# OPERATING MANUAL

for

# THE TEXAS TRIP DISTRIBUTION PACKAGE

by

J. D. Benson
Assistant Research Planner

Charles E. Bell
Data Processing Programmer

Gary D. Long
Engineering Research Associate

and

Vergil G. Stover
Study Supervisor

Research Report 167-1

Urban Travel Forecasting
Research Study Number 2-10-71-167

Sponsored by the
Texas Highway Department
in cooperation with the
U. S. Department of Transportation
Federal Highway Administration

Texas Transportation Institute
Texas A&M University
College Station, Texas
September 1971

(Updated to the status of September 1972)

The opinions, findings, and conclusions expressed in this publication are those of the authors and are not necessarily those of the Federal Highway Administration.

TABLE OF CONTENTS

DATA SPECIFICATIONS

COMPUTATIONAL REQUIREMENTS

# ABSTRACT

The Texas Trip Distribution Package is a collection of computer programs designed to perform trip distributions featuring the application of a constrained interactance model. Other programs, available in the package, provide full support. This manual describes the performance capabilities, execution procedures, data specifications, and computational requirements which are related to the usage of the programs.

# SUMMARY

The Texas Trip Distribution Package is a complete collection of computer programs having the capability of performing several different types of trip distributions. The methods range from directionally expanding existing trip matrices to new totals, to performing synthetic distributions using a constrained interactance model.

The basic interactance model applies trip lengths directly in the distribution process and, consequently, needs no calibration. Other properties of the interactance model are similar to a gravity model, without 'F-factors'. By activating a constraint based upon interchange propensity only selected zone pairs enter in to the distribution rather than all possible zone pair combinations as with the gravity model. A sector structure may be imposed to permit a statistical analysis for, and correction of, sector interchange bias created by socio-economic-topographical travel barriers. Movements having external terminals may be processed simultaneously with the synthetic distribution of internal trips.

The Texas Trip Distribution Package is designed to interface with the Texas Small and Large Network Traffic Assignment Packages. It has been prepared for and implemented on IBM 360/50, IBM 360/65, and IBM 370/155 computers. Although it is programmed largely in the FORTRAN IV language for these computers it does take advantage of many of the options available under these operating systems and may, therefore, be somewhat sensitive to peculiarities between installations. For benefit of the user, simplicity and ease of operation have been emphasized in the development

of the package. A number of options are available to the user which provide the flexibility needed for unusual situations.

The package is capable of accomodating up to 4800 zones using a computer having 512,000 bytes of core storage. By making one minor program modification, the capacity can be varied to conform to the amount of core storage available; the minimum amount of core storage that would be required by the package is about 120,000 bytes.

# IMPLEMENTATION STATEMENT

The Texas Trip Distribution Package has been operational on the IBM 360 computer installation of the Texas Highway Department since September 1970. It has been used in conjunction with urban studies performed in El Paso, Victoria, Sherman-Denison, Brownsville, and Big Spring.

Several additions, revisions and improvements in the package have been implemented since transmittal of the original version of the program package. Research results from the continuing cooperative research program between the Texas Highway Department and the Texas Transportation Institute will undoubtedly lead to additional refinements. Revisions will be made in this manual as future revisions are implemented in the Texas Trip Distribution Package. The format and binding of this manual are designed to facilitate the inclusion of supplementary pages and the substitution of revised pages as necessary; a revision date will be indicated in the bottom margin of such pages.

# INTRODUCTION

The intention of this manual is to present a description of what is available in the Texas Trip Distribution Package, what alternatives may be elected, the default options, and some sample specifications. The user must consider what he has available and what he wants or needs and make certain decisions when utilizing the package. Most of the default options will provide satisfactory results and will simplify application in most situations.

The flexibility of the Texas Trip Distribution Package makes it most difficult to prepare a manual that describes every potential application. The variations between the urban transportation studies and the flexibilities in the operation of the package, precludes mapping every feasible alternative in detail. For example, application of the package in any of the five urban studies mentioned in the Implementation Statement was not performed in exactly the same manner.

The manual has been organized in four basic sections. Each of these sections will have different appeal to the analyst and program operator. The first section presents an operational overview of the package. It describes the data requirements and distribution options available. A list of the routines available and a description of some typical execution sequences are presented.

The second section describes the program elements. It discusses the procedure for specifying the routines to be executed, and presents a description of each individual routine. Each routine's description presents a statement of its function, lists the relevant parameters and corresponding default values, lists the associated data sets and data cards, notes any special execution requirements, discusses the operation of the routine, explains the printed output, and presents any relevant user considerations.

The third section is concerned with specifications. A list of all parameters is presented which displays the default values and a brief description of each parameter. A list of all data sets associated with the package is presented which displays the default unit numbers and a brief description of their contents. Data set formats are discussed and sample specifications are provided. The purpose, associated routines, entry sequences, card layout, and data description for each data card is presented. A cross reference table has been included in this section which shows the linkages of the various routines with the data sets and data cards.

The fourth section describes the computational requirements of the Texas Trip Distribution Package. The matter of program capacity is discussed and instruction are presented for varying the capacity. With the trade-off between computer core usage and service priority in mind, methods for estimating core usage based upon assumed program executions illustrated through sample calculations.

Additional data and knowledge are accumulated with every distribution of trips in a different urban area. This leads to refinements in the process. As future revisions are implemented in the package, this documentation will become obsolete. It has, therefore, been bound in a manner that will facilitate the insertion of supplements and replacement pages.

# EXECUTION PROCEDURES

OPERATIONAL PERSPECTIVE

DATA REQUIREMENTS

DISTRIBUTION OPTIONS

SAMPLE EXECUTION PROCEDURES

## OPERATIONAL PERSPECTIVE

The Texas Trip Distribution Package is basically a collection of routines. Each routine performs a specific function(s) in conjunction with the trip distribution process. The user of the package must specify which routines are to be executed and their order of execution subject to certain constraints. The selection of routines is dependent upon the type of data available or required, the type of distribution to be performed, the analysis to be performed, and the desired output from the package.

There are two fundamental types of routines contained in the Texas Trip Distribution Package: basic routines and auxiliary routines. The basic routines must normally be executed in a specific sequence to achieve a planned objective. Execution of the auxiliary routines is an option of the user; they generally contribute supplementary results.

Basic Routines

MODIFY
UNPACK
SCREEN
BUILD
EDIT
SET
REFINE
ACCEPT
IMPOSE
MODEL
EXPAND
SUM
SWITCH
PACK

Auxiliary Routines

PERUSE
PRINT
WRITE
EQUATE
LIST
RANDOM
GET
MATCH
ALTER
RESTART

## Basic Routines

The following is a brief summary of the functions performed by each of the basic routines:

MODIFY: This routine provides an instrument to define or modify any parameter value at any desired point during program execution.

UNPACK: This routine reformats a trip matrix from the format used in the Texas Large Network Package and the Texas Small Network Package and writes the trip matrix in the format used by this package.

SCREEN: This routine screens out everything but trip reports from the origin/destination survey data set, and writes another data set with abbreviated trip records containing only the data from the trip reports which are required for the trip distribution.

BUILD: This routine builds trip matrices of the type and trip purpose specified by a trip CATEGORY card from the sorted, abbreviated, trip records.

EDIT: This routine edits the interzonal travel separations obtained from the assignment package, and writes a separation matrix for use in trip distribution.

REFINE: This routine refines the parameter estimates of a set of auxiliary models.

ACCEPT: This routine accepts trip generations, trip lengths, and sector interchange bias compensations from data cards.

IMPOSE: This routine imposes movements which are to be included in the trip distribution.

MODEL: This routine models the distribution of travel interchanges and writes a modeled trip matrix.

EXPAND: This routine expands a trip matrix, directionally.

SUM: This routine sums two to five trip matrices.

SWITCH: This routine switches a production/attraction trip matrix to an origin/destination trip matrix.

PACK: This routine reformats any trip matrix prepared by this package into the format required by the Texas Large Network Package and the Texas Small Network Package.

SET: This routine sets arrays with trip generation, trip length, and sector interchange data. It can also produce a copy of the separation matrix data set in which the zone to zone movements which were detected from the survey data are "flagged". If this separation matrix is subsequently used as input to the MODEL routine, the "flagged" zone pairs will be imposed as eligible zone pairs.

## Auxiliary Routines

The functions performed by each of the auxiliary routines are as follows:

PERUSE: This routine provides a means to print the current parameter values at any desired point during program execution.

PRINT: This routine prints a trip matrix for inspection.

WRITE: This routine prints the separation matrix.

EQUATE: This routine equates centroids to sectors.

LIST: This routine prints the trip length distribution for each zone individually.

GET: This routine gets trip generation, trip length, and sector interchange data and prints these data for inspection.

MATCH: This routine matches the characteristics of two trip matrices against each other.

ALTER: This routine alters a trip matrix to compensate for changes in accessibility created through changes in a transportation system.

RESTART: This routine provides the capability of executing additional interations in the MODEL routine without rerunning previous iterations.

## DATA REQUIREMENTS

The Texas Trip Distribution Package requires at least one external input: a data set of interzonal travel separations in the form as prepared by both the Texas Small Network Package and the Texas Large Network Package. This data set must not be structured as a partitioned network.

If survey data are available, it may be supplied either in the form of the actual trip reports, or as a trip table from either the Texas Small Network Package or the Texas Large Network Package. These data sets, too, must not be in partitioned network structures. If the latter means is used, the trip table must be unpacked and organized as a matrix for application in the trip distribution. A routine, named UNPACK, has been provided for this purpose. If trip reports are supplied, which is preferable if both means are available, the SCREEN and BUILD routines may be applied to screen the relevant trip data and build the desired trip matrices.

If survey data are not available, the following information must be provided:

- productions and attractions or origins and destinations for each zone and external station (provided by GENERATION cards)
- a relative trip length frequency (normally provided by LENGTH cards)
- if the interaction constraint is to be applied, a production-interaction curve is necessary and is normally described by INTERACTION cards (note, however, this curve may also be described by the relative production-interaction model via the XP pro parameter).

The formula for relative production-interaction model is:

$$PZ_i = AN\ (1.0 - e^{(XP)(P_i)})$$

where:

$i$ = zone number

$PZ_i$ = expected number of interactions for production zone $i$

AN = number of zones with non-zero attraction volumes

XP = exponent for the relative production model

$P_i$ = production volume for zone $i$.

## DISTRIBUTION OPTIONS

The Texas Trip Distribution Package has the capability of performing several different types of trip distributions. The user may elect to use the routine EXPAND which directionally expands an existing trip matrix to new totals or to use the routine MODEL which produces a modeled trip matrix using a constrained interactance model. When using the MODEL routine, the user is provided with a number of options which directly affect the distribution process. These options include:

- The user may elect to impose interactions between zone pairs which, from the survey data, had one or more interchanges, under this option, the interactions in the modeled trip matrix will include these imposed interactions but will not necessarily be limited to the imposed interactions.
- The user may elect to impose interactions between any desired zone pairs. Again, the interactions in the modeled trip matrix will include the imposed interactions but will not necessarily be limited to the imposed interactions.
- The user may elect to relax the interchange limit constraint used in the model thereby allowing trips to be distributed between all eligible zone pairs.

In addition to the options which directly affect the distribution process, there are a number of options and alternatives associated with the parameters which are directly or indirectly used by the model.

## SAMPLE EXECUTION SEQUENCES

The flexibility of the Texas Trip Distribution Package makes it most difficult to describe every potential application. The variations between the urban transportation studies and the flexibilities in the operation of the package, precludes mapping every feasible alternative in detail. The examples presented in this section are not intended to be used as rigid guidelines nor to limit the use of the package to the types of applications illustrated. The intention is simply to provide the potential user with some insight into how the package might be applied to a few selected situations.

In each of the following examples, a brief description of the data available and the objectives to be achieved by the application of the package is provided. A flow chart is then presented which describes the execution sequence which might be used to achieve the desired objectives.

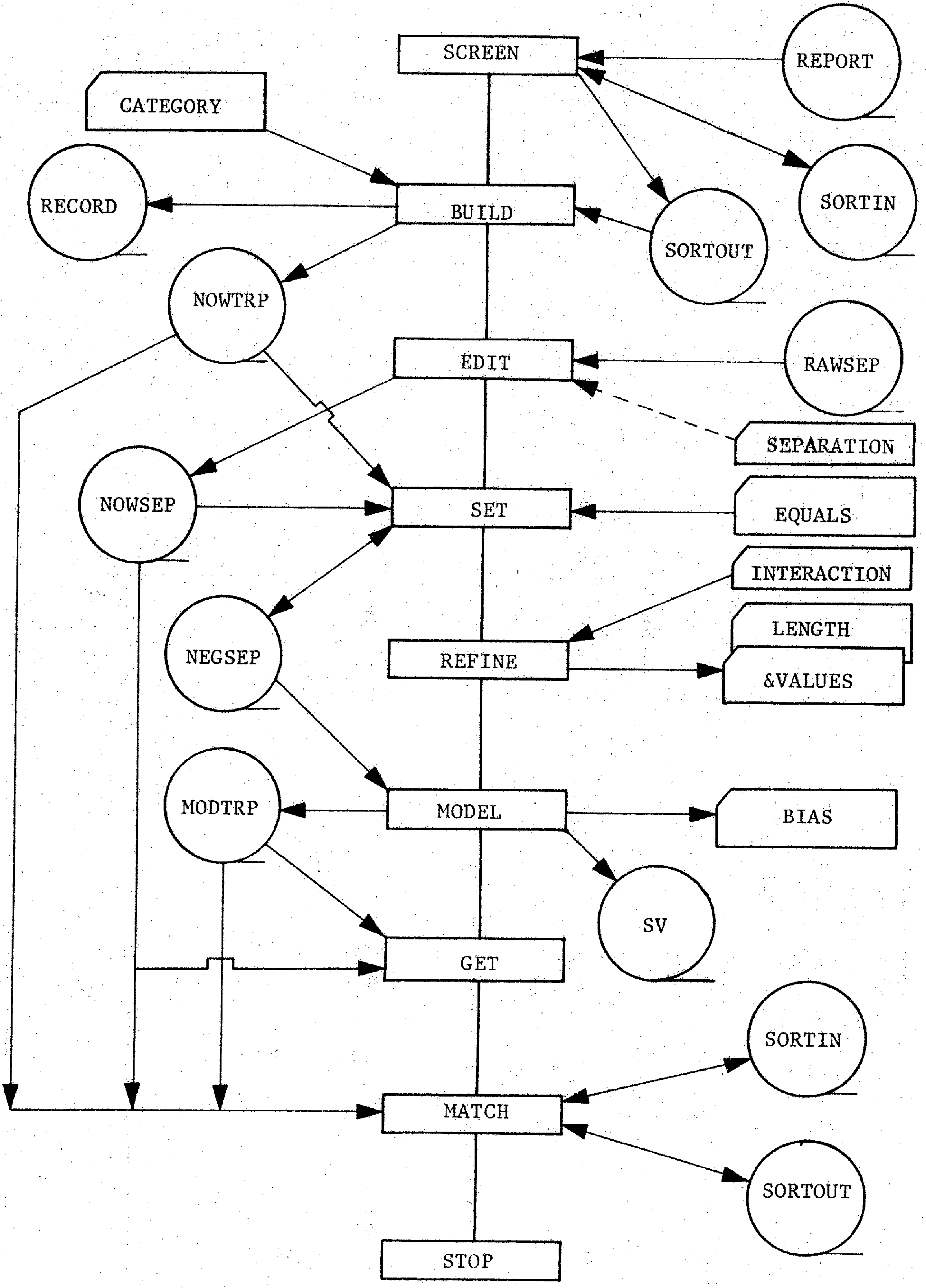
## EXAMPLE I: DISTRIBUTION OF EXISTING TRIPS

Data Available:

- A separation matrix prepared by either the Texas Small Network Package or the Texas Large Network Package (the RAWSEP data set)
- Trip reports obtained from an origin-destination survey (the REPORT data set)
- Sector equivalences (EQUALS cards)
- INTERACTION cards (note, the relative production-interaction model will be used if these cards are not provided)

Objectives:

- To obtain a trip matrix of existing internal home based work auto driver trips using survey data. (This requires a CATEGORY card specifying the type of trips desired. The NOWTRP data set will contain the desired survey trip matrix.).
- To distribute existing internal home based work auto driver trips using a constrained interactance model with interactions detected from survey data imposed. (The MODTRP data set will contain the desired modeled trip matrix.)
- To obtain a summary of the sector-to-sector movements for analysis.
- To obtain a comparison of the survey trip matrix and the modeled trip matrix.


SCREEN
REPORT
CATEGORY
SORTIN
RECORD
BUILD
SORTOUT
NOWTRP
RAWSEP
EDIT
SEPARATION
NOWSEP
SET
EQUALS
INTERACTION
NEGSEP
LENGTH
REFINE
&VALUES
MODTRP
MODEL
BIAS
SV
GET
SORTIN
MATCH
SORTOUT
STOP

## EXAMPLE II: DISTRIBUTION OF FUTURE TRIPS*

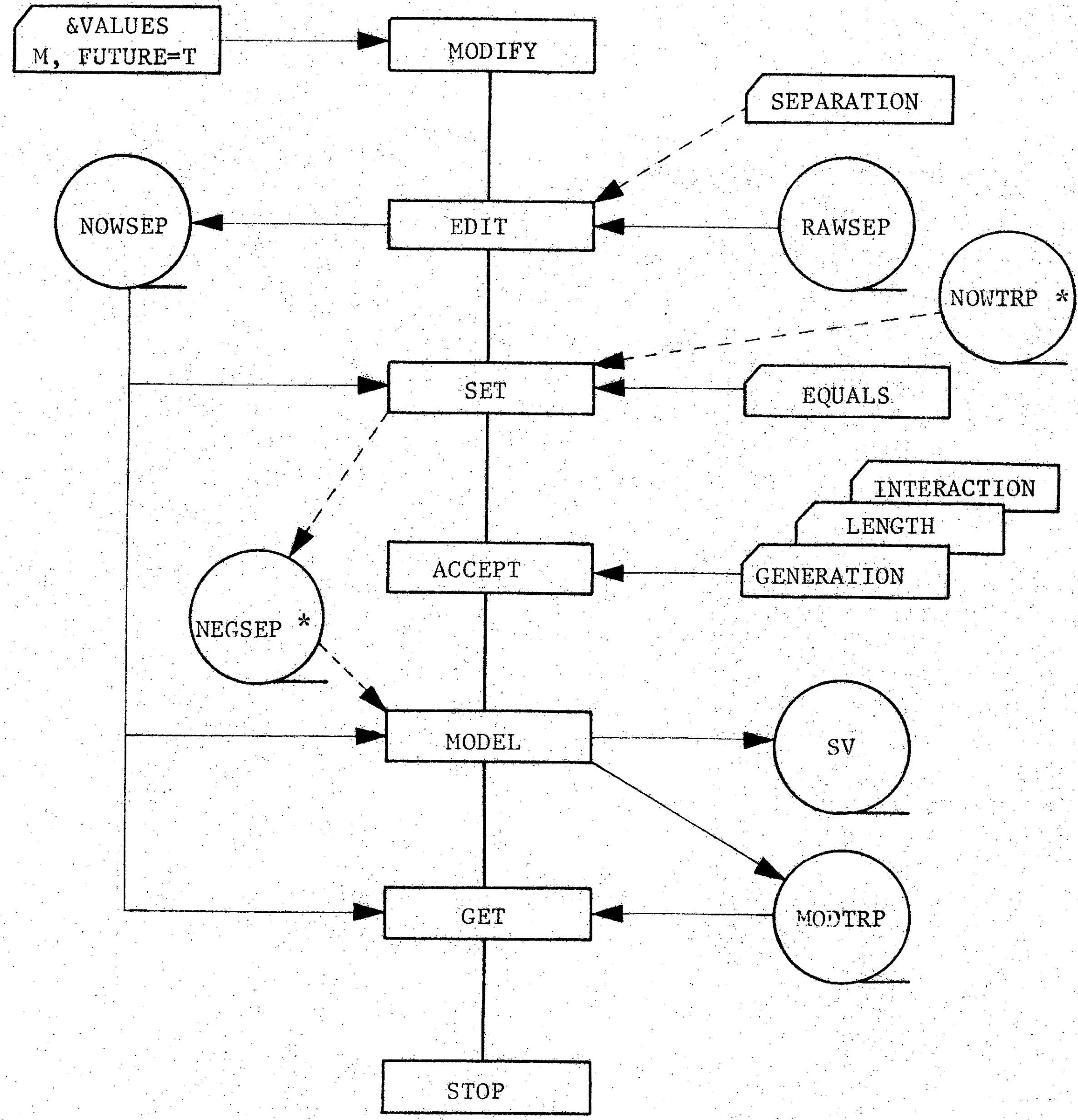
Data Available:

- A separation matrix prepared by either the Texas Large Network Package or the Texas Small Network Package (the RAWSEP data set).
- Modeled future productions and attractions (GENERATION cards)
- Modeled future trip length frequency (LENGTH cards)
- INTERACTION cards
- Largest internal zone number (parameter M)
- Sector equivalances (EQUALS cards)

Objectives:

- to distribute future trips using a constrained interactance model
- to obtain sector-to-sector movements for analysis

*It should be noted that this example is equally applicable to a synthetic study.



* may be a dummy data set

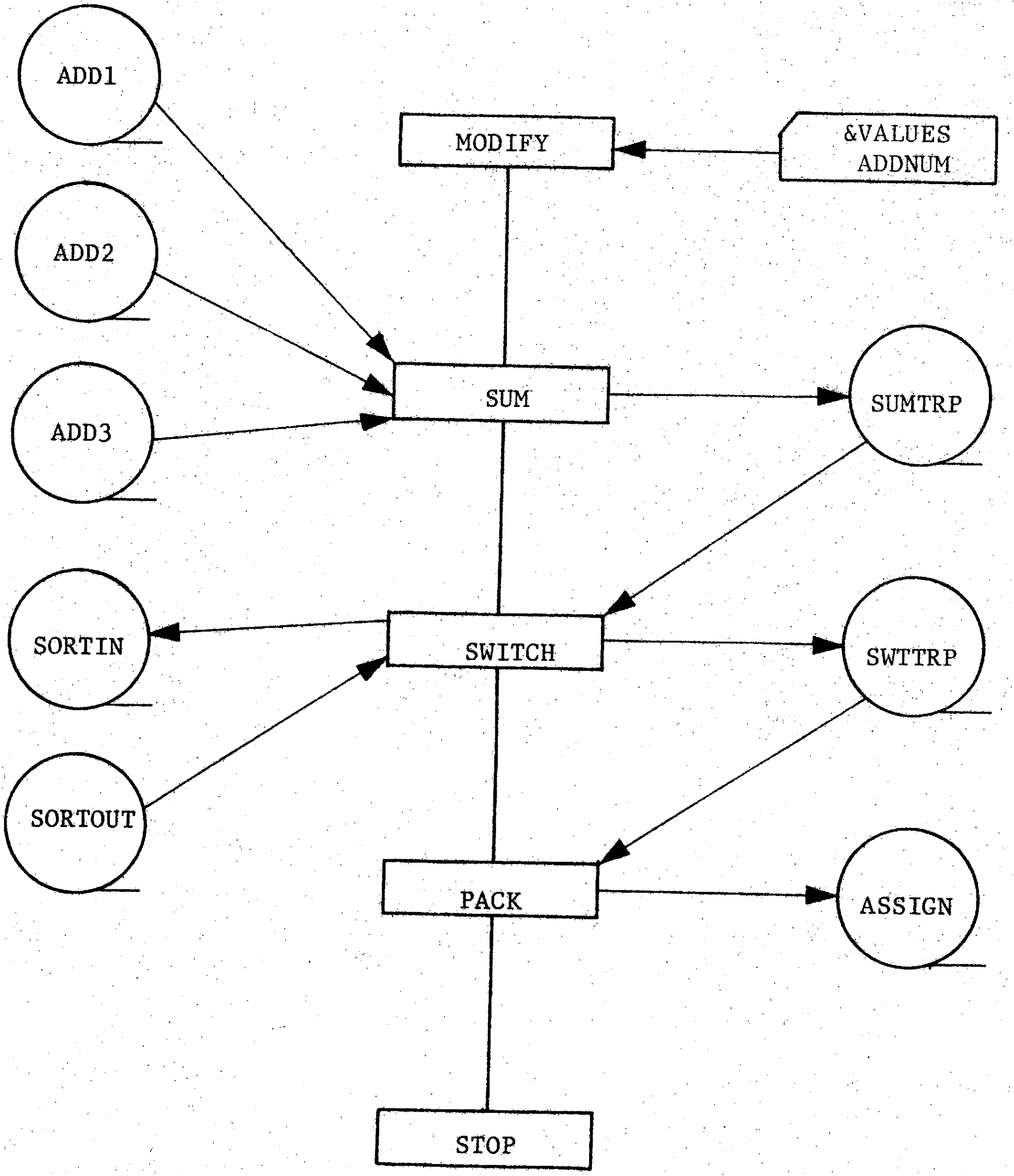
## EXAMPLE III: COMBINING TRIP MATRICES FOR ASSIGNMENT

Data Available:

- Three production/attraction trip matrices prepared by this package (ADD1, ADD2, and ADD3 data sets)
- The ADDNUM parameter (i.e. the number of trip matrices to be combined)

Objectives:

- To combine the three given production/attraction trip matrices into a single production/attraction trip matrix (the SUMTRP data set)
- To switch the production/attraction trip matrix (the SUMTRP data set) to an origin/destination trip matrix (the SWTTRP data set)
- To reformat the origin/destination trip matrix (the SWTTRP data set) for input into either the Texas Large Network Package or the Texas Small Network Package (the Assign data set).


ADD1
MODIFY
&VALUES
ADDNUM
ADD2
ADD3
SUM
SUMTRP
SORTIN
SWITCH
SWTTRP
SORTOUT
PACK
ASSIGN
STOP

# PROGRAM ELEMENTS

INTRODUCTION

DESCRIPTIONS OF INDIVIDUAL ROUTINES

ABNORMAL TERMINATIONS

# INTRODUCTION

All routines in the Texas Trip Distribution Package are referenced by name. The names merely need to be entered on the CONTROL cards in the sequence in which the routines are to be executed. The CONTROL cards must be the first records in the input data stream entered from unit 5. HEADING cards may be intermingled with CONTROL cards in any manner, but no other cards should be encountered before the last CONTROL card. Each CONTROL card that is encountered is scanned for valid control entries. Any improper entries or invalid coding will result in program termination immediately after the first card not identified as either a CONTROL or HEADING card is encountered. Such a termination will produce a STOP code of 1.

Regardless of how many CONTROL cards are used, only a total of 40 routines may be processed at one time. If more than 40 routines are specified, the first 40 entries will be executed and the program will then terminate with a STOP code of 10. The entry STOP does not actually reference a routine but is a command used to terminate the execution of the Texas Trip Distribution Package. If subsequent entries are listed on a CONTROL card following the STOP command, they will be processed and checked for validity, but will never be executed because the program will terminate when the STOP command is encountered. When the Texas Trip Distribution Package encounters the STOP entry in its control sequence, the package terminates with a STOP code of 0. At some computer installations, a STOP code of 0 is considered as the normal termination and does not appear on the processing job log.

It is not essential to enter the STOP command in the CONTROL card sequence. If the STOP command is omitted, a stop entry will be furnished by the subroutine which interprets the CONTROL card so the Texas Trip Distribution Package may terminate with a STOP code of 0, if processing has progressed properly.

The HEADING card may be entered at any point in the data card input stream except in the middle of a contiguous set of data. Normally, a HEADING card may appear as the first card in the input data stream to serve as identification of the cards following it. Occasionally, if a HEADING card is placed as the last card in the data card input stream, an error message will result which implies that the program attempted to read more input data than was provided. This error message may simply be disregarded, or the practice of feeding a HEADING card last may be avoided, or an extra card such as a blank card might be fed in as the last card.

Each time control is passed from one routine to another, the data card input stream is checked for the existence of a HEADING card appearing as the next record. This feature permits changing the heading between routine execution. In normal operation, this feature is not frequently needed.

If successive HEADING cards are entered amid the data card input stream, it may not be desired that the second HEADING card change the heading immediately prior to the execution of the second routine in the sequence. In this instance, it may be desirable to utilize the MODIFY routine to space the second HEADING card as desired. In other words, a &VALUES card which contains no parameters, but is closed by an &END, may be inserted between the two consecutive HEADING cards. At the point at which it is desired to change the heading, the MODIFY entry can be placed in the control sequence.

The HEADING card may also be used as a separator to distinguish between two different data sets which might be used sequentially. As an example, two different sector structures may be utilized in the analysis of travel patterns within a large urban area. It might be planned to execute the GET routine using one sector structure and then redefine the sector structure by executing the EQUATE routine and re-execute GET to summarize the movements with regard to the second sector structure which might contain more or less detail than the first structure. Clearly, there must be some way for the program to distinguish where the first set of EQUALS cards ends, and the second set begins. A HEADING card between the two sets of EQUALS cards will aptly fulfill this purpose. The message on the separator HEADING card could simply be duplicated from the original HEADING card, however, in the context of the particular example cited, it would very likely be beneficial to change the heading message.

It is fully acceptable to operate the Texas Trip Distribution Package without supplying any heading messages. If no HEADING card is encountered, or until the first HEADING card is encountered, headings will simply be blank.

## DESCRIPTIONS OF INDIVIDUAL ROUTINES

The description of each routine in the Texas Trip Distribution Package has been divided into eight sections. These sections state the routine's function, execution requirements, parameter references, data set references, data card references, operation, printed output, and user considerations.

The first section, entitled "Function", contains a very brief statement describing the routine's function.

The second section is titled "Execution Requirements." The statements under this heading will indicate whether the program is an independent or dependent routine. This classification is based upon the arrays which are held in core. Several of the routines require that arrays be defined before the routines are executed; these are classified as dependent routines since they require the prior execution of another routine to define the arrays. The routines to be executed in advance of the dependent routines are noted. Due to reuse of much of the core storage, several of the independent routines can destroy key arrays. Therefore, the status of the key arrays is noted with regard to each routine.

The third section is "Parameter References." Under this heading, either one or two subheadings may appear entitled "Required" and/or "Defined." The "Required" column refers to parameters which are required for proper execution of the routine. The "Defined" column refers to parameters which are either evaluated or revised during the execution of the routine. All parameters referenced under either of these two subheadings appear in the VALUES namelist. Therefore, should the user desire to inspect the values of any of the parameters following the execution of any routine, he merely needs to execute the routine PERUSE following the routine in question. Should the user desire to either define or modify the values of any input parameters, he may use the routine MODIFY, and specify the values for the desired parameters immediately before executing the routine in question. Several of the parameters in the "Required" list are shown equal to a value which is enclosed in brackets. Values enclosed in brackets are default values. These are shown in instances when it is likely that prior routine executions have not affected the parameter. Parameters for which no default value is shown should have been defined by the user, or by a prior execution of another routine. The user must verify that all "Required" parameters are properly specified. It is recommended that after the user determines the sequence of routines which he intends to execute that he backtrace the parameters starting with the last routine to be executed. In essence, the user should be sure that all parameters appearing in a "Required" list for a given routine either (a) appear in a "Defined" list of a routine which will be executed prior to the execution of the given routine, (b) will be initialized through the execution of the MODIFY routine prior to the execution of the given routine, or (c) the default value associated with the parameter is the value desired by the user. Some of the "Defined" parameters are shown being equated to another parameter. These are pointers which are being redefined to point to a different data set.

The fourth section is "Data Set References". Any of three columns may appear under this heading labeled as "Input", "Scratch", and/or "Output". The entries appearing under any of these columns may be either symbolic data set references or pointers. Pointers merely refer to certain data sets, and these are changed following the execution of various routines. This feature usually relieves the user from having to define or change data set references if he exercises descretion in the sequence in which he executes the routines. It should be observed that the user may redefine the pointer and symbolic data set references through the execution of MODIFY prior to executing any routine in question. Following most symbolic data set names is a value enclosed in brackets. This is the default value of the data set. A value, not enclosed in brackets, which follows a data set is the unit number which that data set must have; and the user is provided no option to redefine such a data set reference.

The fifth section concerns "Data Card References". Again, column headings marked "Input" and/or "Output" may be encountered. The input data cards must be placed in the data card input stream in the sequence in which they are listed. It should be noted that if stray data cards should appear in the input data stream, these cards will not be processed properly. All of the routines operate in the same manner with respect to card input data. When a program reaches the point where it is to process data cards, it checks the next entry in the data card input stream for the appropriate type of data card. If the data cards are the type specified under the "Data Card References" section, they are read until a data card is reached which is not the desired type. This last card then will be saved until needed. With this procedure, no delimiter is necessary to indicate the end of a particular group of data cards. It should be remembered that HEADING cards are the only data cards which may be placed in the input data stream which do not require an explicit program reference, in the CONTROL entry sequence, to be read and processed properly.

The sixth section is "Operation". This section consists of a general discussion of how each routine actually operates. Errors leading to abnormal termination conditions are noted.

The seventh section describes the "Printed Output". All of the printed output bears page headings which describe the output. Once the user becomes acquainted with the package, it will not be necessary for him to continually refer to these discussions. In the execution of some of the routines, identical or similar output is received, and rather than repeat long discussions, a mere statement has been provided indicating where the particular type of output is discussed.

The eighth section is entitled "User considerations". The discussions under this heading vary in nature from items of computational efficiency to the basic philosophy of the distribution procedure.

# ACCEPT

## Function

The ACCEPT routine accepts trip generations, trip lengths, sector interchange bias compensations, and the production-interaction curve from data cards.

## Execution Requirements

ACCEPT is an independent routine when executed in conjunction with EXPAND. It requires no initialization, destroys no key arrays, and prepares some key arrays used by other programs.

The ACCEPT routine is a dependent routine in all other applications. It must be preceded by an execution of SET even if a survey data trip matrix is not available. Intervening executions of any routines which destroys key arrays will jeopardize the functioning of ACCEPT. The ACCEPT routine prepares key arrays which are used by other programs.

## Parameter References

| Required | Defined |
|---|---|
| XP (required only if INTERACTION cards are not provided and EXEMPT=F) | TV<br>UT |
| TYPE = [blank] (optional) | AN<br>PN |

## Data Set References

None

Data Card References

Input

FORMAT (for Generation cards)

GENERATION cards

FORMAT (for Length cards)*

LENGTH cards*

FORMAT (for BIAS cards)*

BIAS cards

FORMAT (for INTERACTION cards)*

INTERACTION cards*

Operation

The ACCEPT routine begins by attempting to read a FORMAT cards. If this card is not encountered, execution of the Trip Distribution Package is terminated immediately with a STOP code of 5. If the identification code on the FORMAT card is equal to the type parameter, or if either of these is blank, the format is accepted. If additional format records are encountered, they are judged by these criteria. The last encountered acceptable format is used. If no acceptable format is found, the program terminates with a STOP code of 3.

All GENERATION cards are read. These cards may be in any sequence, but if one entry is not encountered for every centroid and external station, the missing centroid and/or external station numbers are printed and the Trip Distribution Package terminates with a STOP code of 3.

After processing the GENERATION cards, the program searches for LENGTH cards, BIAS cards, INTERACTION cards, and associated FORMAT cards. All of these entries are optional, including the FORMAT cards. If FORMAT

*Optional

cards are not provided for the LENGTH cards and BIAS cards, the last encountered FORMAT card will be used. This is inappropriate since the formats are usually incompatible. The INTERACTION cards, if present, must have a format card. If LENGTH, BIAS, and INTERACTION cards are being supplied, the LENGTH cards should be entered first and the INTERACTION cards last.

Printed Output

A listing of the INTERACTION cards.

User Considerations

If existing trips are being distributed and survey data are available the internal productions read in through the GENERATION cards are scaled so that the total productions from the survey data will equal the total productions fed in on GENERATION cards. The internal attractions are always scaled so that the total attractions will equal the total productions. The trip length distribution for internal movements is scaled so that the total trips in the trip length distribution equals the total internal productions. Likewise, the external distribution is scaled so that the total trips equals the total trip generations through the external stations.

If existing trips are being distributed and survey data are available, the trip length distribution will be obtained from the survey data. Any values entered on LENGTH cards will override those found in the survey data.

If any BIAS cards are encountered, the corresponding factors are applied during the trip distribution and no bias detection is attempted by the program, even if existing trips are being distributed and survey data are available.

If ACCEPT is being used in conjunction with the EXPAND routine, no scaling is performed.

## ALTER

### Function

The ALTER routine alters a trip matrix to compenstate for changes in accessibility created through changes in a transportation system.

### Execution Requirements

The ALTER Program is a dependent routine. It must be preceeded by the execution of ACCEPT or REFINE to establish the desired trip length frequency. It does not affect the key arrays.

### Parameter References

| Required | Defined |
|---|---|
| | MS = NEWSEP |
| | MT = ALTRP |

### Data Set References

| Input | Output |
|---|---|
| NOWSEP = [4] | ALTTRP = [22] |
| NEWSEP = [16] | |
| MODTRP = [3] | |

### Data Card References

None

### Operation

The ALTER routine reads one record from the trip matrix, one record from the old separation matrix, and one record from the new separation matrix. Each interchange volume within the record being considered is adjusted based upon the change in travel separation between the old and new separation matrix. A record containing the adjusted interchange volumes is then written and the process repeated.

Printed Output

None

User Considerations

This routine is currently of research interest only and has been included only for the convenience of on-going research. It is not, therefore, recommended for use in urban transportation studies.

# BUILD

## Function

The BUILD routine builds trip matrices of the type and trip purpose specified by a trip CATEGORY card from the sorted, abbreviated, trip records.

## Execution Requirements

BUILD is an independent routine. It requires no initialization. It does not effect the key arrays. If BUILD is executed after SCREEN has been executed, the SORTOUT data set is copied to the RECORD data set for preservation. If SCREEN is not executed prior to BUILD, the RECORD data set is assumed to have been previously prepared and the SORTOUT data set is not copied.

## Parameter References

| Defined |
|---|
| N |
| M |
| TYPE |
| MT = NOWTRP |

## Data Set References

| Input | Output |
|---|---|
| SORTOUT = 10 | RECORD = [14] |
| RECORD = [14] | NOWTRP = [2] |

## Data Card References

| Input |
|---|
| CATEGORY |

## Operation

The BUILD routine reads and interprets one CATEGORY card. The CATEGORY card provides the information which controls the selection of the trip reports used to construct the trip matrix. If a CATEGORY card is not encountered, the Texas Trip Distribution Package terminates immediately with a stop code of 8. After the CATEGORY card is interpreted, the abbreviated trip records are scanned for entries of the desired category, and the desired trip matrix is formed. The trip matrix is written on the NOWTRP data set.

## Printed Output

One line is printed during the execution of the BUILD routine. This line displays the trip matrix identification as supplied through the CATEGORY card, the sum of the trips contained in the matrix, and a string of consecutive zeroes and ones. The sum is printed for the user to check against any other source which he has available, and the number string is printed to aid in examining the CATEGORY card if an error is apparent. The number string may be interpreted as forty one-digit numbers which are referenced by position. The first nine should be ignored. The tenth refers to category 10, etc. A zero means it is ignored; a one means it is desired.

## User Considerations

If no executions of MATCH or SWITCH are planned, and if BUILD is not to be re-executed during the processing job, the RECORD data set may be defined as a dummy data set and the SORTOUT data set preserved in its place and later entered as the RECORD data set at the next execution of BUILD.

If a CATEGORY card is not entered, the SORTOUT data set is copied on the RECORD data set. This will preserve the data set thereby avoiding the re-execution of the SCREEN routine.

# EDIT

## Function

The EDIT routine edits the interzonal travel separations obtained from the assignment package, and writes a separation matrix for use in trip distribution.

## Execution Requirements

EDIT is an independent routine. It requires that the value of parameter M be preset. It does not prepare any key arrays, but if executed indiscriminately it could destroy some of them. However, since EDIT prepares the separation matrix used by most of the other routines, this controls its execution sequence and almost eliminates the danger of destroying key arrays.

## Parameter References

| Required | Defined |
|---|---|
| M | NF |
| EXTEND = [0] | MS = NOWSEP |

## Data Set References

| Input | Output |
|---|---|
| RAWSEP = [8] | NOWSEP = [4] |

## Data Card References

| Input |
|---|
| SEPARATION (optional) |

## Operation

The EDIT routine is used to edit the interzonal separations that result from the assignment package and convert them to a form usable by the Trip Distribution Package. The EDIT routine first scans the entire interzonal separation data set in order to determine the largest value. This value is then written in a parameter record at the front of the data set. If additional codes are to be used, the largest value is incremented by the number indicated by the variable EXTEND. Any zero value found in the interzonal separations is replaced by a value of unity. Separation cards are optional and may be supplied to replace any value found in the interzonal separation data set with any desired value.

## Printed Output

The EDIT routine prints the table titled "SEPARATION REVISIONS RESULTING FROM THE EDITING PROCESS" and the maximum internal separation.

## User Considerations

An optional field is provided in the SEPARATION cards for special separation codes. These special separation codes must be integers in the range of one to the value of the parameter EXTEND plus one. When a SEPARATION card is encountered with a special separation code, the EDIT routine will compute a replacement separation value for the specified zone pair as follows:

$$\begin{bmatrix}\text{replacement}\\ \text{separation}\\ \text{value}\end{bmatrix} = \begin{bmatrix}\text{largest internal separation}\\ \text{detected in the RAWSEP data}\\ \text{set}\end{bmatrix} + \begin{bmatrix}\text{special}\\ \text{separation}\\ \text{code}\end{bmatrix}$$

The user must be careful when using special separation codes for interzonal separations since the selection of eligible zone pairs for the interaction constraint in the MODEL routine is based on the accessibility measure:

$$\frac{\text{Attraction volume}}{\text{Separation}}$$

It is possible, therefore, that few, if any, of the interzonal movements with a special separation code would be selected as eligibile zone pairs. To avoid this problem will require that the interzonal movements with a special separation code be imposed via ADMIT cards in the IMPOSE routine. Intrazonal movements do not pose a problem since they are selected as eligible zone pairs so long as they have non-zero production and attraction volumes regardless of their separation.

If SEPARATION cards are used, the EDIT routine will check the SEPARATION cards for the following conditions:

- A special separation code which is greater than the value of the parameter EXTEND plus one.
- A separation value which is greater than the largest internal separation (including the separation values computed for the special separation codes).
- A SEPARATION card with both a separation value and a special separation code.
- An invalid centroid number (i.e., a centroid number which is greater than the value of the parameter N)

If either of these conditions exist then a warning message will be printed and the JOB will be abnormally terminated following the EDIT routine with a stop code of 16. When either of these conditions are encountered, the following values will be entered in the separation matrix built by EDIT:

- If a special separation code is encountered which is greater than the value of EXTEND plus one, then the SEPARATION card is ignored.

- If a separation value is encountered which is greater than the value of the largest internal separation (including the separation values computed for the special separation codes), then the SEPARATION card is ignored.

- If a SEPARATION card is encountered with both a separation value and a special separation code, then the special separation code is ignored and the separation value (if valid) is used.

- If an invalid centroid number is encountered then the SEPARATION card is ignored.

The EDIT routine sets the value of the parameter NF as follows:

$$\text{NF} = \left[\begin{array}{l}\text{Largest internal separation}\\ \text{detected in the RAWSEP data}\\ \text{set}\end{array}\right] + \text{EXTEND} + 1$$

## EQUATE

### Function

The EQUATE routine equates centroids to sectors.

### Execution Requirements

EQUATE is an independent routine. It requires that the value for parameter N be present. It prepares one key array which defines the sector equivalency. It does not destroy any key arrays. The SET and GET routines contain automatic calls to the EQUATE program provided that sector equivalences have not been established previously. The EQUATE routine may be used to replace one set of sector equivalences with a different set.

### Parameter References

Required

N

### Data Set References

None

### Data Card References

Input

EQUALS (optional)

### Operation

Sector to centroid equivalence are obtained by the EQUATE routine through EQUALS cards. The EQUATE routine attempts to read an Equals card. In the event that this card is not encountered, the EQUATE routine establishes an equivalence of all zones with sector one. Subsequent processing interprets this to mean that the sector equivalence feature is not being used. If equals cards are encountered, they are processed until the last equals card has been read. All zones are then examined to see if equivalences with sectors have been established. If any unequivalenced zones are discovered, a default sector is established. The default sector is assigned the next number larger than the last defined sector. All remaining zones are equated to the default sector. Multiple entries for any zone are noted in a message and the last encountered equivalence is retained. A table describing the resulting equivalences is finally written.

It should be noted that centroid and external station numbers and sector numbers are checked during processing. Any invalid entries are disregarded. EQUALS cards may be processed in any order. It is recommended that sectors be numbered consecutively starting with the value of one, but this is not a requirement. If the number of sectors used exceeds the capacity of the package, a message will be written and processing terminated. It is cautioned that the use of more than about 15 sectors may be found to be unwieldy in the printed output. Only 15 numbers are printed per line and if more than 15 sectors are used the output tables become "folded".

## Printed Output

A table of the zone to sector equivalences is printed. If any zones are equated to a default sector a message is printed identifying the sector to which they were equivalenced. A message is printed if multiple entries are encountered for any zones.

## User Considerations

The routine EQUATE allows the user a convenient means for correcting mistakes made in keypunching EQUALS cards. Since the EQUATE routine uses the last encountered equivalence, all that is required is that corrected EQUALS cards be added to the back of the already punched EQUALS cards. Messages regarding multiple entries should be ignored in this situation.

# EXPAND

## Function

The EXPAND routine expands a trip matrix, directionally.

## Execution Requirements

The EXPAND program is a dependent routine. It must always follow the execution of either SET or ACCEPT. Intervening executions of either GET or MATCH will destroy input arrays which are required by EXPAND.

## Parameter References

| Required | Defined |
|---|---|
| LIMIT = [5] | MT = MODTRP |

## Data Set References

| Input | Output |
|---|---|
| NOWTRP = [2] | MODTRP = [3] |

## Data Card References

None

## Operation

The EXPAND routine performs iteratively. The limit parameter governs the number of iterations which are repeated. The resulting trip matrix is written on the MODTRP data set during the last iteration.

## Printed Output

At the end of each iteration through the routine EXPAND, a table is printed which reflects the success of the balancing process in applying a destination volume constraint. Each entry in the table refers to an origin zone number, and the user is referred to the discussion presented for the routine MODEL for an interpretation of this output.

## User Considerations

In using the EXPAND routine for expanding external-through movements, each iteration requires skipping through all of the internal centroids just to reach the external centroids.

# GET

## Function

The GET routine gets trip generation, trip length, and sector interchange data and prints these data for inspection.

## Execution Requirements

GET is an independent routine. It requires no initialization. It prepares no key arrays but can destroy some if executed improperly.

## Parameter References

| Required | Defined |
|---|---|
| PLOT = [F] | TV |
| XP (if plot = T and INTERACTION cards have not been input) | AN |
| | PN |

## Data Set References

| Input | Output |
|---|---|
| MT | PLOTTAPE (if plot = T) |
| MS (DD Dummy optional) | |

## Data Card References

Input

EQUALS (optional)

## Operation

The GET routine first checks to see if secotr equivalences have been defined. If they have not been defined and EQUALS cards are available; then the sector equivalences are established. The GET routine checks to see if a separation matris is available, and if it is not, the trip length data will be sacrificed. If the parameter PLOT is equal to TRUE, then Calcomp plots will be prepared.

## Printed Output

The reader is referred to the SET routine for a discussion of the printed output.

## User Considerations

The availability of a separation matrix has been made optional to allow the user to examine the trip generations by zone and by sector without having to wait for the network coding to be completed. Furthermore, it permits examining the results from the EXPAND routine which is an application that does not require a separation matrix and represents an instance when one is not likely to be available. Calcomp plots will not be prepared if the separation matrix is defined as a dummy data set.

If GET is executed as an isolated entry with PLOT equal ture, only the trip length distribution of the associated trip and separation matrices will be plotted. If GET is executed in a sequence with prior executions of either MODEL or REFINE, the plot will show both the desired and resulting trip length distributions on the same graph, for comparison.

## IMPOSE

### Function

The IMPOSE routine imposes movements which are to be included in the trip distribution.

### Execution Requirements

IMPOSE is an independent routine. It requires no initialization. It does not affect key arrays.

### Parameter References

| | Defined |
|---|---|
| | MS = IMPSEP |

### Data Set References

| Input | Output |
|---|---|
| MS | IMPSEP = [15] |

### Data Card References

Input

ADMIT (or LOCAL) cards

### Operation

The IMPOSE routine reads ADMIT cards and determines movements to impose during the trip distribution. If ADMIT cards are not encountered, then the Trip Distribution Package terminates with a STOP code of 6. ADMIT cards must be in numerical sort on the production (or origin) centroid numbers. Each ADMIT card is read, interrogated for errors, and processed.

## Printed Output

No printed output results from a successful execution of the IMPOSE program. However, any errors detected during its execution are printed.

## User Considerations

The user should be aware that the IMPOSE routine makes only one entry for each entry represented on the ADMIT cards. In order to admit both directions of travel between two zones, two distinct entries must be made through the ADMIT cards.

# LIST

## Function

The LIST routine prints the trip length distribution for each zone individually.

## Execution Requirements

LIST is an independent routine. It requires no initialization. It does not affect any key arrays.

## Parameter References

None

## Data Set References

### Input

MT = [2]

MS = [4]

## Data Card References

None

## Operation

The LIST routine simply reads the trip matrix and separation matrix, simultaneously, and prints the trip length characteristics for each production zone.

## Printed Output

The output resulting from the LIST routine is similar to that described for the trip length characteristics under the SET routine.

## User Considerations

Due to the execution time required and the amount of printed output prepared, the LIST routine should be used only when necessary.

# MATCH

## Function

The MATCH routine matches the characteristics of two trip matrices against each other.

## Execution Requirements

MATCH is an independent routine. It requires no initialization. It destroys all key arrays.

## Parameter References

| Required | Defined |
|---|---|
| SAMPLE = [0.125] | SIZE |
| AMOUNT = [110000] | |

## Data Set References

| Input | Scratch |
|---|---|
| NOWSEP = [4] | SORTIN = 9 (DD DUMMY optional) |
| NOWTRP = [2] | SORTOUT = 10 |
| MODTRP = [3] | |

## Data Card References

None

## Operation

The MATCH routine reads a record from three data sets (the survey trip matrix, the model trip matrix, and the separation matrix), then performs comparisons with regard to corresponding interchange volumes. The reciprocal of the nominal sampling rate is used to establish the cell intervals for a cross classification of interchange frequencies between the survey and model trip matrices. High-volume interchanges which do not fall within the limits of this table are listed on the SORTOUT data set for a separate analysis. If the SORTIN data set is a dummy data set only movements in which the differences between the survey interchange volumes and the model interchange volumes which do not exceed N/10 (where N = the number of centroids and external stations) will be presented on the trip volume difference analysis.

## Printed Output

There are five different types of comparisons that are performed by the MATCH routine. The first of these is an analysis of the interchange volumes with respect to production centroids. The second is a volume frequency cross-classification table. Third, a comparison of the high-volume interchanges is printed. Fourth, a comparison of the low-volume interchanges is printed. The fifth is an analysis of the trip volume differences.

## User Considerations

If SORTIN is a dummy data set, the BLKSIZE for the SORTOUT data set can be reduced to a small value, such as 244, with no loss of efficiency and some saving of core storage.

# MODEL

## Function

The MODEL routine models the distribution of travel interchanges, and writes a modeled trip matrix.

## Execution Requirements

MODEL is a dependent routine. It must be preceded by executions of REFINE and/or ACCEPT. Intervening executions of any routine which destroys key arrays will jeopardize the functioning of MODEL.

## Parameter References

| Required | Defined |
|---|---|
| FUTURE | MT = MODTRP |
| UT | |
| LIMIT = [5] | MS = NOWSEP |
| EXEMPT = [F] | |
| DUMP = [T] | |

## Data Set References

| Input | Output |
|---|---|
| MS | MODTRP = [3] |
| | SV = [25] (if DUMP = T) |

## Data Card References

| | Output | |
|---|---|---|
| | FORMAT | (if FUTURE = F) |
| | BIAS | |

## Operation

If EXEMPT is false, the model will be subjected to the interaction constraint, and the eligible zone pairs are selected in a preprocessing phase. The desired number of eligible zone pairs for a given production zone is determined by the production-interaction curve and the eligible zone pairs are selected based on their accessibility to the production zone. The trip distribution for the first interation is then performed. No trip matrix is written until the last iteration is reached. After the initial distribution is performed, the relative values are corrected, and the processes reiterated.

The parameter LIMIT indicates the number of iterations to be repeated. If the parameter FUTURE is FALSE, bias factors will be computed two iterations before the iteration limit is reached. Of course, if no sector structure is utilized, the bias correction feature is inoperative.

If the EXEMPT parameter is TRUE, the model operation is exempted from the interaction constraint and interchange volumes will be calculated for all zone pair combinations. Otherwise, interchange volumes will be computed only if: (1) the eligibility of the zone pair has been imposed through application of the IMPOSE routine, (2) the eligibility of the zone pair has been imposed because of a non-zero survey volume, or (3) because the zone pair was selected as an eligible zone pair during the preprocessing phase is of large enough volume to escape elimination by the interaction constraint.

If the DUMP Parameter is true, various parameters and arrays will be saved after the last iteration so that the process can be restarted using the RESTART routine.

## Printed Output

If EXEMPT is false three tables result from the preprocessor phase which selects the eligible zone pairs. All three of these tables have "ITERATION 0" in their headings. The first table is the Accessibility Eliminator Function. The columns of this table contain the zone number, the production volume, the desired number of interactions as determined by the production-interaction curve and the number of eligible zone pairs including eligible zone pairs imposed either from survey data or the prior execution of the IMPOSE routine. The remainder of this table has been included only for the purpose of monitoring the operation of the program and, therefore, should be of no interest to the transportation analyst.

The second table is entitled "INTERNAL AND EXTERNAL ELIGIBLE ATTRACTION INTERACTIONS". The three columns in this table contain the zone number, the attraction volume and the number of production zones with which the attraction zone may interact. The third table produced during the preprocessor phase is entitled "ELIGIBLE TRAVEL INTERACTIONS". The two columns in this table contain the separations and the number of eligible zone pairs at each separation. The total number of eligible zone pairs is printed as the sum of the second column. If EXEMPT is true the preprocessor phase is omitted and the above three tables are, of course, not produced.

Three tables of printed output result from each iteration of the model. Each of these tables reflects the success of the balancing process in applying the indirect constraints. First in the printed output is the Attraction Volume Balance. Each entry refers to an attraction zone number and successive columns show the desired attraction volume, the resulting attraction volume from the model application, the difference between these two volumes, the percentage of error in the model volume as opposed to the desired volume, the weighted significance of the combination of absolute and relative error, the relative attraction value, the correction factor applied to improve the results of the next iteration, and the new relative attraction value which reflects the adjustment of the correction factor. After these items are listed for each of the attraction zones, some statistical measures are printed which indicate the overall agreement between desired and resulting attraction volumes for all zones considered in the group.

The Trip Length Balance is printed next. The same measures are printed as in the Attraction Volume Balance. Each entry, however, refers to a separation value. The last three entries represent the external movements. In addition, the desired and resulting percentage of trips is printed for each separation. Summary statistics are presented at the end.

The SECTOR INTERCHANGE BALANCE is the last of the printed output that appears. Each line of output refers to one sector-to-sector movement. The table contains the sector numbers, the desired sector interchange volume, the resulting sector interchange volume, the difference between the two volumes, the tolerance volume, the percent error, the weighted significance of the combination of absolute and relative errors, the correction factor, and the new bias factor. Values in the column headed "RELATIVE" of the sector interchange balance can be observed to be flagged with an asterisk. In all such cases, the corresponding value will be observed to be 1.0. This notation is used to designate movements which are not being corrected. Only during the last two iterations will any movements be corrected. Of course, if bias corrections are feed in as input data or if no survey data are available, the sector interchange balance will not even appear.

## User Considerations

The user should be cautious in interpreting the statistical measures which are provided to indicate the degree of agreement between between desired and model resulting values. The statistical measures can be deceptive. For instance, what normally might be deemed to be an excellent correlation can very easily accompany only a mediocre correspondence between desired and resulting values. The reason for this is simple: there is a very obvious correlation between desired and resulting values. It is the degree of agreement which needs to be evaluated. However, no single index yet discovered does an adequate job of supplying this information. Therefore, the user should examine the individual data values and come to his own conclusion regarding the acceptability of the agreement. This does not mean that the summary statistics cannot be used as a guide.

The column entitled chi-square has some interesting properties. The chi-square sum is shown at the end of the data list and this can indeed be interpreted as the chi-square goodness-of-fit test and this statistic checked against a tabled value. It should be recognized, however, that the chi-square test is very sensitive to "tail" discrepancies, and consequently, a single entry may produce a significant statistical difference with respect to the test. The chi-square column is presented here as a means for identifying which individual entries contribute most to the disagreement. Individual chi-square entries represent the product of the difference and the percent error columns. The difference column is not an acceptable measure alone since large differences are important with respect to small volumes but may be in a practical sense insignificant with sufficiently large volumes. This reasoning would suggest that the percent error column might be an adequate indicator, and it is with respect to large volumes, but a modeled value may be in error by 100 percent for a small volume and this error be of no real significance. Since chi-square represents the product of the absolute and relative error, it has some attractive characteristics. If both the absolute and relative errors are small, their products will be very small. If either the absolute or relative error is large and the other is very small, the product will be small. As the magnitudes of either error increase, the product increases. When both errors are large, the product is very large. Therefore, large chi-square terms will serve to identify entries which may have unacceptable errors in a combined absolute and relative sense. If there exist many entries, as there will in the attraction volume balance for a large urban area such as Houston, there is no cause for alarm simply because one or two of the entries display large chi-square values and thus, cause the sum to be large enough to imply that significant statistical differences exist.

# MODIFY

## Function

The MODIFY routine provides an instrument to define or modify any parameter value at any desired point during program execution.

## Execution Requirements

MODIFY is an independent routine. It requires no initialization. It does not affect the key arrays. It may be executed at any point in which it is desired to change any value appearing in the VALUES namelist.

## Parameter References

### Defined

Any desired parameter in the VALUES namelist

## Data Set References

None

## Data Card References

### Input

&VALUES

## Operation

Execution of the MODIFY routine causes an immediate read of the next card in the data card input stream for an &VALUES record. This record is interpreted by the FORTRAN namelist feature. Any parameter appearing in the VALUES namelist may be entered on the &VALUES card.

The value entered for the parameter will replace the former value. If the &VALUES card is coded improperly, the Texas Trip Distribution Package will terminate with a STOP code of 12.

Printed Output

After every execution of the MODIFY routine, the entire VALUES namelist is printed to permit inspection of the current status of the parameter values. This provides the user with the opportunity to verify that his changes were entered as desired, and provides a permanent record in the printed output of the parameter values which were used.

User Considerations

Only the parameters and corresponding values which may be fitted on one data card may be entered during any single execution of the MODIFY routine. Therefore, each &VALUES card requires a separate execution of the MODIFY routine.

A parameter defined by the MODIFY routine may be overridden or redefined by the execution of any routine which defines the same parameter. For example, if the parameter XP was defined using the MODIFY routine, the subsequent execution of the REFINE routine would calculate a value for XP and substitute the calculated value for the current value of XP.

## PACK

### Function

The PACK routine reformats any trip matrix prepared by this package into the format required by the Texas Large Network Package and the Texas Small Network Package.

### Execution Requirements

PACK is an independent routine. It requires no initialization. It does not affect any key arrays.

### Parameter References

None

### Data Set References

| Required | Defined |
|---|---|
| MT | ASSIGN = [13] |

### Data Card References

None

### Operation

The operation of the PACK routine is simple. It reads a trip matrix (in the format used by this package) record by record, converts each record to the format used by the Texas Large Network Package and Texas Small Network Package, and outputs the new record on the ASSIGN data set. If any interchange volumes are encountered during the process which are too large to be packed in assignment form, the maximum acceptable volume is substituted, and a message is written to signal this change. It will be very rare for this condition to occur since the interchange volume must have a numerical value that exceeds 64,000.

## Printed Output

There is no printed output from a successful execution of the PACK routine. However, if an interchange volume in excess of 64,000 is encountered, a message will be printed which reads VOLUME TO LARGE TO ASSIGN and three numbers will follow. The first number represents the production zone, and the second number represents the attraction zone. The third number indicates the magnitude of the trip volume.

## User Considerations

None

# PERUSE

## Function

The PERUSE routine provides a means to print the current parameter values at any desired point during program execution.

## Execution Requirements

PERUSE is an independent routine. It requires no initialization. It does not affect the key arrays. It may be executed at any point in which a printed record of the current parameter values is desired. It should be noted that the parameter values are printed at key points in the execution of the Texas Trip Distribution Package.

## Parameter References

None

## Data Set References

None

## Data Card References

None

## Operation

Execution of the PERUSE routine merely causes printing of the VALUES namelist so that the current status of the parameter values will be displayed.

## User Considerations

None

## PRINT

### Function

The PRINT routine prints a trip matrix for inspection.

### Execution Requirements

PRINT is an independent routine. It requires no initialization. It does not affect any key arrays.

### Parameter References

None

### Data Set References

#### Input

MT

### Data Card References

None

### Operation

The PRINT routine simply reads a trip matrix and prints the volumes. Each production or origin zone is treated separately, and the interchange volumes to successive attraction zones are printed ten per row.

### Printed Output

The PRINT routine prints the trip matrix contained on the MT data set. A table is printed for each non-zero production zone. Each table contains the trip volumes from the production zone to each centroid in the network. The table consists of ten columns which are read from left to right such that the first row contains the trip volumes to the first ten centroids, the second row contains the trip volumes to centroids 11 through 20 etc.

## User Considerations

The printed output from this routine is quite lengthy and, therefore, should not be executed unless needed. If the trip matrix is saved, this program can always be executed later, if desired.

# REFINE

## Function

The REFINE routine refines the parameter estimates of a set of auxiliary models.

## Execution Requirements

REFINE is a dependent routine. It must be preceded by an execution of SET. Intervening executions of any routine which destroys key arrays will jeopardize the functioning of REFINE. The REFINE routine may only be executed when survey data are available. It prepares key arrays which are used by other routines.

## Parameter References

| Required | Defined |
|---|---|
| SAMPLE = [0.125] | XP |
| PLOT = [F] | UT |

## Data Set References

| | Output |
|---|---|
| | PLOTTAPE (if PLOT = T) |

## Data Card References

| | Output |
|---|---|
| | VALUES |
| | FORMAT |
| | LENGTH |

## Operation

The REFINE routine uses the array values established by the SET routine and refines various model parameters based upon the survey data. If SET is not executed prior to REFINE, execution is terminated with a STOP code of 4.

## Printed Output

One line is printed for each of the four models which are calibrated and this line of output displays various statistical indicaters which describe the relative success of the calibration.

## User Considerations

None

# RESTART

## Function

The RESTART routine provides the capability of executing additional iterations in the MODEL routine without rerunning previous iterations.

## Execution Requirements

RESTART is a dependent routine. It must be preceded by execution of MODEL with DUMP = T in a previous JOB.

## Parameter References

### Required

LIMIT = [5]

DUMP = [T]

### Defined

MT = MODTRP

MS = NOWSEP

AN

EXEMPT

EXTEND

FUTURE

M

MR

N

NF

NR

OMIT

ONE

PN

SAMPLE

TV

UT

XP

## Data Set References

### Input

MS (separation matrix used in previous execution of MODEL)

RS = [26] (the SV data set outputted from MODEL)

IMPSEP (if FUTURE = F)

### Output

MODTRP = [3]

SV = [25] (if DUMP = T)

Data Card References

Output

FORMAT<br>
BIAS (if FUTURE = F)

Operation

The RESTART routine uses the information stored on the RS data set (built by MODEL in a previous JOB) to initialize various parameters and arrays so that the MODEL routine may perform additional iterations without rerunning previous iterations. The RESTART routine then calls the appropriate subroutines within the MODEL routine in order to resume the iterative process.

Printed Output

Three tables of printed output are produced for each additional iteration. These tables are the same tables produced by the MODEL routine for each iteration (i.e., the "Attraction Volume Balance" table, the "Trip Length Balance" table, and the "Sector Interchange Balance" table).

User Considerations

The LIMIT parameter does not specify the number of additional iterations but the total number of iterations. For example, if the MODEL routine had run five iterations in the previous JOB and two additional iterations are desired, then the LIMIT parameter should be set to seven by using the MODIFY routine immediately before the RESTART routine.

The RS data set used as input to the RESTART routine is the SV data set built by the MODEL routine (or the RESTART routine) in the previous JOB. A new SV data set will be built by RESTART if the value of the parameter DUMP is true thus providing the capability of again restarting the process to perform still additional iterations at a later time.

## SCREEN

### Function

The SCREEN routine screens out everything but trip reports from the origin/destination survey data set, and writes another data set with abbreviated trip records containing only the data from the trip reports which are required for the trip distribution.

### Execution Requirements

SCREEN is an independent routine. It requires no initialization. It does not affect the key arrays. Either the SORTOUT data set should be protected, or else the BUILD routine should be executed immediately after SCREEN to preserve the sorted trip records.

### Parameter References

| Required | Defined |
|---|---|
| OMIT = [F] | N |
| AMOUNT = [110000] | SIZE |

### Data Set References

| Input | Scratch | Output |
|---|---|---|
| REPORT = [12] | NOWTRP = [2] | SORTOUT = 10 |
|  | SORTIN = 9 |  |

### Data Card References

None

Operation

The SCREEN routine serves as the first step in preparing a trip matrix from the trip reports resulting from an origin/destination survey. It reads the data set of trip reports, screens out all but the data essential for trip distribution, and writes another data set containing only this abbreviated information. Dwelling unit reports and other extraneous information on the trip report data set are disregarded. Trips with external terminals which are reported in the internal survey are disregarded. Volumes of trips merely passing through the study area are divided in half. The passenger's trip purpose, referred to as the secondary trip purpose, is substituted as the purpose of trip for all internal serve passenger trips. Trips having destinations at home are entered twice in the abbreviated data set. This double entry permits future construction of either origin/destination or production/attraction trip matrices. After the end of the trip report data set is reached, the abbreviated trip record data set is sorted. The number of records involved in the sort is indicated by the parameter SIZE. Since external trip reports constitute a large portion of the records involved in the processing, an indicator named OMIT has been provided to omit external trips if they are not desired in any trip matrix. The indicator OMIT simply needs to be set to TRUE.

The SCREEN routine uses the system sorting routines. A parameter named AMOUNT designates the amount of computer storage to be used as a sort work area. This amount may be adjusted to regulate the program region size. It should never be reduced below 40,000 bytes, and larger amounts improve sorting efficiency.

## Printed Output

None

## User Considerations

If the SCREEN routine is executed without the subsequent execution of the BUILD routine the sorted trip reports are placed on unit 10. This data set should be saved for input into the BUILD routine thereby avoiding an unnecessary execution of the SCREEN routine.

## SET

### Function

The SET routine sets arrays with trip generation, trip length, and sector interchange data. It also produces a "flagged" separation matrix which may be used to impose the movements which were found in the survey data upon the trip distribution.

### Execution Requirements

SET is an independent routine. It requires no initialization. It prepares key arrays which are used by other routines.

### Parameter References

Defined

AN

PN

SAMPLE

TV

MS = NEGSEP (if not DUMMY)

### Data Set References

| Input | Output |
|---|---|
| MT = [2] (DD DUMMY optional) | NEGSEP = [1] (DD DUMMY optional) |
| MS = [4] | |

### Data Card References

Input

EQUALS (optional)

## Operation

The SET routine sets array values for later use in the Trip Distribution Package. The routine first checks to see if sector equivalences have been defined and if not, checks to see if EQUALS cards are available. If EQUALS cards are encountered, they are processed by the EQUATE routine.

If an existing trip matrix is not encountered, the distribution of separations in the separation matrix is initialized and control is returned after printing the parameter namelist.

## Printed Output

During the operation of SET several tables are printed. The first of these tables is a Trip Generation Summary which indicates a trip production volume, trip attraction volume, intrazonal trip volume, the number of production and attraction interactions, and the average volume per interaction for each zone. At the end of this table, the total volume over all zones is shown, as well as the number of potential travel interactions among all zone pair combinations which are not eliminated by having zero generations at one terminal.

The next table which is printed exhibits the trip length characteristics for the entire urban area. Each separation interval which exists is shown along with its corresponding zone pair incidence, interaction frequency, trip volume, and other measures calculated from combinations of these parameters. Totals are shown at the bottom of the table plus the characteristics of trips with external terminals are also summarized at the bottom.

The next four tables summarize the travel characteristics by sector. The first table indicates the number of zone pair combinations which exist among various sectors. The second table indicates the number of sector entries which have travel interactions between the zone pairs. The third table indicates the number of trip interchanges between zone pairs in the sectors. The fourth table indicates the tolerances based on the variance in interchange volumes between the zone pairs within the sectors.

If the trip matrix was prepared by the BUILD program in the Trip Distribution Package, the average sample proportion is calculated. The number of zones having non-zero production and attraction volumes are counted.

## User Considerations

In executing the routine SET, the user should note two things. If the data set on unit MT is a dummy data set, a synthetic study is assumed, execution of subroutine SET is bypassed, and the parameter namelist is printed. Also, if the trip matrix on unit MT has not been built by the execution of the routine BUILD, the sampling rate, SAMPLE, will not be calculated and thus will retain its default value of 0.125. In such a case, it would be necessary for the user to run the routine MODIFY to input the sampling rate for the trip matrix being used.

# SUM

## Function

The SUM routine sums two to five trip matrices.

## Execution Requirements

SUM is an independent routine. It requires no initialization. It does not affect key arrays.

## Parameter References

| Required | Defined |
|---|---|
| ADDNUM = [2] | MT = SUMTRP |
| ONE = [1.0] | |

## Data Set References

| Input | Output |
|---|---|
| ADD1 = [3] | SUMTRP = [20] |
| ADD2 = [17] | |
| ADD3 = [18] | |
| ADD4 = [19] | |
| ADD5 = [23] | |

## Data Card References

None

## Operation

The SUM routine interrogates the value of parameter ADDNUM to determine how many matrices are to be summed, and it then assumes these are located sequentially on the add units beginning with ADD1. The sum routine reads each matrix record one at a time, sums the trip volumes, and writes the sum record. Normally, the parameter ONE should have a value of 1.0 as it defaults. Each value in the second and all subsequent trip matrices which are being summed are multiplied by the parameter ONE. This feature is provided to permit factoring up or down a trip matrix during a SUM process.

## Printed Output

None

## User Considerations

The parameter ONE may be set to a value of minus one if it should be desired to subtract one trip matrix from another. Caution should be exercised since a danger exists that negative volumes might result. This feature was originally provided to allow subtracting a trip matrix of only internal movements from a trip matrix containing both internal and external movements to obtain only external movements.

## SWITCH

### Function

The SWITCH routine switches a production/attraction trip matrix to an origin/destination trip matrix.

### Execution Requirements

SWITCH is an independent routine. It requires no initialization. It does not affect key arrays.

### Parameter References

| Required | Defined |
|---|---|
| AMOUNT = [110000] | MT = SWTTRP |
| | SIZE |

### Data Set References

| Input | Scratch | Output |
|---|---|---|
| MT = [2] | SORTIN = 9 | SWTTRP = [24] |
| | SORTOUT = 10 | |

### Data Card References

None

### Operation

The SWITCH routine reads a trip matrix, divides each interchange volume by two, writes each non-zero half volume on a sort input data set. The data set of half volumes is sorted. The trip matrix is reread (again dividing volumes by two) and the half volumes from the trip matrix and the sorted data set are merged and the output trip matrix is written on the SWTTRP data set.

The SIZE parameter reflects a count of the number of sort records to be sorted. The amount parameter controls the amount of core storage which is allocated for sort usage.

Printed Output

None

User Considerations

Due to the presence of the sort, the SWITCH routine consumes a sizeable amount of computer time and computer storage. Therefore, it is advisable not to execute it more than is absolutely necessary.

# UNPACK

## Function

The UNPACK routine reformats a trip matrix from the format used in the Texas Large Network Package and Texas Small Network Package and writes the trip matrix in the format used by this package.

## Execution Requirements

UNPACK is an independent routine. It requires no initialization. It does not affect the key arrays.

## Parameter References

Required

M

TYPE = [blank]

## Data Set References

| Input | Output |
|---|---|
| INTRIP = [21] | NOWTRP = [2] |

## Data Card References

None

## Operation

The UNPACK routine simply reads each record from a trip table in the format used by the Texas Large Network Package and the Texas Small Network Package, reformats each record for use by this package, and writes the reformatted records on the NOWTRP data set.

## Printed Output

None

## User Considerations

None

# WRITE

## Function

The WRITE routine prints the separation matrix.

## Execution Requirements

WRITE is an independent routine. It requires no initialization. It does not affect any key arrays.

## Parameter References

None

## Data Set References

### Input

MS

## Data Card References

None

## Operation

The WRITE routine is used to print the separation matrix. Each zone is treated individually and within each zone each separation value is treated individually. Starting with the separation value of one, all destination zones having this particular separation will be listed in a string. Then the next separation value will be treated.

## Printed Output

WRITE outputs the separation matrix contained on the MS data set. This is done in tabular form for each centroid and external station.

User Consideration

Due to the execution time required and the amount of printed output (a minimum of one page for each centroid and external station), the WRITE routine should be used only when necessary.

## ABNORMAL TERMINATIONS

The following table lists the stop codes which may be encountered, the locations in which they may be encountered, and the probable cause.

| Stop Code | Location | Cause |
|---|---|---|
| 0 | DIRECT | Normal termination (STOP instruction encountered) |
| 1 | SCAN | Invalid CONTROL entry encountered. |
| 2 | EDIT | Fewer centroids in RAWSEP than in trip reports. |
| 3 | ACCEPT | GENERATION cards incomplete; if none listed, no FORMAT identification matched TYPE sought. |
| 4 | REFINE | Array values not prepared by SET. |
| 5 | ACCEPT | FORMAT card for GENERATIONS not encountered. |
| 6 | IMPOSE | ADMIT cards not encountered. |
| 7 | VERIFY | Array lengths exceed specified capacities. |
| 8 | BUILD | CATEGORY card not encountered. |
| 9 | MATCH | Sector combinations must be set at 320 or more when executing MATCH. |
| 10 | MAIN | The number of routines specified for execution exceeded the capacity of 40. |
| 11 | DIRECT | M exceeds N. |
| 12 | PREVUE | Invalid &VALUES card encountered. |
| 13 | ACCEPT and REFINE | FORMAT card for INTERACTION cards not encountered when INTERACTION cards are present. |
| 14 | ACCEPT and REFINE | Production volumes out of sort on INTERACTION cards. |

| Stop Code | Location | Cause |
|---|---|---|
| 15 | MODEL | One or more separation values were encountered at which there are no eligible zone pairs while the expected trip volume at each of these separations were greater than one percent of the total trips. |
| 16 | EDIT | Either special separation code has been encountered in SEPARATION cards which is greater than the value of EXTEND plus one or a SEPARATION card has been encountered which has both a replacement separation value and a special separation code. |

# DATA SPECIFICATIONS

PARAMETER REFERENCES

DATA CARD REFERENCES

DATA SET REFERENCES

CROSS REFERENCE TABLE

## PARAMETER REFERENCES

There are 46 parameters in the Texas Trip Distribution Package. Access to these parameters is provided by the MODIFY routine which uses the FORTRAN NAMELIST option. All parameters are contained in a single NAMELIST named VALUES.

Although a large number of parameters are accessible, only a few (if any) will need to be changed by the user under normal operations. The primary reason for the extensive number of accessible parameters is the flexibility provided for unusual applications.

The following table provides a brief description of each of the parameters and its default value. It should be remembered that the default value for any parameter is overridden by the execution of a routine which defines the parameter.

PARAMETER DESCRIPTIONS AND DEFAULT VALUES

| Parameter | Default Value | Variable Type | Description |
|---|---|---|---|
| ADD1 | 3 | Integer | The unit number containing the first trip matrix to be summed by the SUM routine |
| ADD2 | 17 | Integer | The unit number containing the second trip matrix to be summed by the SUM routine |
| ADD3 | 18 | Integer | The unit number containing the third trip matrix to be summed by the SUM routine |
| ADD4 | 19 | Integer | The unit number containing the fourth trip matrix to be summed by the SUM routine |
| ADD5 | 23 | | The unit number containing the fifth trip matrix to be summed by the SUM routine |
| ADDNUM | 2 | Integer | Number of trip matrices to be summed |
| ALTTRP | 22 | Integer | The unit number which contains the altered trip matrix outputed from the ALTER routine |
| AMOUNT | 110000 | Integer | Number of bytes in core to be used in sorting |
| AN | 1.0 | Real | Number of zones having nonzero attraction volumes |
| ASSIGN | 13 | Integer | The unit number which contains the model trip matrix packed in the format for input to the assignment packages |
| DUMP | TRUE | Logical | When either MODEL or RESTART routine is executed and this variable is true, then necessary arrays and parameters will be saved on the SV data set after the last iteration so that additional iterations may be run, if desired, using the RESTART routine |

PARAMETER DESCRIPTIONS AND DEFAULT VALUES
(cont.)

| Parameter | Default Value | Variable Type | Description |
|---|---|---|---|
| EXEMPT | FALSE | Logical | If EXEMPT equals TRUE, the interchange constraint is not applied |
| EXTEND | 0 | Integer | Factor used to increase the maximum internal separation in the EDIT routine to provide for special separation codes |
| FUTURE | FALSE | Logical | If FUTURE is true, a future or synthetic distribution is being performed |
| IMPSEP | 15 | Integer | The unit number which contains the separation matrix from the IMPOSE routine |
| INTRIP | 21 | Integer | The unit number which contains the packed trip matrix outputed from the assignment packages |
| LIMIT | 5 | Integer | Number of iterations performed in execution of the MODEL or EXPAND routines |
| M | 0 | Integer | Largest internal centroid number |
| MODTRP | 3 | Integer | The unit number which contains the trip matrix constructed by MODEL or EXPAND |
| MR | 0 | Integer | Largest sector number (includes any default sector) |
| MS | 4 | Integer | The unit number which contains the separation matrix being used |
| MT | 2 | Integer | The unit number which contains the trip matrix being used |
| N | 0 | Integer | Largest external station number |
| NEGSEP | 1 | Integer | The unit number which contains the separation matrix from the SET routine |
| NEWSEP | 16 | Integer | The unit number which contains a future separation matrix used as input to ALTER |

## PARAMETER DESCRIPTIONS AND DEFAULT VALUES (cont.)

| Parameter | Default Value | Variable Type | Description |
|---|---|---|---|
| NF | 1 | Integer | Largest internal separation including the special separation codes (i.e., including EXTEND + 1 special separation codes) |
| NOWSEP | 4 | Integer | The unit number which contains the edited separation matrix from the EDIT routine |
| NOWTRP | 2 | Integer | Pointer indicating the unit number which contains the trip matrix from BUILD |
| NR | 0 | Integer | Number of sector pair combinations |
| OMIT | FALSE | Logical | When OMIT equals TRUE, the external trips are not included in the data set constructed by the SCREEN routine |
| ONE | 1.0 | Real | Factor used in the SUM routine |
| PLOT | FALSE | Logical | When PLOT equals TRUE, calcomp plots are outputed from the execution of the REFINE routine <u>or</u> the GET routine |
| PN | 1.0 | Real | Number of zones having nonzero production volumes |
| RAWSEP | 8 | Integer | The unit number which contains the interzonal separations from the assignment package (used as input to EDIT) |
| RECORD | 14 | Integer | The unit number which contains the abbreviated trip records (used in conjunction with the SCREEN and BUILD routines) |
| REPORT | 12 | Integer | Pointer indicating the unit number which contains the survey trip reports in 104 byte records |
| RS | 26 | Integer | The unit number which contains the SV data set which was previously built by either the MODEL or RESTART routine when executed with DUMP = T |

## PARAMETER DESCRIPTIONS AND DEFAULT VALUES (cont.)

| Parameter | Default Value | Variable Type | Description |
|---|---|---|---|
| SAMPLE | 0.125 | Real | Nominal sampling rate as a fraction |
| SIZE | 0 | Integer | Indicates the number of records sorted |
| SUMTRP | 20 | Integer | The unit number which contains the trip matrix constructed by the SUM routine |
| SWTTRP | 24 | Integer | The unit number which contains the trip matrix constructed by the SWITCH routine |
| SV | 25 | Integer | The unit number on which necessary arrays and parameters are saved when either the MODEL or the RESTART routine is executed with DUMP = T |
| TV | undefined | Real | Total trip volume for specified purpose(s) |
| TYPE | 1077952576 | Integer | Contains a four-byte literal used for identification |
| UT | 0.0 | Real | Number of zone pairs having trip interchanges |
| XP | -0.00068 | Real | Exponent for the relative production model |

## DATA CARD REFERENCES

There are 11 types of data cards associated with the Texas Trip Distribution Package. Each type contains a literal identification field. Nine of the card types are either used as input for specific routines or are outputted from certain routines for later use. The other two card types are used either to specify the routines to be executed (i.e. the CONTROL card) or to specify the heading to be used on printed output (i.e. the HEADING card).

Each of the following data card descriptions have been divided into five sections. The sections describe the card's purpose, the routines directly associated with the card type, the entry sequence required, the card layout, and a description of the data contained in the card.

# ADMIT

## Purpose

The ADMIT cards enter individual movements which are to be imposed in the trip distribution.

## Associated Routines

### Input

IMPOSE

## Entry Sequence

All ADMIT cards must be in numerically increasing sequence of production (or origin) zone numbers. Attraction (or destination) zone numbers, with respect to each production zone, may be in any order.

## Card Layout (fixed): FORMAT (A4, 1X, I5,I5)

| Columns | Type | Content |
|---|---|---|
| 1 - 5 | Literal | 'ADMIT' or 'LOCAL' |
| 6 - 10 | Integer | Production (or origin) zone number |
| 11 - 15 | Integer | Attraction (or destination) zone number |
| . | . | . (consecutive fields of five columns) |
| . | . | . |
| . | . | . |
| 76 - 80 | Integer | Attraction (or destination) zone number |

## Data Description

Each data card must contain the production zone number in columns

6-10. The remainder of the data card is interpreted as fourteen five-column fields which are provided for attraction zone numbers. Blank fields are disregarded, attraction zone numbers exceeding the last valid attraction zone are indicated with an error message and disregarded, and all fourteen fields are examined for valid entries. All data must be right-justified in the fields.

As many cards as are necessary may be supplied for each production zone. Consecutive attraction zone numbers appearing in strings may be entered through a shorthand notation by coding only the first and last zone number in the string, with the last number preceeded by a minus sign. The shorthand coding may not span to the next card and the first attraction zone number appearing on any card must not be preceeded by a minus sign.

All entries are error checked. Duplicate entries are ignored.

# BIAS

## Purpose

The BIAS cards enter correction factors which compensate for biases in travel movements between sector pairs.

## Associated Routines

| Input | Output |
|---|---|
| ACCEPT | MODEL |

## Entry Sequence

The BIAS cards immediately follow the FORMAT card which describes their format. BIAS cards may be in any order.

## Card Layout (variable); sample FORMAT (A4, 6X, 2I5, F10.3)

| Field | Type | Content |
|---|---|---|
| 1 | Literal | 'BIAS' |
| 2 | Integer | Sector number or movement code |
| 3 | Integer | Sector number or blank |
| 4 | Real | Bias correction factor |

## Data Description

BIAS cards are interpreted by a variable format as supplied by a FORMAT card. The word BIAS should appear as the first item on every BIAS card. The second and third items on each BIAS card should index the appropriate movement, and the bias correction factor should be punched as the fourth item. BIAS cards are not required if the correction factor is 1.0. Movements not entered on BIAS cards are assumed not to require bias correction.

Movements may be identified on BIAS cards by either one of two schemes. The two related sector numbers may be entered. The other method is to enter the actual movement index. The two methods are distinguished on the data cards by whether or not the field corresponding to the second sector number is blank. If the field is blank, the other field is interpreted to contain the actual movement index. If a nonzero value is encountered, the two sector numbers are assumed to be provided. If two or more entries for any movement are encountered, the last entry will be retained and no message is written to notify the user of this potential error in data preparation.

## CATEGORY

### Purpose

The CATEGORY card enters criteria for selecting trips of a desired category from survey data trip records.

### Associated Routines

#### Input

BUILD

### Entry Sequence

Not applicable

### Card Layout (fixed); FORMAT (A4, 4X, A4, 1412, 10A4)

| Columns | Type | Content |
|---|---|---|
| 1 - 8 | Literal | 'CATEGORY' |
| 9 - 12 | Literal | Literal identification (e.g., HBW, HBNW, NHB, TRTX) |
| 13 - 14 | Integer | Category code |
| . . . | . . . | . . . (consecutive fields of two columns) |
| 39 - 40 | Integer | Category code |
| 41 - 80 | Literal | Literal description used in table headings |

### Data Description

The word CATEGORY should appear in the first eight columns of the CATEGORY card. Columns nine through twelve should be left blank or coded with a four-byte literal which will serve as an abbreviation for the category being specified. This four-byte literal will be inserted as the TYPE parameter. The appropriate FORMAT for the GENERATION cards for this cateogry is identified by the parameter TYPE.

Category codes are punched in columns thirteen through forty in two-digit fields. These category codes determine the type of trip matrix which is to be prepared. The category codes are defined in an accompanying table. All codes have two digits; there are no codes less than ten or greater than forty. At least one code between the values of ten and thirty, inclusively, must be specified. This will select some form of vehicle trips. Indicator codes are provided for selecting other combinations.

If the OMIT parameter was set .TRUE. during the execution of SCREEN, only internal trips will be available and indicators 37, 38 and 39 will have no significance. However, if external trips are included, indicators 37, 38 and 39 should be coded as desired. If none of these three codes are entered, indicators 37 and 38 will be entered by default.

Columns 41 through 80 of the CATEGORY card are used as a literal description of the trip matrix being prepared. This will be entered in the parameter record of the trip matrix, and will subsequently appear in printed page headings.

If indicators 31-36 and the associated purposes (codes 10-30) are coded, only passenger trips will be selected. In order to obtain person trips, indicator 40 must be coded, also.

CATEGORY CODES

| Code | | | | | |
|---|---|---|---|---|---|
| (Indicator) | Origin-Destination | Production Attraction | Description | Type Available | Mode |
| | 10 | | All nonhome-based trips | Internal & External | Auto |
| | 11 | 21 | Home-based work trips | Internal & External | Auto |
| | 12 | 22 | Home-based personal business trips | Internal & External | Auto |
| | 13 | 23 | Home-based medical-dental trips | Internal & External | Auto |
| | 14 | 24 | Home-based school trips | Internal & External | Auto |
| | 15 | 25 | Home-based social-recreational trips | Internal & External | Auto |
| | 16 | 26 | Home-based change travel mode trips | Internal & External | Auto |
| | 17 | 27 | Home-based eat meal trips | Internal & External | Auto |
| | 18 | 28 | Home-based shop trips | Internal & External | Auto |
| | 19 | 29 | Home-based serve passenger trips | Internal & External | Auto |
| | 20 | | Truck trips | Internal & External | Truck |
| | 30 | | Taxi trips | Internal only | Taxi |
| 31 | | | Select walk trips (to work only) | Internal only | Person |
| 32 | | | Select auto passenger trips | Internal & External | |
| 33 | | | Select bus passenger trips | Internal only | Passenger |
| 34 | | | Select taxi passenger trips | Internal only | Passenger |
| 35 | | | Select truck passenger trips | Internal & External | Passenger |
| 36 | | | Select school bus passenger trips | Internal only | Passenger |
| 37 | | | Select Internal-Internal trips | | |
| 38 | | | Select Internal-External trips | | |
| 39 | | | Select External-External trips | | |
| 40 | | | Select person trips | | |

## CONTROL

### Purpose

The CONTROL cards specify the routines to be executed and their sequence.

### Associated Routines

NONE

### Entry Sequence

The CONTROL cards must appear first in the input data stream. Only HEADING cards may be intermingled.

### Card Layout (fixed): FORMAT (A4, 4X, 18A4)

| Columns | Type | Contents |
|---|---|---|
| 1 - 7 | Literal | 'CONTROL' |
| 8 | | blank |
| 9-80 | Literal | Routine names separated by commas |

### Data Description

The word CONTROL must appear in the first seven columns of each and every control card. The eighth column should be blank. The remaining columns, nine through eighty, contain the literal names of the routines which are to be executed. The names should be separated by commas. No CONTROL entry name should contain any embedded blanks. However, blanks are fully acceptable between control entry names. The last entry on a control card may be followed by a period to terminate the card scan.

Multiple control cards may be used when the number of routines extend beyond one card. There is no limit to the number of CONTROL cards which may be used, except that only the first 40 entries will be executed. It is not permissible to split a control entry name and put a portion of the name on a following control card. Any invalid control entry will cause immediate program termination.

## EQUALS

### Purpose

The EQUALS cards enter equivalences between centroids and sectors.

### Associated Routines

Input

EQUATE

### Entry Sequence

The EQUALS cards may be in any sequence.

### Card Layout (fixed): FORMAT (I3, 1X, A4, 2X, 14I5)

| Columns | Type | Content |
|---|---|---|
| 1 - 3 | Integer | Sector number |
| 4 | - | blank |
| 5 - 10 | Literal | 'EQUALS' |
| 11 - 15 | Integer | Centroid number |
| ⋮ | ⋮ | ⋮ (consecutive fields of five columns) |
| 76 - 80 | Integer | Centroid number |

### Data Description

A sector number should be punched right justified in columns one to three of every EQUALS card. Its value should range between one and the largest sector number. Column four should be blank. Columns five through ten should contain the word EQUALS. Fourteen five-digit fields constitute the remainder of each EQUALS card. These should contain valid centroid numbers right justified in each field.

EQUALS cards may be in any order. Within any EQUALS card, centroid numbers may appear in any order. Any of the 14 centroid number fields may be left blank. It is advantageous not to skip any sector numbers when preparing EQUALS cards.

## FORMAT

### Purpose

The FORMAT card enters the format by which GENERATION, LENGTH, or BIAS cards are to be read.

### Associated Routines

| Input | Output |
|---|---|
| ACCEPT | REFINE |
| | MODEL |

### Entry Sequence

The FORMAT card precedes the first GENERATION, LENGTH, or BIAS cards to which it pertains.

### Card Layout (fixed): FORMAT (A4, 2X, 17A4, A2, A4)

| Columns | Type | Content |
|---|---|---|
| 1 - 6 | Literal | 'FORMAT' |
| 7 - 76 | - | Variable format enclosed in parentheses |
| 77 - 80 | Literal | Literal identification |

### Data Description

The word FORMAT should appear in the first six columns of any FORMAT card. The remainder of the card may be used for format coding. The format should be enclosed in parentheses. Columns 77 through 80 may be used for a four-byte literal identification which will be compared with the parameter type.

It might be pointed out that the T-format code may prove very useful in coding variable formats when the data is not in the same sequence as required by the read statement. If questions arise regarding proper format coding, the appropriate FORTRAN reference manual should be consulted.

# GENERATION

## Purpose

The GENERATION cards enter production and attraction (or origin and destination) volumes for each centroid.

## Associated Routines

### Input

ACCEPT

## Entry Sequence

The GENERATION cards immediately follow a FORMAT card specifying their format. The GENERATION cards may be in any order.

## Card Layout (variable): sample (FORMAT (A4, 6X, 15, T26, 2F5.0)

| Field | Type | Content |
|---|---|---|
| 1 | Literal | 'GENERATION' or 'FORECAST' |
| 2 | Integer | Centroid number |
| 3 | Real | Production (or origin) volume |
| 4 | Real | Attraction (or destination) volume |

## Data Description

GENERATION cards are read in a variable format depending upon what is supplied by the preceding FORMAT card. Four items are read from each card. The first item is the word generation or forecast. The second item is the centroid number which should appear as an integer. The third item is the production or origin volume and the fourth item is the attraction or destination volume. These two volumes are read as real variables.

# HEADING

## Purpose

The HEADING card enters a literal message which is used for identification of printed output.

## Associated Routines

NONE

## Entry Sequence

Any encounter of a HEADING card in the data card input stream, between routine executions, results in a heading change.

## Card Layout (fixed); FORMAT (A4, 4X, 18A4)

| Columns | Type | Contents |
|---|---|---|
| 1 - 7 | Literal | 'HEADING' |
| 8 | - | blank |
| 9 - 80 | Literal | Literal heading |

## Data Description

The word HEADING must appear in the first seven columns of every HEADING card. The eighth column should be blank. The remainder of the card, columns nine through eighty, may contain any literal information desired to be used as a heading.

## INTERACTION

### Purpose

The INTERACTION cards enter the points which describe a production interaction curve.

### Associated Routines

Input

ACCEPT

REFINE

### Entry Sequence

The INTERACTION cards immediately follow a FORMAT card specifying their format. These cards should follow the GENERATION cards, LENGTH cards, and BIAS cards. The INTERACTION cards should be in ascending order on production volumes.

### Card Layout (variable): sample (FORMAT (A4, 7X, 2F8.0))

| Field | Type | Content |
|---|---|---|
| 1 | Lateral | 'INTE' (the 'RACTION' is ignored) |
| 2 | Real | production volume |
| 3 | Real | the maximum percent of internal zones with which a production zone of this size would be expected to interact. (For example, 50% would be entered as 50.0) |

### Data Description

The INTERACTION cards are read in a variable format as presented by the preceding FORMAT card. The first field of the card should be a four-column field containing the letters "INTE" (i.e. the first four letters of the word INTERACTION). The second field should contain a production volume. The third field should contain the maximum percent of the internal zones with which a zone having the production volume specified in the second field would be expected to interact.

In essence, the INTERACTION cards are used to describe the production-interaction curve which will be used by the MODEL routine to constrain the number of interactions. Each card represents a point on the production-interaction curve. Straight line interpolation will be used to determine any needed points between two points specified by INTERACTION cards. If points are needed beyond the last point specified by INTERACTION cards, straight line extrapolation will be used to determine the points based on the last two points specified by INTERACTION cards. If the production-interaction curve specifies a percent of the internal zones which is larger than the percent of internal zones with a non-zero attraction volume, then the percent specified by the curve will be ignored and the percent of internal zones with a non-zero attraction volume will be used instead.

## LENGTH

### Purpose

The LENGTH cards enter relative trip volumes corresponding to each value of trip length.

### Associated Routines

| Input | Output |
|---|---|
| ACCEPT | REFINE |

### Entry Sequence

The LENGTH cards immediately follow the FORMAT card which describes their FORMAT, if a FORMAT card is included. The LENGTH cards may be in any order.

### Card Layout (variable): sample FORMAT(A4, 6X, I5, 5X, F10.3, 10X, A4)

| Field | Type | Content |
|---|---|---|
| 1 | Literal | 'LENGTH' |
| 2 | Integer | Separation (length) value |
| 3 | Real | Trip frequency volume |
| 4 | Literal | blank, 'INT-EXT', 'EXT-INT',<br>or 'THRU' |

### Data Description

The LENGTH cards are read in a variable format as presented by the preceding FORMAT card. It should be noted that if no FORMAT card is presented after the GENERATION cards and before the LENGTH cards, the format used for the GENERATION cards will be assumed. This will cause errors since the data is incompatible. The word length should appear as the first item on each LENGTH card. The next item should be a separation value. The following item should be the frequency volume. The last item should be a literal field which is normally left blank. However, if the separation values used for trips with external terminals are not known, the expressions INT-EXT, EXT-INT, or THRU may be supplied in the fourth field to identify these movements and the first field left blank. If literal codes are used then any numerical separation value contained in the first field is ignored.

# SEPARATION

## Purpose

The SEPARATION cards enter replacement values for separations between centroid pairs.

## Associated Routines

### Input

EDIT

## Entry Sequence

All SEPARATION cards must be in numerically increasing sequence with regard to production (or origin) zone numbers. Attraction (or destination) zone numbers, corresponding to each production zone, may be in any order.

## Card Layout (fixed): FORMAT (A4, 6X, 415)

| Columns | Type | Contents |
|---|---|---|
| 1 - 10 | Literal | 'SEPARATION' |
| 11 - 15 | Integer | Production (or origin) centroid number |
| 16 - 20 | Integer | Attraction (or destination centroid number |
| 21 - 25 | Integer | Separation value |
| 26 - 30 | Integer | Special code |

## Data Description

The word SEPARATION should appear in columns one through ten on each SEPARATION card. Columns eleven through fifteen, and sixteen through twenty represent the fields for the production and attraction centroid numbers, respectively. These numbers must be right justified in these fields, and must range between one and the numerical value of the last centroid. Columns 21 through 25 define the field for the replacement separation value. This value must be an integer, right justified in the field, and range between one and the largest internal separation value.

Columns 26 through 30 is an optional field for special separation codes. This value must be an integer, right justified in the field, and range between one and the value of the parameter EXTEND plus one. The replacement separation value for the zone pair will be computed as follows:

$$\text{replacement separation value} = \text{largest internal separation} + \text{special separation code}$$

The user must be careful when using special separation codes for interzonal separations since the selection of eligible zone pairs for the interaction constraint in the MODEL routine is based on the accessibility measure:

$$\frac{\text{Attraction Volume}}{\text{Separation}}$$

It is possible, therefore, that few, if any, of the interzonal movements with a special separation code would be selected as eligible zone pairs. To avoid this problem will require that the interzonal movements with a special separation code be imposed via ADMIT cards in the IMPOSE routine. Intrazonal movements do not pose a problem since they are selected as eligible zone pairs so long as they have non-zero production and attraction volumes regardless of their separation.

The EDIT routine will check the SEPARATION cards for the following conditions:

- A special separation code which is greater than the value of the parameter EXTEND plus one

- A separation value which is greater than the largest internal separation (including the separation values computed for the special separation codes)
- A SEPARATION card with both a separation value and a special separation code
- An invalid centroid number (i.e., a centroid number which is greater than the value of the parameter N)

If either of these conditions exist then a warning message will print and the JOB will be abnormally terminated following the EDIT routine with a stop code of 16. When either of these conditions are encountered, the following values will be entered in the separation matrix built by EDIT:

- If a special separation code is encountered which is greater than the value of EXTEND plus one, then the SEPARATION card is ignored
- If a separation value is encountered which is greater than the value of the largest internal separation (including the separation values computed for the special separation codes), then the SEPARATION card is ignored
- If a SEPARATION card is encountered with both a separation value and a special separation code, then the special separation code is ignored and the separation value (if valid) is used.
- If an invalid centroid number is encountered then the SEPARATION card is ignored.

## &VALUES

### Purpose

The &VALUES card enters changes in parameter values.

### Associated Routines

| Input | Output |
|---|---|
| MODIFY | REFINE |

### Entry Sequence

Random

### Card Layout (fixed): NAMELIST Control

| Columns | Type | Content |
|---|---|---|
| 1 | - | blank |
| 2 - 8 | Literal | '&VALUES' |
| 9 | | blank |
| 10 - 80 | - | Parameter names and corresponding values |
| 9 - 80 (unconfined) | Literal | '&END' |

### Data Description

The first column in the &VALUES card must be blank. The second column must contain an & and the third through the eighth column the word VALUES. The tenth through seventy-fifth columns may be used to input parameter names and corresponding values. An equals sign should separate a parameter name from its associated value. Commas should be used to distinguish entries, and blanks may appear anywhere but within parameter names. Following the last entry on the card should be an & followed immediately by the work END. The &END must appear somewhere on the card. Either a comma or a blank may separate the &END from the last entry. Entires cannot be continued on a following card. If a question should arise in coding this card, it is recommended that the appropriate FORTRAN manual be consulted with regard to NAMELIST usage. If the &VALUES cars is coded improperly, the Texas Trip Distribution Package will terminate with a STOP code of 12.

## DATA SET REFERENCES

Twenty-two data sets are associated with the Texas Trip Distribution Package. It is doubtful that all of these data sets would ever be used in any given application of the package. The data sets needed are determined by the routines to be executed and the options (if any) selected by the user under the various routines. The data set requirements for each routine are listed under the heading "Data Set References" in the descriptions of each individual routine.

The data sets associated can be grouped into four classes as follows:

- Matrix--those data sets which contain either a trip matrix or a separation matrix
- Sort--those data sets used by the sort routine
- Assignment--those data sets created by or to be used as input to either the Texas Large Network Package or the Texas Small Network Package.
- Trip Report--the data set containing the original trip reports used as input to the package (record size=104 bytes)
- Save--those data sets used to save the arrays and parameters needed to restart the distribution process in order to perform additional iterations

The data sets are listed in the following tables along with their default unit numbers, classifications and brief descriptions of their contents.

## DATA SET DESCRIPTIONS, CLASSIFICATIONS,

## AND DEFAULT UNIT NUMBERS

| Data Set Name | Default Unit Number | Class | Contents |
|---|---|---|---|
| ADD1 | 3 | matrix | First trip matrix to be summed by the SUM routine |
| ADD2 | 17 | matrix | Second trip matrix to be summed by the SUM routine |
| ADD3 | 18 | matrix | Third trip matrix to be summed by the SUM routine |
| ADD4 | 19 | matrix | Fourth trip matrix to be summed by the SUM routine |
| ADD5 | 23 | matrix | Fifth trip matrix to be summed by the SUM routine. |
| ALTTRP | 22 | matrix | The altered trip matrix outputed from the ALTER routine. |
| ASSIGN | 13 | assignment | The model trip matrix packed in the format for input to the assignment packages. |
| IMPSEP | 15 | matrix | The separation matrix from the IMPOSE routine |
| INTRIP | 21 | assignment | The packed trip matrix outputed from the assignment packages. |
| MODTRP | 3 | matrix | The trip matrix constructed by Model or Expand |
| NEGSEP | 1 | matrix | The separation matrix from the SET routine. |
| NEWSEP | 16 | matrix | The future separation matrix used as input to ALTER |
| NOWSEP | 4 | matrix | The edited separation matrix from the EDIT routine. |

## DATA SET DESCRIPTIONS, CLASSIFICATIONS, AND DEFAULT UNIT NUMBERS (continued)

| Data Set Name | Default Unit Number | Class | Contents |
|---|---|---|---|
| NOWTRP | 2 | matrix | The trip matrix outputed from BUILD |
| RAWSEP | 8 | assignment | The interzonal separations from the assignment package (used as input to EDIT) |
| RECORD | 14 | sort | The abbreviated trip records (used in conjunction with the SCREEN and BUILD routines) |
| REPORT | 12 | trip report | The survey trip reports in 104 byte records. |
| SORTIN | 9 | sort | Records to be sorted by system sort. |
| SUMTRP | 20 | matrix | The trip matrix constructed by the SUM routine. |
| SWTTRP | 24 | matrix | The trip matrix constructed by the SWITCH routine. |
| PLOTTAPE | -- | -- | The calcomp plots for input to the calcomp plotter. |
| SORTOUT | 10 | sort | The sorted records outputed from the system sort. |
| SV | 25 | save | Arrays and parameters needed to restart the distribution process to perform additional iterations. |
| RS | 26 | save | The same as data set SV. |

Input Data Set Formats

There are basically three externally generated data sets which may be used as input to the package. The first is the separation matrix which is produced by either the Texas Large Network Package or the Texas Small Network Package. This data set is the RAWSEP data set used as input to the EDIT routine in this package. The second externally generated data set which, if available, may be used as input to the package is the data set containing the trip reports. This is the REPORT data set used as input to the SCREEN routine. The third is a trip matrix which is produced by either the Texas Large Network Package or the Texas Small Network Package. This data set is the INTRP data set used as input to the UNPACK routine.

There are, of course, numerous intermediate data sets produced by the package which may be saved for later input to the package on other runs. The preservation of such data sets can, in some instances avoid an unnecessary rerun of the routine by which they were created. Since these data sets are produced by the package and are subsequently used by the package, the details of the exact formats of these data sets will not be needed under normal operating procedures. It should be noted, however, that the formats for these data sets are specified in detail in the program documentation manual for the package.

Due to the compatibility maintained between this package and both the Texas Large Network Package and the Texas Small Network Package, the user need not be concerned with the formats for either the separation matrix (i.e.the RAWSEP data set) or the trip matrix (i.e. the INTRP data set) produced by these packages. The format of the REPORT data set containing the trip reports may, however, be of concern to the user since it is produced externally. The format for this data set is, of course, compatible with the format currently being used by the Texas Highway Department in their urban transportation studies.

The REPORT data set consists of 104 byte records and up to six types of records. The six types of 104 byte records are:

- Type 1 records contain information regarding the dwelling unit in which the home interview was conducted and are identified by an integer 1 in the first byte of the record.
- Type 2 records contain information regarding each trip reported in the home interviews and are identified by an integer 2 in the first byte of the record.
- Type 3 records contain the external trip report information obtained at external stations and are identified by an integer 3 in the first byte of the record.
- Type 4 records contain truck trip information and are identified by an integer 4 in the first byte of the record.
- Type 5 records contain taxi trip information and are identified by an integer 5 in the first byte of the record.
- Type 6 records contain employment information and are identified by an integer 6 in the first byte ɔf the record.

Type 1 and type 6 records are ignored by the trip distribution package. Only selected information is used from the remaining records. The formats for these records (i.e. types 2-5) are contained on the following pages. These formats only specify the contents of fields which are used by the trip distribution package; all other fields are disregarded.

## FORMAT FOR THE REPORT DATA SET

## (TYPE 2 RECORDS)

| Byte | Content |
|---|---|
| 1 | Report type (Integer 2) |
| 2-26 | Disregarded |
| 27-28 | Origin location code (blank if inside study area) |
| 29-40 | Disregarded |
| 41-42 | Destination location code (blank if inside study area) |
| 43-54 | Disregarded |
| 55 | Travel mode |
| 56-59 | Disregarded |
| 60 | Origin primary trip purpose |
| 61 | Destination primary trip purpose |
| 62 | Origin secondary trip purpose |
| 63 | Destination secondary trip purpose |
| 64-77 | Disregarded |
| 78-80 | Trip volume (decimal point assumed between bytes 79 and 80) |
| 81-94 | Disregarded |
| 95-99 | Origin centroid |
| 100-104 | Destination centroid |

## FORMAT FOR THE REPORT DATA SET

## (TYPE 3 RECORDS)

| Byte | Content |
|---|---|
| 1 | Report type (Integer 3) |
| 2-20 | Disregarded |
| 21 | Vehicle occupancy |
| 22-26 | Disregarded |
| 27-28 | Origin location code (blank if inside study area) |
| 29-40 | Disregarded |
| 41-42 | Destination location code (blank if inside study area) |
| 43-54 | Disregarded |
| 55 | Travel mode |
| 56-64 | Disregarded |
| 65 | External origin purpose |
| 66 | External destination purpose |
| 67-77 | Disregarded |
| 78-80 | Trip volume (decimal point assumed between bytes 79 and 80) |
| 81-94 | Disregarded |
| 95-99 | Origin centroid |
| 100-104 | Destination centroid |

## FORMAT FOR THE REPORT DATA SET

## (TYPE 4 RECORDS)

| Byte | Content |
|---|---|
| 1 | Report type (Integer 4) |
| 2-26 | Disregarded |
| 27-28 | Origin location code (blank if inside study area) |
| 29-40 | Disregarded |
| 41-42 | Destination location code (blank if inside study area) |
| 43-54 | Disregarded |
| 55 | Travel mode |
| 56-77 | Disregarded |
| 78-80 | Trip volume (decimal point assumed between bytes 79 and 80) |
| 81-94 | Disregarded |
| 95-99 | Origin centroid |
| 100-104 | Destination centroid |

## FORMAT FOR THE REPORT DATA SET

## (TYPE 5 RECORDS)

| Byte | Content |
|---|---|
| 1 | Report type (Integer 5) |
| 2-26 | Disregarded |
| 27-28 | Origin location code (blank if inside study area) |
| 29-40 | Disregarded |
| 41-42 | Destination location code (blank if inside study area) |
| 43-54 | Disregarded |
| 55 | Travel mode |
| 56-77 | Disregarded |
| 78-80 | Trip volume (decimal point assumed between bytes 79 and 80) |
| 81-94 | Disregarded |
| 95-99 | Origin centroid |
| 100-104 | Destination centroid |

Output Data Set Formats

There is only one data set produced by the package which is intended to be used as input to other programs: the ASSIGN data set. This data set will contain a trip matrix in a format compatible with the Large and Small Network Packages.

Data Set Specifications

In order to operate the Texas Trip Distribution Package, it will be necessary to provide appropriate specifications for each data set involved with each routine being executed. Sample specifications are provided in the following table. All parameters refer to Job Control Language (JCL) Data Definitions (DD). The appropriate IBM manual should be consulted for further description of the requirements.

| Data Set Class | RECFM | LRECL | BLKSIZE | TRTCH |
|---|---|---|---|---|
| Matrix | VBS | 6000 | 6004 | C |
| Sort | VBS | 24 | 6004 | C |
| Assignment | VBS | 416 | 792 | C |
| Trip Report | FB | 104 | 1040 | ET |
| Save | VBS | 1004 | 1008 | C |

# CROSS REFERENCE TABLE

The following table is designed to provide the user a convenient summary of the data sets and data cards associated with each routine.

| Cross Reference | DATA SETS | | | | | | | | | | | | | | | | | | | | | | | | | Pointers | | Data Cards | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | INTRIP | REPORT | RECORD | NOWTRP | RAWSEP | NOWSEP | NEGSEP | IMPSEP | MODTRP | ADD1 | ADD2 | ADD3 | ADD4 | ADD5 | SUMTRP | SWTTRP | NEWSEP | ALTTRP | ASSIGN | PLOTTAPE | SORTIN | SORTOUT | SV | RS | | MT | MS | | | |
| Default Unit | [21] | [12] | [14] | [2] | [8] | [4] | [1] | [15] | [3] | [3] | [17] | [18] | [19] | [23] | [20] | [24] | [16] | [22] | [13] | | 9 | 10 | [25] | [26] | | 2 | 4 | 5 | 7 | |
| MODIFY | | | | | | | | | | | | | | | | | | | | | | | | | | | | I | | &VALUES |
| PERUSE | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| UNPACK | I | | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| SCREEN | | I | | S | | | | | | | | | | | | | | | | | S | 0 | | | | | | | | |
| BUILD | | | 0/I | 0 | | | | | | | | | | | | | | | | | | I | | | | NOWTRP | | I | | CATEGORY |
| PRINT | | | | | | | | | | | | | | | | | | | | | | | | | | I | | | | |
| EDIT | | | | | I | 0 | | | | | | | | | | | | | | | | | | | | | NOWSEP | I | | SEPARATION |
| WRITE | | | | | | | | | | | | | | | | | | | | | | | | | | | I | | | |
| EQUATE | | | | | | | | | | | | | | | | | | | | | | | | | | | | I | | EQUALS |
| SET | | | | | | | 0 | | | | | | | | | | | | | | | | | | | I | I/NEGSEP | I | | EQUALS |
| LIST | | | | | | | | | | | | | | | | | | | | | | | | | | I | I | | | |
| REFINE | | | | | | | | | | | | | | | | | | | | 0 | | | | | | | | | 0 | &VALUES<br>FORMAT LENGTH |
| ACCEPT | | | | | | | | | | | | | | | | | | | | | | | | | | | | I | | FORMAT GENERATION<br>LENGTH BIAS |
| IMPOSE | | | | | | | | 0 | | | | | | | | | | | | | | | | | | | I/IMPSEP | I | | ADMIT |
| MODEL | | | | | | | | | 0 | | | | | | | | | | | | | | 0 | | | MODTRP | I/NOWSEP | | 0 | FORMAT BIAS |
| EXPAND | | | | I | | | | | 0 | | | | | | | | | | | | | | | | | MODTRP | | | | |
| RANDOM | | | | | | | | | 0 | | | | | | S | | | | | | | | | | | | | | | |
| GET | | | | | | | | | | | | | | | | | | | | 0 | | | | | | I | I | I | | EQUALS |
| MATCH | | | | I | | I | | | I | | | | | | | 0 | | | | | S | S | | | | | | | | |
| SUM | | | | | | | | | | I | I | I | I | I | 0 | | | | | | | | | | | SUMTRP | | | | |
| SWITCH | | | | | | | | | | | | | | | | 0 | | | | | S | S | | | | I/SWTTRP | | | | |
| ALTER | | | | | | I | | | I | | | | | | | | I | 0 | | | | | | | | ALTTRP | NEWSEP | | | |
| PACK | | | | | | | | | | | | | | | | | | | 0 | | | | | | | I | | | | |
| RESTART | | | | | | | | | 0 | | | | | | | | | | | | | | 0 | I | | MODTRP | I/NOWSEP | I | | |

I = Input, 0 = Output, S = Scratch

# COMPUTATIONAL REQUIREMENTS

PACKAGE CAPACITY

PROCESSING TIMES

PROGRAM MODULES

SAMPLE DECK SET-UPS AND JCL

# PACKAGE CAPACITY

The INTEGER statement in the FORTRAN program, MAIN controls the capacity of the Texas Trip Distribution Package. Variable A in the INTEGER statement, defines the storage for the 11 vectors related to the centroids; the first number in the dimension for variable A controls the maximum number of centroids. Variable B defines the storage for the 8 vectors related to travel separation; the first number in the dimension for variable B controls the largest separation that can be handled (which includes provisions for 4 external movements). Variable C defines the storage for the 5 vectors related to sector interchanges. The first number in the dimension for variable C controls the largest number of sector movements that can be processed. The largest number of sector movements controls, in turn, the maximum number of sectors (and the largest sector designation number) which are permitted. The number of sector combinations may be computed as follows:

sector combinations = (last sector)*(last sector +1)/2

The capacity of the package may, therefore, be changed by simply redimensioning the variables A, B and/or C in the INTEGER statement in the program MAIN.

An exception to this is the MATCH routine. When the MATCH routine is executed, the variable C must be dimensioned for a minimum of 320 sector combinations (corresponding to about 25 sectors) regardless of how many are actually used. If this condition is not met, the Trip Distribution Package will terminate with a STOP code of 9.

At the beginning of execution, the following message is always printed:

```
EFFECTIVE CAPACITIES
----------------------

  700 CENTROIDS

  100 SEPARATIONS (INCLUDING EXTERNAL CODES)

   25 SECTORS (325 COMBINATIONS)
```

The numbers, of course, may vary depending upon the array dimensions being used.

It should be noted that during processing continuous checks are made to insure that effective capacities are not exceeded. If a routine detects that some capacity has been exceeded, the Trip Distribution Package will terminate with a STOP code of 7, and an appropriate message will be printed to explain the situation.

The core storage requirements under various sets of package capacities are summarized in the following table. This table should serve as a guideline for determining the region size needed for any desired set of capacities.

## CORE STORAGE REQUIREMENTS UNDER VARIOUS SETS OF PACKAGE CAPACITIES

| | | | | | |
|---|---|---|---|---|---|
| Capacities | | | | | |
| Centroids | 400 | 700 | 1600 | 3300 | 4800 |
| Separations | 50 | 100 | 150 | 200 | 250 |
| Sector Combinations | 325 | 325 | 325 | 325 | 325 |
| Core Storage Needed for Arrays | | | | | |
| 11 Centroid Vectors | 4400 | 7700 | 17600 | 36300 | 51800 |
| 8 Separation Vectors | 400 | 800 | 1200 | 1600 | 2000 |
| 5 Sector Combination Vectors | 1625 | 1625 | 1625 | 1625 | 1625 |
| Total Words | 6425 | 10125 | 20425 | 39525 | 55425 |
| Total Bytes | 25,700 | 40,500 | 81,700 | 158,100 | 221,700 |
| Array Storage Required | 25k | 40k | 80k | 155k | 216k |
| Buffer Requirements Per Data Set* | | | | | |
| Matrix | 3.2k** | 14.5k | 14.5k | 14.5k | 14.5k |
| Sort | 14.5k | 14.5k | 14.5k | 14.5k | 14.5k |
| Assignment | 1.5k | 1.5k | 1.5k | 1.5k | 1.5k |
| Trip Report | 2k | 2k | 2k | 2k | 2k |
| Other Core Storage Requirements | | | | | |
| Program Storage (with overlay feature) | 51k | 51k | 51k | 51k | 51k |
| System Storage | 8k | 8k | 8k | 8k | 8k |
| Sort Allocation (Amount = 110,000) | 110k | 110k | 110k | 110k | 110k |

*Assuming the specifications given under Data Control Block Suggestions
**Assumes LRECL = 1608 and BLKSIFE - 1612

# PROCESSING TIMES

Processing times are difficult to estimate. Under MVT, the times printed from the core clock will have significance only if the Texas Trip Distribution Package is the only job being processed by the IBM 360 computer.

Execution times can be controlled to a degree by the user. Several of the routines are heavily input/output bound. IF a large blocking factor is used with the associated date sets, this will improve execution time at a cost to the amount of core storage used. Increasing the AMOUNT of storage allocated to the sort for work space will decrease sort time.

The following table is provided to aid the user in estimating the sort time. It should be recognized that the times are appropriate, and that recent experience has demonstrated considerable variation from the estimates provided.

More zones require more processing time. No relationship has been found to estimate the amount of processing time as a function of the number of zones.

## SYSTEM/360 MODEL 50 SORT ESTIMATES

| Number of Records | Sort Time in Minutes | | |
|---|---|---|---|
| (In thousands) | 44K | 110K | 110K |
| 2 | 1.1 | 1.1 | 1.0 |
| 5 | 1.2 | 1.2 | 1.1 |
| 10 | 1.3 | 1.3 | 1.2 |
| 20 | 2.0 | 1.6 | 1.4 |
| 25 | 2.2 | 1.7 | .15 |
| 50 | 3.3 | 2.5 | 2.0 |
| 75 | 4.5 | 3.2 | 2.5 |
| 100 | 5.6 | 5.7 | 4.6 |
| 125 | 9.1 | 6.9 | 5.5 |
| 150 | 11 | 8.0 | 6.4 |
| 200 | 14 | 11 | 8.1 |
| 300 | 20 | 14 | 11 |

These estimates are taken from C28-6543-3 IBM Sort/MERGE MANUAL from Feb. 1967.

## PROGRAM MODULES

All of the routines prepared especially for use in the Texas Trip Distribution Package, except INVOKE, have been written in the FORTRAN IV programming language. INVOKE has been written in assembly language and functions as an entry into the system sorting package. REREAD is a library routine written in assembly language. It is highly installation dependent and therefore may require alteration to be compatible with many computer systems, if it is not available locally.

The subroutines IMPOSE and SCAN must be compiled under FORTRAN G. All other FORTRAN programs in the package may be compiled under FORTRAN H.

## SAMPLE DECK SET-UPS AND JCL

The flexibility of the Texas Trip Distribution Package makes it most difficult to describe every potential application. The variations between the urban transportation studies and the flexibilities in the operation of the package, precludes mapping every feasible alternative in detail. The examples presented in this section are not intended to be used as rigid guidelines nor to limit the use of the package to the types of applications illustrated. The intention is simply to provide the potential user with some insight into how the package might be applied to a few selected situations.

In each of the following examples, a brief description of the data availalbe and the objectives to be achieved by the application of the package is provided. A flow chart is presented which describes the execution sequence which might be used to achieve the desired objectives. A diagram is then provided for each example which illustrates the deck set-up for the run. Also provided with each example is a sample set of JCL which might be used for the run. The sample JCL assumes that the Texas Trip Distribution Package is in load module form on the private disk pack named TTIPLN.

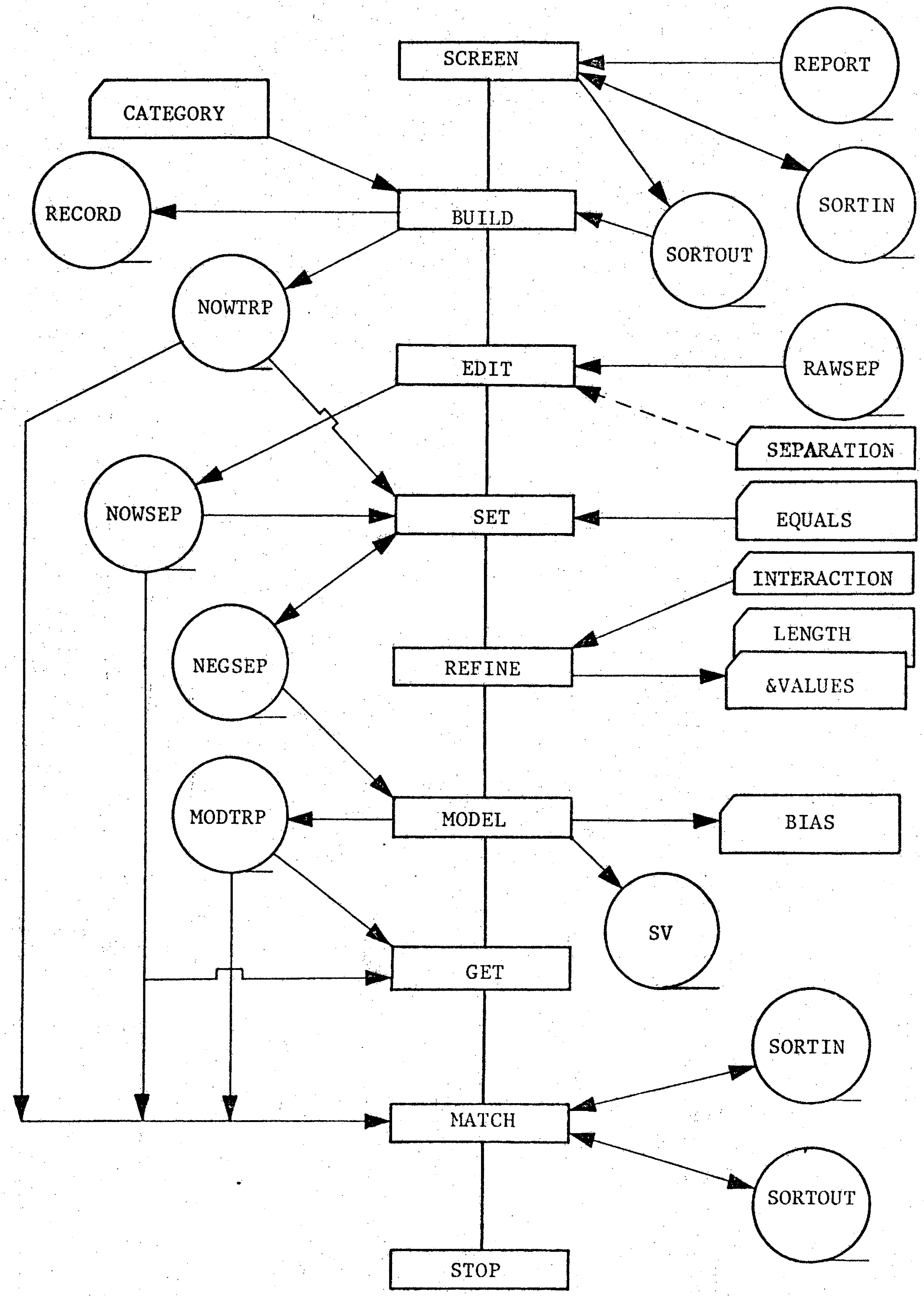
## EXAMPLE I: DISTRIBUTION OF EXISTING TRIPS

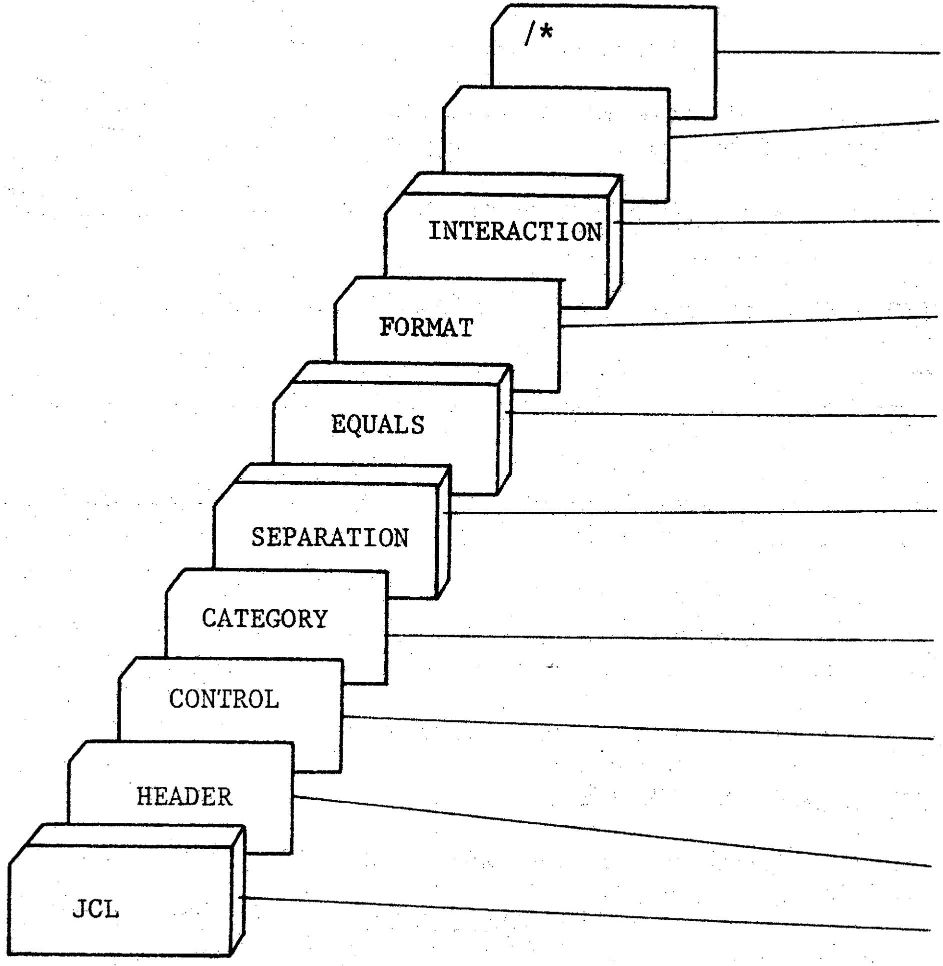
Data Available:

- A separation matrix prepared by either the Texas Small Network Package or the Texas Large Network Package (the RAWSEP data set)
- Trip reports obtained from an origin-destination survey (the REPORT data set)
- Sector equivalences (EQUALS cards)
- INTERACTION cards (note, the relative production - interaction model will be used if these cards are not provided)

Objectives:

- To obtain a trip matrix of existing internal home based work auto driver trips using survey data. (This requires a CATEGORY card specifying the type of trips desired. The NOWTRP data set will contain the desired survey trip matrix.)
- To distribute existing internal home based work auto driver trips using a constrained interactance model with interactions detected from survey data imposed. (The MODTRP data set will contain the desired modeled trip matrix.)
- To obtain a summary of the sector-to-sector movements for analysis
- To obtain a comparison of the survey trip matrix and the modeled trip matrix


SCREEN
REPORT
CATEGORY
SORTIN
RECORD
BUILD
SORTOUT
NOWTRP
EDIT
RAWSEP
SEPARATION
NOWSEP
SET
EQUALS
INTERACTION
NEGSEP
REFINE
LENGTH
&VALUES
MODTRP
MODEL
BIAS
SV
GET
SORTIN
MATCH
SORTOUT
STOP



JCL card (/* card)

Blank card - needed at end of data deck due to the computer programs use of the REREAD option

Describes the production-interaction curve

Specifies the format of the INTERACTION cards

Optional - may be used to define a sector structure

Optional - may be used to enter replacement values for separations between centroid pairs

Specifies the criteria for selecting trips of a desired category from survey data trip records

Specifies the routines to be executed and their order of execution (in this case; SCREEN, BUILD, EDIT, SET, REFINE, MODEL, GET, MATCH, STOP)

Optional - used to identify the run

JCL cards

SAMPLE DECK SET-UP FOR EXAMPLE I

## SAMPLE JCL FOR EXAMPLE I

```
//JOBLIB   DD   UNIT=PRIVATE,VOL=SER=TTIPLN,DISP=SHR,
//   DSN=TTIPLN.THDMODEL
//GO   EXEC   PGM=MODEL,REGION=228K
//FT12F001   DD   UNIT=TAPE9,VOL=SER=PSD112,DISP=OLD,
//   DCB=(RECFM=FB,LRECL=104,BLKSIZE=1040),DSN=SURVEY.DATA
//FT14F001   DD   UNIT=TAPE9,VOL=SER=PSD116,
//   DCB=(RECFM=VBS,LRECL=24,BLKSIZE=6004)
//FT02F001   DD   UNIT=TAPE9,VOL=SER=PSD425,
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004),
//   DSN=HBW.SURVEY.TRIPS
//SORTLIB   DD   DSN=SYS1.SORTLIB,DISP=SHR
//FT09F001   DD   UNIT=SYSDA,DSN=&INSORT,SPACE=(6004,(1000,100)),
//   DCB=(RECFM=VBS,LRECL=24,BLKSIZE=6004)
//SORTIN   DD   DSN=&INSORT,VOL=*.FT09F001,DISP=(OLD,PASS),
//   DCB=(RECFM=VB,LRECL=24,BLKSIZE=6004),SPACE=(6004,(1000,100))
//SORTWK01   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTIN
//SORTWK02   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTWK01
//SORTWK03   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTWK02
//SORTOUT   DD   DSN=&OUTSORT,UNIT=SYSDA,SEP=SORTWK03,
//   DCB=(RECFM=VB,LRECL=24,BLKSIZE=6004),SPACE=(6004,(1000,100))
//FT10F001   DD   DSN=&OUTSORT,VOL=*.SORTOUT,DISP=(OLD,PASS),
//   DCB=RECFM=VBS
//FT08F001   DD   UNIT=TAPE9,VOL=SER=PSD359,DSN=SKIMTREE,DISP=OLD
//FT04F001   DD   UNIT=TAPE9,VOL=SER=PSD008,DSN=UPSKMTRE,
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT01F001   DD   UNIT=SYSDA,SPACE=(6004,(200,100)),
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT03F001   DD   UNIT=TAPE9,VOL=SER=PSD449,DSN=HBWTRIPS,
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT25F001   DD   UNIT=TAPE9,VOL=SER=ZZ0001,DSN=SV,
//   DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004)
//FT07F001   DD   SYSOUT=B
//FT06F001   DD   SYSOUT=A
//FT05F001   DD   *
```

## EXAMPLE II: DISTRIBUTION OF FUTURE TRIPS*

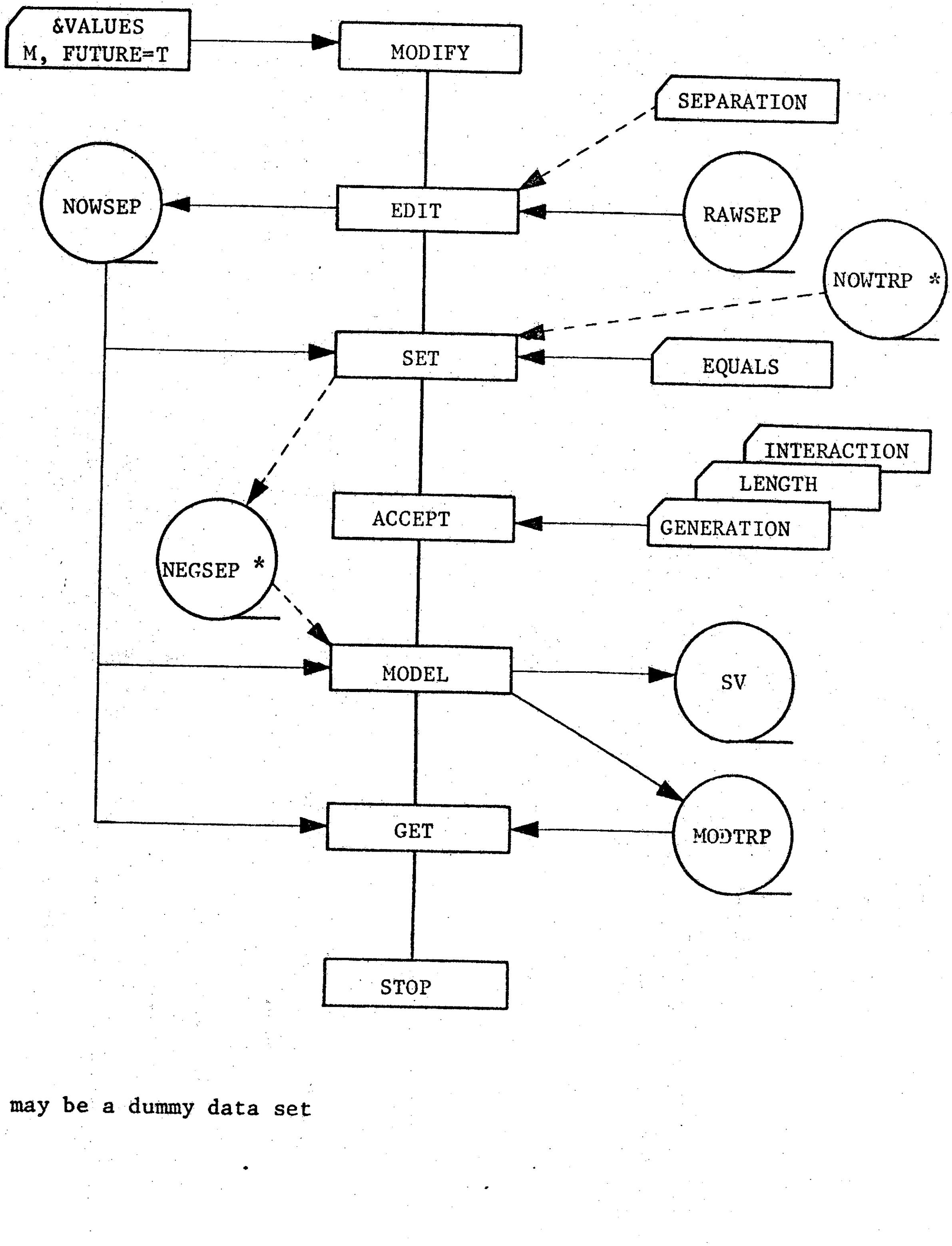
Data Available:

- A separation matrix prepared by either the Texas Large Network Package or th Texas Small Network Package (the RAWSEP data set)
- Modeled future productions and attractions (GENERATION cards)
- Modeled future trip length frequency (LENGTH cards)
- INTERACTION cards
- Largest internal zone number (parameter M)
- Sector equivalances (EQUALS cards)

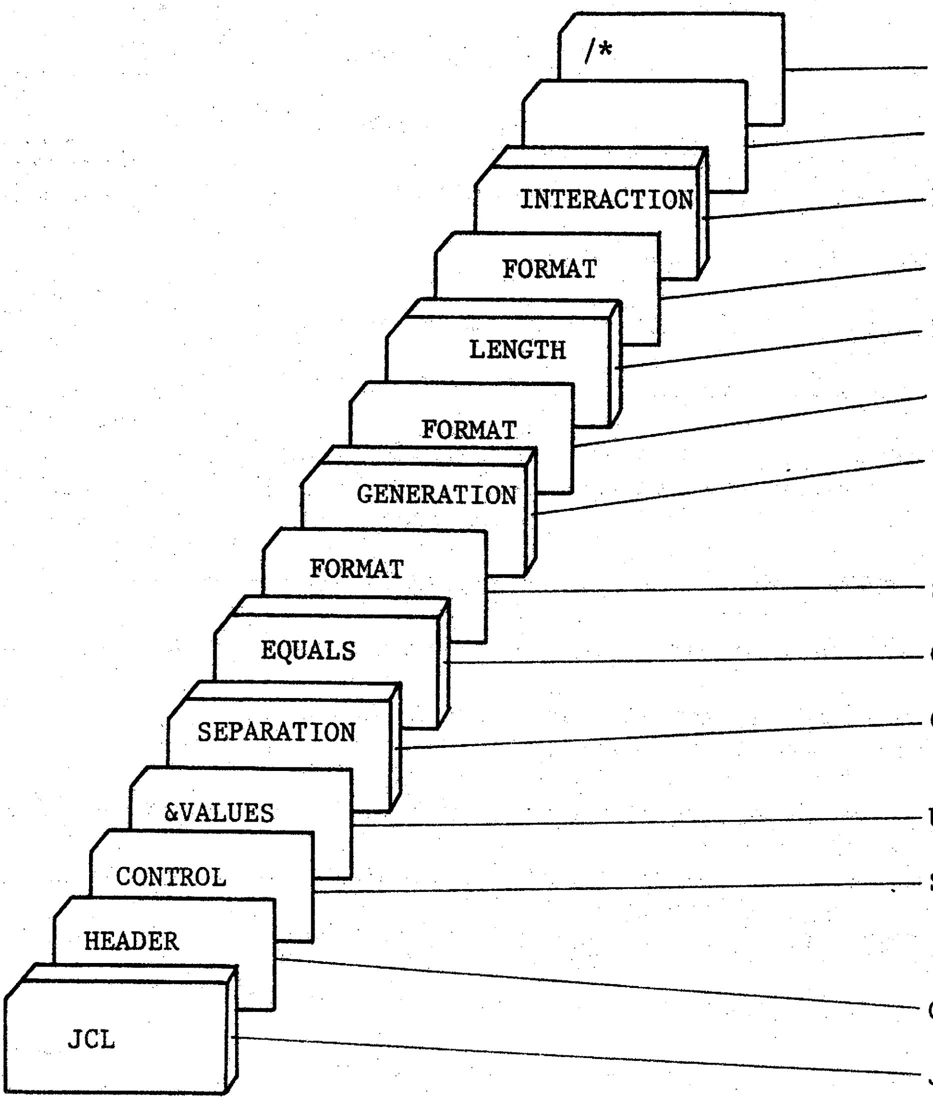
Objectives:

- To distribute future trips using a constrained interactance model
- To obtain sector-to-sector movements for analysis

* It should be noted that this example is equally applicable to a a synthetic study.



* may be a dummy data set



JCL card (/* card)

Blank card

Describes the production-interaction curve

Specifies the format of the INTERACTION cards

Describes the desired trip length frequency

Specifies the format of the LENGTH cards

Enters the production and attraction (or origin and destination) volumes for each centroid.

Specifies the format of the GENERATION cards

Optional - used to define a sector structure

Optional - may be used to enter replacement values for separations between centroid pairs

Used to define the parameters M and FUTURE

Specifies the routines to be executed and their order of execution (in this case: MODIFY, EDIT, SET, ACCEPT, MODEL, GET, STOP)

Optional - used to identify the run

JCL cards

SAMPLE DECK SET-UP FOR EXAMPLE II

## SAMPLE JCL FOR EXAMPLE II

```
//JOBLIB   DD   UNIT=PRIVATE,VOL=SER=TTIPLN,DISP=SHR,
//  DSN=TTIPLN.THDMODEL
//GO   EXEC   PGM=MODEL,REGION=130K
//FT02F001   DD   DUMMY
//FT08F001   DD   UNIT=TAPE9,VOL=SER=PSD359,DSN=SKIMTREE,DISP=OLD
//FT04F001   DD   UNIT=TAPE9,VOL=SER=PSD008,DSN=UPSKMTRE,
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT01F001   DD   DUMMY
//FT03F001   DD   UNIT=TAPE9,VOL=SER=PSD449,DSN=HPWTRIPS,
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT25F001   DD   UNIT=TAPE9,VOL=SER=ZZ0001,DSN=SV,
//   DCB=(RECFM=VBS,LRECL=1000,BLKSIZE=1004)
//FT07F001   DD   SYSOUT=B
//FT06F001   DD   SYSOUT=A
//FT05F001   DD   *
```

## EXAMPLE III: COMBINING TRIP MATRICES FOR ASSIGNMENT

Data Available:

- Three production/attraction trip matrices prepared by this package (ADD1, ADD2, and ADD3 data sets

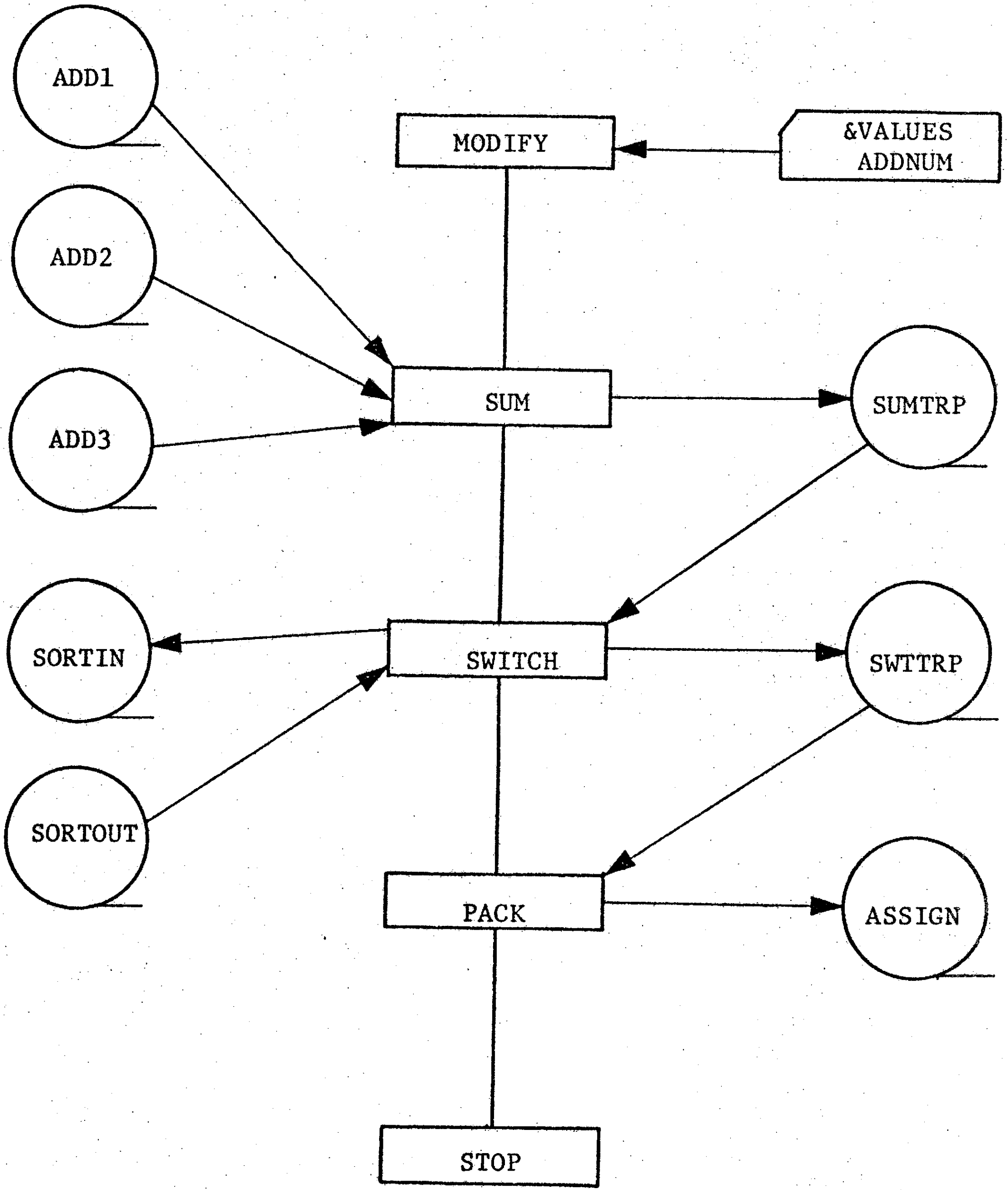
- The ADDNUM parameter (i.e., the number of trip matrices to be combined)

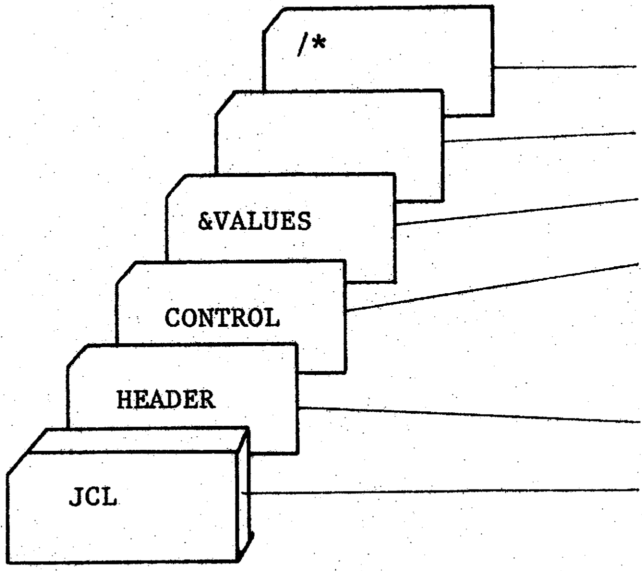
Objectives:

- To combine the three given production/attraction trip matrices into a single production/ attraction trip matrix (the SUMTRP data set)

- To switch the production/attraction trip matrix (the SUMTRP data set) to an origin/ destination trip matrix (the SWTTRP data sets)

- To reformat the origin/destination trip matrix (the SWTTRP data set for input into either the Texas Large Network Package or the Texas Small Network Package (the Assign data set)


ADD1
MODIFY
&VALUES
ADDNUM
ADD2
ADD3
SUM
SUMTRP
SORTIN
SWITCH
SWTTRP
SORTOUT
PACK
ASSIGN
STOP



JCL card

Blank card

Used to define the parameter ADDNUM

Specifies the routines to be executed and their order of execution (in this case: MODIFY, SUM, SWITCH, PACK, STOP)

Optional - used to identify the run

JCL cards

SAMPLE DECK SET-UP FOR EXAMPLE III

## SAMPLE JCL FOR EXAMPLE III

```
//JOBLIB   DD   UNIT=PRIVATF,VOL=SER=TTIPLN,DISP=SHR,
//   DSN=TTIPLN.THDMODEL
//GO   EXEC   PGM=MODEL,RFGION=228K
//SORTLIB   DD   DSN=SYS1.SORTLIB,DISP=SHR
//FT09F001   DD   UNIT=SYSDA,DSN=&INSORT,SPACE=(6004,(1000,100)),
//   DCB=(RECFM=VBS,LRECL=24,BLKSIZE=6004)
//SORTIN   DD   DSN=&INSORT,VOL=*.FT09F001,DISP=(OLD,PASS),
//   DCB=(RECFM=VB,LRECL=24,BLKSIZF=6004),SPACE=(6004,(1000,100))
//SORTWK01   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTIN
//SORTWK02   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTWK01
//SORTWK03   DD   UNIT=SYSDA,SPACE=(6004,(1000),,CONTIG),SEP=SORTWK02
//SORTOUT   DD   DSN=&OUTSORT,UNIT=SYSDA,SEP=SORTWK03,
//   DCB=(RECFM=VB,LRECL=24,BLKSIZE=6004),SPACF=(6004,(1000,100))
//FT10F001   DD   DSN=&OUTSORT,VOL=*.SORTOUT,DISP=(OLD,PASS),
//   DCB=RECFM=VBS
//FT03F001 DD UNIT=TAPE9,VOL=SER=PSD252,DISP=OLD,DSN=HBWTRIPS
//FT017001 DD UNIT=TAPE9,VOL=SER=PSD137,DISP=OLD,DSN=HBNWTRIP
//FT018001 DD UNIT=TAPE9,VOL=SER=PSD077,DISP=OLD,DSN=NHBTRIPS
//FT20F001   DD   UNIT=SYSDA,SPACE=(6004,(200,100)),
//   DCB=(RECFM=VBS,LRECL=6000,BLKSIZE=6004)
//FT13F001   DD   UNIT=TAPF9,VOL=SER=PSD097,DSN=INTTRIPS,
//   DCB=(RECFM=VBS,LRFCL=6000,BLKSIZE=6004)
//FT07F001   DD   SYSOUT=B
//FT06F001   DD   SYSOUT=A
//FT05F001   DD   *
```