSDV1,SPSDT1,SPSDV2,SPSDT2
  600 FORMAT (/T29,'WIND SPEED PARAMETERS'//
     $    T21,'MIN HOURLY MEAN  = ',F4.1,' AT ',F5.2,' HOURS'/
     $    T21,'MAX HOURLY MEAN  = ',F4.1,' AT ',F5.2,' HOURS'//
     $    T21,'MIN HOURLY ST DEV= ',F4.1,' AT ',F5.2,' HOURS'/
     $    T21,'MAX HOURLY ST DEV= ',F4.1,' AT ',F5.2,' HOURS'//)
C
      WRITE (IS,601) SPSKEW, SPRHO
  601 FORMAT (//T29,'SKEW COEFFICIENT ',F5.2/
     $             T29,'LAG-1 COEFFICIENT',F5.2/)
C
      WRITE (IS,491)
C
C
C  PRINT WIND DIRECTION MODEL PARAMETERS
C
      WRITE (IS,620) DRBAR,DRDEV,DRRHO
  620 FORMAT (//T27,'WIND DIRECTION PARAMETERS'//
     $             T31,'MEAN',T42,F6.2/
     $             T31,'ST. DEV.',T42,F6.2/
     $             T31,'LAG-1 COEF',T42,F6.2//)
C
C
      WRITE (IS,630)
  630 FORMAT (/T28,'WIND DIRECTION HISTOGRAM'/)
C
      CALL PRDIST ( DRPDF, DRORD, NUMDR )
C
      WRITE (IS,632) DRA,DRB
  632 FORMAT (//T5,'LEFT BOUND OF HIST. = ',F10.4,5X,
     $                 'RIGHT BOUND OF HIST. = ',F10.4/)
C
      WRITE (IS,491)
C
      IF (TYPE .EQ. 'REGRS') GO TO 665
C
C
C  PRINT DEWPOINT MODEL PARAMETERS ... 'INDEPENDENT GENERATION'
C
      WRITE (IS,640) DWBAR, DWSDEV, DWRHO
```

```
  640 FORMAT (//T27,'DEWPOINT MODEL PARAMETERS'//
     $            T31,'MEAN',T42,F6.2/
     $            T31,'ST DEV',T42,F6.2/
     $            T31,'LAG-1 COEF',T42,F6.2//)
C
      WRITE (IS,491)
C
      GO TO 700
C
  665 CONTINUE
C
C
C  PRINT DEW POINT MODEL PARAMETERS ... 'REGRESSION TYPE '
C
      WRITE (IS,670)(ACOEF(I),I=1,6)
  670 FORMAT (/T21,'DEWPOINT MODEL REGRESSION COEFFICIENTS'//
     $  T22,'A0',2X,E12.5,4X,'A1',2X,E12.5/
     $  T22,'A2',2X,E12.5,4X,'A3',2X,E12.5/
     $  T22,'A4',2X,E12.5,4X,'A5',2X,E12.5//)
C
      WRITE (IS,680) DWBIAS, DWDEV, DWDRHO
  680 FORMAT (//T24,'STOCHASTIC COMPONENT PARAMETERS'/
     $            T28,'DEWPOINT BIAS  ',2X,F6.2/
     $            T28,'ST DEVIATION   ',2X,F6.2/
     $            T28,'LAG-1 COR COEF ',2X,F6.2//)
C
      WRITE (IS,491)
C
  700 CONTINUE
C
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE PRDIST ( H, ORD, NMAX)
C
C  PRINT OUT THE INPUT PROBABILITY MASS FUNCTION
C
C
C
      COMMON /IO/     IN,IS,IR
C
C
      DIMENSION ORD(NMAX)
      DIMENSION H(NMAX)
      DIMENSION FMT(6),FMT1(10)
C
      DATA FMT  /'(T8 ,','','    10','(7H--','------',')) '/
      DATA FMT1 /'    1','    2','    3','    4','    5',
     $           '    6','    7','    8','    9','   10'/
C
      DO 100 J = 1,NMAX,10
C
```

```
      IMAX = 10
      IF( J+10 .GT. NMAX ) IMAX = NMAX - J + 1
C
C
      WRITE (IS,200)(H(I+J-1),I=1,IMAX)
  200 FORMAT (1H ,1X,'HIST ',10(1X,F6.2))
C
      FMT(3) = FMT1(IMAX)
C
C
      WRITE (IS,FMT)
C
C
      WRITE (IS,400) (ORD(I+J-1),I=1,IMAX)
  400 FORMAT(1H ,1X,'ORD  ',10(1X,F6.2)//)
C
C
  100 CONTINUE
C
C
      RETURN
      END
C
C.......................................................................
C
      SUBROUTINE DATE1
C
C         DATE1 INITIALIZES THE DATE COUNTERS.
C         JULIAN DATES ARE USED.
C
C
C         IYR    ... INITIAL YEAR
C         IMO    ... INITIAL MONTH
C         IDAY   ... INITIAL DAY
C         LYR    ... LAST YEAR
C         LMO    ... LAST MONTH
C         LDAY   ... LAST DAY
C
C         JULDAT... CURRENT JULIAN DATE
C         JBEGIN... JULIAN DATE AT BEGINNING OF RUN
C         JULEND... JULIAN DATE AT END OF RUN
C         JRANGE... LENGTH OF RUN
C         JULREF... JAN 1 OF INITIAL YEAR
C         JULREL... JULIAN DATE RELATIVE TO JAN 1 OF CURRENT YEAR
C         JSTART... RELATIVE JULIAN DATE TO BEGIN MONTHLY PARAMETER
C                   ESTIMATION RANGE
C         JSTOP ... RELATIVE JULIAN DATE TO END MONTHLY PARAMETER
C                   ESTIMATION RANGE
C         JYEAR ... YEAR COUNTER
C
C         NXLPYR... JULIAN DATE OF DEC 31 OF NEXT LEAP YEAR
C
C
C
      COMMON /DATES/ IYR, IMO, IDAY, LYR, LMO, LDAY
```

```
      COMMON /JDATES/ JULDAT, JULREL, JBEGIN, JULEND, JRANGE, NXLPY
     $                ,JSTART, JSTOP,  JREND,   JYEAR
      COMMON /IO/     IREAD, IWRITE, IWDUG
C
      INTEGER IDBUG, CAL(12)
C
      DATA CAL /31,28,31,30,31,30,31,31,30,31,30,31/
C
C        SET DEBUG FLAG
C
         IDBUG = 0
C
C
C
C        DETERMINE INITIAL JULIAN DATES
C
      CALL JULIAN (IMO,IDAY,IYR,JBEGIN)
      CALL JULIAN (LMO,LDAY,LYR,JULEND)
      CALL JULIAN (  1,    1,IYR,JULREF)
C
C
      JULREL = JBEGIN - JULREF
      JRANGE = JULEND - JBEGIN + 1
      JULDAT = JBEGIN - 1
C
C        DETERMINE THE NEXT OCCURANCE OF 12/31/(LEAP YEAR)
C                (IE. THE 366TH DAY OF THE YEAR)
C
C
      LASTLP = IYR - MOD(IYR,4)
C
      CALL JULIAN (12,31,LASTLP,NXLPYR)
C
      IF (JULDAT .GE. NXLPYR) NXLPYR = NXLPYR + 1461
C
C
C        NOTE... 1461 = 365 + 365 + 365 + 366
C
C
C     THIS SECTION DEFINES VARIABLES NEEDED FOR MONTHLY
C     PARAMETER ESTIMATION
C
        JYEAR = IYR
        JSTART = JULREL + 1
        LD = CAL(LMO)
C
        CALL JULIAN ( LMO, LD, IYR, JDATE )
C
        JSTOP = JDATE - JULREF + 1
        JREND = JSTOP
C
          IF (JYEAR - MOD(JYEAR,4)) 65,70,65
   70     IF (IMO.EQ.2 .AND. IDAY.EQ.28) JREND = JSTOP + 1
   65   CONTINUE
C
```

```
C
      ENTRY DATE
C
C        THE NEXT SECTION IS USED EACH DAY TO UPDATE
C        THE JULIAN DATE COUNTERS IF ANNUAL PARAMETERS ARE USED.
C
      JULREL = JULREL + 1
      JULDAT = JULDAT + 1
C
C
C        CHECK FOR END OF YEAR
C
      IF (JULREL .LE. 365) GO TO 100
      IF (JULREL .GT. 366) GO TO 200
C
C        CHECK FOR LEAP YEAR
C
      IF (JULDAT.NE.NXLPYR) GO TO 200
C
C        YES, THERE ARE 366 DAYS THIS YEAR.
C        UPDATE NXLPYR TO NEXT LEAP YEAR.
C
      NXLPYR = NXLPYR + 1461
C
C
      IF( IDBUG .NE. 0 ) GO TO 900
C
C
   50 CONTINUE
C
C
      RETURN
C
C
C        RESET RELATIVE JULIAN DATE
C
  200 JULREL = 1
C
C
  100 CONTINUE
         IF( IDBUG .NE. 0 ) GO TO 900
  110 RETURN
C
C
      ENTRY DATEM
C
C     THIS SECTION IS USED EACH DAY TO UPDATE THE JULIAN DATE COUNTERS
C     IF MONTHLY PARAMETER ESTIMATION IS USED.
C
        JULREL = JULREL + 1
        JULDAT = JULDAT + 1
C
        IF (JULREL .LE. JREND) GO TO 400
C
C     UPDATE THE JULIAN COUNTERS
```

```
C
        JYEAR = JYEAR + 1
C
        CALL JULIAN (IMO,01,JYEAR,JBEGIN)
        CALL JULIAN (01,01,JYEAR,JULREF)
C
        JULREL = JBEGIN - JULREF + 1
        JSTART = JULREL
C
        CALL JULIAN (IMO,01,JYEAR,JULDAT)
C
        LD = CAL(LMO)
C
        CALL JULIAN ( LMO, LD, JYEAR, JDATE )
C
        JSTOP = JDATE - JULREF + 1
        JREND = JSTOP
C
        IF(JYEAR - MOD(JYEAR,4)) 400,410,400
  410   IF (LMO.EQ.2 .AND. LDAY.EQ.28) JREND = JREND + 1
  400   CONTINUE
C
        IF ( IDBUG .NE. 0 ) GO TO 900
C
C
        RETURN
C
C
C
  900 CONTINUE
C
C         DEBUG INFORMATION FOR JULIAN DATE CALCULATIONS
C
      WRITE (IWRITE,920) JULDAT,JULREL,JBEGIN,JULEND,JRANGE,NXLPYR
     $                   ,JSTART,JSTOP,JREND,JYEAR
  920 FORMAT (1H ,'JULDAT=',I10,3X,'JULREL=',I10,3X,'JBEGIN=',I10,3X,
     $        /2X,'JULEND=',I10,3X,'JRANGE=',I10,3X,'NXLPYR=',I10,
     $        /2X,'JSTART=',I10,3X,'JSTOP =',I10,3X,'JREND =',I10,
     $         3X,'JYEAR=',I10)
C
      RETURN
C
C
      END
C
C.....................................................................
C
      SUBROUTINE DATT(IDATE,IMO,IDAY,IYR)
C
C     CONVERT JULIAN DATE TO CALENDER DATE
C
      INTEGER CAL(12,2)
      DATA CAL/0,31,59,90,120,151,181,212,243,273,304,334,
     1  0,31,60,91,121,152,182,213,244,274,305,335 /
      I1=(IDATE-1)/1461
```

```
      I2=IDATE-(I1*1461)
C
C       1<=I2<=1461
C
      IF(I2.LE.365) GO TO 10
      IF(I2.LE.730) GO TO 20
      IF(I2.LE.1095) GO TO 30
      I3=3
      I4=I2-1095
      GO TO 40
   10 I3=0
      I4=I2
      I4=I2-365
      GO TO 40
   20 I3=1
      GO TO 40
   30 I3=2
      I4=I2-730
   40 IYR=1900+I3+(4*I1)
      INDX=1
      IF(I3.EQ.3)INDX=2
      DO 100 I=2,12
      IF(I4.LE.CAL(I,INDX)) GO TO 200
  100  CONTINUE
       IMO=12
      IDAY=I4-CAL(12,INDX)
       RETURN
  200 IMO=I-1
      IDAY=I4-CAL(I-1,INDX)
      RETURN
      END
C
C
C.......................................................................
C
      SUBROUTINE JULIAN(MO,DA,YR,ANS)
      INTEGER ANS,CAL(12),DA,YR
      DATA CAL /31,28,31,30,31,30,31,31,30,31,30,31/
C
C         COMPUTE JULIAN DATE FROM JAN. 1, 1973
C
      ANS=0
      I=YR-1900
      ANS=ANS+365*I
      CAL(2)=28
      IF(MOD(YR,4).EQ.0) CAL(2)=29
      J=MO-1
      IF(J.EQ.0) GO TO 20
      DO 10 I=1,J
      ANS=ANS+CAL(I)
   10 CONTINUE
   20 CONTINUE
      ANS=ANS+DA
      RETURN
      END
```

```
C
C.....................................................................
C
      FUNCTION DMS(A)
C
C     FUNCTION DMS CONVERTS ANGLES EXPRESSED IN
C     DEGREES, MINUTES AND SECONDS TO RADIANS
C
      DIMENSION A(3)
      REAL MINUTE
C
C
      DEGREE = A(1)
      MINUTE = A(2)
      SECOND = A(3)
C
      DMS = DEGREE*3.14159/180. + MINUTE*3.14159/180./60.
     1       + SECOND*3.14159/180./60./60.
C
      RETURN
      END
C
C.....................................................................
C
      FUNCTION TAU(ST)
C
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /IO/ IREAD, IWRITE, IWBUG
      COMMON /DEBUG/ NBUG,DEBUG
      DOUBLE PRECISION    ITAU,DEBUG(1)
      DATA ITAU /'TAU'/
C
C
C     THETAS = LONGITUDE OF STANDARD MERIDIAN (RADIANS)
C              75TH MERIDIAN FOR EASTERN STANDARD TIME
C              90TH MERIDIAN FOR CENTRAL STANDARD TIME
C              105TH MERIDIAN FOR MOUNTAIN STANDARD TIME
C              120TH MERIDIAN FOR PACIFIC STANDARD TIME
C     THETAL = LONGITUDE OF OBSERVERS MERIDIAN (RADIANS)
C     TAU    = LOCAL HOUR ANGLE
C     ST     = STANDARD TIME IN THE TIME ZONE OF THE
C              OBSERVER IN HOURS COUNTED FROM
C              MIDNIGHT (EG. 0.00 TO 24.00)
C     EP     = +1 FOR EAST LONGITUDE, -1 FOR WEST LONGITUDE
C     ET     = DIFFERENCE BETWEEN TRUE SOLAR TIME
C              AND MEAN SOLAR TIME (USUALLY NEGLECTED
C              FOR HEAT TRANSFER COMPUTATIONS)
C
C
C     FUNCTION SUBROUTINE TAU CONVERTS THE OBSERVERS
C     STANDARD TIME TO LOCAL HOUR ANGLE IN RADIANS
C
C     OBTAIN TIME DIFFERENCE BETWEEN STANDARD MERIDIAN AND
C     OBSERVERS MERIDIAN (HOURS)
C
```

```
      DTSL = EP*(THETAS - THETAL)* 12.0/3.14159
C
C     COMPUTE OBSERVERS HOUR ANGLE (RADIANS). E = +1 FOR
C     MORNING AND E = -1 FOR AFTERNOON (I.E. SOLARNOON)
C
      IF (ST.GT.12. + DTSL -ET) E = -1.0000
      IF (ST.LE.12. + DTSL -ET) E = +1.0000
C
      TAU = (ST + E*12. - DTSL + ET) * 3.14159/12.0
C
      IF (TAU.GT.6.283185) TAU = TAU - 6.283185
      IF (TAU.LT.0.0) TAU = TAU + 6.283185
C
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 100
      DO 200 I = 1,NBUG
C
      IF (DEBUG(I).NE.ITAU) GO TO 200
      WRITE (IWBUG,250)     ST,PHI,THETAS,THETAL,EP,ET,W,DTSL,TAU
  250 FORMAT (////1H ,'FUNCTION TAU',  2X,
     1 'ST =',F6.3,2X,'PHI =',F6.3,2X,'THETAS =',F6.3,2X,'THETAL ='
     2 F6.3,2X,'EP =',F6.3,2X,'ET =',F6.3,2X,'W =',F6.3,2X,
     3 'DTSL =',F6.3,2X,'TAU =',F6.3)
  200 CONTINUE
C
  100 CONTINUE
C
      RETURN
      END
C
C.......................................................................
C
      SUBROUTINE DECL (RJD,DELTA,SR,SS)
      INTEGER RJD
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /IO/ IREAD, IWRITE, IWBUG
      COMMON /DBUG/ NBUG,DEBUG
      DOUBLE PRECISION    IDECL,DEBUG(1)
      DATA IDECL/'DECL'/
C
C     DELTA  = DECLINATION OF THE SUN (RADIANS)
C     PHI    = OBSERVERS LATITUDE (RADIANS)
C     THETAS = LONGITUDE OF STANDARD MERIDIAN (RADIANS)
C              75TH MERIDIAN FOR EASTERN STANDARD TIME
C              90TH MERIDIAN FOR CENTRAL STANDARD TIME
C              105TH MERIDIAN FOR MOUNTAIN STANDARD TIME
C              120TH MERIDIAN FOR PACIFIC STANDARD TIME
C     THETAL = LONGITUDE OF OBSERVERS MERIDIAN (RADIANS)
C     RJD    = RELATIVE JULIAN DATE (I.E. WITH RESPECT TO JAN 1)
C     ST     = STANDARD TIME IN THE TIME ZONE OF THE OBSERVER
C              IN HOURS COUNTED FROM MIDNIGHT (E.G.0.00 TO 24.00)
C     EP     = +1 FOR EAST LONGITUDE, -1 FOR WEST LONGITUDE
C     ET     = DIFFERENCE BETWEEN TRUE SOLAR TIME AND
```

```
C                 MEAN SOLAR TIME (USUALLY NEGLECTED FOR
C                 HEAT TRANSFER COMPUTATIONS)
C
C COMPUTE TIME DIFFERENCE BETWEEN STANDARD MERIDIAN AND
C OBSERVERS MERIDIAN (HOURS)
C
      DTSL    = EP*(THETAS - THETAL)* 3.81972
C
C COMPUTE DECLINATION OF THE SUN (RADIANS)
C
      DELTA   = 0.4093*COS(0.0172*(172. - FLOAT(RJD)) )
C
C COMPUTE HOUR ANGLE AT SUNSET (RADIANS)
C
C
      TSS     = ACOS(-TAN(DELTA)*TAN(PHI))
C     COMPUTE STANDARD TIME OF SUNST (HOURS)
C
      SS      = TSS*3.81972 + 12. +DTSL -ET
C
C COMPUTE HOUR ANGLE OF SUNRISE (RADIANS)
C
      TSR     = 6.283185 - TSS
C
C     COMPUTE  STANDARD TIME OF SUNRISE (HOUR)
C
      SR      = TSR*3.81972 -12. + DTSL -ET
C
C
C     CONVERT SUNRISE IN STANDARD TIME TO LOCAL TIME
C
      SR = SR - DTSL
C
C     CONVERT SUNSET IN STANDARD TIME TO LOCAL TIME
C
      SS = SS - DTSL
C
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 300
      DO 100 I = 1,NBUG
C
      IF (DEBUG(I).NE.IDECL) GO TO 100
      WRITE (IWBUG,200)  RJD,DTSL,DELTA,TSS,SS,TSR,SR
  200 FORMAT (////,1H ,'SUBROUTINE DECL ','****',' RJD =',
     1 I5,'  DTSL =',F6.3,'  DELTA =',F6.3,'  TSS =',F6.3,
     2 '  SS =',F6.3,2X,'TSR =',F6.3,2X,'SR = ',F6.3)
  100 CONTINUE
C
  300 CONTINUE
C
      RETURN
      END
C
```

```
C.......................................................................
C
      SUBROUTINE RAINST (TB,TR,D,JSINCE,STORM,JHREOS,JHNEXT)
C
C
C  ROUTINE TO INITIALIZE THE RAINFALL MODEL. THIS ROUTINE INSURES
C  THAT THE BEGINNING OF THE MONTH OCCURS RANDOMLY DURING EITHER
C  AN INTRA- OR AN INTER-STORM PERIOD ACCORDING TO THE APPROPRIATE
C  PROBABILITY DISTRIBUTION.
C
       COMMON /RAINS/  TBMEAN, TRMEAN, DMEAN
      COMMON /RAINI/  ITR,ITB
       DATA   ON/'ON'/, OFF/'OFF'/
C
C
       TSUM = 0.0
       DEBUG = OFF
C
C  GENERATE THE TIME SINCE THE LAST STORM.
C
       CALL EXPO (TBMEAN,TSINCE)
C
C  NOW BEGIN TO GENERATE A SEQUENCE OF STORMS THAT WILL BRING US UP
C  TO THE BEGINNING OF THE MONTH.
C
  100  CALL EXPO (TBMEAN,TB)
       TSUM = TSUM + TB
C
C  ARE WE UP TO THE STARTING POINT YET?
C
       IF (TSUM .GE. TSINCE) GO TO 200
C
C  IF NOT, GENERATE A STORM DURATION.
C
       CALL EXPO (TRMEAN,TR)
       TSUM = TSUM + TR
C
C  ARE WE UP TO THE STARTING POINT YET?  IF NOT, GO BACK AND
C  GENERATE THE NEXT INTERSTORM PERIOD.
C
       IF (TSUM .LT. TSINCE) GO TO 100
C
C  IN THIS CASE, THE MONTH BEGINS DURING A STORM.  DETERMINE TIME
C  TILL END OF STORM (TTEOS) AND TURN STORM FLAG ON.
C
       TTEOS = TSUM - TSINCE
       STORM = ON
       JSINCE = 0
       TB = 0.0
C
C  COMPUTE THE STORM DEPTH GIVEN STORM DURATION.
C
       ALPHA = TR/TRMEAN
       BETA  = DMEAN
       CALL GAMMAD (ALPHA,BETA,D)
```

```
C
C  ADJUST STORM DEPTH TO REFLECT ONLY THE PORTION DURING THE
C  CURRENT MONTH.
C
      D = D*(TTEOS/TR)
      TR = TTEOS
C
C  CONVERT TR TO NEAREST INTEGER VALUE
C
      CALL ROUND (TR,ITR)
      IF (ITR .EQ. 0) ITR = 1
      JHREOS = ITR
C
C
      RETURN
C
C
  200 CONTINUE
C
C  IN THIS CASE, THE MONTH BEGINS DURING AN INTER-STORM PERIOD.
C  DETERMINE TIME TILL NEXT STORM.  TURN STORM FLAG OFF.
C
      TTNEXT = TSUM - TSINCE
      STORM  = OFF
      TR     = 0.0
      D      = 0.0
C
C
C  CLOUD COVER MODEL WILL ALSO NEED THE TIME SINCE THE LATEST STORM
C  ENDED.
C
      TSINCE = TB - TTNEXT
C
C  CONVERT TTNEXT TO NEAREST INTEGER
C
      CALL ROUND (TTNEXT,ITB)
      IF (ITB .EQ. 0) ITB = 1
      JHNEXT = ITB
      CALL ROUND(TSINCE,JSINCE)
C
C
C
C
      RETURN
      END
C
C...........................................................................
C
      SUBROUTINE ROUND (X,IX)
C
C  ROUND IS A ROUTINE THAT CONVERTS A REAL VALUE .X, TO THE NEAREST
C  INTEGER VALUE. IN OTHER WORDS, IX IS ROUNDED UP WHEN NECESSARY.
C
      IX = INT(X)
      RX = AINT(X)
```

```
C
C  CHECK IF X IS NEGATIVE OR POSITIVE.
C
      IF (X) 100,200,300
  100 IF (ABS(X-RX) .GT. 0.50) IX = IX - 1
  200 RETURN
  300 IF (ABS(X-RX) .GT. 0.50) IX = IX + 1
C
C
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE PCPN (TB,TR,D,STORM,JHOUR,JHREOS,JHNEXT,JSINCE,RAIN)
C
C  PCPN CHECKS TO SEE IF WE ARE CURRENTLY IN A STORM OR BETWEEN
C  STORMS AND COMPUTES THE HOURLY RAINFALL TOTAL ACCORDINGLY.
C  WHEN NECESSARY, PCPN SELECTS NEW TIMES BETWEEN STORMS, STORM
C  DURATIONS, AND STORM DEPTHS.  THE HOURLY COUNTERS ARE ALSO
C  UPDATED FOR TIME TILL NEXT STORM AND TIME TILL END OF CURRENT
C  STORM.
C
      COMMON /RAINI/  ITR,ITB
      COMMON /RAINS/  TBMEAN,TRMEAN,DMEAN
C
      DATA    ON/'ON'/, OFF/'OFF'/
C        .
C
C  CHECK IF STORM FLAG IS ON OR OFF. IF STORM FLAG IS ON, GO TO THE
C  STORM SECTION.
C
      IF ( STORM .EQ. ON ) GO TO 200
C
C  STORM FLAG IS OFF.  NOW CHECK IF WE HAVE ENDED THE LATEST INTER-
C  STORM PERIOD.
C
      IF ( JHOUR .GT. JHNEXT ) GO TO 100
C
C  STILL IN BETWEEN STORMS.  THEREFORE SET RAIN = 0.0 AND RETURN.
C  ALSO INCREMENT THE COUNTER FOR TIME SINCE LAST STORM.
       JSINCE = JSINCE + 1
C
      D = 0.0
      RAIN = 0.0
      RETURN
C
C
C
  100 CONTINUE
C
C  GENERATE A NEW STORM. FIRST, TURN STORM FLAG ON. SECOND, SELECT A
C  STORM DURATION. THEN SELECT A STORM DEPTH
C
      STORM = ON
```

```
      CALL EXPO (TRMEAN,TR)
      ALPHA = TR/TRMEAN
      BETA  = DMEAN
      CALL GAMMAD (ALPHA,BETA,D)
C
C  CONVERT STORM DURATION TO THE NEAREST INTEGER VALUE.
C
      CALL ROUND (TR,ITR)
C
C  MINIMUM STORM DURATION IS ONE HOUR.
C
      IF (ITR .EQ. 0) ITR = 1
C
C  UPDATE THE TIME TILL END OF STORM.
C
      JHREOS = JHOUR + ITR - 1
      JSINCE = 0
C
C  COMPUTE THE HOURLY RAINFALL DEPTH
C
      RAIN = D/FLOAT(ITR)
C
      RETURN
C
C
C
  200 CONTINUE
C
C
C  STORM FLAG IS ON. NOW CHECK TO SEE IF THE STORM ENDED.
C
      IF (JHOUR .GT. JHREOS) GO TO 300
C
C  THE STORM IS STILL GOING ON.  THEREFORE, COMPUTE RAIN AND RETURN.
C
      RAIN = D/FLOAT(ITR)
      JSINCE = 0
C
      RETURN
C
C
  300 CONTINUE
C
      STORM = OFF
C
C
C  STORM ENDED. SELECT THE NEXT TIME BETWEEN STORMS.
C
      CALL EXPO (TBMEAN,TB)
C
C  CONVERT TIME BETWEEN STORMS TO NEAREST INTEGER VALUE.
C
      CALL ROUND (TB,ITB)
C
C  MINIMUM TIME BETWEEN STORMS IS ONE HOUR.
```

```
C
      IF (ITB .EQ. 0) ITB = 1
C
C  UPDATE THE TIME TILL NEXT STORM.
C
      JHNEXT = JHOUR + ITB - 1
      JSINCE = 1
C
      RAIN = 0.0
C
      RETURN
      END
C
C.............................................................
C
      SUBROUTINE EXPO (EM,T)
C
      COMMON /SEED/ ISEED
C
C     SUBROUTINE TO GENERATE EXPONENTIALLY DISTRIBUTED RANDOM NUMBERS
C     EM = MEAN OF THE DISTRIBUTION
C     T  = RANDOM VARIABLE
C
C     GENERATE U(0,1)
C
      IX = ISEED
CXXXXXCALL RANDU (IX,ISEED,R)
      CALL RAND1 (IX,ISEED,R)
C
C     TAKE THE INVERSE OF THE EXPONENTIAL PDF
C
      T = -EM*ALOG(R)
      RETURN
      END
C
C.............................................................
C
      SUBROUTINE GAMMAD (ALPHA,BETA,X)
C
      COMMON /WARN/ IWARN
      COMMON /SEED/ ISEED
      COMMON /IO/     IN,IS,IB
C
      U = 1.0
      X = 0.0
      K = IFIX(ALPHA)
      GAM = ALPHA - FLOAT(K)
C
C WRITE (5,900) U,X,K,GAM,ALPHA,BETA
  900   FORMAT (1H0,'U=',E12.5,2X, 'X=',E12.5,2X, 'K=',I5,2X,
     1               'GAM=',E12.5,2X, 'ALPHA=',E12.5,2X, 'BETA=',
     2                E12.5)
C
C
C
```

```
      IF (K.EQ. 0) GO TO 100
C
      DO 50 I = 1,K
      IX = ISEED
CXXXX CALL RANDU (IX,ISEED,R)
      CALL RAND1 (IX,ISEED,R)
      U = R*U
C
C WRITE (5,920) I, R, U
  920     FORMAT (1H0, 'TRACE 1 ',' I= ',I5,2X, 'R= ',E12.5,2X,
     1                 'U= ',E12.5)
C
   50 CONTINUE
C
      X = -ALOG(U)
C
      IF (GAM.GE. 0.000001 ) GO TO 100
C
      X = BETA*X
C
C WRITE (5,930) X
  930     FORMAT (1H0,'TRACE 2 ',2X, 'X= ',E12.5)
C
      RETURN
C
  100 CONTINUE
      IX = ISEED
CXXXX CALL RANDU (IX,ISEED,R)
      CALL RAND1 (IX,ISEED,R)
      Z = -ALOG (R)
C
C WRITE (5,940) R,Z
  940     FORMAT (1H0,' TRACE 3 ',' R= ',E12.5,'  Z= ',E12.5)
C
C
C
      DO 200 J = 1,100
      IX = ISEED
CXXXX CALL RANDU (IX,ISEED,U1)
      CALL RAND1 (IX,ISEED,U1)
      IX = ISEED
CXXXX CALL RANDU (IX.ISEED,U2)
      CALL RAND1 (IX,ISEED,U2)
C
C
C     COMPUTE THE VALUES OF EM AND EN
C
C     IF EM AND EN ARE COMPUTED DIRECTLY AS:
C
C        EM = U1**(1.0/GAM)
C        EN = U2**(1.0/(1.0-GAM))
C
C     A MACHINE UNDERFLOW OR OVERFLOW CAN EASILY OCCUR.  THESE
C     CONDITIONS CAN BE ANTICIPATED BY FIRST CALCULATING THE
C     LOG (BASE 10) OF EM AND EN.  THE VALID RANGE OF LOG(EM)
```

```
C      AND LOG(EN) IS MACHINE DEPENDENT BUT HAS NEVERTHELESS
C      BEEN SET TO BETWEEN -37.0 AND +37.0 IN THIS PROGRAM.
C      IF A VALUE OF HAS BEEN FOUND BELOW THIS RANGE, A DEFAULT
C      OF LOG(EM OR EN) = -37.0 IS USED. IF A VALUE OHAS BEEN
C      FOUND ABOVE THIS RANGE, THEN LOG (EM OR EN) = +37.0.
C      EM AND EN ARE THEN FOUND BY TAKING THE APPROPRIATE ANTILOGS.
C
C
       EML10 = (1.0/GAM)*ALOG10(U1)
C
          IF ( EML10 .GE. -37.0 .AND.
     1          EML10 .LE. +37.0        ) GO TO 110
C
             IWARN = IWARN + 1
             IF ( EML10 .GT. +37.0 ) EML10 = +37.0
             IF ( EML10 .LT. -37.0 ) EML10 = -37.0
C
  110 EM = 10.0**EML10
C
C
       ENL10 = (1.0/(1.0 - GAM))*ALOG10(U2)
C
          IF ( ENL10 .GE. -37.0  .AND.
     1          ENL10 .LE. +37.0         ) GO TO 120
C
             IWARN = IWARN + 1
             IF ( ENL10 .GT. +37.0 ) ENL10 = +37.0
             IF ( ENL10 .LT. -37.0 ) ENL10 = -37.0
C
  120 EN = 10.0**ENL10
C
C
C
C WRITE (5,950) J,U1,U2,EM,EN,EML10,ENL10
  950     FORMAT (1H0,'TRACE 4 ',' J= ',I5,' U1= ',E12.5,
     1                    ' U2= ',E12.5, ' EM= ',E12.5,' EN= ',E12.5,
     2                    ' EML10=',E12.5,' ENL10=',E12.5)
C
          IF (EM +EN .LE. 1.0) GO TO 300
  200 CONTINUE
C
C
        WRITE (IS,500)
  500 FORMAT('    END OF DO LOOP IN BETA SUBSECTION OF GAMMAD ')
      STOP
C
C
  300 Y = EM/(EM + EN)
C
      X = BETA*(X + Y*Z)
C
      RETURN
      END
C
C..............................................................
```

```
C
      SUBROUTINE RANDU (IX,IY,YFL)
C
C     GENERATES A UNIFORM DISTRIBUTION
C
      IY = IX*65539
      IF (IY) 10,20,20
   10 IY = IY + 2147483647 + 1
   20 YFL = IY
      YFL = YFL*.4656613E-9
C
      RETURN
C
C
      ENTRY RAND1 (IX,IY,YFL)
C
         YFL = RAND(0)
C
C
      RETURN
      END
C
C.......................................................................
C
      SUBROUTINE STAT(X,SUM,SUMSQ,SUM3,XBAR,XVAR,XSKEW,N)
C
C  ROUTINE TO COMPUTE THE FIRST THREE MOMENTS OF INTEREST
C  --- MEAN --- VARIANCE --- SKEW COEFFICIENT -----------
C       XBAR        XVAR               XSKEW
C
           TRACE = 'OFF'
           IF (TRACE .EQ. 'ON') WRITE (5,901)
  901      FORMAT (' STAT1')
C
      SUM = X + SUM
      SUMSQ = X*X + SUMSQ
      SUM3 = X**3.0 + SUM3
C
           IF (TRACE .EQ. 'ON') WRITE (5,902)
  902      FORMAT (' STAT2')
C
C
C  UPDATE THE MEAN AND VARIANCE COMPUTATION
C
      XBAR = SUM/ N
      XVAR = SUMSQ/N - XBAR*XBAR
      IF (XSKEW .LT. -990.0) RETURN
      XM3 = SUM3/N - 3.0*XBAR*SUMSQ/N + 2.0*XBAR**3.0
C
           IF (TRACE .EQ. 'ON') WRITE (5,903)
  903      FORMAT (' STAT3')
C
      IF (N .LE. 2) RETURN
C
C  COMPUTE SKEW COEFFICIENT
```

```
C
      IF (XVAR .GT. 0.00001) GO TO 100
      XSKEW = 0.0
      GO TO 999
  100 CONTINUE
C
      FACTOR = FLOAT(N*N)/FLOAT((N-1)*(N-2))
      XSKEW = FACTOR*XM3/(XVAR*SQRT(XVAR))
C
         IF (TRACE .EQ. 'ON') WRITE (5,904)
  904    FORMAT (' STAT4')
         IF (TRACE .EQ. 'ON') WRITE (5,905) N, XSKEW, FACTOR, XVAR
  905    FORMAT (I10,3(E12.5,2X))
  999 RETURN
      END
C
C.......................................................................
C
      SUBROUTINE HGRAM (H,IA,X,DT,BASE)
C
C  SUBROUTINE TO UPDATE THE FREQUENCY HISTOGRAMS
C
      COMMON /IO/      IN,IS,IB
      INTEGER H
      DIMENSION H(IA)
C
      DO 100 I = 1,IA
      IF(X.GT.BASE+I*DT) GO TO 100
      H(I) = H(I) + 1
      RETURN
  100 CONTINUE
C
      H(I) = H(I) + 1
      AMAX = BASE + IA*DT
C
      WRITE (IS,900)AMAX,X
  900 FORMAT (1H ,'A VALUE GREATER THAN ',E12.5,'  WAS FOUND. X = ',
     1 E12.5)
C
      RETURN
      END
C
C.......................................................................
C
      SUBROUTINE PRINTH (H,NMAX,DT,BASE,TITLE,NDATA)
C
C  PRINT OUT NORMALIZED HISTOGRAMS OF GENERATED DATA
C
C
C
      COMMON /IO/      IN,IS,IB
C
C
      INTEGER H
      DIMENSION H(NMAX)
```

```
      DIMENSION TITLE(1)
      DIMENSION TA(10)
C
C   NORMALIZE THE HISTOGRAM ELEMENTS
C
C       GO TO 60
      DO 50 I = 1,NMAX
      N = H(I)
       X = 100.0*(FLOAT(N)/FLOAT(NDATA))
      CALL ROUND (X,IH)
C     IF(NMAX.EQ.11)WRITE(5,501)I,NMAX,N,X,IH
  501 FORMAT(I5,1X,I5,1X,I15,1X,E12.5,1X,I15)
      H(I) = IH
   50 CONTINUE
   60 CONTINUE
C
      WRITE (IS,910)
  910 FORMAT (1H1,15(5H     )/1H+,15(5H     ))
      WRITE (IS,900) (TITLE(I),I=1,5)
  900 FORMAT (1H ,14X,'HISTOGRAM OF ',5A5,' (PERCENT)'/)
      DO 100 J = 1,NMAX,10
      IMAX = 10
      IF( J+10 .GT. NMAX ) IMAX = NMAX - J + 1
C
C
       GO TO 199
  196 DO 198 II=1,IMAX
      IZ = II + J - 1
      WRITE (5,197)IZ,II,J,IMAX,NMAX,H(IZ)
  197 FORMAT(5I5,I20)
  198 CONTINUE
  199 CONTINUE
      WRITE (IS,200)(H(I+J-1),I=1,IMAX)
  200 FORMAT (1H ,10(2X,I5))
C
C
      WRITE (IS,300)
  300 FORMAT (1H ,10(7H------- ))
C
C
      DO 350 K = 1,10
  350 TA(K) = (K-1+J)*DT + BASE
C
C
      WRITE (IS,400)(TA(K),K=1,10)
  400 FORMAT(1H ,10(1X,F6.2)//)
C
C
  100 CONTINUE
C
C
      RETURN
      END
C
C.............................................................
```

```
C
C
      SUBROUTINE SOLRAD (RJD,ST1,ST2,T,TB,NMAX,CCA,CCB,PDF,N,COORD,
     1          SWR,CLD,BETA,GAM,CCBAR,CCSD,RHO,SEASON)
C
C     SUBROUTINE SOLRAD COMPUTES INCIDENT SOLAR
C     RADIATION ON THE GROUND OR ON THE TOP OF A
C     VEGETAL CANOPY DURING A SPECIFIED INTERVAL OF TIME
C
C     ST1    = BEGINNING OF INTERVAL - STANDARD TIME
C     ST2    = END OF INTERVAL - STANDARD TIME
C     CSKY   = CLEAR SKY RADIATION - LANGLY
C     CLD    = CLOUD COVER (0.0 - CLD   - 1.0)
C     SWR    = TOTAL INCIDENT SOLAR RADIATION - LANGLY
C     SR     = SUNRISE
C     SS     = SUNSET
C     T1     = BEGINNING OF INTERVAL OF INTEGRATION - LOCAL HOUR ANGLE
C     T2     = END OF INTERVAL OF INTEGRATION - LOCAL HOUR ANGLE
C     RJD    = RELATIVE JULIAN DATE
C     SIALPH = SIN(ALPHA)
C     PDF    = PROBABILITY DENSITY FUNCTION (DISCRETE)
C              FOR NOISE TERM IN CLOUD COVER MODEL
C     RADTYP = INDICATES IF USER WANTS CLRSKY CALCULATIONS ONLY
C     COORD  = COORDINATES OF THE INTERVALS OF PDF
C
      DOUBLE PRECISION    CLEAR,RADTYP
      DOUBLE PRECISION    ISOLRD,DEBUG(1)
      DIMENSION PDF(1), COORD(1)
      DIMENSION RHO(1),CCBAR(1),CCSD(1),BETA(1),GAM(1)
      INTEGER SEASON(1)
      INTEGER RJD
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /RTYPE/ RADTYP
      COMMON /SEED/ ISEED
      COMMON /DBUG/ NBUG,DEBUG
      COMMON /IO/    IN,IS,IB
      DATA ISOLRD /'SOLRAD'/
      DATA CLEAR /'CLEARSKY'/
C       .
C
C     COMPUTE DECLINATION, SUNRISE AND SUNSET
C
      CALL DECL (RJD,DELTA,SR,SS)
C
C
C     SCREENING TO DETERMINE THE PROPER INTERVAL OF INTEGRATION
C
      IF (ST2.LE.ST1) GO TO 100
      IF (ST1.LE.SR.AND.ST2.LE.SR) GO TO 120
      IF (ST1.LE.SR.AND.ST2.GE.SR.AND.ST2.LE.SS) GO TO 130
      IF (ST1.LE.SR.AND.ST2.GE.SS) GO TO 140
      IF (ST1.GE.SR.AND.ST2.LE.SS) GO TO 150
      IF (ST1.LE.SS.AND.ST2.GE.SS) GO TO 160
      IF (ST1.GE.SS.AND.ST2.GE.SS) GO TO 120
C
```

```
C      ST2 IS IN THE AM WHILE ST1 IS STILL IN PM
C
  100 CONTINUE
      IF (ST2.GT.SR) GO TO 130
C
C      NO SHORTWAVE RADIATION IN THIS INTERVAL
C
  120 CSKY = 0.0
      T1 = TAU(ST1)
      T2 = TAU(ST2)
      SIALPH  = 999.
      GO TO 800
C
C      PART OF INTERVAL COMES AFTER SUNRISE. SET BEGINNING
C      OF INTERVAL EQUAL TO THE LOCAL HOUR ANGLE OF SUNRISE.
C      THEN CONVERT ENDING TIME TO LOCAL HOUR ANGLE.
C
  130 T1 = TAU(SR)
      T2 = TAU(ST2)
      GO TO 500
C
C      INTEGRATION INTERVAL INCLUDES ENTIRE INTERVAL FROM SUNRISE
C      TO SUNSET
C
  140 T1 = TAU(SR)
      T2 = TAU(SS)
      GO TO 500
C
C      INTEGRATION INTERVAL IS ENTIRELY WITHIN SUNSHINE PERIOD
C
  150 T1 = TAU(ST1)
      T2 = TAU(ST2)
      GO TO 500
C
C      ENDING TIME OCCURS AFTER SUNSET
C
  160 T1 = TAU(ST1)
      T2 = TAU(SS)
C
C      COMPUTE CLEAR SKY SOLAR RADIATION FOR THE
C      INTERVAL T1 TO T2
C
  500 CONTINUE
      CALL CLRSKY (RJD,T1,T2,NMAX,CSKY,SIALPH,DELTA)
C
C      DETERMINE CLOUD COVER
C
  800 CONTINUE
      IF (RADTYP.EQ.CLEAR) GO TO  900
C     GO TO 801
      CALL COVER (RJD,CCA,CCB,PDF,N,COORD,SEASON,TB,T,BETA,GAM,CCBAR,
     1          CCSD,RHO,CLD)
  801 CONTINUE
C
C      COMPUTE CLOUDY SKY SOLAR RADIATION
```

```
C
      SWR = CSKY*(1.0 - 0.65*CLD*CLD)
      GO TO 950
C
  900 SWR = CSKY
C
  950 CONTINUE
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 1100
      DO 1000 I = 1,NBUG
C
      IF (DEBUG(I).NE.ISOLRD) GO TO 1000
      WRITE (IB,1050)RJD,ST1,ST2,SR,SS,T1,T2,CSKY,SIALPH,CLD
 1050 FORMAT (///,1H ,'SUBROUTINE SOLRAD',2X,'RJD ='
     1 I4,2X,'ST1 =',F7.3,2X,'ST2 =',F7.3,2X,'SR ='
     2 F7.3,2X,'SS =',F7.3,2X,'T1 =',F7.3,2X,'T2 =',
     3 F7.3/T20,'CSKY =',F12.2,2X,'SIALPH =',F8.3,
     4 2X,'CLD =',F7.3)
 1000 CONTINUE
C
 1100 CONTINUE
      RETURN
      END
C
C.....................................................................
C
C
      SUBROUTINE CLRSKY (RJD,T1,T2,NMAX,CSKY,SIALPH,DELTA)
C
C     SUBROUTINE TO NUMERICALLY INTEGRATE THE
C     EQUATION FOR CLEAR SKY RADIATION. SIMPSONS
C     RULE IS USED.
C
C     DELTA  = DECLINATION OF THE SUN (RADIANS)
C     PHI    = OBSERVERS LATITUDE (RADIANS)
C     EN     = TURBIDITY FACTOR
C            = 2.0 FOR CLEAR MOUNTAIN AIR
C            = 4-5 FOR SMOGGY URBAN AREAS
C     W      = SOLAR CONSTANT = 120. LANGLY/HR
C              W IS READ IN AS A VARIABLE TO ALLOW THE USER  TO CHOOSE
C              WHICH VALUE OF W IS APPROPRIATE.
C     RJD    = RELATIVE JULIAN DATE
C     T1     = HOUR ANGLE AT BEGINNING OF INTERVAL
C     T2     = HOUR ANGLE AT END OF INTERVAL
C     NMAX   = NUMBER OF SUBINTERVALS = 2,4,6...
C     CSKY   = FINAL VALUE OF F IS CLEAR SKY RADIATION
C     SIALPH = SIN (ALPHA), WHERE ALPHA IS THE ANGLE
C              OF RADIATION WITH THE HORIZONTAL (RADIANS)
C     ALPHA =  ANGLE OF RADIATION (RADIANS)
C
C     REFERENCE FOR SIMPSONS RULE
C             TI PROGRAMMABLE 58/59 MASTER LIBRARY
C             TEXAS INSTRUMENTS INCORPORATED, 1977 P29-31
```

```
C
      COMMON /DBUG/ NBUG,DEBUG
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,.
      COMMON /ATMOS/ EN
      COMMON /IO/     IN,IS,IB
      INTEGER RJD
      DOUBLE PRECISION     ICSKY,DEBUG(1)
      DATA ICSKY /'CLRSKY'/
C
C
C     IS DEBUG REQUESTED FOR SUBROUTINE CLRSKY?
C
      IBUG = 0
      IF (NBUG.EQ.0) GO TO 910
      DO 900 I = 1,NBUG
C
      IF (DEBUG(I).NE.ICSKY) GO TO 900
      IBUG = 1
      GO TO 910
  900 CONTINUE
C
  910 CONTINUE
C
      IF (IBUG.EQ.0) GO TO 10
      WRITE (IB,930) RJD,T1,T2,NMAX
  930 FORMAT (////,1H0,'SUBROUTINE CLRSKY',2X,' JD =',
     1 I5,2X,'T1 =',F6.3,'T2 =',F6.3,2X,'NMAX =',I5)
C
   10 CONTINUE
C
C
C     DO LOOP PERFORMS INTEGRATION BY SIMPSON'S RULE
C
      X = 0.0
      F=0.000
      IMAX = NMAX + 1
C
C
      D = (T2 - T1)/NMAX
      IF (D.GE.0.0) GO TO  70
      D = ( 6.28318 - T1 + T2 )/NMAX
   70 CONTINUE
C
C
      DO 100 NN = 1,IMAX
C
      N=NN-1
C
C     COMPUTE CURRENT HOUR ANGLE
C
      T = T1 + N*D
C
C     COMPUTE SIN(ALPHA)
C
      SIALPH = SIN(DELTA)*SIN(PHI) + COS(DELTA)*COS(PHI)*COS(T)
```

```
C
C      CHECK TO PREVENT DIVISION BY ZERO OR USING ZERO
C      AS THE ARGUMENT OF A LOG FUNCTION
C
C
C
C      CONSIDER THE TERM
C
C
C         Y = ( 0.128 - 0.054*ALOG10(1./SIALPH))
C      WHEN ALPHA APPROACHES ZERO, THE DECAY FUNCTION STARTS TO GROW.
C      THIS OCCURS DUE TO POLES THAT EXIST AT THE ENDS OF THE INTERVAL
C      OF INTEGRATION. AN APPROXIMATION TO THE DECAY FUNCTION WAS
C      MADE THAT CONSISTED OF A STRAIGHT LINE EXTRAPOLATION OF THE
C      DECAY FUNCTION FROM ALPHA = 0.016 TO ZERO.
C
C
       ALPHA = ASIN(SIALPH)
       IF (ALPHA .GT. 0.016) GO TO 40
       IF ( ALPHA .LT. 0.0 ) ALPHA = 0.0
       IF ( SIALPH .LT. 0.0 ) SIALPH = 0.0
C
       X = 1.293454*ALPHA*SIALPH
C
       GO TO 45
C
   40  CONTINUE
C
       X=(EXP(-EN*(0.128 - 0.054*ALOG10(1./SIALPH))/SIALPH))*SIALPH
C
   45  CONTINUE
C
       IF (MOD(N,2).NE.0) GO TO 200
       M=2
       IF (N.EQ.0) M=1
       IF (N.EQ.NMAX) M=1
       F=F + M*X
       GO TO 50
C
  200  F=F + 4*X
C
C      DEBUG OPTION
C
   50  IF (IBUG.EQ.0) GO TO 100
       WRITE (IB,920) ,N,T,SIALPH,X,F
  920  FORMAT (1H ,T25,'N =',I4,2X,'T =',F6.3,2X,'SIALPH =', F6.2,
     1 2X,'X =',E12.3,2X,'F =',E12.3)
C
  100  CONTINUE
C
       F = F*D/3.0
C
C      COMPUTE CORRECTION FACTOR FOR ELLIPTICAL ORBIT
C
C
```

```
      R =  1.000 + 0.017*COS(6.2832*FLOAT(186 - RJD)/365.)
C
      CSKY = (12.0*60.0/3.1416)*F*W/(R*R)
C
C     DEBUG OPTION
C
      IF (IBUG.EQ.0) GO TO 300
      WRITE (IB,940) F,R,W,CSKY
  940 FORMAT (1H ,T25,'F =',E12.3,2X,'R =',F6.3,
     1 'W =',F8.3,2X,'CSKY =',E12.3)
C
  300 CONTINUE
      RETURN
      END
C
C.......................................................................
C
C
      SUBROUTINE COVER (RJD,A,B,PDF,N,COORD,SEASON,TB,T,BETA,GAM,CCBAR,
     1CCSD,RHO,CLD)
C
      INTEGER  RJD,SEASON(1)
      DIMENSION PDF(1),COORD(1)
      DIMENSION RHO(1),CCBAR(1),CCSD(1),BETA(1),GAM(1)
      COMMON /CLDCOV/ C1
      COMMON /LEAP/ LCHECK
      COMMON /SEAS/ NSEAS
      COMMON /DBUG/ NBUG,DEBUG
      COMMON /IO/    IN,IS,IB
      COMMON /STORMS/ STORM
      DOUBLE PRECISION    ICOVER,DEBUG(1)
      DATA ICOVER /'COVER'/
      DATA ON/'ON'/, OFF/'OFF'/
C
C     SEASON = ARRAY CONTAINING RELATIVE JULIAN DATES OF THE FIRST DAY
C              OF EACH SEASON
C     PDF    = DISCRETE PROBABILITY DENSITY FUNCTION OF CLOUD COVER
C     COORD  = COORDINATES OF PDF (I.E. INTERVALS)
C     N      = NUMBER OF INTERVALS IN PDF. DIMENSION OF PDF
C              AND COORD IS N* (NUMBER OF SEASONS OF CLOUD
C              COVER PARAMETERS)
C     ISEAS  = CURRENT SEASON
C     TB     = TIME BETWEEN STORMS (HOURS)
C     T      = TIME SINCE LAST STORM (HOURS)
C     CCBAR  = MEAN CLOUD COVER
C     CCSD   = STANDARD DEVIATION OF CLOUD COVER
C     RHO    = LAG-1 AUTOCORRELATION COEFFICIENT
C     BETA   = TRANSITION DECAY PARAMETER
C     GAM    = TRANSITION DECAY PARAMETER
C     NSEAS  = NUMBER OF SEASONS PER YEAR
C     ARV    = RANDOM VARIATE FOR THE NOISE TERM IN THE CLOUD COVER
C              MODEL
C     C1     = PREVIOUS VALUE OF THE AR(1) PROCESS
C     C2     = CURRENT VALUE OF THE AR(1) PROCESS
C     F      = VALUE OF THE MODULATION FUNCTION
```

```
C      CLD      = CLOUD COVER
C
C      DETERMINE THE CURRENT SEASON
C
       IF (NSEAS.GT.1) GO TO 50
          ISEAS = 1
          GO TO 150
C
   50  CONTINUE
       IF (RJD .LT. SEASON(NSEAS+LCHECK*NSEAS)) GO TO 60
          ISEAS = NSEAS
          GO TO 150
   60  CONTINUE
C
C
       DO 100 I=1,NSEAS
       IF (RJD.GE.SEASON(I + LCHECK*NSEAS)) GO TO 100
       ISEAS = I-1
       GO TO 150
  100  CONTINUE
       WRITE (IS,160)
  160  FORMAT (1H1,///,'SEASON SELECTION FAILED IN SUBROUTINE COVER')
       STOP
  150  CONTINUE
C
C
C
C      COMPUTE STOCHASTIC COMPONENT
C
       ARV = ARVA(PDF,N,A,B,COORD,ISEAS)
C
       C2 = RHO(ISEAS)*C1 + SQRT(1.-RHO(ISEAS)*RHO(ISEAS))*
      1 (ARV - CCBAR(ISEAS))
C     2 + CCBAR(ISEAS)*(1. - RHO(ISEAS))
C
C   CHECK TO SEE IF A STORM IS GOING ON.  IF NO STORM, COMPUTE THE
C   MODULATION FUNCTION.  IF STORM IS ON, SET CLD = 1.0 AND BY-PASS
C   THE MODULATION FUNCTION.
C
       IF (STORM .EQ. OFF) GO TO 200
       CLD = 1.0
       GO TO 300
  200  CONTINUE
C
C      COMPUTE MODULATION FUNCTION
C
       BEXP = BETA(ISEAS)*T
       GEXP = GAM(ISEAS)*(TB-T)
C
C   CHECK TO SEE IF BEXP OR GEXP WILL CAUSE A MACHINE
C   UNDERFLOW WHEN USED AS THE ARGUMENT IN THE EXP FUNCTION.
C
       IF (BEXP .GT. 37.0*ALOG(10.)) BEXP =37.0*ALOG(10.0)
       IF (GEXP .GT. 37.0*ALOG(10.)) GEXP =37.0*ALOG(10.0)
C
```

```
      P = (1.0 -EXP(-BEXP))*(1.0 -EXP(-GEXP))
C
C     COMPUTE AVERAGE CLOUD COVER FOR INTERVAL
C
C
      CLD = CCBAR(ISEAS) + (1.0 - CCBAR(ISEAS))*(1.0 -P) + C2*P
      IF (CLD.GT.1.0) CLD = 1.00
      IF (CLD.LT.0.0) CLD = 0.00
  300 CONTINUE
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 910
      DO 900 I = 1,NBUG
      IF (DEBUG(I).NE.ICOVER) GO TO 900
      WRITE (IB,920) RJD,ISEAS,C1,C2,ARV ,P,CLD
  920 FORMAT (///1H ,'SUBROUTINE COVER',2X,'RJD =',I5,2X,
     1 'ISEAS =',I4,2X,'C1 =',F7.3,2X,'C2 =',F7.3,
     2 'ARVA =',F6.3,2X,'P =',F6.3,2X,'CLD =',F5.3)
C
      WRITE (IB,930) BETA(ISEAS),GAM(ISEAS),TB,T
  930 FORMAT (1H ,'BETA= ',E12.5,2X,'GAM= ',E12.5,2X,
     $  'TB= ',E12.5,2X,'T= ',E12.5)
  900 CONTINUE
C
  910 CONTINUE
C
C     SAVE CURRENT VALUE OF THE STOCHASTIC COMPONENT FOR
C     USE IN THE NEXT TIME PERIOD
C
      C1 = C2
C
      RETURN
      END
C
C..................................................................
C
C
      FUNCTION ARVA (PDF,N,A,B,COORD,ISEAS)
C
C     FUNCTION ARVA SELECTS A RANDOM VARIABLE FROM AN
C     ARBITRARY DISCRETE PROBABILITY MASS FUNCTION
C
C     PDF    = DISCRETE PROBABILITY DENSITY FUNCTION
C     N      = NUMBER OF INTERVALS
C     A      = LOWER LIMIT OF U(A,B)
C     B      = UPPER LIMIT OF U(A,B)
C     ISEED  = SEED FOR RANDU
C     COORD  = CONTAINS COORDINATES OF THE INTERVALS
C              OF PDF. (COORD(I-1).LT.X.AND.X.LE.COORD(I))
C
C
      REAL PDF(1), PEAK, COORD(1)
      COMMON /SEED/ ISEED
      COMMON /DBUG/ NBUG,DEBUG
```

```
      COMMON /IO/     IN,IS,IB
      DOUBLE PRECISION     IARVA,DEBUG(1)
      DATA IARVA /'ARVA'/
C
C     FIND THE PEAK OF THE DISTRIBUTION
C
      PEAK = 0.0
      DO 100 I = 1,N
      IF (PDF((ISEAS-1)*N +I).GT.PEAK) PEAK = PDF((ISEAS-1)*N +I)
  100 CONTINUE
C
C     SELECT THE FIRST RANDOM NUMBER FROM U(A,B)
C
  150 IX = ISEED
      CALL RAND1 (IX,ISEED,R)
C     CALL RANDU (IX,ISEED,R)
      U1 = A + (B-A) * R
C
C     FIND WHICH INTERVAL U1 BELONGS TO
C
      DO 200 I = 1,N
      IF (U1 .GT. COORD((ISEAS-1)*N + I )) GO TO 200
      J= (ISEAS-1)*N +I
      GO TO 300
  200 CONTINUE
      WRITE (IS,250)
  250 FORMAT (1H1, ' SUBROUTINE ARVA -- U1 IS GREATER ',
     1'THAN THE MAXIMUM INTERVAL FOR THE DISCRETE PDF')
      STOP
  300 CONTINUE
C
C     CALCULATE THE SELECTION CRITERION
C
      F= PDF(J)/PEAK
      IX = ISEED
      CALL RAND1 (IX,ISEED,U2)
C     CALL RANDU (IX,ISEED,U2)
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 600
      DO 500 I = 1,NBUG
      IF (DEBUG(I).NE.IARVA) GO TO 500
      WRITE (IB,550) PEAK,U1,U2,F
  550 FORMAT (///1H ,'FUNCTION ARVA',2X,'PEAK =',F5.3,
     1 'U1 =',F6.3,2X,'U2 =',F6.3,2X,'F =',F6.3)
  500 CONTINUE
C
  600 CONTINUE
C
C     ACCEPT OR REJECT U1
C
      IF (U2.GT.F) GO TO 150
      ARVA=U1
      RETURN
```

```
      END
C
C.......................................................................
C
      SUBROUTINE MARKOV (N,PDF,COORD,A,B,XBAR,XDEV,XRHO,XLAG1,K,X)
C
C  MARKOV IS A GENERAL ROUTINE TO COMPUTE A STOCHASTIC VARIATE
C  GENERATED BY A FIRST ORDER MARKOV PROCESS.
C
      DIMENSION PDF(1),COORD(1),XBAR(1),XDEV(1),XRHO(1)
C
C
      NSEAS = 1
C
      GO TO  (200,300) K
C
  200 CONTINUE
C
C  SECTION 1 -- USE THIS SECTION WHEN ARV IS SELECTED FROM
C               AN ARBITRARY PDF WITH MEAN = XBAR AND
C               STANDARD DEVIATION = XDEV
C
C  DETERMINE THE RANDOM VARIATE
C
      ARV = ARVA (PDF,N,A,B,COORD,NSEAS)
C
      X= XBAR(NSEAS) + XRHO(NSEAS)*(XLAG1 - XBAR(NSEAS)) +
     $    SQRT(1.0-XRHO(NSEAS)*XRHO(NSEAS))*(ARV-XBAR(NSEAS))
C
C
      GO TO 800
  300 CONTINUE
C
C  SECTION 2 -- USE THIS SECTION WHEN ARV IS FROM A
C               STANDARDIZED NORMAL DISTRIBUTION ( N(0,1) )
C     ARV = ARVA (PDF,N,A,B,COORD,NSEAS)
      CALL NORMAL (ARV)
      X= XBAR(NSEAS) + XRHO(NSEAS)*(XLAG1 - XBAR(NSEAS)) +
     $    SQRT(1.0-XRHO(NSEAS)*XRHO(NSEAS))*(ARV*XDEV(NSEAS))
C
C
  800 CONTINUE
C
C
C
C     XLAG1 COULD BE SET EQUAL TO X AT THIS POINT OR CHECKED FOR
C     NEGATIVE NUMBERS.  HOWEVER THE NATURE OF THESE CHECKS DEPENDS
C     ON THE VARIATE BEING GENERATED.  THEREFORE, THESE CHECKS ARE
C     MADE IN THE CALLING ROUTINE WHERE THE IDENTITY OF THE VARIATE
C     IS KNOWN ALONG WITH THE PECULIARITIES ASSOCIATED WITH IT.
C
      IDEBUG = 0
      IF (IDEBUG .EQ.0) RETURN
      WRITE (5,100) XBAR(NSEAS),XDEV(NSEAS),XRHO(NSEAS),ARV,XLAG1,X
  100 FORMAT (/1X,6(E11.4,1X))
```

```
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE LIMITS (DTSL,R,S,TO,RHO,T12,SIGMA,T23)
C
C     FIND LIMITS FOR TEMPERATURE INTEGRATION
C
C
      TO  = - DTSL
      T23 = 23.00 - DTSL
C
      IF (DTSL.LT.0.0) GO TO 50
C
C     FIND LIMITS OF INTEGRATION WHEN OBSERVER IS
C     WEST OF THE STANDARD MERIDIAN
C
C     FOR SUNRISE
C
      RHO = AINT(R+1.0) - DTSL
      IF (RHO .LT. R) RHO = RHO + 1
C
C     FOR SUNSET
C
      SIGMA = AINT(S+1) - DTSL
      IF (SIGMA .LT. S) SIGMA = SIGMA + 1
C
C     FOR LOCAL NOON
C
      T12 = 13.0 - DTSL
C
      GO TO 75
C
   50 CONTINUE
C
C      FIND LIMITS OF INTEGRATION WHEN OBSERVER
C      IS EAST OF THE STANDARD MERIDIAN
C
      RHO = AINT(R) - DTSL
      IF (RHO .LT. R) RHO = RHO + 1
C
      SIGMA = AINT(S) - DTSL
      IF (SIGMA .LT. S) SIGMA = SIGMA + 1
C
      T12 = 12.0 - DTSL
C
   75 CONTINUE
C
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE TEMPK (DELTA,PHI,B, TPRIME, K0, K1, K2, K3, K4, K5, K6)
```

```
C
C      SUBROUTINE TEMPK COMPUTES THE COEFFICIENTS
C      FOR THE TEMPERATURE EQUATION
C
C
C
C      DELTA  = DECLINATION OF SUN IN RADIANS
C      PHI    = LATITUDE IN RADIANS
C      B      = VECTOR OF REGRESSION COEFFICIENTS
C      TPRIME = YESTERDAY'S TEMPERATURE AT 11 PM
C      K0-K6  = COEFFICIENTS IN TEMPERATURE EQUATION
C      B0-B6  = EQUIVALENCED VARIABLES WITH B VECTOR ELEMENTS
C      P      = CONSTANT = 2*PI/24
C      B2P2   = INTERMEDIATE VARIABLE USED FREQUENTLY
C
C
C
       REAL K0,K1,K2,K3,K4,K5,K6
       DIMENSION B(1)
C
       B0 = B(1)
       B1 = B(2)
       B2 = B(3)
       B3 = B(4)
C
       P        = 3.14159/12.0
       B2P2     = B1*B1 + P*P
C
       K0       = TPRIME
C
       K1       = B0/B1
C
       K2       = B2*SIN(DELTA)*SIN(PHI)/B1
C
       K3       = B1*B2*COS(DELTA)*COS(PHI)/B2P2
C
       K4       = P*B2*COS(DELTA)*COS(PHI)/B2P2
C
       K5       = P*P*B3*COS(DELTA)*COS(PHI)/B2P2
C
       K6       = P*B1*B3*COS(DELTA)*COS(PHI)/B2P2
C
       RETURN
       END
C
C
C.............................................................
C
       SUBROUTINE TEMPSN ( ST, DTSL, R,S,B, K0,K1,K2,K3,K4,K5,K6,
      $                  CLD, KBAR, GTO, WSP, WDR, TMPLAG, THAT, T )
C
C
       INTEGER IDBUG
       DOUBLE PRECISION DEBUG(1), DTEMPS
       REAL K0,K1,K2,K3,K4,K5,K6
```

```
      REAL I0,I1,I2,I3,I4,I5,I6,I7, KBAR
C
      DIMENSION B(1)
C
      COMMON /INTEG/ I0,I1,I2,I3,I4,I5,I6,I7
      COMMON /IO/    IREAD, IWRITE, IWBUG
      COMMON /DEBUG/  NBUG, DEBUG
      COMMON /SWITCH/ SWICH1, SWICH2
C
      DATA DTEMPS / 'TEMPSN  ' /
C
C
C         STATEMENT FUNCTIONS FOR INTEGRALS I2 AND I3
C
C
      FUNC1(A,B) = K2*(EXP(B1*A) - EXP(B1*B))
      FUNC2(A)   = EXP(B1*A)*(K3*COS(P*A) + K4*SIN(P*A))
      FUNC3(B)   = EXP(B1*B)*(K6*SIN(P*B) - K5*COS(P*B))
C
C
C
C
C         SET DEBUG FLAG
C
             IDBUG = 1
C
C
C
C         SET SWITCHES THAT DETERMINE WHICH PREDICTORS ARE USED
C
             SWICH1 = 1
             SWICH2 = 0
C
C
      B1 = B(2)
      B4 = B(5)
      B5 = B(6)
      B6 = B(7)
      B7 = B(8)
C
C
C8000 WRITE (5,9000) (B(J),J=1,8)
C9000 FORMAT (1H0,'TEMPS B VECTOR',5X, 4(E12.5,5X)/T20,4(E12.5,5X))
C
C
         IF ( SWICH1 .EQ. 0 ) KBAR = 1.000000
C
C
      P = 3.14159/12.00000
C
C
C         CONVERT STANDARD  TIME TO LOCAL TIME
C
      T = TAU(ST)* 12.0/3.14159   -  12.0
C
```

```
      IF ( ST .LT. 0.5 .AND. T .LT. 0.0 ) GO TO 5
      IF ( ST .GT.22.5 .AND. T .GT. 24. ) GO TO 5
      IF ( T .LT. 0.0 ) T = T + 24.00
      IF ( T.GT. 24.00 ) T = T - 24.00
    5 CONTINUE
C
C
C     IN ADDITION TO SUNRISE AND SUNSET DETERMINE THE LIMITS
C     OF THE RANGES OF THE TEMPERATURE EQUATIONS
C
      T0 = -DTSL
      T12 = 12.0 - DTSL
      T23 = 23.0 - DTSL
C
      TP = T0 - 1.0
C
      IF(IDBUG.EQ.0) WRITE (IWRITE,10) T0,R,T12,S,T23,TP
   10 FORMAT (1H ,T40,6(2X,E10.3)/)
C
C
C         THE FORM OF INTEGRALS I1, I4, I5, I6, AND I7 ARE THE SAME
C         FOR ALL TIMES OF THE DAY.  I2 AND I3 WILL VARY IN
C         FORM DEPENDING ON THE TIME OF DAY.
C
C
C         COMPUTE I1, I4, I5, I6, I7
C
C
      I1 = K1*(EXP(B1*T) - EXP(B1*TP))
C
C
          IF ( SWICH1 .EQ. 0 ) GO TO 40
C
C
C
      PP = (1.0-EXP(-B1))*EXP(B1*T)/B1
      QB = 1.579E-8*(1.00+0.17*CLD**2.)*(TMPLAG+460.)**6.
      I4 = B4*QB*PP + I4
C
   40 CONTINUE
C
          IF ( SWICH2 .EQ. 0 ) GO TO 50
C
      I5 = B5*GTO*PP + I5
C
C
   50 CONTINUE
C
      I6 = B6*WSP*PP + I6
      I7 = B7*WDR*PP + I7
C
C
C
C         CALCULATE SUBTOTAL
C
```

```
      SUBTOT = I1 + I4 + I5 + I6 + I7
C
C
C
      IF ( T .GT. R ) GO TO 100
C
C
C******** RANGE 1 -- AFTER MIDNIGHT AND BEFORE SUNRISE
C
C
C
      GTT = SUBTOT
C
      GO TO 900
C
C
  100 IF ( T .GT. R + 1.00 ) GO TO 200
C
C
C******** RANGE 2 --FIRST HOUR OR FRACTION AFTER SUNRISE
C
C
C
C
      I2 = FUNC1(T,R) - FUNC2(T) + FUNC2(R)
      I2 = I2*KBAR
C
      I3 = FUNC3(T) - FUNC3(R)
      I3 = KBAR*I3
C
      GTT = I2 + I3 + SUBTOT
C
      GO TO 900
C
C
C
  200 IF( T .GT. T12 ) GO TO 250
C
C
C******** RANGE 3 -- AFTER SUNRISE AND BEFORE NOON
C
          Q1 = FUNC1(T,T-1.0)
          Q2 = FUNC2(T)
          Q3 = FUNC2(T-1.0)
C
      TI2 = FUNC1(T,T-1.0) - FUNC2(T) + FUNC2(T-1.0)
      I2 = TI2*KBAR + I2
C
          IF(IDBUG.EQ.0)WRITE(IWBUG,991)KBAR, I2
  991      FORMAT (2X, 'KBAR ', 2(E12.5,3X))
C
      TI3 = FUNC3(T) - FUNC3(T-1.0)
      I3  = TI3*KBAR + I3
C
C
```

```
      GTT = I2 + I3 + SUBTOT
C
      GO TO 900
C
C
C
  250 IF ( T .GT. T12 + 1.0 ) GO TO 300
C
C
C
C********* RANGE 4 -- FIRST HOUR AFTER LOCAL NOON
C
      TI2 = FUNC1(T,T-1.0) - FUNC2(T) + FUNC2(T-1.0)
      I2  = TI2*KBAR + I2
C
      TI3 = FUNC3(12.0) - FUNC3(T-1.0)
      I3  = TI3*KBAR + I3
C
      GTT = I2 + I3 + SUBTOT
      GO TO 900
C
  300 IF ( T .GT. S ) GO TO 400
C
C
C******** RANGE 5 -- AFTER LOCAL NOON BUT BEFORE SUNSET
C
C
      TI2 = FUNC1(T,T-1.0) - FUNC2(T) + FUNC2(T-1.0)
      I2  = TI2*KBAR + I2
C
C
      GTT = I2 + I3 + SUBTOT
C
      GO TO 900
C
C
C
  400 IF (T .GT. S+1.0 ) GO TO 500
C
C
C******** RANGE 6 -- FIRST HOUR AFTER SUNSET
C
C
      TI2 = FUNC1(S,T-1.0) - FUNC2(S) + FUNC2(T-1.0)
      I2  = TI2*KBAR + I2
C
C
      GTT = I2 + I3 + SUBTOT
C
      GO TO 900
C
C
C
  500 CONTINUE
C
```

```
C
C
C******** RANGE 7 -- AFTER SUNSET
C
C
C         NOTE: I2 DOES NOT CHANGE WHEN T .GT. S
C               I3 DOES NOT CHANGE WHEN T .GT. 12.0
C         THUS USE THE PREVIOUSLY COMPUTED VALUES FOR I2 AND I3
C         THAT HAVE BEEN STORED IN THE COMMON /INTEG/
C
C
      GTT = I2 + I3 + SUBTOT
C
C
C
  900 CONTINUE
C
C
C         NOW THAT THE FUNCTION GTT HAS BEEN EVALUATED,
C         COMPUTE THE TEMPERATURE AT TIME T.
C
C
          IF(IDBUG.EQ.0)WRITE(5,990) GTT,I1,I2,I3,I4,I5,I6,I7
  990     FORMAT ( 2X, 'GTT ',8(E12.5,3X))
C
C
C
      THAT = K0*EXP(-B1*(T-TP)) + GTT*EXP(-B1*T)
C
C
C8010 WRITE (5,9010) THAT,K0,B1,T,TP,GTT
C9010 FORMAT (1H0, 'TEMPSN', 6(E12.5,5X))
C
C
C
C**********************************************************************
      RETURN
 5990 CONTINUE
C**********************************************************************
C
C
C         DEBUG INFORMATION FOR TEMPSN
C
C
      WRITE (5,6000)
 6000 FORMAT (1H1/2(1H*,100(1H )/),1H*,25(4H****))
C
      WRITE (5,6005)
 6005 FORMAT (1H ,10(4H****),T44,'DEBUG TEMPSN',T61,10(4H****)//)
C
      WRITE (5,6010) T, TP, R, S
 6010 FORMAT (1H ,'TIME PARAMETERS ',7X,1HT,12X,2HTP,13X,1HR,
     $            13X,1HS/15X,4(3X,F11.4))
C
      WRITE (5,6020)
```

```
 6020 FORMAT (//1H ,7X,'K0',12X,'K1',12X,'K2',12X,'K3',12X,
     $           'K4',12X,'K5',12X,'K6')
C
      WRITE (5,6030)K0,K1,K2,K3,K4,K5,K6
 6030 FORMAT (1H ,8(3X,E11.4))
C
      WRITE (5,6040)
 6040 FORMAT (//,1H ,7X,2HB0,12X,2HB1,12X,2HB2,12X,2HB3,12X,2HB4,
     $           12X,2HB5,12X,2HB6,12X,2HB7)
C
      WRITE (5,6030) (B(M),M=1,8)
C
      WRITE (5,6035)
 6035 FORMAT (//1H ,6X,3HCLD,10X,6HTMPLAG,9X,3HWSP,12X,3HWDR)
      WRITE (5,6030) CLD, TMPLAG, WSP, WDR
C
      WRITE (5,6050)
 6050 FORMAT (//1H ,6X,3HGTT,12X,2HI1,12X,2HI2,12X,2HI3,12X,2HI4,
     $             12X,2HI5,12X,2HI6,12X,2HI7)
C
      WRITE (5,6030) GTT,I1,I2,I3,I4,I5,I6,I7
      WRITE (5,6060)
 6060 FORMAT (//1H ,5X,4HTIME,11X,4HTEMP)
C
      WRITE (5,6030) T,THAT
C
      WRITE (5,6070)
 6070 FORMAT (//1H ,25(4H****))
C
C
C
C*******************************************************************
C*******************************************************************
C
C
      RETURN
      END
C
C...................................................................
C
      SUBROUTINE LONGWV (TF1,TF2,TDF1,TDF2,CLD1,CLD2,LW)
C
C
C         COMPUTE LONGWAVE RADIATION
C
C         TC.....TEMPERATURE IN DEGREES CELSIUS
C         CLD....CLOUD COVER (0<= CLD >=1)
C         LW.....COMPUTED LONGWAVE RADIATION
C         TDC....DEWPOINT TEMPERATURE IN DEG C
C         VP.....VAPOR PRESSURE IN MILLIBARS
C         SVP....SATURATED VAPOR PRESSURE IN MILLIBARS
C
      REAL LW
C
      COMMON /VAPORP/ VP
```

```
C
C
C
C  CONVERT DEG F TO DEG C
C
      TDAVE = (TDF1+TDF2)/2.0
      TAVE  = (TF1+TF2)/2.0
C
      TC = (TAVE - 32.0)*(5.0/9.0)
      TDC = (TDAVE - 32.0)*(5.0/9.0)
C
C
C         CONVERT CELSIUS TO KELVIN
C
      TK = TC + 273.16
C
C
C         DEFINE THE VALUE OF THE STEPHEN-BOLTZMAN CONSTANT
C            (CAL/(CM**2 * MIN * K**4))
C
      S = 0.826E-10
C
C
C         DEFINE THE VALUE OF ATMOSPHERIC EMISSIVITY
C
      CALL VAPOR (TC,TDC,VP,SVP)
C
      E = 0.70 + 5.95E-05*VP*EXP(1500./TK)
C
C
C         COMPUTE LONGWAVE RADIATION
C
      LW = E*S*TK**4.0
C
C
C         ACCOUNT FOR CLOUDINESS
C
      C = ( 1.0 + 0.17*CLD**2.0 )
C
      LW = C*LW
C
C
C  COMPUTE TOTAL LONGWAVE FOR ONE HOUR (IE. 60 MINUTES)
C
      LW =  LW*60.0
C
C
      RETURN
      END
C
C.....................................................................
C
      SUBROUTINE DEWSIM (ACOEF,DEWLAG,TEMP,CLD,WDIR,WSP,DEW)
C
C
```

```
C   DEWSIM USES THE FOLLOWING MODEL TO GENERATE DEWPOINTS
C
C   TD(T) = A0 + A1*TD(T-1) + A2*TMP + A3*CLD + A4*WDIR + A5**WSP
C
C
        DIMENSION ACOEF(1)
C
C   GENERATE TODAY'S DEWPOINTS
C
C
C
        DEW = ACOEF(1) +ACOEF(2)*DEWLAG +ACOEF(3)*TEMP +
       $      ACOEF(4)*CLD + ACOEF(5)*WDIR +ACOEF(6)*WSP
C
C
        RETURN
        END
C
C.......................................................................
C
        SUBROUTINE MSTAT (N,A,B,C,D,E,RAWSUM,XXT)
C
C
C           ACCUMULATE RAW SUMS AND RAW SUMS OF SQUARES AND CROSS PRODUCTS
C
C
        DIMENSION A(N),B(N),C(N),D(N),E(N),RAWSUM(5),XXT(5,5),X(5)
C
C
        DO 100 I = 1,N
C
C         LOAD DATA INTO WORK ARRAY
C
        X(1) = A(I)
        X(2) = B(I)
        X(3) = C(I)
        X(4) = D(I)
        X(5) = E(I)
C
C
C         COMPUTE RAW SUMS
C
        DO 200 J = 1,5
C
           RAWSUM(J) = RAWSUM(J) + X(J)
C
        DO 200 K = 1,5
C
           XXT(K,J) = XXT(K,J) + X(K)*X(J)
C
  200 CONTINUE
C
  100 CONTINUE
C
        RETURN
```

```
      END
C
C.................................................................
C
C
      SUBROUTINE FSTAT (IDIM,RAWSUM,XXT,MEAN,COVMAT,CORMAT,NDATA)
C
C
C         COMPUTE THE MEAN VECTOR, THE COVARIANCE MATRIX, AND THE
C            CORRELATION MATRIX
C
      DIMENSION RAWSUM(IDIM), XXT(IDIM,IDIM), MEAN(IDIM)
      DIMENSION COVMAT(IDIM,IDIM), CORMAT(IDIM,IDIM)
      DIMENSION MMT(5,5)
      REAL MEAN,MMT
C
C
C     COMPUTE MEANS AND AVERAGE CROSS PRODUCTS
C
      DO 100 I = 1,IDIM
        MEAN(I) = RAWSUM(I)/NDATA
      DO 100 J = 1,IDIM
        XXT(J,I) = XXT(J,I)/NDATA
  100 CONTINUE
C
C
C     MULTIPLY THE MEAN VECTOR BY ITS TRANSPOSE
C
      DO 200 I = 1,IDIM
      DO 200 J = 1,IDIM
         MMT(J,I) = MEAN(J)*MEAN(I)
  200 CONTINUE
C
C
C     COMPUTE COVARIANCE MATRIX
C
      DO 300 I = 1,IDIM
      DO 300 J = 1,IDIM
         COVMAT(J,I) = XXT(J,I) - MMT(J,I)
  300 CONTINUE
C
C
C     COMPUTE THE CORRELATION MATRIX
C
      DO 400 I = 1,IDIM
      DO 400 J = 1,IDIM
         CORMAT(J,I) = COVMAT(J,I)/SQRT(COVMAT(J,J)*COVMAT(I,I))
  400 CONTINUE
C
C
      RETURN
      END
C
C.................................................................
C
```

```
      SUBROUTINE RAWLAG (IDIM,JP,DATA,SUM,SUMSQ,SUM3,XBAR,XVAR,XSKEW,
     $                        NR,R)
C
C
C   UPDATE ARRAY FOR AUTOCORRELATION ANALYSIS
C
C   IDIM.....DIMINSION OF DATA ARRAY AND MAX LAG
C   JP.......POINTER FOR CURRENT OR LATEST DATUM
C   DATA.....DATA ARRAY ( A 'CIRCULAR' DATA ARRAY )
C   R........SUM OF SQUARES AND CROSS-PRODUCT ARRAY
C
C
      DIMENSION DATA(IDIM), R(IDIM)
      COMMON /IO/ IN,IS,IB
C
            BUG = 'OFF'
      IK = JP
C
C
C   REFERENCE FOR THE EQUATIONS TO COMPUTE AUTOCORRELATION:
C
C              HAAN, CHARLES T.;STATISTICAL METHODS IN HYDROLOGY,
C              IOWA STATE UNIVERSITY PRESS,1977, PAGE 228, EQ (11.13)
C
C
       IF (BUG .EQ. 'ON') WRITE (5,910)
  910  FORMAT (' RA.LAG1')
C
C
      X = DATA(JP)
      CALL STAT (X,SUM,SUMSQ,SUM3,XBAR,XVAR,XSKEW,NR)
C
      DO 100 K = 1,IDIM
C
      IF (BUG .EQ. 'ON') WRITE (5,920)
  920 FORMAT (' RAWLAG2')
C
C
           IF (BUG .EQ. 'ON' ) WRITE (IB,900) K,JP,IK
C
C
      R(K) = R(K) + DATA(JP)*DATA(IK)
C
      IK = IK - 1
      IF (IK .LE. 0) IK = IK + IDIM
C
  100 CONTINUE
C
C
  900 FORMAT (1X,'K= ',I2,2X,'JP=',I2,2X,'IK=',I2)
C
C
      RETURN
      END
C
```

```
C.............................................................................
C
      SUBROUTINE AUTOCO (MLAG, RHO, XBAR, XVAR, NN, TITLE)
C
C  DETERMINE THE AURTOCORRELATION FUNCTION.  THE MAXIMUM LAG IS MLAG.
C
C
C  REFERENCE FOR THE EQUATIONS TO COMPUTE AUTOCORRELATION :
C
C            HAAN, CHARLES T.;STATISTICAL METHODS IN HYDROLOGY,
C            IOWA STATE UNIVERSITY PRESS,1977, PAGE 228, EQ (11.13)
C
C
C  MLAG.....MAXIMUM LAG
C  RHO......RAW DATA IN --- AUTOCORRELATION OUT
C  XBAR.....MEAN OF CURRENT DATA TYPE
C  XVAR.....VARIANCE OF CURRENT DATA TYPE.
C  NN.......NUMBER OF DATA POINTS IN MONTH
C
C
      DIMENSION RHO(MLAG)
      DIMENSION TITLE(1)
      COMMON /IO/ IN, IS, IB
C
          BUG = 'OFF'
          IF (BUG .EQ. 'ON') WRITE (IB,9900) (TITLE(M),M=1,5)
          IF (BUG .EQ. 'ON') WRITE (IB,9910) (RHO(K),K=1,MLAG),NN,XBAR,
     $                                                          XVAR
C
      DO 100 K = 1,MLAG
C
      RHO(K) = (RHO(K) - NN*XBAR*XBAR)/((NN-1)*XVAR)
C
  100 CONTINUE
C
          IF (BUG .EQ. 'ON') WRITE (IB,9910) (RHO(K),K=1,MLAG)
C
      WRITE (IS,900)
  900 FORMAT (1H1,15(5H     ))
C
      WRITE (IS,910) (TITLE(M),M=1,5)
  910 FORMAT (1H ,15X, 'AUTOCORRELATION FUNCTION FOR ',5A5/)
C
      WRITE (IS,920) (K,K=0,11),(RHO(K),K=1,12)
  920 FORMAT (7X,'LAG ',12I5/6X,13(5H-----)/7X,'RHO ',12F5.2/)
      WRITE (IS,920) (K,K=12,23),(RHO(K),K=13,24)
C
 9900 FORMAT (1X,16A5)
 9910 FORMAT (1X,4(6F10.2/)/1X,'NN= ',I6,5X,'XBAR= ',F10.2,
     $                                       'XVAR= ',F10.2//)
C
C
      RETURN
      END
C
```

```
C.............................................................................
C
      SUBROUTINE MARGAM(N,PDF,COORD,A,B,XBAR,XDEV,XRHO,SKEW,XLAG1,X,ARV)
C
C  GENERATE THE NEXT DATUM OF A FIRST ORDER MARKOV PROCESS WHOSE
C  VARIATES ARE GAMMA DISTRIBUTED.
C
C  N........NUMBER OF ORDINATES IN PDF
C  PDF......ARRAY CONTAINING ELEMENTS OF PROBABILITY DISTRIBUTION
C           FUNCTION (HISTOGRAM FORM) WHICH IS N(0,1).
C  COORD....COORDINATES OF PDF
C  A........LEFT BOUND OF PDF
C  B........RIGHT BOUND OF PDF
C  XBAR.....PROCESS MEAN
C  XDEV.....PROCESS STANDARD DEVIATION
C  XRHO.....PROCESS LAG-1 AUTOCORRELATION COEFFICIENT
C  XLAG1....PREVIOUS VALUE OF PROCESS
C  SKEW.....GAMMA DISTRIBUTION SKEW COEFFICIENT
C  X........CURRENT VALUE OF THE PROCESS
C
C
C  REFERENCE:
C           HAAN, CHARLES T.:STATISTICAL METHODS IN HYDROLOGY,
C           IOWA STATE UNIVERSITY PRESS, 1977
C
C
      DIMENSION PDF(1),COORD(1),XBAR(1),XDEV(1),XRHO(1),SKEW(1)
C
C  SET NUMBER OF SEASONS TO ONE
C
      NSEAS = 1
C
C  EVALUATE RANDOM COMPONENT DISTRIBUTED ACCORDING TO PDF
C
C  10 ARV = ARVA (PDF,N,A,B,COORD,NSEAS)
   10 CALL NORMAL (ARV)
C
C  TO CONTERACT THE PROBLEM OF SUDDEN SHIFTS IN A GENERATED TIME SERIES
C  WHOSE VARIATE IS SKEWED AND HAS A HIGH (EG. GREATER THAN .8) LAG-1
C  AUTOCORRELATION COEFFICIENT, RESTRICT THE USAGE OF THE TAIL OF THE
C  N(0,1) THAT CAUSES THE PROBLEM.
C
C  BY RESTRICTING EXCURSIONS INTO THE OFFENDING TAIL TO ABSOLUTE VALUES
C  BELOW 2.8, ONLY 0.26 PERCENT OF THE DISTRIBUTION IS RESTRICTED.
C
C     1. IF THE SKEW IS NEGATIVE, RESTRICT THE NEGATIVE TAIL OF N(0,1)
C     2. IF THE SKEW IS POSITIVE, RESTRICT THE POSITIVE TAIL OF N(0,1)
C
      IF (ABS(ARV) .LE. 2.8) GO TO 40
C
      IF (ARV) 20, 40, 30
   20 IF (SKEW(NSEAS)) 10, 40, 40
   30 IF (SKEW(NSEAS)) 40, 40, 10
C
   40 CONTINUE
```

```
C
C
C
C   EVALUATE RANDOM COMPONENT
C
       CSE = (1.0 -XRHO(NSEAS)**3.0)*SKEW(NSEAS)/
      $           (1.0 - XRHO(NSEAS)**2.0)**1.5
C
       E   = (2.0/CSE)*(1.0+CSE*ARV/6.0 - CSE*CSE/36.)**3.0
      $           - 2.0/CSE
C
C   GENERATE THE NEXT VALUE OF THE PROCESS
C
       X = XBAR(NSEAS) + XRHO(NSEAS)*(XLAG1 - XBAR(NSEAS)) +
      $           E*XDEV(NSEAS)*SQRT(1.0-XRHO(NSEAS)*XRHO(NSEAS))
C
C
C      XLAG1 COULD BE SET EQUAL TO X AT THIS POINT OR CHECKED FOR
C      NEGATIVE NUMBERS.  HOWEVER THE NATURE OF THESE CHECKS DEPENDS
C      ON THE VARIATE BEING GENERATED.  THEREFORE, THESE CHECKS ARE
C      MADE IN THE CALLING ROUTINE WHERE THE IDENTITY OF THE VARIATE
C      IS KNOWN ALONG WITH THE PECULIARITIES ASSOCIATED WITH IT.
C
       IDEBUG = 0
       IF (IDEBUG .EQ.0) RETURN
       WRITE (5,100) XBAR(NSEAS),XDEV(NSEAS),XRHO(NSEAS),ARV,XLAG1,X
      $                 ,SKEW,CSE,E
  100 FORMAT (/1X,6(E11.4,1X))
       RETURN
       END
C
C.....................................................................
C
       SUBROUTINE PUNCH (IPUNCH, RAINM, VP, WSPM, SWR, LW, TEMPM)
C
C   CONVERT THE DATA GENERATED BY THE CSCS MODEL TO DATA WITH UNITS
C   COMPATABLE WITH MILLY'S LAND SURFACE MODEL.
C
C*********************************************************************
C******   INPUT VARIABLES   ******************************************
C*********************************************************************
C
C   IPUNCH ... UNIT NUMBER FOR OUTPUT DATA FILE
C   RAINM  ... PRECIPITATION IN MM/HR
C   VP     ... VAPOR PRESSURE IN MILLIBARS
C   WSPM   ... WIND SPEED IN M/SEC
C   SWR    ... SOLAR RADIATION IN LANGLEYS/HR
C   LW     ... LONGWAVE RADIATION IN LANGLEYS/HR
C   TEMPM  ... TEMPERATURE IN DEG C
C
C*********************************************************************
C******   OUTPUT VARIABLES   *****************************************
C*********************************************************************
C
C   PRECIP ... PRECIPITATION IN CM/SEC
```

```
C   RHOVA   ... WATER VAPOR DENSITY IN GRAMS/CM**3
C   UA      ... WIND SPEED IN CM/SEC
C   RADS    ... SHORTWAVE RADIATION IN LANGLEY/SEC
C   RADLD   ... LONGWAVE RADIATION IN LANGLEY/SEC
C   TEMPM   ... TEMPERATURE IN DEG C
C
C************************************************************************
C
      REAL LW
C
C
C*****  PRECIPITATION CONVERSION
C         CM/SEC = (MM/HR)*(CM/10MM)*(HR/3600SEC)
C
      PRECIP = RAINM/36000.00
C
C
C*****  VAPOR DENSITY CONVERSION
C
      RHOVA = (0.622/2.876E+06) * VP / (273.16 + TEMPM)
C
C
C
C*****  WIND SPEED CONVERSION
C         CM/SEC = (M/SEC)*(100CM/M)
C
      UA = WSPM*100.00
C
C
C*****  RADIATION CONVERSION
C         LANGLEY/SEC = (LANGLEY/HR)*(HR/3600.)
C
      RADS = SWR/3600.
      RADLD = LW/3600.
C
C
C*****  DATA OUTPUT SECTION
C
C     WRITE (IPUNCH,900 ) PRECIP, RHOVA, UA, RADS, RADLD, TEMPM
      WRITE (IPUNCH,900) RAINM, VP, WSPM, SWR, LW, TEMPM
  900 FORMAT (6E10.3)
C
C************************************************************************
C
      RETURN
      END
C
C.......................................................................
C
      SUBROUTINE PLOT (IM,ID,IH,NMAX,TEMP,DEW,SWR,WRL,CLD,RAIN,
     $                 WSP,WDIR,STORM,IPL,TS,TE)
C
C  DATA PLOTTING SUBROUTINE
C
C  IM.......CURRENT MONTH
```

```
C   ID.......CURRENT DAY
C   IH.......CURRENT HOUR
C   NMAX.....MAXIMUM NUMBER OF LINES BEFORE NEW TITLE AND HEADING
C   TEMP.....TEMPERATURE
C   DEW......DEWPOINT TEMPERATURE
C   SWR......SHORT WAVE RADIATION
C   WRL......LONG WAVE RADIATION
C   CLD......CLOUD COVER
C   RAIN.....RAINFALL
C   WSP......WIND SPEED
C   WDIR.....WIND DIRECTION
C   IPL......FLAG FOR WHICH COMBINATION OF DATA IS PLOTTED
C              IPL = 1  TEMP, DEW, SWR, WRL, CLD, AND RAIN PLOTTED
C              IPL = 2  SWR, WRL, CLD, RAIN PLOTTED
C              IPL = 3  WSP, WDIR PLOTTED
C   STORM....ON/OFF FLAG TO DETERMINE IF IT IS RAINING
C   TB.......BEGINNING OF TEMPERATURE RANGE FOR ORDINATE SCALE
C   TE.......END OF TEMPERATURE RAND FOR ORDINATE SCALE
C
C
      DIMENSION SYMBOL(135)
      INTEGER CAL(12)
      INTEGER PPTEMP, PPDEW, PPOFF, PPSWR, PPLWR, PPCLD
C
      COMMON /LINES/  NLINES
      COMMON /IO/     IN,IS,IB
      DATA CAL /31,28,31,30,31,30,31,31,30,31,30,31/
C
C
      IC = 5
C
C
C   SET UP PAGE HEADINGS
C
      IF (NLINES .GT. 0) GO TO 200
C
      WRITE (IC,870)
  870 FORMAT(1H1,110(1H )/1H+ ,110(1H ))
C
      GO TO (100,120,140) IPL
C
  100 CONTINUE
      CALL ROUND (TE,ITE)
      CALL ROUND (TB,ITB)
C
C************************************************************************
C
C
C   HEADINGS FOR PLOTTING 6 DATA POINTS
C
      WRITE (IC,880)
  880 FORMAT (1H ,112(1H:))
      WRITE (IC,881)
  881 FORMAT (1H ,35X,' CONSTRAINED STOCHASTIC CLIMATE SIMULATION',
     $                 '   (CSCS)   ',8X)
```

```
      WRITE (IC,882)
  882 FORMAT (1H ,112(1H:)/)
      WRITE (IC,901)
  901 FORMAT (1H ,T15,'HOURLY TEMPERATURES (DEG C)', T71, 'RADIATION ',
     $        '(LANGLY/HOUR)', T105,'CLOUD (*)')
      WRITE (IC,902)
  902 FORMAT (1H ,T18,'(T = TEMP, D = DEW PT)',T68,'(S = SHORT WAVE, '
     $        ,'L = LONG WAVE)',T104,'***RAIN***')
      INCR = 10
      WRITE (IC,903) (I,I=ITB,ITE,INCR), (I,I=20,80,20)
  903 FORMAT (1H ,T6,5I10,T60,4I10,T103,'0    .5    1')
      WRITE (IC,904)
  904 FORMAT (1H ,'MM/DD:HR.',9(5H----+),'---.',8(5H----+),
     $         4H---.,2(5H----+))
C
      NLINES = NLINES + 8
C
      GO TO 200
C
C*****************************************************************************
C
C
  120 CONTINUE
C
C  WRITE HEADINGS FOR 4 VARIABLE PLOT
C
      WRITE (IC,890)
  890 FORMAT (1H ,63(1H:))
  900 FORMAT (1H ,8X,'  CONCEPTUAL STOCHASTIC CLIMATE SIMULATION'
     $                        ,'  (CSCS)  ',8X)
      WRITE (IC,900)
      WRITE (IC,891)
  891 FORMAT (1H ,63(1H:)/)
      WRITE (IC,910)
  910 FORMAT (1H ,T18,'RADIATION (LANGLY/HOUR)', T55,'CLOUD (*)')
      WRITE (IC,911)
  911 FORMAT(1H ,T15,'(S = SHORTWAVE, L = LONGWAVE)',T54,
     '        '*** RAIN **')
      WRITE (IC,912) (I,I=10,70,20)
  912 FORMAT (1H ,T6,4I10,T54,'0    .5    1')
      WRITE (IC,913)
  913 FORMAT (1H ,'MM/DD:HR.',8(5H----+),4H---.,2(5H----+))
C
      NLINES = NLINES + 8
C
      GO TO 200
C
C*****************************************************************************
C
  140 CONTINUE
C
C  RESERVED FOR HEADINGS FOR WIND AND WIND DIRECTION
C
C
C*****************************************************************************
```

```
C
  200 CONTINUE
C
C   DETERMINE PLOTTING POSITIONS
C
C   INITIALIZE THE PLOTTING POINT OFFSET
C
      PPOFF = 1
      IRMIN = 1
C
      GO TO (210,220,230) IPL
C
C
C   DETERMINE PLOTTING POSITIONS
C
  210 TT = TEMP
      CALL ROUND (TT,IT)
      DD = DEW
      CALL ROUND (DD,IDW)
C
C   ADD PLOTTING POSITION OFFSET
C
C   FIRST, CONVERT TB TO UNITS OF 2 DEGREES
C
      TB1 = TB
      CALL ROUND (TB1,ITB1)
C
C   ACCOUNT FOR THE OFFSET FROM THE LEFT SIDE OF THE GRAPH TO TB.
C
      ITB0 = ITB1 - 5
C
C   ADD PLOTTING POSITION OFFSET
C
      PPTEMP = IT - ITB0
      PPDEW  = IDW - ITB0
      PPOFF  = 49 + PPOFF
C
      IRMIN =   PPOFF
C
      IF (PPTEMP .GT. PPOFF) PPTEMP = PPOFF
      IF (PPDEW  .GT. PPOFF) PPDEW  = PPOFF
C
  220 CONTINUE
C
      SW = SWR/2.0
      CALL ROUND (SW,ISW)
      WL = WRL/2.0
      CALL ROUND (WL,IWL)
C
      PPSWR = ISW + PPOFF
      PPLWR = IWL + PPOFF
      PPOFF =  44 + PPOFF
C
      IF (PPSWR .GT. PPOFF) PPSWR = PPOFF
      IF (PPLWR .GT. PPOFF) PPLWR = PPOFF
```

```
C
      CLDY = CLD*10.
      CALL ROUND (CLDY,ICLD)
C
      IF (ICLD .LT. 0) ICLD = 0
      IF (ICLD .GT. 10) ICLD = 10
C
      PPCLD = ICLD + PPOFF
      PPOFF = PPOFF + 10
C
      IF (PPCLD .GT. PPOFF) PPCLD = PPOFF
      NPMAX = PPOFF
      GO TO 250
C
C
C**********************************************************************
C
C
  230 CONTINUE
C
C  RESERVED FOR SETTING PLOTTING POSITIONS FOR WIND AND WIND DIRECTION
C
  250 CONTINUE
C
C**********************************************************************
C
C
C  SET UP SYMBOL ARRAY
C
      DO 300 I = 1,135
      SYMBOL(I) = ' '
      IF (IH .NE. 23) GO TO 300
      IF (NLINES .GE. NMAX-1) GO TO 300
      SYMBOL(I) = '_'
  300 CONTINUE
C
      GO TO (305,320,600) IPL
C
  305 CONTINUE
      GO TO 315
      IF (IDW .LT. 10) GO TO 315
      DO 310 I = 10,IDW,10
      SYMBOL(I) = '!'
  310 CONTINUE
  315 CONTINUE
C
      DO 330 I = PPDEW,PPTEMP
      SYMBOL(I) = '-'
  330 CONTINUE
C
      SYMBOL(PPDEW) = 'D'
      SYMBOL(PPTEMP)= 'T'
  320 CONTINUE
C
C
```

```
      DO 360 I = IRMIN,PPSWR
      SYMBOL(I) = '^'
  360 CONTINUE
      SYMBOL(PPSWR) = 'S'
      SYMBOL(PPLWR) = 'L'
C
      SYMBOL(NPMAX) = 'I'
      SYMBOL(NPMAX-10) = 'I'
      SYMBOL(PPCLD) = '*'
      SYMBOL(1)      = 'I'
      SYMBOL(NPMAX-54) = 'I'
C
C***********************************************************************
C
C
      IF (STORM .EQ. 'OFF') GO TO 370
C
      NPMAX = NPMAX - 10
C
C  PLOT DATA IF STORM IS 'ON'
C
      GO TO (400,410) IPL
C
  400 WRITE (IC,950) IM,ID,IH,(SYMBOL(I),I=1,NPMAX),RAIN
  950 FORMAT (1H ,I2,'/',I2,':',I2,94A1,'***',F4.1,1X,'**')
      GO TO 500
C
  410 WRITE (IC,960) IM,ID,IH,(SYMBOL(I),I=1,NPMAX),RAIN
  960 FORMAT (1H ,I2,'/',I2,':',I2,45A1,'***',F4.1,1X,'**')
      GO TO 500
C
  370 CONTINUE
C
      WRITE (IC,970) IM,ID,IH,(SYMBOL(I),I=1,NPMAX)
  970 FORMAT (1H ,I2,'/',I2,':',I2,120A1)
C
  500 CONTINUE
C
      NLINES = NLINES + 1
      IF (ID.EQ.CAL(IM) .AND. IH.EQ.23) NLINES = 0
      IF (NLINES .GE. NMAX) NLINES = 0
C
      IF (NLINES .GT. 0) GO TO 600
      GO TO (560,570) IPL
  560 WRITE (IC,904)
      WRITE (IC,903) (I,I=ITB,ITE,INCR), (I,I=20,80,20)
      GO TO 600
  570 WRITE (IC,913)
      WRITE (IC,912) (I,I=10,70,20)
  600 CONTINUE
      RETURN
```

APPENDIX F

TEMPERATURE MODEL PARAMETER ESTIMATION PROGRAM LISTING

```
C
C.....................................................................
C
      PROGRAM TEMPER
C
C
C   PROGRAM TEMPER IS USED TO ESTIMATE THE REGRESSION COEFFICIENTS FO
C   THE DETERMINISTIC COMPONENT OF THE TEMPERATURE MODEL.  THE PROGRAM
C   IS CURRENTLY SET UP FOR MONTHLY PARAMETER ESTIMATION.  IF ANY OTHER
C   PERIODS ARE TO BE USED, SUBROUTINE DATE1 WILL HAVE TO BE APMODIFIED
C   IN ORDER THAT THE DATE COUNTERS ARE UPDATED PROPERLY.
C
C   PROGRAMMER:
C                    DAVID C. CURTIS
C                    NORTHEAST RIVER FORECAST CENTER
C                    705 BLOOMFIELD AVE
C                    BLOOMFIELD, CT  06002
C
C                    (203) 244-2520
C
C
      INTEGER RANGE
      INTEGER SEASON(12)
C
      REAL*8 DEBUG(20)
      REAL*8 TZONE,TZ(4)
      REAL KBAROB
C
      DIMENSION XY(8),XXT(8,8)
      DIMENSION A(3),B(3)
      DIMENSION TPRIME(25), THAT(25)
      DIMENSION KBAROB(25), GRTEMP(25), CLOUD(25)
      DIMENSION WSPEED(25), WDIR(25)
      DIMENSION ACOEF(8), BCOEF(8)
C
      COMMON /IO/ IREAD,IWRITE,IWBUG,IPARM, IBCOE
      COMMON /DBUG/ NBUG, DEBUG
      COMMON /SEAS/ NSEAS
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /SUN/   DELTA, DTSL, SR,SS
      COMMON /JDATES/ JULBAT, JULREL, JBEGIN, JULEND, JRANGE, NXLPYR
     $               ,JSTART, JSTOP,  JREND,  JYEAR
      COMMON /DATES/ IYR, IMO, IDAY, LYR, LMO, LDAY
      COMMON /YSTAT/ YSUM, YSUMSQ, YMEAN, RSQUAR
C
      DATA TZ/8HEASTERN  ,8HCENTRAL  ,8HMOUNTAIN  ,8HPACIFIC  /
C
C   SET INPUT/OUTPUT UNIT NUMBERS
C
      IREAD = 21
      IWRITE=  5
      IWBUG = 23
      IPARM = 24
      IBCOE = 25
C
```

```
C
C
C
C     CALL TRANSLATOR PROGRAM FOR INTERACTIVE INPUT
C
      CALL TRTEMP (IREAD)
C
C
C
C
C0000000000000000000000........OPEN........00000000000000000000
C
C   DATA FILE DEFINITION:
C
C     DCCTMP.DAT ... INPUT DATA FILE
C     DCCTMP.OUT ... OUTPUT DATA FILE
C     DCCTMP.BUG ... DEBUG DATA FILE
C     DCCTMP.PRM ... OBSERVED DATA FILE
C     DCBCOF.DAT ... REGRESSION COEFFICIENT OUTPUT FILE
C
C
      OPEN (UNIT=IREAD,DEVICE='DSK',ACCESS='SEGIN',FILE='DCCTMP.DAT')
      IF (IWRITE .EQ. 5) GO TO 85
      OPEN (UNIT=IWRITE,DEVICE='DSK',ACCESS='SEGOUT',FILE='DCCTMP.OUT')
   85 CONTINUE
      OPEN (UNIT=IWBUG,DEVICE='DSK',ACCESS='SEGOUT',FILE='DCCTMP.BUG')
      OPEN (UNIT=IPARM,DEVICE='DSK',ACCESS='SEGIN',FILE='DCCTMP.PRM')
      OPEN (UNIT=IBCOE,DEVICE='DSK',ACCESS='SEGOUT',FILE='DCBCOF.DAT')
C
C0000000000000000000000........OPEN........00000000000000000000
C
C
C   INPUT DATA SECTION
C
CXXXXXREAD (IREAD,100) IREAD, IWRITE, IWBUG
CX100 FORMAT (3I5)
C
      READ (IREAD,110) NBUG, (DEBUG(I), I=1,NBUG)
  110 FORMAT (I5,5X,7(A8,2X)/(10X,7(A8,2X)))
C
 8000 WRITE (IWRITE,9000) NBUG
 9000 FORMAT (1H0,'NBUG=',2X,I10)
C
C
      READ (IREAD,120) IDAY,IMO,IYR,LDAY,LMO,LYR
  120 FORMAT (2(3X,I2,3X,I2,1X,I4))
      WRITE (IWRITE,120)IDAY,IMO,IYR,LDAY,LMO,LYR
C
C
C
C
C
C
C
      READ (IREAD,140) (A(I),I=1,3),(B(I),I=1,3),TZONE
```

```
  140 FORMAT (2(3F5.0,5X),T51,A8)
C
C
C  CONVERT DEGREES TO RADIANS
C
      PHI = DMS(A)
      THETAL = DMS(B)
C
C
C  CHECK THE TIME ZONE TO GET THE PROPER STANDARD MERIDIAN OF THE
C  OBSERVER LOCATION
C
      IF (TZONE.NE.TZ(1)) GO TO 200
      THETAS = 75.0*2.0*3.14159/360.0
      GO TO 300
C
  200 IF (TZONE.NE.TZ(2)) GO TO 210
      THETAS =  90.0*2.0*3.14159/360.
      GO TO 300
C
  210 IF (TZONE.NE.TZ(3)) GO TO 220
      THETAS = 105.0*2.0*3.14159/360.0
      GO TO 300
C
  220 IF (TZONE.NE.TZ(4)) GO TO 230
      THETAS = 120.0*2.0*3.14159/360.0
      GO TO 300
C
  230 WRITE (IWRITE,240)
  240 FORMAT (1H1,'TIME ZONE REQUESTED IS NOT VALID'///)
      WRITE (IWRITE,250) TZONE,(TZ(I),I=1,4)
  250 FORMAT (1H0,T10,'REQUESTED TIME ZONE',T36,5H*****,
     1A8,5H*****/T10,'AVAILABLE TIME ZONES',T36,5H*****,
     2A8,5H*****/T36,5H*****,A8,5H*****/T36,5H*****,
     3A8,5H*****/T36,5H*****,A8,5H*****)
C
  300 CONTINUE
C
C
      READ (IREAD,260) EP,ET,W
  260 FORMAT (16F5.0/9F5.0)
C
C
C         READ DATA BOUND VALUES
C             TLB...TEMPERATURE LOWER BOUND
C             TUB...TEMPERATURE UPPER BOUND
C             WUB...WIND SPEED  UPPER BOUND
C
      READ (IREAD,260) TLB, TUB, WUB
C
C:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C
C         INITIALIZE THE ARRAYS USED IN THE REGRESSION ALGORITHM.
C
      DO 100 I = 1,8
```

```
          XY(I) = 0.0
      DO 100 J =1,8
          XX1(J,I) = 0.0
  100 CONTINUE
C
C
          YSUM   = 0.0
          YSUMSQ = 0.0
          YMEAN  = 0.0
C
C
C
C
C
      IRANK = 7
C
C
C           INITIALIZE THE DATE VARIABLES
C
      CALL DATE1
C
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C
C         EVALUATE OBSERVED DATA TO DEVELOP COEFFICIENTS FOR THE
C            TEMPERATURE MODEL.
C
  325 CONTINUE
C
C  FOR EACH DAY, 25 OBSERVATIONS OF EACH DATA TYPE ARE USED IN THE
C  PARAMETER ESTIMATION.  THE DATA TIME SEQUENCE IS:
C    11PM, MIDNIGHT, 1AM, ... , 11AM, NOON, 1PM, ... , 11PM.
C  IN THIS FORMAT, THE 11PM OBSERVATION APPEARS TWICE. IN THE 25TH
C  LOCATION FOR DAY N AND IN THE 1ST LOCATION FOR DAY N + 1.
C
C
      READ (IPARM,260,END=345) TPRIME
      READ (IPARM,260,END=345) CLOUD
      READ (IPARM,260,END=345) WSPEED
      READ (IPARM,260,END=345) WDIR
C
C
C         CHECK INPUT DATA TO MAKE SURE DATA ARE WITHIN
C            REASONABLE BOUNDS.
C
      DO 326 LL = 1,25
C
      L = LL
         IF (TPRIME(L).GE.TLB .AND. TPRIME(L).LE.TUB)GO TO 327
            CALL DCHECK ( JULREL, 1, TPRIME, L )
  327    IF (CLOUD(L).GE.0.00.AND.CLOUD(L).LE.1.00)  GO TO 328
            CALL DCHECK ( JULREL, 2, CLOUD , L )
  328    IF (WSPEED(L).GE.0.00.AND.WSPEED(L).LE.WUB) GO TO 329
            CALL DCHECK ( JULREL, 3, WSPEED, L )
```

```
  329     IF (WDIR(L).GE.0.00.AND. DIR(L).LE.360.00)  GO TO 326
             CALL DCHECK ( JULREL, 4, WDIR,    L )
C
  326 CONTINUE
C
C
C
C         ESTIMATE RADIATION ATTENUATION DUE TO CLOUD COVER
C
C
          DO 330 I = 1,24
             KBAROB(I) = 1. - 0.65*CLOUD(I)**2.
             IF(WDIR(I) .GT. 180.) WDIR(I)=ABS(WDIR(I)-360.)
  330     CONTINUE
C
C
C         CALL DATA ANALYSIS ROUTINES
C
      CALL PRMEST (TPRIME,KBAROB,CLOUD,GRTEMP,WSPEED,WDIR,IRANK,
     $                  XXT,XY)
C
C         UPDATE THE DATE COUNTERS
C              ENTRY DATE ... FOR YEARLY PARAMETER ESTIMATION
C              ENTRY DATEM... FOR MONTHLY PARAMETER ESTIMATION
C
C
C-----CALL DATE
      CALL DATEM
C
C
C         CHECK TO SEE IF END OF TEST PERIOD HAS BEEN REACHED
C
C
      IF ( JULDAT .LE. JULEND ) GO TO 325
C
  345 CONTINUE
C
C
C         DETERMINE THE 'A' COEFFICIENTS
C
      CALL COEF ( IRANK, XXT, XY, ACOEF )
C
C
C         DETERMINE THE 'B' COEFFICIENTS
C
      CALL ATOB (ACOEF,BCOEF)
C
C
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
C
C
      WRITE (IWRITE,600)
  600 FORMAT (1H1/,2(1H+,100(1H )/),1H ,'TEMPERATURE MODEL PARAMETER ',
```

```
     $             'ESTIMATION PROGRAM OUTPUT'///)
C
      WRITE (IWRITE,610) (ACOEF(I),I=1,8)
  610 FORMAT (1H ,T24, 'A COEFFICIENTS'//
     $          1H ,12X,2HA0,12X,2HA1,12X,2HA2,12X,2HA3/
     $          1H ,4X,4(2X,E12.5)//
     $          1H ,12X,2HA4,12X,2HA5,12X,2HA6,12X,2HA7/
     $          1H ,4X,4(2X,E12.5)///)
C
C
      WRITE (IWRITE,620) (BCOEF(I),I=1,8)
  620 FORMAT (1H ,T24, 'B COEFFICIENTS'//
     $          1H ,12X,2HB0,12X,2HB1,12X,2HB2,12X,2HB3/
     $          1H ,4X,4(2X,E12.5)//
     $          1H ,12X,2HB4,12X,2HB5,12X,2HB6,12X,2HB7/
     $          1H ,4X,4(2X,E12.5)///)
C
      WRITE (IBCOE,625) (BCOEF(I),I=1,8)
  625 FORMAT ('BCOEF 0-3 ',4F12.5/'BCOEF 4-7 ',4F12.5)
C
C
C
      WRITE (IWRITE,630) YMEAN, RSQUAR
  630 FORMAT (1H ,4X,7HYMEAN = ,2X,F5.2, 10X,
     $        10HRSQUARED =  ,2X,F5.2)
C
C
      STOP
      END
C
C
C.....................................................................
C
      SUBROUTINE TRTEMP (IREAD)
C
C
      INTEGER A1(6)
      REAL*8 FILE
      REAL*8 DEBUG(20),TZONE , TZ(4)
      DIMENSION A(3),B(3),BC(7)
      DIMENSION TPRIME(25)
C
C
      IC = 5
C
C
C
C
      WRITE (IC,905)
  905 FORMAT (1H0,T10, 'IF AN INPUT DATA FILE EXISTS, YOU CAN DO THE',
     1    ' FOLLOWING:'/
     2  T15, 'ACTION',T30,'RESPONSE'/T15,'......',T30,'........'/
     3 T10, '1. TYPE ''OLDFILE''',T30,'EXISTING INPUT FILE IS USED'/
     4 T10, '2. TYPE ''NEWFILE''',T30, 'INPUT NEW DATA AS REQUESTED'//)
C
```

```
C
      READ (IC,906) FILE
  906 FORMAT (2A)
C
C
C
C
      IF ( FILE .EQ. 'OLDFILE' ) RETURN
      IF ( FILE .EQ. 'NEWFILE' ) GO TO 14
C
         WRITE (IC,907) FILE
  907    FORMAT (1H0,'INVALID INPUT FILE DESIGNATION' ,T35,'******',A5,
     2             '******'/6X,'VALID DESIGNATIONS ARE*',
     3             T35,'*****OLDFILE*****'/
     4             T35,'*****NEWFILE*****'//)
        STOP
C
C
C
   14 CONTINUE
C
C
C             SET UP OUTPUT FILE TO RECIEVE TRANSLATED INPUT DATA
C
C
C00000000000000000000.........OPEN.........00000000000000000000
C
      OPEN (UNIT=IREAD,DEVICE='DSK',ACCESS='SEQOUT',FILE='DCCTMP.DAT')
C
C00000000000000000000.........OPEN.........00000000000000000000
C
C
C
C
C
C
C
      WRITE (IC,920)
  920 FORMAT (1H0,'INPUT DEBUG INFO -  NBUG, (DEBUG(I),I=1,NBUG)')
C
C
      READ (IC,20) NBUG, ( DEBUG(I), I = 1,NBUG )
   20 FORMAT ( I, 7A/(7A))
C
C
C
      WRITE ( IREAD,25 ) NBUG, ( DEBUG(I),I=1,NBUG)
   25 FORMAT ( I5, 5X, 7(A8,2X)/(10X ,7(A8,2X)))
C
C
      WRITE (IC,930)
  930 FORMAT (1H0,'INPUT BEGINNING DAY, MONTH, YEAR (4 DIGITS IN ',
     $    'YEAR AND ENDING DAY, MONTH, YEAR')
C
C
```

```
      READ (IC,30) IDAY,IMO,IYEAR,LDAY,LMO,LYEAR
   30 FORMAT (6I)
C
C
C
      WRITE (IREAD,35) IDAY, IMO, IYEAR, LDAY, LMO, LYEAR
   35 FORMAT (2(3X,I2,3X,I2,1X,I4))
C
C
C
C
      WRITE (IC,950)
  950 FORMAT (1H0,'INPUT STATION LAT-LONG IN DMS AND TIME ZONE OF '
     1                   'STATION')
C
C
      READ (IC,50)  (A(I),I=1,3), (B(I),L=1,3), TZONE
   50 FORMAT ( 6F, A)
C
C
C
      WRITE (IREAD,55) (A(I),I=1,3), (B(I),I=1,3), TZONE
   55 FORMAT (2(3F5.2,5X),T51,A8)
C
C
      WRITE (IC,960)
  960 FORMAT (1H0,' INPUT EP, ET, W')
C
C
      READ (IC,60) EP, ET, W
   60 FORMAT (3F)
C
C
C
      WRITE (IREAD,65) EP, ET, W
   65 FORMAT ( 3F5.2)
C
C
      WRITE (IC,70)
   70 FORMAT (1H0,' INPUT DATA CHECK VALUES'/
     $ ' TEMP LOWER BOUND, TEMP UPPER BOUND, WIND SPEED UPPER  BOUND')
C
      READ (IC,60) TLB, TUB, WUB
C
      WRITE (IREAD,970) TLB, TUB, WUB
  970 FORMAT  (3F5.0)
C
CCCCCCCCCCCCCCCCCCCCCC.........CLOSE.......CCCCCCCCCCCCCCCCCCC
C
      CLOSE (UNIT=IREAD,DEVICE='DSK',ACCESS='SEQOUT',FILE='DCCTMP.DAT')
C
CCCCCCCCCCCCCCCCCCCCCC.........CLOSE.......CCCCCCCCCCCCCCCCCCC
C
C
      RETURN
```

```
      END
C
C
C.................................................................
C
      SUBROUTINE DATE1
C
C         DATE1 INITIALIZES THE DATE COUNTERS.
C         JULIAN DATES ARE USED.
C
C
C         IYR    ... INITIAL YEAR
C         IMO    ... INITIAL MONTH
C         IDAY   ... INITIAL DAY
C         LYR    ... LAST YEAR
C         LMO    ... LAST MONTH
C         LDAY   ... LAST DAY
C
C         JULDAT... CURRENT JULIAN DATE
C         JBEGIN... JULIAN DATE AT BEGINNING OF RUN
C         JULEND... JULIAN DATE AT END OF RUN
C         JRANGE... LENGTH OF RUN
C         JULREF... JAN 1 OF INITIAL YEAR
C         JULREL... JULIAN DATE RELATIVE TO JAN 1 OF CURRENT YEAR
C         JSTART... RELATIVE JULIAN DATE TO BEGIN MONTHLY PARAMETER
C                   ESTIMATION RANGE
C         JSTOP ... RELATIVE JULIAN DATE TO END MONTHLY PARAMETER
C                   ESTIMATION RANGE
C         JYEAR ... YEAR COUNTER
C
C         NXLPYR... JULIAN DATE OF DEC 31 OF NEXT LEAP YEAR
C
C
C
      COMMON /DATES/ IYR, IMO, IDAY, LYR, LMO, LDAY
      COMMON /JDATES/ JULDAT, JULREL, JBEGIN, JULEND, JRANGE, NXLPYR
     $                ,JSTART, JSTOP,  JREND,  JYEAR
      COMMON /IO/    IREAD, IWRITE, IWBUG, IPARM, IBCOE
C
      INTEGER IDBUG
C
C         SET DEBUG FLAG
C
          IDBUG = 0
C
C
C
C         DETERMINE INITIAL JULIAN DATES
C
      CALL JULIAN (IMO,IDAY,IYR,JBEGIN)
      CALL JULIAN (LMO,LDAY,LYR,JULEND)
      CALL JULIAN (  1,   1,IYR,JULREF)
C
C
      JULREL = JBEGIN - JULREF
```

```
      JRANGE = JULEND - JBEGIN + 1
      JULDAT = JBEGIN - 1
C
C         DETERMINE THE NEXT OCCURANCE OF 12/31/(LEAP YEAR)
C                 (IE. THE 366TH DAY OF THE YEAR)
C
C
      LASTLP = IYR - MOD(IYR,4)
C
      CALL JULIAN (12,31,LASTLP,NXLPYR)
C
      IF (JULDAT .GE. NXLPYR) NXLPYR = NXLPYR + 1461
C
C
C         NOTE... 1461 = 365 + 365 + 365 + 366
C
C
C     THIS SECTION DEFINES VARIABLES NEEDED FOR MONTHLY
C     PARAMETER ESTIMATION
C
        JYEAR = IYR
        JSTART = JULREL + 1
        CALL JULIAN ( LMO, LDAY, IYR, JDATE )
        JSTOP = JDATE - JULREF + 1
        JREND = JSTOP
          IF (JYEAR - MOD(JYEAR,4)) 65,70,65
   70     IF (IMO.EQ.2 .AND. IDAY.EQ.28) JREND = JSTOP + 1
   65   CONTINUE
C
C
      ENTRY DATE
C
C         THE NEXT SECTION IS USED EACH DAY TO UPDATE
C            THE JULIAN DATE COUNTERS.
C
      JULREL = JULREL + 1
      JULDAT = JULDAT + 1
C
C
C         CHECK FOR END OF YEAR
C
      IF (JULREL .LE. 365) GO TO 100
      IF (JULREL .GT. 366) GO TO 200
C
C         CHECK FOR LEAP YEAR
C
      IF (JULDAT.NE.NXLPYR) GO TO 200
C
C         YES, THERE ARE 366 DAYS THIS YEAR.
C         UPDATE NXLPYR TO NEXT LEAP YEAR.
C
      NXLPYR = NXLPYR + 1461
C
C
      IF( IDBUG .NE. 0 ) GO TO 900
```

```
C
C
   50 CONTINUE
C
C
      RETURN
C
C
C        RESET RELATIVE JULIAN DATE
C
  200 JULREL = 1
C
C
  100 CONTINUE
         IF( IDBUG .NE. 0 ) GO TO 900
  110 RETURN
C
C
      ENTRY DATEM
C
C     THIS SECTION IS USED EACH DAY TO UPDATE THE JULIAN DATE COUNTERS
C     IF MONTHLY PARAMETER ESTIMATION IS USED.
C
        JULREL = JULREL + 1
        JULDAT = JULDAT + 1
C
        IF (JULREL .LE. JREND) GO TO 400
C
C     UPDATE THE JULIAN COUNTERS
C
        JULREL = JSTART
        JYEAR  = JYEAR + 1
        CALL JULIAN (IMO,IDAY,JYEAR,JULDAT)
        JREND = JSTOP
        IF(JYEAR - MOD(JYEAR,4)) 400,410,400
  410   IF (LMO.EQ.2 .AND. LDAY.EQ.28) JREND = JREND + 1
  400   CONTINUE
C
        IF ( IDBUG .NE. 0 ) GO TO 900
C
C
        RETURN
C
C
C
  900 CONTINUE
C
C        DEBUG INFORMATION FOR JULIAN DATE CALCULATIONS
C
      WRITE (IWBUG,920) JULDAT,JULREL,JBEGIN,JULEND,JRANGE,NXLPYR
     $                  ,JSTART,JSTOP,JREND,JYEAR
  920 FORMAT (1H ,'JULDAT=',I10,3X,'JULREL=',I10,3X,'JBEGIN=',I10,3X,
     $         /2X,'JULEND=',I10,3X,'JRANGE=',I10,3X,'NXLPYR=',I10,
     $         /2X,'JSTART=',I10,3X,'JSTOP =',I10,3X,'JREND =',I10,
     $          3X,'JYEAR=',I10)
```

```
C
      RETURN
C
C
      END
C
C.................................................................
C
      SUBROUTINE DATT(IDATE,IMO,IDAY,IYR)
C
C     CONVERT JULIAN DATE TO CALENDER DATE
C
      INTEGER CAL(12,2)
      DATA CAL/0,31,59,90,120,151,181,212,243,273,304,334,
     1  0,31,60,91,121,152,182,213,244,274,305,335 /
      I1=(IDATE-1)/1461
      I2=IDATE-(I1*1461)
C
C       1<=I2<=1461
C
      IF(I2.LE.365) GO TO 10
      IF(I2.LE.730) GO TO 20
      IF(I2.LE.1095) GO TO 30
      I3=3
      I4=I2-1095
      GO TO 40
   10 I3=0
      I4=I2
      I4=I2-365
      GO TO 40
   20 I3=1
      GO TO 40
   30 I3=2
      I4=I2-730
   40 IYR=1900+I3+(4*I1)
      INDX=1
      IF(I3.EQ.3)INDX=2
      DO 100 I=2,12
      IF(I4.LE.CAL(I,INDX)) GO TO 200
  100  CONTINUE
       IMO=12
      IDAY=I4-CAL(12,INDX)
       RETURN
  200 IMO=I-1
      IDAY=I4-CAL(I-1,INDX)
      RETURN
      END
C
C
C.................................................................
C
      SUBROUTINE JULIAN(MO,DA,YR,ANS)
      INTEGER ANS,CAL(12),DA,YR
      DATA CAL /31,28,31,30,31,30,31,31,30,31,30,31/
C
```

```
C         COMPUTE JULIAN DATE FROM JAN. 1, 1973
C
      ANS=0
      I=YR-1900
      ANS=ANS+365*I
      CAL(2)=28
      IF(MOD(YR,4).EQ.0) CAL(2)=29
      J=MO-1
      IF(J.EQ.0) GO TO 20
      DO 10 I=1,J
      ANS=ANS+CAL(I)
   10 CONTINUE
   20 CONTINUE
      ANS=ANS+DA
      RETURN
      END
C
C.......................................................................
C
      FUNCTION DMS(A)
C
C     FUNCTION DMS CONVERTS ANGLES EXPRESSED IN
C     DEGREES, MINUTES AND SECONDS TO RADIANS
C
      DIMENSION A(3)
      REAL MINUTE
C
C
      DEGREE = A(1)
      MINUTE = A(2)
      SECOND = A(3)
C
      DMS = DEGREE*3.14159/180. + MINUTE*3.14159/180./60.
     1      + SECOND*3.14159/180./60./60.
C
      RETURN
      END
C
C.......................................................................
C
      FUNCTION TAU(ST)
C
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /IO/ IREAD, IWRITE, IWBUG, IPARM, IRCOE
      COMMON /DBUG/ NBUG,DEBUG
      REAL*8    ITAU,DEBUG(20)
      DATA ITAU /'TAU'/
C
C
C     THETAS = LONGITUDE OF STANDARD MERIDIAN (RADIANS)
C              75TH MERIDIAN FOR EASTERN STANDARD TIME
C              90TH MERIDIAN FOR CENTRAL STANDARD TIME
C              105TH MERIDIAN FOR MOUNTAIN STANDARD TIME
C              120TH MERIDIAN FOR PACIFIC STANDARD TIME
C     THETAL = LONGITUDE OF OBSERVERS MERIDIAN (RADIANS)
```

```
C      TAU      = LOCAL HOUR ANGLE
C      ST       = STANDARD TIME IN THE TIME ZONE OF THE
C                 OBSERVER IN HOURS COUNTED FROM
C                 MIDNIGHT (EG. 0.00 TO 24.00)
C      EP       = +1 FOR EAST LONGITUDE, -1 FOR WEST LONGITUDE
C      ET       = DIFFERENCE BETWEEN TRUE SOLAR TIME
C                 AND MEAN SOLAR TIME (USUALLY NEGLECTED
C                 FOR HEAT TRANSFER COMPUTATIONS)
C
C
C      FUNCTION SUBROUTINE TAU CONVERTS THE OBSERVERS
C      STANDARD TIME TO LOCAL HOUR ANGLE IN RADIANS
C
C      OBTAIN TIME DIFFERENCE BETWEEN STANDARD MERIDIAN AND
C      OBSERVERS MERIDIAN (HOURS)
C
       DTSL = EP*(THETAS - THETAL)* 12.0/3.14159
C
C      COMPUTE OBSERVERS HOUR ANGLE (RADIANS). E = +1 FOR
C      MORNING AND E = -1 FOR AFTERNOON (I.E. SOLARNOON)
C
       IF (ST.GT.12. + DTSL -ET) E = -1.0000
       IF (ST.LE.12. + DTSL -ET) E = +1.0000
C
       TAU = (ST + E*12. - DTSL + ET) * 3.14159/12.0
C
       IF (TAU.GT.6.283185) TAU = TAU - 6.283185
       IF (TAU.LT.0.0) TAU = TAU + 6.283185
C
C
C      DEBUG OPTION
C
       IF (NBUG.EQ.0) GO TO 100
       DO 200 I = 1,NBUG
C
       IF (DEBUG(I).NE.ITAU) GO TO 200
       WRITE (IWBUG,250)     ST,PHI,THETAS,THETAL,EP,ET,W,DTSL,TAU
  250  FORMAT (////1H ,'FUNCTION TAU',  2X,
      1 'ST =',F6.3,2X,'PHI =',F6.3,2X,'THETAS =',F6.3,2X,'THETAL ='
      2 F6.3,2X,'EP =',F6.3,2X,'ET =',F6.3,2X,'W =',F6.3,2X,
      3 'DTSL =',F6.3,2X,'TAU =',F6.3)
  200  CONTINUE
C
  100  CONTINUE
C
       RETURN
       END
C
C.................................................................
C
       SUBROUTINE PRMEST ( TPRIME, KBAROB, CLOUD, GRTEMP, WSPEED,
      $                      WDIR,  IRANK, XXT, XY)
C
C
C         PRMEST IS THE CONTROLLING SUBROUTINE FOR THE PARAMETER ESTIMATI
```

```
C           FOR THE TEMPERATURE PROJECTION ROUTINE
C           BASED ON TODAY'S OBSERVED TEMPS., CLD COVER, ETC, PRMEST
C           DETERMINES THE PARAMETERS NECESSARY TO PROJECT
C           TEMPERATURES FOR TOMORROW.
C
C
      DIMENSION TPRIME(1), XXT(8,8), XY(8)
      DIMENSION KBAROB(1), CLOUD(1), GRTEMP(1)
      DIMENSION WSPEED(1), WDIR(1)
      REAL KBAR, KBAROB
      INTEGER RANGE, IDBUG
C
C
      COMMON /SUN/    DELTA, DTSL, SR, SS
      COMMON /ORBIT/  PHI, THETAS, THETAL, EP, ET,W
      COMMON /JDATES/ JULDAT, JULREL, JBEGIN, JELEND, JRANGE, NXLPYR
     $                ,JSTART, JSTOP,  JREND,  JYEAR
      COMMON /DATES/  IYR, IMO, IDAY, LYR, LMO, LDAY
      COMMON /IO/     IREAD, IWRITE, IWBUG, IPARM, IPCOE
C
C
C        SET DEBUG FLAG
C
            IDBUG = 0
C
C
C
C
C
C        COMPUTE THE ANGLE ADJUSTMENT BETWEEN THE
C        STANDARD MERIDIAN OF THE OBSERVER'S TIME ZONE AND THE
C        OBSERVER'S LOCAL MERIDIAN.
C
      DTSL = EP*(THETAS-THETAL)*3.81972
C
C
C        COMPUTE THE DECLINATION OF THE SUN
C
      CALL DECL (JULREL,DELTA, SR,SS)
C
C
C        DETERMINE THE LIMITS OF INTEGRATION FOR THE TEMPERATURE
C           GENERATION ALGORITHM
C
      CALL LIMITS (DTSL, SR,SS,T0,RHO,T12,SIGMA,T23)
         IF ( IDBUG .EQ.0 ) GO TO 951
            WRITE(IWBUG,950)DELTA,DTSL,T0,RHO,SR,T12,SIGMA,SS,T23,PHI
  950          FORMAT (T2,'SUBROUTINE PRMEST'/T2,'DELTA=',1X,F10.4,
     $               T20,'DTSL =',1X,F10.4,T40,'T0    =',1X,F10.4,
     $               T60,'RHO  =',1X,F10.4,T80,'SR    =',1X,F10.4/
     $               T2 ,'T12  =',1X,F10.4,T20,'SIGMA=',1X,F10.4,
     $               T40,'SS   =',1X,F10.4,T60,'T23   =',1X,F10.4,
     $               T80,'PHI  =',1X,F10.4)
  951    CONTINUE
C
```

```
C
C
C
C         INITIALIZE THE STANDARD TIME COUNTER.
C
      ST = 0.0
C
C
C         BEGIN LOOP TO ANALYZE TODAY'S TEMPERATURES.  COMPUTE
C            THE SET OF COEFFICIENTS FOR THE TEMPERATURE PROJECTIONS.
C
      DO 200 I = 2,25
C
         TMPLAG = TPRIME(I-1)
         KBAR   = KBARCB(I)
         TGD    = GRTEMP(I)
         CLD    = CLOUD(I)
         WSP    = WSPEED(I)
         WDR    = WDIR(I)
C
C            COMPUTE THE PREDICTORS X1, X2, X3, X4, X5, X6, X7
C
         CALL X1X2X3 (ST,TMPLAG,PHI,DELTA,TG,SR,T12,SS,T23,RHO,
     $      SIGMA, CLD, KBAR, TGD, WSP, WDR, X1,X2,X3,X4,X5,X6,X7,
     $      RANGE, T)
C
C
         IF ( IDBUG .EQ. 0 ) GO TO 961
            WRITE(IWBUG,960)ST,TMPLAG,PHI,X1,X2,X3,X4,X5,X6,X7,RANGE
  960    FORMAT (T2,'ST=',F5.0,'   TMPLAG=',F5.1,'   PHI=',F10.4,
     $           '   X=',7(1X,F10.5),'   RANGE=',I2)
  961    CONTINUE
C
C           DETERMINE THE HOURLY TEMPERATURE CHANGE
C
         Y = TPRIME(I) - TPRIME(I-1)
C
C
C           UPDATE REGRESSION MATRIX AND VECTOR
C
         CALL REGRES (Y,X1,X2,X3,X4,X5,X6,X7,XY,IRANK,XXT)
          GO TO 830
         IF(ST.LT.22.5) GO TO 830
      WRITE(5,850)JULDAT,((XXT(IZ,IY),IY=1,IRANK),IZ=1,IRANK)
  850      FORMAT(1X,7HJULDAT=,1X,I7/7(1X,7(E11.4)/))
  840      CONTINUE
  830    CONTINUE
C
            IF ( IDBUG .EQ. 0 ) GO TO 962
            WRITE (IWBUG,965) ST,T,Y,TPRIME(I),TPRIME(I-1),X1,X2,X3
     $                  ,X4,X5,X6,X7
  965       FORMAT('ST=',F4.0,2X,'T=',F7.3,2X,'Y=',F7.3,' TPI=',F7.3,2X,
     $         'TPI1=',F7.3,2X,'X1=',F7.3,2X,'X2=',E12.5,2X,'X3=',E12.5,
     $         2X,'X4=',E7.2,2X,'X5=',E7.2,2X,'X6=',E7.2,2X,'X7=',F7.2)
  962       CONTINUE
```

```
C
            IF ( IDBUG .EQ. 0 ) GO TO 916
C
C           DEBUG STATEMENTS....CHECK  MATRIX OPERATION RESULTS
C
      DO 910 II = 1,IRANK
  910 WRITE (IWBUG,900) (XXT(II,J),J=1,IRANK)
  900 FORMAT (1X/' XXT=',1X,6(F12.5,3X))
      WRITE (IWBUG,903) (XY(J),J=1,IRANK)
  903 FORMAT (1X/'  XY=',1X,6(F12.5,3X))
C
  916 CONTINUE
C
C
C           UPDATE THE STANDARD TIME COUNTER
C
         ST = ST + 1.0
C
C
  200 CONTINUE
C
      RETURN
      END
C
C
C.................................................................
C
      SUBROUTINE COEF (IRANK, XXT, XY, ACOEF )
C
C
C        SUBROUTINE COEF DETERMINES THE REGRESSION COEFFICIENTS
C           REQUIRED FOR THE TEMPERATURE MODEL.
C
C
      DIMENSION LWORK(8), MWORK(8), A(64)
      DIMENSION XXT(8,8), XY(8), ACOEF(8) ,XXTINV(8,8)
C
      INTEGER RANGE, IDBUG
C
      COMMON /IO/  IREAD, IWRITE, IWBUG, IPARM, IBCOE
      COMMON /JDATES/ JULDAT, JULREL, JBEGIN, JULEND, JRANGE, NXLFYR
     $                 ,JSTART, JSTOP,  JREND,  JYEAR
      COMMON /YSTAT/ YSUM, YSUMSQ, YMEAN, RSQUAR
C
C
C        SET DEBUG FLAG
C
      IDBUG = 0
C
C
C
C
C              WHEN IRANK IS LESS THAN THE PROGRAM DIMENSIONS
C              FOR XXT, PROBLEMS WILL OCCUR WHEN XXT IS INVERTED.
C              THESE PROBLEMS ARISE DUE TO THE WAY DATA IS STORED
```

```
C               IN MATRIX FORM.  THE SOLUTION IS TO CONVERT XXT(J,I)
C               TO VECTOR FORM, A(L).
C
C
C
C
C
C
C
                DO 205 I = 1,IRANK
                DO 205 J = 1,IRANK
C
                 JJ = (I-1)*IRANK + J
                 A(JJ) = XXT(J,I)
C
          IF(IDBUG.EQ.1) WRITE (IWBUG,204)  J,I,JJ,A(JJ),XXT(J,I)
  204     FORMAT (1H0,3I5, 2(2X,E12.5))
  205     CONTINUE
C
C
C
C         INVERT THE REGRESSION MATRIX, XXT.
C
      CALL MINV (A,IRANK,DXXT,LWORK, MWORK)
C
C
C
C
C
                DO 305 I = 1,IRANK
                DO 305 J = 1,IRANK
  305           XXTINV(J,I) = A((I-1)*IRANK + J)
C
C
      IF ( IDBUG .EQ. 0 ) GO TO 921
      DO 920 I = 1,IRANK
  920 WRITE (IWBUG,904) (XXTINV(I,J),J=1,IRANK)
  904 FORMAT (1X/' XXTINV=',2X,6(F12.5,3X))
C
  921 CONTINUE
C
C
C         DETERMINE THE REGRESSION COEFFICIENTS
C
      CALL MATMLT (XY,XXTINV,ACOEF,IRANK)
C
C
C SINCE PREDICTER X5 IS NOT BEING USED, THE ELEMENTS OF ACOEF
C HAVE BEEN REARRANGED SLIGHTLY (SEE SUBROUTINE X1X2X3).
C NOW REORDER ACOEF.
C
          ACOEF(8) = ACOEF(7)
          ACOEF(7) = ACOEF(6)
          ACOEF(6) = 0.0
C
```

```
      AXY = 0
C
C
C
C
C         COMPUTE MULTIPLE COEFFICIENT OF DETERMINATION
C
      DO 400 I = 1,8
  400 AXY = ACOEF(I)*XY(I) + AXY
C
      XXTYY = XXT(1,1)*YMEAN*YMEAN
C
      RSQUAR = (AXY-XXTYY)/(YSUMSQ-XXTYY)
C
C
      IF ( IDBUG .EQ. 1 ) GO TO 919
      WRITE (IWBUG,918) JULREL, DXXT, (ACOEF(IJ),IJ=1,IRANK)
  918 FORMAT (1H ,'JULREL=',I5,5X,'DET. OF XXT=',E12.5/
     $     '   ACOEFF=',8(2X,E11.4))
C
      WRITE (IWBUG,930) AXY,XXTYY,XXT(1,1),YSUMSQ
  930 FORMAT (1H ,3HAXY ,1X,E12.5,5X,5HXXTYY,1X,E12.5,5X,
     $          8HXXT(1,1) ,1X,E12.5,5X,6HYSUMSQ,1X,E12.5)
  919 CONTINUE
C
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE DECL (RJD,DELTA,SR,SS)
      INTEGER RJD
      COMMON /ORBIT/ PHI,THETAS,THETAL,EP,ET,W
      COMMON /IO/ IREAD, IWRITE, IWBUG, IPARM, IBCOE
      COMMON /DBUG/ NBUG,DEBUG
      REAL*8    IDECL,DEBUG(20)
      DATA IDECL/'DECL'/
C
C     DELTA  = DECLINATION OF THE SUN (RADIANS)
C     PHI    = OBSERVERS LATITUDE (RADIANS)
C     THETAS = LONGITUDE OF STANDARD MERIDIAN (RADIANS)
C              75TH MERIDIAN FOR EASTERN STANDARD TIME
C              90TH MERIDIAN FOR CENTRAL STANDARD TIME
C              105TH MERIDIAN FOR MOUNTAIN STANDARD TIME
C              120TH MERIDIAN FOR PACIFIC STANDARD TIME
C     THETAL = LONGITUDE OF OBSERVERS MERIDIAN (RADIANS)
C     RJD    = RELATIVE JULIAN DATE (I.E. WITH RESPECT TO JAN 1)
C     ST     = STANDARD TIME IN THE TIME ZONE OF THE OBSERVER
C              IN HOURS COUNTED FROM MIDNIGHT (E.G.0.00 TO 24.00)
C     EP     = +1 FOR EAST LONGITUDE, -1 FOR WEST LONGITUDE
C     ET     = DIFFERENCE BETWEEN TRUE SOLAR TIME AND
C              MEAN SOLAR TIME (USUALLY NEGLECTED FOR
C              HEAT TRANSFER COMPUTATIONS)
C
C COMPUTE TIME DIFFERENCE BETWEEN STANDARD MERIDIAN AND
```

```
C OBSERVERS MERIDIAN (HOURS)
C
      DTSL    = EP*(THETAS - THETAL)* 3.81972
C
C COMPUTE DECLINATION OF THE SUN (RADIANS)
C
      DELTA   = 0.4093*COS(0.0172*(172. - FLOAT(NJD)) )
C
C COMPUTE HOUR ANGLE AT SUNSET (RADIANS)
C
C
      TSS     = ACOS(-TAN(DELTA)*TAN(PHI))
C     COMPUTE STANDARD TIME OF SUNST (HOURS)
C
      SS      = TSS*3.81972 + 12. +DTSL -ET
C
C COMPUTE HOUR ANGLE OF SUNRISE (RADIANS)
C
      TSR     = 6.283185 - TSS
C
C     COMPUTE  STANDARD TIME OF SUNRISE (HOUR)
C
      SR      = TSR*3.81972 -12. + DTSL -ET
C
C
C     CONVERT SUNRISE IN STANDARD TIME TO LOCAL TIME
C
      SR = SR - DTSL
C
C     CONVERT SUNSET IN STANDARD TIME TO LOCAL TIME
C
      SS = SS - DTSL
C
C
C     DEBUG OPTION
C
      IF (NBUG.EQ.0) GO TO 300
      DO 100 I = 1,NBUG
C
      IF (DEBUG(I).NE.IDECL) GO TO 100
      WRITE (IWBUG,200)  RJD,DTSL,DELTA,TSS,SS,TSR,SR
  200 FORMAT (////,1H ,'SUBROUTINE DECL ','****',' RJD =',
     1 I5,'  DTSL =',F6.3,'  DELTA =',F6.3,'  TSS =',F6.3,
     2 '  SS =',F6.3,2X,'TSR =',F6.3,2X,'SR =  ',F6.3)
  100 CONTINUE
C
  300 CONTINUE
C
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE LIMITS (DTSL,R,S,TO,RHO,T12,SIGMA,T23)
C     FIND LIMITS FOR TEMPERATURE INTEGRATION
```

```
C
C
      TO  = - DTSL
      T23 = 23.00 - DTSL
C
      IF (DTSL.LT.0.0) GO TO 50
C
C     FIND LIMITS OF INTEGRATION WHEN OBSERVER IS
C     WEST OF THE STANDARD MERIDIAN
C
C     FOR SUNRISE
C
      RHO = AINT(R+1.0) - DTSL
      IF (RHO .LT. R) RHO = RHO + 1
C
C     FOR SUNSET
C
      SIGMA = AINT(S+1) - DTSL
      IF (SIGMA .LT. S) SIGMA = SIGMA + 1
C
C     FOR LOCAL NOON
C
      T12 = 13.0 - DTSL
C
      GO TO 75
C
   50 CONTINUE
C
C      FIND LIMITS OF INTEGRATION WHEN OBSERVER
C      IS EAST OF THE STANDARD MERIDIAN
C
      RHO = AINT(R) - DTSL
      IF (RHO .LT. R) RHO = RHO + 1
C
      SIGMA = AINT(S) - DTSL
      IF (SIGMA .LT. S) SIGMA = SIGMA + 1
C
      T12 = 12.0 - DTSL
C
   75 CONTINUE
C
      RETURN
      END
C
C.................................................................
C
      SUBROUTINE X1X2X3 ( ST, TMPLAG, PHI, DELTA, TO, R, T12, S, T23,
     $        RHO, SIGMA, CLD, KBAR, TGD, WSP, WDR, X1,X2,X3,X4,X5,X6,X7,
     $        RANGE,T)
C
C
C         COMPUTE THE PREDICTERS X1,....,X7.
C
C
C         ST      ... LOCAL STANDARD TIME
```

```
C         T         ... LOCAL TIME
C         TMPLAG    ... TEMPERATURE AT PREVIOUS TIME PERIOD
C         PHI       ... STATION LATITUDE (RADIANS)
C         DELTA     ... DECLINATION (RADIANS)
C         R         ... LOCAL SUNRISE
C         S         ... LOCAL SUNSET
C         TO        ... VALUE OF T AT LOCAL STANDARD MIDNIGHT
C         RHO       ... FIRST OBSERVATION HOUR AFTER SUNRISE
C         T12       ... FIRST OBSERVATION HOUR AFTER LOCAL NOON
C         SIGMA     ... FIRST OBSERVATION HOUR AFTER SUNSET
C         T23       ... VALUE OF T AT LOCAL STANDARD 2300 HR (IE  11 PM)
C
      INTEGER RANGE, IDBUG
      INTEGER SWICH1, SWICH2
      REAL KBAR
      COMMON /SWITCH/ SWICH1, SWICH2
      COMMON /IO/ IREAD, IWRITE, IWBUG, IPARM, IBCOE
C
C
      SWICH1 = 1
      SWICH2 = 0
C
C
C
      A  = 0.0005
      PI = 3.14159
      IDBUG = 0
C
C
C        CONVERT STANDARD TIME TO LOCAL TIME
C
      T = TAU(ST)*(12.0/PI) - 12.0
C
         IF ( T .GT. 24.0 ) T = T - 24.0
         IF ( T .LT.  0.0 ) T = T + 24.0
C
C
C
C        X1 IS JUST THE LAG-1 TEMPERATURE
C
      X1 = TMPLAG
C
C
C        DETERMINE THE APPROPRIATE RANGE FOR X2 AND X3
C
C
C     ... BEFORE SUNRISE         ...
C
      IF ( TO .LE. T  .AND.  T .LT. R ) GO TO 100
C
C
C     ... SUNRISE                ...
C
      IF ( RHO- A .LE. T  .AND.
     1     RHO+ A .GE. T          ) GO TO 200
```

```
C
C
C     ... MORNING HOURS          ...
C
      IF ( RHO+A .LE. T  .AND.  T .LE. 12 ) GO TO 300
C
C
C     ... NOON                   ...
C
      IF ( T12- A .LE. T  .AND.
     1     T12+ A .GE. T         ) GO TO 400
C
C
C     ... AFTERNOON HOURS        ...
C
      IF ( T12+ A .LE. T  .AND.  T .LT. S ) GO TO 500
C
C
C     ... SUNSET                 ...
C
      IF ( SIGMA- A .LE. T  .AND.
     1     SIGMA+ A .GE. T       ) GO TO 600
C
C
C     ... EVENING HOURS          ...
C
      IF ( SIGMA+A .LE. T  .AND.  T .LE. T23 ) GO TO 700
C
C
C----------------------------------------------------------------------
C
  100 X2 = 0.0
      X3 = 0.0
      RANGE = 1
      GO TO 900
C
C----------------------------------------------------------------------
C
  200 A = PI*R/12.0
      B = PI*RHO/12.0
C
      X2 = (RHO-R)*SIN(PHI)*SIN(DELTA)
      X2 =X2  - (12.0/PI)*COS(DELTA)*COS(PHI)*(SIN(B) - SIN(A) )
C
      X3 = COS(DELTA)*COS(PHI)*(COS(A)-COS(B))
      RANGE = 2
      GO TO 900
C----------------------------------------------------------------------
C
  300 A = PI*T/12.0
      B = PI*(T-1.0)/12.0
C
      X2 = SIN(DELTA)*SIN(PHI)
      X2 = X2 - (12.0/PI)*COS(DELTA)*COS(PHI)*(SIN(A)-SIN(B))
C
```

```
      X3 = COS(DELTA)*COS(PHI)*(COS(B)-COS(A))
      RANGE = 3
      GO TO 900
C
C-----------------------------------------------------------------------
C
  400 A = PI*T/12.0
      B = PI*(T-1.0)/12.0
      C = PI*(T12-1.0)/12.0
C
      X2 = SIN(DELTA)*SIN(PHI)
      X2 = X2 - (12.0/PI)*COS(DELTA)*COS(PHI)*(SIN(A)-SIN(B))
C
      X3 = COS(DELTA)*COS(PHI)*(COS(C)+1.0)
C
      RANGE = 4
      GO TO 900
C
C-----------------------------------------------------------------------
C
  500 A = PI*T/12.0
      B = PI*(T-1.0)/12.0
C
      X2 = SIN(DELTA)*SIN(PHI)
      X2 = X2 - (12.0/PI)*COS(DELTA)*COS(PHI)*(SIN(A)-SIN(B))
C
      X3 = 0.0
      RANGE = 5
      GO TO 900
C
C-----------------------------------------------------------------------
C
  600 A = PI*S/12.0
      B = PI*(SIGMA-1.0)/12.0
C
      X2 = (S-SIGMA+1.0)*SIN(DELTA)*SIN(PHI)
      X2 = X2 + (12.0/PI)*COS(DELTA)*COS(PHI)*(SIN(B)-SIN(A))
C
      X3 = 0.0
      RANGE = 6
      GO TO 900
C
C-----------------------------------------------------------------------
C
  700 X2 = 0.0
      X3 = 0.0
      RANGE = 7
C
C-----------------------------------------------------------------------
C
  900 CONTINUE
C
C
         IF ( SWICH1 .EQ. 0 ) GO TO 905
C
```

```
      X2 = KBAR*X2
      X3 = KBAR*X3
C
  905 X4 = 1.579E-8*(1.00+0.17*CLD**2.)*(TMPLAG+460.)**6.
      X5 = TGD*SWICH2
      X6 = WSP
      X7 = WDR
C
C
C
         IF ( IDBUG.EQ.0) RETURN
C
         WRITE (IWBUG,909) CLD, KBAR, TGD
  909    FORMAT (1H0, 'CLD=',E12.5,2X, 'KBAR=',E12.5,2X, 'TGD=',
     $           E12.5)
         WRITE(IWBUG,910) ST, X1, X2, X3, X4, XIWBUG, RANGE, T, A, B
  910    FORMAT(1X,F5.0,1X,5(E12.5,1X),I5,F10.5,2(1X,E12.5))
      RETURN
      END
C
C.............................................................
C
      SUBROUTINE REGRES ( Y, X1, X2, X3, X4, X5, X6, X7, XY, N, XXT )
C
C
C        SET UP THE VECTOR XY AND THE MATRIX XXT THAT ARE
C        NECESSARY TO ESTIMATE THE REQUIRED TEMPERATURE EQUATION
C        COEFFICIENTS.  THIS SUBROUTINE IS CALLED ONCE EACH
C        TIME PERIOD. (I.E. EVERY TIME THE TEMPERATURE CHANGE, Y,
C        IS COMPUTED)
C
C        Y......TEMPERATURE CHANGE IN LAST TIME PERIOD
C        X1.....PREDICTOR X1 IN THE TEMPERATURE MODEL
C        X2.....PREDICTOR X2 IN THE TEMPERATURE MODEL
C        X3.....PREDICTOR X3 IN THE TEMPERATURE MODEL
C        X4.....PREDICTOR X4 IN THE TEMPERATURE MODEL
C        X5.....PREDICTOR X5 IN THE TEMPERATURE MODEL
C        X6.....PREDICTOR X6 IN THE TEMPERATURE MODEL
C        X7.....PREDICTOR X7 IN THE TEMPERATURE MODEL
C        XY.....VECTOR OBTAINED BY MULTIPLYING THE PREDICTOR VALUES
C                  BY THE OBSERVED TEMPERATURE CHANGES. ( THE
C                  ELEMENTS OF XY ARE SUMMATIONS)
C        N......DIMENSION OF XY
C        XXT....MATRIX OBTAINED BY POSTMULTIPLYING THE VECTOR
C                  X BY ITS TRANSPOSE. (THE ELEMENTS OF XXT ARE
C                  SUMMATIONS)
C
      DIMENSION XY(8), XXT(8,8), X(8)
C
      COMMON /YSTAT/ YSUM, YSUMSQ, YMEAN, RSQUAR
C
C
C        PUT PREDICTOR VALUES IN PREDICTOR VECTOR
C
      X(1) = 1.0
```

```
      X(2) = X1
      X(3) = X2
      X(4) = X3
      X(5) = X4
      X(6) = X6
      X(7) = X7
      X(8) = X5
C
C8000 WRITE (5,9000) (X(J),J=1,8)
C9000 FORMAT (1H0,'REGRES X VECTOR',5X,4(E12.5,5X)/T22,4(E12.5,5X))
C
C
C
C
C        UPDATE THE XY VECTOR
C
      DO 100 I = 1,N
      XY(I) = X(I)*Y + XY(I)
  100 CONTINUE
C
C
C        UPDATE THE XXT MATRIX
C
      DO 200 I = 1,N
      DO 200 J = 1,N
      XXT(J,I) = X(J)*X(I) + XXT(J,I)
  200 CONTINUE
C
C
C        UPDATE Y STATISTICS
C
      YSUM   = Y + YSUM
      YSUMSQ = Y*Y + YSUMSQ
      YMEAN  = YSUM/XXT(1,1)
C
C
      RETURN
      END
C
C.......................................................................
C
C
C     ..................................................................
C
C        SUBROUTINE MINV
C
C        PURPOSE
C           INVERT A MATRIX
C
C        USAGE
C           CALL MINV(A,N,D,L,M)
C
C        DESCRIPTION OF PARAMETERS
C           A - INPUT MATRIX, DESTROYED IN COMPUTATION AND REPLACED BY
C               RESULTANT INVERSE.
```

```
C            N - ORDER OF MATRIX A
C            D - RESULTANT DETERMINANT
C            L - WORK VECTOR OF LENGTH N
C            M - WORK VECTOR OF LENGTH N
C
C         REMARKS
C            MATRIX A MUST BE A GENERAL MATRIX
C
C         SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C            NONE
C
C         METHOD
C            THE STANDARD GAUSS-JORDAN METHOD IS USED. THE DETERMINANT
C            IS ALSO CALCULATED. A DETERMINANT OF ZERO INDICATES THAT
C            THE MATRIX IS SINGULAR.
C
C      ..................................................................
C
      SUBROUTINE MINV(A,N,D,L,M)
      DIMENSION A(1),L(1),M(1)
C
C         ...............................................................
C
C         IF A DOUBLE PRECISION VERSION OF THIS ROUTINE IS DESIRED, THE
C         C IN COLUMN 1 SHOULD BE REMOVED FROM THE DOUBLE PRECISION
C         STATEMENT WHICH FOLLOWS.
C
C     DOUBLE PRECISION A,D,BIGA,HOLD
C
C         THE C MUST ALSO BE REMOVED FROM DOUBLE PRECISION STATEMENTS
C         APPEARING IN OTHER ROUTINES USED IN CONJUNCTION WITH THIS
C         ROUTINE.
C
C         THE DOUBLE PRECISION VERSION OF THIS SUBROUTINE MUST ALSO
C         CONTAIN DOUBLE PRECISION FORTRAN FUNCTIONS.  ABS IN STATEMENT
C         10 MUST BE CHANGED TO DABS.
C
C         ...............................................................
C
C         SEARCH FOR LARGEST ELEMENT
C
      D=1.0
      NK=-N
      DO 80 K=1,N
      NK=NK+N
      L(K)=K
      M(K)=K
      KK=NK+K
      BIGA=A(KK)
      DO 20 J=K,N
      IZ=N*(J-1)
      DO 20 I=K,N
      IJ=IZ+I
   10 IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20
   15 BIGA=A(IJ)
```

```
      L(K)=I
      M(K)=J
   20 CONTINUE
C
C        INTERCHANGE ROWS
C
      J=L(K)
      IF(J-K) 35,35,25
   25 KI=K-N
      DO 30 I=1,N
      KI=KI+N
      HOLD=-A(KI)
      JI=KI-K+J
      A(KI)=A(JI)
   30 A(JI) =HOLD
C
C        INTERCHANGE COLUMNS
C
   35 I=M(K)
      IF(I-K) 45,45,38
   38 JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-A(JK)
      A(JK)=A(JI)
   40 A(JI) =HOLD
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   45 IF(BIGA) 48,46,48
   46 D=0.0
      RETURN
   48 DO 55 I=1,N
      IF(I-K) 50,55,50
   50 IK=NK+I
      A(IK)=A(IK)/(-BIGA)
   55 CONTINUE
C
C        REDUCE MATRIX
C
      DO 65 I=1,N
      IK=NK+I
      HOLD=A(IK)
      IJ=I-N
      DO 65 J=1,N
      IJ=IJ+N
      IF(I-K) 60,65,60
   60 IF(J-K) 62,65,62
   62 KJ=IJ-I+K
      A(IJ)=HOLD*A(KJ)+A(IJ)
      IF(ABS(A(IJ)).GT.1.E 37.OR.ABS(A(IJ)).LT.1.E-37)WRITE(5,936)A(IJ)
  936  FORMAT(1X,'A(IJ)=',1X,E12.5)
   65 CONTINUE
```

```
C
C         DIVIDE ROW BY PIVOT
C
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF(J-K) 70,75,70
   70 A(KJ)=A(KJ)/BIGA
   75 CONTINUE
C
C         PRODUCT OF PIVOTS
          IF((ALOG10(D)+ALOG10(BIGA)).LT.37.0) GO TO 77
          IF (IFLAG.GT.0) GO TO 76
          IFLAG=1
C
          WRITE(5,800)
  800   FORMAT(///,1X,'SUBROUTINE MINV: DETERMINANT SIZE EXCEEDS ',/1X,
     $    'MACHINE CAPACITY. CALCULATION IS GREATER THAN 1.E+37. ',/1X,
     $    'PROCESSING CONTINUES'///)
      GO TO 76
   77 CONTINUE
      D=D*BIGA
      IF (D.GT.1.E 30)WRITE(5,935)D
  935 FORMAT(1X,'D=',1X,E12.5)
   76 CONTINUE
C
C         REPLACE PIVOT BY RECIPROCAL
C
      A(KK)=1.0/BIGA
   80 CONTINUE
C
C         FINAL ROW AND COLUMN INTERCHANGE
C
      K=N
  100 K=(K-1)
      IF(K) 150,150,105
  105 I=L(K)
      IF(I-K) 120,120,108
  108 JQ=N*(K-1)
      JR=N*(I-1)
      DO 110 J=1,N
      JK=JQ+J
      HOLD=A(JK)
      JI=JR+J
      A(JK)=-A(JI)
  110 A(JI) =HOLD
  120 J=M(K)
      IF(J-K) 100,100,125
  125 KI=K-N
      DO 130 I=1,N
      KI=KI+N
      HOLD=A(KI)
      JI=KI-K+J
      A(KI)=-A(JI)
  130 A(JI) =HOLD
```

```
      GO TO 100
  150 RETURN
      END
C
C.....................................................................
C
      SUBROUTINE MATMLT (A, B, C, N)
C
C
C
C        MATMLT POST MULTIPLIES AN  N X N  MATRIX BY A
C           VECTOR, A, OF LENGTH N. THE RESULT IS A VECTOR, C,
C           OF LENGTH N.
C
C
      DIMENSION A(8), B(8,8), C(8)
C
C
      DO 50 I = 1,N
   50 C(I) = 0.0
C
C
      DO 100 J = 1,N
      DO 100 I = 1,N
C
         C(I) = B(I,J)*A(J) + C(I)
C
  100 CONTINUE
C
C
      RETURN
      END
C.....................................................................
C
      SUBROUTINE ATOB (A,B)
C
C
C        DERIVE THE TEMPERATURE MODEL COEFICIENTS (I.E.
C        ELEMENTS OF VECTOR B) FROM THE REGRESSION VECTOR A
C
C
C        A.....REGRESSION COEFFICIENT VECTOR
C        B.....VECTOR OF TEMPERATURE MODEL COEFFICIENTS
C
C
C        A(1) = A0        B(1) = B0
C        A(2) = A1        B(2) = B1
C        A(3) = A2        B(3) = B2
C        A(4) = A3        B(4) = B3
C        A(5) = A4        B(5) = B4
C        A(6) = A5        B(6) = B5
C        A(7) = A6        B(7) = B6
C        A(8) = A7        B(8) = B7
C
C
```

```
C
      DIMENSION A(1),B(1)
C
C
C
C        DETERMINE B1
C
C
      B(2) = -ALOG(1.0+A(2))
C
C
C
      BA = B(2)/(-A(2))
C
C
C        DETERMINE BI'S
C
      DO 100 I = 1,8
C
         IF ( I .EQ. 2 ) GO TO 100
C
         B(I) = A(I)*BA
C
  100 CONTINUE
C
C
C8000 WRITE (5,900) (B(J),J=1,8)
C 900 FORMAT (1H0, 'B VECTOR', 5X,4(E12.5,5X)/T15,4(E12.5,5X))
C
C
C
      RETURN
      END
C
C
C..............................................................
C
      SUBROUTINE DCHECK (JULREL, ID, DATA, L )
C
      DIMENSION DATA(1), TYPE(4), FMT(15)
C
      DATA FMT /'(1H ,8HJULREL =,1X,I3,  ''XXXX DATA OUT OF BOUNDS'',
     $2X,E12.5,3X,''L ='',2X,I2)'/
C
      DATA TYPE(1)/'TEMP'/, TYPE(2)/'CLD'/, TYPE(3)/'WSP'/,
     $      TYPE(4)/'WDIR'/
C
C     WRITE (5,100) FMT
  100 FORMAT (1H ,10(1X,A5)/)
C
      FMT(6) = TYPE(ID)
C
      WRITE (5,FMT) JULREL, DATA(L), L
C
C
```

```
RETURN
END
```