NOT FOR QUOTATION
WITHOUT PERMISSION
OF THE AUTHOR

AN INTERACTIVE COMPUTER MODEL FOR LAND ALLOCATION IN REGIONAL PLANNING

PART II: SYSTEM DESIGN AND USER MANUAL

Geoffrey G. Roy*

April 1981
WP-81-47

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
A-2361 Laxenburg, Austria

PREFACE

Since 1979, the Regional Development Task at IIASA is engaged in a case study of economic and demographic development, land-use and related problems in the region of southwestern Skane in Sweden. The case study is the third in a series of attempts made by the Regional Development Task to apply systems analytic methods to regional planning problems in regions with different economic structures, resource endowments and organizational settings.

The research in the Swedish case study is done in collaboration with the Southwest Skane Municipal Board, as a part of their ongoing work in physical and public transport planning for the metropolitan region of Malmo, and its neighboring municipalities. The research is partly sponsored by the Swedish Council for Building Research.

In the case study an integrated systems analytic package of models is used which has been developed within the Regional Development Task in cooperation with a group of Swedish researchers and planners. In that package, separate models have been developed for interregional economic and demographic problems, and for intraregional land-use problems.

The current paper deals with an interactive computer model developed by Geoffrey G. Roy, University of Western Australia, and Folke Snickars, Regional Development Task, IIASA. The theoretical foundation and specification of that model is found in the first of a series of three papers. The current, second paper describes the design of the computer system and also provides a user manual. A third paper will describe the use of the model in the Skane case study.

The interactive computer model described here has been implemented in the planning environment in southwestern Skane. It should also prove useful in other planning contexts.

Boris Issaev
Regional Development Task
IIASA, Laxenburg
February 1981

TABLE OF CONTENTS

AN INTERACTIVE COMPUTER MODEL FOR LAND ALLOCATION IN REGIONAL PLANNING

PART II: SYSTEM DESIGN AND USER MANUAL

Geoffrey G. Roy

1. INTRODUCTION

ISP is an interactive procedure designed to assist planners in the solution of land use planning problems. It is based on the proposition of a two level hierarchy: the macro level at which the user allocates and manipulates areas of land and the micro level which includes the detailed definition of the problem into production sectors and subregions. A detailed description of this model is given in Roy and Snickars (1981a).

The successful application of ISP depends on a number of principle assumptions, viz:

(a) That planning objectives can, in general, be formally defined and evaluated numerically. It is important that the user should understand the significance of these objectives as he must be able to make subjective evaluations of the objectives during the interactive sessions. It is not necessary to assume that all planning objectives must be defined explicitly providing the user is satisfied that sufficient information is available to him to make whatever subjective evaluation he feels necessary.

(b) That the user is mainly interested in allocating land (at the macro level) to subsets of aggregated production sectors over some sets of aggregated subregions. The level of aggregation is defined when the problem is structured, and the data prepared, and can range from little (or none) to high levels of aggregation. No aggregation is done across time.

(c) That the user is also interested in manipulating some characteristics of the system at the micro level by imposing (interactively) sets of special constraints.

(d) That, unless otherwise required by the user or the apriori defined production goals, the current state of the system represents some state of "equilibrium" and that radical changes are not to be expected.

ISP must be configured for each planning problem. In addition to the required data files, two user written subroutines must be supplied. These routines are required to compute the values of the objective functions as required by the main program.

The program is designed to be operated on a graphics terminal (Tektronix 4010/12/14/15/16 currently), but will operate on a conventional printing or VDU terminal, without graphic output. Since ISP is designed to take advantage of graphic displays to concisely present large amounts of information, it will loose some of its appeal if operated in a non-graphic environment.

Although it is intended that ISP should be used by persons without "computer or programming expertise", the successful installation of the program would require such skills. It should also be mentioned, that since ISP is driven by "user commands" a prospective user should endeavor to learn something of the scope of the commands available before starting his first interactive session.

2. OPERATIONAL OVERVIEWS

2.1 Problem Size

The maximum size of any problem which can be handled depends on the availability of computer memory and suitable dimensioning of arrays in the program. Array dimensioning is discussed in more detail in Appendix D. Currently, however, the program is dimensioned for a problem of 30 production sectors in 40 subregions across 10 time periods (at the micro level). At the macro level, 10 sectors in 10 regions are allowed and up to 20 objectives can be included. These later restrictions are governed not only by the available computing facilities but also the ability of the user to comprehend the macro state. It is suggested that a proposal to include more than 10 sectors and/or regions at the macro level should be considered most carefully.

On the VAX running UNIX at IIASA, ISP requires approximately 275,000 bytes of memory for the above size restrictions. The ancillary reporting program (ISPREP) requires 152,000 bytes.

2.2 Data and other disk files

Several data files are normally required by ISP. These files must be available to ISP on the user's disk area. Some programming changes will generally be necessary to the definition of these files, if the program is to be implemented on a different computer system. These data files are:

(a) Problem configuration data file containing problem definition and objective function data (essential)

(b) map data file containing the geometric data to define a map of the region and subregions, and to locate centres of pie charts (required if graphic terminal specified),

(c) "results" file contains complete set of results for all trial solutions in sequential order. This file will always exist except when establishing the first trial solution or when the user wishes to start afresh with a totally new first trial (normally essential),

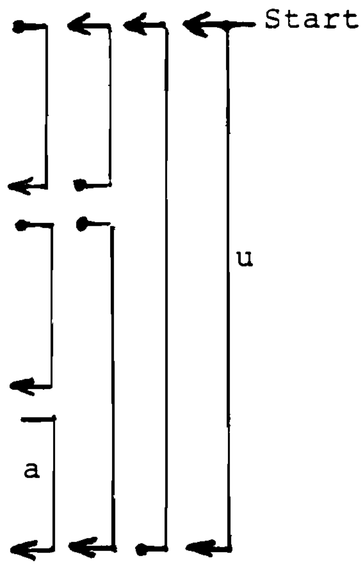
(d) special constraints file contains the complete description of any set of special constraints defined by the user. If the user wishes to save a set of special constraints for a future interactive session he can create a file on the disk which can be read back as required. Several such files, may exist as the user must define a suitable name for each file when he saves it (optional).

2.3 Operation Sequence

The operation of ISP is divided into five phases (see Fig. 1)

Program ISP

Phase 1 : Adjusting macro allocation of land (and creating first trial allocation)
Phase 2 : Adding/modifying/deleting special constraints
Phase 3 : Solving for new micro allocation at each time period
Phase 4 : Examining results



Program ISPREP

Phase 5 : Preparing detailed hard copy reports of selected trial solutions

Figure 1. Operational Phases of ISP

Phase 1 to 4 contain the complete ISP model and Phase 5 is a separate program (ISPREP) to prepare detailed reports of selected trial solutions. The arrows in Figure 1 indicate the allowable moves between the first four phases. In most cases the moves are made at the users command. There are two exceptions. The move from Phase 3 to 4 (marked "a") is automatic once Phase 3 is complete. Also, it is usual from the "start" to go straight to Phase 4 (marked "u") once a first trial solution has been established and a "results" file already exists.

In normal circumstances the user will begin an interactive session starting from the last solution generated at the previous session. The program immediately enters Phase 4 to allow the user to examine any of his previous trials contained in the "results" file. If desired, the user can use ISP only to examine previous solutions. There is no requirement to create new solutions.

2.4 Reliability

Except for as yet undiscovered "bugs", ISP should produce predictable results. There is, however, one problem which may occur if the user attempts to solve an infeasible problem (for example, by imposing special constraints which violate the apriori defined production and land area constraints). In this case the algorithm used to solve the maximum entropy problem may fail. The various failure modes are discussed later. In some cases the failure is "soft", i.e. the convergence tolerances were not quite met and in other cases the failure is "hard". In this latter case the user should <u>not</u> save this solution (as it is likely to be garbage) but he should return to Phase 1 and Phase 2 to make suitable adjustments to his scheme. It is also possible that the failure is so "hard" that the program "crashes" and must be started again.

2.5 Convergence

As the user develops more trial solutions based on the results of previous solutions, he will gradually build up a set of non-dominated solutions. It is not envisaged that the user will be always able to select from this set a single "best" solution. He should, however, be able to select a set of "good" solutions which he will need examine in more detail before a final ranking is proposed.

There are no specific aids to convergence and hence convergence as such cannot be guaranteed. It is assumed that the user will learn something of the behaviour of his planning problem and will be able to use his own experience and judgement to make adjustments to his allocations in an endeavor to find a better solution. It is envisaged that some degree of trial-and-error will be present during the first interactive sessions.

2.6 The Eriksson algorithm for solving the maximum entropy problem

The maximum entropy problem which is formulated as a means to "forecast" the most likely new micro state at each time increment is solved by the algorithm developed by Eriksson (1980). This algorithm is ideally suited to this form of problem and provides an efficient means of coping with large problems. The algorithm takes advantage of the often sparsness of the matrix of constraint coefficients and only the storage of non-zero elements is required. The algorithm is also quite fast, for example, taking less than 2 seconds (cpu time) to solve one time step for a problem with 26 activities and 35 subregions at the micro level, and 7 activities and 10 regions at the macro level on the VAX at IIASA.

There are, however, some problems with the algorithm, particularly when the user attempts to solve a highly constrained or infeasible problem. In the first case, the convergence tolerances may not be quite met, but most likely the results will be quite acceptable. In the second case, the algorithm should report the difficulty, but on occasions it may crash.

3. PROBLEM DEFINITION

3.1 Macro-micro structure

The macro structure of the land allocation problem is defined by two sets of parameters [igv(i), igr(j)] which associates each variable, and region, with a particular macro grouping. To be meaningful these groupings must be based on some inherent characteristics of the activity sectors and subregions respectively. For the case of activities, some common element should be present, e.g. all activities associated with agriculture may be logically grouped together. For the case of the spatial aggregation, political and social boundaries could be the major factors influencing the grouping of regions.

In certain circumstances, some activities may not be included at the macro level but still exist at the micro level. Activities which, for example, do not consume land may fall into this category. In such cases, these activities are given a macro grouping of "Ø".

It may also be desirable to exclude certain regions from the macro state. This is possible in two ways, depending on the desired purpose. Firstly, the user may be primarily interested in a subset of regions even though the apriori defined production goals are set for the region as a whole. It may be useful, therefore, to group all regions outside the set of interest and label them as "outside" regions (rout). If this is done the regions are still included for the purpose of allocating land but are excluded from the reporting phase where the macro state is reported. This is a useful feature since the data for the "outside" region may force the scaling of the graphical presentation to be too small as the program attempts to get all the map or graph on the screen of the graphic terminal.

It is also possible to exclude certain regions from the macro state entirely ("rext" regions) by defining their regional grouping to be "Ø". In this case, these external regions are still included in the micro state but the user is not required to make specific allocation of land at the macro level. It is not possible to directly check the consistency of the macro land allocation for total production goals so the user must do this himself. When type "Ø" regions are included, and the user is allocating land (see section 4.2), the production deviations (as shown in Fig.2) may be negative. The value of this negative number indicates the level of production required from all external regions to meet the production goals. The user must, with the knowledge of the current production, decide whether this figure is reasonable. If the user implies that too much production is to be achieved from the external regions, there may be insufficient land available, and an infeasible solution result.

3.2 Activity, region and time descriptors

Each activity and region (both at micro and macro levels) is defined by a four character descriptor. Throughout the operations of ISP, these descriptors are used to refer to the relevant element and hence some care should be exercised in selecting a suitable set of descriptors. Also, time periods are described by four digit year numbers.

The activity and region descriptions can be any four characters except for the reserved command words as listed below:

vall	rall	cont	back	list
rati	wipe	call	exit	sing
show	nond	rest	sele	over
rout	rext	save	read	disp
devi	skip			

All descriptors chosen by the user must be unique including macro and micro descriptors referring to the same activity or region. Distinctions may be made using upper/lower case characters.

3.3 Initial system state

The initial state of the region must be defined at the micro level in the same units chosen for each activity sector. The production level of each activity in each subregion must be defined. It is permissible to have zero elements in this array though the user should be aware of their significance. The maximum entropy algorithm used to solve the new micro allocations will, by its nature, force all activities to a zero level if their initial state is zero. Hence if the user wishes to introduce a new production sector into a region he must make a positive intervention to alter the initial state. This facility is available in Phase 2 where special constraints are imposed.

3.4 Production goals

For each production sector, the minimal levels to be satisfied across the whole region for each time period must be available. In general these levels are treated as lower bounds on production. It is however possible for the user to produce solutions which over satisfy the production goals, if he so wishes.

3.5 Available land areas

For each subregion the total land available for allocation at each time period must be determined. Mostly, we would expect that the land available will remain constant with time, but variation is permissible. In general it is assumed that any production sector can use any of the available land. Any limitations in this direction must be included (manually) by the user during the land allocation (Phase 1) and special constraint (Phase 2) phases. Also there is no general way of allowing for two or more production sectors occupying the same piece of land. It can be handled (in a limited way) by defining the "smaller" consumer of land to be "non-land consumptive" (see 3.6).

3.6 Production coefficients

The conversion from production units into land consumption is done by means of a "production coefficient" (i.e. the production per unit area of land). These coefficients must be defined for each activity in each subregion. The time varying characteristics for these coefficients is included by an exponential factor describing the rate of change (per year) of the coefficients in each sector. The coefficient should be evaluated for the time corresponding to the initial system state.

Certain activities may be "non-land consumptive", i.e. their production coefficient (as defined above) would approach infinity. To cope with this possibility a "land consumption" coefficient (Ø or 1) is used so that, the conversion from production units to land area will take the following form:

$$\text{land area} = \frac{\gamma_i \text{ (production level)}}{\lambda_{ij}\, e^{\mu_i (t-t_o)}} \qquad (1)$$

where λ_{ij} is the production for unit area of land for activity i in region j, μ_i is the rate of annual change for sector i (+ve for increases, -ve for decreases), t_o is the time of the initial state and γ_i is the (Ø/1) land consumption coefficient for activity i.

In certain situations, a particular activity (though included in the model) may have no influence and the production coefficient may be undefined. In this case a zero can be inserted providing that activity is also defined as being non-land consumptive (i.e. γ_i=Ø).

3.7 Structure of objectives

The defined planning objectives can take any form desired by the user provided they can be computed from a given system micro state and any required tables of user supplied coefficients. The user must provide two subroutines to be compiled with ISP. The first will read from the data file the required data to define each objective, the second will compute the objectives as required by ISP (see Section 6).

Objectives can be defined as being maximizing or minimizing and the "best" and "worse" (or upper and lower bounds) supplied for each objective. These bounds are required only for the graphical presentation of objectives and do not effect the numeric values. The "best" and "worse" values are taken to be constant over time.

The objectives defined for a planning problem are not aggregated over time, but different levels of spatial aggregation are included:

(a) micro objectives are the values of the objectives computed for each micro region
(b) macro objectives are those values computed for each macro region
(c) global objectives are those computed for the region as a whole or some set of regions.

The values of the objectives at each of these levels are available to the user, but it is the "global" objectives which are intended for first consideration. It is these objectives which are used to check for non-dominated solutions and for a graphical pair-wise comparison between different trials.

Objectives have to be defined as being "active" or "inactive". Active ojectives are used for computing non-dominance and displaying (graphically) comparisons between pairs of solutions. In all other respects, active and inactive objectives are treated identically. The complete set of objectives should be viewed as performance indicators which would, or could, be used somewhat selectively by any particular user. The "active" objectives can be redefined (interactively) by a user and hence may be viewed as a more importance sub-set of indicators. Comparisons between alternative solutions can hence be made using different set of active objectives. A maximum of 5 objectives can be active at any one time.

3.8 Map of region

When using a graphic terminal, a part of the graphic display options uses a map of the region to display the spatial distribution of the land allocations. The data for this map defines the boundaries of each region by sets of x,y coordinates (see Section 5 for details). Also, the location of the "pie" charts is chosen manually to provide for minimum interference and maximum readability.

It would be usual to "stylize" the actual map to some degree (to reduce the data requirements and drawing time). Even so the use of a digitizer is one way of reducing the work necessary to prepare a large set of x,y coordinates. The numerical scale of the co-ordinate data is unimportant as it is automatically scaled to fit the largest map on the screen. Currently, the size of the pie charts is fixed, but this may require some modification for particular problems.

4. COMMAND DESCRIPTIONS

4.1 Introduction

ISP is driven totally (virtually) by user commands and hence the effectiveness of the program will depend to a large extent on the users knowledge of the commands available. The commands are issued to ISP following the prompt (">") which appears when the previous command has been executed. In some situations other prompts are used, as detailed in the following sections.

Commands often have several arguments separated by commas. The arguments are generally four characters long (these are sometimes extended to six characters) or just a single character. Numeric arguments can be from 1 to 7 digits (including the decimal point which is only required if a fractional value is to be inserted). All commands are terminated with a "carriage return".

Many have optional and/or alternative arguments wich are detailed in the following sections. In the command descriptions that follow, square brackets ([.]) are used to indicate optional arguments and curly brackets ({.}) indicate alternative arguments. Underlined characters (e.g. prompts) are those output by the program. In the command examples, the following arguments have particular meanings.

vnam	-	means any valid micro activity name
rnam	-	means any valid micro subregion name
vmac	-	means any valid macro activity name
rmac	-	means any valid macro subregion name
rout*	-	refers to those regions "outside" the macro state
rext*	-	refers to those regions "external" to the macro state
vall*	-	refers to all micro variables
rall*	-	refers to all micro regions
time	-	means any valid year label
timeø	-	means the time of the initial state
øøøø*	-	means all time periods
value	-	means any (1-7 digit) numeric value
k,k1,k2	-	means an integer value
file	-	refers to a valid disk file name (up to 10 characters)

Those arguments above with an "*" have special significance for ISP and can be used (when permitted) to express the appropriate command.

It is important to note that all map drawing and graph plotting commands are affected by the current state of the "display" table as decribed in Section 4.5.11.

4.2 Phase 1 commands: macro allocation of land

4.2.1 General

This group of commands allows the user to examine and modify a land use allocation at the macro level. They can also be used to construct the first trial allocation. If no "results" file exists, ISP enters Phase 1 automatically and requests the user to input a first trial allocation for each time period.

The available commands enable the user to modify any element of his allocation, examine his allocation, plot maps showing the spatial distribution of his allocation by means of pie charts on a map of the region and plot graphs showing the variations with time of his allocations. Throughout this phase a ":" prompt is used to remind the user, that the first argument of all commands is a single character. When the user is inserting a new first trial allocation, the program requires that the data be typed in, in the same order as displayed in Figure 2. For each macro region, the program prompts the user by printing the line containing the allocations at the previous time step. Under this line the user must type in his new allocation. Each numeric value should be typed directly under the corresponding one above as shown in the "<...>" brackets. The number can in fact be anywhere in the field defined by the "<...>", any number of spaces can be inserted between the numbers.

4.2.2 The "c" command

Once an allocation of land is complete or the user has completed some desired modifications the "c" command instructs ISP to continue on to Phase 2 (special constraint phase), e.g.

```
:c
```

4.2.3 The "d" command

Given a complete allocation at any time period (usually exists except when part way through inserting a first trial allocation) the "d" command prints a table showing the macro allocation of land at the current time period (see Sections 4.2.5 and 4.2.6). The Figure 2 shows an example of this table. Included in the table are the production deviations, land area deviations and the allocation at the previous time step.

The production deviation (shown along the second top line) indicates by how much (in production units) the given allocation of land would over satisfy the production goals. These deviations should never be negative. (except if type 'Ø' regions exist, see section 3.1)

The land area deviations (shown down the left hand side under each macro region name) indicate the net unused land remaining in each macro region. These deviations should never be positive.

The values in "<...>" above each allocation are the corresponding values at the previous time step. In the first instance the initial system state values are displayed, e.g.

:d

Time =1980

	agrb	indb	eneb	serb	houb	govb	slab
	326*	324*	18*	1072*	7285*	2066*	2*
kavk ---- <	11931><	167><	250><	16><	743><	43><	906><
(-1838)	11454	169	256	17	8C4	41	907
lomk ---- <	4105><	22><	6><	12><	439><	32><	468><
(-504)	3962	18	6	13	485	38	469
lunk ---- <	3C436><	228><	43><	96><	1874><	729><	938><
(-10410)	29893	223	45	98	1927	840	938
stak ---- <	9395><	81><	3><	20><	469><	44><	156><
(-1212)	8942	84	3	21	489	64	156
burk ---- <	1C22><	153><	18><	47><	212><	25><	74><
(-348)	909	163	19	52	212	31	75
malk ---- <	8343><	1296><	118><	836><	2226><	551><	1119><
(-3190)	7118	1183	121	845	2138	625	1120
svek ---- <	18471><	32><	6><	1334><	521><	24><	106><
(-5283)	16667	33	7	1436	629	31	106
velk ---- <	9553><	21><	12><	16><	917><	56><	2038><
(-1623)	9574	22	13	15	988	67	2039
trek ---- <	29816><	270><	24><	97><	965><	70><	1481><
(-2990)	28523	254	26	112	959	79	1481
rout ----	<756731><	3582><	275><	1595><	23362><	756><	15758><
(-161894)	762857	3561	279	1625	24115	950	15758

:

Figure 2. Example table of macro allocation

4.2.4 The "v" command

Except when inserting a new first trial macro allocation (when it is done automatically) the user must validate his allocation after making changes. A series of consistency checks are necessary to ensure that sufficient land is available and that the regional production goals can be met from the trial macro allocation of land. The checks inserted are not infallible but they will pick up obvious errors. Checks are made to ensure that the production deviations are non-negative and the land area deviations are all non-positive. If any type Ø (i.e. external) regions exist the checks on the production levels are not made and it is assumed (at this stage) that any shortfall in production can be made up from activities in the "external" regions, e.g.

:v

If inconsistencies exist, the user is informed and an indication given of the amount of adjustments required to remove the problem.

4.2.5 The "n" command

When Phase 1 is first entered, the current time period is set at the first specified time period. The "n" command enables the user to step on to the next time period. During the input of a first trial solution, the "n" command will instruct ISP that the user is ready to input the allocation for the next time period. The "n" command has no effect once the final time period is reached, e.g.

:n

4.2.6 The "b" command

At any time the user can step back to previous periods with the "b" command. The command is ineffective once the first time period is reached, e.g.

:b

4.2.7 The "g" command

The "g" command allows the user to examine the macro allocation by means of graphical displays, e.g.

(a) :g

This command will produce a map of the set of macro regions showing (by means of pie charts) the proportion of land allocated to each macro activity at the currently selected time. This map excludes "outside" (rout) regions, e.g.

(b) :g, vmac

This command produces a graph showing the variation of the land allocation with time for macro variable "vmac" in all macro regions (excluding "rout" regions), e.g.

(c) :g, rmac

This command produces a graph showing the variation of the land allocation in macro region "rmac" for all macro activities, with time.

In each of the above two cases (b and c), the screen is erased and a new graph drawn. A new graph can be drawn on the existing one (without erasing) by the following command:

$$:g, \left\{ \begin{matrix} rmac \\ vmac \end{matrix} \right\}, over$$

In this case the scale used for plotting the first graph is used for the second hence the user should plot the graph with the larger (numerically) data values first.

The general form of the "g" command is represented by:

$$:g \left[, \left\{ \begin{matrix} \text{rmac} \\ \text{vmac} \end{matrix} \right\} \right] \left[, \text{over} \right]$$

For problems where the above map and graph may appear cluttered and difficult to interpret if all macro variables and regions are plotted the user can modify the display table in Phase 3 (see Section 4.5.11).

4.2.8 The "m" command

The "m" command allows the user to modify any element of his current macro allocation. The modification is carried out at the currently selected time period (which can be changed with the "n" or "b" commands), e.g.

```
:m, vmac, rmac, value
```

This command replaces the existing value of macro variable "vmac" in macro region "rmac" with "value". Following any set of changes to the macro allocation the "v" command must be used to check its consistency. Failure to do so may mean that the user attempts to solve and infeasible problem.

4.3 Phase 2 commands: Special constant phase

4.3.1 General

During this phase the user can insert a series of special constraints into the land allocation problem. There are several alternatives available:

(a) Single variable constraints at the micro level of the form:

$$x_{ijk} \left\{ \begin{matrix} \leq \\ = \\ \geq \end{matrix} \right\} \text{const.}$$

(b) Ratio constraints between pairs of micro variables in the form:

$$x_{ijk}/x_{lmk} \left\{ \begin{matrix} \leq \\ = \\ \geq \end{matrix} \right\} \text{const.}$$

(c) Modification of initial system state, e.g. to replace a zero element with a non-zero value so that a new sector can appear or to make a value zero to force a sector to dissappear.

(d) Modify macro land constraints from an (the default) upper bound form to equality or lower bound form.

Commands also exist for listing and modifying particular special constraints. The user is also able to save (permanently) on disk his set of special constraints for use in future interactive sessions.

In this phase a ">" prompt is used and the first argument is always four characters long.

4.3.2 The "cont" command

Once all special constraints have been inserted and/or modified, or if no special constraints are required, the "cont" command takes the user on to Phase 3 (solution phase), e.g.

```
> cont
```

4.3.3 The "back" command

Once in Phase 2, the user can go back to Phase 1 at any time with the "back" command, e.g.

```
≥ back
```

4.3.4 The "list" command

The "list" command allows the user to selectively list the special constraints currently imposed, e.g.

```
≥ list
```

This command will list all single variable constraints. The general form is:

$$\geq \text{list}, \left[\begin{Bmatrix} \text{vnam} \\ \text{rnam} \\ \text{rati} \\ \text{land} \end{Bmatrix}\right], \left[\text{time}\right]$$

The single variable constraints can be listed more selectively by specifying matching activity names ("vnam") or region names ("rnam") and "time" periods. Ratio constraints are selected with the "rati" argument and the macro land allocation constraints with the "land" argument. In these last two cases the "time" argument is not active. Only those macro land constraints which have been changed from the default upper bound form are included in the list.

4.3.5 The "wipe" command

This command enables the user to remove all or selected sets of special constraints. It obviously should be used with caution and to be effective the user is required to repeat the first argument upon request. If "wipe" is not typed a second time no action is taken. The general form is:

```
> wipe, { sing }
        { rati }
        { land }
        { call }
```

"sing" removes only single variable constraints, "rati" only ratio constraints, "land" restores all macro land constraints to the default upper bounds, "call" completes <u>all</u> three operations.

4.3.6 The "save" command

The "save" command allows the user to save on a disk file the complete set of special constraints, including modification to the initial system state. The user is required to specify a valid file name for the disk file (up to 10 characters), hence several different sets of special constraints can be saved. The general form of the command is:

```
> save
File name:     file
```

4.3.7 The "read" command

At any time during Phase 2, the user can replace the current set of special constraints with one from a disk file using the "read" command:

```
> read
File name:     file
```

A user beginning an interactive session may well use this command to restore the set of special constraints current during his previous session.

4.3.8 To insert/modify single variable constraints

Any command that begins with a variable name is taken as being an instruction to insert or modify a special constraint. For the following commands, the relational indicators (<,=,>,*) are used to define the form of constraint desired. "<" implies ≤, ">" implies ≥ and "*" makes a constraint inactive.

To insert a single variable constraint, the following command form is used.

$$\geq \left\{\begin{matrix}\text{vnam}\\ \text{vall}\end{matrix}\right\}, \left\{\begin{matrix}\text{rnam}\\ \text{rall}\end{matrix}\right\}, \left\{\begin{matrix}\text{time}\\ \text{ØØØØ}\end{matrix}\right\}, \left\{\begin{matrix}<\\ =\\ >\\ *\end{matrix}\right\} \text{value}$$

Note that the use of the general arguments "vall", "rall" and ØØØØ include suites of constraints. Also, the relational indicator is attached to the "value" hence no comma is inserted between the indicator and "value".

If a set of constraints exist, the user can modify them (i.e. modify the relational indicator and/or the right hand "value" by the command

$$\geq \left\{\begin{matrix}\text{vnam}\\ \text{vall}\end{matrix}\right\}, \left\{\begin{matrix}\text{rnam}\\ \text{rall}\end{matrix}\right\}, \left\{\begin{matrix}\text{time}\\ \text{ØØØØ}\end{matrix}\right\}, \left\{\begin{matrix}<\\ =\\ >\\ *\end{matrix}\right\} \text{value, r, k}$$

In this case the "r" is the instruction to replace the old relational indicator and right hand side with the new ones. If k is not included (or k=Ø), all matching constraints are modified. If $k \geq 1$ only the k^{th} occurance in the list of single variable constraints is modified.

In some situations the user may wish to force the "value" on the right hand side of the constraint to take the appropriate initial state value. To achieve this the following command is used.

$$\geq \left\{\begin{matrix}\text{vnam}\\\text{vall}\end{matrix}\right\}, \left\{\begin{matrix}\text{rnam}\\\text{rall}\end{matrix}\right\}, \left\{\begin{matrix}\text{time}\\\text{ØØØØ}\end{matrix}\right\}, \left\{\begin{matrix}<\\=\\>\\*\end{matrix}\right\} \text{Ø}, \text{ i}$$

In this case the right hand side value is taken from the initial state.

The user can examine elements in the initial state with the command

≥vnam, rnam, timeØ

following which the initial state value is printed. The initial state can be modified by the command

≥vnam, rnam, timeØ, = value, r

in which case "value" is inserted into the appropriate element of the initial state.

4.3.9 To insert/modify ratio constraints

The command structure to insert a ratio constraint is similar to that for the single variable constraints except that two lines are needed. The general form to insert a ratio constraint is:

```
≥  {vnam}  {rnam}  {time}
   {vall}, {rall}, {ØØØØ}, /
                           {<}
*  {vnam}  {rnam}  {time}  {=}
-  {vall}, {rall}, {ØØØØ}, {>} value
                           {*}
```

The first line contains the numerator definition and the second line the denominator definition together with the relational indicator and the right hand size "value". The "time" in the numerator must be the same as that in the denominator.

To modify existing ratio constraints the "r,k" arguments can be appended (see Section 4.3.8), thus:

```
≥  {vnam}  {rnam}  {time}
   {vall}, {rall}, {ØØØØ}, /
                           {<}
*  {vnam}  {rnam}  {time}  {=}
-  {vall}, {rall}, {ØØØØ}, {>} value, r, k
                           {*}
```

When using these commands the "/" followed by a "carriage-return" brings up the "*" prompt for the second line.

4.3.10 To modify macro land constraints

By default, all the constraints on the user defined macro land allocations are taken as upper bounds. The user is free to change any of those to either lower bound or equality constraints. He can also specify selected constraints to be inactive.

The following command is used:

$$\geq \text{vmac, rmac, } \left\{ \begin{matrix} \text{time} \\ \text{ØØØØ} \end{matrix} \right\}, \left\{ \begin{matrix} < \\ = \\ > \\ * \end{matrix} \right\}$$

This command replaces the current relational indicator. The right hand side value is not changed from that defined by the user in his macro allocation.

4.4 Phase 3 commands: Solution phase

4.4.1 General

Phase 3 is the phase where the new micro allocations are found to satisfy the user defined macro allocation and set of special constraints. For each time period a new solution is obtained, using the solution at the previous time step as a starting point. As each solution is found, a brief report is printed for the user, e.g.

```
Time = time    Iterations = k1    Ifail = k2
```

k1 was the number of iterations taken and k2 the code to indicate the success or failure of the algorithm. An Ifail = Ø means that the solution was successful within the specified tolerances, etc. Other possible Ifail codes are as follows:

- Ifail=4 the problem is inconsistent, most likely caused by the imposed special constraints violating the apriori defined production and land area constraints.
- Ifail=5 the specified accuracy not achieved, most likely resulting from a problem which is highly constrained. The solution could be acceptable.

If this Ifail code appears, a list of variables will be printed which do not satisfy the convergence tolerances together with the value and latest value change. The user can use these results to decide if the solution is acceptable.

Ifail=6 the specified maximum number of Newton iterations reached in the final step. Probably caused by an extremely tightly constrained problem, the results are likely to be unacceptable.

Ifail=7 the specified number of iterations reached (currently set at 1000). Solution may, or may not, be acceptable.

Ifail=9 an infeasible problem, the results will be garbage.

The occurence of other Ifail codes (e.g. 1,2,3 and 8) will indicate some problems with the basic data. If an Ifail code other than Ø occurs, the user is given the option to continue (save) the solution or to return to Phase 1. Some experience will be necessary with each problem before a user will have an idea of what "problem" solution can be saved. If the failure is so "hard" that the program "crashes" then the user will have to restart the program. Only results from the current trial will be lost.

The user should refer to Appendix C for a discussion on the setting of convergence tolerances as these can play an important role in the success rate for a particular problem.

4.4.2 The "cont" command

Once Phase 3 is entered the "cont" command instructs ISP to complete the solution of the latest trial, e.g.

```
≥ cont
```

4.4.3 The "skip" command

The user may decide that he wants to step on to Phase 4 (Reporting Phase) without solving the latest trial. The "skip" command will achieve this, e.g.

```
≥ skip
```

If this action is taken the current macro allocation may not be consistent with the latest micro solution. This can be corrected by the use of the "rest" command in Phase 4 (see Section 4.5.5).

4.5 Phase 4: Reporting Phase

4.5.1 General

The reporting phase provides a variety of facilities for the user to examine any of the solutions so far generated. If an existing "results" file is present when ISP is started, then Phase 4 is entered immediately. Phase 4 can be used to examine solutions without the need to generate new solutions.

Many of the commands available require the use of a graphic display, but in most cases a tabular report is given if operating in a non-graphic environment.

4.5.2 The "cont" command

The "cont" command enables the user to leave Phase 4 and return to Phase 1 so that adjustments can be made before developing a new trial solution, e.g.

```
≥ cont
```

4.5.3 The "exit" command

The "exit" command is used to terminate the execution of ISP and should only be used when the user wishes to conclude an interactive session, e.g.

```
≥ exit
```

When used, the user must confirm the command by re-typing "exit" a second time (upon request) before the command is executed.

4.5.4 The "select" command

By default, the trial solution selected for reporting is the last solution generated. To select any other trial the "select" command is used, e.g.

```
≥ select,[k]
```

where k is the desired trial number. If the "k" argument is left off, the currently selected solution number and the total number of trials so far generated is output.

4.5.5 The "rest" command

The user may wish to use some previous solution (other than the last one which is the default) as the basis for a new trial macro allocation. The "rest" command will replace the current macro allocation with that one from the currently selected solution (see "select" command).

```
≥ rest
```

When beginning a new interactive session the user can specify which trial solution he wishes to begin from. By default, the last solution the 'results' file is automatically chosen, but the user can use the 'select' command to nominate any other. In either case, the user must restore the corresponding macro allocation of land.

4.5.6 The "nond" command

Once the user has found two or more solutions he may well wish to compare the set of solutions to test for dominance. The "nond" command will compare all solutions, at the global objectives level, and report those which are non-dominated, e.g.

```
≥ nond
```

Not all objectives are tested for non-dominance, only those defined as being active (see Section 4.5.7).

4.5.7 The "show" command

The "show" command is used to display (in various ways) the values of the defined planning objectives:

(a)

```
≥ show
```

will give a brief report of the global objectives for the currently selected solution

(b)

$$\geq \text{show}, \left\{\begin{matrix} m \\ i \end{matrix}\right\}, k$$

produces a graphical plot showing, for one objective, the variations of the macro or micro level values with time. "m" selects the macro objectives and "i" the micro objectives. "k" specifies the objective number. If it is desired that a graph should be plotted over a previous one (without erasing the screen) a fourth argument "1" is added, e.g.

$$\geq \text{show}, \left\{\begin{matrix} m \\ i \end{matrix}\right\}, k, 1$$

In this case, the scale of plotting is that for the first plot. If a non-graphic terminal is being used then this command will produce a numerical output in tabular form for the macro or micro objectives as required. If a tabular output is required on a graphic terminal a "-1" final argument is used, e.g.

$\geq$ show, $\left\{\begin{matrix} m \\ i \end{matrix}\right\}$, k, -1

A table showing the values of all macro or micro objectives for a specified time period can be obtained by:

$\geq$ show, $\left\{\begin{matrix} m \\ i \end{matrix}\right\}$, time

(This command is only currently available if the total number of objectives is five or less).

As an aid to making direct comparisons between selected pairs of solutions the "show, c" command is used. This command is only available on a graphic terminal. The general form is

(c) $\geq$ show, c, k1, k2

where k1 and k2 are the solution numbers selected for comparison. An example of the resulting output is shown in Figure 3.

It is envisaged that the user may wish to use this command several times in succession. Following the first "show, c" command a "*" prompt is issued and all subsequent "show, c" commands are handled more efficiently. To return to the general command level a single "carriage-return" is used.

Only those objectives defined as being active are displayed with the "show, c" command (see (d)).

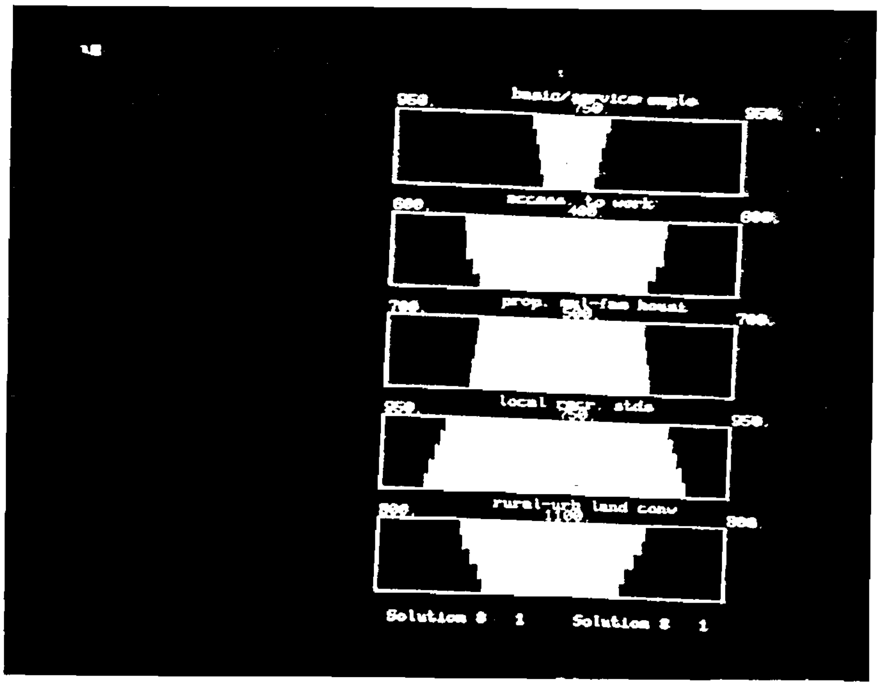


Figure 3. Example of comparison of two solutions

Objectives can be changed, interactively, from being active to inactive and vice versa, as follows:

(d) ≥ show, s, k, $\begin{Bmatrix}0\\1\end{Bmatrix}$

The "s" arguments select objective number "k" and the fourth argument defines whether the objective is active or not: 1 for active, Ø for inactive.

A list of objectives, with their respective active codes is obtained as follows:

≥ show, 1

Currently no more than 5 objectives can be active at any one time.

4.5.8 To examine macro allocations

As allowed in Phase 1, the user can also examine his macro allocation of land either spatially (for a specified time) or variations across time for nominated macro variables or regions. The command

(a) $\underline{>}$ rall, g, time

will produce a map of all macro regions showing the proportion of land allocated to the various macro activities for the specified "time".

To produce graphs showing the variation with time for selected macro variables or regions the following command is used:

(b) $\underline{>} \left\{ \begin{matrix} \text{vmac} \\ \text{rmac} \\ \text{rout} \end{matrix} \right\}, \text{ g, } [1]$

The optional third argument "1" is used for plotting a second graph over an existing one.

4.5.9 To examine micro allocation

The complete allocation at the micro level is available to the user in various ways. The user can examine the micro allocation in any micro region with the command

(a) $\underline{>} \left\{ \begin{matrix} \text{vnam} \\ \text{rnam} \end{matrix} \right\}$

Since the activities in each region can be on different measurement scales, a tabular output is always produced when a region is selected, showing the variation across time.

The user has the option of plotting a graph if a variable is chosen, by the command:

$$\geq \text{vnam, g, } \left[1 \right]$$

since variable "vnam" has the same units across all regions. The third, and optional, argument is used to plot a second graph over an existing graph.

From the solved micro allocation it is possible to compute the land usage by micro activities in micro regions. For any macro region, the user can display the micro subregions and the computed land use for the micro activities, e.g.

(b)

$$\geq \begin{Bmatrix} \text{rmac} \\ \text{rout} \\ \text{rext} \end{Bmatrix}, \text{ g, time}$$

will produce a map of the selected macro region showing the micro land allocations at the specified time. It should be emphasized that this command shows allocations in "area of land" units while those in (a) above display the actual production units.

4.5.10 The "devi" command

The "devi" command allows the user to examine the constraint deviations. All constraints (i.e. production, land area, macro land and special constraints) can be examined selectively. This information can tell the user which constraints were active for the solution of the <u>latest</u> micro solution. The data used for the "devi" command is only available following a new trial solution, as it is not retained in the "results" file. To obtain the deviation for an old solution the user needs to solve the problem again.

The general form of the command is:

$$\geq \text{devi}, \left\{\begin{matrix} p \\ l \\ m \\ s \end{matrix}\right\}, \text{time}$$

The second argument selects the constraint set to be examined (p = production constraints, l = land area constraints, m = macro land allocation constraints, s = special constraints).

As outlined in Section 4.3.10, the user defined macro land allocations are created as upper bounds. After solving a trial allocation the user may wish to remove all the slack from these constraints as this will have no effect on the solution obtained. This implies a modification to the macro land allocation. This can be done automatically by the command

$$\geq \text{devi}, r$$

which replaces the current macro allocation with a "slack-free" one. Both default upper bound constraints and any user defined lower bound constraints will have their slack removed. This adjustment is not made to the macro allocation already included in the "results" file. If the user wishes to save permanently this "slack-free" allocation he must re-solve the problem and create a new trial solution. In making adjustments to the macro allocation only 99% of the computed slack values are used. This is necessary to help prevent the new trial allocation being too tightly constrained making it difficult to solve. Also the user should check the new allocation at each time period (in Phase 1) to make certain that all production and land area constraints can be satisfied at the macro level. Small numerical changes in the estimated "macro production coefficients" (see Reference 1) may mean that some small violations need to be manually removed.

4.5.11 The "disp" command

For all graphing and mapping options in ISP, a display table is used to select which activities and/or regions are actually displayed. This is necessary where large numbers activities and/or regions are present and cluttered maps and graphs will result. By default all activities and regions (macro and micro) are displayed. The user can delete (or include) any activity and region at any time by modifying the display table. In the display table, each activity/region is given a Ø/1 flag; a 1 includes the respective element while a Ø excludes it. To gain access to the display table, the following command is used:

```
> disp
```

and a "*" prompt appears to enable the user to issue one of a series of commands:

$$\underline{*}\begin{Bmatrix} i \\ d \\ l \\ c \\ r \\ z \end{Bmatrix}, \left[\begin{Bmatrix} \text{vnam} \\ \text{vall} \\ \text{rnam} \\ \text{rall} \\ \text{vmac} \\ \text{rmac} \end{Bmatrix}\right], \ldots$$

The "i" followed by a set of valid variable or region names (macro or micro) will insert the elements as being active. Up to ten names can be included in the single command line. The "d" will delete the following elements from being active. The general names "vall" and "rall" are used to include all micro variables and micro regions respectively in the command string.

The "r" and "z" options (with no further arguments) will restore <u>all</u> elements to being active or zero (i.e. make inactive) <u>all</u> elements respectively.

The "l" command produces a listing of the current display table. The "c" will allow the user to continue back to the general command level.

4.6 Phase 5: The reporting program

A complete record of all trial solutions is retained in the "results" file. Although this file is "readable" by the user, it can become rather large and contain a lot of uninteresting trial solutions. A separate program, ISPREP is therefore available to report, in detail, selected trial solutions. These print-outs will provide a permanent record for the user for reference and detailed study of the trial solutions. Upon execution of the program ISPREP, the user can supply up to 20 trial numbers for reporting. If more than 20 are required a second execution is necessary.

The reports generated by ISPREP can be tailored to suit user requirements. These are seven basic reports available:

1. User defined macro allocation of land
2. Macro sectors in macro regions
3. Macro sectors in micro regions
4. Micro sectors in macro regions
5. Micro sectors in micro regions
6. Total production values
7. Objective values

The user can select any one, or any combination of the above reports, to be printed. All reports are tables with time across the page. Reports 2, 3, 4 and 5 are presented in terms of units of production and computed land area. Report 6 compares actual and apriori defined production levels across the whole region.

5. DATA FILE PREPARATION

5.1 General

All fields are (limited) free format with up to 5 digits for integer numbers and 7 digits (including decimal point if necessary) for floating point numbers. Decimal points need only be inserted if absolutely necessary. Each number must be separated by a comma (,) or a space, if less than the above field widths are used. A maximum of ten numbers can be included in a single line, extra lines are inserted until the complete set of values is included.

Two data files are required. The first is the problem configuration file and the second, the map data file. This later file is not required if a non-graphic terminal is being used. Example of these data files are given in Appendix G. Note that fields 16, 17 and 18 are repeated for each objective.

5.2 Configuration Data File Formats

Field	No of Records	Variables	Description
1	1	nv,nr,nt,nvg,nrg	No.of micro variables, No. of micro regions, No.of time periods, No.of macro variables, No.of macro regions
2	1	igv(i),i=1,nv	Macro variable associated with each micro variable
3	1	igr(j),j=1,nr	Macro region associated with each micro region
4	1	gvdes(i),i=1,mvg	4 character names for macro variables
5	1	grdes(j),j=1,nrg	4 character names for macro regions

Field	No of Records	Variables	Description
6	1	vdes(i),i=1,nv	4 character names for micro variables
7	1	rdes(j),j=1,nr	4 character names for micro regions
8	1	itimeØ, itime(k) k=1,nt	4 digit year descriptions for initial state and each analysis period
9	nv	xØ(i,j),j=1,nv i=1,nv	Initial micro state of system
10	nv	s(i,k),k=1,nt i=1,nv	Minimum production levels for each activity in each time period
11	nr	a(j,k),k=1,nt	Maximum available land areas in each region at each time period
12	1	fac(i),i=1,nv	Land area consumption coefficients for each activity (Ø/1)
13	nv	fprod(i,j),j=1,nr i=1,nv	Production coefficients (production/unit area of land) at initial time period
14	1	fpt(i),i=1,nv	Time dependent exponent factor for production coefficients
14	1	nobj	Number of objectives

Field	No of Records	Variable	Description
16	1	minmax, oname	Minmax code ("min" for a minimizing objective, "max" for a maximizing objective), objective name (up to 20 characters)
17	1	iqtype, qmax, qmin	Active/inactive code (Ø or 1), best and worse values for objective
18	varies	varies	Sets of required coefficients to evaluate objectives.

5.3 Map Data File Formats

Field	No of Records	Variable	Description
1	1	np	Number of co-ordinate points defining map of all regions
2	np	i,cx(i), cy(i)	Point No, x co-ordinate, y co-ordinate
3	nv	j,crx(j), cry(j)	Centroids of micro regions to locate pie charts
4	nrg	j,cgrx(j), cgry(j)	Centroids of macro regions to locate pie charts

Field	No of Records	Variable	Description
5	2*nr	i,nrp(i) ip(i,j),j=1,nrp(i)	Micro region No, No of boundary points (first line), list of boundary points with first = last to complete boundary (second line)

Note that for Field 5, the list of points defining the boundary of a micro region must be in clockwise or anti-clockwise sequence, ending with the starting number so that the boundary is completed.

6. USER SUPPLED SUBROUTINES (OBDAT and OBVAL)

For each planning problem, the evaluation of the planning objectives have to be formulated by (or for) the user into two subroutines which must be compiled with the main part of the code. Two subroutines are required, "OBDAT" to read in the required data and "OBVAL" to compute the values of the objectives as required by ISP. Examples of the two subroutines are included in Appendix H. The "OBDAT" routine must also be compiled with the reporting program "ISPREP".

In the preparation of the problem-specific routines for a new planning problem, the general notation and format should be copied from the examples in Appendix H. The variables, nobj, minmax, oname, qmax, qmin and iqtype must be included as indicated. Common block b1 gives the user access to the basic dimensions of the problem. Common block b5 contains the logical input/output unit numbers. The data file containing the configuration data file is "in". Any tables of coefficients must be appropriately dimensioned and included in common block b7, which appears in both OBDAT and OBVAL routines. The general sequence for the data should follow the formats given in Section 5.2.

The second subroutine, "OBVAL", computes the values of the objectives. These values must be computed at three levels (micro, macro and global) viz:

(a) Micro Objectives : are the values of the objectives computed for each micro region. The value of the variable "it" sets the time step for the required computation. The values are inserted into the array, q(l,j):j=1,nv;l=1,nobj.

(b) Macro Objectives : are the values of the objectives computed for each macro region. The results are inserted into the array, qm(l,j),j=1,nrg;l=1,nobj.

(c) Global Objective ; these are the values of the objectives aggregated across any sub set of regions, or including the whole region. The global objective can thus be used to focus attention on a part of the total region. They are stored in the array gq(l,k) ; k = 1,nt, l = 1, nobj.

APPENDIX A: COMMAND SUMMARY

1. Definitions

Alternative arguments are shown in {.}
Optional arguments are shown in [.]

vnam	:	refers to a valid four character micro variable name
rnam	:	refers to a valid four character micro region name
vmac	:	refers to a valid four character macro variable name
rmac	:	refers to a valid four character macro region name
rout	:	refers to those regions defined as being "outside" the macro state
rext	:	refers to those regions defined as being "external" to the macro state
vall	:	refers to (and includes) all micro variables
rall	:	refers to (and includes) all micro regions
time	:	refers to a valid four digit year label

timeØ	:	refers to a four digit year label for the defined initial state
ØØØØ	:	refers to (and includes) all time periods in the planning problem
value	:	refers to any (1-7 digit) numeric value
k, k1 or k2	:	refer to an integer value
file	:	refers to a valid disk file name

2. Phase 1 Commands: Allocation of land at macro land

c	continue to Phase 2
d	display table of allocation
v	verify allocation
n	step to next time
b	go back to previous time
g	plot map of macro allocation
g, {vmac / rmac / rout}, [over]	plot graph of macro allocations versus time
m, vmac, rmac, value	to modify element in macro allocation

3. Phase 2: Special Constraints

cont	to contiue to Phase 3
back	to go back to Phase 1
list, [, {vnam / rnam / rati / land}], [time]	to list special constraints
wipe, {sing / rati / land / call}	to remove special constraints
save	to save set of special constraints on disk
read	to read from disk set of special constraints

$\begin{Bmatrix}\text{vnam}\\\text{vall}\end{Bmatrix}, \begin{Bmatrix}\text{rnam}\\\text{rall}\end{Bmatrix}, \begin{Bmatrix}\text{timeØ}\\\text{time}\\\text{ØØØØ}\end{Bmatrix}, \begin{Bmatrix}<\\=\\>\\*\end{Bmatrix}\text{value}, \left[\begin{Bmatrix}\text{i}\\\text{r}\end{Bmatrix}\right], [\text{k}]$	to insert or change a single variable constraint at the micro level
$\begin{Bmatrix}\text{vnam}\\\text{vall}\end{Bmatrix}, \begin{Bmatrix}\text{rnam}\\\text{rall}\end{Bmatrix}, \begin{Bmatrix}\text{time}\\\text{ØØØØ}\end{Bmatrix}, /$	
$\begin{Bmatrix}\text{vnam}\\\text{vall}\end{Bmatrix}, \begin{Bmatrix}\text{rnam}\\\text{rall}\end{Bmatrix}, \begin{Bmatrix}\text{time}\\\text{ØØØØ}\end{Bmatrix}, \begin{Bmatrix}<\\=\\>\\*\end{Bmatrix}\text{value}, [\text{r}], [\text{k}]$	to insert or change a ratio type constraint at the micro level
$\text{vmac, rmac, }\begin{Bmatrix}\text{time}\\\text{ØØØØ}\end{Bmatrix}, \begin{Bmatrix}<\\=\\>\\*\end{Bmatrix}$	to modify macro land allocation constraints

4. Phase 3: Solution Phase

cont	to solve problem and then enter Phase 4
skip	to skip solving problem and to Phase 4 directly

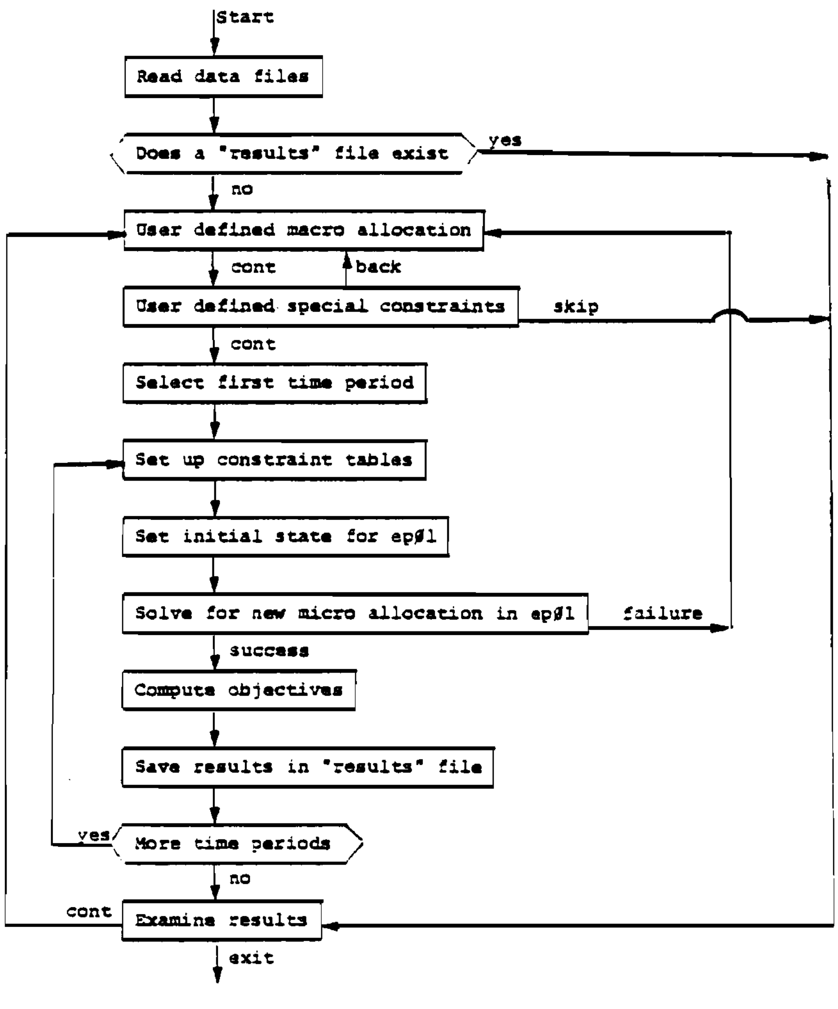
5. Phase 4: Reporting Phase

cont	return to Phase 1
exit	to terminate session
select, [k]	to select trial "k" for reporting
rest	to replace current macro allocation with that one currently selected
nond	to list current set of non-dominated solutions
$\text{show, }\begin{Bmatrix}\text{l}\\\text{s}\end{Bmatrix}, \left[\text{k,}\begin{Bmatrix}\text{Ø}\\\text{1}\end{Bmatrix}\right]$	to display and change objective active/inactive codes

$\text{show}, \left[\begin{Bmatrix} \text{m} \\ \text{i} \end{Bmatrix}\right], \left[\begin{Bmatrix} \text{k1} \\ \text{time} \end{Bmatrix}\right], \left[\text{k2}\right]$	to show objective values by graph or tabular presentations
show, c, k1, k2	to compare two trials by graphical display
rall, g, time	to plot a map of the macro regions to display macro land allocations
$\begin{Bmatrix} \text{vmac} \\ \text{rmac} \\ \text{rout} \end{Bmatrix}, \text{g}, \text{l}$	to display graphs showing macro allocations versus time
$\begin{Bmatrix} \text{vnam} \\ \text{rnam} \end{Bmatrix}$	to print micro solution in tabular form versus time
vnam, g,	to graph micro variable "vnam" in all regions versus time
$\begin{Bmatrix} \text{rmac} \\ \text{rout} \\ \text{rext} \end{Bmatrix}, \text{g}, \text{time}$	to plot a map of macro regions showing micro land uses
$\text{devi}, \begin{Bmatrix} \text{p} \\ \text{l} \\ \text{m} \\ \text{s} \end{Bmatrix}, \text{time}$	to display constraint deviations
devi, r	to replace current macro allocations with "slack free" allocation
$\text{disp}\ (\text{cr})\ \begin{Bmatrix} \text{i} \\ \text{d} \\ \text{l} \\ \text{c} \\ \text{r} \\ \text{z} \end{Bmatrix}, \left[\begin{Bmatrix} \text{vnam} \\ \text{vall} \\ \text{rnam} \\ \text{rall} \\ \text{vmac} \\ \text{rmac} \end{Bmatrix}\right], \ldots$	to examine and modify display tables

APPENDIX B: PROGRAM LOGIC AND SUBROUTINES

1. ISP logic



2. Subroutine functions

Main: "Isplan"	. calls "data" . calls "obdat" . calls "mapdat" . sets up initial macro state . calls "gall" . calls "setcon" . sets up initial solution for "epØ1" . calls "epØ1" . calls "obval" . saves solution in "results" file . calls "access"
Subroutine "data"	. reads basic configurations data
Subroutine "obdat"	. reads objective function data
Subroutine "gall"	. allows user to set up and modify macro allocation . allows user to add and modify special constraints . calls "mapp" . calls "mgraph"
Subroutine "setcon"	. sets up constraint coefficient table for each time period for "epØ1"
Subroutine "epØ1"	. solves micro allocation problem
Subroutine "access"	. allows user to report from latest or any, trial solution . calls "compar" . calls "mapp" . calls "mgraph"

Subroutine "mapdat"	:	. plots map of selected set of regions and draws pie charts showing proportioning of land use
Subroutine "compar"	:	. prepares a comparisons bar chart for any two selected trial solutions
Subroutine "mgraph"	:	. produces graphs of specified characteristics versus time
Tektronix Graphic Subroutines	:	. "amode", "move", "erase", "draw", "home", "line", "chr"

APPENDIX C: SOME IMPORTANT VARIABLES

The following internal variables may need some special consideration when setting up each new problem.

(a) Array: x(i): this array is used to store the new micro allocation for each call of "epØ1". The first n elements contain the new solution upon exit from "epØ1".

(b) Array: b(i): this array contains the m constraint deviations for the computed micro solution; that is, the given right hand side constant minus the computed left hand side function.

(c) Variables: racc, aacc: these two variables set the tolerances for the solution of the micro solution in "epØ1" and may need adjusting for each new problem. "racc" defines the relative accuracy and "aacc" the absolute accuracy of the micro allocation variables. Convergence is achieved in "epØ1" when one *or* other of these tolerances is achieved.

(d) <u>Variable: maxgen</u>: this variable sets the maximum number of iterations allowed in "epØ1".

(e) <u>Variables: iout, inp, idata, ires, lp, isave</u>: these variables define various logical unit numbers for input/output operations.

inp : is used for standard tty input
iout : is used for standard tty output
idata : is used for configuration data file
lp : is used for the map data file
ires : is used for the results data file
isave : is used for the saved sets of special constraints

APPENDIX D: ARRAY DIMENSIONING

The current version of ISP is dimensioned to the following limits:

Number of activity sectors at micro level (nv)	:	30
Number of regions at micro level(nr)	:	40
Number of macro activity sectors (nvg)	:	10
Number of macro regions (nrg)	:	10
Number of time steps (nt)	:	10
Number of objectives (nobj)	:	20

In addition, several other limiting array sizes exist which depend on some combination of various other parameters. The important ones are:

(a) Number of variables for maximum entropy problem:

$$n = nv * nr$$

(b) Number of constraints active for each time step for the maximum entropy problem:

$$m = nv + nr + (nvg * nrg) + ms + mr$$

where ms is the maximum number of single variable constraints active at any time period and mr is the maximum number of ratio type constraints active at any time period.

(c) Number of non-zero elements in the matrix of coefficients of the set of constraints:

$$nz = 2\ (nv * nr) + \sum_I \sum_J \left(nv_I * nr_J\right) + ms + 2\ (mr)$$

where nv_I is the number of activities in macro group I and nr_J is the number of regions in macro group J.

For these dimensions the current limits are:

ixsize = 2n + 6m	:	3000
m	:	400
nz	:	3000

(d) There are also limits on the total number of single variable and ratio type constraints (across all time periods).

Total number of single variable constraints (ncmax) : 100

Total number of ratio type constraints (nsmax) : 50

(e) The maximum number of trial solutions which can be searched for non-dominance and accessed for the "show, c" command: (maxsol) : 50

(f) There are also limiting values placed on the size of the map data:

The maximum number of co-ordinate points used to define the region map (np) : 225

The maximum number of points required to define a single region boundary (nbp) : 30

The array dimensions for all variables is given in the following table.

ARRAY	DIMENSION
anlist	10
area	nr,nt
as	nz
b	m
box	5
boy	5
cgrx	nrg
cgry	nrg
condev	m,nt
crx	nr
cry	nr
cx	np
cy	np
exc	ncmax
fac	nv
fprod	nv,nr
fpt	nv
g	m
garea	nrg,nt
ggq	maxsol,5,nt
gq	nobj,nt
grdes	nrg
grdes	nvg
gsup	nvg,nt
gx	nvg,nrg,nt
gx0	nvg,nrg
ic	nz
iexc	ncmax,4
igr	nr
igv	nv
imct	nv+nr+nvg*nrg,nt
ip	nr,nbp
iqtype	nobj
irel	m
irp	m+1
ispd	nsmax,4
ispn	nsmax,2
itime	nt
ll	2*m

ARRAY	DIMENSION
macr	nrg
macv	nvg
micr	nr
micv	nv
minmax	nobj
nond	maxsol
nrp	nr
oname	nobj
q	nobj,nr
q1	5,nt
q2	5,nt
qm	nobj,nrg
qmax	nobj
qmin	nobj
rdes	nr
sp	nsmax
sup	nv,nt
vdes	nv
x	ixsize
x0	nv,nr
x00	nv*nr
xc	nv,nr
xd	max(nv,nr),nt
xdiff	n

APPENDIX E: SCREEN FORMATING

The current version of ISP is set up specifically for a Tektronix 4010/12 screen but should also work on a 4014/15 screen. Several of the output operations may require re-formulating if other types of VDU (in a non graphic environment) or graphic terminal are to be used. Some changes may be necessary for the larger screen tektronix (4014/15) if optimum use is to be made of the bigger screen. The main areas to be examined are as follows:

(a) Format of the "macro allocation table": this table is output, upon request, in subroutine "gall". The formating will depend on the size of the numbers to be output and the width of the screen (in number of characters). The current version is formated for up to seven macro activity sectors and seven digit numbers (no fractional values). Format numbers, 998, 997, 996, 988, 987, may require adjustment.

(b) Location of map on screen: this is set in "tekpoint" units in subroutine "mapp". The lower left hand corner is set at co-ordinate location (txmin, tymin) and the upper right hand corner by the co-ordinates (txmax, tymax).

(c) Size of pie charts: the size of the pie charts may need adjusting and this is achieved by altering the values of the variables "rado" and "radi" in subroutine mapp. "rado" is the radius (in tekpoints) of the outer circle and "radi" the radius of the inner circle.

(d) Location of bar charts: the bar charts are used to compare two solutions. The location of the centre of the bottom of the lower bar chart is given by the co-ordinate pair (cx, cy) in tekpoints in subroutine "compar".

(e) Location and scaling of graphs: in subroutine "mgraph" the location of the graph origin is given by (cx, cy) the length of the x-axis by "xl" and the length of the y-axis by "yl". The time increment interval along the x-axis by "xt" (all in tekpoints). Currently each time interval is plotted an equal distance along the x-axis.

All maps are scaled automatically to be the largest size for the available space. The y-axis on all graphs are scaled automatically so that the largest value to be plotted is placed at the full scale position.

APPENDIX F: EXAMPLE PRELIMINARY RUN COMMANDS ON VAX AT IIASA

```
  isp.x 1=skd.007 3=skane.map
Specify terminal type
(0)-VDU; (1)-TEK4010/12; (2)-TEK4015; * 0
 Does a results file exist? (y/n) :y
 End-of-file found       250
 Last solution on file is No.    1
 Reporting phase
 Current solution: No.    1
>
```

```
  sskrep.x 1=skd.007 10=results 11=printer
 Output options -
     1 - User defined macro allocation of land
     2 - Macro sectors in Macro regions
     3 - Macro sectors in Micro regions
     4 - Micro sectors in Macro regions
     5 - Micro sectors in Micro regions
     6 - Total Production values
     7 - Objectives
     9 - All above
:1,3,7
 Input list of trials to be reported (up to 20)
1,5,20
```

APPENDIX G: EXAMPLE DATA FILE

1. Configuration Data File

```
10,10,3,5,5,      nv,nr,nt,nvg,nrg
1,1,1,1,2,2,3,3,4,5,  igv
1,1,1,1,2,3,2,3,4,5,  igr
gva1,gva2,gva3,gva4,gva5,     group variable names
gre1,gre2,gre3,gre4,gre5,     group region names
var1,var2,var3,var4,var5,var6,var7,var8,var9,var0, variable names
reg1,reg2,reg3,reg4,reg5,reg6,reg7,reg8,reg9,reg0, region names
1975,1980,1985,1990,        times
10,8,20,5,15,5,6,2,10,9,        initial state x0
5,3,25,8,12,8,8,6,15,4
15,5,30,9,13,4,12,5,20,3
20,4,2,4,25,3,19,4,25,8
5,9,4,6,5,8,7,3,30,6
30,10,5,12,30,6,3,2,25,9
15,11,11,6,10,9,8,1,15,5
5,3,16,9,26,3,30,5,10,2
20,6,25,4,38,1,5,6,5,1
10,5,30,5,4,4,4,10,8,3
160,170,180,             min producton levels
100,110,120
120,130,140
110,120,130
180,190,200
110,120,130
120,130,140
140,150,160
135,145,155
80,90,100
150,150,150,             max region areas
220,220,220
140,140,140
120,120,120
170,170,170
110,110,110
150,150,150
130,130,130
190,190,190
120,120,120
1,1,1,1,1,1,1,1,1,1,        land area consuption coefficients
1,1,1,1,1,1,1,1,1,1,        production coeffs
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
0,0,0,0,0,0,0,0,0,0,      time dependent exponent coeffs for
                          activities
```

```
2,             no. of objectives
min interaction cost                          description
1,2000,4000,                              iqtype,qmax,qmin
10,5,5,5,6,6,2,2,2,0,                     interaction coeffs.
5,10,4,3,5,5,2,2,3,0
5,4,10,3,4,4,2,2,4,0
5,3,3,10,3,3,2,2,5,0
6,5,4,3,2,2,1,1,6,7
6,5,4,3,2,2,1,1,6,7
2,2,2,2,1,1,1,1,1,1
2,2,2,2,1,1,1,1,1,1
2,3,4,5,6,6,1,1,1,1
0,0,0,0,7,7,1,1,1,1
1,6,3,3,8,5,8,8,9,11,                     interregion distance
6,1,8,4,3,8,5,9,8,11
3,8,1,4,8,3,7,5,7,8
3,4,4,1,5,4,4,6,6,8
8,3,8,5,1,7,2,8,5,9
5,8,3,4,7,1,5,2,4,5
8,5,7,4,2,5,1,6,3,7
8,9,5,6,8,2,6,1,3,3
9,8,7,6,5,4,3,3,1,4
11,11,8,8,9,5,7,3,4,1
min industrial density                            objective descr.ption
1,20,60,                                   iqype,qmax,qmin
1,1,1,1,1,1,1,1,1,1,                    matix of coefficients
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
%
```

2. Map Data File

```
19,              boundary point co-ordinates
1,36,155
2,70,153
3,116,150
4,148,102
5,127,82
6,108,67
7,80,17
8,41,20
9,36,66
10,15,90
11,13,128
12,57,133
13,45,115
14,0,0
15,88,125
16,79,108
17,42,102
18,70,88
19,65,47
1,40,137,     micro region centres
2,102,136
3,30,102
4,65,114
5,110,105
6,49,83
7,98,88
8,52,57
9,84,59
10,63,31
1,50,125,     macro region centres
2,102,95
3,51,71
4,84,59
5,63,31
1,6,          boundary points
1,2,12,13,11,1
2,6
2,3,4,15,12,2
3,6
11,13,17,9,10,11
4,7
12,15,16,18,17,13,12
5,5
15,4,5,16,15
6,4
17,18,9,17
7,5
16,5,6,18,16
8,5
9,18,19,8,9
9,5
18,6,7,19,18
10,4
8,19,7,8
%
```

APPENDIX H: EXAMPLES "OBDAT" AND "OBVAL" SUBROUTINES

```
c***********************************************************
c*
c*      Subroutine OBDAT
c*
c***********************************************************
        subroutine obdat(nobj,oname,minmax)
        character*3 minmax(20)
        character*20 oname(20)
        common /b1/ nv,nr,nt,nvg,nrg
        common /b5/ iout,inp,in,ires,lp,itty
        common /b7/ d1(30,40),a1(30,40),c2(30,40)
        common /m2/ qmax(20),qmin(20),iqtype(20)
        read(in,999) nobj
999     format(20i5)
        read(in,997) minmax(1),oname(1)
        read(in,996) iqtype(1), qmax(1),qmin(1)
997     format(a3,a20)
        do 20 i=1,nv
20      read(in,998) (a1(i,j),j=1,nv)
996     format(i5,2f7.0)
998     format(10f7.0)
        do 30 j=1,nr
30      read(in,998) (d1(j,k),k=1,nr)
        read(in,997) minmax(2),oname(2)
        read(in,996) iqtype(2),qmax(2),qmin(2)
        do 40 i=1,nv
40      read(in,998) (c2(i,j),j=1,nr)
        return
        end
```

```
c*********************************************************
c*
c*   Subroutine OBVAL
c*
c*********************************************************
        subroutine obval(x,q,it,qm,gq)
        dimension x(1),q(20,50),sup(50,10),area(50,10),x0(30,40),
     1 igv(50),igr(50),qm(20,10),gq(20,10)
        common /b1/ nv,nr,nt,nvg,nrg
        common /b3/ sup,area,x0,igv,igr
        common /b7/ d1(30,40),a1(30,40),c2(30,40)
        integer r,s
c
c  evaluate objective 1
c
        do 100 r=1,nr
        sum=0.
        do 90 i=1,nv
        ir=r+nv*(i-1)
        do 90 j=1,nv
        do 90 s=1,nr
        js=s+nv*(j-1)
90      sum=sum+x(ir)*a1(i,j)*d1(r,s)*x(js)
100     q(1,r)=sum/10000.
        do 105 j=1,nrg
105     qm(1,j)=0.
        do 110 j=1,nr
110     qm(1,igr(j))=qm(1,igr(j))+q(1,j)
        gq(1,it)=0.
        do 115 j=1,nr
115     gq(1,it)=gq(1,it)+q(1,j)
c
c  evaluate objective 2
c
        do 200 j=1,nr
        sum=0.
        do 150 i=1,nv
        ij=j+nv*(i-1)
        sum=sum+100.*c2(i,j)*x(ij)
150     continue
200     q(2,j)=sum
        do 205 j=1,nrg
205     qm(2,j)=0.
        do 210 j=1,nr
210     qm(2,igr(j))=qm(2,igr(j))+q(2,j)
        gq(2,it)=0.
        do 215 j=1,nr
215     gq(2,it)=gq(2,it)+q(2,j)
        do 220 j=1,nr
220     q(2,j)=q(2,j)/area(j,it)
        do 225 j=1,nrg
        sum=0.
        do 224 jj=1,nr
224     if(j.eq.igr(jj)) sum=sum+area(jj,it)
        qm(2,j)=qm(2,j)/sum
225     continue
        sum=0
        do 230 j=1,nr
230     sum=sum+area(j,it)
        gq(2,it)=gq(2,it)/sum
        return
        end
%
```

REFERENCES

Roy, G.G., and F. Snickars. 1981. An Interactive Computer Model for Land Allocation in Regional Planning. Part I: Theoretical Foundations and Operational Principles. WP-81-00. Laxenburg, Austria: International Institute for Applied Systems Analysis (forthcoming).

Eriksson, J. 1980. On Solving Linearly Constrained Maximum Entropy Problem. Research Report. LITH-MATH-R-1980-14. Linköping, Sweden: Department of Mathematics, Linköping University.