APPLIED

STOCHASTIC

TECHNIQUES

PERSONAL COMPUTER VERSION 5.2

USER'S MANUAL

OCTOBER 1990

APPLIED STOCHASTIC TECHNIQUES

(LAST Personal Computer Package Version 5.2)

USER MANUAL
October 1990

APPLIED STOCHASTIC TECHNIQUES
(Personal Computer Version)

USER MANUAL

By

Dr. William L. Lane
Dr. Donald K. Frevert
Earth Sciences Division
Denver Office
Bureau of Reclamation
U.S. Department of the Interior

Denver, Colorado
October 1990

Key words: Hydrology, Statistics, Stochastic, Time Series

❧❧❧

Reprinted March 1998
Technical Service Center
Water Resources Services
Denver, Colorado

DISCLAIMER

This program is furnished by the Government and is accepted and used by the recipient upon the express understanding that the United States Government makes no warranties, express or implied, concerning the accuracy, completeness, reliability, usability, or suitability for any particular purpose of the information and data contained in this program or furnished in connection therewith, and the United States shall be under no liability whatsoever to any person by reason of any use made thereof.

The program herein belongs to the Government. Therefore, the recipient further agrees not to assert any proprietary rights therein or to represent this program to anyone as other than a Government program.

FOREWORD

The Bureau of Reclamation encourages comments and suggestions from users. Should a user detect any inaccuracies it is suggested that he/she copy the page (or pages) in error and indicate the problem. Please send all comments to:

Bureau of Reclamation
Technical Service Center
Attention: Code D-8510
Denver Federal Center
PO Box 25007
Denver CO 80225

or call: (303) 445-2473

CONTENTS

I. INTRODUCTION

In April of 1977, work was undertaken in the Engineering and Research Center, Bureau of Reclamation, to develop a general computerized approach for the application of Stochastic Techniques to hydrologic data. The work was directed and carried out by Dr. William Lane on Reclamation's main frame CDC Cyber Computer. The package name LAST is an acronym made up of the author's name and the title, Applied Stochastic Techniques.

In June of 1978, Dr. Stephen Burges of the University of Washington was hired to review this computer package (Burges, 1978). His helpful comments and constructive criticisms added greatly to this package and are gratefully acknowledged.

In 1980, Dr. Lane transferred to Reclamation's Pacific Northwest Regional Office and although he has subsequently returned to the Denver Office, responsibility for maintaining and improving the program now rests with Donald K. Frevert.

This document is intended primarily to provide guidance for the user of the programs included in this package. In addition to the information needed to run the programs and interpret the results, information of interest to programmers which will aid in any future modifications and improvements is included. Background material, including derivations of various equations and rationale for the techniques, is also included. Because of the multipurpose nature of this document, technical references will be made where appropriate.

A preliminary draft version of this documentation was written in the spring of 1978. It was then finalized in December of 1979. Revised editions have been written as new changes, additions, improvements, or corrections were completed. The personal computer version of the program was developed in 1988, 1989 and 1990 with the help of Mr. Fredrick Dockhorn and Mr. David Read. This manual summarizes the current status of that set of programs.

A. <u>Purpose</u>

This set of programs allows for the examination and analysis of seasonal and annual streamflow data at a number of interrelated stations, will determine the statistical properties of the data, will estimate the parameters required for a stochastic model, and will generate any number of synthetic streamflow sequences of any desired length. The historically observed statistical properties are estimated and then preserved in the generated sequences. Several options are available which allow the user to choose from various levels of complexity for the stochastic model. Guidance is provided to aid the user in making these choices. While the programs were developed with streamflow in mind, they could be applied to other variables, such as rainfall, evaporation, water quality, sediment, and crop water requirements, either alone or in combination. However, use of these programs for variables other than streamflow should only be undertaken with great caution and care. This package is not presently designed to be applicable to event dominated series such as daily rainfall or runoff.

Data may be in annual, semiannual, quarterly, monthly, or other seasonal periods desired subject to limitations which are described in Appendix E. The seasons need not be equally spaced within the year. The user, however, must provide data which are continuous, i.e., no missing data, and homogeneous. By homogeneous, we mean that the data must reflect the same conditions throughout their entire record. In addition, all stations must have the same number of seasons. No more than twelve seasons per year allowed.

In most cases, this means that historically recorded data must be adjusted to remove the effects of changes due to diversions, reservoirs, irrigation return flows, and any other factors which have altered the streamflow from that which would have occurred naturally.

In order to run these programs, the periods of record for the various stations of a multistation application must coincide exactly.

B. Approach

The essential attributes of the approach are as follows:

1. Ability to preserve year-to-year serial correlations with a multilag linear autoregressive model in addition to seasonal serial correlations.

2. Ability to preserve cross correlations on an annual basis.

3. Ability to generate "key" stations and to disaggregate those values into component substations on an annual basis.

4. Ability to likewise disaggregate annual values into seasonal values preserving both serial correlations and cross correlation between variables on a seasonal basis.

5. Ability to generate annual and seasonal values which come from distributions statistically indistinguishable from those observed historically. The degree to which generated data are to conform to the historically observed distribution is determined by the user. A very flexible approach is maintained with respect to the choice of distributions.

These attributes are achieved by the use of linear correlations and nonlinear transformations. Much of the methodology follows the form and appearance of approaches written on by Matalas (1967) and Valencia and Schaake (1973). Considerable extensions and modifications were made to these works, along with some original techniques developed specifically for this package. In this chapter, the approach will be covered in a general manner. The mathematical details are covered in detail in Chapter V, Mathematics. The ramifications of the various choices for the stochastic model are covered in Chapter III, Computational Options. The actual mechanics of running the programs are covered in Chapter II, Operation.

The main characteristic of this approach which differentiates it from other widely used approaches is the use of "key" and "substations." Key stations are stations of major importance, usually stations indicative of large portions of the basin whose flows are actually the summation of several substations. Key stations are analyzed and generated totally separate from the substations. Substations are analyzed and generated taking into account the intercorrelations between key and substations. In this way, the substations may easily and accurately be generated subject to the constraint that they add together properly to give a reasonable hydrologic trace at the key stations. Improvements to the LAST package now allow two level disaggregation so that a station can be a substation in one group and a key station in a second group. It should be noted that one is not required to use this attribute.

If this option is not used, the annual approach is reduced essentially to that of Matalas (1967).

In addition to the standard autoregressive model, an autoregressive moving average model [ARMA (1, 1)] is a planned addition for the generation of key station annual data. The ARMA model is of use mostly in cases exhibiting strong long-term persistence in the historical data.

From a user's viewpoint, the steps involved in using this package are as follows:

1. Organize the historic data
2. Homogenize the data
3. Fill in missing data
4. Examine and study the data
5. Decide upon the basic structure of the stochastic model
6. Normalize the data
7. Estimate the parameters for each generation group
8. Generate data
9. Analyze and check the generated data

Having presented a broad overview, the approach is now investigated in more detail. The various aspects are discussed in the logical order that they would normally be performed.

1. Data base. - A consistent data set resulting from a homogeneous data base is essential regardless of the type of stochastic approach selected. Without adequate data, it would be difficult to assess the success or failure of any stochastic generation scheme. A homogeneous virgin or near-virgin data base is recommended. Missing data must be filled in or estimated in order for the data base to be used. Data base preparation is not considered to be a task of these programs. Therefore, the preparation of a data base is left to the user.

2. Structure the problem. - Key stations must be identified and grouped for calculation purposes. Substation generation groups must also be identified along with the groupings to be used in disaggregation of annual data into seasonal data. Chapter III, Computational Options, will describe in detail how to structure a problem.

3. Reduction of data to normal. - Through the use of appropriate transformations, it is possible to change the basic data set into a set of data which follows the normal probability distribution. The transformations may be any of several types.

The reduction of the data to normally distributed data is necessary at this stage to ensure that the generated data will adequately follow the observed distribution and will reproduce accurately the desired higher moments. These transformations will be performed both on the annual and the seasonal data.

4. Estimation of parameters for the generation of annual values at key stations. - Once the data have been normalized, the next step is to undertake the task of estimating the parameters needed for generating values on an annual basis at key stations.

5. Estimation of parameters for the disaggregation of annual values at key stations into annual values at substations. - The disaggregation process proposed by Valencia and Schaake (1973), which is designed for disaggregation of annual data into seasonal data, provides the incentive for the approach taken here. The disaggregation approach used here while similar to that of Valencia and Schaake (1973) is used for an entirely different purpose.

6. Estimation of parameters for the disaggregation of annual values into seasonal values. - The disaggregation process developed by Valencia and Schaake (1973) provides the basis for the approach taken here. Their approach has gained fairly wide support. Mejia and Rousselle (1976) expand the basic approach slightly and also influence the approach taken here.

This approach will automatically preserve the variations of the seasonal serial correlation within the year and also allows for the use of different numbers of seasons within the year. It is not confined to monthly subdivisions. The fact that normalized data are used at this point permits the use of differing distributions for each season of the year. Since the data being used are transformed, the seasonal data will not sum exactly to the values generated for the annual data. The differences, while expected to be negligible in effect, are eliminated by adjusting the generated data to ensure that the seasonal values sum to exactly give the annual values.

7. Generation of synthetic data. - Once all the parameters have been estimated, the generation of synthetic data may be performed using the parameters which have been estimated.

8. Checking of synthetic data. - At least initially with each new application, the generated data should be examined to ensure that the desired statistics have been adequately preserved and that the generated values appear reasonable. Moments, crossing properties, and marginal distributions, help in this examination. In addition to calculating various statistics, plots will aid in this task.

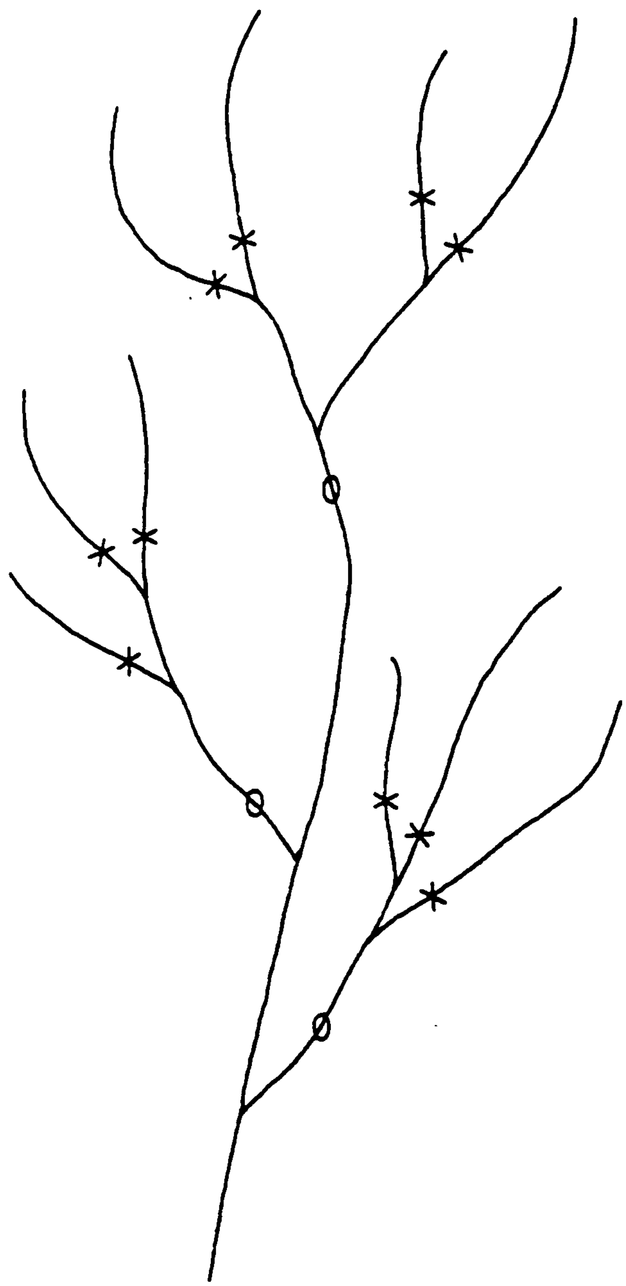
Figure 1.1 illustrates the basic analysis and synthetic generation steps. These will be discussed in more detail in the subsequent chapters.

Six basic equations are involved in this approach. They are written in matrix notation as follows:

Key Station Generation - Annual

$$K_{i+1} = AK_i + Be_{i+1}$$

equation 1



O - Key Station
X - Substations

<u>Analysis Steps</u>

1. Transform data to normal
2. Estimate parameters for key station generation - annual
3. Estimate parameters for key to substation disaggregation - annual
4. Estimate parameters for annual to seasonal disaggregation

<u>Synthetic Generation Steps</u>

1. Generate normalized key station data - annual
2. Generate normalized substation data - annual
3. Generate normalized seasonal data
4. Transform normalized data into properly distributed data

Figure 1.1. - Overall approach.

All Station Generation - Annual

$$N_{i+1} = CK_{i+1} + Df_{i+1} + EN_i \quad \text{equation 2}$$

All Station Generation - Seasonal

$$M_{i+1} = FN_{i+1} + Gg_{i+1} + HM_i \quad \text{equation 3}$$

Transformation - Annual

$$Q_i = T_1 (N_i) \quad \text{equation 4}$$

Transformation - Seasons

$$S_i = T_2 (M_i) \quad \text{equation 5}$$

Adjustment - Adjust either S or Q values such that:

$$Q_i = IS_i \quad \text{equation 6}$$

Notation:

Data Matrices -

- Q (mx1 for 1 year) Annual Series
- N (mx1 for 1 year) Normalized Annual Series
- K (px1 for 1 year) Normalized Annual Series for Key Stations
- S (nx1 for 1 year) Seasonal Series
- M (nx1 for 1 year) Normalized Seasonal Series

Coefficient Matrices -

- A (pxp)
- B (pxp)
- C (mxp)
- D (mxm)
- E (mxm)
- F (nxm)
- G (nxn)
- H (mxn)
- I (mxn)

Stochastic Components - All Normal (0, 1) -

- e - (mx1)
- f - (mx1)
- g - (nx1)

Transforms -

- T_1
- T_2

Subscripts -

- i - year

Dimensions -
m - Number of stations
n - Number of stations times number of seasons per year
p - Number of key stations

Equation 1 is based on an approach first proposed by Matalas (1967), later applied by Young and Pisano (1968), and further expanded upon by Finzi, et al., (1975) and O'Connell (1973). (The equation shown is only one of several options available for Key Station generation.)

Equation 2, used to disaggregate key station data into substation data, is based on an approach similar in form to that used by Valencia and Schaake (1973) for seasonal disaggregation.

Equation 3 is based on the seasonal disaggregation approach first proposed by Valencia and Schaake (1973). This approach was improved by Mejia and Rousselle (1976). Other useful references include Tao and Delleur (1976) and Gupta and Fordham (1974).

Equations 4 and 5 are used to change the data into transformed variates which follow the normal probability distribution. The transformations are purposely not specified at this time. Several options are available including no transformation at all.

Equation 6 is used to ensure that the seasonal data generated add up identically to the annual data generated. It amounts to a minor correction for the adverse effects of the two transformation equations and a correction for a minor shortcoming of the disaggregation scheme used. Several options are available to accomplish this.

The equations presented are in a very general form and, as a result, the coefficient matrices will have a great number of zeros. For example, in disaggregating key station data into data at all sites, each key station will only affect its own substations. In actual operation, these general equations are broken into more compact equations. Chapter V, Mathematics, will cover the derivation and form of the equations as actually applied.

C. <u>Capabilities and Limitations</u>

All of these programs are written to allow at least up to 30 stations in the input data files. Up to 10 key stations may be analyzed or generated at once as a group. Several separate groups of key stations may be generated. Substations may be generated with up to 10 substations in a group conditioned on up to 3 key stations. Analysis or generation of seasonal values may be done for up to 10 stations at a time. These limitations and others are covered more completely in Appendix E, Individual Program Details.

It is very important to recognize however that problem size (in terms of station numbers) has a more than linear relationship to execution time, and on some computers this may be a critical factor. On some personal computers, problem sizes approaching these limits could become impractical due to excessive execution times.

Users may experience some problems due to FORTRAN compiler differences between operating systems or brands of computers. Some unusual programming is utilized to efficiently carry out the calculations. These aspects include unusual use of equivalence statements, common statements, and passing dimensions via calling lists. Appendix E, Individual Program Details, covers these aspects and others in detail and should be read with great care.

REFERENCES

Burges, S. J., Assessment of Mathematical Models and Software Reported in the Applied Stochastic Techniques User Manual, USBR, December 1978

Finzi, G., E. Todini, and J. R. Wallis, Comment upon Multivariate Synthetic Hydrology, Water Resources Research, Vol. 11, No. 6, December 1975, pp. 844-850

Gupta, V. L. and I. W. Fordham, Multisite Streamflow Simulation of Truckee River, Nevada, Miscellaneous Publication No. 1275, Agricultural Research Service, June 1974, pp. 26-46

Matalas, N. C., Mathematical Assessment of Synthetic Hydrology, Water Resources Research, Vol. 3, No. 4, 1967, pp. 937-947

Mejia, J. M. and J. Rousselle, Disaggregation Models in Hydrology Revisited, Water Resources Research, Vol. 12, No. 2, April 1976, pp. 185-186

O'Connell, P. E., Multivariate Synthetic Hydrology: A Correction, J. Hydraulics Division, ASCE, Vol. 99, 1973, pp. 2392-2396

Tao, P. C. and J. W. Delleur, Multistation, Multiyear Synthesis of Hydrologic Time Series by Disaggregation, Water Resources Research, Vol. 12, No. 6, December 1976

Valencia, D. and J. C. Schaake, Jr., Disaggregation Processes in Stochastic Hydrology, Water Resources Research, Vol. 9, No. 3, 1973, pp. 580-585

Young, G. K. and W. C. Pisano, Operational Hydrology Using Residuals, Journal of the Hydraulics Division, ASCE, Vol. 94, No. 4, 1968, pp. 909-923

II. OPERATION

This chapter documents how to actually initiate runs of the various programs. The options involved will be briefly introduced and the required inputs will be specified. A detailed explanation of the computational options and the interpretation of results will be contained in Chapter III. That chapter will also discuss the process of structuring the problem in terms of key and substations and in terms of generation groups. As far as this chapter is concerned, it is assumed that the problem structure has already been decided upon and that the input data file has been prepared. The input data file is covered in Chapter IV. For the average user, this chapter along with the next two chapters is of the most interest and perhaps sufficient.

Due to the interactive nature, a large portion of this chapter will be a simple example of how a run is initiated for each of the programs. This should give the user a good feel for the type of questions asked and the form of typical interactive responses.

In most cases, the desired program itself is interactive and will ask questions of the user. This above explanation may seem mysterious to all but someone interested in the programming details. For the average user, he need only remember that, the procedure is to simply type the program name and then to answer the questions as asked. No knowledge is required as to where or how these questions came to be asked.

All of the programs are initiated using the same procedure file. The command "MENU" activates an interactive driver program by the same name which then allows the user to select the procedure he/she needs to use. The selected program is then executed, and upon completion, the user is returned to the MENU format. Ultimately, when all necessary programs have been run, the user can use the EXIT option from MENU and examine the result file.

Before presenting examples with discussion, it is well to list the various programs and their functions:

BEGIN -To create a parameter file.

CORREL -To calculate correlation matrices.

KEYPAR -To estimate the parameters needed to generate key station data on an annual basis.

DISPAR -To estimate the parameters needed to generate substation data on an annual basis by disaggregation of key station data.

SEAPAR -To estimate the parameters needed to generate data on a seasonal basis by disaggregation of annual data.

TRNPAR -To estimate the parameters for the transformation of nonnormal data into normal data and the inverse of this transform, also creates the transformed data file.

FLWGEN -To generate synthetic data utilizing the estimated parameters.

TERMPL -Plots graphs similar to INIT except that the graphs are produced on either a terminal or line printer.

The computer files of interest to the user are:

*.DAT - Basic input file named by the user (six characters); referred to herein as "generic problem name."

*.OUT - Result file (generic problem name + .OUT).

For a complete list of all files associated with this package and their interrelationships see Appendix D, Organization.

The next portion of this chapter is devoted to taking a hypothetical case (without specifying details) and explaining the order in which the programs are to be run and the options among which the user must choose. The first step is to prepare the basic or original Input Data File. This file is composed of heading material and historical data. For some river basins it may be very important how the groups are specified, especially the selection of key stations and substations (in some cases this choice is not obvious and the user may want to initially structure the problem more than one way, run the programs, and make the final choice based on the performance of generated data). In many cases the structure may be based on the results of plotting or calculated correlations. The details of structural choices are deferred to the next chapter, Chapter III, Computational Options and Interpretation of Intermediate Results.

For this hypothetical case, it is assumed that six stations are involved (stations A, B, C, D, E, and F): stations A and B are treated as key stations with their annual data to be generated together, annual data at station A are to be disaggregated into annual data at substation F, annual data at station B are to be disaggregated into annual data at substations C, D, and E. The annual values will be disaggregated into seasonal values for all stations together. Thus, there are one key station generation group, two key/substation generation groups, and one annual to seasonal generation group. The hypothetical case has a physical basin configuration as shown in figure 2.1.

In general, there are two purposes to having key and substations rather than having all stations be key stations. One is simply to reduce the number of parameters. The second reason is so that several stations whose values essentially add up to the value of a single station may be generated in a manner which preserves the statistical properties of both the sum (key stations) and of the individual substations. This is done by generating substation dependent on the key station or stations selected. In most cases the user will find it desirable to reorganize the data in an expanded form. For the hypothetical basin shown, the flows at stations A, B, C, D, E, and F are total instream flows. To go to an expanded form, two additional stations (sets of data) would be created; one would be the increment of flow between station F and station A, the other would be the increment of flow between stations C, D, and E and station B. Thus, two substations would be generated dependent on station A and four substations would be generated dependent on station B.

The steps in using this set of programs are:

1. Prepare historical data in the input data file format.

2. Select a problem structure and enter this into the input data file format.

3. Utilize programs TERMPL and CORREL to verify that the problem is structured in an acceptable manner and to detect errors in the input data.

4. Create a parameter storage file by running program BEGIN.

5. Use program TERMPL or TRNPAR to aid in selecting transformations in order to normalize the data.

6. Run program TRNPAR twice, first to specify the transformation parameters and second to create a transformed data file.

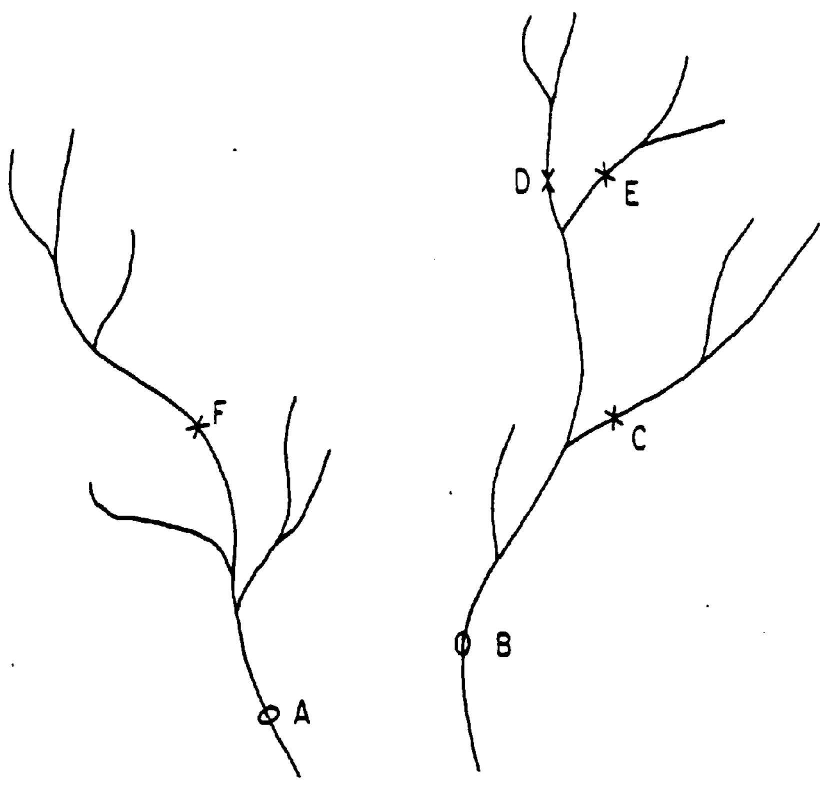


Figure 2 . 1 Hypothetical basin configuration
0 key stations
X substations

7. Run program KEYPAR to estimate the key station generation parameters (once for each key station group).

8. Run program DISPAR to estimate the key/substation disaggregation parameters. (Once for each key/substation group. For the hypothetical example, two runs are required.)

9. Run program SEAPAR to estimate the annual to seasonal disaggregation parameters. (Once for each annual/seasonal group.)

10. Run program BEGIN in order to list and check to ensure that all parameters have been stored correctly.

11. Run program FLWGEN to generate synthetic streamflow values.

12. Use programs TERMPL and CORREL to verify that the generated data behave as expected.

It should be noted that the order given is not mandatory. Most of the programs may be run in a different order or run several times. The main requirements are that a parameter file must be created (BEGIN) before any parameters may be saved and a transformed data file must have been created in order to run programs KEYPAR, DISPAR and SEAPAR. This completes the discussion of the hypothetical basin.

Following is a summary of the running of each program. The normal responses and inputs are indicated for a typical run. The main options available are introduced along with the main modes of running. For a complete list of computational options with explanations, the reader is referred to Chapter III, Computational Options. Chapter VI, Quick Guides, also will aid the new user. The Quick Guides are of value to all users, both new and experienced. These guides include a list of all of the input, output, or control options.

A. Program BEGIN

This program is designed to either create a new parameter file or to read and list an existing parameter file. An existing parameter file may be listed at any stage: before the parameters have been defined, after some of the parameters have been defined, or after all of the parameters have been defined. The structure of the parameter file is based on the generation groupings specified in the basic input data file. If this information is changed, a new parameter file must be created. Required inputs are the generic problem name (i.e., the basic input data file) which is needed for all of these programs and a string of input-output controls. The string of controls is a string of zeros and ones each of which controls one option. Each option either affects the data input, the output, or the methods used. To activate an option, a one should be typed in the position of the string corresponding to that option. For example, to activate the second and fourth options out of a seven-option string the user should respond with "0101000" or "0101." Zeros are the normal response and do not activate the options. For this particular program, the main option is the first. If this option is not activated, a new parameter file is created. If this option is activated, an existing parameter is listed.

B. Program CORREL

This program computes correlations among the data. Original, transformed, or generated data may be utilized, respectively, from the original data file, the transformed data file, or generated data file. Presently the program requires that all three data files be available even if only the original data is to be analyzed. If it is desired to run CORREL on the original data set (prior to the creation of transformed or generated data) it is suggested that the original data be copied into "dummy" files with names corresponding to the transformed and generated data sets (see page VI-3). Correlations may be calculated for annual data, seasonal data, and between annual and seasonal data. A normal run is made with all of the input-output controls set to zero.

C. Program KEYPAR

Program KEYPAR is designed to estimate the parameters needed to generate annual data at a group of key stations. The data analyzed are the data contained in the transformed data file. The user has the option of either a long-term persistence model, ARMA (1, 1), or several short-term persistence models. In general, the short-term models will be adequate. Therefore, the discussion here will be limited to those models. Cases where the ARMA model would be preferred are discussed in the next chapter. (Note the ARMA model is presently not available.)

The user has the option of directly preserving either lag one or both lag one and lag two correlations. In general lag one correlations alone are adequate but in some rare situations lag two may also be desirable. These two alternatives may be referred to as first and second order linear auto- regressive models. The user also has the choice between three methods for preserving the lagged correlations. The first, the full Matalas approach, preserves both same series lagged correlations and cross correlations between the various stations. The second, herein called a Matalas approach with lagged cross correlations ignored, only preserves same series lagged correlations. The third, the decoupled approach, is identical in form to the second; however, the parameters are estimated in a slightly different manner. The decoupled approach had advantages and disadvantages when compared with the Matalas method with lagged cross correlations ignored. The disadvantage is that at least theoretically the moments (correlations) are not preserved as they are estimated from historical data. In reality the error is in most cases insignificant. The advantage is that, as the name implies, the calculation may be broken into several discrete steps. Although this is not done here, this allows the hydrologist to examine serially independent time series and their intercorrelations prior to calculating the cross correlation generation parameters. In most cases, the Matalas approach with lagged cross correlations ignored is adequate. If this approach does not adequately indirectly preserve the lagged cross correlations, the full Matalas approach should be used. The recommended approach is to first try the Matalas approach with the lagged correlations ignored with a single lag. If this approach is not adequate then either the full Matalas approach or the second lag should be considered. The normal run is made with all input-output controls set to zero.

D. Program DISPAR

This program is very similar to KEYPAR except that rather than estimating key station generation parameters, the parameters being estimated are those used to generate substation annual flows by disaggregating already generated key station annual flows. This program accounts for lag zero correlations between the key and substations and among the substations and also lag one correlations between the substations. There is an option to ignore the lag one correlations between the substations. The data being analyzed are the annual data from the transformed data file. The normal run is made with all input-output controls set to zero.

E. <u>Program SEAPAR</u>

This program is similar to KEYPAR and DISPAR. The program calculates the parameters needed to disaggregate generated annual data into generated seasonal data. Same year correlation between the annual data and seasonal data are preserved along with lag zero and lag one correlation among the seasonal data. There are no options open to the user as to the number of lags or the method utilized. The data analyzed are from the transformed data file. The normal run is made with all input-output controls set to zero.

F. <u>Program TRNPAR</u>

This program is designed to ask the user what transformations are to be performed on the original data to create the transformed data base. A minimum of two runs of this program are required as it performs two tasks. First this program is used to define the transformation parameters. It may be run in this mode several times, each time defining the parameters for some of the stations and/or redefining parameters. The second mode of running the program is to create a transformed data file utilizing the parameters already defined. The purpose of the transformations is to reduce the original data into transformed data which are nearly normally distributed. Program TERMPL with its plotting capability is of great help in selecting transformations. The transformations available are of two forms, a power transformation and a logarithmic transformation. These are expressed mathematically below:

$$t = (o+a)^b$$

and

$$t = \ln (o+a)$$

where o is the original data, t is the transformed data, a and b are user specified parameters. The user may also opt not to transform the data at all. The parameters specified are also saved in the parameter file and are used by FLWGEN, the generation program, to convolute normally distributed generated data back into the proper distribution. Normal runs are made with all input-output controls set to zero.

G. <u>Program FLWGEN</u>

This program generates synthetic data. The parameters to be used for generation have already been defined by programs KEYPAR, DISPAR, SEAPAR, and TRNPAR and saved in the parameter file. There are three options as to how the generation of synthetic data is started. They are (1) a "lead-in" years approach recommended only for the short-term persistence models, (2) a random start approach recommended for both short- or long-term persistence models, and (3) a fixed start approach utilizing initial conditions which can be used for both short- and long-term persistence. Only the "lead-in" years approach is presently available on this package.

To use the "lead-in" years approach the user must furnish the number of years to be generated, the number of "lead-in" years, and the random number seed. To reproduce a generated sequence the random number seed must be input identically. Only a small change (i.e., one digit) in the seed value will change the generated trace entirely. It is recommended that three lead-in years be used. These lead-in years ensure that the serial correlations have become stabilized. If no lead-in years have been specified, the initial couple years of generated data will be questionable.

To use the random start approach, the user must furnish the random number seed. Again only a small change in the seed value is required to completely change the generated trace. (Not presently available on this package.)

The normal run is made with all input-output controls set to zero.

The user may opt to generate only key stations or annual data rather than the entire data set. This may be of aid in checking the acceptability of the synthetic data before all of the parameters have been defined.

H. Program TERMPL

This program plots graphs on a line printer. The plots can use annual or seasonal data which can be read from the original or generated data files. Statistical properties as well as the data itself may be plotted. Presently the program requires that both original and generated data sets be available. If it is desired to plot the original data before the generated data is created, it is suggested that a "dummy" file be set up by copying the original data file into a file with a name corresponding to the generated data set (see page VI-9).

REFERENCES

Mejia, J. M. and J. Rousselle, Disaggregation Models in Hydrology Revisited, Water Resources Research, Vol. 12, No. 2, April 1976, pp. 185-186

Valencia, D. and J. C. Schaake, Jr., Disaggregation Processes in Stochastic Hydrology, Water Resources Research, Vol. 9, No. 3, 1973, pp. 580-585

III. COMPUTATIONAL OPTIONS AND INTERPRETATION OF INTERMEDIATE RESULTS

This chapter attempts to explain in some detail the various options available in running the programs. Recommendations as to how to choose between options are given and the interpretation of intermediate results is also included. The setting up of the problem structure is covered in some detail. The logic of deciding which stations should be key stations and which stations should be substations along with the pros and cons of alternate approaches is discussed. The topic of verifying the validity of generated synthetic data is briefly discussed.

Chapter V, Mathematics, covers in detail the mathematical methods used while this chapter sticks primarily to the philosophy of approach. Some statistical tests are indicated, referenced, and discussed only to the point where a user can use them for guidance in making the decisions needed to run these programs in a reasonably efficient manner.

This chapter is designed primarily for the general user. The more ambitious user interested in the background or a person working on additions, improvements, or modifications to the programs will want to read in detail Chapter V, Mathematics, along with several of the references.

A. Initial Activities

The first step is to set up a tentative data file for the problem at hand. This data file should include complete sets of data for each of the stream stations of interest. This may mean that the user has already done considerable work in gathering data and filling in missing records. Data may be in terms of either flow at the stations or increments (gains) of flow between stations. Increments have an advantage in basins where several stations gauge flows along a common tributary or where their increments add to give a "key" station. Increments also tend to lend themselves well to later river basin modeling. This approach requires additional stations over the total flow approach but is generally preferred. If a person initially wants to analyze more stations than he is intending to utilize in the final product, this is entirely possible. The larger set of stations may be worked with and at the appropriate time be edited down to the final choice of stations for analysis and generation. The input data format does have the disadvantage of requiring the station index on each line of data. However, due to the location of the index next to the line type identifier, changing these indexes is a trivial task using "Editor." The initial structuring of the problem into generation groups in the input files should be done to avoid leaving blanks in the input file.

Once the tentative data file has been created, the task becomes to examine the data to aid in recognizing the statistical properties of the data and to reveal any blunders in the data. To check for errors or anomalies in the data, it is recommended that program TERMPL be used extensively. A serial plot of each station's data is definitely justified. Plots of statistics will also help the user get a feel for the data. Correlation plots are also an aid to determine if the data are homogeneous. Both double mass diagrams and simple correlations plots are of help.

Correlation plots will also aid in determining if the linear relations used to preserve correlations during the generation of synthetic data are adequate. If the correlation plots do not plot as straight lines (with some scatter) the user will want to very carefully watch the effect transformations of the data have on these plots. Also, to get an idea of the intercorrelations between stations, the user will find program CORREL useful. This program computes various correlation matrices between the stations for annual and seasonal data. The correlation matrices will aid the user in deciding which stations must be generated together. Relative physical location of the stations should also affect this decision. The probability plots (cumulative distribution function) will aid in alerting the user to the severity of the transformations which will be required. Ideally, the user would like to see normally distributed data (requiring no transformations) with linear correlations.

Once the data have been examined thoroughly, the user should have a fair understanding of the statistical properties of his data and now his problem may be reasonably structured. He should at this time be prepared to decide upon a final problem structure by selecting the various generation groups.

B. Selection of Generation Groups

The main objective to keep in mind in the selection of generation groups is to preserve important correlations found in the input data. It may not be easy to recognize the "important correlations" so some general rules will be presented which may be of help to the user. Before presenting these rules some comments are appropriate.

A complication is that correlations can be preserved directly or indirectly. For example, if the correlations between key stations A and B are preserved directly, station C is grouped as a substation of key station A, and station D is grouped as a substation of key station B, then some correlation between stations C and D will be indirectly preserved in the generated data.

There are some very sound reasons to minimize the number of parameters (referred to as parsimony of parameters). A judicious choice of groups can aid in doing exactly this. The user may want to develop several structures for his one problem and compare the number of parameters required. A key station generation group with N stations requires approximately $3N^2$ parameters. A key to substation disaggregation group with N key stations and M substations requires approximately $MN+2M^2$ parameters. An annual to seasonal disaggregation with N stations and S seasons requires approximately $3SN^2$ parameters.

The most critical choices are the grouping of key stations and the choices of which stations are key stations and which should be substations. This problem is of great importance if the problem involves a basin where several of the time series add to form a single important station's time series. For example, if we are jointly generating several stations which add to give one station or if the statistical properties of the sum are of great importance, then the key to substation scheme may need to be applied.

All of the stations could be treated as key stations with no substations if these considerations were not of importance. Also, if the data do not require any transformations to become normally distributed, one may not need to resort to disaggregation.

Usual experience has shown severe problems in modeling synthetic flows when a key to substation disaggregation has not been used. When generated intervening values were summed to provide values at stations along the river, it was noted that statistical properties were not well preserved for these summation stations. Some attempts had been made to generate the stream stations directly, but in these cases intervening values calculated from the generated station values were poorly behaved.

The user should have a healthy sense of mistrust for the estimated correlations. The sampling variation of these estimates can be quite important and, as a result, decisions should not be based entirely on these estimates. When later examining the adequacy of generated data, this also will be of importance. In general, we expect all cross correlations between stations to be positive. Therefore, negative estimates should be viewed as being the same as no correlation. An idea of the variability of estimates for the correlation coefficient may be obtained by calculating the estimate's standard deviation as:

$$S_r = \frac{1 - r^2}{\sqrt{N}}$$

This approximation is, of course, gross because of several necessary assumptions. Also, the distribution of the correlation coefficient is not symmetric and, therefore, the standard deviation is not truly meaningful. The equation presented is equation 11.6 of Yevjevich (1972). If the correlation coefficient is set equal to zero, a typical sample size of 50 years gives a standard deviation, Sr, equal to approximately 0.14. This means that the estimates may vary widely about the true values. In fact, one could only expect about 60 percent of the estimates to be within a standard deviation of their true values. For the purposes of making decisions concerning the structuring of a problem, a difference between two correlation coefficients (historical and generated) of less than say one-tenth should not alone dictate decisions. By the same token, low values may be considered to be zero. In the case where the correlation between two streams is near zero, synthetic data for the two streams could be generated independently. The discussion here is purposely loose because we are not performing statistical tests of significance but rather using statistical properties of an estimate to guide our decisionmaking. Unlike significance testing, we are not strongly biased towards a null hypothesis and are not overly concerned if we preserve a small correlation which does not truly exist except as a sampling error. As an aside, it may be noted that if data for two stations are generated independently, the results are not greatly different from the results of generating the data while preserving a correlation coefficient of one-tenth.

Hopefully, the preceding remarks will help to clarify the following "rules" which of course are not hard and fast rules but rather guides strongly flavored with the authors' biases.

1. Key stations are total instream flow time series while substations are intervening increments of flow. "Rim" stations may be either key or substations.

2. Key stations values generally can be calculated by simply adding together the values for several substations. This rule does not apply to rim stations which are also key stations.

3. Only substations or key stations having no tributary substations (that is, rim stations) are later used for modeling.

4. Annual cross correlations (lag zero) are of primary importance; all other cross correlations (especially seasonal) will take care of themselves. The decisions regarding problem structure need not include any other cross correlations.

5. Limit the number of key stations to 10 per group, preferably 5 per group.

6. Base the selection of substations for a given key station (or stations) primarily on location. As a secondary input, correlations should also be considered.

7. For a key to substation disaggregation, limit the number of key stations to 3 and the number of substations to 10. For most cases, one key station with about four substations would be most practical and advisable.

8. Limit the number of stations in the seasonal disaggregation to 10 or fewer, preferably 4 or 5. Base the grouping primarily on location. Note that many seasonal cross correlations can be preserved indirectly by the annual cross correlations.

C. Transformations

The transformation of data to a normal form is a required step which if handled poorly will cause the generated synthetic data to be worthless. This step requires a great deal of judgment. In modeling studies where a great deal of reservoir storage is available, the distribution of generated data will not be as critical. In cases where extreme flows are of great importance or cases where the observed distribution is greatly skewed (unsymmetrical), the transformations are crucial. If the data reflect a bounded distribution, such as a zero bound, the choice of a transformation will determine how closely this bound will be preserved in the generated synthetic data.

The specified transformations are used twice. Once when analyzing data to transform observed data into normally distributed data which are stored in the transformed data file. The second time is when generating synthetic data generated normally distributed data is transformed back into a distribution similar to that which had been observed. Of course, the second use is really with the inverse of the transformation.

Experience has shown that the blind adoption of any one transform with an automatic fitting (even where it has two or three parameters estimated from the data) will result in unacceptable generated data a large percentage of the time. Automatic transformation fitting capabilities are available in program TRNPAR, but results should be examined graphically using program TERMPL to assure that the transformation selected is acceptable for the purposes intended. The user is cautioned to avoid overcompensating in the transformation because of only one or two "wild" observed values. Experience has shown that overcompensating for the highest observed values without equal weighting being given to the lower values will often produce a distribution for generated data which may reproduce the peaks adequately but may produce unrealistic low values (such as large negative values where the observed distribution was obviously bounded by zero).

As with correlations, there are sampling errors involved. Therefore, one does not have to be too concerned with the way the distribution fits one or two points. Rather he should be concerned with the overall fit and the pattern of errors. It is well to test individual points with a simple statistical test to ensure that all points are reasonably close to the normal distribution once the data have been transformed.

A good test to use is the Smirnov-Kolmogorov test. The test is quite simple to apply and is contained in most statistics texts. For the purposes here, an approximation of the test is presented. The first step is to determine the maximum deviation of the observed (transformed data) cumulative distribution function from the fitted formal cumulative distribution function. This value is then compared with a critical value equal to 1.36/ N. If it exceeds the critical value, the fit is rejected. This critical value corresponds approximately to a confidence level of 0.95. For a more exact version of this test or for other confidence levels the reader should refer to a statistics text. This test has the disadvantage that it, like all distribution tests, is very weak. That is, it is very hard to reject the distribution being tested. For this reason the reader is urged to use the interactive graphics (program TERMPL) to examine the distribution of various transforms of the original data.

All of the activities associated with choosing the transforms will be centered around the use of probability plotting. These may be made through the use of program TERMPL. In general, it is expected that the midportion of the observed cumulative probability distribution (really frequency distribution to be more correct) will closely follow a normal distribution while the high and low values will be the areas of obvious nonnormality. The problem becomes one of finding the proper transformation to make the data plot as a straight line on normal probability paper. The cumulative probability distribution plots done by program TERMPL are on normal probability paper.

The user may note that only unimodal distributions can be accounted for by these programs. This is not felt to be a serious limitation. Most cases where the data could be interpreted as having a bimodal distribution, the choice between a unimodal or bimodal distribution is still a very subjective choice. It is very rare that distribution could be proven bimodal. In addition, for most hydrologic studies, the results of a study would not be affected in the least by this limitation.

The following steps apply to fitting a transformation of the form $t = (o+a)^b$ where t are the transformed data, o are the original data, and a and b are parameters to be estimated:

1. Plot the original data using the cumulative distribution plot of program TERMPL.

2. Estimate a such that the smallest value of o+a will be slightly larger than zero. If the smallest value of o is positive, estimate a as zero. Estimate b as 1.

3. Plot the cdf (cumulative distribution function) using program TERMPL. By eye, draw the best straight line you can through the plotted data. If the data plot as a straight line, you are through. All points should as a minimum pass the Smirnov-Kolmogorov test.

4. Concentrate on the larger values. If the observed data curve upward away from the straight line, estimate b as a number smaller than the last estimate for b and return to step 3. If the observed data curve downward away from the straight line, increase the estimate for b and return to step 3. Changes in b should be made quite gradually and it is very doubtful that the value of b should range less than 0.1 or greater than 1.5 as extremes. Probably 0.3 to 1.2 would be an expected range. If the upper part of the distribution is nearly a straight line, proceed to step 5.

5. At this point, the middle and upper portion of the distribution should closely follow the normal distribution. If b still equals 1, nothing can be done to alter the shape of the distribution by varying parameter a. Therefore, if the lower portion is poorly fit and b is equal to zero, b must be altered slightly (say 0.9, 0.95, or 1.05) in a way that least severely affects the shape of the upper portion of the distribution.

6. A cdf plot is made of the transformed data with current estimates of a and b. If the lower portion of the cdf is straight, go on to step 8.

7. If the lower portion of the observed data curves upwards away from the straight line, reduce the estimate of a. If it curves the other way, increase the estimate of a.

8. At this point, the lower portion of the cdf should be fit very well by the fitted straight line. Hopefully, the middle and upper sections are also well fitted and the task has been completed. If the upper portion is not good enough, return to step 4 and carry on until the best fit possible has been determined.

It is not expected that any major problems will be encountered in adequately fitting the midsection. It should be well fitted once the end sections are fit. If seemingly unsurmountable problems are encountered, the user probably is placing too much faith in a few "wild" points which could easily by explained as sampling errors. One must be careful to avoid putting too much faith in the highest two or three points or in the lowest two or three points.

A similar procedure may be used for fitting the logarithmic transformation. The transformation in this case has the form $t = \ln (o+a)$. Only parameter a, the constant term, may be varied. No general rule can be developed for fitting a as it must account for imperfections both at the upper and lower tails of the distribution. However, the lower tail is always affected more by a change in the parameter than the upper tail.

D. Key Station Generation Parameters

Two basic models are planned for this package, a long-term persistence model [ARMA (1, 1)] and a short-term persistence model. If long-term persistence (i.e., low-frequency persistence) is important to a study and if the Hurst coefficient as calculated by program CORREL is greater than about 0.6, then the ARMA (1, 1) model may be of help (Burges, 1978). For all other cases, short-term persistence models are most appropriate. The ARMA model is not presently available but is planned for the future.

If the short-term persistence model is selected, the user has several choices as to the exact form of the model. It is expected that regardless of the form selected (full Matalas, Matalas without lagged cross correlations, or decoupled) and regardless of the number of lags selected (one or two), the results will be substantially the same. The choice of which form is up to the user. Some guidance will be given here, but the user may confidently use the full Matalas approach with two lagged correlations.

It is assumed that there is always at least lag one persistence in any annual streamflow series. To guide the user in the choice between lag one and lag two, the following steps may be taken to evaluate the significance of the second lag:

1. Determine the first and second serial correlation coefficient either by running program CORREL or taking the values from an autocorrelation plot. These values are denoted as r_1 and r_2.

2. Calculate values of a_1 and a_2 as follows:

$$a_1 = \frac{r_1 - r_1 r_2}{1 - r_1^2}$$

$$a_2 = \frac{r_2 - r_1^2}{1 - r_1^2}$$

3. Calculate the explained variance for lag one, R_1^2 and for lag two, R_2^2, as follows:

$$R_1^2 = r_1^2$$

$$R_2^2 = a_1 r_1 + a_2 r_2$$

4. Calculate the increase in R^2 brought about by the second lag:

$$\Delta R^2 = R_2^2 - R_1^2$$

A simple test is if R^2 is larger than a test value, the second lag is justified. Test levels proposed (Yevjevich, 1972) are 0.01 or 0.02.

In order to aid in the decision as to the need for the lagged cross correlations to be preserved directly, the following steps should prove adequate:

1. Run program KEYPAR for the stations in question. Use the Matalas approach with lagged cross correlations ignored and with only one lag. Utilize the option to print out the covariance matrices.

2. Compare the calculated M_1 covariance matrix with the altered M_1 covariance matrix. Compute a difference M_1 matrix if the differences look significant. If the values are quite close, you may at this point assume the lagged cross correlations are adequately preserved indirectly. If the differences are not small, proceed with the following steps. The remaining steps shall only treat one term taken from the i-th row and the j-th column of the M_1 matrix.

3. Compute two qualities, σ_i and σ_j using the M_o matrix as follows:

$$F_i = m_o\ (i,\ i)$$

$$F_j = m_o\ (j,\ j)$$

where m_o (i, j) indicates the element of M_o taken from the i-th row and j-th column.

4. Divide the difference element by the product of σ_i and σ_i.

5. If the result is large, the lagged cross correlation should be preserved directly. One can consider 0.10 as large for the purposes here. This test is of course very crude and not any way near being a rigid statistical test. If less than 0.10, the user can safely assume that the indirect preservation will produce generated data very much the same as data generated using a direct preservation of lagged cross correlations.

E. Key-Substation Generation Parameters

Since it is assumed that the structure of the problem has been set, there is little in the way of options in running program DISPAR to estimate the parameters. The approach taken directly preserves lag one correlations although this is modified somewhat by the dependence of the substations on the key stations. Some additional lag two correlation is indirectly possible due to that preserved in key stations. This is probably more than adequate. The additive nature of substation values into key station values may be preserved exactly if no transformations are required. Due to transformations, this may be altered slightly but should be only a very minor degradation of this important attribute of key substations scheme.

F. Seasonal Generation Parameters

The program SEAPAR estimates these parameters with no options available to the user. The use of transforms and model inadequacies affect the exactness of the additive nature of seasonal flows into annual flows. Program FLWGEN, when generating synthetic data, will automatically readjust the seasonal data to preserve the additive nature after the inverse transformations have been performed. Many cross correlations and lag correlations are indirectly preserved through the annual correlations and the annual to seasonal correlations. The indirect correlations are hard to estimate; therefore, in selecting stations to have their annual data disaggregated into seasonal data together, the user should base his choice primarily on geographic locations of the stations. It is highly recommended that the number of stations per group be held to a minimum because of the large number of parameters involved. The best number of stations per group is three but from two to six is entirely satisfactory. In general, larger groups should be avoided.

G. Checking of Generated Data

The only true test of the adequacy of generated data is to examine them in terms of the intended use of the data. The statistical behavior of the data should be compared to the behavior of the original data to ensure that all important aspects are adequately preserved. This is an area in which statistical tests should be applied judiciously and the effects of sampling variations should be kept in mind. The tests already mentioned could be used or the user can apply more exacting tests found in most statistics textbooks.

One method of checking is to make several separate generations of synthetic data. The results may then be viewed as samples from a population and tested statistically for differences from the original data.

Both the plottings produced by program TERMPL and the statistics calculated by program CORREL will be of use in checking the generated data.

H. Use of Synthetic Data

The philosophy of synthetic data is that the natural process can be characterized in a statistical manner and that synthetic data may be produced which preserve these properties. Since the synthetic data behave statistically the same as the observed historical data, all of the generated traces are equally likely to occur in the future. The historical data are no more likely to reoccur than any one synthetic trace is to occur.

Synthetic data are generally used either in connection with river basin modeling or are analyzed directly to determine the probabilities of various potential future events.

REFERENCES

Burges, S. J., Assessment of Mathematical Models and Software Reported in the Applied Stochastic Techniques User Manual, USBR, December 1978

Mejia, J. M. and J. Rousselle, Disaggregation Models in Hydrology Revisited, Water Resources Research, Vol. 12, No. 2, April 1976, pp. 185-186

Valencia, D. and J. C. Schaake, Jr., Disaggregation Processes in Stochastic Hydrology, Water Resources Research, Vol. 9, No. 3, 1973, pp. 580-585

Yevjevich, V. M., Probability and Statistics in Hydrology, Water Resources Publications, Fort Collins, Colorado, 1972, 302 p.

IV. INPUT DATA FILE

This data file is furnished by the user and contains not only the hydrologic data to be analyzed, but also a considerable amount of what might be termed preliminary or heading material. This heading material includes titles and subtitles for the data file, a creation date, information concerning the total number of stations, the number of seasons, the number of generation groups of each type (i.e., key, key to substation, and annual to seasonal), and one line of preliminary information for each stream station. The lines in this file must adhere strictly to the order specified in this chapter. The lines of input will be covered in the correct order. Table 4-1 shows the organization of the input data file.

Table 4-1 - Input Data File Organization

Line type	Number of lines	Function
Heading material: one set of lines		
T	1	Title
S	1	Subtitle
C	1	Creation date, data set identifier
N	1	Number of stations, number of seasons
M	1	Number of generation groups
K	1 for each key station	Lists stations in the group
X	1 for each key subgroup	Lists stations in the station group
Y	1 for each annual-seasonal group	Lists stations in the group
H	1 for each station	Preliminary information
Annual data: one set of lines for each station		
I	1	Station label
A	1 for each 8 years of data	Contains annual flow data
Seasonal data: One set of lines for each station		
I	1	Station label
D	1-2 or more lines/year	Contains seasonal flow data

Table 4-1 - Input Data File Organization (cont.)

All of the input lines are described below. The information contained in parenthesis are the actual formats used by the programs.		
T Line - Purpose: Title for input data file		
Cols. 1-2	(A2)	Punched as "T " (Note: T followed by a blank space)
Cols. 3-80	(7A10, A8)	Title
S Line - Purpose: Subtitle for input data file		
Cols. 1-2	(A2)	Punched as "S "
Cols. 3-80	(7A10, A8)	Subtitle
C Line - Purpose: Creation Date		
Cols. 1-2	(A2)	Punched as "C "
Cols. 3-22	(2A10)	Creation date
Cols. 33-42	(I10)	Data set identifier
N Line - Purpose: To specify total number of stations and number of seasons		
Cols. 1-2	(A2)	Punched as "N "
Cols. 4-5	(I2)	Total number of stations
Cols. 9-10	(I2)	Number of seasons
M Line - Purpose: specify numbers of groupings		
Cols. 1-2	(A2)	Punched as "M "
Cols. 4-5	I2)	Number of key station generation groups
Cols. 9-10	(I2)	Number of key to substation generation groups
Cols. 14-15	(I2)	Number of annual to seasonal generation groups
K Line - Purpose: Specify key station generation groupings (one line for each group)		
Cols. 1-2	(A2)	Punched as "K "
Cols. 4-5	(I2)	Number of key stations in the group

Table 4-1 - Input Data File Organization (cont.)

Cols. 9-10	(I2)	Index of key station
Cols. 14-15	(I2)	Index of key station
•	•	•
•	•	•
•	•	•
Up to Cols. 79-80		(For all key stations)
X Line - Purpose: specify key/substation generation groupings (one line for each group)		
Cols. 1-2	(A2)	Punched as "X "
Cols. 4-5	(I2)	Number of key stations in the group
Cols. 9-10	(I2)	Number of substations in the group
Cols. 14-15	(I2)	Index of key station
Cols. 19-20	(I2)	Index of key station
	•	•
•	•	•
•	•	•
•		(For all key stations)
•	(I2)	Index of substation
•	(I2)	Index of substation
•	•	•
•	•	•
•	•	•
Cols. 79-80	(I2)	(For all sub stations)
Y Line - Purpose: specify annual/seasonal generation groupings (one line for each group)		
Cols. 1-2	(A2)	Punched as "Y "
Cols. 4-5	(I2)	Number of stations in the group
Cols. 9-10	(I2)	Index of station

Table 4-1 - Input Data File Organization (cont.)

Cols. 14-15	(I2)	Index of station
•	•	•
•	•	•
•	•	•
Cols. 79-80	(I2)	(For all stations in group)
H Line - Purpose: Preliminary information for each station (one line for each station)		
Cols. 1-2	(A2)	Punched as “H ”
Cols. 4-5	(I2)	Station index (must be sequential)
Cols. 22-25	(I4)	Starting year of data
Cols. 27-30	(I4)	Ending year of data
Cols. 36-75	(4A10)	Station title
I Line - Purpose: Identification of blocks of data, station titles		
Cols. 1-2	(A2)	Punched as “I ”
Cols. 3-4	(I2)	Station index
Cols. 11-50	(4A10)	Station title
A Line - Purpose: Contains annual data		
Cols. 1-2	(A2)	Punched as “A ”
Cols. 3-4	(I2)	Station index
Cols. 5-16	(A8, A4)	Station short title
Cols. 17-20	(I4)	Beginning year on line
Cols. 21-24	(I4)	Ending year on line
Cols. 25-80	(8F7.0)	Annual data
D Line - Purpose: Contains seasonal data		
Cols. 1-2	(A2)	Punched “D ”
Cols. 3-4	(I2)	Station index
Cols. 5-16	(A8, A4)	Station short title
Cols. 17-20	(I4)	Year of data

Table 4-1 - Input Data File Organization (cont.)

Cols. 21-24	(I4)	Line number (i.e., 1 if first line for the year, 2 if second line for the year)
Cols. 25-80	(8F7.0)	Seasonal data

V. MATHEMATICS

This chapter covers the technical details involved in the mathematics of the approaches. It covers both the equations used in the programs and the derivations of many of the parameter estimation equations. The symbols used herein are chosen to correspond as closely as possible to the variable names used in the computer programs. Where references exist, they will be noted to aid the reader who is interested in further study or perhaps expansion or improvements of the programmed techniques.

A. Notation

The symbols used are very similar to those used in Chapter I, Introduction. Matrices are generally indicated by capital letters except for row matrices made up of random numbers which are denoted by lower case letters. Individual elements of matrices are also denoted by lower case letters. The equations and notation expressed in Chapter I are modified slightly here to be exactly the same as actually programmed. The following list notes the principal equations and the primary uses of symbols in this chapter:

Equations:

Key Station Generation - Annual - lag one AR model

$$X_i = AX_{i-1} + Ce_i \qquad \text{equation A.1}$$

Key Station Generation - Annual - lag two AR model

$$X_i = AX_{i-1} + BX_{i-2} + Ce_i \qquad \text{equation A.2}$$

Key Station Generation - Annual - ARMA (1, 1) model

$$X_i = AX_{i-1} + Be_i + Ce_{i-1} \qquad \text{equation A.3}$$

Key to Substation Disaggregation - Annual

$$N_{i+1} = CK_{i+1} + Df_{i+1} + EN_i \qquad \text{equation A.4}$$

Annual to Seasonal Disaggregation

$$M_{i+1} = FN_{i+1} + Gg_{i+1} + HM_i \qquad \text{equation A.5}$$

Transformation of Generated Data

$$Q_i = Tr\ (N_i) \qquad \text{equation A.6}$$

$$S_i = Tr\ (M_i) \qquad \text{equation A.7}$$

Adjustment of Generated Data

$$Q_i = IS_i \qquad \text{equation A.8}$$

Upper Case Symbols:

A - Coefficient matrix (dimensions d1 by d1) in equations A.1, A.2, and A.3
(Note d1 and the other dimension symbols will be defined later)
B - Coefficient matrix (d1 by d1) in equations A.2 and A.3
C - Coefficient matrix (d1 by d1) in equations A.1, A.2, and A.3
C - Coefficient matrix (d2 by d3) in equation A.4
(Note that the symbol C is used for two different matrices)
D - Coefficient matrix (d2 by d2) in equation A.4
E - Coefficient matrix (d2 by d2) in equation A.4
F - Coefficient matrix (d4 by d5) in equation A.5
G - Coefficient matrix (d4 by d4) in equation A.5
H - Coefficient matrix (d4 by d4) in equation A.5
I - Coefficient matrix (one by d6) in equation A.8
K - Normalized key station annual data matrix (d3 by one) in equation A.4
M - Normalized seasonal data matrix (d4 by one) in equation A.5
M - Normalized seasonal data matrix (d6 by one) in equation A.7
M_0 - Lag zero moment matrix (d7 by d7)
M_1 - Lag one moment matrix (d7 by d7)
M_2 - Lag two moment matrix (d7 by d7)
N - Normalized substation annual data matrix (d2 by one) in equation A.4
N - Normalized annual data matrix (d5 by one) in equation A.5
N - Normalized annual data matrix (one by one) in equation A.6
Q - Annual data matrix (one by one) in equation A.6
S - Seasonal data matrix (d6 by one) in equations A.7 and A.8
T - Transformed data matrix (d7 by one)
X - Normalized key station annual data matrix (d1 by one) in equations A.1, A.2, and A.3
Z - Matrix equal to matrix product CC^T Lower Case Symbols:

Lower Case Symbols:

a - Element of matrix A
b - Element of matrix B
cc^T - Element of matrix product CC^T
e - Column matrix of random normal numbers (d1 by one) having zero mean and unity variance in equations A.1, A.2, and A.3
f - Column matrix of random normal numbers (d2 by one) having zero mean and unity variance in equation A.4
g - Column matrix of random normal numbers (d4 by one) having zero mean and unity variance in equation A.5
i - Subscript indicating observation year
m_O - Element of matrix M_O

m_1 - Element of matrix M_1
m_2 - Element of matrix M_2

Greek Symbols:

μ - Mean value
ρ - Serial correlation coefficient
σ - Standard deviation
τ - Seasonal index or subscript

Transforms:

Tr - Used in equations A.6 and A.7

Dimensions:

d1 - Number of key stations in the current key station generation group
d2 - Number of substations in the current key to substation generation group
d3 - Number of key stations in the current key to substation generation group
d4 - Number of stations times number of seasons in the current annual to seasonal generation group
d5 - Number of stations in the current seasonal generation group
d6 - Number of seasons in one year
d7 - Number of stations

B. Key Station Generation

The approaches used herein are rather straightforward. The derivation of the parameter estimation formulas for the key station generation will be covered in detail. The parameter estimation formulas for the other generation equations will be derived in a later section with much less detail. Three linear AR (autoregressive) models are available for key station generation. With each AR alternative, either lag one or lag two correlations may be preserved directly. All of the approaches start with annual data which have been transformed into normally distributed data with a mean of zero. That is, the original data, q, have been transformed by program TRNPAR into data, t, which are normally distributed but with a nonzero mean. These data are used by program KEYPAR to estimate the parameters required. For the derivation of the parameters, it is more convenient to work with data having a mean of zero given by:

$$x_i = t_i - \mu \qquad \text{equation B.1}$$

where μ is the population mean value.

Lag one full Matalas. - The first alternative is based on Matalas (1967). The basic equation for which parameters are to be estimated is:

$$X_i = AX_{i-1} + C\, e_i \qquad \text{equation B.2}$$

Where A and C are parameter matrices. Postmultiplying by X^T_{i-1}
(a capital T superscript indicates the transpose of a matrix)

$$X_i X^T_{i-1} = AX_{i-1} X^T_{i-1} + Ce_i X^T_{i-1} \qquad \text{equation B.3}$$

Taking expected values (denoted by E [.]), the last term drops out and we obtain

$$E [X_i X^T_{i-1}] = A E [X_{i-1} - X^T_{i-1}] \qquad \text{equation B.4}$$

Defining M_O and M_1 as

$$M_O = E [X_{i-1} X^T_{i-1}] \qquad \text{equation B.5}$$

and

$$M_1 = E [X_i X^T_{i-1}] \qquad \text{equation B.6}$$

then

$$M_1 = AM_0 \qquad \text{equation B.7}$$

Solving for parameter matrix A, we obtain

$$A = M_1 M_0^{-1} \qquad \text{equation B.8}$$

Note that a superscript "-1" indicates a matrix inverse. Whenever possible, an inverse computation should be avoided because of possible numerical round off errors. This is particularly true for large matrices. In this package, matrix inversions are held to an absolute minimum and, to minimize any problems, matrices are held down to a relatively small size.

Matrix C is now determined by postmultiplying equation B.2 by X_i^T which gives

$$X_i X^T_i = AX_{i-1} X_i + Ce_i X^T_i \qquad \text{equation B.9}$$

Taking expected values

$$E [X_i X^T_i] = A E [X_{i-1} X^T_i] + CE [e_i X^T_i] \qquad \text{equation B.10}$$

Noting that e_i is a vector of independent random normal numbers with mean of zero and unit variance the last term of equation B.10 evaluated as

$$C E [e_i X^T_i] = C E [e_i(AX_{i-1} + Ce_i)^T] \qquad \text{equation B.11}$$

$$= C E [e_i X^T_{i-1} A^T + e_i e^T_i C^T]$$

$$= CC^T$$

Evaluating equation B.10, we now obtain

$$M_0 = AM^T{}_1 + CC^T \qquad \text{equation B.12}$$

This may be written as

$$CC^T = M_0 - AM^T{}_1 \qquad \text{equation B.13}$$

Insert the solution for A given by equation B.8

$$CC^T = M_0 - M_1\, M_0^{-1}\, M_1^T \qquad \text{equation B.14}$$

Any solution for C which will reproduce CC^T is a valid solution. There exist an infinite number of solutions which will reproduce CC^T. A particular easy solution can be obtained by assuming C is a lower triangular matrix. A solution of this form has been proposed by Young and Pisano (1968). Their solution was good for all CC^T which are positive definite. Here we will use a solution which has the same lower triangular form as the Young and Pisano approach but will also work for all CC^T which are singular (i.e., all CC^T which are positive semidefinite). For a positive definite CC^T, the solution reduces identically to that of Young and Pisano. First we set

$$Z = CC^T \qquad \text{equation B.15}$$

to simplify the notation. Then the values for the elements of C are given by

$$c(k,i) = O \quad \text{for all } k < i \qquad \text{equation B.16}$$

$$c(k,i) = O \quad \text{for all } k \geq i \quad \text{when}$$

$$z(i,i) - \sum_{j<i} [c(i,j)\ c(i,j)] \leq 0 \qquad \text{equation B.17}$$

and

$$c(k,i) = \frac{z(k,i) - \sum_{j<i} [c(i,j)\ c(k,j)]}{\sqrt{z(i,i) - \sum_{j<i} [c(i,j)\ c(i,j)]}}$$

for all $k \geq i$

when

$$z(i,i) - \sum_{j<i} [c(i,j)\ c(i,j)] > 0$$

equation B.18

Equations B.16, B.17, and B.18 are applied first to calculate the first column, top to bottom, then the second column, third, etc. Equations B.16 and B.18 are all that are required for the solution of a positive definite CC^T. They provide the same solution as Young and Pisano. It can easily be seen that CC^T must always be at least positive semidefinite. However, due to computer round off errors, it is common for a singular matrix to appear to be negative definite. The modification made by equation B.17 successfully overcomes this predicament in addition to handling the singular case. This solution will be used in later sections to solve similar matrix products of the same form.

<u>Lag two full Matalas</u>. - The basic generation equation has the form

$$X_i = A\,X_{i-1} + B\,X_{i-2} + C\,e_i \qquad \text{equation C.1}$$

where A, B, and C are parameter matrices

Postmultiplying by X^T_{i-1}

$$X_i\,X^T_{i-1} = A\,X_{i-1}\,X^T_{i-1} + B\,X_{i-2}\,X^T_{i-1} + C\,e_i\,X^T_{i-1} \qquad \text{equation C.2}$$

Taking expected values

$$E\,[X_i\,X^T_{i-1}] = A\,E\,[X_{i-1}\,X^T_{i-1}] + B\,E\,[X_{i-2}\,X^T_{i-1}] \qquad \text{equation C.3}$$

Now, as in equations B.5 and B.6, defining

$$M_0 = E\,[X_{i-1}\,X^T_{i-1}] \qquad \text{equation C.4}$$

and

$$M_1 = E\,[X_i\,X^T_{i-1}] = E\,[X_{i-1}\,X^T_{i-2}] \qquad \text{equation C.5}$$

Also defining

$$M_2 = E\,[X_i\,X^T_{i-1}] \qquad \text{equation C.6}$$

Transposing equation C.5

$$M^T_1 = E\,[X_{i-2}\,X^T_{i-1}] \qquad \text{equation C.7}$$

Substituting these into equation C.3, we obtain

$$M_1 = A\,M_0 + B\,M^T_1 \qquad \text{equation C.8}$$

Now by postmultiplying equation C.1 by X^T_{i-2}, we obtain

$$X_1\,X^T_{i-2} = A\,X_{i-1}\,X^T_{i-2} + B\,X_{i-2}\,X^T_{i-2} + C\,e_i\,X^T_{i-2} \qquad \text{equation C.9}$$

Taking expected values and simplifying

$$M_2 = A\,M_1 + B\,M_0 \qquad \text{equation C.10}$$

Solving for equation C.8 for A, we obtain

$$A = (M_1 - B\, M^T{}_1\,)\, M_0^{-1} \qquad \text{equation C.11}$$

Substituting equation C.11 into equation C.10

$$M_2 = M_1\, M_0^{-1} \quad M_1 - B\, M_1^T \;\; M_0^{-1} \quad M_1 + B\, M_0 \qquad \text{equation C.12}$$

Solving for B gives

$$B = (M_2 - M_1\, M_0^{-1}\, M_1)(M_0 - M^T{}_1\, M_0^{-1}\, M_1)^{-1} \qquad \text{equation C.13}$$

Thus with B solved for, A is given by substituting the solution for B into equation C.11. Note that although A may easily be solved for either as

$$A = (M_1 - (M_2 - M_1\, M_{0\text{-}1}\, M_1)(M_0 - M^T{}_1\, M_0^{-1}\, M_1)^{-1}\, M^T{}_1)M_0^{-1} \qquad \text{equation C.14}$$

or

$$A = (M_2 - M_1\, M_1^{T\text{-}1}\, M_0)\,(M_1 - M_0\, M_1^{T\text{-}1}\, M_0)^{-1} \qquad \text{equation C.15}$$

It is easier computationally to solve first for B using equation C.13 and then to substitute that solution into equation C.11 to obtain A.

Now we proceed to solve for matrix C. Postmultiplying equation C.1

by $X^T{}_i$,

$$X_i\, X^T{}_i = A\, X_{i\text{-}1}\, X^T{}_i + B\, X_{i\text{-}2}\, X^T{}_i + C\, e_i\, X^T{}_i \qquad \text{equation C.16}$$

$$E\,[X_i\, X^T{}_i] = A\, E\,[X_{i\text{-}1}\, X^T{}_i] + B\, E\,[X_{i\text{-}2}\, X^T{}_i] + C\, E\,[e_i\, X^T{}_i] \qquad \text{equation C.17}$$

Noting that

$$E[e_i X_i^T] = C^T \qquad \text{equation C.18}$$

We obtain

$$M_0 = A\, M^T{}_1 + B\, M^T{}_2 + CC^T \qquad \text{equation C.19}$$

Solving for CC^T,

$$CC^T = M_0 - A\, M^T{}_1 - B\, M^T{}_2 \qquad \text{equation C.20}$$

Matrix C is solved for by using the value of CC^T from equation C.20 and applying equations B.16, B.17, and B.18.

An alternate approach for solving for A, B, and C may be developed by rewriting equation C.1 as

$$X_i = [A \;\; B]\begin{bmatrix} X_{i-1} \\ X_{i-2} \end{bmatrix} + C \; e_i \qquad \text{equation C.21}$$

When written in this form, the solution proceeds in exactly the same manner as for the lag one case. The solution that results is exactly the same as that obtained through the more detailed approach shown here. This exercise is left to the reader. Note that many of the linear stochastic models (including some disaggregation models) may be written in the same general form:

$$X_i = A \; Y_i + C \; e_i \qquad \text{equation C.22}$$

and the parameters may be solved for as

$$A = E \; [X_i \; Y_i^T] \; E \; [Y_i \; Y_i^T]^{-1} \qquad \text{equation C.23}$$

and as a matrix C which satisfies

$$CC^T = E \; [X_i X_i^T] - A \; E \; [Y_i \; X_i^T] \qquad \text{equation C.24}$$

This form of solution is identical to that found by the more detailed approach. The only argument for avoiding this elegant solution is that some feel is lost for the moments involved. It does provide an excellent approach for checking solutions.

<u>Lag one Matalas with lagged cross correlations ignored</u>. - The basic equation is equation B.3. In this case, only lag one same series correlations are directly preserved. The values for the elements of matrix A are given by

$$a(i,j) = 0.0 \text{ for i not equal to j} \qquad \text{equation D.1}$$

and

$$a(i,i) = \rho_i(1) \qquad \text{equation D.2}$$

where $\rho_i(1)$ is the lag one serial correlation coefficient for station "i." An adjusted or calculated M_1 can now be manufactured (not estimated from data) using the just estimated matrix A and the lag zero moment matrix M_0.

$$M_1 = A \; M_0 \qquad \text{equation D.3}$$

It can be noted that this adjusted M_1 matrix is what will be preserved in the generated data. The adequacy of ignoring lag one cross correlations may be judged by a comparison of the adjusted M_1 matrix and an M_1 matrix estimated from the data. Using the adjusted M_1 matrix, CC^T can be estimated as before using

$$CC^T = M_0 - A \; M^T_1 \qquad \text{equation D.4}$$

Matrix CC^T may also be estimated without calculating an adjusted M_1.

$$CC^T = M_0 - A\, M^T_0\ A^T \qquad \text{equation D.5}$$

C is solved for using this CC^T as input to equations B.16, B.17, and B.18.

<u>Lag two Matalas with lagged cross correlations ignored</u>. - The basic equation is equation C.1. This case is similar to the last except that it is complicated by a second lagged term. Elements of matrix A are given by

$$a(i,j) = 0.0 \text{ for i not equal to j} \qquad \text{equation E.1}$$

and

$$a(i,i) = \frac{\rho_i(1) - \rho_i(1)\ \rho_i(2)}{1 - \rho_i(1)\rho_i(1)} \qquad \text{equation E.2}$$

Elements of Matrix B are given by

$$b(i,j) = 0.0 \text{ for i not equal to j} \qquad \text{equation E.3}$$

and

$$b(i,i) = \frac{\rho_i(2) - \rho_i(1)\ \rho_i(1)}{1 - \rho_i(1)\ \rho_i(1)} \qquad \text{equation E.4}$$

The ρ_i's are serial correlation coefficients, at the i-th station; the values in parenthesis are the lag.

As in the previous case CC^T is solved for through the use of altered M_1 and M_2 moment matrices. The easiest way to calculate an altered M_1 and M_2 is to examine equation C.8. Since A and B are diagonal matrices, equation C.8 may be written in terms of its elements as

$$m_1(i,j) = a(i,i)\ m_0(i,j) + b(i,i)\ m_1(j,i) \qquad \text{equation E.5}$$

also

$$m_1\ (j,i) = a(j,j)\ m_0\ (j,i) + b(j,j)\ m_1\ (i,j) \qquad \text{equation E.6}$$

So, by substituting equation E.6 into equation E.5, we can solve for m_1 (i,j) as

$$m_1\ (i,j) = \frac{a(i,i)\ m_0\ (i,j) + b(i,i)\ a(j,j)\ m_0\ (j,i)}{1 - b(i,i)\ b(j,j)} \qquad \text{equation E.7}$$

or

$$m_1\ (i,j) = \frac{[a(i,i) + b(i,i)\ a(j,j)]\ m_0\ (i,j)}{1 - b(i,i)\ b\ (j,j)} \qquad \text{equation E.8}$$

Thus equation E.8 is used to calculate an altered M_1. We can now easily calculate an altered M_2 from equation C.10, using the altered M_1 just calculated:

$$M_2 = A\ M_1 + B\ M_0 \qquad \text{equation E.9}$$

and CC^T from

$$CC^T = M_0 - AM^T{}_1 - B\ M^T{}_2 \qquad \text{equation E.10}$$

where M_1 and M_2 are the altered matrices. Again, C is calculated from the solution for CC^T using equation B.16, B.17, and B.18.

<u>Decoupled lag one</u>. - The basic equation is equation B.2. The values for matrix A are the same as for the lag one Matalas with lagged cross correlations ignored (equations D.1 and D.2). Serially independent but spatially dependent numbers are arrived at as

$$Ce_i = X_i - A\ X_{i-1} \qquad \text{equation E.11}$$

Postmultiplying both sides of equation E.11 by themselves transposed, we obtain

$$Ce_i\ (Ce_i)^T = (X_i - A\ X_{i-1})\ (X_1 - AX_{i-1})^T \qquad \text{equation E.12}$$

Taking expected values

$$CC^T = M_0 - A\ M^T{}_1 - M_1\ A^T + A\ M_0\ A^T \qquad \text{equation E.13}$$

Again C is solved for using equations B.16, B.17, and B.18. This approach has the advantage of separating the serial and spatial correlations. One may remove the serial correlations from a set of series and then examine the spatial correlations. It does have the disadvantage that the observed M_0 is not preserved exactly. (The diagonal elements of M_1 are preserved while as expected the nondiagonal elements are not. The serial correlations are not preserved.) It should be noted that if the observed data exactly bear out the assumption of no lag one cross causal structure, then the resulting solution reduces exactly to that given for the lag one Matalas approach with lagged cross correlations ignored. For the observed data to exactly bear out the assumption of no lag one cross causal structure, the observed M_0 and M_1 would have to satisfy the equation

$$M_1 = AM_0 \qquad \text{equation E.14}$$

when A is given by equations D.1 and D.2.

In most cases, the errors in the generated M_0 and M_1 are very minor. In order to quantify the errors, we may calculate the moment matrices which will be preserved by the equations in the generated data. Rewriting equation D.5 in terms of generated data (with an M^G to indicate the expected values of a generated moment matrix).

$$M^G{}_0 = A\ M^G{}_0\ A^T + CC^T \qquad \text{equation E.15}$$

Since A is diagonal and M^G_0 is symmetric, the elements of M^G_0 may be solved for as

$$m^G_0\,(i,j) = cc^T\,(i,j)/[1 - a(i,i)\;a(j,j)] \qquad \text{equation E.16}$$

The expected generated M^G_1 is easily calculated as

$$M_1^G = A\;M_0^G \qquad \text{equation E.17}$$

Comparing M_1^G and M_0^G with the historical estimates for M_1 and M_0 gives a basis for testing the adequacy of this approach. The decoupled approach (both lag one and lag two forms) has been advocated by several authorities. The main advantage, that of treating the serial and cross correlations separately, is not utilized in this package. The decoupled approaches are included mainly for comparison purposes.

Decoupled lag two. - The basic equation is equation C.1. The values for matrix A are given by equations E.1 and E.2. The values for matrix B are given by equations E.3 and E.4. As in the last case, the serially independent but spatially dependent numbers are arrived at by rewriting the basic equation, equation C.1

$$Ce_i = X_i - AX_{i-1} - BX_{i-2} \qquad \text{equation F.1}$$

Postmultiplying both sides of equation F.1 by themselves transposed, we obtain

$$Ce_i(Ce_i)^T = (X_i - AX_{i-1} - BX_{i-2})\;(X_i - AX_{i-1} - BX_{i-2})^T \qquad \text{equation F.2}$$

Taking expected values

$$CC^T = M_0 - A\;M^T{}_1 - B\;M^T{}_2 - M_1A^T - M_2B^T + AM_0A^T + A\;M_1\;B^T + B\;M_1\;A^T + B\;M_0\;B^T$$

equation F.3

Again C is solved for using equations B.16, B.17, and B.18. This approach has the same advantages as the decoupled lag one approach. A simple solution for the generated M_O, M_1, and M_2 matrices has not been found. They can be found by iteratively applying equations C.8, C.10, and C.19.

Autoregressive Moving Average. The basic ARMA(1,1) model is given by

$$X_i = AX_{i-1} + B\;e_i + C\;e_{i-1} \qquad \text{equation G.1}$$

(This model presently is not available but will be added at a later time.)

C. Key Substation Disaggregation

This approach is somewhat similar to a disaggregation scheme proposed by Valencia and Schaake (1973) and modified by Mejia and Rousselle (1976). That scheme was used for disaggregating annual data into seasonal data. By seasonal data we mean data recorded in periods which are subunits of a year such as monthly, weekly, biweekly, etc.

The approach used here is to disaggregate annual data at key stations into annual data at substations. In reality, the substations are generated dependent on previously generated key stations. They do not have to sum into the key station(s). If they do actually sum into key station, this attribute can be preserved in the disaggregated data. As with the key station generation, transformed data which follow the normal distribution are used. Since transformed data are utilized, it is doubtful that the substations would sum to give the key station. The substations may be generated dependent on more than one key station.

The equations will be derived in terms of standardized (and normalized) data having zero means. The data need not have unit variance. The data used are given by

$$k_i = t_i - \mu \qquad \text{equation H.1}$$

and

$$n_i = t_i - \mu \qquad \text{equation H.2}$$

where k and n are standardized data for key and substations, t are transformed data with normal distribution, and μ is the population mean for the t data.

The basic generation equation for which parameters are to be estimated is

$$N_{i+1} = C\ K_{i+1} + D\ f_{i+1} + E\ N_i \qquad \text{equation H.3}$$

where N is a vector containing the substations and K is a vector containing the key stations.

Moments are denoted using a different notation as follows:

$$S_{ab}(i\text{-}j) = E\ a_i\ b_j^T \qquad \text{equation H.4}$$

For example, with i = 0 and j = 1, $S_{ab}(1)$ is the lag one covariance matrix between vectors a and b.

Postmultiplying equation H.3 by $K^T{}_{i+1}$ and taking expected values, we obtain

$$S_{NK}(0) = CS_{KK}(0) + ES^T\ KN(1) \qquad \text{equation H.5}$$

Likewise, by postmultiplying equation H.3 by $N^T{}_i$ and taking expected values, we obtain

$$S_{NN}(1) = CS_{KN}(1) + E\ S_{NN}(0) \qquad \text{equation H.6}$$

Equation H.6 may be rewritten as

$$E = [S_{NN}(1) - C\ S_{KN}(1)]\ S_{NN}^{-1}(0) \qquad \text{equation H.7}$$

Substituting equation H.7 into equation H.5 and solving for matrix C, we obtain

$$C = [S_{NK}(0) - S_{NN}(1)\, S_{NN}^{-1}(0)\, S^{T}_{KN}(1)]\, [S_{KK}(0) - S_{KN}(1)\, S_{NN}^{-1}(0)\, S^{T}_{KN}(1)]^{-1} \qquad \text{equation H.8}$$

which is used to provide estimates for matrix C from sample moments. Using the estimate for matrix C, matrix E may be solved for using equation H.7.

The estimate for matrix D may be obtained by postmultiplying equation H.3 by N^{T}_{i+1} and taking expected values. This results in

$$S_{NN}(0) = CS_{KN}(0) + DD^{T} + ES^{T}_{NN}(1) \qquad \text{equation H.9}$$

From which matrix DD^{T} may be solved for as

$$DD^{T} = S_{NN}(0) - CS_{KN}(0) - ES^{T}_{NN}(1) \qquad \text{equation H.10}$$

Again, matrix D is solved for matrix DD^{T} as outlined by equations B.16, B.17, and B.18.

A careful examination of the equations used for estimation reveals that equations H.5 and H.6 make use of the moment $S_{KN}(1)$. This moment has no corresponding dependence structure in the disaggregation model or in the key station generation. In fact upon generation, it is found that $S_{KN}(1)$ cannot be preserved. Because of this model inadequacy, all of the generated moments are adversely affected. Both $S_{KN}(1)$ and $S_{NN}(1)$ are poorly preserved. This should have been expected since the model structure does not correspond to the moments used in estimation. The proper approach is to replace the sample estimate for $S_{KN}(1)$ by the value for $S_{KN}(1)$ which would theoretically result from the model structure. To do this some assumption must be made as to the model (or more rightly the correlation structure) of the key stations. The easiest and perhaps most generally valid assumption is that of the full Matalas lag one linear autoregressive model, AR(1) model. In most cases, even if the key station model were not AR(1), this assumption would cause very little error. Under this assumption, it can easily be shown that the generated $S_{KN}(1)$ will be

$$S_{KN}(1) = A\, S_{KK}(0)\, C^{T} + A\, S_{KN}(1)\, E^{T} \qquad \text{equation H.11}$$

where A is the parameter matrix for the key station AR(1) generation. C and E are parameter matrices for the spatial disaggregation model. Evaluating these matrices and simplifying we obtain

$$S^{*}_{KN}(1) = S_{KK}(1)\, S^{-1}_{KK}(0)\, S_{KN}(0) \qquad \text{equation H.12}$$

This value, $S^{*}_{KN}(1)$, should be used in equations rather than the sample $S_{KN}(1)$ value. Theoretically, the disaggregation model will directly preserve $S_{NN}(1)$, $S_{NN}(0)$, and $S_{NK}(0)$, the key station generation will directly preserve $S_{KK}(0)$ and $S_{KK}(1)$, and $S^{*}_{KN}(1)$ will be preserved as given by equation H.12 indirectly through the model structure. However, in practice, it is often found that the parameter matrices cannot be solved for and also that additivity, if present in the data, is no longer preserved. For example, the matrix DD^{T} can easily turn out to not be positive semidefinite. The reason for these problems is that the new $S_{KN}(1)$ now is not consistent with the sample estimate of $S_{NN}(1)$. The easiest method to make $S_{NN}(1)$ consistent and to ensure that any additivity is properly preserved is to estimate a new $S_{NN}(1)$ as

$$S^*_{NN}(1) = S_{NN}(1) + S_{NK}(0)\, S_{KK}^{-1}(0)\, [S^*_{KN}(1) - S_{KN}(1)] \qquad \text{equation H.13}$$

The superscript * indicates the adjusted moments. All other moments are sample moments. The use of the adjusted $S_{KN}(1)$ and $S_{NN}(1)$, equations H.12 and H.13, will ensure additivity preservation and generational stability.

In the program, the normal run is using both moment adjustments. Optionally the user may elect to estimate parameters without these adjustments. There also is an option for ignoring the lagged substation covariances. In this case two approaches may be taken, either of which produce identical parameter estimates. Both result in parameter matrix E being a null matrix. The first approach is to set E to null in equation H.5 and solve for matrix C as

$$C = S_{NK}(0) S_{KK}^{-1}(0) \qquad \text{equation H.14}$$

The matrix D is then easily solved for from DD^T using equation H.10 where matrix E has to be set to null.

$$DD^T = S_{NN}(0) - CS_{KN}(0) \qquad \text{equation H.15}$$

The second approach is to adjust the moment matrix $S_{NN}(1)$ to

$$S^*_{NN}(1) = S_{NK}(0)\, S_{KK}^{-1}(0)\, S_{KN}(1) \qquad \text{equation H.16}$$

and to solve for the parameter matrices as originally. This second approach is that used in the program.

As a final note on adjusted moment matrices, if the key station generation is an AR(2) model, the adjusted $S_{KN}(1)$ is given as

$$S^*_{KN}(1) = AS_{KN}(0) + BS^T_{KK}(1)\, C^T + BS_{KN}(0)\, E^T \qquad \text{equation H.17}$$

where A and B are parameter matrices of the key station AR(2) model and C and E are parameter matrices of the spatial disaggregation.

The theoretic moments preserved upon generation can be obtained by iteratively applying equations H.5, H.6, H.9, and H.12.

D. <u>Annual-Seasonal Generation</u>

The approach taken here closely follows the disaggregation scheme proposed by Valencia and Schaake (1973). That scheme has been extended by Mejia and Rousselle (1976). The approach of Valencia and Schaake has also been applied by Tao and Delleur (1976). These references provide much in the way of background material.

Full Model

Again, the data are first transformed to follow the normal probability distribution and then adjusted to have zero means. For the seasonal data

$$m_{i,\tau} = t_{i,\tau} - \mu \qquad \text{equation I.1}$$

where m is the transformed seasonal data adjusted to zero mean and t is the seasonal data after transformation, i is the year and τ refers to the season. For the annual data

$$n_i = t_i - \mu \qquad \text{equation I.2}$$

where n is the transformed annual data adjusted to zero mean, and t is the annual data after transformation. The basic equation is

$$M_{i+1} = FN_{i+1} + G\, g_{i+1} + H\, M_i \qquad \text{equation I.3}$$

where i refers to the year.

First, the parameters' estimates will be developed for this formulation and then two improved formulations will be examined. Only the third formulation, referred to as the condensed model, will be used in the computer program.

Using the previously introduced moment notation,

$$S_{ab}(i - j) = E(a_i\, b_j^T) \qquad \text{equation I.4}$$

and postmultiplying equation I.3 by N_{i+1} and taking expected values we obtain

$$S_{MN}(0) = F\, S_{NN}(0) + H\, S^T_{NM}(1) \qquad \text{equation I.5}$$

Postmultiplying equation I.3 by M^T_i and taking expected values, we obtain

$$S_{MM}(1) = F\, S_{NM}(1) + H\, S_{MM}(0) \qquad \text{equation I.6}$$

Solving equation I.6 for H, we obtain

$$H = [S_{MM}(1) - F\, S_{NM}(1)]\, S^{-1}_{MM}(0) \qquad \text{equation I.7}$$

Substituting equation I.7 into equation I.5, F may be solved for as

$$F = [S_{MN}(0) - S_{MM}(1)\, S^{-1}_{MM}(0)\, S^T_{NM}(1)]\, [S_{NN}(0) - S_{NM}(1)\, S^{-1}_{MM}(0)\, S^T_{NM}(1)]^{-1}$$

equation I.8

Now matrix F has been solved for in terms of moments which may be estimated from the available data. Matrix H may be also solved for using equation I.7 with the already estimated value for matrix F.

Now to estimate matrix G, we postmultiply equation 1.3 by M^T_{i+1} and take expected values obtaining

$$S_{MM}(0) = F\, S_{NM}(0) + GG^T + H\, S^T_{MM}(1) \qquad \text{equation I.9}$$

This is easily solved for GG^T as

$$GG^T = S_{MM}(0) - F\, S_{NM}(0) - H\, S^T_{MM}(1) \qquad \text{equation I.10}$$

from which matrix G may be determined using equations B.16, B.17, and B.18.

Again, for this disaggregation model, it can be noted that one of the moment matrices, $S_{NM}(1)$, has no corresponding structure in the model by which to be preserved. Upon generation, one finds that $S_{NM}(1)$ and $S_{MM}(1)$ are very poorly preserved. Of course, all generated moments are adversely affected. To overcome this, the sample estimate for $S_{NM}(1)$ may be replaced in equations I.7 and I.8 by

$$S^*_{NM}(1) = S_{NN}(1)\, S_{NN}^{-1}(0)\, S_{NM}(0) \qquad \text{equation I.11}$$

This modification will ensure the preservation of all the other moment matrices, but unfortunately destroys any additivity properties of the model. As with the spatial disaggregation, an additional adjustment to a second moment matrix is required. In this case, the moment matrices $S_{MM}(1)$ must be adjusted as

$$S^*_{MM}(1) = S_{MM}(1) + S_{MN}(0)\, S_{NN}^{-1}(0)\, [S^*_{NM}(1) - S_{NM}(1)] \qquad \text{equation I.12}$$

If one desires to use the given solution technique, but to also ignore lag one seasonal moments, i.e., set matrix H to null, $S_{MM}(1)$ may be adjusted as

$$S^*_{MM}(1) = S_{MN}(0)\, S_{NN}^{-1}(0)\, S_{NM}(1) \qquad \text{equation I.13}$$

The theoretic moments preserved upon generation can be obtained by iteratively applying equations I.5, I.6, I.9, and I.11.

<u>Limited Model</u>

In actual application, the Full Model approach is wasteful in terms of computer storage and preserves more moments than are really desirable. A logical modification is to change the basic equation to

$$M_{i+1} = F\, N_{i+1} + Gg_{i+1} + H\, Z_i \qquad \text{equation I.14}$$

where Z is a matrix of the M matrix values for only the last season of the previous year. In this case, the solutions are slightly modified. Equation I.7 becomes

$$H = [S_{MZ}(1) - F\ S_{NZ}\ (1)]\ S_{ZZ}^{-1}(O) \qquad \text{equation I.15}$$

The solution for matrix F becomes

$$F = [S_{MN}(0) - S_{MZ}(1)\ S_{ZZ}^{-1}(0)\ S^{T}_{NZ}(1)]\ [S_{NN}(0) - S_{NZ}(1)\ S_{ZZ}^{-1}(0)\ S^{T}_{NZ}(1)]^{-1} \qquad \text{equation I.16}$$

and the solution for matrix GG^T becomes

$$GG^T = S_{MM}(0) - F\ S_{NM}(0) - H\ S^{T}_{MZ}(1) \qquad \text{equation I.17}$$

As with the other disaggregation models, one of the moment matrices used in estimation has no corresponding model structure and should be reestimated. In this case, moment $S_{NZ}(1)$ should be replaced by

$$S^{*}_{NZ}(1) = S_{NN}(1)\ S^{-1}_{NN}(0)\ S_{NZ}(0) \qquad \text{equation I.18}$$

In order to preserve additivity and ensure valid solutions for the parameter matrices, $S_{MM}(1)$ should be adjusted as

$$S^{*}_{MZ}(1) = S_{MZ}(1) + S_{MN}(O)\ S_{NN}^{-1}(0)\ [S^{*}_{NZ}(1) - S_{NZ}(1)] \qquad \text{equation I.19}$$

If one desires to set matrix H to null, $S_{MM}(1)$ may be adjusted as

$$S^{*}_{MZ}(1) = S_{NN}(0)\ S_{NN}^{-1}(0)\ S_{NZ}(1) \qquad \text{equation I.20}$$

The theoretic moments preserved upon generation can be obtained by iteratively applying equation I.18 and the easily obtained equations which were the basis for I.15, I.16, and I.17.

<u>Condensed Model</u>

Noting that equation I.11 still preserves more than lag one correlations for all months except the first month of the year, and noting that the number of lagged correlations being preserved is not consistent, a further improvement will be made. In order to bring about consistency, a single lag model will be designed. The basic equation is now written in terms of seasons as

$$M_\tau = F_\tau\ N_\tau + G_\tau\ g_\tau + H_\tau\ M_{\tau-1} \qquad \text{equation I.21}$$

This equation is a version of the basic equation (equation I.3) in which many of the parameters in matrices F, H, and G have been set to zero. The notation deserves some explanation at this point because in the interests of minimizing the number of sub and superscripts, it has become somewhat cryptic. The Greek subscript tau, τ , denotes the current season being generated. Thus, M_τ is only a portion of matrix M. Matrix M contains the whole year's seasonal data while M_τ

contains only one season's data. N_τ is a matrix of annual data for the year to which the current season's data belong. $M_{\tau-1}$ is a matrix of the previous season's data (which would be the last season from the previous year if the current season is the first season of a year).

If data are monthly, values for τ cycle from 1 to 12 and then repeat. Matrix g_τ is a column matrix of random normal numbers. Matrices M_τ, N_τ, g_τ, and $N_{\tau-1}$ also should have additional notation as to the year of data. To simplify notation, this has been intentionally left out. Since the seasonal subscripts imply the time sequence and the yearly subscripts are fairly obvious, this should not create any problems.

The parameter matrices F_τ, G_τ, and H_τ have one set of values for each season. In terms of the original basic equation, I.3, these parameter matrices are only a small portion of the total F, G, and H matrices. As a result, this modification results in a tremendous savings of computer storage because of this large reduction in the number of parameters. This form is the one actually used by this package of computer programs.

In order to estimate the parameters, it is well to introduce some new notation for moments:

$$S_{ab}(\tau, \tau - j) = E(a_\tau b^T_{\tau-j}) \qquad \text{equation I.22}$$

Postmultiplying equation I.21 by the transpose of N_τ and taking expected values we obtain:

$$S_{MN}(\tau, \tau) = F_\tau S_{NN}(\tau, \tau) + H_\tau S^T_{NM}(\tau, \tau - 1) \qquad \text{equation I.23}$$

Postmultiplying equation I.21 by the transpose of $M_{\tau-1}$, we obtain

$$S_{MM}(\tau, \tau - 1) = F_\tau S_{NM}(\tau, \tau - 1) + H_\tau S_{MM}(\tau - 1, \tau - 1) \qquad \text{equation I.24}$$

Solving equation I.24 for H_τ, we obtain

$$H_\tau = [S_{MM}(\tau, \tau - 1) - F_\tau S_{NM}(\tau, \tau - 1)]\, S_{MM}^{-1}(\tau - 1, \tau - 1) \qquad \text{equation I.25}$$

Substituting equation I.25 into equation I.23, we obtain a solution for F_τ in terms of moments alone

$$F_\tau = S_{MN}(\tau, \tau) - S_{MM}(\tau, \tau - 1) S_{MM}^{-1}(\tau - 1, \tau - 1) S_{MN}(\tau - 1, \tau)$$

$$S_{NN}(\tau, \tau) - S_{NM}(\tau, \tau - 1) S_{MM}^{-1}(\tau - 1, \tau - 1) S_{MN}(\tau - 1, \tau)^{-1}$$

equation I.26

Having solved for F_τ, H_τ is obtained by using the solution for F_τ in equation I.19. In order to solve for F_τ, we first postmultiply equation I.21 by the transpose of M_τ and, after taking expected values, obtain

$$S_{MM}(\tau, \tau) = F_\tau S_{NM}(\tau, \tau) + GG^T_\tau + H_\tau S_{MM}(\tau - 1, \tau) \qquad \text{equation I.27}$$

This is easily solved for GG^T_τ as

$$GG^T_\tau = S_{MM}(\tau, \tau) - F_\tau \; S_{NM}(\tau, \tau) - H_\tau \; S_{MM}(\tau - 1, \tau) \qquad \text{equation I.28}$$

Again matrix G_τ is solved for from matrix GG^T_τ as outlined earlier by equations B.16, B.17, and B.18.

As with the other disaggregations, a careful examination of the equations reveals that a moment is used for estimation which does not have a corresponding dependence structure in the model. In this case, moment $S_{NM}(\tau, \tau - 1)$ in equations I.23 and I.24 could potentially be a problem. The problem only exists for the first season of each year. For that one season, $\tau = 1$, adjusted values for $S_{NM}(\tau, \tau - 1)$ and $S_{MM}(\tau, \tau - 1)$ should be calculated and used in equations I.23 and I.24. These adjusted matrices may be written as

$$S^*_{NM}(\tau, \tau - 1) = S_{NN}(\tau, \tau - 1) \, S_{NN}^{-1}(\tau - 1, \tau - 1) \, S_{NM}(\tau - 1, \tau - 1) \qquad \text{equation I.29}$$

and

$$S^*_{MM}(\tau,\tau-1) = S_{MM}(\tau,\tau-1) + S_{MN}(\tau,\tau) \, S^{-1}_{NN}(\tau, \tau) \, [S^*_{NM}(\tau,\tau-1) - S_{NM}(\tau,\tau-1)] \qquad \text{equation I.30}$$

If one desires to neglect the lag one seasonal correlations by setting parameter matrix H to null, this may be accomplished by adjusting $S_{MM}(\tau, \tau - 1)$ for the season in question, τ, according to equation I.31.

$$S^*_{MM}(\tau, \tau - 1) = S_{MN}(\tau, \tau) \, S_{NN}^{-1}(\tau, \tau) \, S_{NM}(\tau, \tau - 1) \qquad \text{equation I.31}$$

The theoretic moments preserved upon generation can be obtained by iteratively applying equations I.23, I.24, I.27, and I.29 for season one and I.23, I.24, and I.27 for all other seasons.

It should be noted that since the correlation with only 1 previous month is being preserved, that the generated seasonal data will not add exactly to give the annual data. This difference is usually corrected for by an adjustment being made to each of the generated seasonal values. This adjustment also corrects for any lack of correspondence between the sum of the seasonal generated values and the annual generated values caused by data transformations. The reader should note that the adjustment process has the potential of causing serious problems.

To examine the advantage of the selected approach over the approach of Valencia and Schaake (1973), as represented by all but the last product of equation I.3, and over the approach of Mejia and Rousselle (1976) as represented by equation I.14, the number of parameters will be calculated for various examples. The table below shows the results of this examination. For the Mejia and Rousselle approach, only one season is included in matrix Z.

Seasonal Disaggregaton Number of Parameters

Number of stations	Number of seasons	Valencia and Schaake (1973) disaggregation	Mejia and Rousselle (1976) disaggregation	Last package disaggregation
1	12	156	168	36
1	24	600	624	72
1	52	2,756	2,808	156
1	121	14,762	14,883	363
2	12	624	672	144
2	24	2,400	2,496	288
3	12	1,404	1,512	324
p	s	$p^2s + p^2s^2$	$2p^2s + p^2s^2$	$3p^2s$

E. Transformations

A goal of this package of computer programs was to allow considerable flexibility in the choice of transformations. Although only two types of transformations are available, due to their form, they provide a wide range of effects. By varying the parameters, the user has good control over the transformation.

The first type of transformation is a power transform,

$$t = (o + a)^b \qquad \text{equation J.1}$$

and the second is a logarithmic transform,

$$t = \ln (o + a) \qquad \text{equation J.2}$$

where o is the original data, t is the transformed data, and a and b are user specified parameters. The manner in which the programs are written allows for future addition of other types of transforms.

F. Random Normal Numbers

In the generation of synthetic data, normally distributed random numbers must be created. These numbers are created by resorting to an algorithm which produces numbers which behave as if random although they are in fact deterministic. These numbers, often termed "pseudorandom" numbers, are more than adequate for our purpose. The microcomputer function RANF written by R.D. McKisson is used to produce random numbers which are uniformly distributed over the range zero to one.

G. Estimation of Moments

Several approaches may be used to estimate moments. The programs offer two alternatives for the calculation of moments: circular definition and open series definition. Open series estimates for lag zero, lag one, and lag two moments are given by

$$M_o = \sum_{i=1}^{N} X_i X_i^T / N \qquad \text{equation L.1}$$

$$M_1 = \sum_{i=2}^{N} X_i X_{i-1}^T / (N - 1) \qquad \text{equation L.2}$$

and

$$M_2 = \sum_{i=3}^{N} X_i X_{i-2}^T / (N - 2) \qquad \text{equation L.3}$$

The circular series estimates are given by

$$M_o = \sum_{i=1}^{N} X_i X_i^T / N \qquad \text{equation L.4}$$

$$M_1 = (\sum_{i=2}^{N} X_i X_{i-1}^T + X_1 X_N^T) / N \qquad \text{equation L.5}$$

and

$$M_2 = (\sum_{i=3}^{N} X_i X_{i-2}^T + X_1 X_{N-1}^T + X_2 x_N^T) / N \qquad \text{equation L.6}$$

Where X is a column vector of observations for all of the variables, and N is the number of observations. The X values are already adjusted to have zero means. Equivalent equations are used in the programs.

Since the moment matrix estimates are biased, an attempt is made to correct for this bias before using these estimates. In this case, the denominator in each of the equations is reduced by one. This would be a fruitful area for future improvement as better ways for "unbiasing" estimates are possible.

Wallis, Matalas, and Slack (1974) are helpful in this area.

H. Serial Correlation Coefficients

Serial correlation coefficients are calculated in two slightly different manners. In program CORREL, serial correlation coefficients are calculated using one approach while program TERMPL uses a second approach. The differences in values are usually so minor that the user will not notice. The first approach is basically equation 2.1 of Yevjevich (1972) while the second approach is given by equation 2.7 of Yevjevich (1972).

These two equations are given here as equations M.1 and M.2 in terms of covariances and variances.

$$r_k = \frac{\operatorname{cov}(X_i, X_{i+k})}{\operatorname{var}(X)} \qquad \text{equation M.1}$$

$$r_k = \frac{\operatorname{cov}(X_i, X_{i+k})}{(\operatorname{var}(X_i)\operatorname{var}(X_{i+k}))^{1/2}} \qquad \text{equation M.2}$$

I. Parsimony of Parameters

Program BEGIN is designed to check to ensure that enough data exist, so that the required parameters may be estimated. In general, one likes to keep the ratio of data elements to the number of parameters as large as possible. Salas (1978) has very effectively commented on this issue. He suggests that a ratio of 20 to 1 is desirable. For the purpose of this package, this suggestion has been expanded as follows:

Ratio of data values to parameters	Comments
$R < 1$	Impossible
$1 \le R < 3$	Foolish
$3 \le R < 5$	Poor
$5 \le R < 10$	Fair
$10 \le R < 20$	Good
$20 \le R$	Very Good

The comments given may be a bit lenient when considering key station generation. For disaggregation, this is felt to give a fairly good guide.

J. Adjustment of Seasonal Data

In order to ensure that the generated seasonal data add exactly to give the annual data, two alternatives are possible. The first approach is to take the seasonal values as they have been generated and to recalculate the annual values. This approach is not generally recommended for several reasons. Any errors due to model limitations are "concentrated" in the annual values. For example, if correlations are not truly linear this would contribute some error. It is felt that it is better to spread the errors of this type to ensure that the impact is not of any consequence. Another argument would be that if the annual data had been generated via a sophisticated model, particularly a long-term memory model, the important aspects of the annual model may be adversely affected. This has not been demonstrated by actual generated data. It is this author's opinion that, since the disaggregation models lack the mechanism to affect long-term memory except in a very indirect manner, this concern is not justified. A final reason for not generally using this approach is that, where the seasonal disaggregation of this package is used, the lag zero annual correlations are very poorly preserved. This is as expected since the seasonal disaggregation does not preserve all of the cross correlations among the seasonal data. For Valencia and Schaake (1973) or Mejia and Rousselle (1976), this argument is not applicable.

The second approach is to adjust the generated seasonal data to correspond to the generated annual data. Several methods of adjustment are possible.

The present default form of adjustment is

$$s^*_\tau = s_\tau + \frac{(q - \sum s_i)\ s_\tau - \mu_\tau}{\sum |s_i - \mu_i|} \qquad \text{equation 0.1}$$

where s_τ is the seasonal data for the τ - th season, q is the generated annual value, the summations are taken over all seasons, and the superscript * indicates the adjusted value. This approach is the preferable method available at this time.

An alternate approach is to proportionally adjust the seasonal data by

$$s^*_\tau = s_\tau\ (q\ /\ \sum s_i) \qquad \text{equation 0.2}$$

A future planned form of adjustment is to adjust the seasonal data proportionally to the seasonal standard deviations, denoted by

$$s^*_\tau = s_\tau + (q - \sum s_i)\ \sigma_\tau\ /\ \sum \sigma_i \qquad \text{equation 0.3}$$

REFERENCES

Box, G.E.P., and M. E. Muller, A Note on the Generation of Normal Deviates, Ann. Mathematical Statistics, No. 28, 1958, pp. 610-611

Matalas, N. C., Mathematical Assessment of Synthetic Hydrology, Water Resources Research, Vol. 3, No. 4, 1967, pp. 937-947

Mejia, J. M. and J. Rousselle, Disaggregation Models in Hydrology Revisited, Water Resources Research, Vol. 12, No. 2, April 1976, pp. 185-186

Salas, J. D., Applied Stochastic Modeling of Hydrologic Series, Chapter V of Lecture Notes for the Computer Workshop in Statistical Hydrology Held at Colorado State University, July 1978

Tao, P. C., and J. W. Delleur, Multistation, Multiyear Synthesis of Hydrologic Time Series by Disaggregation, Water Resources Research, Vol. 12, No. 6, December 1976

Valencia, D., and J. C. Schaake, Jr., Disaggregation Processes in Stochastic Hydrology, Water Resources Research, Vol. 9, No. 3, 1973, pp. 580-585

Wallis, J. R., N. C. Matalas, and J. R. Slack, Just a Moment!, Water Resources Research, Vol. 10, No. 2, 1974, pp. 211-219

Yevjevich, V. M., Stochastic Processes in Hydrology, Water Resources Publications, Fort Collins, Colorado, 1972, 276 pp.

Young, G. K., and W. C. Pisano, Operational Hydrology Using Residuals, Journal of the Hydraulics Division, ASCE, Vol. 94, No. 4, 1968, pp. 909-923

VI. QUICK GUIDES

This chapter is composed of one-page guides to help users apply these programs. For each program, the guide indicates the main input requirements, a list of the various options which are available, and a listing of the files normally associated with running the program.

Quick Guide to

Program BEGIN

Program Function: To create a parameter file or read and list an existing parameter file.

Required Terminal Input:

1. Generic problem name
2. Input-output control string

Files Required:

BEGIN.FOR - Source version (used only for making changes)

BEGIN.EXE - Executable version of BEGIN.FOR
*.DAT - Original data file

Files Created:

*.OUT - Result file
*.PAR - Parameter storage file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name.

Quick Guide to

Program CORREL

Program Function: To estimate covariances and correlation coefficients for original historical data, transformed data, or generated data.

Required Terminal Input:

1. Generic problem name
2. Data file to be read from
3. Input-output control string
4. Type of correlations: Annual, seasonal or cross between annual and seasonal

Input-output Controls:

IOC(1)---Print of annual and seasonal generated data as read
IOC(2)---Printout of original or transformed annual data as read
IOC(3)---Printout of original or transformed seasonal data as read
IOC(4)---Final unbiased covariance matrices printout
IOC(5)---Printout of annual data matrix for original or transformed data
IOC(6)---Printout of seasonal data matrix for original or transformed data
IOC(7)---Noncircular definition of moments
IOC(8)---Prints biased covariance matrices
IOC(9)---Printout of data mean values

Files Required:

CORREL.FOR - Source version (used only for making changes)
CORREL.EXE - Executable version of CORREL.FOR
*.DAT - Original data file
*.TRN - Transformed data file
*.GEN - Generated data file

Files Created:

*.OUT - Result file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name.

Quick Guide to

Program KEYPAR

Program Function: To estimate the parameters needed to generate key station data on an annual basis.

Required Terminal Input:

1. Generic problem name
2. Input-output control string
3. Desired key station generation group
4. Indexes of key stations to be used
5. Number of lagged correlations to be preserved (1 or 2)
6. Method to be used (choice of four):
   a. Full Matalas approach with lagged cross correlations preserved
   b. Matalas approach with lagged cross correlations ignored
   c. Decoupled approach with lagged cross correlations ignored
   d. ARMA (1, 1) approach (future addition)

Input Output Controls:

IOC(1)---Input printout (from transformed data file)
IOC(2)---Intermediate moment matrices printout
IOC(3)---Transformed data matrix printout
IOC(4)---Covariance matrices printout
IOC(5)---Suppress answer matrices print
IOC(6)---Forces print of matrix CC^T
IOC(7)---Noncircular definition of moments
IOC(8)---Suppresses output to the parameter file
IOC(9)---Print generated moments

Files Required:

KEYPAR.FOR - Source version (used only for making changes)

KEYPAR.EXE - Executable version of KEYPAR.FOR
*.TRN - Transformed data file
*.PAR - Parameter storage file

Files Created:

*.OUT - Result file

Run Initiated:

Type in "MENU"

Note "*" indicates the generic problem name

Quick Guide to

Program DISPAR

Program Function: To estimate the parameters needed to disaggregate key station data into substation data on an annual basis.

Required Terminal Input:

1. Generic problem name
2. Input-output control string
3. Desired key to substation disaggregation group
4. Indexes of key stations to be used
5. Indexes of substations to be used

Input-output Controls:

IOC(1)---Input (transformed) data printout
IOC(2)---Intermediate moment matrices printout
IOC(3)---Transformed data matrix printout
IOC(4)---Covariance matrices printout
IOC(5)---Suppress answer matrices print
IOC(6)---Forces print of matrix DDT
IOC(7)---Noncircular definition of moments
IOC(8)---Suppresses output to the parameter file
IOC(9)---Print generated moments
IOC(1O)---Use $S_{KN}(1)$ and $S_{NN}(1)$ estimated from actual data
IOC(11)---Use $S_{NN}(1)$ estimated from actual data
IOC(12)---Ignore lagged substation correlations

Files Required:

DISPAR.FOR - Source version (used only when changes are to be made)
DISPAR.EXE - Executable version of DISPAR.FOR
*.TRN - Transformed data file
*.PAR - Parameter storage file

Files Created:

*.OUT - Results file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name

Quick Guide to

Program SEAPAR

Program Function: To estimate the parameters needed to disaggregate annual data into seasonal data.

Required Terminal Input:

1. Generic problem name
2. Input-output control string
3. Desired annual to seasonal disaggregation group
4. Indexes of stations to be used

Input-output Controls:

IOC(1)---Input file heading material printout
IOC(2)---Annual (transformed) input data printout
IOC(3)---Seasonal (transformed) input data printout
IOC(4)---Annual(transformed) data matrix printout
IOC(5)---Seasonal (transformed) data matrix printout
IOC(6)---Intermediate moment matrices printout
IOC(7)---Noncircular definition of moments
IOC(8)---Suppresses output to the parameter file
IOC(9)---Stop program after reading of data
IOC(10)---Stop program after first season's estimation
IOC(11)---Covariance matrices printout
IOC(12)---Use $S^*_{NM}(1,0)$ and $S_{MM}(1,0)$ estimated from actual data
IOC(13)---Use $S_{MM}(1,0)$ estimated from actual data
IOC(14)---Print generated moments for first season

Files Required:

SEAPAR.FOR - Source version (used only when changes are to be made)
SEAPAR.EXE - Executable version of SEAPAR.FOR
*.TRN - Transformed data file
*.PAR - Parameter storage file

Files Created:

*.OUT - Results file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name

Quick Guide to

Program TRNPAR

Program Function: Transform the original historical data into normally distributed data and provide the parameters of the inverse of the transform.

Required Terminal Input:

1. Generic problem name
2. Input-output control string

Input-output Controls:

IOC(1)---Printout of transformed data as calculated
IOC(2)---Annual input data printout
IOC(3)---Seasonal input data printout
IOC(4)---Printout of transformed data as written
IOC(5)---Unused
IOC(6)---Printout of original means (Not presently available)
IOC(7)---Printout of transformed means (Not presently available)

Files Required:

TRNPAR.FOR - Source version (used only when changes are to be made)
TRNPAR.EXE - Executable version of TRNPAR.FOR

*.DAT - Original data file
*.PAR - Parameter storage file

Files Created:

*.OUT - Results file
*.TRN - Transformed data file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name

Quick Guide to

Program FLWGEN

Program Function: To generate synthetic data.

Required Terminal Input:

1. Generic problem name
2. Input-output control string
3. Desired number of years to be generated
4. Approach for starting generation (lead-in years, random, or fixed start)
5. Random number seed

Input-output Controls:

IOC(1)---General Information on data and parameter files
IOC(2)---Idata array printout
IOC(3)---Parms array printout
IOC(4)---Printing in MATMOV, MATFIL, MATMVS
IOC(5)---Printing in MATRND
IOC(6)---Printing of matrices in BASGEN
IOC(7)---Printing in ONEYER (including generated data)
IOC(8)---Only generate key stations
IOC(9)---Only generate annual data
IOC(10)---Do not output generated data
IOC(11)---Allow more than 100 years
IOC(12)---Allow limits on random numbers
IOC(13)---Sum seasonal values to give annual
IOC(14)---Suppress all adjustments
IOC(15)---Check on random numbers
IOC(16)---Proportional adjustment of seasonal values

Files Required:

FLWGEN.FOR - Source version (used only when changes are to be made)
FLWGEN.EXE - Executable version of FLWGEN.FOR
*.PAR - Parameter storage file

Files Created:

*.OUT - Results file
*.GEN - Generated data file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name

Quick Guide to

Program TERMPL

Program Function: To examine data and plot graphs on machine or on printer.

Required Terminal Input:

1. Generic problem name
2. Input-output control string
3. Indexes of stations to be used
4. Types of graphs desired

Input Output Controls:

IOC(1)---Hazen plotting position (Weibull normally)
IOC(2)---Annual input data printout (Not available for generated data)
IOC(3)---Seasonal input data printout (Not available for generated data)
IOC(4)---Unused
IOC(5)---Annual data matrix printout
IOC(6)---Seasonal data matrix printout

Files Required:

TERMPL.FOR - Source version (used only when changes are to be made)
TERMPL.EXE - Executable version of TERMPL.FOR
*.DAT - Original data file
*.GEN - Generated data file
*.TRN - Transformed data file

Files Created:

*.OUT - Results file

Run Initiated:

Type in "MENU"

Note: "*" indicates the generic problem name

APPENDIX A

TRANSFORMED DATA FILE

This data file is created by the transformation program TRNPAR. It is based on the input data file and follows a similar order. The heading and introductory material is identical with the only exception being that the creation date line will include (automatically) a notation that these are the transformed data and an updated creation date and that the K, X and Y cards showing the generation groupings are not included. For each line in the input data file, there is a corresponding line in the transformed data file.

APPENDIX B

PARAMETER FILE

This file is created by running the program BEGIN. Parameters as estimated by the various estimation programs are written into this file are used as input into the generation program. The typical user does not require any knowledge of the structure of this file. For the benefit of programmers, this appendix will introduce the structure of this important file.

This file is organized into lines as follows:

First five lines:	Contains lines T, S, C, N, and M in virtually the same form as the input data file (see chapter IV)
Next set of lines:	One line for each station containing H line information
Next set of lines:	Number of stations in each generation group
Next set of lines:	Indexes of stations in each generation group
Next set of lines:	Key station generation parameters
Next set of lines:	Key to substation disaggregation parameters
Next set of lines:	Annual to seasonal disaggregation parameters
Next set of lines:	Distribution transformation parameters

APPENDIX C

GENERATED DATA FILE

This appendix describes the form of the generated data file. This file is written in a form with the heading material very similar to that of the input data file.

First five lines:	Contains lines T, S, C, N, and M in virtually the same form as the input data file
Next set of lines:	One line for each station containing H line information
Next three lines:	Contains the number of years to be generated, number of lead-in years, and the random number seed
Next set of lines:	Generated data. Data written by years (Annual totals by seasonal totals)

APPENDIX D

ORGANIZATION

This appendix documents the overall organization of this set of programs with primary attention called to the interaction between the various computer programs and computer files. The organization of the individual programs is covered in the next appendix.

Figure D-1 illustrates the overall file handling and the manner in which control is maintained.

The definitions for the various files involved in running this set of programs are illustrated by figure D-2. Because of the manner in which files are named utilizing a generic name plus a suffix letter, several users may use this package at one time. Each user would have his own unique generic problem name. This remains true even if all of the users have their files stored under the same directory.

Before using this package, users should convince themselves that the file names as shown in figures D-1, and D-2 do not conflict with any files presently stored in their directory.

Figure D-1. - OVERALL FILE HANDLING

Execution Command "MENU"

T
*
↓

<u>Required Files:</u>

*	Input file (required for all programs)
*.TRN	Transformed Data File (required for programs KEYPAR SEAPAR and DISPAR)
*.PAR	Parameter File (required for programs KEYPAR SEAPAR DISPAR and FLWGEN)

Executable Version of Program

T
*
↓

<u>Files Produced or Modified:</u>

*.OUT	Result File (produced by all programs)
*.PAR	Parameter File (produced or modified by programs BEGIN KEYPAR SEAPAR DISPAR and TRNPAR)
*.TRN	Transformed Data File (produced by program TRNPAR)
*.GEN	Generated Data File (produced by program FLWGEN)

Figure D-2. - OVERALL FILE NAMING

File Name	Purpose/Description
Generic Files	
*.DAT	Original Input Data File
*.TRN	Transformed Data File
*.GEN	Generated Date File
*.PAR	Parameter File
*.OUT	Result File

Program Files (Executable version name in parenthesis)

MENU.FOR (MENU.EXE)	Program for Overall Program Operation
BEGIN.FOR (BEGIN.EXE)	Parameter File Creation Program
CORREL.FOR (CORREL.EXE)	Correlation Program
KEYPAR.FOR (KEYPAR.EXE)	Key Station Parameter Estimation
DISPAR.FOR (DISPAR.EXE)	Key/Sub Station Parameter Estimation
SEAPAR.FOR (SEAPAR.EXE)	Seasonal Parameter Estimation
TRNPAR.FOR (TRNPAR.EXE)	Transformation Parameter Estimation
FLWGEN.FOR (FLWGEN.EXE)	Generation Program
TERMPL.FOR (TERMPL.EXE)	Terminal Plotting Program

APPENDIX E

Individual Programs Details

This appendix documents several areas which are primarily of interest to programmers, especially those involved in making modifications to these programs or those involved in using these programs at another facility. Topics include individual program organization, library functions, problem size limitations, and unusual programming techniques.

The appendix is organized on a program-by-program basis. For each program, the organization of subroutines is illustrated by a figure. Some subroutines which perform concise functions are used by several of the programs. These routines are classified as utility subroutines. The utility subroutines are not shown in the figures illustrating the organization of the various functions. Their use by the programs and their functions are shown by figures E-17 and E-18. A set of plotting routines for interactive plotting (program TERMPL) was developed and is shown in figure E-19.

Unusual programming techniques common to many of the programs which may affect users at other facilities include: passing of dimensions via calling lists, equivalencing of variables (especially where one variable is a real and one is an integer), use of random access files, use of labeled common, reliance on the storage pattern of arrays in memory, and changing of dimensions between routines (such as using part of a three-dimensional array as a two-dimensional array by passing an appropriate starting address).

Error messages will not be discussed in detail. The user, therefore, is cautioned not to rely upon the programs to detect errors in input data. However, a great deal of error checking is done to protect the user from blunders. Size limitations are checked by each program in addition to checking of the form of input data. Some checking is also performed to detect abnormal results during calculations.

Depending on the personal computer being used, some modifications may be required. Problem size limitations are shown in figure E-20.

Program BEGIN

This program is set to handle a data file with up to 30 stations. The parameter file created by this program may have up to 300 records. This number should be adequate for all applications. The file is checked by the program. Up to 1O key station generation groups may be specified. The maximum number of stations per group is 1O. Up to 1O key-to-substation generation groups may be specified with up to 3 key stations and 1O substations per group. Up to 1O annual to seasonal generation groups may be specified with up to 1O stations per group. The organization and function of the subroutines are illustrated by figures E-1 and E-2.

Program CORREL

This program is set to handle up to 30 stations. The number of years of data allowable is equal to 1,500 divided by the number of stations. For general data, an unlimited number of years may be used. The number of seasons for generator data is limited to 360 divided by the number of stations. For other data, the number of seasons is limited to 12,240 divided by the product of the number of seasons and the number of stations. Also, the number of seasons is limited to 4,800 divided by the number of stations squared, regardless of the source of data. It should be noted that all stations are used from the data file. The organization and function of the subroutines are illustrated by figures E-3 and E-4.

Program KEYPAR

This program is set to handle up to 30 stations in the input data file. Only a maximum of 10 stations may be included in any one key station generation group. Up to 100 years of data are allowable. The organization and function of the subroutines are illustrated by figures E-5 and E-6.

Program DISPAR

Up to 30 stations may be included in the input data file. Only a maximum of 10 substations and 3 key stations may be included in any one group. Up to 100 years of data are allowable. The organization and function of the subroutines are illustrated by figures E-7 and E-8.

Program SEAPAR

This program is limited to 30 stations in the input data file. A maximum of 10 stations may be included in any one seasonal generation group. Up to 50 years of data may be used (or up to 500 divided by the number of stations in the generation group if larger). Up to 12 seasons may be used (or 6,000 divided by the product of stations and years actually used if larger). The organization and function of the subroutines are illustrated by figures E-9 and E-10.

Program TRNPAR

This program has a limit of 30 stations in the input data file. Up to 150 years of data may be used. Up to 12 seasons may be used. The product of seasons and years cannot exceed 1,800 and the number of seasons cannot exceed 390 divided by the number of years minus 1. This may be the most severe size restriction involved in this package. The organization and function of the subroutines are illustrated by figures E-11 and E-12.

Program FLWGEN

This program has size limitations consistent with those specified for the other programs. The organization and function of the subroutines are illustrated by figures E-13 and E-14.

Program TERMPL

This program allows 30 stations in the input data file. The number of seasons cannot exceed 8,100 divided by the product of number of stations and number of years. Nor can it exceed 1,200 divided by the number of seasons. The number of years cannot exceed the lesser of 1,500 divided by the number of stations or 100. Figures E-15 and E-16 show the organization and function of the program's major subroutines. Figure E-19 lists plotting subroutines used in the program.

Main routine (BEGIN)

Subroutine HEAD

Subroutine MHEAD

Subroutine CONTROL

Subroutine READ1

Subroutine READER

Subroutine COMPUT

Subroutine OUT

Subroutine WLIST

Subroutine MEND

Note: Minor matrix operation subroutines not shown.

Figure E-1. - Organization of program BEGIN

Subroutine	Function
HEAD	Writes out a heading to output file
MHEAD	Writes heading on screen
CONTROL	Sets up controls
READ1	Reads introductory information
READER	Reads in data
COMPUT	Computes data storage locations
OUT	Writes out new parameter file
WLIST	Lists an existing parameter file
MEND	Completes timing checks and closes files

Note: Minor matrix operation subroutines not listed.

Figure E-2. - Program BEGIN Subroutines

Main routine (CORREL)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CONTROL
- Subroutine READING
    - Subroutine READ1
        - Subroutine READ2
        - Subroutine READ3
- Subroutine ANMOM
    - Subroutine PRODSUM
    - Subroutine PRODOP
    - Subroutine PRODCIR
    - Subroutine STAND
    - Subroutine MATCOR
- Subroutine SEAMOM
    - Subroutine PRDSEAS
    - Subroutine OPSEAS
    - Subroutine CIRSEAS
    - Subroutine STAND
    - Subroutine MATCOR
    - Subroutine SMATCOR
- Subroutine CRSMOM
    - Subroutine ANSEAS
        - Subroutine MATDOT
    - Subroutine ANSEOP
    - Subroutine ANSECR
- Subroutine MEND

Note: Minor matrix operation subroutine not shown.

Figure E-3. - Organization of program CORREL

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading on screen
CONTROL	Sets up controls
READING	Controls reading of data
READ1	Reads data
READ2	Read annual data
READ3	Read seasonal data
ANMOM	Calculates annual moments
PRODSUM	Calculates annual product sums
PRODOP	Adjusts annual product sums for open series approach
PRODCIR	Adjusts annual product sums for circular series approach
STAND	Calculates standard deviations
MATCOR	Calculates annual correlation coefficients
SEAMOM	Calculates seasonal moments
PRDSEAS	Begins computations for covariance of seasonal data
OPSEAS	Adjusts seasonal products sums for open series approach
CIRSEAS	Adjusts seasonal product sums for circular series approach
MATCORS	Calculates lag 1 seasonal correlation coefficients
CRSMOM	Calculates combination annual seasonal moments
ANSEAS	Calculates combination product sums
MATDOT	Calculates dot product
ANSEOP	Adjusts combination product sums for open series approach
ANSECR	Adjusts combination product sums for circular series
MEND	Completes timing checks and closes files

Note: Minor matrix operation subroutines not listed.

Figure E-4. - Program CORREL subroutines

Main routine (KEYPAR)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CNTROL
- Subroutine READER
  - Subroutine COR
  - Subroutine READ2
- Calculate moments (three calls each to MATVAR and UNBIAS)
- Estimate parameters (one call to either FULL1, FULL2, NOLAG1, NOLAG2, DCPL1, or DCPL2 dependent on the method selected)
- Subroutine GENMOM
- Subroutine PAROUT
- Subroutine DSOLVE

Note: Minor matrix operation subroutines not shown.

Figure E-5. - Organization of program KEYPAR

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading on screen
CNTROL	Sets up controls
READER	Controls reading of data
READ2	Reads annual data
COR	Sets up a table of correspondence
FULL1	Solves for parameters, full Matalas approach lag one
FULL2	Solves for parameters, full Matalas approach lag two
NOLAG1	Solves for parameters, limited Matalas approach lag one
NOLAG2	Solves for parameters, limited Matalas approach lag two
DCPL1	Solves for parameters, decoupled approach, lag one
DCPL2	Solves for parameters, decoupled approach, lag two
DSOLVE	Solves a matrix SDT for matrix D
PAROUT	Outputs parameters
GENMOM	Calculates generated moments

Note: Minor matrix operations subroutines not listed.

Figure E-6. - Program KEYPAR subroutines

Main routine (DISPAR)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CNTROL
- Subroutine READER
  - Subroutine READ2
  - Subroutine COR
- Calculate moments (five calls each to MATVAR and UNBIAS)
- Subroutine MODIFY
- Subroutine CSOLVE
- Subroutine ESOLVE
- Subroutine DDTCLC
- Subroutine DSOLVE
- Subroutine PAROUT
- Subroutine GENMOM

Note: Minor matrix operation subroutines not shown.

Figure E-7. - Organization of program DISPAR

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading on screen
CNTROL	Sets up controls
READER	Controls reading of data
READ2	Reads annual data
COR	Sets up a table of correspondence
CSOLVE	Solves for matrix C
ESOLVE	Solves for matrix E
DDTCLC	Solves for matrix DDT
DSOLVE	Solves for matrix D
PAROUT	Outputs parameters
MODIFY	Calculates SKN matrix
GENMOM	Calculates generated moments

Note: Minor matrix operation subroutines not listed.

Figure E-8. - Program DISPAR subroutines

Main routine (SEAPAR)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CNTROL
- Subroutine READER
  - Subroutine COR
  - Subroutine READ1
    - Subroutine READ2
    - Subroutine READ3
- Subroutine ESTMAT
  - Subroutine CALC
    - Subroutine DSOLVE
  - Subroutine GENMOM
  - Subroutine PAROUT

Note: Minor matrix operation subroutines not shown.

Figure E-9. - Organization of program SEAPAR

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading on screen
CNTROL	Sets up controls
READER	Controls reading of data
COR	Sets up a table of correspondence
READ1	Reads data
READ2	Reads annual data
READ3	Reads seasonal data
ESTMAT	Handles parameter estimation
CALC	Calculates parameters
DSOLVE	Solves matrix DDT for D
PAROUT	Outputs parameters
GENMOM	Calculates generated moments

Note: Minor matrix operation subroutines not listed.

Figure E-10. - Program SEAPAR subroutines

Main routine (TRNPAR)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CONTROL
- Subroutine READING
  - Subroutine READ1
    - Subroutine READ2
      - Subroutine MEAN
      - Subroutine NORM
        - Subroutine TRAN
      - Subroutine TRDOUT
    - Subroutine READ3
      - Subroutine MEAN
      - Subroutine NORM
        - Subroutine TRAN
      - Subroutine TRDOUT
- Subroutine PAROLD
- Subroutine PARNEW
- Subroutine PAROUT

Note: Minor matrix operation subroutines not shown.

Figure E-11. - Organization of program TRNPAR

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading to screen
CONTROL	Sets up controls
READING	Controls reading of data
READ1	Reads data
READ2	Reads annual data
READ3	Reads seasonal data
PAROLD	Reads old parameters
PARNEW	Inputs new parameters
NORM	Subroutine to manage the transformation
TRAN	Transforms data
MEAN	Calculates mean values
TRDOUT	Writes out transformed data
PAROUT	Writes new parameters

Note: Minor matrix operation subroutines not listed.

Figure E-12. - Program TRNPAR subroutines

Main routine (FLWGEN)

- Subroutine MHEAD
- Subroutine HEAD
- Subroutine CONTROL
- Subroutine READER
- Subroutine ONEYER
  - Subroutine BASGEN
  - Subroutine TRNDAT
  - Subroutine ADJUST
  - Subroutine GOUT

Note: Minor matrix operation subroutines not shown.

Figure E-13. - Organization of program FLWGEN

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading to screen
CONTROL	Sets up controls
READER	Reads in parameters
ONEYER	Generates data 1 year at a time
BASGEN	Calculates basic generation equation
TRNDAT	Transforms generated data
ADJUST	Adjusts generated data
GOUT	Outputs generated data

Note: Minor matrix operation subroutines not listed.

Figure E-14. - Program FLWGEN subroutines

Main routine (TERMPL)

- Subroutine HEAD
- Subroutine MHEAD
- Subroutine CONTROL
- Subroutine READING
  - Subroutine READ1
    - Subroutine READ2
    - Subroutine READ3
  - Subroutine READ4
- Subroutine PASK
  - Subroutine POUT
    - Subroutine SUPER
      - Subroutine AUTO
        - Subroutine PAUTO
          - Subroutine MINMAX
          - Subroutine SIZE
        - Subroutine AUTCAL
        - Subroutine TRANS
          - Subroutine SET
      - Subroutine CORE
        - Subroutine PCOREL
          - Subroutine MINMAX
          - Subroutine SIZE
        - Subroutine TRANS
          - Subroutine SET
      - Subroutine DIST
        - Subroutine PCDF
          - Subroutine NORMP
          - Subroutine ORDER
          - Subroutine PPOSIT
          - Subroutine SDTR
          - Subroutine SIZE
        - Subroutine PPDF
          - Subroutine DERIV
          - Subroutine MINMAX
          - Subroutine ORDER
          - Subroutine PPOSIT
          - Subroutine SIZE
          - Subroutine SMOOTH
        - Subroutine RESEE
        - Subroutine TRANS
          - Subroutine SET
      - Subroutine SERIAL
        - Subroutine PSER
          - Subroutine MINMAX
          - Subroutine SIZE
        - Subroutine TRANS
          - Subroutine SET
      - Subroutine STAT
        - Subroutine PSTAT
          - Subroutine MINMAX
          - Subroutine SIZE
        - Subroutine STACAL
          - Subroutine SET
          - Subroutine TRANS
            - Subroutine SET

Note: Minor matrix operation and plotting subroutines not shown.

Figure E-15. - Organization of program TERMPL

Subroutine	Function
HEAD	Writes heading to output file
MHEAD	Writes heading to screen
CONTROL	Sets up controls
READING	Controls reading of data
READ1	Reads data
READ2	Reads annual data
READ3	Reads seasonal data
READ4	Reads generated data
PASK	Asks plot control parameters
POUT	Controls plot supervision routine
SUPER	Supervises the plotting
AUTO	Handles autocorrelation plots
SERIAL	Handles serial plots
TRANS	Transforms data
PSER	Does serial plots
MINMAX	Finds minimum and maximum values
SIZE	Sizes a plot
CORE	Handles correlation plots
PCOREL	Does correlation plots
STAT	Handles statistics plots
STACAL	Calculates statistics
SET	Moves data from one array to another
PSTAT	Does statistical plots
DIST	Handles distribution plots
RESEE	Rearranges seasonal data
PCDF	Plots cumulative distribution function
ORDER	Arranges data in ascending order
PPOSIT	Calculates plotting positions
SDTR	Converts probabilities to standard deviations
PPDF	Plots probability density function
SMOOTH	Smooths data
DERIV	Calculates derivative
AUTO	Handles autocorrelation plots
AUTCAL	Calculates autocorrelation coefficient
PAUTO	Plots autocorrelation coefficients
NORMP	Draws normal probability paper
DATER	Gets current date

Note: Minor matrix operation and plotting subroutines not listed. Plotting subroutines written specifically for TERMPL are listed in figure E-20.

Figure E-16. - Program TERMPL subroutines

Routine	Function
CLS	Clears screen
CONCAT	Concatinates character strings
DCTIME	Finds elapsed time since last call
DATE1	Gets current date
EXFILE	Checks to see if file exists
ETIME	Initiates time counter
FCHECK	Checks to see if file name is misspelled
FILEO	Lists existing files and calls routine to open files
HIT	Registers carriage returns
LENGTH	Determines length of a character string
IMPROP	Informs user of improper response
CHKIN	Informs a user if an input file does not exist
CHKOUT	Informs a user if an output file already exist
MATADD	Performs matrix addition
MATAVE	Averages the columns of a matrix
MATCON	Multiplies each element of a matrix by a constant
MATDOT	Takes the dot product of a matrix with itself
MATEQ	Sets one matrix equal to another
MATEQ2	Sets a portion of one matrix equal to a portion of another
MATFIL	Fills a matrix with information from a second matrix according to a selected pattern
MATINV	Performs matrix inversion
MATMOM	Calculates matrix moments about zero
MATMOV	Similar to MATFIL but pattern reversed
MATMSI	Inputs matrix information from a random access file
MATMSW	Outputs matrix information from a random access file
MATMUL	Performs matrix multiplication
MATMVS	Similar to MATMOV but for seasonal data, not annual
MATPRD	Performs matrix multiplication of one matrix times the transpose of another matrix
MAT2PRD	Performs matrix multiplication of one matrix times the transform of another matrix with the results added to a third matrix
MATPRN	Performs formatted print of a matrix
MATPRI	Performs formatted print of an integer matrix
MATPRND	Prints the diagonal of a matrix
MATRND	Fills a matrix with normal (O, 1) random numbers
MATSUB	Performs matrix subtractions
MATTRN	Transposes a matrix
MATVAR	Calculates covariance matrix
MATZER	Sets a matrix's elements to zero
ICHECK	Checks for a yes or no (Y/N) response
MATZER1	Sets an integer matrix's elements to zero
MEND	Ending routine to close files and write completion message
OPENF	Calls other subroutines to open files
TIMER	Gets current time
TIMEF	Converts wall clock time to a real number
MTREDF	Reads a matrix
UDATE	Gets current date and time from the system using microsoft compiler
UNBIAS	Corrects a covariance matrix for lost degrees of freedom

Figure E-17. - Utility subroutines and functions

Program

Utility Routine	BEGIN	CORREL	KEYPAR	DISPAR	SEAPAR	TRNPAR	FLWGEN	TERMPL
MATADD		X	X	X	X		X	
MATAVE			X	X	X			
MATCON		X	X	X	X			
MATDOT		X						
MATEQ		X	X	X	X		X	
MATFIL							X	
MATINV			X	X	X			
MATMOM			X	X	X			
MATMOV							X	
MATMUL			X	X	X		X	
MATMVS							X	
MATPRD		X	X	X	X			
MAT2PRD		X						
MATPRN	X	X	X	X	X	X	X	X
MATPRI	X						X	
MATRND							X	
MATSUB		X	X	X	X			
MATTRN			X	X	X			
MATVAR			X	X	X			
MATZER	X	X	X	X	X		X	
MATZRI	X							
MTREDF		X						X

Figure E-18. - Utility Matrix subroutine usage

Routine	Purpose
BGNPL	Initiates a plot and sets plotting variables
TITLE	Writes titles on graph
HEADIN	Writes heading on graph
FRAME	Controls framing on graph
GRAF	Sets graph dimensions and axis system
GRID	Draws grid marks on graph
CURVE	Draws a curve on graph
ENDPL	Outputs the graph
FRAMER	Draws a frame around graph
MARKER	Allows selection of a symbol
AXSDEF	Sets up axes
NUMBER	Puts numbers on axes
SETCHAR	Puts labels on axes
SETTER	Checks ranges and proportions of axes

Figure E-19. - Plotting routines for program TERMPL

Item	BEGIN	CORREL	KEYPAR	DISPAR	SEAPAR	TRNPAR	FLWGEN	TERMPL
A. Number of stations in input data file	30	30	30	30	30	30	30	30
B. Number of key stations per key station gen. group	10		10				10	
C. Number of key stations per key-sub gen. group	3			3			3	
D. Number of substations per key-sub group	10			10			10	
E. Number of stations per seasonal gen. group	10				10		10	
F. Number of key station gen. groups	10						10	
G. Number of key-sub gen. groups	10						10	
H. Number of seasonal gen. groups	10							10
I. Number of seasons* (Cannot exceed 12 in any program)		$\frac{12240}{A*J}$			$\frac{6000}{E*J}$	$\frac{390}{A}{}^{-1}$	$\frac{360}{A}$	$\frac{8100}{A*J}$
		$\frac{360}{A}$				$\frac{1800}{J}$		$\frac{1200}{J}$
		$\frac{4800}{A*A}$.						
J. Number of years*		$\frac{1500}{A}$	100	100	$\frac{500}{E}$	150		$\frac{1500}{A}$ 100

*Where multiple tests are indicated, all must be passed.

Figure E-20. - Problem size limitations