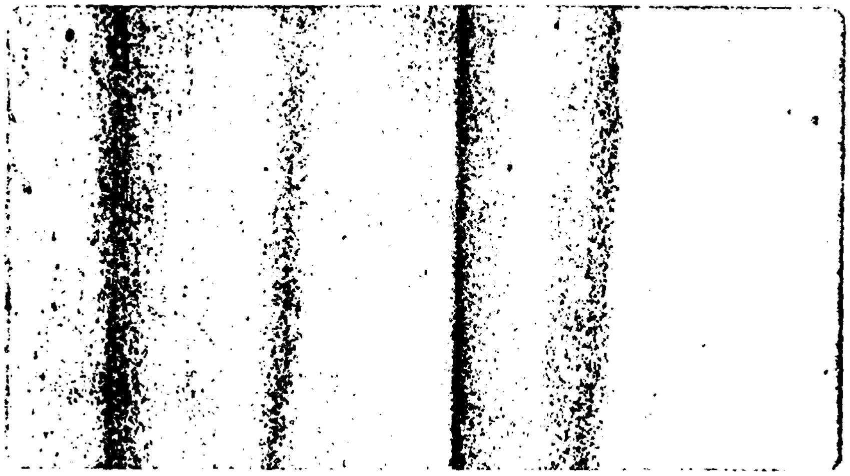

ACADEMY FOR INTERSCIENCE METHODOLOGY

CHICAGO, ILLINOIS

ACADEMY FOR INTERSCIENCE METHODOLOGY

A nonprofit corporation devoted to scientific and educational purposes

MUSEUM OF SCIENCE AND INDUSTRY
CHICAGO, ILLINOIS 60637

Area Code 312
Phone: 667-0240

Final Report

Contract N00014-81-C-0098

December 1981

AIM 81-T-9

by

Merl L. Kardatzke
Norman H. Painter
Adelaide E. Bialek
Samuel S. Ellis
Leo R. Katzenstein
Hubertina S. Loobeek
Douglas W. Smith

Prepared for

Department of the Navy
Office of Naval Research
Arlington, Virginia

Contents

Overview

This report summarizes work completed between 1 January 1981 and 31 December 1981 under contract N00014-81-C-0098 for the U.S. Navy (OP654) by the Academy for Interscience Methodology.

In addition to the work reported in this volume, an investigation of current defense concepts has been completed. The defense concepts investigation is reported in a separate volume which is titled "Current ABM Concepts", by Douglas W. Smith. A briefing of the results of the defense concepts effort has been presented to OP654 by Herbert R. Hesse and Douglas W. Smith.

Chapter 1 of this volume contains a description of the thirty five calls currently available in the RPM 6 model which is under development. Several features of RPM 6 were implemented under Contract N00014-81-C-0098. In particular, user constructed variable masks have been eliminated. RPM 6 calls reference all weapon system, installation and warhead characteristics only by alphabetic name. The variables I1 and I2, which limited weapon wave span in warhead lists in RPM 5, have been eliminated. The number of weapon waves which can be subject to range constraints, which was 18 in RPM 5, has been raised to 32. These and other attributes of RPM 6 have been designed to simplify user inputs to the model.

The Force Mix Model generates weapon allocations which are input to RPM when laydown details and detailed blast and fallout assessments are needed for an analysis. In the past, data taken from the Force Mix Model was transformed into RPM input by hand. Chapter 2 reports the automation of a bridge between Force Mix outputs and RPM 6 STRIKE inputs.

Chapter 3 discusses an update of the program systems used in the Optimum Booster Assignment Method (OBAM). This method uses a network algorithm to solve the problem of assigning SLBM's from specific areas to hard military targets so as to maximize target value killed. The method is now organized into one program package.

A series of support tasks which were completed under Contract N00014-81-C-0098 are discussed in Chapter 4.

Two of these tasks were necessary in order to revise the Optimal Booster Assignment Method as presented in Chapter 3. The MOTION OPTION 6 portion of the SIRNEM program computes probability of damage and value damaged for RV's on boosters.

This routine, which was embedded in SIRNEM, is now a stand alone program. In addition, the subroutine which computes probability of damage was rewritten as a standard subroutine which may be attached to any program requiring damage calculations. This will simplify maintenance as updates to damage functions become effective.

A modification of the FOZAUX program has been completed. This program aggregates DGZ's so that footprinting in the FOZ program may be efficiently executed.

A program to help explore some aspects of procurement stability was written.

A series of runs of the FOZ program developed footprints for 6 data bases for 4 weapon types. These results were used to compute footprint factors for the Force Mix Model.

Finally, a revision of considerable scope of the Force Mix Model (LINMIX) has been completed. A description of this revision concludes Chapter 4.

A volume which is titled "Historic Compendium of U.S. Nuclear Strategic Forces Policy and Doctrine", by Robert A. Blaise was completed under this contract.

Mr. Paul Garvin, OP654, provided the guidance and direction for the technical effort completed in 1981. The Academy takes pleasure in acknowledging its appreciation of his efforts and interest.

Chapter 1. RPM 6 Development

RPM is a computer program which contains a collection of integrated routines. The main thrust of this collection of routines is to provide means by which nuclear exchanges may be examined. If the analyst desires to examine a single detonation or the detonations of many thousand weapons from several countries, the task can be readily handled provided that the underlying geographic data bases are available.

RPM input is a collection of military analyses key words which call for some action. For example, BLAST is a call to compute blast damage. A set of parameters are input following the call to designate the subject and the object of the action. The name of a list of warhead coordinates and the name and location of an installation data set would follow the keyword BLAST. Blast damage to collateral sites and to population may also be computed. Other action calls include FALOUT which evaluates fallout damage to urban and rural population, WHIZ (warhead intended ground zero) which computes weapon aimpoints and GROUP which forms target islands for an installation data set.

To support these action key words, calls such as PRINT and MAP are used to display results. Parameters tell RPM what to print and what to map. Allocation and damage data may by written onto files. These files are used as input to other programs such as the Force Mix Model.

SORT and CHANGE are examples of calls which can be used to manipulate the installation data and the weapon systems data required for action calls.

In order to use RPM, a set of calls are prepared. These calls read data bases, invoke computations required for the analysis and display results. The text which follows describes the data base processing and key word input structure which is currently available in RPM 6.

TITLE Call

TITLE (any message) /

The TITLE call produces a standard title page. Descriptive text which will appear on the title page may be included.

END Call

END /

This call is used to terminate an RPM run. A listing which shows file status and a listing which shows the contents of the Atlas are printed when a run is terminated with an END call.

SCENE Call

SCENE PLOT /

any other calls to be executed /

EXIT /

This structure is used to enter a scenario named 'PLOT' into the RPM Atlas. The scenario consists of a series of RPM calls which are to be executed by subsequent RERUN calls. The dollar sign symbol may be used in place of specific parameters within the series of RPM calls. The parameters which are to replace the dollar signs are input on RERUN calls.

RERUN Call

RERUN PLOT NUMBER ARGUMENTS /

RERUN executes scenario 'PLOT' the stated 'NUMBER' of times with 'ARGUMENTS' replacing $ parameters within the scenario. This call is order dependent.

FORCE Call

```
FORCE ARMS EXTRAS /

WPSYS1 DATA ... /

WPSYS2 DATA ... /

... /

WEND /
```

The FORCE call defines a facility of weapon system characteristics which is named 'ARMS'. Each weapon system has at least 14 characteristics. The number of words reserved for additional characteristics is input as the variable 'EXTRAS". The standard weapon system characteristics, in the required order, are as follows.

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

WAVES Call

WAVES ARM WAVEBEGIN WAVEEND SYSTEMBEGIN ... SYSTEMEND /

The WAVES call identifies the FORCE facility ('ARMS') which contains the names and characteristics of the weapon systems to be used in subsequent RPM calls. One or several FORCE facilities may exist in the Atlas. A WAVES call naming a FORCE facility must precede those calls which refer to weapon systems or waves. These calls are DGZ, WHIZ, ZONE, STRIKE, TARGET, BLAST, WIND and FALOUT.

The WAVES call associates a range of wave numbers, WAVEBEGIN to WAVEEND, with the characteristics of weapon systems SYSTEMBEGIN through SYSTEMEND. For example,

```
WAVES REDWP 1 2 SS17 SS9  /
```

means that wave 1 weapon system characteristics are those of the system named SS17 which is found in FORCE facility 'REDWP'. Wave 2 weapon characteristics are those of system SS9 which is found in the same FORCE facility.

FILE Call

FILE typeunit1 typeunit2 ... typeunit8 MEMSITES /

Lists of data can be read from file and written to file in RPM. Eight standard files may be used.

This call is used to identify binary and coded files. Card images of site data which are to be read into the RPM Atlas and card images written by the RPM PRINT call are coded files. Grouped files, files produced by the RPM READ call, etc. are binary files.

Files are designated as follows.

WB - permits writing and reading binary files (default type).

WC - permits writing and reading coded files.

RB - permits reading only of binary files

RC - permits reading only of coded files

* - spacer used to indicate no change in unit status

If the first definition of a unit with a FILE call is as RB (read binary) or RC (read coded), RPM will rewind the unit and scan for input file names. The name of each file on the unit is entered by RPM into a table of file names.

The MEMSITES parameter is available to redefine the size of Atlas space as MEMSITES of normal width. (Normal width is 7 for CDC; 9 real otherwise.) (May need other action to raise limit on some computers.) This call is position dependent.

CONSTRUCTION OF REMAINING CALLS

On all the calls which remain, parameter data is supplied in order independent sequences. A sequence is of the form LABEL=values. LABEL is used to identify the parameter. It may be as short as one letter or as long as desired provided it ends with an equal sign (no blanks between). The printed output retains the first three letters of the LABEL. Many parameters are optional, in which case standard defaults are obtained by simply not specifying the parameters at all. In many cases, the presence of the LABEL is all that is necessary to select an option. For example, PRINT= or PRT= or P= often is all that is needed to obtain extra information on a call. In other cases, a name or number or more values should be given. The LIST= parameter usually signifies an indefinite list of values separated by spaces or commas, i.e. LIST=13 15,21 27,33 45.

Certain parameters are required by some calls. Most calls need to know the name and/or location of one or two input facilities. If only one input facility is needed and the name is unique, INP=name is sufficient. The program will search first memory and then file 1,2,...,8 until it finds a match. One may limit the search by saying INP=name,location. If location is 0, only memory is searched. If location is 3, only file 3 is searched. If two names are needed, INP=primary facility name,location secondary facility name,location. A particular facility or data set may also be located numerically by INP=position,file location as I=3,2. In this case the third facility on file two is taken.

BCD site data files can be referenced on a FACE (facility) call. Site data may or may not have header lines. In any case the FILE call attempts to read and enter names in the file table. Location by position is an appropriate way to reach such data sets. Positioning by numerical record reference is also valid for binary files, but positioning by name is usually easier and more reliable. Positioning by numerical record requires that the file be explicit, and if BCD input is used for the facility then the file is to be type RC.

One or two output locations are sometimes needed: OUT=name,location or OUT=name1,location name2,location. Usually, output locations are explicit. If a specific file location is not given then the program will search backward from file 8,7,6... for the first available type WB file. A BCD output file for the PRINT call should be type WC or RC.

READ Call

This call is used to transfer a facility from one location to another.

PARAMETER	Effect
I=name of facility to be transferred,location	The input parameter, I, is required.
O=name of transferred facility,location	The output parameter, O, is required.
G=first group,last group	This parameter identifies which groups (or slices or blocks) are to be read if the facility to be transferred is on file. The output facility which is created will contain just those groups, slices or blocks which are between the G parameter limits.

If the facility to be transferred is in the RPM Atlas, a one block facility is written onto an output file location.

If the facility which is output is to be written in memory (location is zero) then the input facility must be on file. The output facility in memory is created by concatenating all the records in the input facility on file. There must be sufficient room in memory to hold all of the sites in the facility.

If the facility to be transferred is on file and the destination is a (different) file then a copy of the facility to be transferred will be written on the output file. Space in memory is required to transfer the largest record. The output file will be the same kind of file as the input file, i.e., a group file, a blocked file, or a slice file.

ERASE Call

This call is used to remove file names from the RPM file name table.

PARAMETER	EFFECT
I=facility name,location	The input parameter, I, names the facility to be erased and gives the location of the facility to be erased.
R=	If the facility to be erased is in the Atlas then the presence of the R parameter causes all facilities which follow the named facility to be deleted as well. If the facility to be erased is on a file, all facilities following the named facility on the file will be erased. If the R parameter is not present and the facility is not the last on the file, the name table will contain an unknown name in place of the facility name. The file remains the same physically.

FACE Call

The FACE call is used to create a facility from BCD input data. The FACE call creates an RPM geographic warhead list or site facility. The FACE, FORCE and SCENE calls are used to construct the geographic data base, weapon system data and scenarios which comprise the RPM ATLAS.

PARAMETER	EFFECT
I=position,file location	If the input parameter, I, is omitted, free format lines of site data are expected. This site data follows the FACE call. Free format site data will be read up to a cutoff line which is as follows. X / If the I parameter is given, input should be on a file which has been designated read coded (RC) or write coded (WC) by means of a FILE call. The initial position of the data records may be set numerically. Lines will be read from this file until a cutoff line is read or an end of file is reached.
O=facility name,location	The output parameter, O, is required. If the location of the facility which is created is 0 then the facility is added to the ATLAS in memory; otherwise it is written as a blocked facility onto file. Only the O parameter is required on a FACE call. All of the other parameters are optional.

T=facility type

The type parameter, T, should be a letter or a word which indicates a site type data structure. The T parameter is omitted when a warhead list is input.

A=

If the A parameter is present, site radii will be computed from site areas which are input.

B=size

The B parameter is used if a blocked file is being made. The default block size, 1000 sites, (100 for PDP) may be increased or decreased.

D=

The presence of the D parameter indicates that the input data set has a header record.

J=number of extra words per site

If the J parameter is omitted, no extra words per site are included in site data width in the ATLAS. If parameter J is 1, 2, or 3, space is included for that many extra words. Data for these extra words need not be input.

P=

If the parameter P is omitted, longitude slice numbers are inserted as group numbers.

If the parameter P is present, latitude slices are inserted as group numbers.

Neither will replace non-zero group numbers, if group variable is read.

F=

The fields parameter, F, is used to define the order in which site characteristics or warhead characteristics are input. The F parameter and the variable format which may follow will adapt RPM data input requirements to user data bases.

If the F parameter is omitted, the first 8 variables are read for each site. The first 7 variables are read for each warhead. The variables are assumed to be in standard order.

The F parameter may be set to an integer value. The number of variables read per site or warhead will be equal to the value of the F parameter.

The F parameter may also be used to re order the default variable order which is as follows.

Default Variable
<u>Order for Site Input</u>

1.	LAT	(L)	-	Latitude
2.	NOR	(N)	-	North or South
3.	MERIDEN	(M)	-	Longitude
4.	EAST	(E)	-	East or West
5.	VALUE	(V)	-	Site Value
6.	CAT	(C)	-	RPM Category Code
7.	RAD	(R)	-	Site Radius in nautical miles
8.	QVN	(Q)	-	Site Vulnerability
9.	KILL	(K)	-	Probability site is dead
10.	SEQ	(S)	-	Numeric Site Identifier
11.	GROUP	(G)	-	Group Number
12.	X	(X)	-	First Extra Word
13.	Y	(Y)	-	Second Extra Word
14.	Z	(Z)	-	Third Extra Word

Standard RPM 6 format for site variables includes provision for a cutoff in column 1. The format is a follows.

(A1,F9.5,1X,A1,F10.5,1X,A1,F6.0,F3.0,F6.2,1X,A4,F5.0,
F6.0,F5.0,1X,A5,1X,A5,1X,A5)

Default Variable
Order for Warhead Input

1.	LAT	(L)	-	Latitude
2.	NOR	(N)	-	North or South
3.	MERIDEN	(M)	-	Longitude
4.	EAST	(E)	-	East or West
5.	VALUE	(V)	-	Warhead Value
6.	WAVE	(W)	-	Warhead Wave Number
7.	HOB	(H)	-	Height of burst in kilofeet
8.	KILL	(K)	-	Probability Warhead is dead multiplied by 1000.
9.	SEQ	(S)	-	Numeric Warhead Identifier
10.	GROUP	(G)	-	Group Number
11.	X	(X)	-	First Extra Word
12.	Y	(Y)	-	Second Extra Word
13.	Z	(Z)	-	Third Extra Word

Standard RPM 6 format for warhead variables includes provision for a cutoff in column 1. The format is as follows.

(A1,F9.5,1X,A1,F10.5,1X,A1,F6.0,F3.0,F6.2,5X,F5.0,
F6.0,F5.0,1X,A5,1X,A5,1X,A5)

The default value for the KILL variable, K, is zero.

These default values for the SEQ variable, S, are set sequentially from 1.

The default values for the GROUP parameter, G, are slice group numbers. (See parameter P).

If the input data does not give the variables in standard order then the F parameter may be set equal to a list of labels which indicate the order in which the variables appear. For example

F=L=N=M=E=V=C=R=Q=K=S=G=X=Y=Z=

is a complete list including extra words of the standard order for site variables.

As another example the standard order and number for a warhead facility is as follows

F=L=N=M=E=V=WAVES=HOB=

If one is retrieving data from a file made by the RPM 6 PRINT Call, and if all variables through the group number are to be read then F=11 is sufficient.

In general the order and number of variables input for each site or each warhead in a facility may be altered by specifying an explicit list for parameter F. Thus

F=X=V=C=L=M=

is interpreted as five variable fields are to be read for each site. These five fields are the first extra word, value, category, latitude and longitude.

Default values will be supplied for variables which are not read. Default values are shown in the table which follows.

Site Variable	Default Value
LAT	0.00000
NOR	North
MERIDEN	0.00000
EAST	East
VALUE	0
CAT	0
RAD	0
QVN	10X0
KILL	0
SEQ	Site Counter
GROUP	Slice Number

Warhead Variable	Default Value
LAT	0.00000
NOR	N
MERIDEN	0.00000
EAST	E
VALUE	0
WAVE	0
HOB	0
KILL	0
SEQ	Warhead Counter
GROUP	Slice Number

(FORMAT)

A format statement, enclosed in parentheses may be included in the FACE call. If (FORMAT) is specified it should be the last parameter. The format may be up to 160 characters long including blanks and parentheses.

If (FORMAT) is omitted, lines are read in free format whether from an input deck, or from file. Only the number of fields specified by the F parameter need be given. The order should be as stated in the description of the F parameter, remembering blank fields cannot be detected. If a line has fewer than the usual fields, the previous site's values are continued to fill out the missing, trailing data.

If a format and an input file are specified, then fixed format reading is done for precisely the number of fields specified by the F parameter. Order may be implied (standard). The order may be altered by the F parameter, altered by tabs in the format, or both.

A variable format should begin with a prefix cutoff field, 'A1', tabbed if necessary. This field does not count as a variable.

There are three special formats: (6), (5), and (0). These mean respectively use the standard RPM6 fixed format, the standard RPM5 fixed format, or the last variable format given.

LIST Call

The LIST call displays the facility names currently in memory and on file.

PRINT Call

This call is used to display data in a facility. It is also used to put facility data on a BCD file (type WC or RC).

PARAMETER	EFFECT
I=name of facility to be printed,location	The parameter I is required. The facility may be in memory or be a blocked, sliced, or group file.
O=name of facility which is written,file location	If the parameter O is omitted, no BCD file is written.
G=first group,last group	If the G parameter is omitted, all records of file are used. If the G parameter is used only the "groups" between the limits specified are used.
S=first site,last site	If the S parameter is omitted, all sites are printed. If the S parameter is given and the facility being printed is in memory, the sites or warheads between the limits specified are printed. If input is from file, the limits specified are relative to each record used (which may be limited by parameter G).

T=

If the T parameter is given, a value total is displayed (for the sites indicated if parameter S applies) for each record and a grand total (for the records used if parameter G applies).

Q=

If the parameter Q is given, no display of the sites is made. This does not supress the file data (parameter O option) or the tab (parameter T option).

K=

If the SORT call has made a sort key without actually rearranging the sites, PRINT may use this key to output the data in that order when parameter K is present. If parameter K is not present the order is the current one.

The parameter K should not be used if no key exists. (To use the key option reliably, the PRINT call should immediately follow one or more key SORT calls.)

CHANGE Call

The CHANGE call is used to change the values of a variable (Y) in a facility in memory or file by selected ranges of a second variable (X).

PARAMETER	EFFECT
I=name of the facility to be changed,location	The parameter I identifies the facility to be changed.
O=name of the changed facility,file location	The parameter O is required if the facility to be changed is on file.
Y=standard variable name	The parameter Y identifies the variable to be changed.

The standard variable names are shown in the table which follows.

Site Variables	Explanation
VAL	Site Value
CAT	Site Category
RAD	Radius in nautical miles or Orientation in tens of degrees
VN	Vulnerability
KILL	Probability of Kill multiplied by 1000
SEQ	Site Sequence Number
GRP	Group Number

Warhead Variables	Explanation
VAL	Warhead Value
WAVE	Wave Number
HOB	Height of Burst in Kilofeet
KILL	Probability Warhead is dead multiplied by 1000
SEQ	Warhead Sequence Number
GRP	Group Number

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

X=standard variable name

If the parameter X is omitted, site count is used by default. If the parameter X is present the standard variable name may be any of appropriate kind for the facility. The standard variable name specified by the parameter X may be the same as the name specified by the parameter Y.

L=Xlow,Xhigh,Ynew, ...

The parameter L should specify at least three values. More than one set of triples may be given however.

Xlow and Xhigh indicate that sites whose X variable fall within this range (inclusive) should be changed. Ynew is the new value given to the standard variable name specified by parameter Y for these sites. If the input is a multiple record file (group, slice, or block), the list is scanned for each record and applied as appropriate. X=seq may be used rather than site count.

SPLIT Call

The SPLIT call is used to divide the sites of a facility in memory or on file into two parts, i.e. those sites which are in range and those sites which are out of range.

PARAMETER	EFFECT
I=name of facility to be split,location	The parameter I is required.
O=name of facility of in range sites,location name of facility of out of range sites,location	If input is a file, at least one output file is required. This file will contain the sites which are in range. The file containing the out of range sites is not written unless two specifications follow parameter O.
X=standard variable name	The parameter X is required to identify the variable on which to split.

The standard variable names are shown in the table which follows.

Site Variables	Explanation
VAL	Site Value
CAT	Site Category
RAD	Radius in nautical miles or Orientation in tens of degrees
VN	Vulnerability
KILL	Probability of Kill multiplied by 1000
SEQ	Site Sequence Number
GRP	Group Number

Warhead Variables	Explanation
VAL	Warhead Value
WAVE	Wave Number
HOB	Height of Burst in Kilofeet
KILL	Probability Warhead is dead multiplied by 1000
SEQ	Warhead Sequence Number
GRP	Group Number

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

B=Xlow,Xhigh lower bound,upper bound

The parameter B is required. This parameter specifies the lower and upper variable limits on sites which are in range.

N=name of facility of out of range sites

The parameter N is an option which is used when input is in memory and a second facility in memory is desired to hold the out of range sites.. When the N parameter is specified, the O parameter does not apply. (No parameter O option).

SORT Call

The SORT call is used to reorder the sites of a facility in memory or in a file according to the values of some variable.

PARAMETER	EFFECT
I=name of facility to be sorted,location	The I parameter identifies the facility to be sorted.
O=name of sorted facility,file location	The O parameter is required if the input facility which is to be sorted is on file. The input file is sorted separately within records.
X=standard variable name	The X parameter defines the variable on which the sort is to be made. If the X parameter is omitted, no new sort is made.

The standard variable names are shown in the table which follows.

Site Variables	Explanation
VAL	Site Value
CAT	Site Category
RAD	Radius in nautical miles or Orientation in tens of degrees
VN	Vulnerability
KILL	Probability of Kill multiplied by 1000
SEQ	Site Sequence Number
GRP	Group Number

Warhead Variables	Explanation
VAL	Warhead Value
WAVE	Wave Number
HOB	Height of Burst in Kilofeet
KILL	Probability Warhead is dead multiplied by 1000
SEQ	Warhead Sequence Number
GRP	Group Number

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

D=

Normally a sort is in ascending order. Specifying the parameter D gives descending order.

C=

SORT primarily generates a sort key to record the new order. Initially this key does not exist and some other calls, WHIZ, STRIKE, etc., may destroy a key. Even if a previous key is there the user may not want it. In all these cases the clear option, parameter C should be specified. If parameter C is not specified, the old key is used in generating the new one. In this way previous sorts may become subsorts of the latest sort.

M=

If the parameter M is omitted, sites are not actually shuffled. Specifying parameter M forces an actual shuffle. The final key shows a natural order.

A=

The parameter A is an option for the PDP 11/34 only to force an alphanumeric sort.

GSPLIT Call

The GSPLIT call is used to deal the records of an input file into two separate files.

PARAMETER	EFFECT
I=name of facility to be split,file location	
	The Parameter I identifies the facility on file which is to be split.
O=primary output facility name,file location secondary output facility name,file location	
	The O parameter is required. The records in the primary file are those which are between the ranges specified by the L parameter. The secondary file contains the remainder, if any. The secondary file is not made if its name and location are omitted.
P=	
	The parameter P is an option to display record movement.
L=g1,g2,g3,g4,g5,g6. ...	
	The parameter L defines a list of pairs of desired primary range intervals. Usually the input file is a group or slice file and the range intervals are the desired group numbers assigned to the records. (Site group numbers are not checked.) If the file is a blocked file, the ranges should be actual record counts.

GMERGE Call

The GMERGE call is used to combine the records from two files into one. Three file specifications are required. The input files should be in the same kind, either grouped or sliced.

If no options are given, all records from both input files are interlaced in order or merged into one record when they match.

PARAMETER	EFFECT
I=primary facility name,file location secondary facility name,file location	
O=merged facility name,file location	
M=	The M parameter is an option, which if specified, merges only the secondary records that match the group numbers of the primary file.
S=	The S parameter is an option which places the records of the secondary file after those of the primary file. If necessary, the secondary records are renumbered to higher values.

SLICE Call

The SLICE call is used to create a slice file from the individual site group numbers in a facility on file. The parameters I and O are both required. The input file may be blocked, sliced or grouped but is usually a blocked file containing slice numbers. A blocked file may be created by a READ call for a facility in memory. A better alternative for a facility in memory is a GROUP call to write the facility to file.

PARAMETER	EFFECT
I=name of input facility,file location	The I parameter is required.
O=name of slice file which is output,file location	The O parameter is required.
P=	The P parameter is an option for more detailed print.

GROUP Call (Input on File)

The GROUP call is used to create a geographic island grouping of the sites in an input facility or to refine a previous grouping.

PARAMETER	EFFECT
I=name of facility to be grouped,file location	Parameter I is required. If the facility to be grouped is on file, then parameter O is required. The input file must be a sliced or grouped facility.
O=name of grouped facility,file location	Parameter O names the facility which is created by this GROUP call.
D=distance	The parameter D is required with distance the desired linking (and separation) distance in nautical miles. If the input file is a previously geographically grouped facility, distance should be smaller than the distance previously used to be effective.
R=	If the parameter R is not specified, the input records are assumed to be slices and the groups may spread over consecutive slices. If parameter R is specified, the sites in each record are examined independently and regrouped to the new distance.

P=

The parameter P is an option for the print of more detailed slice flow that is usually not valuable unless a complex analysis is needed.

GROUP Call (Input in Memory)

This GROUP call is used to create a geographic island grouping of sites in an input facility, to write a group by group file for a facility in memory or to refine a previous grouping.

PARAMETER	EFFECT
I=name of facility to be grouped,O	
	The I parameter is required.
O=name of grouped facility,file location	
	If the parameter O is present, a grouped file will be written. If the D parameter is not specified and the O parameter is then no new grouping is made. The input facility is sorted by group number and a grouped file is written. Facilities in memory which have longitude or latitude slice numbers in the group field may be converted to slice files in this way. The slice file produced in this manner may be used as input to a GROUP Call (Input on File) to achieve a geographic distance grouping
D=distance	
	If the D parameter is present a geographic grouping by distance will be executed. Sites will be sorted by group number. For large facilities, grouping from a slice input file is more efficient.

COLECT Call

The COLECT call is used to form groups of sites with matching variable values. Input may be a facility in the Atlas or a block, slice or group file. The output file, if any, is a block file which has the same record for record structure as the input file, it is sorted within records. "Collecting" however is across record boundaries.

PARAMETER	EFFECT
I=name of facility to be "collected",location	The I parameter is required.
O=name of "collected" facility,file location	The O parameter is required if the input facility is on file.
X=standard variable name	The X parameter is required. It names the standard variable on which the grouping is made.

The standard variable names are shown in the table which follows.

Site Variables	Explanation
VAL	Site Value
CAT	Site Category
RAD	Radius in nautical miles or Orientation in tens of degrees
VN	Vulnerability
KILL	Probability of Kill multiplied by 1000
SEQ	Site Sequence Number
GRP	Group Number

Warhead Variables	Explanation
VAL	Warhead Value
WAVE	Wave Number
HOB	Height of Burst in Kilofeet
KILL	Probability Warhead is dead multiplied by 1000
SEQ	Warhead Sequence Number
GRP	Group Number

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

RELATE Call

The RELATE call correlates the sites of a facility to the groups of a basic facility by distance or variable. Uncorrelated sites are placed in group 4095.

PARAMETER	EFFECT
I=name of basic facility,0 name of facility to be correlated,location	The I parameter is required. The basic facility must be in memory.
O=name of correlated facility,file location	If the facility to be correlated is on file then an output file is required
B=Xlow,Xhigh	The B parameter supplies bounds which apply to the variable specified by the X parameter.
X=standard variable name	If the X parameter is omitted, correlation is by distance. If a site is closer than Xlow of a basic site, it is immediately assumed to belong to the same group. Else, the minimum distance is found to all basic sites and, if less than Xhigh belongs to the closest group. Otherwise it is uncorrelated.

The standard variable names are shown in the table which follows.

Site Variables	Explanation
VAL	Site Value
CAT	Site Category
RAD	Radius in nautical miles or Orientation in tens of degrees
VN	Vulnerability
KILL	Probability of Kill multiplied by 1000
SEQ	Site Sequence Number
GRP	Group Number

Warhead Variables	Explanation
VAL	Warhead Value
WAVE	Wave Number
HOB	Height of Burst in Kilofeet
KILL	Probability Warhead is dead multiplied by 1000
SEQ	Warhead Sequence Number
GRP	Group Number

Force Variables	Explanation
NAME	Weapon System Name
FULL	Inventory Fullness
BASE	Source Facility
RVS	Warheads per Carrier
CEP	CEP in Nautical Miles
YLD	Yield in Megatons
ABT	Abort Probability
HGT	Height of Burst in Kilofeet
FF	Fission Fraction
WGT	Warhead Weighting Value
RMAX	Maximum Range in Nautical Miles
RMIN	Minimum Range in Nautical Miles
CARS	Number of Carriers (Boosters)
HDEG	Optimum Height Degradation Factor

MAP Call

PARAMETER	EFFECT
I=name of first input facility,location, name of second input facility,location	Either one or two facilities may be mapped. The first facility to be mapped will be represented by the character X. The second facility to be mapped will be represented by the character O. Map center and scale will be computed automatically.
G=first group,last group	This parameter identifies groups for which maps are to be made. If the G parameter is not specified, all groups on the input file will be mapped.

REDUCE Call

PARAMETER	EFFECT
I=name of facility to be reduced,location	
Q=name for reduced facility,location	
G=first group,last group	This parameter identifies which groups are to be reduced. If the G parameter is not specified, all groups will be reduced.
H=	The presence of the H parameter causes each group to be reduced to one average multiple base site. If the H parameter is not present then each group is reduced to one area site with density fit.

CIRCLE Call

PARAMETER — **EFFECT**

I=name of input group file,file location

O=name of output file of DGZ's,file location

D=maximum coverage circle size

> The D parameter is the maximum coverage circle radius in units of nmi.

V=site value below which sites will not be covered by circles.

P=

> If the P parameter is present, details of the circle construction will be printed.

G=first group,last group

> This parameter identifies the groups for which coverage circles are to be computed. If the G parameter is not specified, all groups on the input file will be examined.

M=first group to be mapped,last group to be mapped

> This parameter identifies the groups for which maps are to be printed. Group numbers may range from 1 to 4095. If the M parameter is not specified, no group will be mapped. If the first group to be mapped is specified and the last group to be mapped is not specified then all groups past the first group will be mapped.

DGZ Call

PARAMETER	EFFECT
I=name of input group file,file location	
O=name of output file of DGZ's,file location	
W=weapon wave number	This parameter identifies the wave number of the weapon system which is to be used. A WAVE Call must have been given previously to associate the wave number with a weapon system in a FORCE facility. The W parameter must be specified. The height of burst specified for the weapon system will be used in the DGZ calculation.
C=upper limit on category code for objectives,lower limit on category code for sites to be avoided	The C parameter is used to specify which sites in the input group file are objectives and which sites are to be avoided. If the input group file contains no sites which are to be avoided, the second value for the C= parameter may be omitted. If the C parameter is omitted then all category codes through 31 will be objectives and no sites will be avoided.
P=	If the P parameter is present, details of the DGZ development will be printed.

A=avoidance requirement

If the A parameter is less than 1 then the kill on any site to be avoided will remain below this fraction. If the A parameter is 1 or greater then the avoidance site value which is killed will be below the value which is input. If the A parameter is omitted, the default value is .5.

V=residual target value not worth a weapon

The residual target value below which the target will not be considered. This input must be greater than 1. If the V parameter is not specified, the default residual target value not worth a weapon is 100.

G=first group,last group

This parameter identifies the groups of installations for which impact locations will be developed.

If the G parameter is not specified, all groups on the input file will be examined.

M=first group to be mapped,last group to be mapped

This parameter identifies the groups for which maps are to be printed. Group numbers may range from 1 to 4095. If the M parameter is not specified, no group will be mapped. If the first group to be mapped is specified and the last group to be mapped is not specified then all groups past the first group will be mapped.

WHIZ Call

PARAMETER	EFFECT
I=name of input group file,location	
O=name of output file of weapon aimpoints,location	
W=first wave,last wave	This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. The W parameter must be specified. If "last wave" is not specified only the first wave will be allocated.
H=weapon height of burst option	G - Use the weapon height of burst which is given in the FORCE facility. L - Use the maximum of the given height of burst and the optimum height of burst. S - Use surface bursts. If the H parameter is not specified, optimum heights of burst will be used.
G=first group,last group	This parameter identifies the groups of installations for which weapon aimpoint locations will be developed.

If the first group is specified and the last group is not specified then all groups past the first group will be considered.

M=first group to be mapped,last group to be mapped

This parameter identifies the groups for which maps are to be printed. Group numbers may range from 1 to 4095. If the M parameter is not specified, no group will be mapped. If the first group to be mapped is specified and the last group to be mapped is not specified then all groups past the first group will be mapped.

C=lowest objective category code,highest objective category code

Installations are objectives if their RPM category codes lie within the limits set by parameter C. If parameter C is not specified, categories 0 through 59 are objectives.

A=lowest avoidance category code,highest avoidance category code

Sites will be avoided if their RPM category codes lie within the limits set by the parameter A. If parameter A is not specified, no sites will be avoided. The default value for the highest avoidance category code is 59.

B=

If the B parameter is present then weapon placement will be influenced by by-product damage to sites which are not objectives.

Q=avoidance value ratio

A potential DGZ is acceptable if the ratio of avoidance damage to objective damage is less than or equal to the value input for the parameter Q.

If the parameter Q is omitted then the avoidance value ratio is .2.

R=

If the R parameter is present then the centroid of each group will be checked to verify that the group is within range of each weapon system.

P=lowest group number for detailed print,highest group number for detailed print

This parameter identifies the groups for which maps are to be printed. Group numbers may range from 1 to 4095. If the P parameter is not specified, no group will be printed. If the first group to be printed is specified and the last group to be printed is not specified then all groups past the first group will be printed.

L=vs.ps vs.ps vs.ps ....

Objective damage requirements are considered satisfied if the surviving value of a site is less than or equal to the input vs or if the site probability of survival is less than or equal to the input value ps, whichever is first. This pair of inputs is given for each objective category code beginning with category code 0.

V=residual target value not worth a weapon (special case - WGZ calculation)

The presence of the V parameter indicates that this calculation is restricted to one weapon system. Avoidance may be computed. This mode of WHIZ parallels the WGZ call of RPM5. The height of burst option, H, is not available. Heights of burst stored with the weapon systems will be used.

S=

The presence of the S parameter indicates that STRIKE has allocated weapons to objectives. The STRIKE allocation is used as WHIZ inventory on a group by group basis.

ZONE Call

PARAMETER	EFFECT
I=name of facility to be zoned,0	
W=first wave,last wave	This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. If the W parameter is not specified then waves 1 through 31 will be used.
Z=	The Z parameter causes a print of ranging data for individual sites.

STRIKE Call

PARAMETER	EFFECT

I=name of facility to be struck,0

W=first wave,last wave

This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. If the W parameter is not specified then waves 1 through 31 will be used.

H=weapon height of burst option

G - Use the weapon height of burst which is given in the FORCE facility.

L - Use the maximum of the given height of burst and the optimum height of burst.

S - Use surface bursts.

If the H parameter is not specified, optimum heights of burst will be used.

C=lowest objective category code,highest objective category code

Installations are objectives if their RPM category codes lie within the limits set by parameter C. If parameter C is not specified catagories 0 through 31 are objectives.

N=the maximum number of passes

If any weapon system becomes inventory limited, the number of passes will be limited to the N parameter value. If the N parameter is not present, the number of passes is limited to 9.

A=allocate simultaneously using a maximum of this number of cycles

The presence of the A parameter indicates STRIKE is to allocate weapon systems simultaneously. The value of the A parameter is the limit on the number of cycles over all weapon systems. The default value for the maximum number of cycles is 3.

B=multiple base mode indicator

The presence of the B parameter indicates that this STRIKE is in the multiple base mode.

M=maximum number of warheads

The M parameter specifies the maximum number of warheads to be allocated to any one target.

D=probability of surviving defense

The D parameter determines the probability each warhead survives the defense.

R=check ranges

The presence of the R parameter indicates that ranges from bases to objectives are to be checked.

P=

If the P parameter is present details of the STRIKE will be printed.

TARGET Call

PARAMETER	EFFECT
I=name of facility which was struck,0	
O=warhead facility name,location	
	A warhead facility will be written only if the U parameter is absent.
W=first wave,last wave	
	This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. If the W parameter is not specified then waves 1 through 31 will be used.
C=lowest objective category code,highest objective category code	
	Installations are objectives if their RPM category codes lie within the limits set by parameter C. If parameter C is not specified categories 0 through 31 are objectives.
M=multiple or single warhead per site in warhead facility	
	If this parameter is 0 then each entry in the warhead facility may represent several impacts. If this parameter is 1 then each warhead is a separate entry in the warhead facility.

H=weapon height of burst option

G - Use the weapon height of burst which is given in the FORCE facility.

L - Use the maximum of the given height of burst and the optimum height of burst.

S - Use surface bursts.

If the H parameter is not specified, optimum heights of burst will be used.

P=

If the P parameter is present, details of the TARGET call will be printed.

U=

If the U parameter is present, target values will be updated. If the U parameter is absent a warhead facility will be written on the output file.

X=number of extra words for each warhead in the warhead facility

This parameter is used only of the U parameter is absent.

WIND Call

PARAMETER	EFFECT
I=name of warhead facility,0 name of sequential weather data file,file location	
	This parameter identifies the warhead facility in the Atlas for which weather data is to be computed. If the source of the weather data is CDC sequential weather data then the name and location of the weather data are also identified using the I parameter.
R=file which contains GAD or GWC random access wind data	
	The presence of the R parameter determines that random access GAD or GWC weather data will be used (as opposed to CDC sequential weather data).
H=cloud height in kilofeet for CDC sequential winds	
	If the H parameter is not specified for CDC sequential winds a default cloud height of 20 kilofeet will be used.
S=effective crosswind shear component for all sites in knots per kilofeet	
	If the S parameter is not specified for GAD or GWC weather data, an effective crosswind shear component for each site will be computed from wind data.

P=

If the P parameter is present, effective wind vector data will be printed.

Q=

If the Q parameter is present, details of the random access WIND call will be printed.

BLAST Call

PARAMETER	EFFECT
I=name of warhead facility,O	name of site facility,location
	The I parameter identifies the warhead facility for which blast damage is to be computed. This facility must be in the Atlas. In addition, the I parameter identifies the site facility for which damage is to be assessed. This facility may be in the Atlas or may be a group file.
O=name of output file of sites,file location	
	If the site facility to be assessed is on file, and if site probability of survival is to be updated (U parameter) then an output file must be named. This output file will contain the updated site data.
N=evaluation point density	
	N is the number of points at which damage will be evaluated per square nautical mile. If the N parameter is negative then value is uniformly distributed. (Rural evaluation uses uniform distribution independent of sign.) If the N parameter is positive then value is normally distributed. The default value for the N parameter is -2.

W=first wave,last wave

This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. If the W parameter is not specified then waves 1 through 63 will be used.

V=site vulnerability

The V parameter specifies a vulnerability which will be used for all sites for which damage is being assessed. If the V parameter is not present, the vulnerabilities in the Atlas will be used.

G=first group,last group

The G parameter identifies the groups for which blast damage is to be computed if input is from a group file.

P=

The P parameter provides output options.

1 - Print assessement of each site.

2 - Same as 1 plus print the damage each warhead does to site it affects. Successive warheads damage site surviving value.

K=

If the K parameter is present, site kill level is updated.

Q=

If the Q parameter is present, details of intermediate blast damage calculations are printed.

FALOUT Call

PARAMETER	EFFECT
I=name of warhead facility,0 name of facility site to be assessed,location	
O=name of assessed sites,file location	
W=first wave,last wave	This parameter identifies the wave numbers of the weapon systems which are to be used. A WAVE Call must have been given previously to associate these wave numbers with weapon systems in a FORCE facility. If the W parameter is not specified then waves 1 through 31 will be used.
U=	If the U parameter is present, site kill level is updated.
G=first group,last group	The G parameter identifies the groups for which fallout damage is to be computed if input is from a group file.
N=evaluation point density	N is the number of points at which damage wil be evaluated per square nautical mile. If the N parameter is negative then value is uniformly distributed. (Rural evaluation uses uniform distribution independent of sign.) If the N parameter is positive then value is normally distributed.

The default value for the N parameter is -2.

P=

If the P parameter is present, details of site assessment will be printed.

S=gate.shear

The integer part of the S parameter is the dosage for which rectangular calculation gates are generated.

The fractional part of the S parameter is the crosswind shear component.

If shear is zero, the EM-1 deposition mode will be used for fallout calculations.

If shear is greater than zero and less than .99, the WSEG/NAS deposition mode with this fixed shear in knots/kilofeet will be used for all warheads.

If shear is equal to .99, shear components stored in the warhead list by the WIND call will be used with the WSEG/NAS deposition mode.

PROTAB Call

PARAMETER	EFFECT
L=protection factor percent of population at this protection factor ...	This call is used to input a table of protection factors. This table is used when population damage due to fallout is computed. At most twelve pairs of numbers may be input.

Chapter 2. Automation of Force Mix Outputs to RPM 6 Inputs

The Force Mix Model (LINMIX) not only uses results from RPM detailed runs as input but, in addition, for studies such as the two-sided Net Assessment, LINMIX output is further analyzed in RPM. The solution of the Mixed Integer Linear Programming (LP) model provides allocations of weapon inventory to hard target types and soft target types as defined for LINMIX. These allocations to target types may be input to RPM for analysis of the detailed data bases which correspond to each LINMIX target type. Individual DGZ's for objectives are developed by RPM and RPM may be used to compute detailed collateral damage. In the past these allocation results were only available on the printed output from LINMIX from which they were extracted and input to RPM. LINMIX also stores and uses weapon system data required by RPM. In the task reported in this chapter, after analysis of the LINMIX to RPM linkage, a new procedure was designed to simplify the process.

LINMIX provides the allocations to the LP problem in a FORTRAN readable solution file. This optional output is available under the APEX III verb called OUTPUT. This file may be saved for subsequent processing after the LINMIX run.

A separate LINMIX to RPM editor program was developed which uses the permanent data base and the solution file from LINMIX. This editor, called ELIX, provides a complete RPM run which reflects the LINMIX allocation. The permanent data base accumulated by LINMIX and designed for LINMIX to create Linear Programming problems is now used by ELIX to provide weapon system data for RPM. Elix also calculates retargetable reliability-survivability rates to adjust the allocations input to RPM in terms of numbers of weapons of each weapon system type assigned to each LINMIX target type. A complete input deck for RPM is generated for each target type. These RPM decks are written as separate files on an output file which is called LIX. This file may be positioned for any target type used in LINMIX.

For example: if there are three (3) hard target types, A01, B01, and C01; and if there are six (6) soft target types, A51, A52, B51, B52, C51, and C52; then there will be nine (9) files on the output file named LIX. The sixth file would be for target type B51, which could be positioned and used as RPM input as follows.

```
REWIND(LIX)            or  ATTACH(LIX,...)
COPYBF(LIX,DUMMY,5)
RPM,,LIX.
```

The RPM 6 input decks on the file called LIX may be copied to the PUNCH file and punched on cards. These cards can be modified or combined in different RPM runs.

Example of RPM 6 Input Prepared by ELIX

Figure 1 is an example of the RPM decks produced by ELIX. Hard targets have a limit of M=2 on the STRIKE card and soft target types have a limit of M=250. M is used by RPM as the limit on the number of weapons per DGZ. The RPM calls used in this example are explained in Chapter 1 of this volume.

```
LISTING OF INPUT DECK

 TITLE ( RPM6 STRIKE TEST WITH LINMIA/APEX INPUT FOR TARGET TYPE BS1 ) /
 ( FOR COUNTRY-TARGET TYPE BS1                    COPYRF,LIX,DUMI, 5.      ) /
  FILE  RC RC /
 FACE D=GRID,0 I=1,1 T=A F=11 (5) /
 FACE U=BASE,0 I=1,2 T=A F=11 (5) /
 CHANGE I=GRID,0  Y=CAT L=1 38 1 /
 PRINT I=GRID /
 PRINT T=BASE /
 FORCE WEAP /
 W01B51 1 BASE 1    .1000   .1000   .0690 0.00 1.00 1.00 9999.  0.     0 0.  /
 W02B51 1 BASE 1    .2000   .2000   .7625 0.00 1.00 1.00 9999.  0.     0 0.  /
 W04B51 1 BASE 1    .5000   .5000   .0690 0.00 1.00 1.00 9999.  0.     0 0.  /
 W05B51 1 BASE 1    .5000   .5000   .7625 0.00 1.00 1.00 9999.  0.    32 0.  /
 W06B51 1 BASE 1    .2333   .5000   .5710 0.00 1.00 1.00 9999.  0.     0 0.  /
 W20B51 1 BASE 1    .0500   .5000   .9575 0.00 1.00 1.00 9999.  0.     0 0.  /
 W22B51 1 BASE 1    .5000   .5000   .0500 0.00 1.00 1.00 9999.  0.     0 0.  /
 W03B51 1 BASE 1    .3000   .3000   .0500 0.00 1.00 1.00 9999.  0.     0 0.  /
 WEND /
 WAVES WEAP 1  1 W04B51 /
 PRINT I=WEAP,0 /
 STRIKE I=GRID,0  W=1, 1 C=1 P= M=250 D=.5 A= /
 PRINT I=GRID,0 /
 TARGET I=GRID,0 O=WHERE,0 C=1      /
 TARGET I=GRID,0 C=1 U= P= /
 PRINT I=GRID,0 /
 PRINT I=WHERE,0 /
 END /
```

Figure 1. Example of RPM INPUT Prepared by ELIX.

Multiple Time Periods

LINMIX has the capability to produce a semidynamic problem which extends over several time periods. Allocations of weapons to each target type must occur to meet the specified requirements and objectives of each time period. Weapon characteristics and reliability-survivability estimates may also change in the permanent data base. This problem has been solved by including the time period in the permanent data base as a variable.

The LINMIX to RPM editor, ELIX, may be executed for each time period using the permanent data base that is appropriate for that time period. In order to prepare RPM decks by the ELIX editor it is necessary to save the permanent data base for each time period. This can be done by writing the PERM DB on a file as each time period of the LP problem is prepared by LINMIX. Before execution of ELIX the permanent data base should be positioned on TAPE11 for the current time period.

For example the following execution selected the PERM DB for time period 2. In this case the file attached as TAPE11 contained two files: one for time period 1 and one for time period 2.

```
ATTACH(TAPE11,MLKXTAPE7A,...)
ATTACH(TAPE12,MLKXSOLUTIONFILE...)
COPYBF(TAPE11,DUMMY)
ATTACH(ELIX,MLKXELIX...)
ELIX.
CATALOG(LIX,MLKXLIX30NOV81,...)
```

Sample of ELIX Control Card Set

ELIX requires no input except from files. RPM input is written by ELIX onto a file. In this sample, files TAPE11 and TAPE12 were prepared by LINMIX and used by ELIX. RPM is executed twice using previously prepared data base files for the target set and the weapon source. In this sample the target data base is on file TAPE1 and the weapon source data base is on fle TAPE2.

```
ELIX,T20,IO100.
USER,XXXXXXX,XXX.
PROJECT,*XXX*.
HEADING. ELIX
ATTACH(ELIX,MLKXELIX30NOV81,ID=XXXXXXX)
ATTACH(TAPE12,MLKXSOLUTIONFILE16NOV81,ID=XXXXXXX)
ATTACH(TAPE11,MLKXTAPE7A,ID=XXXXXXX)
COPYBF(TAPE11,DUM)
ELIX.
ATTACH(TAPE1,MLKXT1FORRPM6,ID=XXXXXXX)
ATTACH(TAPE2,MLKXT2FORRPM6,ID=XXXXXXX)
ATTACH(RPM,AEBXRPM6,ID=XXXXXXX)
REWIND(LIX)
RPM,,LIX.
REWIND(TAPE1)
REWIND(TAPE2)
REWIND(LIX)
COPYBF(LIX,DUMI,5)
RPM,,LIX.
(END OF FILE)
```

ELIX Program Card

The ELIX program uses four files. No input from the file called INPUT is required. The program card is reproduced here for those who may use this information when executing ELIX.

```
PROGRAM ELIX(TAPE11,LIX,TAPE12,OUTPUT)
```

Chapter 3. OBAM - Optimal Booster Assignment Method

The Optimal Booster Assignment Method was designed to allocate missile boosters to military targets. The weapon systems and the inventory of each weapon system may vary from launch point to launch point. The targets will vary in hardness, in value, and in distance from any given launch point. The amount of damage done in terms of value will vary with launch area, weapon system, target value, target hardness, and target location. Hence the task of choosing an optimal assignment of boosters to footprints in terms of value destroyed is not trivial. One requirement for these problems is that the DGZs be footprinted. Because of this requirement the program FOZ is used in the methodology. The FOZ program determines combinations of RV's which form feasible footprints from one or more areas. Another requirement is that no footprint receive a second weapon until all footprints have received at least one weapon. To satisfy this requirement the problem is solved using a suboptimal, iterative procedure. The first iteration of this procedure solves the subproblem in which only one booster can go down on any given footprint. The values of the footprints are updated after this and subsequent subproblems. Each subsequent iteration also allows only one booster to be assigned per footprint. Eventually all boosters to be assigned are assigned or the problem is infeasible. Note that while each subprogram in this procedure may be solved optimally, the total procedure is not optimal, but suboptimal.

In the past OBAM consisted of programs FOZ, MOTION OPTION 6 in SIRNEM, COST, PKNET, NETFLOW, and SETUP. In order to solve the allocation problem, first FOZ, MOTION OPTION 6 and COST were executed. Then repeated executions of PKNET, NETFLOW and SETUP were required. The COPE program and Control Data Corporation's CCLINK capability were used to execute this program sequence. CCLINK allowed storage and use of preprogrammed control cards. These control cards ran the above programs in the proper sequence until the problem was solved. NETFLOW was a CDC proprietary program. It was necessary to build PKNET as a preprocessor and SETUP as a postprocessor in order to solve the iterative problem mentioned above. Because CDC no longer supports CCLINK, because other computers are not compatable with CCLINK, and because a non-proprietary improvement on NETFLOW, PNET, had been incorporated into the sequence it was decided to rewrite the methodology.

In addition, OBAM was rewritten because the original methodology was restrictive in the problems it could solve. One restriction in the old methodology was that the target values were updated by footprints rather than by DGZs, thus potentially introducing inaccuracies into the problem. The option of updating either by footprints or DGZs is preferable. Also the OBAM methodology was limited to one weapon type/target type. The new OBAM will handle up to three weapon types with three target types, giving nine different possibilities. Note that in reality the nine possibilities could be nine weapon types with one target type, or one weapon type with nine target types,etc. That is, the true capability of the program is nine weapon type/target type cases, even though the labelling in the program implies a limit of three weapon types and three target types.

The current OBAM methodology today consists of the programs FOZ, SHRINK, CEPCOM, PNETIN, AND ITERA. ITERA contains as a subroutine the network code, PNET. All iterations take place within one program, ITERA. All the control language needed to run the three programs PKNET, PNET, and SETUP is incorporated in the new program ITERA.

FOZ is the first program in the methodology. The Foz program constructs footprints for the DGZs and produces a matrix of launch areas vs footprints. In this matrix an entry of one means that missiles in the given launch area can reach the targets in the given footprint. A zero entry means that the footprint can not be reached. MOTION OPTION 6 (CEPCOM) takes this matrix of zeros and ones and replaces the ones with probability of kill (PK) multiplied by value. CEPCOM computes these PKs as a function of the range from the launch point to the DGZ. The CEPCOM program is described in detail in Section A of Chapter 4 of this report. The PKs, the values, and the DGZ names are written on a file to be used later by program ITERA. No other data is written to this file. ITERA uses these variables to update the values of the DGZs after each subprogram is solved.

Program PNETIN is run after CEPCOM and its purpose is to create an input deck for PNET. PNET exists in two forms, as an independent program and as a subroutine in ITERA. The input deck which PNETIN creates will work with either PNET form. Hence the program PNETIN was written in a more general form than the other programs in the OBAM methodology. It was designed to be used not only as input to the OBAM methodology but as immediate input to PNET for other types of netflow problems. It is essentially a preprocessor for PNET. In the OBAM methodology, after PNETIN has created an input deck, ITERA is run with the output of PNETIN as input. ITERA is the program that contains the iterative procedure needed to solve the problem where one weapon must go down on every target before any target receives a second weapon. ITERA is set up to solve several allocation problems at once. The requests for boosters to be assigned must be sorted with the largest request for boosters to be assigned coming first. By inputing the problem this way, the OBAM methodology can store and reuse solutions to subprograms that are common to any of the later problems. After the first set of iterations for the first problem, it will take one iteration or less in ITERA to solve each subsequent problem. This is true since the subprogram solution steps for the allocation of the largest number of boosters are also solution steps for subsequent problems.

The inputs to the OBAM methodology are just the inputs to the programs that OBAM consists of; FOZ, SHRINK, CEPCOM, PNETIN, and ITERA. The main FOZ inputs are the DGZs used to create target footprints. In addition FOZ launch areas are input, so that it can be determined which launch areas can reach which footprints. A matrix of zeros and ones is created from this information identifying which launch area can reach which footprint. This matrix and a list of the footprints are put out on files. SHRINK has no user inputs, it takes the file with the zero/one matrix and reduces the number of areas in the problem if any area cannot reach any footprint. This modified matrix file and the footprinted DGZ file are inputs to CEPCOM. For each weapon type/target type desired a new FOZ run must be made. If multiple weapon type/target types are to be run in one problem, then the files are stacked for CEPCOM by the program SHRINK.

CEPCOM computes the PKs for each missile/DGZ combination. It replaces the ones in the matrix created by FOZ with the sum of the products of each DGZ value and the associated PK for that given launch area. The values of the DGZs come from the file of footprinted DGZs while the PKs are computed by CEPCOM from user input. CEPCOM then outputs the matrix and a list of the footprinted DGZs with their associated PKs and values.

PNETIN is the next program to be run in the OBAM methodology. Its input is mainly from the user. As described earlier, PNETIN creates an input deck for PNET, the netflow program. Creating this input deck consists mainly of writing the arcs of the netflow problem. A subroutine in PNETIN writes one set of arcs at a time, and the user inputs information describing how that set of arcs is to be made. With this flexibility, the user can create netflow problems other than the one in this project. Each arc in the input deck has five parameters; the node the arc is coming from, the node the arc is going to, the cost of the arc for one unit of flow, the upper bound of flow on the arc, and the lower bound of flow on the arc. Many functions, such as boosters per weapon type/target type, are built into the program for use as bounds and costs. These functions must be named in the input if they are to be activated. If a function is not called in the input program, then it is not necessary to enter data for it.

New functions are relatively easy to add to the program as need should arise. The matrix or matrices created by CEPCOM can also be input to PNETIN and also must be called for.

ITERA is the last program in the OBAM sequence. It contains the program PNET as a subroutine. Inputs to ITERA request a given number of boosters to be assigned. This information is inserted into the input deck for PNET created by PNETIN. The list of DGZs and their associated values and PKs are used in this program to update potential return after each iteration. PNET outputs the same deck inputted to it, but with flow and flow/cost columns added. Both the values of the DGZs in the footprints attacked and the number of boosters to be assigned are updated in the output deck. The deck is then ready as input for the next subproblem. This procedure continues until all boosters are assigned or the problem is determined to be infeasible. ITERA can solve several problems in one run. In this case the number of boosters to be allocated should be ordered with the largest number first. The program saves the solutions to the subproblems in the first problem.

OBAM Input

CARD SET 1 (For Program FOZ)

See FOZ input manual.
Up to nine separate runs for one problem may be made.

END OF RECORD CARD

CARD SET 2 (For Program CEPCOM)

See CEPCOM input description in Section A of Chapter 4.

CEPCOM will solve multiple problems in one run. Stack as many CEPCOM decks as the number of runs made with FOZ. Do not put end of file or end of record cards between the CEPCOM decks.

END OF RECORD CARD

CARD SET 3 (For Program PNETIN) (Two cards)

Card 1

		Column
	'PRINT'	1
IDAT	If IDAT equals 'COSDAT' then print COSDAT	11
ITOT	If ITOT equals 'COSTOT' then print COSTOT	21
JAPW	If JAPW equals 'APWTTT' then print APWTTT	31

The file COSDAT is the input deck to PNETIN. COSTOT is the output of PNETIN. The file APWTTT contains some of the basic problem parameters which will be passed to the program ITERA. For example, the number of weapon types, the number of target types, and the number of areas per weapon type/target type will be found in file APWTTT.

For example:

```
Card column
1234567890123456789012345678901234
PRINT     COSDAT    COSTOT    APWTTT
```

Files COSDAT, COSTOT, APWTTT will be printed.

Card 2

TITLE	Title card	Columns 1-54	Format 9A6

For example:

```
Card column
1

TEST CASE
```

CARD SET 4   (For Program PNETIN)

The purpose of these cards is to define the values for any functions which are to be used in CARD SET 5.

Cards I and I+1

Card I

IFN	Name of function	Column 1	Format A5

Card I+1

...	Function values for the function in the previous card.	Free Format

These cards come in pairs. The first card in the pair names the function to be inputted. The function named must be one of the following:

B	Total boosters
BA	Boosters per area
BW	Boosters per weapon type
BT	Boosters per target type
BAW	Boosters per area/weapon type
BAT	Boosters per area/target type
BWT	Boosters per weapon type/target type
BAWT	Boosters per area/weapon type/target type
MP	Maximum number of prints in any weapon type/ target type
PW	Prints per weapon type
PT	Prints per target type
PWT	Prints per weapon type/target type
W	Number of weapon types
T	Number of target types
A	Number of areas
BMP	Boosters minus prints
PSA	Probadility of survival
V	The matrix of launch areas by footprints
BTA	Boosters to be assigned
PMBTA	Prints minus boosters to be assigned
TP	Total number of prints from all weapon type/ target type combinations
IFAC	A factor to be multiplied times a weapon type/target type

The second card in the pair must define all the values of the function. How many values this will be depends on three basic parameters; the maximum number of areas in any weapon type/ target type (A), the number of weapon types (W), and the number of target types (T). The one exception to the rule of two cards per function is V. V causes the CEPCOM matrix to be read from file and hence no card is needed to read in its values. Please note that before V is requested both the number of weapon types and the number of target types should be defined. On the same file as the CEPCOM matrix is the maximum number of areas, the number of areas per weapon type/target type, and the number of prints per weapon type/target type. Hence these variables will be read in whenever V is read in and do not have to be inputted separately. As many pairs of functions and their values as necessary may be inputted.

Last card

IFN 'END'

This card ends the input of functions and their values.

For example:

```
Card column
1

W
2 /
T
1/
V
BAWT
2 2 2 2 2 2 2 2 2 0 /
BTA
87 /
END
```

This input defines two weapon types and one target type. The V instructs the program to read in the matrix from CEPCOM. The last two pairs of inputs define the number of boosters per area/weapon type/target type and the number of boosters to be assigned.

CARD SET 5 (For Program PNETIN)

CARDS N and N+1

CARD N		Columns	Format
AZ	First letter of starting nodes	1	A1
I	Number of type I	2	I3
J	Number of type J	5	I2
K	Number of type K	7	I2
BZ	First letter of ending nodes	11	A1
L	Number of type L	12	I3
M	Number of type M	15	I2
N	Number of type N	17	I2
EZ	First letter of dummy nodes	21	A1
LT=1	Means a dummy for each type L	22	I3
MT=1	Means a dummy for each type M	25	I2
NT=1	Means a dummy for each type N	27	I2
ID1=1	Means I=L	31	I1
ID2=1	Means J=M	32	I1
ID3=1	Means K=N	33	I1
JC=1	Don't write zero cost arcs	41	I1
C1	Cost for npde to node	51	I5
U1	Upper bound for node to node	56	I5
L1	Lower bound for node to node	61	I5
C2	Cost for node to dummy	66	I5
U2	Upper bound for node to dummy	71	I5
L2	Lower bound for node to dummy	76	I5

Types I through L can stand for any variable. They are specifically set up to handle areas or footprints for either or both I and L, weapon types for either or both J and M, and target types for either or both K and N. Note that both I and L can handle three digit integers while J, K, M, and N can only handle two digit integers.

There is one basic requirement on the input deck to PNET. This requirement is that the arcs are in order by starting nodes. An arc is defined on a card by listing first its starting node and then its ending node. The requirement can then be restated as follows; all arcs with the same starting node must be together in the deck. This requirement forced the need for a dummy node option. If a dummy node can not be inserted as an ending (or second) node in a set, then the requirement for the input deck will not be met.

The input JC allows the user to decide if arcs with no cost are to be input into the problem. The user can selectively choose to use or not use the zero cost arcs on an arc set by arc set basis.

The inputs ID1, ID2, and ID3 force the types I and L, J and M, and K and N, respectively, to vary together. This is useful, for instance, if areas of a given weapon type/target type are to go only to targets of the same weapon type/target type.

The inputs C1, U1, L1, C2, U2, and L2 are the costs and bounds for the node to node and node to dummy arcs respectively.

If any of the numbers for types I through L or any of the costs or bounds are negative the next card will provide functions that will give values to replace the negative numbers.

CARD N+1		Columns	Format
IX	Function name (for index I)	1-5	A5
JX	Function name (for index J)	6-10	A5
KX	Function name (for index K)	11-15	A5
LX	Function name (for index L)	16-20	A5
MX	Function name (for index M)	21-25	A5
NX	Function name (for index N)	26-30	A5
CC1	Function name for cost from node to node	31-35	A5
UU1	Function name for UB from node to node	36-40	A5
LL1	Function name for LB from node to node	41-45	A5
CC2	Function name for cost from node to dummy	46-50	A5
UU2	Function name for UB from node to dummy	51-55	A5
LL2	Function name for LB from node to dummy	56-60	A5

Again function names must be chosen from the list in CARD SET 5. This feature lets the number of types I thru N or the costs or bounds be defined in terms of other inputted variables. For example if type I is representing areas, the number of areas can be read from earlier programs in the OBAM sequence by requesting V or can be read into PNETIN through CARD SET 4.

These two cards can be repeated as often as needed.

Last card		Column	Format
AZ	'E'	1	A1
BZ	'N'	11	A1
EZ	'D'	21	A1

END OF RECORD CARD

For example:

CARD N

Card column

```
        10        20        30        40        50        60
123456789012345678901234567890123456789012345678901234567890
A -1-1-1  P -1-1-1  Y0000101  011       1            -1    1
```

```
                  70        80
          12345678901234567890
              0    0   -1     0
```

CARD N+1

Card column

```
        10        20        30        40        50        60
123456789012345678901234567890123456789012345678901234567890
A    W    T    MP   W    T    V                   MP
```

EOR (END OF RECORD CARD)

This example writes arcs from all areas in a weapon type/target type to all footprints in a weapon type/target type; and it does so for all weapon type/target type cases. Note that an arc will be written to a dummy node for each weapon type/target type case. The arcs not going to dummy nodes get their costs from the CEPCOM matrix . Their upper bounds are one, and their lower bounds are zero. The arcs going to dummy nodes have costs and lower bounds of zero, with an upper bound of MP, the maximum number of footprints in any weapon type/target type.

CARD SET 6 (For Program ITERA)

CARD 1

		Column
	'AVERAGE'	1
AVER	'DGZ' gives average value per DGZ.	11
	'FOOTPR' gives average value per footprint.	

For example:

```
Card column
12345678901234567890

AVERAGE   DGZ
```

This says use average value per DGZ.

CARDS 2-N

BTA	Boosters to be assigned. Enter booster requests from highest to lowest.	FREE FORMAT

END OF RECORD

For example:

```
Card column
1

35
34
15
EOR (END OF RECORD CARD)
```

This says 35 boosters are to be assigned in the first problem, 34 assigned in the second problem, and 15 assigned in the third problem.

CARD SET 7 (For Subroutine PNET) (One card)

CARD 1

		Column	Format
ICMND	'SOLVE'	1-5	A6
JCMND	'1'	7	R1

END OF RECORD
END OF FILE

For example:

Card column
1

SOLVE 1

EOR (END OF RECORD CARD)
EOF (END OF FILE CARD)

Chapter 4. Navy Support

A series of support tasks completed under this contract are reported in this chapter.

A. User's Guide to the CEPCOM Program

Introduction

The CEPCOM program is a stand alone version of the MOTION Option 6 module in SIRNEM. It requires a substitution matrix whose dimensions are the number of launch areas and the number of footprints. This matrix may be input from cards, from a binary file on TAPE2 (which is created by FOZ), or may be internally generated by the CEPCOM program. Each element of the matrix is either 0, if the launch area cannot produce the corresponding footprint, or 1, if it can. The elements of an internally generated matrix are all 1.

In addition, the CEPCOM program requires DGZ information (number of DGZs in a footprint, VN, target radius, latitude, longitude, and value) which is input from cards and from file TAPE1, and information about each launch area (latitude, longitude, yield, reliability, and HOB) which is input from cards. There are no restrictions on the number of DGZs per footprint, but in the CEPCOM program no more than 72 launch areas are permitted. In FOZ, however, a limit of 30 launch areas exists.

The CEP of each specific warhead can be made a function of the range it has to travel to its target, or of range and latitude. The CEPCOM program computes CEP by a fifth degree polynomial in either one variable, range, or in two variables, range and latitude. This choice is input, as are the coefficients of the 5th degree polynomial. The CEP of each launch area-DGZ combination is computed.

Once the CEP is known, the CEPCOM program determines the probability of kill associated with each launch area-DGZ combination. This is accomplished by a call to the damage subroutine PKSUB.

A reliability factor can be input to multiply the PK of each DGZ. The CEPCOM program will compute an expected value (PK * reliability factor * value) for each DGZ-launch area pair. This quantity will then be summed over all the DGZs in a given footprint and a value matrix, whose dimensions are the number of launch areas and the number of footprints, will be output as a binary file on TAPE3. Each element of the value matrix may be either the cumulative total of the expected values for each DGZ in a footprint, or the average expected value for a footprint.

The value matrix may then be processed as described in Chapter 3 of this report.

The CEPCOM program requires input both from cards and from file. Two files may be utilized. TAPE1, which contains DGZ information, is required, whereas TAPE2, which may contain the substitution matrix, is optional. Up to eight input card sets may be input, but only four of these are required.

Input From Cards

Card 1, the CEPCOM parameter card, is required. This card specifies the name of the launch area v.s. footprints substitution matrix (MATNAM - this name may be assigned in FOZ), and the name of the launch area set (NAMARA). In addition, card 1 indicates the origin of the substitution matrix (KMAT), whether CEP is a function of range or of both range and latitude (KCOF), whether the value matrix is to contain average expected values or a sum of expected values (KVAL), and three switches (KPRT,KDGZ,KPKL) which control the amount of output printed.

Card set 2 contains one card for each launch area and is also required. These cards provide the latitude, longitude, and name of each launch area. In addition, NGRP, the launch area number, specifies the column of the substitution matrix to which the launch area corresponds.

Either Card 3 or the four cards which comprise Card Set 4 must be input, but not both. These card(s) provide the coefficients of the fifth degree polynomial which determines CEP. The same coefficients will be used throughout the program.

Card 5, the DGZ card, is also required. This card specifies the name of the DGZ set, the number of DGZs per footprint (NP), the yield for all warheads, the VN for all DGZs, the reliability factor for all areas, the HOB for all areas, the evaluation point density, the PK distribution (uniform or normal), the orientation angle for all VN's of type A to F, and the DGZ radius for all DGZs. Some of these quantities may be dependent on the individual area or DGZ, and will thus be provided in Cards 6 and 7, and in the DGZ file on TAPE1. A further explanation is given below.

Card Sets 6 and 7 are optional and specify warhead yield and reliability by area, if this option was selected in fields 4 and 6 of Card 5. If fewer than 72 launch areas are utilized, yield and reliability cards for the remaining areas should not be entered.

Card 8 and Card Set 9 are optional and are to be input only when the substitution matrix is to be input from cards. Card 8 specifies the dimensions of the substitution matrix. Each card of Card Set 9 corresponds to a footprint (row of the matrix) and specifies the matrix elements for this row. Thus the total number of cards in Card Set 9 must be equal to the number of footprints, and the total number of matrix elements on each card must be equal to the number of launch areas.

Input From File

The CEPCOM program utilizes a file of DGZs on TAPE1. These DGZs must be in order according to footprint number. Thus, if the number of DGZs per footprint is given by NP, the first NP DGZs on TAPE1 must correspond to footprint 1, DGZs NP+1 to 2*NP must correspond to footprint 2, etc.

The DGZ file may be generated by RPM, by SIRNEM, by FOZ, or it may be externally created. If generated by RPM, the DGZ file on TAPE1 may be an RPM warhead list or an RPM DGZ list.

a) If TAPE1 is an RPM warhead list, it will contain individual height of bursts in field 7, but not target vulnerabilities or target radii/orientation angles. Hence a positive common VN for all DGZs must be specified in field 5 of Card 5. If this VN is of type A to F, a common orientation angle must be specified in field 10 of Card 5. Otherwise, a positive common DGZ radius must be specified in field 11 of Card 5.

b) If TAPE1 is an RPM DGZ list, it will contain target vulnerabilities in field 8 and target radii/orientation angles in field 7, but not height of bursts. Hence a positive common HOB for all DGZs or a HOB of +1 must be specified in field 7 of Card 5. If a HOB of +1 is entered, the program will compute and use the optimum HOB for each DGZ.

c) If TAPE1 is created by FOZ, it will be in the form of either an RPM warhead list or an RPM DGZ list.

The DGZ file TAPE1 must be in standard RPM format. In particular, column 1 is utilized as a cut-off or header card indicator. If column 1 contains anything other than a blank, the remainder of the card is ignored. TAPE1 may be terminated either by an EOF of by a nonblank in column 1.

If the substitution matrix of 0's and 1's is not input from cards or generated internally, it must be input from a binary file on TAPE2 in standard SIRNEM data list format. The FOZ program can create such a file.

The file TAPE2 must contain two binary records. The first record will contain eight words as follows:

1. 1
2. 'MATRIX'
3. 2
4. MATNAM: name of matrix (up to 6 characters)
5. 1
6. NARA: number of launch areas
7. NPRT: number of footprints
8. NELEP: number of matrix elements + 3

The second record will contain four words, plus one additional word for each matrix element as follows:

1. 2
2. Name of matrix plus '1' in bit 39 (bits are counted from highest order bit beginning with bit 1)
3. '1' in bit 39, 0's elsewhere.
4. 0

5- matrix elements, one complete column
--- (launch area) at a time.

CEPCOM Output

The CEPCOM program will write a value matrix as a binary file on TAPE3. The structure of this file is identical to that of TAPE2, except that each '1' in the substitution matrix is replaced by the corresponding cumulative or average expected value.

An Input Manual for the CEPCOM program follows.

CEPCOM Input

CARD 1 CEPCOM PARAMETER CARD (REQUIRED)

Field	Column	Description	Format
1	1-6	MATNAM: Name of launch areas v.s. footprints substitution matrix	A6
2	7-10	Blank	4X
3	11-16	NAMARA: Name of area set	A6
4	17-20	Blank	4X
5	21	KMAT: Matrix origin 0 = Input from file TAPE2 1 = Input from cards 2 = Generate and set to 1	I1
6	22	KCOF: Input CEP coefficients 0 = Function of range 1 = Function of range and latitude	I1
7	23	KVAL: Value matrix output 0 = Average value 1 = Sum of values	I1
8	24	KPRT: Print output 0 = Print value for each area only 1 = Value matrix (ten prints per line) 2 = Value matrix plus number and name of first DGZ of each print 3 = 2 - Plus range, CEP, PK, D.E. (reliability*PK), VN, WR for first DGZ of each print 4 = 2 - Plus range, CEP, PK, D.E., VN, WR for <u>all</u> DGZ's of all prints	I1

CARD 1 continued

Field	Column	Description	Format
9	25	KDGZ: Print input DGZ's 0 = Do not print 1 = Print	I1
10	26	KPKL: Print probability of kill summary 0 = Do not print 1 = Print	I1

CARD SET 2 LAUNCH AREAS (REQUIRED)

One card for each launch area.
A maximum of 72 areas is allowed.

Field	Column	Description	Format
1	1	NA: File Character X = Last card. Remainder of card is ignored.	A1
2	2-9	ALAT: Latitude	F8.4
3	10	NNS: N or S	A1
4	11-19	ALON: Longitude	F9.4
5	20	NEW: E or W	A1
6	21-25	NVAL: Value (not used)	I5
7	26-30	ARA: Radius (not used)	F5.2
8	31-34	NVNA: Hardness (not used)	A4
9	35-40	NKW: Kill level and category code (not used)	A6
10	41-46	NAMEA: Name of launch area	A6
11	47-50	NGRP: Launch area number. Specifies the column of the substitution matrix to which this launch area corresponds.	I4

CARD 3 RANGE COEFFICIENTS (OPTIONAL)

When KCOF (Field 6, Card 1) = 0

Note: When KCOF = 0 then a five degree polynomial is input as a function of range only (R). Example:

$$A + B + CR^2 + DR^3 + ER^4 + FR^5 = CEP,$$

where A, B, C, D, E, F, are coefficients of polynomial.

Field	Column	Description	Format
1	2-7	Label: RANGE (Optional)	A6
2	8	Blank	1X
3-8	9-80	ABC(1-6): Coefficients A through F are input	6E12.4

CARD 4(A) RANGE AND LATITUDE COEFFICIENTS (OPTIONAL)

When KCOF (Field 6, Card 1) = 1.

Field	Column	Description	Format
1	2-7	Label: RANLAT (Optional)	A6
2	8	Blank	1X
3-8	9-80	BB(1-6): First 6 coefficients of 5 degree polynomial as function of range and latitude	6E12.4

CARD 4(B) through 4(D) (OPTIONAL)

Field	Column	Description	Format
1	2-7	Label: RNLATB, RNLATC or RNLATD (Optional)	6H
2	8	Blank	1X
3-8	9-80	BB(7-21): Last 15 coefficients of 5 degree polynomial. (6 on Card 4(B) and Card 4(C), and 3 on Card 4(D).)	6E12.4

Note: A five degree polynomial as function of range (R) and Latitude (L) will have a total of 21 coefficients according to matrix below.

Field*	Power (Range)	Power (Latitude)	Card (Sub-set)
3	0	0	A
4	1	0	A
5	0	1	A
6	2	0	A
7	1	1	A
8	0	2	A
3	3	0	B
4	2	1	B
5	1	2	B
6	0	3	B
7	4	0	B
8	3	1	B
3	2	2	C
4	1	3	C
5	0	4	C
6	5	0	C
7	4	1	C
8	3	2	C
3	2	3	D
4	1	4	D
5	0	5	D

An example of a five degree polynomial as a function of Range (R) and Latitude (L):

$$A+BR+CL+DR^2+ERL..........TL^4R+UL^5.$$

Then the "A" coefficient goes in Card 1, Field 3, "B" goes in Card 1, Field 4, etc.

* Note: Field 3 starts at Column 9 of each card.

CARD 5 DGZ SET (REQUIRED)

Field	Column	Description	Format
1	2-7	NAMF: Name of DGZ set	A6
2	8	Blank	1X
3	9-14	NP: Number of DGZ's per footprint	I6
4	15-20	YLD: Yield of warhead in megatons > 0 = Use YLD for all areas < 0 = Use yields dependent upon area, see Card 6	F6.2
5	21-26	VN: Vulnerability number (example: 99Q9) VN > 0 = Use this VN for all DGZs VN < 0 = Use vulnerability dependent upon DGZ from file TAPE1	(F4.0,A1,A1)
6	27-32	RELY: Reliability factor between 0 and 1 ≥ 0 = Use RELY for all areas < 0 = Use reliability dependent upon area, see Card 7	F6.4
7	33-38	ALT: Height of burst ≠ 1 and ≠ -1 = HOB in feet for all areas 1 = Program will compute optimum HOB -1 = Use HOB dependent upon DGZ from file TAPE1 Note: If ALT = -1, the same radius for all DGZ's must be input in Field 11 of Card 5, and the orientation angle for all VN types A to F must be input in Field 10 of Card 5.	F6.0

CARD 5 continued

Field	Column	Description	Format
8	39-44	DENS: Evaluation points 0 = Program will provide appropriate number >0 = Number per square nautical mile	F6.3
9	45-50	JDIS: PK distribution 0 = Uniform 1 = Normal	I6
10	51-56	VNANG: Orientation angle in degrees. This input only if VN (Field 5 of this card) is positive and is of type A through F. Note: If VN is negative, then ALT (Field 7 of this card) must not be -1. Orientation angle is then dependent upon DGZ on file TAPE1.	F6.0
11	57-62	RADDGZ: DGZ radius in nautical miles $\geq$ 0 = Same radius for all DGZ's < 0 = Radius dependent upon DGZ from file TAPE1 Note: Also see Field 7 and Field 10 of this card.	F6.2

CARD 6(A) YIELD BY AREAS (OPTIONAL)

When YLD (Field 4, Card 5) is <0.

Field	Column	Description	Format
1	2-7	Label: YIELD (Optional)	A6
2	8	Blank	1X
3	9-14	YLDA(1): Yield for area 1	F6.2
4-15	15-80	YLDA(2-12): Yield for area 2 ⋮ Yield for area 12	11F6.2

CARD 6(B) Same as Card 6(A) for next 12 areas.

(Yield for areas 13-24)

⋮

CARD 6(L) Same as Card 6(A) for next 12 areas.

(Yield for areas 60-72)

CARD 7(A) RELIABILITY BY AREAS (OPTIONAL)

When RELY (Field 6, Card 5) is <0.

Field	Column	Description	Format
1	2-7	Label: RELY (Optional)	A6
2	8	Blank	1X
3	9-14	RELA(1): Reliability for area 1	F6.4
4-15	15-80	RELA(2-12): Reliability for areas 2-12	11F6.4

CARD 7(B) Same as Card 7(A) for next 12 areas.

(Reliability for areas 13-24)

.
.
.

CARD 7(L) Same as Card 7(A) for next 12 areas.

(Reliability for areas 60-72)

CARD 8 MATRIX INPUT FROM CARDS (OPTIONAL)

When KMAT (Field 5, Card 1) = 1.

Field	Column	Description	Format
1	2-7	Label: MATRIX (Optional)	6H
2	8	Blank	1X
3	9-14	NARA: Number of areas	I6
4	15-20	NRPT: Number of prints	I6

(Number of areas $\leq$ 72)

(Number of areas x number of prints = number of matrix elements)

CARD SET 9 MATRIX ELEMENTS (OPTIONAL)

When KMAT (Field 5, Card 1) = 1.

Field	Column	Description	Format
1	2-7	Label: ARPRT (Optional)	6H
2	8	KC: 0 = Additional cards follow 1 = This is the last card of this set. Entire card will be processed.	A1
3-74	9-80	KPMAT: One card for each print (row of matrix). Input number of elements (columns) on a card equal to number of areas input. (Number of cards = number of prints.)	72I1

TARGET-DGZ FILE TAPE1

DGZ's must be in order according to footprint number.

Field	Column		Description	Format
1	1		NA: File character * = Header record or last record. Remainder of record is ignored.	A1
2	2-9		DLAT: Target latitude	F8.4
3	10		NNS: N or S	A1
4	11-19		DLON: Target longitude	F9.4
5	20		NEW: E or W	A1
6	21-25		DVAL: Target value	I5
7	26-30		DRAD: Target radius in nautical miles	F5.2
		or	DHOB: HOB in kilofeet	
		or	DVNANG: Orientation angle in tens of degrees	
8	31-34		NDVN: Vulnerability	A4
9	35-40		NKW: Kill level and category code (not used)	A6
10	41-46		NAMED: Target name	A6
11	47-50		NGRP: Group number (not used)	I4
12	51-80		INFN: Identification information (not used)	5A6

B. The Subroutine PKSUB

The probability of damage function found in the PKTABLES program computes weapon radius and probability of kill as a function of vulnerability, yield, height of burst, CEP, and offset distance. In addition to the PKTABLES program, this damage function occurs in the following AIM programs: RPM, PKFIVE, PDCALC, LINMIX, PKDEC, CEPCOM, INTACT, and SIRNEM. The damage function has been converted into a subroutine form to facilitate its upkeep and distribution.

The subroutine form of the damage function is named PKSUB. It also computes weapon radius and PK as a function of VN, yield, HOB, CEP and offset distance. However, unlike the PKTABLES program, only one offset distance may be used each time PKSUB is called. Alternately, the PKSUB routine may be used to find the offset distance needed to achieve a specific desired PK (to within .001). Information is passed from the main calling routine to the subroutine via a common block named /PKIN/. Information is passed from the subroutine to the calling routine via a common block named /PKOUT/.

The common block /PKIN/ is utilized to pass the following information to the subroutine: NUMVN (vulnerability, which must be an integer in A4 format), yield (in megatons), HOBFT (height of burst in feet, '-1' indicates the optimum HOB is to be computed), CEPNM (circular probability error in nautical miles), OFSET (offset distance from target aimpoint in nautical miles), and the area or orientation of the target (in square nautical miles or degrees). In addition, the following switches must be passed via the /PKIN/ common block: IDBUG, IMETHOD, DENS, IWGT, and IRISK. Lastly, the tape number of the output file, named LOUT, must be specified.

The IDBUG switch controls the output written by the subroutine. Three types of output may be produced: a summary of the contents of /PKIN/ and /PKOUT/, a listing of the evaluation point weights and probabilities for area type targets, and a listing of the interpolation points when interpolation is requested. Any combination of these three outputs may be written on tape LOUT (the default value for LOUT is tape 6) by summing the parameters for the desired outputs. The parameters are as follows:

IDBUG	Subroutine Output
0	No output
1	Summary only

IDBUG	Subroutine Output
2	Evaluation points only
4	Interpolation points only

Hence, a summary is printed when IDBUG=1,3,5, or 7; the evaluation points are printed when IDBUG=2,3,6, or 7; and the interpolation points are printed when IDBUG=4,5,6, or 7.

The IMETHOD, DENS, IWGT, and IRISK switches correspond to the IMETHOD, DENS, WEIGHTED, and IRISK input options in PKTABLES. The options are as follows:

OPTION	VALUE	RESULT
IMETHOD	0	PK computed for each evaluation point.
	1	Interpolation is used to find PK.
DENS	0.	The number of evaluation points is computed automatically.
	AN 361	The number of points per square nautical mile is input. The maximum number of points for any size target is 361.
IWGT	0	Uniform distribution of value over evaluation points.
	1	Normal distribution of value over evaluation points.
IRISK	0	For automatic density, the minimum number of points is 9.
	1	The minimum number of points is 25.

The contents of the common block /PKIN/ are only altered by the subroutine if the offset distance search option is employed. This search is requested by entering a negative offset distance. PKSUB will then locate the offset distance required to achieve the PK specified in the variable 'PKILL' (contained in /PKOUT/) to within a PK of .001. This offset distance is then placed into the variable 'OFSET' (in /PKIN/), and the actual PK achieved is placed into the variable 'PKILL' (in /PKOUT/). No other variables in the /PKIN/ common block are changed. This is accomplished by immediately renaming the input variables. If an unrecognizable VN is encountered, the subroutine prints an error message and stops.

The subroutine PKSUB returns the following information through the common block /PKOUT/: PKILL (probability of kill), WRNM (weapon radius in nautical miles), HOBOUT (unscaled computed height of burst in feet), SHOB2 (scaled height of burst in feet), NP (number of evaluation points in the rectangular array), IAAMM (number of evaluation points within the target radius), DENSI (density value used when automatic density is requested), DENSITY (evaluation point density = IAAMM/area of target), PSUM (sum, over all points within the target radius, of individual point PK*WEIGHT), PNTSUM (sum, over all points within the target radius, of individual point weights), IALT (0/1 switch indicating whether optimum HOB is requested) and IATOE (0/1 switch indicating whether VN is of type A to F).

The PKTABLES Program

The addition of a subroutine version of the damage function to the PKTABLES program resulted in extensive internal modifications to the program, but has had few external effects. The PKTABLE input cards have the same format and structure as before, and the same PKTABLES input decks may be used. However, the following should be noted:

1) The RPM-SIRNEM choice has been removed. RPM is now the only option possible, hence it need not be specified in the input card set.

2) The maximum number of points for any size target is now 361 rather than 400.

3) The default for the IRISK option is now 0 (i.e., 9 points minimum). All other defaults remain as they were, hence all defaults are now 0.

C. FOZAUX Update

The FOZ program groups lists of desired ground zeros (DGZ's) into footprints for MIRVed weapon types in specific launch areas. The DGZ locations which are to be printed may be preprocessed by a computer program called FOZAUZ. A new version of the FOZAUX Program has been developed. The principal reason for this development was to make the file structure of the FOZAUX program more compatible with the recent update of the FOZ program. The FOZAUX update retains only one function of the older version: geographic aggregation. This simplification allowed a corresponding simplification in the usage of the FOZAUX update. In addition to simplifying the model, variable storage technology was added, which allows for more efficient utilization of computer resources. FOZAUX develops its aggregation in one sequential pass through the data. Aggregates represent a set of points that are within a specified distance of a given central point. Each input point is checked in turn to see if it lies within a specified distance of one of the currently defined centers, and if so it is placed into that aggregate. Otherwise the current point becomes the center of a new aggregate.

Using the FOZAUX Program

The FOZAUX Program aggregates geographic points by distance and writes a file of aggregates and a file of individuals suitable for input to the FOZ program.

Program Files

FOZAUX uses five files: INPUT, OUTPUT, TAPE1, TAPE3, and TAPE4. These files appear on the program card in the above order and positions. The INPUT file contains a single input card. This card contains the parameters for the current run.

No OUTPUT is printed by the FOZAUX program.

TAPE1 contains the input points. These points should be 80 column card images with co-ordinates having NSEW indications. TAPE1 is not rewound at the end of the run.

TAPE3 contains the aggregates which are output. The output format for aggregates is the same as the input format for DGZ's. TAPE3 is rewound at the end of the run.

TAPE4 contains the individual DGZ's which are output. All 80 columns of input data are written to this output file.

TAPE4 is rewound at the end of the run.

FOZAUX Input

There is one input card for each run of FOZAUX. This input card contains the aggregation distance, the format of the input sites on TAPE1, and a flag which indicates whether or not the first record on TAPE1 is a site record or a header record.

Variable	Description	Column	Format
N	Flag. If blank the first record on TAPE1 will be treated as a header record. If non-blank the first record on TAPE1 will be treated as a valid site.	1	A1
D	Aggregation distance in nautical miles. Points in an aggregate will lie within a distance D of the center of the aggregate.	2-10	F9.0
IVF	Optional variable format. FOZAUX uses five variables from each record. LAT - Latitude NS - North or South LON - Longitude EW - East or West V - Value Both the latitude and longitude are expressed in degrees, minutes, and seconds in the form dd.mmss. If the variable format field is left blank the format that is used is (1X,F8.4,A1,F9.4,F5.0) which fits RPM standard format. If a variable format is supplied LAT, LON, AND V should be associated with a floating point format specification and NS and EW should be associated with an A1 specification.	11-70	6A10

Note that if the NSEW indicators are blank then positive latitude is North, and positive longitude is East.

Implications Of Variable Storage

Variable storage increases the flexibility of FOZAUX. The user must request adequate field length for the run.

The program requires 14000B (5500) words. Each individual DGZ requires 11B (9) words, and each aggregate requires 5B (5) words. Since the number of aggregates is not known until after execution the user should make an estimate based on the geography of the input points.

D. Model to Investigate Procurement Stability

Questions of stability arise in the formulating of a strategic policy. Strategic stability is usually addressed in terms of incentives for one side to strike first. There is another kind of stability that pertains to strategic force procurement programs. The procurement of strategic forces is a dynamic evolution dictated by the strategic force procurement program required to support strategic policy.

In setting strategic policy and thus strategic force procurement programs the real or perceived strategic policy of the opposing side must be considered. This statement is true for both sides. Depending on the policies adopted by each side one or both of the procurement programs can be coupled to the other. The force characteristics and numbers procured by side Blue can depend on what forces Red builds and vice versa. Thus one can have a dynamic system consisting of two opposing procurement programs that are coupled together. Procurement stability addressed the behavior of this system under various assumptions of opposing strategic policies.

Program DIFYQ was written to explore some aspects of this problem. It is an adaptation of feedback and control theory to the general problem outline in Figure 2. For a specific problem, parameters are set in the program consistent with an assumed Red and Blue strategic policy and the resulting procurement and targeting policy. The output from the program is the force postures of each side as a function of time, where each side tries to attain a posture consistent with its adopted policy.

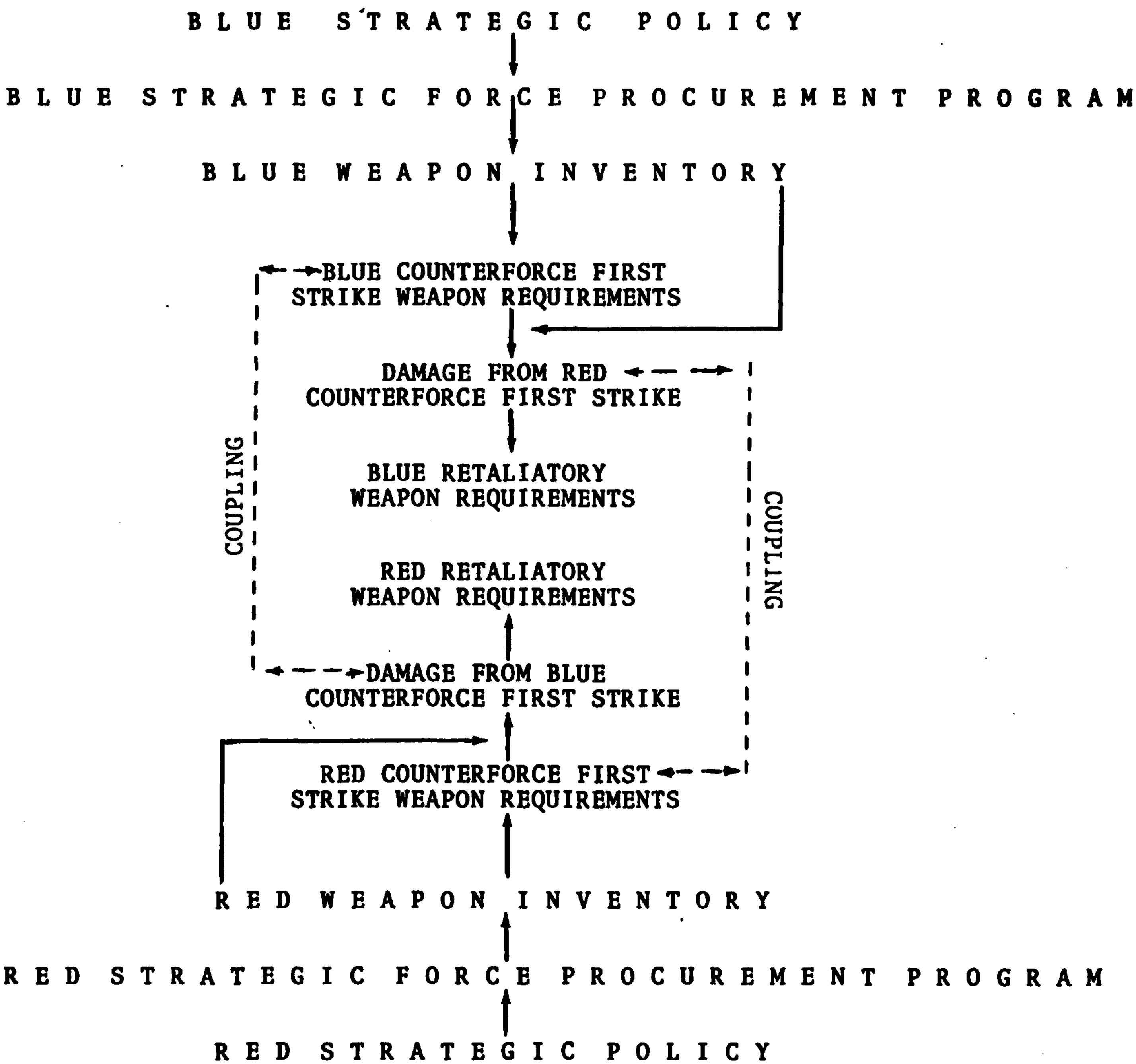

Figure 2. Procurement Stability Model.

E. Footprint Factor Development for the Force Mix Model

The purpose of this project was to calculate footprinting factors for combinations of representative databases and weapon systems for use in the LINMIX methodology. The footprinting factor will represent the effect of footprinting DGZs in the LINMIX methodology.

The LINMIX methodology uses the equation

PK(I,J)=1-EXP(-(G(I)*X(I,J))**H),

to curvefit different allocations of DGZs for a given weapon type/target type into a weapon effectiveness curve. This equation captures the essence of the weapon effectiveness for the weapon type/target type and gives LINMIX an efficient method of storing the data. To expand the methodology, it was of intrest to add the effect of footprinting the DGZs.

The method used to represent the effect of footprinting is as follows:

For each of the DGZ data bases, WHIZ calls were made using the yields of the weapon systems of interest. The DGZs associated with a given combination of database and weapon system were aggregated for a given allocation of DGZS using the FORFOZ scenario in RPM. The DGZs are then footprinted in FOZ. Two sets of data for each combination of database and weapon system result; the footprinted DGZs and the unfootprinted DGZs. Note that the unfootprinted DGZs are equivalent to footprinted DGZs with a loading of one. Next the two sets of data for a given weapon type/target type are curvefit together as one set of data; with each set retaining its loading as part of the curvefit. This was done by modifying the weapon effectiveness equation to include the loading. The weapon effectiveness equation is as follows

PK(I,J)=1-EXP(-(G(I)*X(I,J))**H) .

By substituting the relationships,

1/C = H and (L(I)**(-B)) / A = G(I) ,

the equation becomes

PK(I,J)=1-EXP(-(L(I)**(-B)) / A * X(I,J))**1/C),

where L(I) is the loading.

This curvefit produces weapon effectiveness curves with and without the effects of footprinting. The X(I,J)'s were chosen to give PKs from .3 to .9 in order to give a good curvefit of the PKs of interest on the weapon effectiveness curve.

A curvefit was done for each combination of database and weapon system. The curvefitting was done using the HP-41C. The curvefit produced values for A, B, and C, which along with L(I) (the loading), give G(1), G(I), and H since

G(1) = 1**(-B) / A = 1 / A

G(I) = L(I)**(-B) / A

H = 1 / C .

Then

X(I) = A * L(I)**B * (-LN(1-PK))**C

since

PK = 1 - EXP(-(L(I)**(-B) / A * X(I))**1/C) .

Hence

X(I) / X(1) = L(I)**B

which is called the footprinting factor. The footprinting factor is essentially a multiplier to the G(I)s in the weapon effectiveness equation for the unfootprinted DGZs. Hence, given a weapon effectiveness curve for unfootprinted DGZs and given the loading used in the curvefit, this factor can be used to produce in LINMIX a weapon effectiveness curve for footprinted DGZs with that loading.

F. Force Mix Model Update

Introduction

LINMIX 5 is five dimensional. The dimensions are:

Time
Weapon Systems
Component System Classes
Countries
Target Types

Time is represented by careful selection of time frames. The time frames represent stages at which the long term effects of current decisions will be examined. Time frames may be at intervals of several years as weapon systems are developed. The entry into the force of a new vehicle may be followed by stages which represent increases in numbers of the vehicle until a peak number is reached. Later, refitting of the vehicle with a new weapon system may be represented.

Weapon Systems are represented by a list of weapon systems characteristics and by arrays of probabilities of damage and efficiency coefficients which are computed by LINMIX for each target against which the weapon may be used. Weapon system is a complete concept from the carrier component to the warhead including reliability, survivability, accuracy, and basing.

Component System Classes are the subsystems, facilities, or capabilites which are required for various weapon systems. Each weapon system may have different requirements, but in some component system classes they may compete for the same resources. Component system classes may be thought of as a multi-layered set of networks of components such as bombers, submarines, guidance systems, warheads, bases, boosters, defensive subsystems, and so forth. LINMIX allows a great deal of detail, but these component systems should be selected to reflect the decisions and the trade offs which will be made. Cost is finally based upon component costs. Many costs can best be reflected with respect to the component subsystems which are required for a given weapon system. This results in a total force mix cost where the costs of individual systems can be determined after an overall optimum has been found. The costs of an individual weapon system is dependent upon competitive demands for the same resources: ship yards, bases, personnel, etc.

Countries are a category grouping of target types. Target types may be used to give geographic detail within a specific country. In this case country categories may represent geographic grouping within a country. Other kinds of target categories may be considered based on time dependent value or strategic characteristics of a collection of target types. Within LINMIX any category of target types will be called country or country category.

Target Types (within country) includes hard target types and soft target types which are modeled differently in LINMIX. In LINMIX 5 the number of hard and soft target types allowed in each country has been increased. However, it would seem unwise to fracture the data base too much. Not only could this result in an expensive, overly large model with overwhelming volumes of output, but this model is designed to handle non-homogeneous target data bases. There is a built in assumption that the damage to two nearby targets is taken into account for targets in the same soft target type. If nearby soft targets are not in the same target type, the formulation of LINMIX simply does not consider any collateral damage.

In the case of hard targets, the model size expands rapidly with the number of hard target types. Each hard target type is considered to be a collection of isolated hard targets with the same hardness. If necessary, different hardness and different geographic regions could be considered within a given country or power block by defining separate hard target types.

In designing the names of variables and equation labels it is necessary to set practical limits to the above mentioned dimensions. The character sets which will be used can also be documented. The labeling conventions are included in the following section together with other features of the revised Force Mix Model (LINMIX 5). After the list of special features, the arrays and variables of the permanent and temporary data bases are described. The input manual for LINMIX 5 is the final section of this update report.

List of Special Features

Special features of the revised Force Mix Model (LINMIX 5) are listed below. The maximum dimensions of the problem which LINMIX can handle have been expanded. Additional factors are included within the model.

1. The number of countries was expanded from 3 to 26. The country code is now alphabetic: A to Z.

2. The number of hard target types was expanded from one per country to 50 per country. The target type code of

hard targets is 01 to 50.

3. The number of soft target types was expanded from two per country to forty-nine per country. The target type code for soft targets is 51 to 99.

4. The number of weapon types remains not more than 99, coded 01 to 99.

5. There remain four TRIAD classes: S, I, B, and O as in SLBM, ICBM, bomber and other, respectively.

6. The maximum number of time periods is 9. Each time period may be represented by different weapons characteristics, different numbers of targets, and different target characteristics. Damage requirements are set for each time period. Weapon allocations by LINMIX must be made from the inventories available in each time period. For the purposes of detailed analysis after the allocations are made and damage is calculated by LINMIX, a special editor has been developed (see Chapter 2). To distinguish between different time periods, the variable for time period is included in the permanent data base.

7. A cost and scheduling procedure relates weapon systems availability in each time period with respect to required component subsystems. Component subsystems of a whole weapon system includes necessary elements such as: warheads, guidance, sensors, boosters, aircraft, submarines, silos, and even naval support facilities. The features relevant for modeling a given decision problem are to be carefully selected, but up to 26 component networks for different kinds of things are provided for use in the procedure called SYSTEMTREE.

8. A new parameter called footprint factor, FPF, is included in the permanent data base. There is a value for each weapon type to country-target type combination. Footprint factor is modeled as a factor divided into perfect weapon effectiveness. Therefore, if a target type is more difficult for a certain weapon to print, then to achieve the same level of damage more weapons will be required than before footprinting was considered.

9. A new parameter called probability of connectivity, PC, was formulated and added to the permanent data base. A different value may be input for each weapon type by

country-target type combination. PC is assumed to be part of the non-retargetable probability of reliability used in computing imperfect weapon effectiveness. If this is not true and weapons may be replaced from inventory, the PC should be input as negative.

10. The reliability-survivability mode, J, continues to have five possible levels for each weapon type. However, the permanent data base does not store separate values of the effectiveness coefficients for each level of J as was done in previous versions of LINMIX.

11. Effort has been made to clarify input variables used in the data analysis for perfect and imperfect weapons data. Variables with several uses have been separated and the input manual has been simplified.

12. The permanent and temporary data bases have been restructured and are described in the section which follows.

13. Existing permanent data bases may be written on a binary file and saved and later read from file and revised. The values of parameters and the dimensions of the permanent data base may be changed. The number of weapons, the number of countries, and/or the number of hard or soft target types may be increased. This is useful because once a permanent data base has been built it is usually copied and changed rather than starting from scratch for a new study.

14. A permanent data base created for earlier versions of LINMIX may be read and used as a starting data base for this revised version of the Force Mix Model (LINMIX 5).

15. The dimensions of the temporary data base and its location in storage depend upon the dimensions of the permanent data base. Therefore the temporary data base cannot be input before the permanent data base is in place. If a new permanent data base is read in or a dimension is changed, the temporary data base must be replaced.

Permanent and Temporary Data Base Structures

The PERManent data base structure is open ended with respect to number of countries, number of hard target types in each country, number of soft target types in each country, and the number of weapon types. The dimensions of arrays of the PERM data base are determined by the number of each of these types. Arrays are grouped by their dimensionality.

Target types are designated within country categories so that to make an unambiguous reference to a given target type it is necessary to also include the country reference. In LINMIX 5 the country code is an alphabetic letter A to Z and the target code is a two digit number. Hard target types are indicated by numbers less than or equal to 50 and soft targets are indicated by numbers greater then 50. Therefore, A01 is a hard target type and B55 is a soft target type. The target numbers within each country should be used consecutively. Also notice that target A01 is not the same as target B01 and that A51 is not the same target type as B51 or C51. They represent entirely different target sets. Therefore, reference to particular target types are indicated as "country-target types" and the letters K_L together are used for the symbolic reference to country type K and target type L as the unique reference of a given target type.

The variables and arrays of the permanent data base are divided into seven sections. The reference numbers for the variables in each section are given in the sections which follow with the section number and a two digit number of the variable number in each section.

The TEMPorary data base structure is also open ended for the same dimensional types as above. TEMP input is divided into seven sections, however the fixed arrays of Section 1 are stored in a separate common block. TEMP data base is not stored on file under LINMIX5. It is required that a new TEMP data base will be input whenever the permanent data base is changed in dimension by adding or subtracting or by reading a new PERM data base.

PART A. PERMANENT DATA BASE

Section 1. PERM Data Base Variables

Fourteen variables and a data base description are included in this section. The name of the data base and the description are input data. The other variables are counted by the program itself as data is entered.

Reference Numbers	Variables and Descriptions
1.01.-1.07.	These are the section limits for each section of the PERM Data Base.
1.08.	NAME, Name of the data base.
1.09.	NCTRY, Number of country types.
1.10.	NHT, Number of hard target types.
1.11.	NST, Number of soft target types.
1.12.	NHST, Number of hard and soft target types.
1.13.	NWT, Number of weapon types.
1.14.	NDSC, Number of words in the data base description.
1.15.	NCRIP, Description of the PERM data base on up to 26 cards useing 8 words per card.

Section 2. Country Type Variables

There are five entries for each country category, K = A, B, C, ..., . The name of the country is an input. The other variables are counted as data is entered by the program itself.

Reference Numbers	Variables and Descriptions
2.01.	Name of country category K.
2.02.	Reference number of the last hard target type in this country.
2.03.	Reference number of the last soft target type in this country.
2.04.	Cumulative number of hard target types including this country.
2.05.	Cumulative number of soft target types.

Section 3. Hard Target Type Variables.

This array contains one entry for each hard target type. There are five variables and each will be input for each hard target type.

Reference Numbers	Variables and Descriptions
3.01.	Country code for this hard target type.
3.02.	Target reference number within country category. (L = 01 to 50 for hard targets and they should be used in consecutive order.)
3.03.	Name of this target type.
3.04.	Number of hard targets of this type.
3.05.	Vulnerability of targets of this type.

Section 4. Soft Target Type Variables

For each soft target type, 12 parameters are stored in PERM DB. These may be input or they may be computed in the data analysis section of LINMIX. Two cards are required for each soft target type. Seven variables are on the first card and five on the second in the following order.

Reference Numbers	Variables and Descriptions
4.01.	Country code for this soft target type
4.02.	Target type reference number within this country category. (L = 51 to 99 for soft target types.)
4.03.	Name of this soft target type.
4.04.-4.09.	Perfect weapon parameters, six for each soft target type, are used to estimate probability of damage for perfect weapons of this weapon type. Probability of damage is a function of number of weapons X determined by the parameters A, B, C, D, E and H for each soft target type. Probability of damage for perfect weapons is then, PDPW(X) = A+B*EXP(-(G*X)**H), where G = C+D*LNY+E*LNY**2 is a quadratic in LNY with LNY = $\log_e$(Yield). The yield is in megatons.
4.10.-4.12.	Imperfect weapon parameters, three for each soft target type, are used to estimate probability of damage for perfect weapons: PDIMP(W) = AA+BB*EXP(-(GAM(I,K,L_*W)**F). In this formula W is the number of weapons used and probability of damage is a function of W. The parameter AA, BB, and F may be input or computed, and GAM(I,K,L) is the efficiency coefficient for weapon I against country-targer type K_L stored in Section 6 of this data base.

Section 5. Weapon Type Variables

There are 17 variables which are defined for one value for each weapon type. Thexe are input on three cards; six variables to the card and five on the last card. These are required for each weapon type.

Reference Numbers	Variables and Descriptions
5.01.-5.06.	Name of weapon type and other weapon characteristics.
5.07.	Throw weight per booster.
5.08.	Material per RV.
5.09.	Number of RV's per carrier.
5.10.	Cost per booster.
5.11.	Buy-in cost.
5.12.	Maximum number of RV's for the buy-in cost.
5.13.	Reserve force effectiveness.
5.14.	Prelaunch reliability.
5.15.	Inflight reliability.
5.16.	J-mode for reliability-survivability. (Stored as an integer.)
5.17.	Class for triad: S = SLBM, I = ICBM, B = Bombers, O = other.

Section 6. Weapon by Target Type Arrays

There are six variables defined for each weapon by target type combination. Therefore six arrays are reserved for each weapon type by country-target type combination. The efficiency coefficients, GAM(I,K,L), for soft targets, are initialized to 0 as is the probability of damage, PK(I,K,L), for hard targets. The other variables are initialized as 1.0. These valuse may be modified by input. GAM and PK can be computed by the data analysis functions of LINMIX, in which case, the values would be transferred into the arrays by the program.

Reference Numbers	Variables and Descriptions
6.01.	Alert rate.
6.02.	Prelaunch survivability.
6.03.	Inflight survivability.
6.04.	First weapon PK for hard target types or efficiency coefficient, GAM(I,K,L), for soft target types.
6.05.	Second weapon PK for hard target types or the footprint factor, FPF, for soft target types.
6.06.	Probability of connectivity, PC, for hard and soft targets is generally applied to the non-retargetable probability of delivery. Negative values of PC are applied to the retargetable portion of reliability.

Section 7. Data Analysis Arrays

Space is provided for data analysis of probability of damage to soft targets. Sixty-five (65) spaces are reserved for each weapon type. This data is not preserved when a PERM db is written on tape or on a permanent file. It is zero filled when a PERM db is read in.

Reference Numbers	Variables and Descriptions
7.01.	Yield of the weapon for data analyses purposes. If not input the value already input in the PERM data base will be used.
7.02.	Begin selected data. (Observation Number)
7.03.	End selected data. (Observation Number)
7.04.	Name of the data set taken from the label card. Columns 5 and 6 are the weapon number. The remainder of the name is optional.
7.05.	Factor for the total value of the data base considered. When probability is input for the value of damage, then 1.0 is the appropriate input. If nothing is input, 1.0 is used.
7.06.-7.65.	Up to 30 pairs of values for observed raw data: (1) Probability of damage and (2) Number of RV's (Reentry vehicle warheads). This may be raw data for perfect weapon runs with RPM or it may be imperfect weapon data from RPM. LINMIX may generate its own imperfect weapon data by the algorithm for that purpose. This generated data may be stored in this space in the same was as other raw data.

For each weapon type selected for data analysis, data for only one target type at a time may be stored in this array.

PART B. TEMPORARY DATA BASE

TEMP Section 1. TEMP Data Base Variables

This section is used for input of the name of the TEMP data base and several other variables not included in the open-ended blank common. Budget variables, time period number, name changes for rows or columns, reset of the number of weapon types in the TEMP data base, and selection of static measures are also input in this section.

TEMP Section 2. Country Selection

In this section there is only one variable per country to select countries to be included.

Reference Numbers	Variables and Descriptions
201.	KNOTIN: 0 if this country is not included and 1 otherwise.

TEMP Section 3. Hard Target Requirements.

Target selection and constraints are the same for hard and soft target types.

Reference Numbers	Variables and Descriptions
301.	LNOTIN: 0 if this target type is not included.
302.	Row type: N, G, E or L for linear constraint type.
303.	Minimum number of boosters.
304.	Maximum number of boosters.
305.	PKMIN, required mean probability of damage.

TEMP Section 4. Soft Target Requirements.

Reference Numbers	Variables and Descriptions
401.	LNOTIN: 0 if this target type is not included
402.	Row type: N, G, E or L for linear constraint type.
403.	Booster minimum.
404.	Booster maximum.
405.	PKMIN, required level of damage.

TEMP Section 5. Triad Requirements.

The number of triad requirements is open ended in LINMIX 5 because the number of country-target types is open ended. Four variables are included for each triad requirement row. Triad row names are of the form "TRIADCK_L" where C is the class: S, I, B, or O and where K_L is the country-target type.

Reference Numbers	Variables and Descriptions
501.	0 if this TRIAD type is not included.
502.	Row Type: N, G, E, or L.
503.	Minimum required probability of damage.
504.	Maximum probability of damage. (This is not generally used.)

TEMP Section 6. Weapon Requirements.

Weapon requirements include inventory limits on number of boosters which is a limit used in the linear programming problem. Limits on number RV's and probability of damage (PD) are used to select RAW data or imperfect weapon data for data analysis.

Reference Numbers	Variables and Descriptions
601.	0 if this weapon type is not included.
602.	Inventory limit lower bound on number of boosters.
603.	Inventory upper limit on boosters.
604.	Minimum number of RV's for valid observation for data analysis.
605.	Maximum RV's desired for data analysis.
606.	Minimum PD to include in data analysis.
607.	Maximum PD desired for data analysis.
608.	Switch for use of footprint factor. It is initialized 'FPF'. Footprint factors input in the PERM data base will be used unless 'NO FPF' is input here. Zero also works to avoid use of the footprint factor.

TEMP Section 7. Weapon by Target Type Selections.

One variable is defined for each combination of weapon type and country-target type.

Reference Number	Variables and Descriptions
701.	0 if this weapon-target combination is not allowed. This array is initialized to 0 for all hard target types and 1 for all soft target types as weapons are added to TEMP DB.

Input Manual for LINMIX 5

This input manual is organized in three parts. First, the general formats for LINMIX input are given. The general formats include title cards, function cards, and end cards. The second part covers a minimum set of LINMIX functions. These functions are sufficient to produce a semi-dynamic LP problem beginning with raw data and ending with a problem in standard format. Detail cards necessary for these functions are also specified for the user. In the third part of this manual several additional functions are given. These functions aid the user in changing data bases, doing data analysis, and controlling output.

Part A. General Types of LINMIX Input

The input for LINMIX begins with a TITLE Card and ends with an END OF JOB card. In between there may be any number of function cards. Many of these functions have their own input subsets. Ten column fields are used for LINMIX throughout. Alphanumeric entries should be left justified to begin in columns 1, 11, 21, 31, ... , 71. Numeric entries should include the decimal. The decimal point will follow integer values unless indicated otherwise. In some cases the names used in the 10 column fields have a substructure. When this is true the details will be specified.

1. Title Card.

This card image is printed at the beginning of the printout for each function. It forms the equivalent of a page divider. The first column is used as carriage control. The last 10 column field is used to control the output so that certain variables which may be classified will not be printed.

Field	Column	Contents	Format
1-8	1-80	TITLE: Alphanumeric title. Column 1 is carriage control. '1' in column 1 causes each title to start a new page. If columns 1-3 spell 'END', this will terminate a job.	8A10

If columns 71-80 spell 'CLASSIFIED', then certain variables will not be deleted from the output. If it is known that there is nothing classified in the data then the word 'UNCLASSIFIED' in columns 69-80 contains as a subset the key word in columns 71-80. Thus the variables will not be deleted and the word 'UNCLASSIFIED' will appear with each title line.

Examples

```
1  TITLE CARD -------------------------STARTS NEW PAGE ---

   TITLE CARD -------------------------SKIPS FOUR LINES ---

0  TITLE CARD -------------------------SKIPS AN EXTRA LINE ---

+  TITLE CARD -------------------------SKIPS ONE LESS LINE ---

*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

2. Function Card.

This is the general format of the function card which is specified in detail for each function in the pages which follow. If the function name in column 11-20 is not recognized as one of the legal functions, the job will stop after a diagnostic message. To end the series of functions an 'END' function is used. It is a special case of a function card and will be specified separately.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label.	A10
2	11-20	NAMFCT: Name of function.	A10
3	21-30	FILE: Reference number.	F10.0
4	31-40	NAMM: Reference name.	A10
5	41-50	FNUM: Second reference number.	F10.0
6	51-60	NAMPRM: Second reference name.	A10
7	61-70	NOTE1: These are available as	A10
8	71-80	NOTE2: additional names, but are usually not used.	A10

Examples of Various Function Cards Without Data Subsets:

```
LINMIX****READ                PERM1A

LINMIX****CHANGE

LINMIX****TEMP INPUTPK/TEMP,1A

LINMIX****READ PD(W)

LINMIX****DATASELECT

LINMIX****FIT RAW

LINMIX****GENERATE

LINMIX****FIT GAM             FIX EXP.
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

3. The 'END' Function.

This function stops the series of functions and switches to read the next TITLE card. The end function may be followed by another TITLE card or by an END OF JOB card.

An END OF JOB card is a special case of TITLE card and is specified below.

Field	Column	Contents	Format
1	1-10	NAMOPT: 'END' in columns 1-3 is sufficient to be an end function.	A10
2	11-20	NAMFCT: 'END' in columns 11-13 with columns 14-20 blank is also recognized as the end function.	A10

Example

```
END 1ST PROBLEM
*        *        *        *        *        *        *
1 Column 11       21       31       41       51       61
```

4. End of Job Card.

This card ends the execution of LINMIX. It enables control to go to the next tasks in the control deck which may include execution of APEX. It is necessary to have a normal termination of LINMIX to go on to APEX and use the LP problem which LINMIX generates.

Field	Column	Contents	Format
1	1-10	'END' in columns 1-3 will end LINMIX execution.	A10

Example

```
END JOB TO START APEX
*        *        *        *        *        *        *
1 Column 11       21       31       41       51       61
```

Part B. A Minimum Set of LINMIX Functions

The revised Force Mix Model (LINMIX 5) can produce a semidynamic linear programming problem which is solved by the LP solver in the APEX III applications system. There is a set of eleven functions which produce such a problem from beginning to end. This set will provide a straight-forward reference for the other functions so that the user can consider alternate approaches to get the problem done. The functions listed below will be covered in this part of the Input Manual.

	Function	Brief Description
1.	READ	Read data for the PERM data base.
2.	TEMP INPUT	Read control parameters.
3.	READ PD(W)	Read raw data for probable damage.
4.	DATA SELECT	Select raw data or generated data for subsequent analysis.
5.	FIT RAW	Fit the quadradic and the exponent of the perfect weapon curve.
6.	GENERATE	Generate a set of imperfect weapon data.
7.	FIT GAM	Fit efficiency coefficients and the imperfect weapon exponent.
8.	PK	Probability of damage for hard targets is computed.
9.	LP PREP	Linear programming problems are generated one time period at a time.
10.	ASSEMBLE	LP problems are assembled into a single problem.
11.	SYSTEMTREE	A revise deck is generated which relates weapon requirements across time periods to components costs.

1. 'READ' Function

This function reads input for the permanent data base, PERM DB. The PERM DB contains seven sections. This function reads the input data for six of them and initializes the reserved storage remaining. The input is described for each section.

Section 1. Name and Description of the PERM DB.

This section inputs the name of the PERM DB and an optional description.

Function Card, 'READ'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: Name of function, 'READ'.	A10
3	21-30	FILE: Input is from the input file unless another file is indicated here.	A10
4	31-40	NAMM: Name of PERM DB.	A10

Card Set 1, Description (Up to 26 Cards).

Field	Column	Contents	Format
1-8	1-80	Optional description of PERM DB. 'END' in columns 1-3 will end this section.	8A10

Example

```
LINMIX****READ                PERM1A
          LINMIX5H. OPEN ENDED SEMIDYNAMIC ONE WAY MIX
END OF DESCRIPTION
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Section 2, Names of Countries.

The number of countries is determined by the number of name cards entered here. Names should be entered for each country for which targets, either hard or soft, are entered. Any number of countries (up to 26) may be entered.

Card Set 2, Country Cards.

Field	Column	Contents	Format
1	1-10	Optional label. 'END' in columns 1-3 will end set.	A10
2	11-20	Name of country.	A10

Example

Note the label SECTION 2 as the first card is accepted, but the card is ignored as input. These section labels also serve as terminators for the previous section if a terminator is missing.

```
SECTION 2
C6        COUNTRY 1
C7        COUNTRY 2
C8        COUNTRY 3
END S2

*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Section 3, Hard Target Data.

Any number of hard target types (up to 50 per country) may be input. Each hard target type is input with only one value of the vulnerability number (VN), therefore targets should be grouped so that one VN can characterize the targets within a given hard target type. The PK function in LINMIX will compute probability of damage using the PDCALC routines.

There is one input card per hard target type. The targets of each country should be in ascending order.

Card Set 3, Hard Target Cards.

Field	Column	Contents	Format
1	1-10	Optional label. 'END' in columns 1-3 will end the set.	A10
2	11	K: Country code is alphabetic A to Z and refers to the countries in the order entered in Section 2 above.	R1
3	12-13	L: Target code is numeric 01 to 50 for hard targets and should be in ascending order.	I2
4	21-30	Name of target.	A10
5	31-40	Number of hard targets of this type.	F10.0
6	41-44	Vulnerability of targets.	A4

Example

```
SECTION 3
C11       A01       CTRY1 HARD 2200.    30Q9
C12       B01       CTRY2 HARD 229.     46P7
C13       C01       CTRY3 HARD 30.      40P0
END SEC.3

*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Section 4, Soft Target Data.

Any number of soft target types (up to 49 per country) may be input. Each soft target type is characterized by a perfect weapon equation with a expression for effectiveness that is quadratic in log yield. An imperfect weapon equation is also used as a characteristic of the target type. These equations may be calculated by the program using perfect or imperfect weapon data at the appropriate stages of data analysis.

There are two cards per soft target type. The targets of each country should be entered in ascending order.

Card Set 4, Soft Target Card 1.

Field	Column	Contents	Format
1	1-10	Optional label. 'END' in columns 1-3 will end the set.	A10
2	11	K: Country code is alphabetic A to Z. The code refers to countries in the order entered in Section 2 above.	R1
3	12-13	L: Target code is numeric 51 to 99 for soft targets and should be assigned in ascending order.	I2
4	21-30	Name of target.	A10
5	31-40	A: Usually 1. is input.	E10.0
6	41-50	B: Usually -1. is input.	E10.0
7	51-60	C: Constant term.	E10.0
8	61-70	D: Linear term coefficient.	E10.0
9	71-80	E: Quadratic term coefficient.	E10.0

Card Set 4, Soft Target Card 2.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Optional label.	A10
3	21-30	H: The perfect weapon exponent.	E10.0
4	31-40	AA: Usually 1. is input.	E10.0
5	41-50	BB: Usually -1. is input.	E10.0
6	51-60	F: The imperfect weapon exponent.	E10.0

If LNY = log (yield), with yield in megatons, for a given weapon let G= C+D*LNY+E*LNY**2, then the perfect weapon equation for probability of damage as a function of number of weapons, X, is

PDPW(X)=A+B*EXP(-(G*X)**H).

The perfect weapon equation for probability of damage as a function of the number of weapons, W, is

PDIMP(W)=AA+BB*EXP(-(GAM(I,K,L)*W)**F) where

GAM(I,K,L) is the efficiency coefficient for weapon I against country-target type K_L.

Examples of Soft Target Type Input Data

```
SECTION 4
CARD16    A51       SOFT12    1.        -1.       .00473    .001298   .000123
CARD17    A51       .679      1.        -1.       .5694
CARD20    B51       SOFT22    1.        -1.       .00372    .000022   -.00002
CARD21    B51       .716      1.        -1.       .6626
CARD24    C51       SOFT32    1.        -1.       .00589    .000361   -.00012
CARD25    C51       1.040     1.        -1.       .7447
CARD18    A52       SOFT13    1.        -1.       .001351   .000161   .000005
CARD19    A52       .643      1.        -1.       .6037
CARD22    B52       SOFT23    1.        -1.       .006310   .000214   -.00007
CARD23    B52       .746      1.        -1.       .6888
CARD26    C52       SOFT33    1.        -1.       .000762   .000059   -.00000073
CARD27    C52       .5240     1.        -1.       .5019
END SEC.4

*         *         *         *         *         *         *         *        +
1 COLUMN 11 NUMBER 21         31        41        51        61        71       80
```

Section 5, Weapon Type Data.

Any number of weapon types (up to 99) may be input. There are 17 variables which are defined for each weapon type. There are three cards per weapon type. The order in which weapons are entered determines the weapon number used to reference everything related to that weapon throughout LINMIX. A card with 'END' in columns 1-3 should follow the last weapon type.

Card Set 5, Weapon Type Card 1.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Name of weapon type.	A10
3	21-30	Yield in megatons.	F10.0
4	31-40	CEP for weapon 1.	F10.0
5	41-50	CEP for weapon 2.	F10.0
6	51-60	ET factor.	F10.0
7	61-70	HA factor.	F10.0

Card Set 5, Weapon Type Card 2.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Throw weight per booster.	F10.0
3	21-30	Material per RV.	F10.0
4	31-40	Number of RV's per carrier.	F10.0
5	41-50	Cost per booster.	F10.0
6	51-60	Buy-in cost.	F10.0
7	61-70	Maximum number of RV's for the buy-in cost.	F10.0

Card Set 5, Weapon Type Card 3.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Reserve force effectiveness.	F10.0
3	21-30	Prelaunch reliability.	F10.0
4	31-40	Inflight reliability.	F10.0
5	41-50	J-mode for reliability survivability. (J is stored as an integer.)	F10.0
6	51-60	Class for triad: S = SLBM, I=ICBM, B = Bombers, O = other. (Only the left most character is used.)	A10

Example

```
SECTION 5
WEP1 CD1  ICBM4A     .010      .2        .2        .046      .157
     CD2  8.3        15.2      1.        6.9       0.        54.
     CD3  1.         .5        .67       3.        ICBM
WEP2 CD1  SLBM1B     .200      .15       .17       .342      7.066
     CD2  6.         13.       8.        54.55     6013.     99999.
     CD3  1.         .7        .94       3.        SLBM
WEP3 CD1  SLBM1C     .650      .17       .19       .75       17.015
     CD2  6.         46.       8.        54.55     6015.     99999.
     CD3  1.         .94       .95       3.        SLBM
WEP4 CD1  BOMB4A     .010      .2        .2        .046      .157
     CD2  8.3        15.2      1.        6.9       0.        54.
     CD3  1.         .5        .67       3.        B
END WEP DATA

*         *          *         *         *         *         *
1 Column 11          21        31        41        51        61
```

Section 6, Weapon By Target Variables.

This section would require a great deal of input, therefore a method of input by exception is used. All six variables may appear on one card. If identical values are wanted for all weapons then the weapon number field should be filled with asterisks. If one variable has the same value for all weapons and all target types then columns 5 to 9 should be filled with asterisks. If a variable should not be entered, the field for that variable should be blank. Parameters are changed in the order of the input.

These input cards are of the same form as change cards for Section 6 and therefore they are called Section 6 Change Cards.

Card Set 6, Section 6 Change Cards.

Field	Column	Contents	Format
1	1-10	Change code: decoded as follows: 'END' in columns 1-3 ends this set.	A10
	1-4	Optional Label: 'C601' is standard.	A4
	5,6	Weapon number, I (** for all weapon types),	I2
	7	Country code K (* for all countries),	R1
	8,9	Target code, L (** for all targets),	I2
2	11-20	Alert rate.	E10.0
3	21-30	Prelaunch survivability.	E10.0
4	31-40	Inflight survivability.	E10.0
5	41-50	First weapon PK for hard targets or the efficiency coefficient, GAM(I,K,L), for soft target types.	E10.0

Field	Column	Contents	Format
6	51-60	Second weapon PK for hard targets or the footprint factor, FPF, for soft target types.	E10.0
7	61-70	Probability of connectivity, PC, for hard or soft targets. If PC is retargetable input as negative.	E10.0

Example

In the following example Variable 6, PC, was never changed. Some blank fields are included and they result in no change. Asterisks in the weapon field or in the target number fields are used causing changes across that classification to all parameters for non-blank fields.

```
SECTION 6
C60101A01 .95       .99       .80       .22222
C60102A01 .66       .99       .85       .345      .25
C60103A01 .66       .99       .85       .554
C60103A02 .66       .99       .85       .554
C60103A** .66       .99       .85       .554
C60101A01 .95                                     .20
C60101A51 .95       .99       .80       .0002356
C60101A52 .95       .99       .80       .0005356
C60101A53 .95       .99       .80       .0001356
C601**B** .22       .23       .299      .14       .11
C60102B** .42       .44       .299      .24       .21
C60103C** .42       .44       .299      .24       .21
C601**C51 .77       .88       .55       .0005872  1.2
END T*WEP

*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

This card set completes the description of input for the 'READ' function.

2. 'TEMP INPUT' Function

This function places data into the temporary data base (TEMP DB) or changes a TEMP DB that is already stored. If the PERM DB is changed in any of its dimensions, the TEMP DB is destroyed. A new TEMP DB should be read if the PERM DB is changed substantially or a new PERM is read in. The weapons data is automatically reinitialized unless either the number of weapons variable is set or the SAMEOLDWEP card follows the Function Card.

There are eleven card sets which may follow the TEMP INPUT Function card. These cards may be mixed and input in any order with two exceptions. If the SAMEOLDWEP card is used, it must follow the function card. The END card must be last.

Function Card, 'TEMP INPUT'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'TEMP INPUT'.	A10
3	21-30	Name of TEMP data base.	A10
4	31-40	Number of weapons from a previous TEMP DB which can be used in this function.	F10.0

Card Set 1, 'SAMEOLDWEP'

Field	Column	Contents	Format
1	1-10	NAMCARD: 'SAMEOLDWEP'.	A10

<u>Example</u>

```
LINMIX****TEMP INPUTNAME TEMP
SAMEOLDWEP

*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Card Set 2, 'BUDGET' or 'NO BUDGET' card.

By default, the TEMP Data Base is set for NO BUDGET. That is budget rows will not be produced unless requested. Once the budget option is switched on, it will remain on in subsequent TEMP Data Bases unless switched off.

Field	Column	Contents	Format
1	1-10	NAMECARD: 'BUDGET' or 'NO BUDGET'	A10
2	11	Type Code (E, L, G, or N) for budget row.	A1,9X
3	21-30	Lower bound (if any) on budget.	F10.0
4	31-40	Maximum budget	F10.0

Example

```
BUDGET    N
*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Card Set 3, 'TIMEPERIOD' Card.

This card results in all row names and columns names being changed by inserting a number in column 2 of each name.

Field	Column	Contents	Format
1	1-10	NAMCARD: 'TIMEPERIOD'.	A10
2	11-12	Only the second column in this field is used.	1XA1,9X

Example

```
TIMEPERIODT1
*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Card Set 4, 'COUNTRIES" Cards.

Any number of country cards may be used. Country codes are alphabetic letters. The country codes may be placed in columns 11-70 with or without spaces in between. Countries selected in a previous 'TEMP INPUT' function will be retained and may need to be deleted. All targets in a given country category are deleted when the country is not included in the TEMP DB.

Field	Column	Contents	Format
1	1-10	NAMCARD: 'COUNTRIES'.	A10
2	11-80	NTN(1-70): The code letter for each country included in the TEMP DB. If a country should be excluded, a negative sign may be placed before that country code. Blank columns are ignored and so this field is 'free field'. If all countries on a particular card should be deleted from the TEMP DB, input a zero (0) in column 11.	70R1

Example

```
COUNTRIES A B C

*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Card Set 5, 'TARGETS*K' Cards.

Target by country cards selects those targets within countries which will be included in the TEMP DB. None are included unless selected in a TEMP DB. Unless the permanent data base is changed and the TEMP DB is lost, the targets selected remain in effect. Targets may be deleted by deleting a whole country, but the individual selections of targets in the country remain stored. If the country is again selected these targets will again be active selections. Any number of target selection cards may be included in a TEMP INPUT function.

Field	Column	Contents	Format
1	1-10	NAMCARD: 'TARGETS*K'.	A10
2	11-80	NTN(1-70): The country-target code consists of a letter followed by two digits. A country code followed by two asterisks selects all targets for this country. An asterisk in the country code indicates all countries for the target code selected (if the PERM DB contains them). This field is free field, meaning blanks are ignored. Zero in column 11 means delete all targets on this card. A negative sign preceeding a country-target code means this target is deleted.	70R1

Examples

```
TARGETS*K A01 B01 C01 A51 B51 C51 A52 B52 C52
TARGETS*K A01 A02 A03 A04 A05 A06 B01-B02 B03
TARGETS*K 0 C51 C52
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Card Set 6, 'TGT LIMIT' Cards.

This card set provides the limits of any target type for use in producing requirements for an LP problem. Blank parameter fields are interpreted as zero unless indicated otherwise. Any number of target limit cards may be input.

Field	Column	Contents	Format
1	1-10	NAMCARD: 'TGT LIMIT' OR the alternate 'TARGET KL'.	A10
2	11	Country code: Any country in the PERM DB may be selected. Asterisk (*) selects all countries.	R1
3	12,13	Target code: Any target may be selected for which there is a target of that code for that country in the PERM DB. Asterisks (**) indicate all targets for the country(ies) selected.	I2
4	21	Row type: N, G, E, or L for linear constraint type. Blank is interpreted as G. This governs the damage estimate row.	A1,9X
5	31-40	Minimum number of boosters required on this target type. This is a separate constraint from the damage requirement.	F10.0
6	41-50	Maximum number of boosters on this target type.	F10.0
7	51-60	PKMIN, damage requirement for this country-target type.	F10.0

Example

```
TARGET KL *53      E         300.      1300.      .80
TGT LIMIT A**      G         500.      1000.      .80
TARGET KL A01                                     .232
TARGET KL B01                                     .150
*         *        *         *         *          *         *
1 Column '1        21        31        41         51        61
```

Card Set 7, 'TRIAD' Cards.

Triad row names are of the form "TRIADCK_L" where C is the class: S, I, B or O and K_L is the country-target type code.

Field	Column	Contents	Format
1	1-10	NAMCARD: Substructure is decoded as follows:	A10
	1-5	'TRIAD' is required.	A5
	6	C: S, I, B, or O for the weapon class included in this	A1
	7	K: Country code.	R1
	8,9	L: Target code. C, K, or L may be filled with asterisks (*) to generalize and include all of that level of classification. Blanks or zeros are also treated as asterisks in these fields.	I2
2	11	Zero in column 11 will cause this triad to be deleted.	A1,9X
3	21	Row type: N, G, E, or L.	A1,9X
4	31-40	Minimum required probability of damage.	F10.0
5	41-50	Maximum probability of damage. (This is not generally used.)	F10.0

Examples

```
TRIADS*** 1         N         .2        0
TRIADB1*  1         G         .8        0         .9111
TRIADB**  1         G         .8        0         .9111
TRIADS              G         .2
TRIADIC52           G         .1

*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Card Set 8, Static Measures.

Political and psychological intermediate measures include multi-dimensional measures of each class of weapon in each time period. These measures are evaluated in a static manner based upon inventory or they may be discounted to the value surviving a first strike on the ground.

Field	Column	Contents	Format
1	1-10	NAMCARD: The name must match one in a list of 96 static measures. These names can be split into four subfields.	A10
	1-3	ROWNAM: 'ROW'.	
	4	'I' for Inventory, 'S' for Surviving.	
	5,6	'CA' number of carriers, 'RV' number of RV's, 'MT' total yield in megatons, 'ET' equivalent megatons, 'MA' material, 'HA' special hard metric, 'TW' throw weight, 'RF' reserve force.	
	7-9	Classes: 'SIB' total of 3 classes, '/S' SLBM only, '/I' ICBM only, '/B' bombers only, '/O' others only, '/SI' missile subtotal.	
2	11-20	SNOTIN: Input 1. if this static measure is to be included in the LP problem. Input 0. to delete.	F10.9
3	21	Row type: N, G, E, or L.	A1,9X
4	31-40	Minimum required value.	F10.0
5	41-50	Maximum required value.	F10.0

Examples

```
ROWICASIB 1         N
ROWICA/S  1         G         200.
ROWIRV/S  1         G         500.
ROWIMASIB 1         L                   8000.
*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Card Set 9. Weapon Limits Cards.

These cards set limits for a given weapon type. Limits on inventory are used for producing the LP problem. Other limits are used for selection of data for curve fitting. Limits may be set for minimum and maximum values for both number of RV's and probability of damage (PD) to be used in data analysis. A switch may be set if the footprint factor should be ignored.

Field	Column	Contents	Format
1	1-10	NAMCARD: Enter name of weapon type as it is in the PERM DB or 'WEP NUM=' followed by the weapon number in columns 9,10.	A10
2	11-20	'FPF' is input to use footprint factors stored in the PERM DB. 'NO FPF' means ignore the factor. 'LIMITS' means the footprint factor switch is not changed.	A10
3	21-30	Inventory lower bound on boosters.	F10.0
4	31-40	Inventory upper bound on boosters.	F10.0
5	41-50	Minimum RV's for data analysis.	F10.0
6	51-60	Maximum RV's for data analysis.	F10.0
7	61-70	Minimum PD for data analysis.	F10.0
8	71-80	Minimum PD for data analysis.	F10.0

Blanks in fields 3-8 are ignored and do not change the limits already stored. The limits actually used in data analysis are also affected by default limits for the target type.

Example

```
ICBM4A    LIMITS              3000.
WEP NUM=02LIMITS    1000.     2200.     100.      1000.
WEP NUM=04NO FPF
WEP NUM=05LIMITS    1000.     2200.     100.      1000.
BOMB1A    NO FPF    100.      1000.     100.      1000.
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Card Set 10. Eligible Targets per Weapon Cards.

This card set selects target types to which a given weapon type may be allocated. Weapons are selected by being included in this set or Card Set 9. This card set may also select to delete a given weapon. When the TEMP data base is initialized all soft targets are selected and all hard targets are not included in the eligible set for each weapon type.

Field	Column	Contents	Format
1	1-10	NAMCARD: Enter name of weapon type as it appears in PERM DB or 'WEP NUM=' followed by the number in columns 9 and 10.	A10
2	11-80	NTN(1-70): The country-target codes are entered free field. Blanks are ignored. '-' before the country code deletes that country-target type. Asterisks indicate all countries or targets. Zero (0) in column 11 maybe used to delete this weapon type.	70R1

Example

```
WEP NUM=02A** B** C01
ICBM4A      B01B02B03B05B07B09B11B13B15 B51 B52 B53
WEP NUM=03 *01*02
WEP NUM=060
WEP NUM=07A** B** C01 -A01
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Card Set 11, 'END' Card.

Field	Column	Contents	Format
1	1-3	'END'	A3

Example

This example shows the end card in its correct position at the end of a TEMP data base.

```
LINMIX****TEMP INPUTNAME TEMP
SAMEOLDWEP
BUDGET    N
TIMEPERIODT2
COUNTRIES   ABC
TARGETS*K  *01 B** C51   C52
TARGET KL *53       E         300.      1300.     .80
TGT LIMIT A**       G         500.      1000.     .80
TARGET KL A01                                     .232
TARGET KL B01                                     .150
TRIADS*** 1         N         .2        0
TRIADIC52           G         .1
ROWICASIB 1         N
ROWIRV/S  1         G         500.
ROWSRV/S  1         G         500.
ROWIMASIB 1         L                   8000.
ICBM4A    LIMITS              3000.
WEP NUM=02LIMITS    1000.     2200.     100.      1000.
WEP NUM=04NO FPF
ICBM1C    FPF       100.      1000.     100.      1000.
BOMB1A    NO FPF    100.      1000.     100.      1000.
WEP NUM=02A** B** C01
ICBM4A      B01B02B03B05B07B09B11B13B15 B51 B52 B53
WEP NUM=03 *01*02
WEP NUM=060
WEP NUM=07A** B** C01 -A01
END TEMP DB
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

This card set completes the description of input for the 'TEMP INPUT' function.

3. 'READ PD(W)' Function.

This function stores data in Section 7 of the PERM data base. Damage assessment data from RPM may be input from soft target type data bases for perfect weapons or imperfect weapons. Damage assessment for pure attacks of one weapon type on one target data base can be used as input to LINMIX. The data base selected becomes a target type as defined in LINMIX. This raw data may be accumulated in the PERM DB over a series of calls of this function.

Raw data for only one soft target type may be input and analyzed at a time. Data should be stored for each weapon type to be used on the target type.

Function Card, 'READ PD(W)".

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'READ PD(W)'.	A10

Card Set 1, Weapon Card.

This card specifies the weapon number in columns 5 and 6. A factor for the total value in the target type data and the yield of the weapon may be specified. This card is the header card before a data set for a particular weapon type.

Field	Column	Contents	Format
1	1-10	NAMOPT: decoded as follows:	A10
	1-4	Label: 'END' in columns 1-3 will end the function.	A4
	5,6	I: Weapon number.	I2
	7-10	Copied but ignored.	A4

Field	Column	Contents	Format
2	11-20	FACTOR: Total value of the data base considered. When probability is input, 1.0, the default value, is appropriate. This factor is divided into the value of damage for the target type, herein referred to as probability of damage (PD).	F10.0
3	21-30	Yield: The yield from the PERM DB for this weapon is input if this field is zero. The value of yield input here is local for data analysis and does not affect the PERM DB.	F10.0

Card Set 2, Raw Data Points.

This may be raw data for perfect weapons or it may be for imperfect weapon runs from RPM. Actually the source of the damage data could be any curve relating probability of damage to the number of weapons used. Up to 30 raw data points may be input. Data should be input in order with the number of weapons monotonically increasing.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label. 'END' in columns 1-3 will end the set.	A10
2	11-20	Probability of damage or value damaged.	F10.0
3	21-30	Number of weapons.	F10.0

Data on an END card will be ignored. After a set of raw data points is ended, the program will try to read a Weapon card (Card Set 1). The program requires two END cards after the last set of data to return to the next function.

Example of Raw Data in the 'READ PD(W)' Function.

```
LINMIX****READ PD(W)
C70101A51 1.        .040
02,01033  .3617     100.
04,01033  .5525     200.
06,01033  .6520     300.
08,01033  .7230     400.
09,01033  .7940     500.
10,01033  .8651     600.
END
C70102A51 1.        .100
02,06033  .3892     100.
04,06033  .5863     200.
06,06033  .6829     300.
08,06033  .7539     400.
09,06033  .8249     500.
10,06033  .8959     600.
17,06033  .9907     787.
END
C70103A51 1.        .200
02,07033  .4039     100.
04,07033  .6040     200.
06,07033  .7025     300.
08,07033  .7735     400.
09,07033  .8445     500.
10,07033  .9156     600.
14,07033  .9781     690.
END
C70104A51 1.        .500
02,09033  .4176     100.
04,09033  .6261     200.
06,09033  .7333     300.
08,09033  .8044     400.
09,09033  .8754     500.
10,09033  .9464     600.
12,09033  .9754     641.
END
C70105A51 1.        1.500
02,47033  .4277     100.
04,47033  .6437     200.
06,47033  .7656     300.
08,47033  .8366     400.
09,47033  .9077     500.
10,47033  .9786     600.
END
END DATA
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

4. 'DATASELECT' Function.

This function selects either the raw data or the generated data for subsequent analysis. Two kinds of input are required after the function card.

Function Card, 'DATASELECT'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'DATASELECT'	A10

Card Set 1, Select Card.

This card chooses the country-target type. It also sets default limits for the analysis which follows.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Country-target code.	R1,I2,7X
3	21-30	Minimum probability of damage.	F10.0
4	31-40	Maximum probability of damage.	F10.0
5	41-50	Minimum number of warheads.	F10.0
6	51-60	Maximum number of warheads.	F10.0

Card Set 2, Weapon Order Cards.

This set chooses which weapons will be included in the analysis and in what order. Any number of cards may be used.

Field	Column	Contents	Format
1	1-10	Optional Label. 'END' in columns 1-3 will end the set.	A10
2	11-80	IKL(IY): Weapon Number. Blank fields must consist of 2 columns.	35I2

Example of DATASELECT Function with Detail Cards

```
LINMIX****DATASELECT
SELECT    C52        .3        .9
WEP ORDER 0103040502
END
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

5. 'FIT RAW' Function.

This function will fit the equation for probability of damage by perfect weapons. With A and B fixed the following equation is fitted.

PDPW(X)=A+B*EXP(-(G(I)*X)**H).

This function also fits a quadratic estimate for G(I) where LNY is the log yield for weapon I.

G(I) = C+D*LNY+E*LNY**2.

For example, a set of raw data was fit in the above equations. In this case A = 1.2 and B = -1.2. *The exponent H is solved* iteratively giving H = .6269.

G(I) = .003220 + .0003102*LNY + .000003064*LNY**2.

Weapon Number	Yield	G(I)	Quadratic Estimate
1	.040	.002256	.002253
6	.100	.002525	.002522
7	.200	.002705	.002728
9	.500	.003031	.003006
47	1.500	.003338	.003346

There are three options for the 'FIT RAW' function:

(1) The value of H may be optimized to reduce the standard error for deviations from the estimated PDPW.

(2) The value of H is fixed at the value in the PERM DB.

(3) The value of H is fixed at the value on the input card.

Function Card, 'FIT RAW'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'FIT RAW'.	A10
3	21-30	Not used.	A10
4	31-40	'FIX EXP.' Fix the exponent H.	A10
5	41-50	VV: Value if exponent is input.	E10.0

Examples

```
LINMIX****FIT RAW
LINMIX****FIT RAW                   FIT EXP.
LINMIX****FIT RAW                   FIT EXP.  .78256
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

6. 'GENERATE' Function.

This function will use the curve fitted to perfect weapon data and through an integration process it will generate a set of imperfect weapon data points. These points have been adjusted to take into account reliability and survivability factors which are not retargetable. The integration process is followed by fitting the generated data. For this curve fitting the exponent may be fixed or determined iteratively. Two card sets are required as detailed cards for this function.

Function Card, 'GENERATE'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'GENERATE'.	A10
3	21-30	Not used.	A10
4	31-40	'FIX EXP.' Fix the exponent F.	A10
5	41-50	VV: Value if exponent is input.	E10.0

Card Set 1, Select Card.

This card chooses the country-target type and it sets default limits for the analysis which follows.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	Country-target code.	R1,I2,7X
3	21-30	Minimum probability of damage.	F10.0
4	31-40	Maximum probability of damage.	F10.0
5	41-50	Minimum number of warheads.	F10.0
6	51-60	Maximum number of warheads.	F10.0

Card Set 2, Weapon Order Cards.

This set choses which weapons will be included in the analysis and in what order.

Field	Column	Contents	Format
1	1-10	Optional label. 'END' in columns 1-3 will end the set.	A10
2-36	11-80	IKL(IY): Weapon number codes. Blank fields are ignored, but each blank field is 2 columns long.	35I2

Example of GENERATE Function With Detail Cards

```
LINMIX****GENERATE
SELECT****A51       .3        .85
WEP ORDER 0102030405
END
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

7. 'FIT GAM' Function.

This function will fit an equation to imperfect weapon data which is read in as raw data or is generated in LINMIX. The following is the imperfect weapon equation.

PDIMP(W) = AA + BB * EXP(-(GAM(I,K,L)*W)**F),

where F is the exponent for this soft target type and GAM(I,K,L) is the efficiency coefficient for weapon I against country-target type K_L. AA and BB are input beforehand in the PERM data base and are not optimized here.

Function Card, 'FIT GAM'.

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-20	NAMFCT: 'FIT RAW'.	A10
3	21-30	Not used.	A10
4	31-40	'FIX EXP.' Fix the exponent F.	A10
5	41-50	VV: Value if exponent is input.	E10.0

Examples

```
LINMIX****FIT GAM
LINMIX****FIT GAM                   FIX EXP.
LINMIX****FIT GAM                   FIX EXP.  .98256
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

8. 'PK' Function

This function uses the PDCALC subprograms to give probability of damage or 'PK'. PK is computed for hard targets in each country and for each weapon type selected in the TEMP data base. If the CEP is different for this weapon type when used as the second warhead on a hard target type, then a second PK is computed. If fields 3 to 6 are blank, the default values are assumed: altitude is optimum, offset is zero, and area is zero.

Function Card, 'PK'.

The PK function card causes PK's to be computed against the hard targets in each country for each weapon type. Countries, target types, and weapon types are selected in the TEMP data base. The PK function also computes certain parameters which are used for static measures and it enters these in the PERM data base.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label.	A10
2	11-20	NAMFCT: 'PK'.	A10
3	21-30	FILE: Offset distance in nautical miles.	F10.0
4	31-40	NAMM: Altitude in feet.	F10.0
5	41-50	FNUM: Area of target in in nautical mile square.	F10.0
6	51-60	NAMPRM: 'WEIGHTED' means an exponential distribution of weights over evaluation points is applied.	A10

Example

```
LINMIX****PK
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

9. 'LP PREP' Function

This function produces a linear programming (LP) problem in standard mathematical programming (MPS) format. The LP problem in MPS format is written on TAPE2 by default. The parameter FILE on the function card may direct this output to another file. A weapon order set has been added and is required as the second card set. A BOUNDS set is also required. Bounds for binary variables used for buyin costs are produced automatically. Additional bounds may be input in MPS format. An END card is required.

Function Card, 'LP PREP'.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label.	A10
2	11-20	NAMFCT: 'LP PREP ', prepare the LP problem.	A10
3	21-30	FILE: The file number for MPS output.	F10.0
4	31-40	NAMM: Name of the problem.	A10
5	41-50	FNUM: Not used.	F10.0
6	51-60	NAMPRM: The name of RHS set.	A10
7	61-70	NOTE1: The name of bounds set.	A10
8	71-80	NOTE2:The name of ranges set.	A10

Card Set 1, Weapon Order Cards (required).

Field	Column	Contents	Format
1	1-10	Optional label.	A10
2	11-80	IKL(IY): Weapon number codes to select weapon types and the order of weapons in this problem.	35A2

Any number of weapons cards may be used. 'END' in column 1-3 will end the weapon order card set. An end is required.

Card Set 2, Bounds Cards.

An end bounds card is required.

Field	Column	Contents	Format
1	1-10	BND(1): 'END' in columns 1-3 will end the bounds cards.	A10
2-8	11-80	BND(2-8): Must be in MPS Format.	7A10

MPS Format For Bounds

Field	Column	Contents	Format
1	2,3	Type code of variable.	1XA2,1X
2	5-14	Name of bounds set.	A10
3	15-24	Name of variable.	A10
4	25-36	Value of bound. A decimal is inserted after the last digit.	F12.0

Example

```
LINMIX****LP PREP   2.        TESTI01             RHS01     BO
WEP ORDER 0102040506202203
END ORDER
ENDBOUNDS
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

The LP PREP function card goes beyond 60 columns. The last 60 columns are shown below for this example.

```
2.        TESTI01             RHS01     BOUNF01   RANGE01
*         *         *         *         *         *         +
1 Column 31         41        51        61        71        80
```

10. 'ASSEMBLE' Function

The LP PREP function, which has just been described, produces a complete LP problem for a given time period. In order to produce semidynamic LP problems over several time periods these separate problems must be combined into a single problem. Many important additions will then be necessary to produce a logical complete semidynamic problem. Much of this is produced in the revision deck produced by the next procedure called SYSTEMTREE.

Function Card, 'ASSEMBLE'.

Problems for several time periods are assembled into one MPS formated problem deck. This function uses the scratch file and the standard MPS file containing one problem deck for each time period.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label.	A10
2	11-20	NAMFCT: 'ASSEMBLE '.	A10

Example

```
LINMIX****ASSEMBLE
*        *        *        *        *        *        *
1 Column 11       21       31       41       51       61
```

11. 'SYSTEMTREE' Function

This is the components cost procedure which connects weapon systems to their required component subsystems. In a semidynamic problem this function may be used not only to reflect costs, but to reflect lead times and to relate requirements of problems generated for several time periods.

Function Card, 'SYSTEMTREE'.

This is a new function called by LINMIX. The function card has the same format as other function cards. SYSTEMTREE is called to produce a revision deck on the file called COMTRE.

Field	Column	Contents	Format
1	1-10	NAMOPT: Optional label.	A10
2	11-20	NAMFCT: 'SYSTEMTREE', prepare revision.	A10
3	21-30	FILE: Number of time periods.	F10.0
4	31-40	NAMM: Name of this revision deck.	A10
5	41-50	FNUM: Number of weapon types initialized. If FNUM=0, do not initialize.	F10.0
6	51-60	Not used.	A10
7	61-70	NOTE1: The name of bounds set.	A10

Example

```
LINMIX****SYSTEMTREE 3.        COMTRE    22.                 BOUND01
*         *         *          *         *         *         *
1 Column  11        21         31        41        51        61
```

There are six input card sets for the SYSTEMTREE function. The data in each section conforms to the general 10 column format with an END card for each section. In the following section for example the first ten columns are used for optional fields, the end designator, and the weapon system number. The weapon system number is usually written as 'X01' for example. The first column is ignored and the weapon type number is in columns 2 and 3. The weapon number must not exceed the number of weapons initialized (FNUM above). It may be the same as the PERM data base number or it may use a different weapon from PERM in each time period.

In columns 11-80 there are 7 ten column fields. These fields are divided into subfields to allow for interpretation of input. For example the first component may be written 'P01A' beginning in column 11 and the number of weapon systems per component may follow anywhere in the next 6 columns with or without a decimal.

The card sets for SYSTEMTREE must be in the order given.

Set 1, Components Section.

The components of each weapon system are input on one or more cards. The letter input in column 11 indicates the hierachical class to which a component belongs. Within that class the weapon or component is indicated by a 2 digit number. Subsets of types of components called mods are indicated by an alphanumeric suffix. These four character combinations indicate the components required for each weapon system. The weapon systems must not use more than one of each class of component, indicated by the first letter. The number of weapon systems per component is also entered. For example, the number of boosters on a submarine is input.

Card Set 1, Components Cards.

Field	Column	Contents	Format
1	1	NAMOPT: Optional field.	1X
2	2,3	I: Weapon number. 'END' in column 1-3 will end the section.	I1
3	4-10	NAVEND: Optional field.	A7
4	11	LTREE(J): Letter used as name of the class of the component.	A1
5	12,13	LA(1,J,I): Type of component (numeric).	A2
6	14	LA(2,J,I): Mod of component (alphabetic).	A1
7	15-20	LA(3,J,I):Number of weapon systems per component. If blank, 1.0 is assumed.	A6
8-11	21-30	LTREE(J),LA(1-3,J,I): Second component for I.	A1,A2,A1,A6
.	.	.	.
.	.	.	.
.	.	.	.

Up to 7 components for one weapon system type may go on each card. Any number of cards may be used to input up to 40 components per weapon type. Blank fields are ignored.

EXAMPLE

```
X01       P01A 16   C03A 1
X02       P01B 16   B04A 1
X03       T01A 24   B04A 1
X22       T01A 24   B04A 1    A01A
X04       T01B 24   B05A 1    A01A 1.
X05       T01B 24   B051 1    A01A 1.
X07       I07A 1.
X08       F01A 20.
END COMPONENTS
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Set 2, Levels of Component Tree.

The minimum level is level 1. Components of a weapon system are not connected to a common node for that component.

Level 2 connects the weapon system level to a components level. Attrition may flow out of the level 2 nodes.

Level 3 connects components that have the same number but may differ in the mode (suffix letter). Attrition is possible from level 3.

Level 4 connects all components belonging to a given class. This is the level for a complete tree, while the other levels are truncated.

Card Set 2, Levels Cards.

Field	Column	Contents	Format
1	1-3	NA: Optional field. 'END' will end the section.	A3
2	4-10	Optional field.	A7
3	11	LL(1,1): Name of class of component.	A1
4	12	LL(2,1): Level of class. Default = 4.	I1,8X
5	21	LL(1,2): Name of class.	A1
6	22	LL(2,2): Level of class.	I1,8X
.	.	.	.
.	.	.	.
.	.	.	.

Up to 7 levels may be entered per card. Only one is needed for each class. Blank fields are ignored.

Example

```
LEVELS    P2        T4        B4        C2        A4        I3
END LEVELS
*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Set 3, Links to PERM DB.

Field	Column	Contents	Format
1	1	NAMOPT: Optional field.	1X
2	2,3	I: Weapon number. Column 1-3 = 'END' will end the section.	I2
3	4-10	NAVEND: Optional field.	A7
4	11-20	LB(1,I): Name of variable for number of weapon systems in the time period 1, such as 'B1BX01' for LB(1,1).	A10
5	21-30	LB(2,I): The variable for time period 2.	A10
6	31-40	LB(3,I): The variable for time period 3.	A10
.	.	.	.
.	.	.	.
.	.	.	.

Up to seven time periods may go on a card. One card is used per weapon system unless 8 or 9 periods are involved and then columns 11-20 and 21-30 on a second card are used for each weapon system.

Example

```
X01       B1BX01    B2BX23    B3BX45
X02       B1BX02    B2BX24    B3BX46
X03       B1BX03    B2BX25    B3BX47
X06       B1BX22    B2BX44    B3BX66
X04       B1BX04    B2BX26    B3BX48
X05       B1BX05    B2BX27    B3BX49
X07       B1BX12    B2BX12    B3BX56
X08       B1BX19    B2BX41    B3BX63
END LINKS
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Set 4, Conversion Arcs Section.

This set creates arcs between nodes at level 2 or between nodes at level 3. It corresponds to conversion between related mods of a given component type or conversion between different types belonging to the same class.

Card Set 4, Conversion Arcs Cards.

Field	Column	Contents	Format
1	1-3	NA: Optional field. 'END' will end the section.	A3
2	4-10	Optional field.	A7
3	11	'C'	1X
4	12	LC(1,I,K): Time period number. If this is blank or -, then the arc will be generated for each time period.	I1
5	13-16	LC(2,I,K): Root word for the arc. Level 2 arc between P1XV01A and P1XV01B has root word XV01. Level 3 arc between P1XV01 and P1XV02 has root XV.	A4
6	17,18	LC(3,I,K): Suffix for origin node of arc. Level 2 suffixes must be left justified.	A2
7	19,20	LC(4,I,K): Suffix for terminal node. Level 2 suffixes must be left justified.	A2
8-12	21-30	Repeat the fields 3-7 for the second arc.	1XI1,A4,2A2
⋮	⋮	⋮	⋮

Up to 7 arcs may be on each card. Blank fields will be ignored. The number of arcs should not exceed the number that can be read on the number FNUM cards. For 22 weapons or FNUM=22, up to 154 arcs may be input.

Example

```
CONVERT   C-XP01A B C-XT01A B
END ARCS
*         *         *         *         *         *         *
1 Column 11         21        31        41        51        61
```

Set 5, Bounds Option Section.

This section selects the column vectors which will be fixed at zero. Four types of bounds may be selected: attrition arcs (A), intermediate nodes (P), outflow arcs (O), or total number of components (S). S nodes and P nodes for time period zero are selected by default if not suppressed. Time period, class of component and level of component tree to be bounded are each variables. Only one set may be specified on each bounds card.

Level 1 is only used for S-type bounds. Level 4 is used only for types A and P. Time period zero is only for S and P.

Card Set 5, Bounds Cards.

Field	Column	Contents	Format
1	1-3	NA: Optional field. 'END' will end the section.	A3
2	4-10	Optional field.	A7
3	11	LX(1): Type of bound = A, P, O, S. If type of bound is omitted, all are included.	A1
4	12	LX(2): Time period number. If time period is omitted, all are included.	A1
5	13	LX(3): 'X'.	A1
6	14	LX(4): Letter used as the class of component. If omitted, all are included.	A1

Field	Column	Contents	Format
7	15	LX(5): Level to be bounded. Only certain levels are used for given types of nodes and arcs. If this field is omitted, all levels applicable are included.	A1
8	16-20	Not used	4A1,1X
9	21-30	NOTBND: If this field is blank then the bounds set will be produced. Otherwise the bounds set will be omitted.	A10
10	31-80	Optional comments.	5A10

Examples		Explanation
BOUND *** A1XP2		Selects A-type arcs, the time period 1 , Class P, Level 2. such as A1XP01A or A1XP99M.
BOUND *** P*X**		Selects all type P bounds.
BOUND *** P2X**	NO BOUND	Drops the time period 2 bounds of type P.
BOUND *** P*XV*	NO BOUND	Drops the P*XV bounds of any level and any time period.
BOUND *** P2XV2		Includes level 2 bounds such as P2XV04D.
BOUND *** P2XV3		Includes level 3 bounds such as P2VX04.
BOUND *** P2XV4		Includes level 4 bounds such as P2XV.
BOUND *** P2XV*		Includes all three levels.
END BOUNDS		

```
*        *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Set 6, Cost Vectors or Additional Rows

One or more cost rows may be entered. If noe are entered the section is terminated by an ENDCOST card. If more than one cost row is used the word 'NEXT' or 'NEXT COST' is used to separate rows using previously defined columns or by adding new columns.

Card Set 6, Card 1, Name of Cost Card.

Field	Column	Contents	Format
1	1-10	NACOST: Name or cost row.	A10
2	11-20	KIND: Kind of row, E, G, L, or N. (Default = N).	A10
3	21-30	LD(1,J): Column name for J=1 .	A10
4	31-40	LD(2,J): Coefficient of column variable.	A10
5	41-50	LD(1,J): Column name for J=2 .	A10
6	51-60	LD(2,J): Coefficient of column variable.	A10
7	61-70	LD(1,J): Column name for J=3 .	A10
8	71-80	LD(2,J): Coefficient of column variable.	A10

Blank column names are ignored. Any number of continuation cards may be used. The end of a row may be followed by a name of cost card. 'NEXT' or 'NEXT COST' in a name field signals the end of a row.

Card Set 6, Card 2, ENDCOST Card.

Field	Column	Contents	Format
1	1-10	LD(1,J): 'ENDCOST'.	A10

Example

```
ENDCOST
*         *         *         *         *         *         *
1 Column 11        21        31        41        51        61
```

Example of A Complete Input Deck for SYSTEMTREE

The card sets of SYSTEMTREE must be in the order given.

```
LINMIX****SYSTEMTREE 3.       COMTRE    22.                           BOUND01
X01       P01A 16   C03A 1
X02       P01B 16   B04A 1
X03       T01A 24   B04A 1
X06       T01A 24   B04A 1    A01A
X04       T01B 24   B05A 1    A01A 1.
X05       T01B 24   B05A 1    A01A 1.
X07       I07A 1.
X08       F01A 20.
END COMPONENTS
LEVELS    P2        T4        B4        C2        A4        I3        F2
END LEVELS
X01       B1BX01    B2BX23    B3BX45
X02       B1BX02    B2BX24    B3BX46
X03       B1BX03    B2BX25    B3BX47
X06       B1BX22    B2BX44    B3BX66
X04       B1BX04    B2BX26    B3BX48
X05       B1BX05    B2BX27    B3BX49
X07       B1BX12    B2BX34    B3BX56
X08       B1BX19    B2BX41    B3BX63
END LINKS
CONVERT   C-XP01A B C-XT01A B
END ARCS
BOUNDS    A
BOUNDS    P
END BOUNDS
ENDCOST

          *         *         *         *         *         *         *
1 Column 11        21        31        41        51        61        71
```