

COMPUTERIZED STRATIFIED RANDOM SITE-SELECTION APPROACHES  
FOR DESIGN OF A GROUND-WATER-QUALITY SAMPLING NETWORK  
by Jonathon C. Scott

---

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 90-4101



Oklahoma City, Oklahoma

1990

DEPARTMENT OF THE INTERIOR  
MANUEL LUJAN, JR., Secretary  
UNITED STATES GEOLOGICAL SURVEY  
Dallas L. Peck, Director

---

For additional information  
write to:

District Chief  
U.S. Geological Survey  
215 Dean A. McGee, Room 621  
Oklahoma City, Oklahoma 73102

Copies of this report can  
be purchased from:

Books and Open-File Reports  
U.S. Geological Survey  
Federal Center, Bldg. 810,  
Box 25425  
Denver, Colorado 80225

## CONTENTS

---

	Page
Abstract .....	1
Introduction .....	2
Concepts .....	2
Purpose and scope .....	3
Approaches .....	4
Site-selection biases .....	4
Weighted estimates of the population mean and variance .....	5
Approaches for defining the population of potential sites .....	6
Approaches for selecting sites .....	7
Simple random selection .....	9
Random selection within cells .....	10
Random selection with equal-area distribution .....	10
Random selection with iterating grids .....	12
Convergence criteria .....	12
Iteration procedure .....	13
Description of the software .....	15
Hardware and software requirements .....	15
User-supplied data requirements .....	15
Use of the software .....	16
Overview .....	16
Messages .....	18
Hypothetical study region .....	20
Products of the software .....	20
Category-index file .....	20
Site report .....	23
Plot file .....	23
Coverages .....	26
Definition of the population .....	27
Options .....	27
Sample run .....	30
Selection of sites .....	36
Options .....	36
Sample run using simple random selection .....	38
Sample run using random selection with equal-area distribution ..	40
Sample run using random selection with iterating grids .....	44
Site selection for the Central Oklahoma aquifer study region .....	50
Software design .....	54
Installation of the software .....	54
Data used by the software .....	57
Error messages .....	59
References .....	64
Glossary .....	66
Terms .....	66
Symbols .....	68
Appendix. Selection probabilities .....	69
Attachments .....	72

---

## ILLUSTRATIONS

---

	Page
Figure 1.--Diagram showing a stratified study region .....	2
2.--Diagram showing two populations of potential sites .....	3
3.--Flow diagram of the site-selection process .....	8
4.--Diagram showing subareas for a study region .....	10
5.--Diagram showing four equal-area cells for category "A" .....	11
6.--Simple schematic of the site-selection software .....	19
7.--Diagram of a hypothetical stratified study region .....	21
8.--An example of a site-selection report .....	24
9.--An example drawing of a plot file created by the site-selection software .....	25
10.--Diagram showing a population of potential sites for the hypothetical study region .....	31
11.--Diagram showing subareas of the hypothetical study region used to create cells for random selection of sites with equal-area distribution .....	32
12.--Diagram showing results of selecting sites from the "MIXED" category using random selection with equal-area distribution .....	45
13.--Diagram showing results of selecting sites from the "RURAL" category using random selection with iterating grids .....	51
14.--Map showing an equally spaced population of potential sites in the alluvium and terrace deposits of the Central Oklahoma aquifer study unit .....	52
15.--Map showing site selections in the alluvium and terrace deposits for the Central Oklahoma aquifer study unit .....	53

---

## TABLES

---

	Page
Table 1.--Comparison of attributes of the site-selection approaches ...	9
2.--Contents of the category-index file .....	22
3.--Descriptions of Fortran programs used in the site-selection software .....	55
4.--Locations of call statements and descriptions of ARC/INFO subroutines .....	58

---

## ATTACHMENTS

---

	Page
Attachment A.--Arc macro language program SAMPLE.AML listing .....	72
B.--Statements inserted into several Fortran programs .....	84
C.--Subroutines used in several Fortran programs .....	84
D.--Fortran program ITEXST listing .....	85
E.--Fortran program GRIDIM listing .....	86
F.--Fortran program DEFIDX listing .....	87
G.--Fortran program CHKIDX listing .....	91
H.--Fortran program PICKEM listing .....	92
I.--Fortran program EQARSA listing .....	94
J.--Fortran program NFLATE listing .....	99
K.--Fortran program SAMINF listing .....	100
L.--Fortran program UPTIDX listing .....	105
M.--Fortran program CATRPT listing .....	106
N.--Fortran program FCATRP listing .....	107
O.--Example projection-file listing .....	109

COMPUTERIZED STRATIFIED RANDOM SITE-SELECTION APPROACHES  
FOR DESIGN OF A GROUND-WATER-QUALITY SAMPLING NETWORK

By Jonathon C. Scott

ABSTRACT

Computer software was written to randomly select sites for a ground-water-quality sampling network. The software uses digital cartographic techniques and subroutines from a proprietary geographic information system. The report presents the approaches, computer software, and sample applications.

It is often desirable to collect ground-water-quality samples from various areas in a study region that have different values of a spatial characteristic, such as land use or hydrogeologic setting. A stratified network can be used for testing hypotheses about relations between spatial characteristics and water quality, or for calculating statistical descriptions of water-quality data that account for variations that correspond to the spatial characteristic.

In the software described, a study region is subdivided into areal subsets that have a common spatial characteristic to stratify the population into several categories from which sampling sites are selected. Different numbers of sites may be selected from each category of areal subsets.

A population of potential sampling sites may be defined by either specifying a fixed population of existing sites, or by preparing an equally spaced population of potential sites. In either case, each site is identified with a single category, depending on the value of the spatial characteristic of the areal subset in which the site is located. Sites are selected from one category at a time. One of two approaches may be used to select sites. Sites may be selected randomly, or the areal subsets in the category can be grouped into cells and sites selected randomly from each cell.

## INTRODUCTION

This report describes approaches for designing a ground-water-quality sampling network. Computer software for implementing these approaches is described and listed in attachments.

This report was produced as part of the pilot National Water-Quality Assessment (NAWQA) Program. The purpose of the pilot NAWQA Program is to test and refine concepts for a full-scale program that would describe the status and trends in the quality of the Nation's streams and aquifers and provide a sound, scientific understanding of the primary natural and human factors affecting the quality of these resources (Hirsch, Alley, and Wilber, 1988).

The approaches and computer software were designed for and used in the pilot NAWQA Program. However, the approaches can be used for designing ground-water-quality sampling networks for other hydrologic studies.

To avoid confusion, specific terminology is used in this report. These terms are underlined when first defined and used. Definitions of terms and symbols used are included in the glossary located at the end of the report.

### Concepts

During the design of a regional ground-water-quality network, it is often desirable to randomly select measurement sites to obtain a statistical representation of the ground-water quality in a particular study area. The study area is frequently divided into smaller contiguous areas, each of which has a particular spatial characteristic such as land use or hydrogeology. These smaller areas are referred to as areal subsets.

The spatial characteristic is used to categorize the different areal subsets. Each possible value of a spatial characteristic is referred to as a category. The categories may help to explain variations in water quality throughout the region. The term category is used instead of the more common term strata to avoid confusion with strata in the geological sense. A stratified study region is composed of areal subsets that are identified by spatial characteristics (fig. 1).

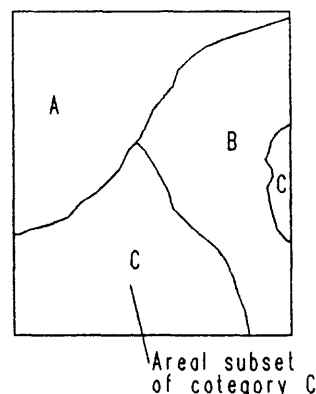
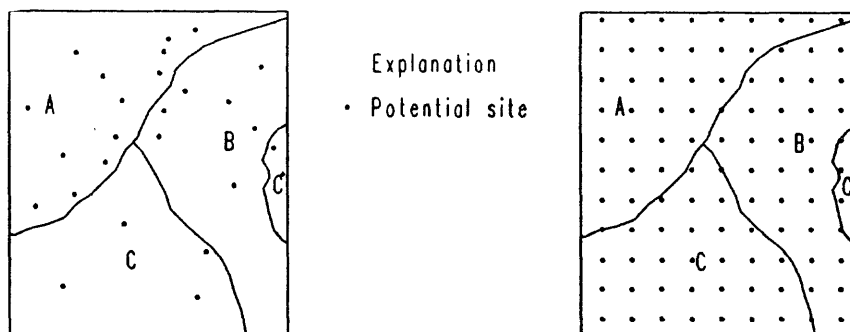


Figure 1.--A stratified study region.



(a) A population of existing wells.

(b) An equally spaced population.

Figure 2.--Two populations of potential sites.

Sites are selected from a group of potential sites. The population is the group of all sites in the study region that have the potential to be selected. Potential sites may be existing wells (fig. 2a), or may be points arranged in a equally spaced pattern throughout the study region (fig. 2b).

Sites are often selected from each category to test hypotheses regarding the variation of water quality among the categories, or to compute weighted estimates of the population mean and variance that account for the variation among categories. In either case, random selection of sites is necessary to obtain valid statistical estimates. Random selection of sites by category is called stratified random selection. The term selection is used rather than the more common term sampling to avoid confusion between use of the word sampling in a statistical sense and in reference to water-quality sampling.

### Purpose and Scope

This report describes approaches and computer software that can be used during the design of a ground-water-quality sampling network. The software includes several options for stratified random selection of sites. The region of interest can be any combination of one or more irregular polygons.

The software can be used to randomly select sites from a single hydrogeologic or land-use setting or from several different settings. The computer program is designed to select sites from each category, one category at a time.



This report first presents the theoretical discussion of various approaches for designing a ground-water-quality sampling network. The discussion is followed by a description of software that can be used for applying the approaches. The description includes application of the software to a hypothetical study region. Statistical analysis of the approaches presented in this report is described in an appendix.

### Approaches

A number of different approaches may be used to select sites from a category. The most basic approach is simple random selection. In this approach, a set of  $n$  sites is selected from  $N$  potential sites in a manner such that each site has an equal chance of being one of the  $n$  selected, and the selection of one site does not influence the selection of other sites (Gilbert, 1987).

Simple random selection is appropriate if the goal is to estimate selected statistics about a study area, and there are no major spatial patterns in water quality. In practice, this assumption is unlikely to be met, although its validity will not be known *a priori*. Furthermore, because potential measurement sites (existing wells) are often clustered in particular areas, simple random selection is unlikely to achieve a set of sampling sites that are areally distributed throughout the region of interest.

A useful modification of simple random selection to ameliorate these problems is to divide the region into cells and to randomly select one or more measurement sites within each cell. Cells are subdivisions of the study area used to areally distribute the locations of selected sites. This approach, referred to as random selection within cells, is likely to produce a more uniform areal distribution of sites than does simple random selection.

Random selection within cells is easy to apply throughout a single rectangular region. However, the region of interest in designing a ground-water-quality sampling network is rarely rectangular and commonly consists of one or more irregular polygons that result from the two-dimensional representation of the region on a map. For example, the region may correspond to areas having a particular land use or to the areal extent of a selected depth range within an aquifer.

### Site-selection Biases

Neither of the approaches described in this report, nor any known method, can remove all biases and ensure a set of sites that is representative of the water resource. However, the approaches described in this report are objective procedures that will reduce many biases. Some of the key potential sources of bias that need to be considered when designing a ground-water-quality sampling network include:

- (1) Existing water-supply wells commonly are located in areas known to produce usable volumes of potable water. Thus, the use of water-supply wells may introduce a bias that tends to minimize water-quality problems.
- (2) Wells in rural settings frequently are drilled near the landowners' houses. Therefore, large areas of the water resource that underlie other land uses and land covers may not be represented adequately by samples from existing water-supply wells in rural settings.
- (3) Techniques for constructing water-supply wells commonly are not suitable for the purpose of sampling water from a specific geohydrologic unit. Well screens may be placed in contact with multiple units, making it impossible to determine the source of the water sample. Even when a well is screened within a specific geohydrologic unit, the annulus may be gravel packed through multiple units, allowing water to enter into the well from different units.
- (4) Some observation wells emphasize contamination because the original purpose for which they were drilled was to determine the extent and longevity of a known contamination problem.
- (5) Existing wells may not be areally distributed throughout the study region. Thus, the results may tend to overemphasize the water quality near clusters of wells.
- (6) A conscious or unconscious bias to obtain interesting results can be caused when sites are selected without using a random-selection procedure. This bias can be caused even when sites are not selected from existing wells.

The random selection within cells approach can remove some of the effect of areal biases. However, this approach causes the probability of selection of a site to vary from cell to cell, depending on the population of sites in each cell. Thus, random selection within cells is not statistically random with respect to the population of potential sites, although the approach is random with respect to each cell. Formulas for calculating the selection probabilities for the approaches described in this report and the departure from statistical randomness are presented in the appendix.

#### Weighted Estimates of the Population Mean and Variance

Mathematical procedures have been developed for calculating weighted estimates of the mean and variance of a population that has been sampled by stratified random-selection methods, such as the ones described in this report (Cochran, 1977, p. 89-92; Gilbert, 1987, p. 46-47). When the variances within categories (strata) are less than the variance of the population as a whole, the estimate of the population mean can be improved by use of these mathematical procedures. Thus, if categories are defined in a manner that accounts for some of the population variance, these procedures

can be used to calculate a population mean with a correspondingly lower variance than would be calculated when using an unstratified site-selection method.

The weights used in these procedures are equal to the ratio of the size of the population in a category to the total population in a study region. If the population of potential sites is large and spaced equally, the weights computed in this manner are approximately equal to the ratio of the total area of areal subsets in the category to the total area of the study region. In this instance, either method of computing the values of the weights provides areally weighted estimates.

The computation of weights can be avoided by proportionally allocating the number of sites selected per category to the relative amount of category area in the study region. When proportional allocation is used, statistical estimates for the population are intrinsically areally weighted (Steel and Torrie, 1960, p. 420).

Calculation of weighted estimates of the population mean and variance do not remove all the biases discussed in the previous section, even when sites are selected randomly from existing wells. In this case, weighted estimates of the mean and variance describe the population of these wells, not necessarily the study region. Furthermore, it is questionable whether areally computed weights would produce estimates that characterize the study region because of the biases introduced by the siting of the wells.

#### APPROACHES FOR DEFINING THE POPULATION OF POTENTIAL SITES

Two different approaches may be used to define the population of potential sites from which sites are selected for water-quality sampling. With the first approach, a fixed population of existing sites is specified. With the second approach, one of two methods may be used to create an arbitrary population of potential sites arranged in an equally-spaced pattern throughout the study region. Either population-definition approach can be used with either of the site-selection approaches discussed in the next section.

The first population-definition approach allows the user to specify the population of potential sites by providing a coverage of points. (A coverage is a computerized representation of the locations and characteristics of spatial data.) Typically, these potential sites are the locations of known water wells. Because the locations of these wells are known, the coverage is referred to as a fixed population of potential sites. In some cases, the locations of all potential measurement sites can be identified readily from maps and (or) computer files. In these cases, the first approach can be used to specify the population of potential sites.

However, in some cases extensive effort would be required to identify all potential measurement sites. In other cases, new wells will be installed as part of the sampling program. Under either of these circumstances, the second approach for defining the population of potential sites

is appropriate. The computer software can generate an arbitrary, equally spaced population of potential sites. After sites are selected from an arbitrary population, a sampling crew can then locate the nearest suitable location for sampling.

An equally spaced coverage of potential sites may be created by either of two methods. The two methods differ in the way in which the spacing between sites is determined. With the first method, the user specifies the approximate number of potential sites, and the software calculates the spacing of sites by computing the square root of the ratio of the area of the study region divided by the population size. With the second method, the user specifies the spacing between potential sites.

After the spacing of potential sites has been decided, the software randomly selects a point below and to the left of the lower-left corner of the study region for the first potential site. Then, an equally spaced population of potential sites is prepared that covers the entire study region.

#### APPROACHES FOR SELECTING SITES

The software provides two different approaches for site selection: simple random selection and random selection within cells. Both approaches randomly select sites from a single category of a stratified population. The site-selection process is performed once for each category to be sampled. The process of selecting sites for a category is referred to as a site-selection pass (fig. 3). Thus, only the sites in a user-specified category are eligible for selection during a site-selection pass. For example, if a study area was stratified by land use, one pass would be used to select sites for each land-use type.

When beginning a site-selection pass, the user specifies the category, the number of primary sites to be selected from the category ( $\eta_p$ ), and the number of alternate sites to be selected ( $\eta_a$ ) for each primary site. Alternate sites are water-quality sampling locations selected to provide in advance for potential difficulties that may be encountered when visiting sites. For example, it may be impossible to drill at a selected site. When such a difficulty occurs, the hydrologist will have one or more alternative sites to visit if one or more alternate sites are specified during site selection.

A site-selection pass is divided into rounds. A round is the process of selecting one group of sites for a category. Primary sites are selected during the first round. These sites comprise the first choices for selection. Alternate sites are selected during an additional  $\eta_a$  rounds. During each round,  $\eta_p$  sites are selected. The number of rounds ( $\eta_r$ ) in a site-selection pass is equal to  $(1 + \eta_a)$  (fig. 3). The total number of sites ( $\eta_t$ ) selected during a pass is  $\eta_t = \eta_p \cdot \eta_r$ .

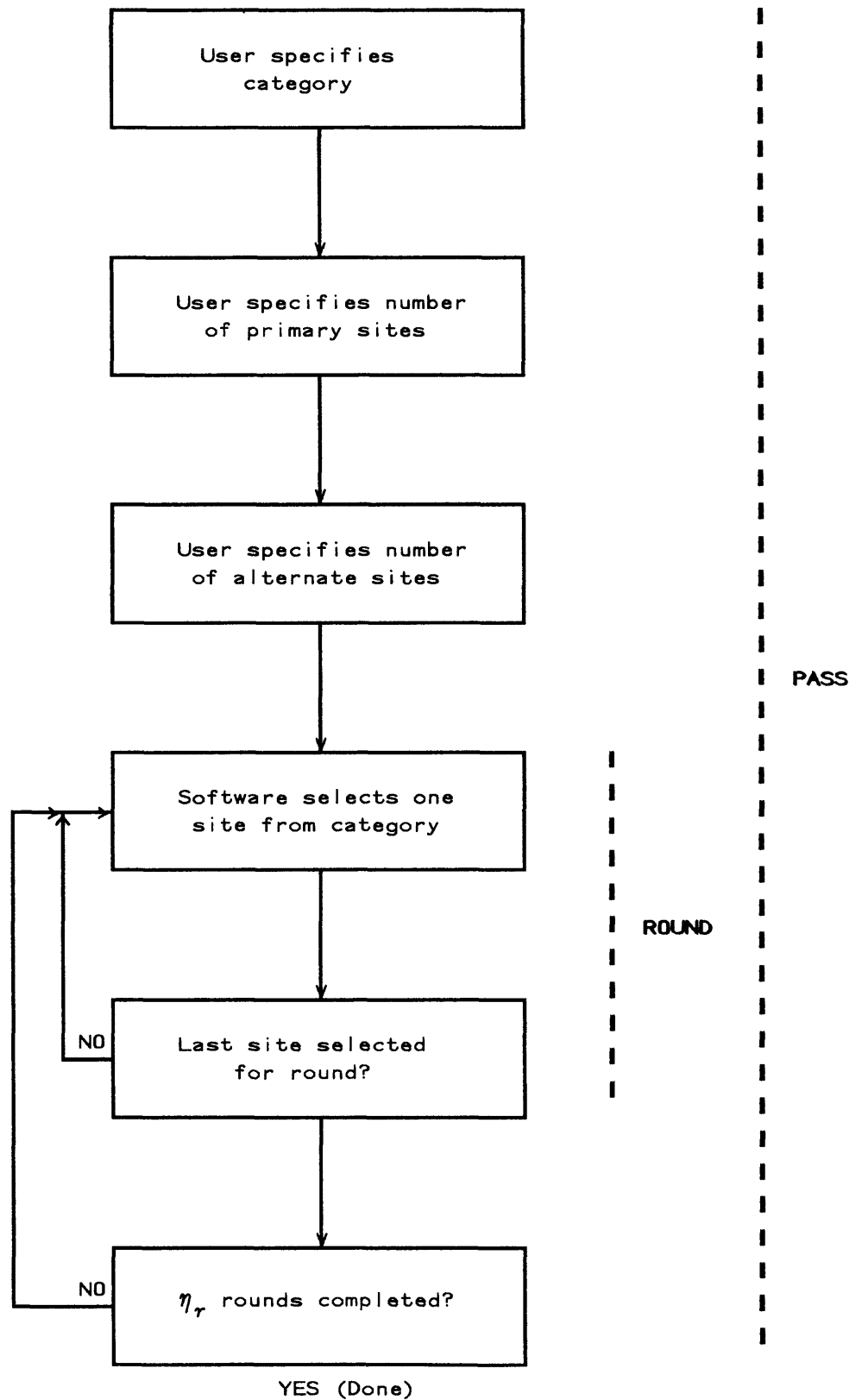


Figure 3.--Flow diagram of the site-selection process.

Table 1.--Comparison of attributes of the site-selection approaches

Attribute	Approach		
	Simple random selection	Random selection within cells	
		With equal-area distribution	With iterating grids
Complexity	Simple	Moderate	Complex
Time required to select sites	Minutes	Hours	Hours
Areal distribution	Random	Sites are selected randomly from within cells that contain equal areas of the category in the study region.	Sites are selected randomly from within cells distributed areally.
Cells	One cell includes the entire study region.	Cell definitions are reproducible.	Cell definitions are not reproducible.

The following sections describe each of the two approaches. A summary of the attributes of the site-selection approaches, including two different options for the random selection within cells approach, is presented in table 1.

### Simple Random Selection

Simple random selection is performed by first enumerating all potential sites from 1 to  $N$ , where  $N$  is the total number of potential sites in the category. Then, an integer number, between 1 and  $N$ , is chosen randomly, and the site associated with the chosen integer is selected. Next, another integer number and corresponding site is chosen randomly. This process is repeated until the desired number of primary sites,  $\eta_p$ , has been selected. The selections constitute the first round. Site selection continues until  $\eta_r$  rounds have been completed (fig. 3). Once a site has been selected, it is not available for selection again. In statistics texts, this technique often is referred to as "sampling without replacement."

With the stratified random-selection method, only one cell containing all areal subsets for the category is used for site selection. Sites located in the user-specified category are selected randomly. As noted earlier, simple random selection of sites does not ensure that the selected sites will be distributed areally throughout the study region or throughout all areal subsets of the same category.

### Random Selection Within Cells

To provide better areal distribution of site selections than may be achieved using the simple random-selection approach, the study region can be divided into cells. Then, sites are selected from within the cells. For a site-selection pass, one site is selected during each round from every cell containing area for the category.

Two methods are provided in the software for creating cells within the study region. The first method, random selection with equal-area distribution, creates cells that contain an equal amount of area for the category. The second method, random selection with iterating grids, creates cells by iteratively overlaying square grid patterns onto the study region until a pattern is found that satisfies the user's site-selection requirements.

### Random Selection with Equal-area Distribution

Random selection with equal-area distribution provides areal distribution of the selected sites. If one site from each cell is used for water-quality sampling, each site selected is associated with an equal area for the category.

The study region is divided into small subareas before site selection begins. The subareas are initially small square polygons. These square polygons are used to subdivide the areal subsets to create small irregularly shaped polygons, each of which contains area for a single category (fig. 4). The resultant subareas are later aggregated to create cells from which sites are selected. The subareas are created once for a study region. The subareas are reused when site selection is performed for different categories.

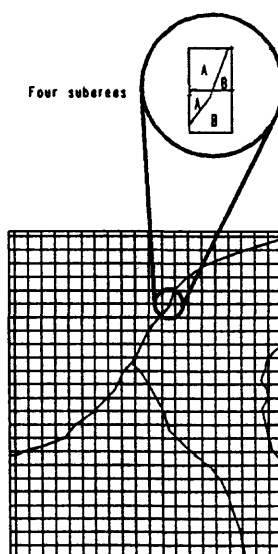


Figure 4.--Subareas for a study region.

The user-specified number of primary sites to be selected for a category ( $\eta_p$ ) determines the number of cells that are created. Each cell contains an area approximately equal to the total area of the category in the study region divided by the number of sites to be selected for the category.

Cells are created by first dividing the study region into vertical strips. The number of strips is computed by taking the square root of the user-specified number of sites. The number is rounded to the nearest integer. Each vertical strip contains approximately an equal amount of area for the category. Beginning in the lower-left corner of the study region, subareas in each horizontal row within the strip are aggregated to form a cell containing the desired area. When the correct area for the cell has been obtained, formation of the next cell begins.

The subarea aggregation continues until the top of the vertical strip is reached. If insufficient area has been aggregated to correctly form the current cell, the aggregation continues with the top row of the next vertical strip to the right. The aggregation in this strip continues with each row downward until the correct area for the cell has been reached. Formation of the next cell then begins. A similar process occurs when the bottom of the vertical strip is reached. Cell formation continues until all the subareas are assigned to cells (fig. 5).

Cells formed by this method are reproducible when the method is repeated using the same study region, areal subsets, subareas, and number of sites selected. Altering the size of subareas affects the smoothness of the cell boundaries, but does not noticeably affect the shape of the cells unless the subarea sizes are changed significantly.

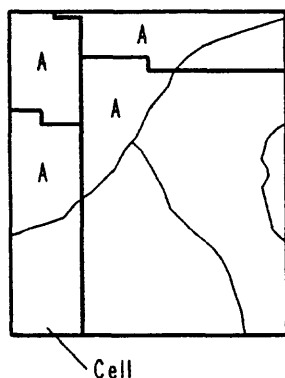


Figure 5.--Four equal-area cells for category "A".

When using the random selection with equal-area distribution method with a fixed population of potential sites that is not areally distributed, a solution may not be feasible. It is possible that some cells will be created that do not contain any potential sites, or that contain fewer than  $\eta_r$  sites. In the latter case, it may be feasible to design the network with the equal-area distribution method by decreasing  $\eta_p$ . Alternatively, either difficulty may be resolved using the random-selection with iterating grids method.



## Random Selection with Iterating Grids

This method, a second variation of random selection within cells, is the most complex of the three methods. In this method, the cells are determined by a grid of square cells that is overlaid on the study region. The origin of the grid is located randomly somewhere below and to the left of the lower-left corner of the study region. Grids are overlaid until there are  $\eta_p$  grid cells that contain at least  $\eta_r$  sites in the category of interest or secondary convergence conditions have been satisfied, as explained below. The size of the cells and the origin of the grid are altered between successive iterations. Grids created in this manner are not reproducible, and a solution meeting the convergence criteria is not unique.

This iterative procedure can resolve the difficulties described previously, when a fixed population is not distributed areally. However, some combinations of user requirements, areal-subset distributions, and populations of sites can cause a situation where some potential sites have little or no probability of being selected by the method.

Consider an example study region stratified by land-use with one large agricultural areal subset on the western half of the region, and a few small widely-separated agricultural areal subsets in the eastern half of the region. If few primary sites are being selected (that is, a coarse grid is used) and alternate sites are required, sites in the small agricultural areal subsets have little chance of being selected. This is because the software is unlikely to be able to create a grid with a cell covering the small agricultural areal subsets that contains the required number of sites ( $\eta_r$ ).

This type of problem is resolved by relaxing the restriction that every cell must contain  $\eta_r$  sites. The user can specify a number ( $f_a$ ), between zero and one, that is the fraction of cells that must contain at least as many potential sites as the number of site-selection rounds. This stipulation allows site selection to be made from two groups of cells. Each cell in the first group contains at least  $\eta_r$  sites, and sites are selected randomly from each one. Each cell in the second group contains less than  $\eta_r$  sites. All sites remaining to be selected are chosen from the second group without reference to cell boundaries.

This method can be adjusted to favor or not favor different parts of the study region depending upon the geometry of the areal subsets, the population of potential sites, and the convergence criteria. For the previously described example, decreasing  $f_a$  increases the probability that a site from one of the small agricultural areal subsets will be selected. Conversely, increasing  $f_a$  and (or)  $\eta_a$  decreases the probability that a site from the small agricultural areal subsets will be selected.

## Convergence Criteria

In addition to entering the category, number of primary sites, and number of alternate sites, the user enters additional data to control the

iteration process. The iteration-control variables are (1) the fraction of cells that must contain  $\eta_r$  sites ( $f_a$ ), (2) the maximum number of iterations, and (3) the damping factor ( $f$ ). The number of primary sites, the number of alternate sites, and the fraction of cells that must contain  $\eta_r$  sites comprise the convergence criteria.

The most stringent convergence criteria exist when the fraction of cells variable is set to one. At the end of each iteration, the software counts the number of cells containing at least  $\eta_r$  sites in the user-specified category. When this count of cells, defined as  $N_{ct}$ , exactly equals the number of primary sites to be selected, iteration stops and sites are selected from the cells in the same manner as the previous method.

When the fraction is set to a value less than one, convergence can occur if the previously described condition is met. Alternatively, another set of conditions can meet the convergence criteria. These secondary convergence conditions are expressed mathematically by the following two inequalities.

$$f_a \leq ( N_{ct} / \eta_p ) < 1 \quad (1)$$

$$N_a \geq \eta_r \cdot ( \eta_p - N_{ct} ) \quad (2)$$

where  $f_a$  = the user-specified fraction of cells that must contain at least  $\eta_r$  sites,

$N_{ct}$  = the number of cells that contain  $\eta_r$  sites,

$\eta_p$  = the user-specified number of sites,

$N_a$  = the number of potential sites not located in cells that contain  $\eta_r$  sites, and

$\eta_r$  = the user-specified number of alternate sites plus one.

When the secondary convergence conditions are met, site selection is split between cells that contain at least as many potential sites as the number of rounds, and cells that do not. The first secondary convergence condition requires that the ratio of  $N_{ct} / \eta_p$  must be greater than or equal to  $f_a$  and less than one. This condition pertains to the group of cells that contain at least  $\eta_r$  sites.

Additionally, the number of potential sites not located in cells containing  $\eta_r$  sites must be sufficient for selection of the remaining sites. The number of sites located in cells not containing  $\eta_r$  sites ( $N_a$ ) must be at least as large as the product of  $\eta_r$  and the result of  $\eta_p$  minus  $N_{ct}$ .

#### Iteration Procedure

The amount of time required for a site-selection pass can be lengthy. The maximum number of iterations used during a pass is set to eight in the software, but can be increased or decreased by the user. When the maximum

number of iterations is reached without finding a satisfactory grid, the user has the option of stopping the software, or attempting to select sites again.

When the grid tested during an iteration fails to meet the convergence criteria, a new grid is prepared. The number of cells in the new grid is based on how well the previous grid satisfied the convergence criteria. A grid is prepared by estimating how many cells are needed to satisfy the convergence criteria. Because the origin of the grid is moved randomly during the preparation of the grid, the method may use approximately the same number of cells repeatedly.

At the beginning of the first iteration, the initial estimate for the number of cells needed is calculated from the following formula.

$$G_i = \eta_p \cdot A / A_c \quad (3)$$

where  $G_i$  = estimate for the number of cells needed in the current iteration,

$A$  = total area of the study region, and

$A_c$  = total area of all areal subsets in the category.

When an iteration fails to converge, a new estimate for the next iteration is calculated. The formula shown below is based on the assumption that a linear change in the number of cells will produce a grid that satisfies the convergence criteria.

$$G_{i+1} = G_i \left( \frac{(\eta_p - N_{ct})}{N_{ct}} \cdot f_i + 1 \right) \quad (4)$$

where  $G_{i+1}$  = estimate for number of cells needed in the next iteration,

and  $f_i$  = damping factor for the iteration.

When  $f_i = 1$ ,

$$G_{i+1} = G_i \left( \frac{\eta_p}{N_{ct}} \right).$$

The value of the damping factor for the iteration ( $f_i$ ) is either one or the user-specified damping factor ( $f$ ). When the  $N_{ct} / \eta_p$  ratio is less than 0.6 or greater than 1.4 the damping factor for the iteration is equal to 1.0. Otherwise, the damping factor for the iteration is equal to the user-specified damping factor.

The damping factor causes the change in the estimate for the number of cells to be increased (when  $f > 1.0$ ) or decreased (when  $f < 1.0$ ) when a solution is near. The geometry of the areal subsets in some study regions can cause the method to continue to iterate in the vicinity of a potential solution without quickly finding the solution. The damping factor is used to cause larger or smaller changes in the estimate in these cases, thereby accelerating solution of the problem.

## DESCRIPTION OF THE SOFTWARE

### Hardware and Software Requirements

The site-selection software was developed on a Prime<sup>1</sup> minicomputer using the Primos operating system, revisions 21 and 22. It has not been installed or tested on other computer systems. The site-selection software is dependent on the hardware and software currently used by the U.S. Geological Survey (Rennick, 1986), and must be modified to use the software with a different configuration of hardware and software.

The site-selection software is designed to be used with a computer terminal that can display bold characters using standardized codes (American National Standards Institute, 1979). However, this terminal dependence can be easily removed or changed by modifying the program in attachment A.

The ARC/INFO geographic information system (GIS) (Environmental Systems Research Institute, Inc., 1987a; 1987b), revision 5.0.1, is used within the software to perform many digital cartographic and data-manipulation processes. Use of the software requires licenses for the ARC geographic information system software and the INFO file-management software, revision 9.42 (Henco, Inc, 1983).

The interaction between the user of the software and the programs used by the software is handled by a computer program written in the Arc Macro Language (AML) (Environmental Systems Research Institute, Inc., 1987c). AML can be used with several different computer-operating systems. The AML program performs operations using the ARC/INFO GIS and also runs other computer programs that are part of the site-selection software.

The other computer code (attachments C through N) is written in Fortran 77 (Johnson, 1983) and performs parts of the site-selection process. All of the Fortran computer programs call subroutines in the ARC/INFO subroutine library, revision 5.0.1 (Environmental Systems Research Institute, 1989). Most of these computer programs are written using standard Fortran 77 (American National Standards Institute, 1978). The exceptions to standard Fortran are identified in the Installation of the Software section of this report.

### User-supplied Data Requirements

A digital map (coverage) of the study region is needed before the software can be used. The coverage needs to contain polygons for areal subsets that will be used to stratify the selection of sites. Additionally, a character variable in the coverage needs to identify the category of each

---

<sup>1</sup> Use of brand, firm, and trade names in this report is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

areal subset. (The maximum number of characters that can be used to identify categories is 16.) In the simplest case, the coverage consists of polygons that are identified as either in or out of the set of polygons from which sites will be selected. More often, each polygon is identified by an alphanumeric code indicating the land use, land cover, or hydrogeologic setting of the area.

The digital cartographic processes used in the software operate with two-dimensional Cartesian representations of the earth's surface. The user selects a mathematical transformation, known as a map projection, to convert three-dimensional coordinates to two-dimensional coordinates. All map projections introduce a combination of errors in the relative locations, shapes, areas, or lengths of map features (Synder, 1987). It is not possible to recommend a particular map projection that is ideal for use of the software for any possible study region. The map projection selected by the user needs to provide a continuous two-dimensional coordinate system and minimize errors of scale. An equal-area map projection needs to be considered if the random selection with equal-area distribution method is used, or if areas from the coverage will be used in statistical computations.

If a fixed population of potential water-quality sampling sites will be used, a coverage containing the locations of these points is needed. If a point coverage of potential sites is supplied, it needs to be in the same units and map projection used for the polygon coverage of areal subsets.

## Use of the Software

### Overview

The site-selection software is interactive. Interactive software uses two-way communication between the user in the form of entries typed on the terminal keyboard, and the software in the form of messages displayed on the terminal screen.

The site-selection software will work correctly only when the user is running the ARC/INFO software and the "Arc:" prompting message is displayed. To start the site-selection software, the following command is typed.

Arc: **&R SAMPLE**

As shown above, font changes are used in this report to distinguish text of the report, messages displayed on the terminal, and information typed by the user. In the example above, "Arc:" is displayed on the terminal and "&R SAMPLE" is typed by the user.

Each time the software is started, messages are displayed that prompt the user to enter the site-selection method, the name of the areal-subset coverage, and the category-variable name. The prompting messages are displayed consecutively; after a response is entered for each, the next one is displayed, as shown on the next page.

Site selection may be performed using one of three methods:

Code	Method
1	Simple random selection
2	Random selection with equal-area distribution
3	Random selection with iterating grids

Enter method code:

Enter name of areal-subset coverage:

Enter name of category variable:

The software checks for an existing areal-subset coverage with the user-specified name containing the user-specified category-variable name. If a problem is detected, an error message is displayed, and the prompting message for the areal-subset coverage name is displayed again.

The software displays next a list of options called the main menu, as shown below. When software is run for the first time with a specific areal-subset coverage, the second option, "Select sites from population", is not available. A population must be defined using option one, before sites can be selected using option two.

Site-selection software main menu

Code	Option
1	Define population
2	Select sites from population
3	Exit from site-selection software

Enter option code:

When the first main-menu option is selected (Define population), the user is given an opportunity to exclude sites that are located within certain areal subsets. These areal subsets may be identified by specifying a minimum area and (or) a minimum area-perimeter ratio. It may be useful to avoid selection of sites in small or extremely elongated areal subsets. Sometimes the accuracy of assigning a spatial characteristic to such an areal subset is questionable because the actual location of the border of the areal subset is uncertain.

Next, the user selects one of three options for defining the population of potential sites. As described previously, two of these options cause the software to create a coverage of equally spaced points. The other option allows the user to specify the name of a point coverage of potential sites. If the random-selection with equal-area distribution method is used, the user next enters specifications for preparing a coverage of subareas. When the population has been defined, the main menu is displayed again.

When the second main-menu option is selected, the software prints messages prompting the user to enter specifications for selecting sites for

a single category. Then, sites are selected, the selections are stored, and the main menu is displayed.

When the third main-menu option is selected, the software stops and the "Arc:" prompting message is displayed again. A simple schematic showing the flow of site-selection software is presented in figure 6.

The results are stored each time the site-selection software is used. All work performed has been stored already when the third main-menu option, "Exit from site-selection software", is selected. This feature permits repeated use of the site-selection software for selecting sites for a study region. Thus, the problem of designing a ground-water-quality sampling network can be separated into stages, and each stage performed when time is available.

For example, a study region is stratified into five categories. The problem could be separated into six stages, each of which can be performed at a different time. The first stage necessarily is definition of the population. Each of the other five stages consist of selecting sites for a particular category and can be performed in any order.

### Messages

There are four different types of messages displayed by the software: (1) Prompting messages, (2) informative messages, (3) operational messages, and (4) error messages.

Prompting messages are displayed by the software to elicit a response from the user. Some prompting messages are menus that present a numbered list of options. The user selects an option from a menu by entering the corresponding number (referred to as a code). If a value is entered that is not listed on the menu, a message is displayed explaining the error, and the menu is displayed again.

Other prompting messages are simply questions that need to be answered. Questions that require an affirmative (yes) or negative (no) response may be answered in several ways. An affirmative response may be entered by typing YES, YE, Y, OKAY, OKA, OK, or O. A negative response may be entered by typing NO, N, QUIT, QUI, QU, or Q. If none of these responses are entered, the software displays the message "Enter YES, OKAY, NO, or QUIT:". An acceptable affirmative or negative response should be entered.

Some prompting messages require a numeric answer. If a non-numeric answer is entered, an error message is displayed and the prompting message or previously displayed menu is displayed again. Other questions require responses that are integers or positive numbers. If an invalid response is entered, the software displays an appropriate error message, and displays the prompting message again.

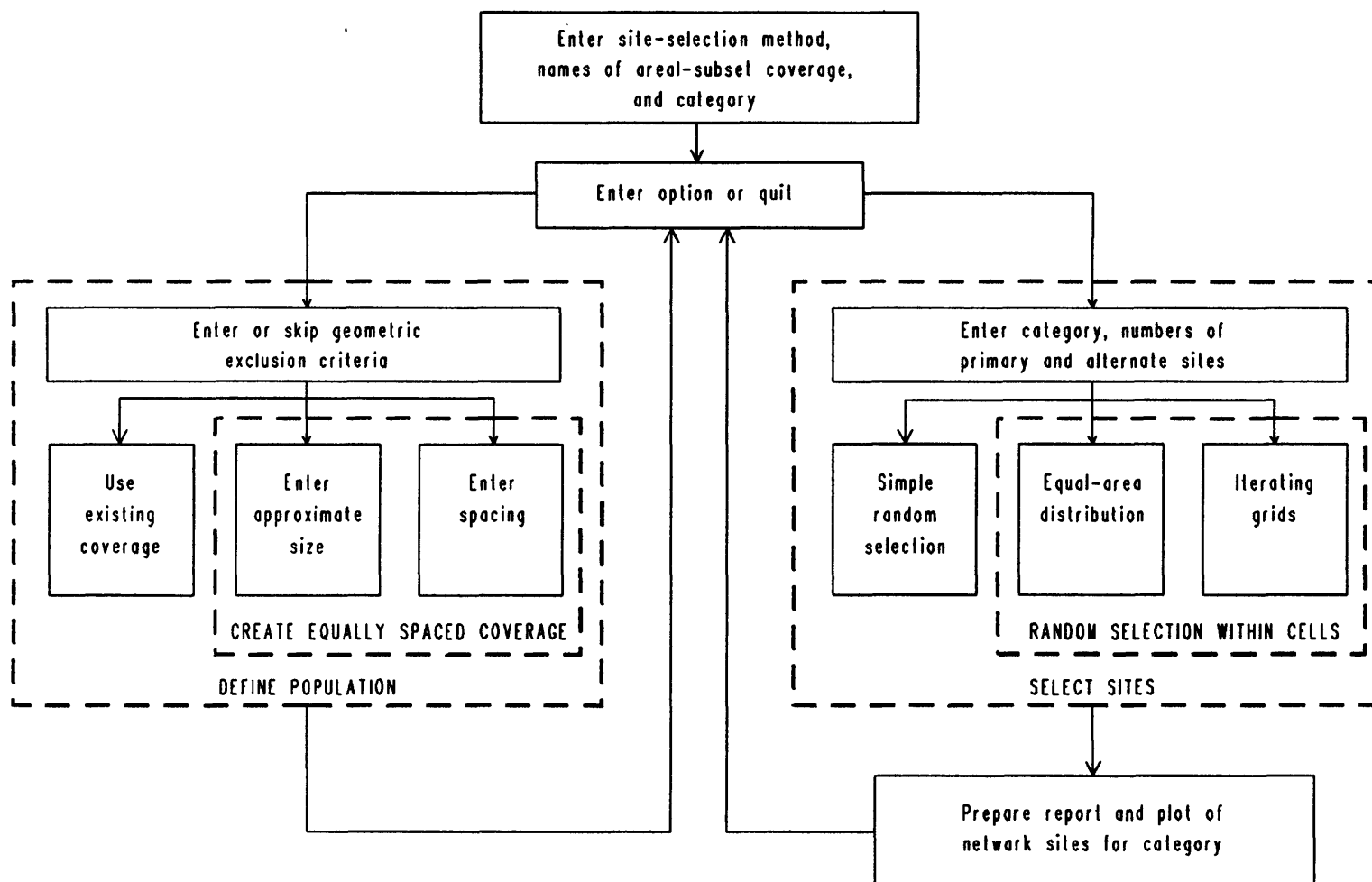


Figure 6.--Simple schematic of the site-selection software.



Informative messages are displayed by the software to describe the data-processing activities during different stages of the site-selection process. Most informative messages are displayed in bold characters to separate them from operational messages.

Operational messages are displayed by the software while performing various operations. Operational messages may not be very informative to a user not conversant with the jargon of the ARC/INFO GIS. Operational messages that are displayed by the software are presented in this report with samples runs of the software.

Error messages are displayed by the software when problems occur during site selection. Some error messages are discussed in this section, and all error messages likely to be displayed are described in a separate section, "Error Messages", near the end of this report.

### Hypothetical study region

For the purpose of describing the site-selection software, a stratified areal-subset coverage (fig. 7) was created for a hypothetical study region. Three categories are identified: "URBAN", "MIXED", and "RURAL". The areal-subset coverage is named LUSE and contains a category variable named CATEGORY.

Sample sessions are shown in subsequent sections of this report. In the samples, numbers shown on the left side of each line are not displayed on the terminal when the software is run. These numbers are used for referring to the lines in the text of this report. In the sample sessions, information typed by the user is printed in bold characters, and messages displayed by the software are printed in a sans-serif font. Informative messages, displayed in bold characters on the terminal, are printed in italics. The characters <CR> are printed where a carriage return is typed by the user without entering any data.

### Products of the Software

After all of the data needed to select sites have been entered and site selection has been performed for a category, the software stores the results in several places. These storage locations and the information they contain are described in the following paragraphs.

#### Category-index File

When a population is defined, the software creates a category-index file (table 2), and updates the file every time sites are selected. The name given to the category-index file is CATIDX. The file can be accessed using the INFO file-management system (Henco Inc, 1983) as shown on page 22.

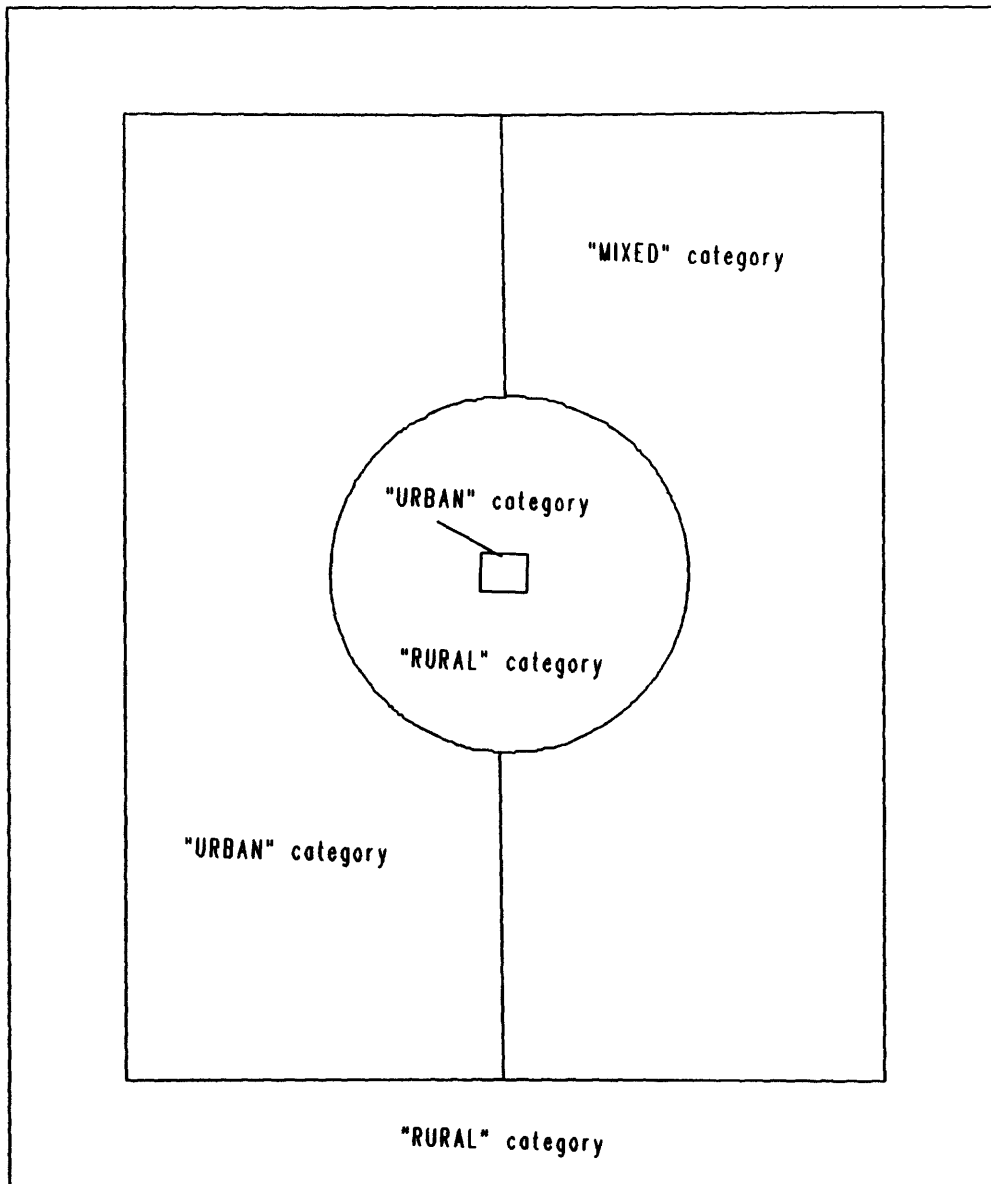


Figure 7.--A hypothetical stratified study region.

Table 2.—Contents of the category-index file

[Field-type codes: C = character, F = real number stored in binary format, and B = integer stored in binary format]

Variable name	Position of first byte (bytes)	Field width (bytes)	Output- display width (char- acters)	Field type (code)	Number of decimal places	Description
CATEGORY	1	16	16	C	—	Name of category
AREA	17	4	12	F	3	Total area of all areal subsets in the category
NUMPOLY	21	4	5	B	—	Number of areal subsets in the category
NUMPTS	25	4	5	B	—	Number of potential sites in the category
FRACTION	29	4	12	F	4	Ratio of category area divided by total area of the coverage
NOSAMP	33	4	5	B	—	Number of sites selected for the category

Arc: INFO

INFO EXCHANGE CALL

THU, OCT 12 1989

INFO 9.42 12/ 1/86 52.74.63\*

COPYRIGHT 1986 HENCO SOFTWARE, INC.

PROPRIETARY TO HENCO SOFTWARE, INC.

US GOVT AGENCIES SEE USAGE RESTRICTIONS IN HELP FILES (HELP RESTRICTIONS)

ENTER USER NAME>ARC

ENTER COMMAND >SELECT CATIDX

4 RECORD(S) SELECTED

ENTER COMMAND >LIST

\$RECNO	CATEGORY	AREA	NUMPOLY	NUMPTS	FRACTION	NOSAMP
1		0.000	0	1	0.0000	0
2	RURAL	81508.880	2	257	0.4579	0
3	MIXED	47996.840	1	154	0.2697	0
4	URBAN	48485.970	2	153	0.2724	0

ENTER COMMAND >Q STOP

Arc:

There is one record in the category-index file for every category in the study region, plus one record for unidentified areal subsets (the category variable contains blanks). The number of sites in areal subsets that are excluded geometrically is added to the total number of sites (NUMPTS) in the blank category. However, the areas of subsets that are excluded geometrically are not added to the total area (AREA) of the blank category. Instead, the areas of subsets excluded geometrically are added to the total area for the category of which the areal subsets are members. The computations are performed in this manner to facilitate the computation of weighted estimates of the population mean and variance.

The software uses the category-index file to keep track of the number of primary sites selected from each category and to make an initial estimate for preparing the first grid when selecting sites using the random-selection with iterating grids method.

### Site Report

The software writes a file called the site-selection report. The file is named by joining together the characters "SITE.REPORT." and value of the category variable. An example site-selection report from a file named SITE.REPORT.URBAN is shown in figure 8.

Most of the information written on the heading of the report is from the category-index file. The locations of the selected sites are written to the report. The locations are sorted by site number and selection-round number. There is one site number for each primary site.

After each site number, latitude and longitude are listed for the selected sites, along with an round number. The round number designates the order for visiting primary and alternate sites. When selecting a particular site, if the site with round-number 1 can be used for the project, the alternate sites with larger round numbers are not used. Otherwise, the alternate site with the next larger round number is visited until an acceptable site is found.

### Plot File

For illustrating the locations of the selected sites, the software prepares a binary file suitable for plotting using the ARC/INFO GIS. Several ARC/INFO commands (Environmental Systems Research Institute, 1987e) can be used to display the contents of the plot file. The binary plot file is named by joining the together the character "#", the name of the areal-subset coverage, a period, and the value of the category variable.

The contents of the plot file depict the locations of the selected sites with circles. Within each circle is the round number of the site. The borders of the areal subsets are included. If the random selection within cells approach is used, the borders of the cells also are included in the plot file. An example drawing of a plot file is shown in figure 9.

RESULTS OF RANDOM SELECTION FOR CATEGORY 'URBAN

Total area of category	=	48486.0
Number of areal subsets	=	2
Number of potential sites	=	153
Number of primary sites selected	=	4
Number of alternate sites selected	=	2

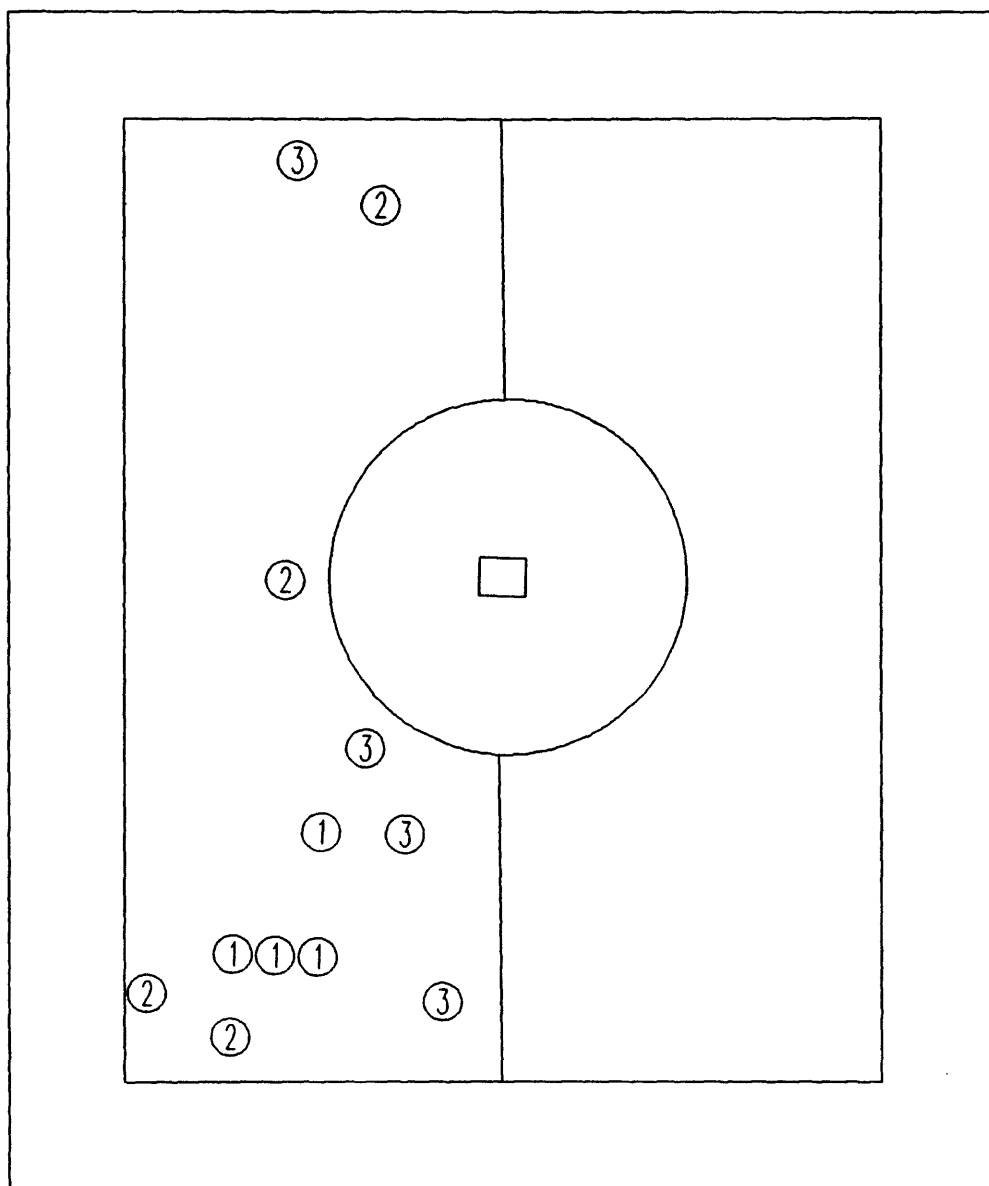
----- SITE # 1 -----		
ROUND	LATITUDE	LONGITUDE
1	34 00 04	95 59 57
2	34 00 03	95 59 53
3	34 00 03	95 59 58

----- SITE # 2 -----		
ROUND	LATITUDE	LONGITUDE
1	34 00 03	95 59 57
2	34 00 12	95 59 54
3	34 00 08	95 59 52

----- SITE # 3 -----		
ROUND	LATITUDE	LONGITUDE
1	34 00 03	95 59 54
2	34 00 02	95 59 53
3	34 00 13	95 59 56

----- SITE # 4 -----		
ROUND	LATITUDE	LONGITUDE
1	34 00 06	95 59 55
2	34 00 04	95 59 56
3	34 00 02	95 59 55

Figure 8.--An example site-selection report.



EXPLANATION

○ SELECTED SITE, ROUND NUMBER  
SHOWN INSIDE CIRCLE

Figure 9.--An example drawing of a plot file created by the site-selection software.

## Coverages

The site selections are stored in the coverage containing the population of potential sites. The coverage is named by joining together the name of the areal-subset coverage and the characters ".POP".

Three variables in the population coverage are used to identify the sites. The first of these is an integer variable named CHOSEN. When site selection begins for a category, the value of CHOSEN is set to zero for all potential sites in the category. When a site is selected, the value of CHOSEN is set to the round number during which the site is selected. The second variable is an integer variable named CELLNO. Unless the simple random-selection method is used, the value of the CELLNO variable is equal to the cell in which the site is located. The third variable is a character variable that identifies the areal subset in which the site is located. This variable has the same name as the corresponding variable in the areal-subset coverage.

If the simple random-selection approach is used, one cell is used for the entire study region. Otherwise, a polygon coverage is created containing the cells. The name of the coverage is determined by joining together the name of the areal-subset coverage, the characters ".CELL.", and the value of the category variable. Each polygon in the coverage is identified by a variable named CELL. The value of the CELL variable corresponds to the CELLNO variable in the population coverage and the site number in the site-selection report.

The population and cell coverages are useful for creating plots that illustrate the water-quality sampling network. The reader may refer to the ARC/INFO documentation for information about creating plots using these coverages (Environmental Systems Research Institute, 1987d).

When a population of potential sites is defined, using option one of the main menu, any existing category-index file is deleted and a new one is created. Also, the contents of the variables named CHOSEN and CELLNO in an existing population coverage are lost, because any existing population coverage is deleted before the new one is created. If a coverage of subareas exists, it is deleted and a new one is created.

When sites are selected from a category more than once, the values of the variables CHOSEN and CELLNO in the population coverage are recalculated. Also, the value of the NOSAMP variable in the category-index file is changed, and any existing site-report file or plot file for the category is rewritten. Finally, if a coverage of cells exists for the category, it is deleted and a new one is created.

## Definition of the Population

### Options

The first task required when selecting sites with the software is to define the population of potential sites by selecting option one from the main menu. Next, the following message is displayed.

Do you wish to exclude areal subsets using geometry (CR=NO)?

This prompting message provides an option for excluding some areal subsets from the site-selection process. Areal subsets of all categories that contain less than some user-specified area or less than some user-specified ratio of area divided by perimeter can be excluded from the site-selection process. To use this geometric exclusion, an affirmative response is entered. A carriage return or a negative response is entered to omit geometric exclusion.

When an affirmative response is entered, the following two prompting messages are displayed.

Enter minimum area (CR=no minimum):

Enter minimum area/perimeter ratio (CR=no minimum):

Either or both of these exclusion criteria may be used by entering an appropriate value for each question. As indicated by the prompting messages, entering a carriage return without a value disables the corresponding criterion. Values need to be entered in the same units as the areal-subset coverage.

**WARNING:** The software implements the geometric-exclusion criteria by setting the value of the category variable to a single blank character for all polygons containing less than minimum area or minimum area/perimeter ratio. Thus, when geometric exclusion is used, a blank should not be used in the areal-subset coverage to identify a category that will be used for site selection.

After geometric exclusion has been performed, or if the user selects not to perform geometric exclusion, the following prompting messages are displayed.

You may define a population using any of three methods:

Code	Method
------	--------

- |   |   |
|---|---|
| 1 | Specify the approximate number of potential sites |
| 2 | Specify the distance between each potential site  |
| 3 | Specify an existing point coverage                |

Enter a method code:



The first two methods create an equally spaced population and the third method uses a fixed population (see the "Approaches for defining the population of potential sites" section of this report).

When the first method is selected, the following prompting message is displayed.

Enter approximate number of sites (CR=10,000):

A positive integer should be entered. Because the origin of a population of equally spaced sites is placed randomly, the value entered is approximate. Typing a carriage return without entering a value causes the software to define a population with approximately 10,000 potential sites.

When the second method is selected, the following prompting message is displayed.

Enter distance between each site:

A positive number in length units of the areal-subset coverage should be entered.

When the third method is selected, the following prompting message is displayed.

Enter name of point coverage:

The name of the coverage containing the fixed population of potential sites should be entered. If the named coverage does not exist, the software displays an error message and displays the prompting message again.

After one of these three methods has been selected and the information required for the method has been entered, the software creates the coverage for the population of potential sites. Next, the main menu is displayed again, unless the random-selection with equal-area distribution method is being used.

When the random-selection with equal-area distribution method is used, additional prompting messages are displayed to elicit information used to create a coverage of subareas. These messages begin with the menu shown below.

You may define subareas using either of two methods:

Code	Method
------	--------

- |   |  |
|---|--|
| 1 | Specify the approximate number of subareas, or |
| 2 | Specify the width of subareas                  |

Enter a method code:

The methods for defining subareas are similar to the first two methods for defining a population. However, in this case square polygons are defined, rather than points. If the first method is selected, the following prompting message is displayed.

Enter approximate number of subareas (CR=10,000):

A positive integer should be entered. If carriage return is typed without entering a value, approximately 10,000 subarea polygons are created.

The number of subareas determines the resolution of the cell boundaries that are created when sites are selected. A smaller number of subareas causes the percent-difference for areas of different cells to be larger and the boundaries of the cells to be more jagged than would a larger number of subareas.

If the second method is selected, the following prompting message is displayed.

Enter width of each subarea:

A positive number should be entered in length units of the areal-subset coverage. With this method, the user explicitly controls the resolution of cells.

After one of these two methods is used to define the subareas, the software creates a coverage of the subareas and displays the main menu.

Normally, the population of potential sites is created only once for a study region. **WARNING:** If the user defines the population again, after site selection has been performed, the coverage containing the results of the previous site selections will be deleted. When the population-definition option is selected from the main menu after a population coverage has been created already, the software displays the following prompting message.

If you continue, the coverage containing site selections is deleted!  
Population coverage already exists, continue (CR=NO)?

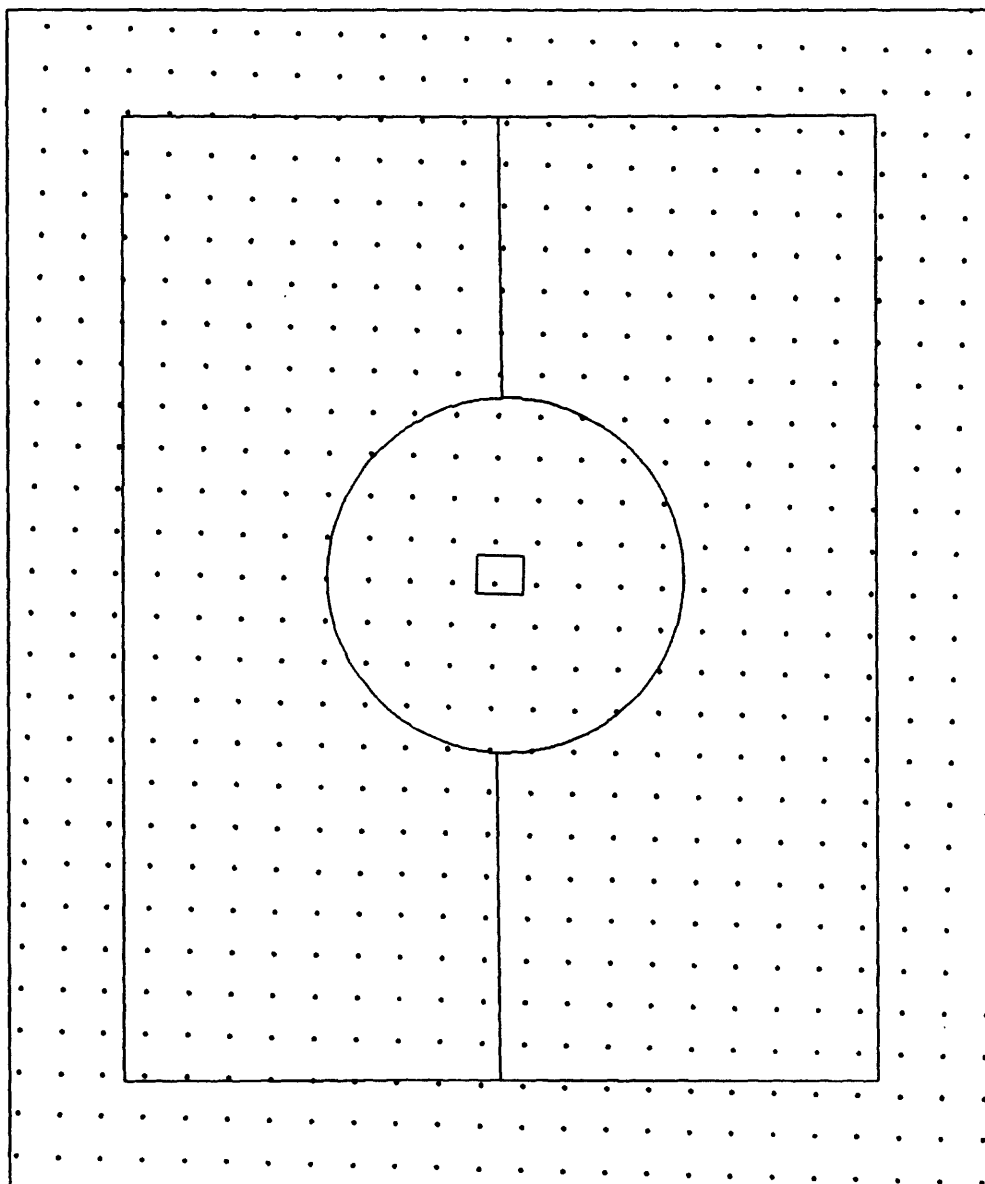
If a carriage return or a negative response is entered, the main menu is displayed again. If an affirmative response is entered, the coverage is deleted and the population is defined again.

The process for defining the population of potential sites is identical for all methods used for selecting sites. However, if the random-selection with equal-area distribution method has been specified and the population-definition option is selected from the main menu, the software creates a coverage of subareas in addition to the coverage of potential sites.

## Sample Run

In this sample, an equally-spaced population (fig. 10) is defined and a subarea coverage (fig. 11) is created. Site-selection method two, random selection with equal-area distribution is specified (line 11). Because this method is selected, the software creates a coverage of subareas. In the sample site-selection sessions presented later in this report, "Selection of Sites", all three methods are used to select sites from the population created in the following sample session.

```
( 1) Arc: NR SAMPLE
( 2)
( 3) Site selection may be performed using one of three methods:
( 4)
( 5)      Code      Method
( 6)
( 7)      1      Simple random selection
( 8)      2      Random selection with equal-area distribution
( 9)      3      Random selection with iterating grids
(10)
(11) Enter method code: 2
(12) Enter name of areal-subset coverage: LUSE
(13) Enter name of category variable: CATEGORY
(14)
(15) Checking areal-subset coverage specifications
(16) Submitting command RESUME NAWSOFT>SAMP>FINAL>ITEXST.RUN
(17) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(18) INFO user name is ARC
(19) INFO file name is LUSE.PAT
(20) Variable name is CATEGORY
(21)
(22) Site-selection software main menu
(23)
(24)      Code      Option
(25)
(26)      1      Define population
(27)      3      Exit from site-selection software
(28)
(29) Enter option code: 1
(30)
(31) Do you wish to exclude areal subsets using geometry (CR=NO)? Y
(32) Copied LUSE to LUSE.SUB
(33) Enter minimum area (CR=no minimum): 300
(34) Enter minimum area/perimeter ratio (CR=no minimum): <CR>
(35) Submitting command INFO
(36) INFO EXCHANGE CALL
(37) FRI, OCT 13 1989
(38) INFO 9.42 12/ 1/86 52.74.63*
(39) COPYRIGHT 1986 HENCO SOFTWARE, INC.
(40) PROPRIETARY TO HENCO SOFTWARE, INC.
```



EXPLANATION  
POTENTIAL SITE

Figure 10.--A population of potential sites for the hypothetical study region.

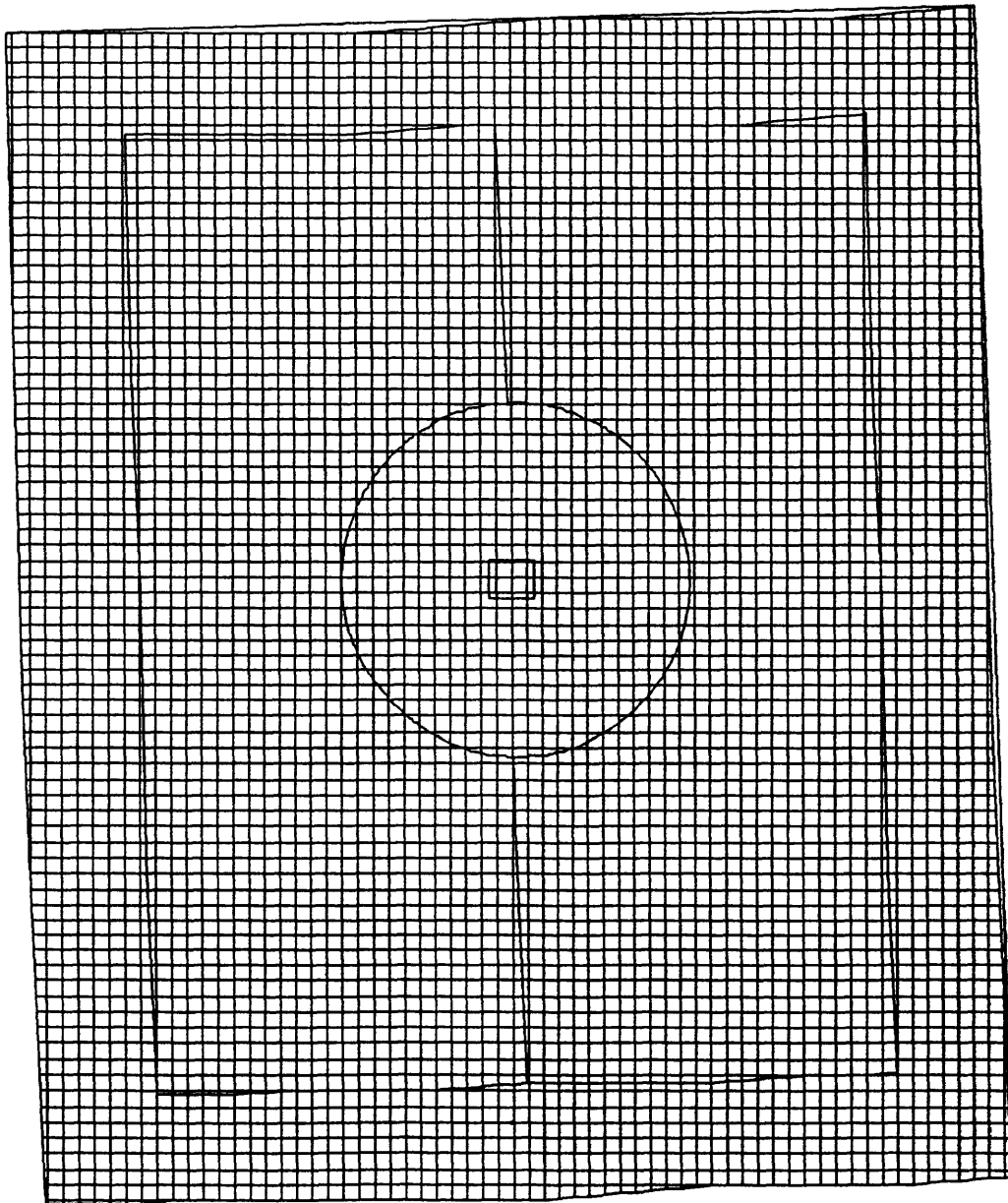


Figure 11.--Subareas of the hypothetical study region used to create cells for random selection of sites with equal-area distribution.

```

( 41)  US GOVT AGENCIES SEE USAGE RESTRICTIONS IN HELP FILES (HELP RESTRICTIONS)
( 42)  ENTER USER NAME>ARC
( 43)
( 44)  ENTER COMMAND >SELECT LUSE.SUB.PAT
( 45)        6 RECORD(S) SELECTED
( 46)
( 47)  ENTER COMMAND >RESELECT AREA LE 300
( 48)        2 RECORD(S) SELECTED
( 49)
( 50)  ENTER COMMAND >MOVE ' ' TO CATEGORY
( 51)
( 52)  ENTER COMMAND >QUIT STOP
( 53)
( 54)  You may define a population using any of three methods:
( 55)
( 56)  Code   Method
( 57)
( 58)      1   Specify the approximate number of potential sites
( 59)      2   Specify the distance between each potential site
( 60)      3   Specify an existing point coverage
( 61)
( 62)  Enter a method code: 1
( 63)  Enter approximate number of sites (CR=10,000): 700
( 64)  Defining population characteristics...
( 65)  Submitting command RESUME NAWSOFT>SAMP>FINAL>GRIDIM.RUN
( 66)  Boundary used from the coverage LUSE.SUB
( 67)  Method code is 1
( 68)  Approximate number is 700
( 69)  Building a population with 750 sites...
( 70)  (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
( 71)  All Rights Reserved Worldwide
( 72)  [GENERATE Version 5.0.1]
( 73)
( 74)
( 75)  Externalling BND and TIC...
( 76)
( 77)  Building points...
( 78)  Clipping LUSE.GRID with LUSE to create LUSE.POP.
( 79)  Overlaying points...
( 80)  Creating LUSE.POP.PAT...
( 81)  Overlaying annotation...
( 82)  Killed LUSE.GRID
( 83)  Producing identity of LUSE.GRID with LUSE.SUB to create LUSE.POP
( 84)  Overlaying points...
( 85)  Creating LUSE.POP.PAT...
( 86)  ** Item "AREA" duplicated, Join File version dropped **
( 87)
( 88)  ** Item "PERIMETER" duplicated, Join File version dropped **
( 89)
( 90)  ** Item "AREA" duplicated, Join File version dropped **
( 91)
( 92)  ** Item "PERIMETER" duplicated, Join File version dropped **
( 93)

```

```

( 94) Killed LUSE.GRID
( 95) Adding CHOSEN to LUSE.POP.PAT to produce LUSE.POP.PAT.
( 96) Adding UNIQUE to LUSE.POP.PAT to produce LUSE.POP.PAT.
( 97) Adding CELLNO to LUSE.POP.PAT to produce LUSE.POP.PAT.
( 98) Dropping LUSE.SUB# from LUSE.POP.PAT to create LUSE.POP.PAT
( 99) Dropping LUSE.SUB-ID from LUSE.POP.PAT to create LUSE.POP.PAT
(100) Dropping LUSE.GRID# from LUSE.POP.PAT to create LUSE.POP.PAT
(101) Dropping LUSE.GRID-ID from LUSE.POP.PAT to create LUSE.POP.PAT
(102) Submitting command INFO
(103) INFO EXCHANGE CALL
(104) FRI, OCT 13 1989
(105) INFO 9.42 12/ 1/86 52.74.63*
(106) COPYRIGHT 1986 HENCO SOFTWARE, INC.
(107) PROPRIETARY TO HENCO SOFTWARE, INC.
(108) US GOVT AGENCIES SEE USAGE RESTRICTIONS IN HELP FILES (HELP RESTRICTIONS)
(109) ENTER USER NAME>ARC
(110)
(111) ENTER COMMAND >SELECT LUSE.POP.PAT
(112) 656 RECORD(S) SELECTED
(113)
(114) ENTER COMMAND >CALC UNIQUE = $RECNO
(115)
(116) ENTER COMMAND >Q STOP
(117)
(118) Preparing category-index file, CATIDX, in INFO
(119) Submitting command RESUME NAWSOFT>SAMP>FINAL>DEFIDX.RUN
(120) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(121) Coverage of areal subsets is LUSE
(122) Coverage of population is LUSE.POP
(123) Category-variable name is CATEGORY
(124) Number of categories defined = 4
(125) Defining subareas...
(126)
(127) You may define subareas using either of two methods:
(128)
(129) Code Method
(130)
(131) 1 Specify the approximate number of subareas, or
(132) 2 Specify the width of subareas
(133)
(134) Enter a method code: 1
(135) Enter approximate number of subareas (CR=10,000): 5000
(136) Defining subarea characteristics...
(137) Submitting command RESUME NAWSOFT>SAMP>FINAL>GRIDIM.RUN
(138) Boundary used from the coverage LUSE.SUB
(139) Method code is 1
(140) Approximate number is 5000
(141) Preparing 5148 subareas...
(142) (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
(143) All Rights Reserved Worldwide
(144) [GENERATE Version 5.0.1]
(145)
(146)

```

```

(147) Externalling BND and TIC...
(148)
(149) Building polygons...
(150) Clipping LUSE.SUBA with LUSE to create LUSE.GRID.
(151) Sorting...
(152) Intersecting...
(153) Assembling polygons...
(154) Creating new labels...
(155) Creating NAWSOFT>SAMP>FINAL>LUSE.GRID.PAT...
(156) Overlaying annotation...
(157) Killed LUSE.SUBA
(158) Unioning LUSE.SUB with LUSE.SUBA to create LUSE.GRID
(159) Sorting...
(160) Intersecting...
(161) Assembling polygons...
(162) Creating new labels...
(163) Creating LUSE.GRID.PAT...
(164) ** Item "AREA" duplicated, Join File version dropped **
(165)
(166) ** Item "PERIMETER" duplicated, Join File version dropped **
(167)
(168) ** Item "AREA" duplicated, Join File version dropped **
(169)
(170) ** Item "PERIMETER" duplicated, Join File version dropped **
(171)
(172) Adding CELL to LUSE.GRID.PAT to produce LUSE.GRID.PAT.
(173) Adding VSTRIP to LUSE.GRID.PAT to produce LUSE.GRID.PAT.
(174) Dropping LUSE.SUB# from LUSE.GRID.PAT to create LUSE.GRID.PAT
(175) Dropping LUSE.SUB-ID from LUSE.GRID.PAT to create LUSE.GRID.PAT
(176) Dropping LUSE.SUBA# from LUSE.GRID.PAT to create LUSE.GRID.PAT
(177) Dropping LUSE.SUBA-ID from LUSE.GRID.PAT to create LUSE.GRID.PAT
(178) Killed LUSE.SUB
(179) Killed LUSE.SUBA
(180)
(181) Site-selection software main menu
(182)
(183) Code    Option
(184)
(185)      1    Define population
(186)      2    Select sites from population
(187)      3    Exit from site-selection software
(188)
(189) Enter option code: 3
(190) Arc:

```

The session begins with entry of the site-selection method code, the name of the areal-subset coverage, and the name of the category variable (lines 1-13). The specifications are checked by the software (lines 14-20). Next, the main-menu is displayed (lines 21-29) and the option code is entered.



The user excludes areal subsets with fewer than 300 coverage area units (lines 31-34). Operational messages are displayed during the geometric-exclusion process (lines 35-53). The small rectangular areal subset in the center of figure 7 is excluded. The population-definition menu is displayed (lines 54-62), and an equally-spaced population with approximately 700 sites is specified (lines 62-63). Operational and informative messages are displayed during creation of the population coverage (lines 64-117). Then, the category-index file is created (lines 118-124).

If the random-selection with equal-area distribution method had not been specified, the software would have displayed the main menu beginning on line 125. However in the sample session, the menu for defining subareas is displayed (lines 125-134). Approximately 5,000 subareas are specified (lines 134-135). Operational and informative messages are displayed during creation of the subarea coverage (lines 136-179). Finally, the main menu is displayed and the session is ended (lines 180-189).

## Selection of Sites

### Options

Selection of sites begins when the second option of the main menu, "Select sites from population", is chosen. Three prompting messages are displayed consecutively to elicit information about the site selection, as shown below. When the software displays these messages, the word CATEGORY is replaced by the user-specified category-variable name, and word the VALUE is replaced by the user's response to the first prompting message.

```
Enter value of CATEGORY for site selection (or QUIT):  
Enter number of primary sites to be selected when CATEGORY = VALUE:  
Enter number of alternate sites (CR = 2):
```

When the first prompting message is displayed, the name of the category from which sites will be selected should be entered. If a carriage return is typed without entering a category or the word QUIT is entered, the software stops and the "Arc:" prompting message is displayed.

When the second prompting message is displayed, the number of primary sites to be selected from the previously identified category should be entered. A positive integer is entered normally. If a negative or zero value is entered, the software stops and the "Arc:" prompting message is displayed.

When the third prompting message is displayed, the number of alternate sites to be selected should be entered. Typing a carriage return without entering a value causes two alternate sites to be chosen for each primary site selected.

After the user has responded to all three prompting messages, the user-specified category value is checked to ensure that at least one areal subset is identified with the category value. If not, an error message is displayed and the first of the three prompting messages is displayed again.

Next, a check is made to determine if sites have previously been selected for the user-specified category value. If so, the following prompting message is displayed.

Do you want to reselect sites for this category (CR=NO)?

**WARNING:** If an affirmative response is entered, the results of the previous site selection will be removed from the population coverage and sites will be selected for the category again. If a negative response is entered, the three prompting messages that begin site selection are displayed again, beginning with "Enter value of CATEGORY...".

If the simple random-selection or the random-selection with equal-area distribution method is used, the software begins the site selection process. If the random-selection with iterating grids method is used, three more prompting messages are displayed consecutively as shown below.

Enter fraction of cells to attain 3 sites for convergence, (CR = .75):  
Enter maximum number of iterations, (CR = 8):  
Enter damping factor, (CR = .90):

When the first prompting message is displayed, the software displays the number of rounds where the number three is shown above. The user enters a value between zero and one to complete the specifications for the secondary convergence criteria. A value of one disables secondary convergence. (See the "Random selection with iterating grids" section of this report.) If a carriage return is typed without entering a value, the fraction-of-cells variable is set to 0.75.

After the second prompting message is displayed, the user enters the maximum number of gridding iterations to be tried. If a carriage return is typed without entering a value, a maximum of eight iterations will be used.

If the maximum number of iterations is reached without finding a grid that meets the convergence criteria, the user is given the opportunity to enter different specifications for site selection. In this case, the prompting messages shown at the beginning of this section of the report ("Selection of sites") are displayed again.

After the third prompting message is displayed, the user enters a damping factor. The user-specified damping factor sometimes is used by the software for preparing a new grid after a failed iteration, as explained in the "Random selection with iterating grids" section of this report. If a carriage return is typed without entering a value, the user-specified damping factor is set to 0.90.

At the end of each failed iteration, the software displays informative messages pertaining to the results of the iteration. These informative messages can be used for specifying a different damping factor if the maximum number of iterations is reached without finding a satisfactory grid.

If the number of cells in the grid that contain at least  $\eta_c$  sites is consistently larger than the desired number of sites, a larger damping

factor or a larger number of alternate sites should be used. If the number of cells in the grid that contain  $\eta_r$  sites oscillates above and below the desired number of primary sites, the damping factor should be decreased.

### Sample run using simple random selection

In this sample, four sites are selected from the previously defined population using the simple random-selection method. Lines 1-29 differ from the previous sample session only where the site-selection method is specified on line 11.

```
( 1) Arc: &R SAMPLE
( 2)
( 3) Site selection may be performed using one of three methods:
( 4)
( 5)      Code      Method
( 6)
( 7)      1      Simple random selection
( 8)      2      Random selection with equal-area distribution
( 9)      3      Random selection with iterating grids
(10)
(11) Enter method code: 1
(12) Enter name of areal-subset coverage: LUSE
(13) Enter name of category variable: CATEGORY
(14)
(15) Checking areal-subset coverage specifications
(16) Submitting command RESUME NAWSOFT>SAMP>FINAL>ITEXST.RUN
(17) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(18) INFO user name is ARC
(19) INFO file name is LUSE.PAT
(20) Variable name is CATEGORY
(21)
(22) Site-selection software main menu
(23)
(24)      Code      Option
(25)
(26)      1      Define population
(27)      2      Select sites from population
(28)      3      Exit from site-selection software
(29)
(30) Enter option code: 2
(31)
(32) Ready for site selection...
(33)
(34) Enter value of CATEGORY for site selection (CR = QUIT): URBAN
(35) Enter number of primary sites to be selected when CATEGORY = URBAN: 4
(36) Enter number of alternate sites (CR = 2): <CR>
(37) Submitting command RESUME NAWSOFT>SAMP>FINAL>CHKIDX.RUN
(38) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(39) Category is URBAN
(40) Submitting command RESUME NAWSOFT>SAMP>FINAL>PICKEM.RUN
(41) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
```

```

( 42) Coverage of population is LUSE.POP
( 43) Selection expression is CATEGORY EQ 'URBAN'
( 44) Number of primary sites is 4
( 45) Number of alternate sites is 2
( 46) 178 sites in the cell
( 47) Site selection completed....Updating index file...
( 48) Submitting command RESUME NAWSOFT>SAMP>FINAL>UPTIDX.RUN
( 49) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 50) Category is URBAN
( 51) Number of primary sites is 4
( 52) Preparing plot file: #LUSE.URBAN
( 53) (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
( 54) All Rights Reserved Worldwide
( 55) ARCPL0T Version 5.0.1
( 56)
( 57) Enter Plot filename #LUSE.URBAN
( 58) LUSE.POP points : 12 of 656 selected.
( 59) Leaving ARCPL0T...
( 60)
( 61) Saving locations of selected sites...
( 62) Submitting command RESUME NAWSOFT>SAMP>FINAL>CATRPT.RUN
( 63) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 64) Coverage of population is LUSE.POP
( 65) Category-variable name is CATEGORY
( 66) Category is URBAN
( 67) Cell-variable name is
( 68) Projecting to latitude-longitude...
( 69) Locations projected.....formatting report...
( 70) Submitting command RESUME NAWSOFT>SAMP>FINAL>FCATRP.RUN
( 71) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 72) Category is URBAN
( 73) Number of primary sites is 4
( 74) Number of alternate sites is 2
( 75)
( 76) Report is ready in the file: SITE.REPORT.URBAN
( 77)
( 78) Site-selection software main menu
( 79)
( 80) Code    Option
( 81)
( 82)      1    Define population
( 83)      2    Select sites from population
( 84)      3    Exit from site-selection software
( 85)
( 86) Enter option code: 3
( 87) Arc:

```

The main menu is displayed and the option to select sites is specified (lines 21-30). The user specifies selection of four sites from category "URBAN" with two alternate sites each (lines 34-36).

Operational and informative messages are displayed on lines 37-76. The first operation (lines 37-39) checks that a valid category value was entered

and that a sufficient number of potential sites exist. The second operation randomly selects sites (lines 40-46). The third operation updates the category-index file (lines 47-51). The fourth operation writes the binary plot file (lines 52-59). The fifth operation prepares the site-selection report (lines 60-76). The main menu is displayed again on lines 78-86. The plot file created during this sample session is shown in figure 9.

#### Sample run using random selection with equal-area distribution

Four sites in category "MIXED" are chosen using the random-selection with equal-area distribution method. The first 29 lines of this sample are the same as the sample presented for defining the population. The option to select sites is chosen (line 30). Again, four sites with two alternate sites each are specified (lines 35-36).

```
( 1) Arc: &R SAMPLE
( 2)
( 3) Site selection may be performed using one of three methods:
( 4)
( 5)   Code    Method
( 6)
( 7)     1      Simple random selection
( 8)     2      Random selection with equal-area distribution
( 9)     3      Random selection with iterating grids
(10)
(11) Enter method code: 2
(12) Enter name of areal-subset coverage: LUSE
(13) Enter name of category variable: CATEGORY
(14)
(15) Checking areal-subset coverage specifications
(16) Submitting command RESUME NAWSOFT>SAMP>FINAL>ITEXST.RUN
(17) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(18) INFO user name is ARC
(19) INFO file name is LUSE.PAT
(20) Variable name is CATEGORY
(21)
(22) Site-selection software main menu
(23)
(24)   Code    Option
(25)
(26)     1      Define population
(27)     2      Select sites from population
(28)     3      Exit from site-selection software
(29)
(30) Enter option code: 2
(31)
(32) Ready for site selection...
(33)
(34) Enter value of CATEGORY for site selection (CR = QUIT): MIXED
(35) Enter number of primary sites to be selected when CATEGORY = MIXED: 4
(36) Enter number of alternate sites (CR = 2): <CR>
(37) Submitting command RESUME NAWSOFT>SAMP>FINAL>CHKIDX.RUN
```

```

( 38) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 39) Category is MIXED
( 40)
( 41) Creating 4 equal-area cells
( 42) Submitting command RESUME NAWSOFT>SAMP>FINAL>EQARSA.RUN
( 43) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 44) Coverage of subarea polygons is LUSE.SUBA
( 45) Category-variable name is CATEGORY
( 46) Category is MIXED
( 47) Number of primary sites is 4
( 48) Number of alternate sites is 2
( 49) Number of vertical strips =                2
( 50)      Total area          =          47749.63
( 51)      Strip area         =          23874.81
( 52)      Equal area for cell =          11937.41 (25.0%)
( 53)
( 54)      Equal-area grid cells:
( 55)
( 56)      Cell      Area      Percentage
( 57)      1      11938.7      (25.0%)
( 58)      2      11937.5      (25.0%)
( 59)      3      11952.0      (25.0%)
( 60)      4      11921.4      (25.0%)
( 61)      Producing identity of LUSE.POP with LUSE.SUBA to create LUSE.SAM
( 62)      Overlaying points...
( 63)      Creating LUSE.SAM.PAT...
( 64)      ** Item "AREA" duplicated, Join File version dropped **
( 65)
( 66)      ** Item "PERIMETER" duplicated, Join File version dropped **
( 67)
( 68)      ** Item "AREA" duplicated, Join File version dropped **
( 69)
( 70)      ** Item "PERIMETER" duplicated, Join File version dropped **
( 71)
( 72)      ** Item "CATEGORY" duplicated, Join File version dropped **
( 73)
( 74) Preparing coverage of equal-area cells
( 75)      Dissolving LUSE.SUBA by CELL to create LUSE.CELL.MIXED
( 76)      Creating LUSE.CELL.MIXED.PAT format...
( 77)      Creating dissolve table...
( 78)      Dissolving...
( 79)      Number of Polygons (Input,Output) =          5168          5
( 80)      Number of Arcs      (Input,Output) =          10473          9
( 81)      Creating LUSE.CELL.MIXED.PAT...
( 82)      Selecting sites from cell #1
( 83)
( 84) Submitting command RESUME NAWSOFT>SAMP>FINAL>PICKEM.RUN
( 85) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 86) Coverage of population is LUSE.SAM
( 87) Selection expression is CELL = 1 AND CATEGORY EQ 'MIXED'
( 88) Number of primary sites is 1
( 89) Number of alternate sites is 2
( 90)      38 sites in the cell

```

```

( 91)  Selecting sites from cell #2
( 92)
( 93)  Submitting command RESUME NAWSOFT>SAMP>FINAL>PICKEM.RUN
( 94)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 95)  Coverage of population is LUSE.SAM
( 96)  Selection expression is CELL = 2 AND CATEGORY EQ 'MIXED'
( 97)  Number of primary sites is 1
( 98)  Number of alternate sites is 2
( 99)    49 sites in the cell
(100)  Selecting sites from cell #3
(101)
(102)  Submitting command RESUME NAWSOFT>SAMP>FINAL>PICKEM.RUN
(103)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(104)  Coverage of population is LUSE.SAM
(105)  Selection expression is CELL = 3 AND CATEGORY EQ 'MIXED'
(106)  Number of primary sites is 1
(107)  Number of alternate sites is 2
(108)    44 sites in the cell
(109)  Selecting sites from cell #4
(110)
(111)  Submitting command RESUME NAWSOFT>SAMP>FINAL>PICKEM.RUN
(112)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(113)  Coverage of population is LUSE.SAM
(114)  Selection expression is CELL = 4 AND CATEGORY EQ 'MIXED'
(115)  Number of primary sites is 1
(116)  Number of alternate sites is 2
(117)    48 sites in the cell
(118)  Site selection completed....Updating index file...
(119)  Submitting command RESUME NAWSOFT>SAMP>FINAL>UPTIDX.RUN
(120)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(121)  Category is MIXED
(122)  Number of primary sites is 4
(123)  Recording results in population coverage
(124)  Submitting command INFO
(125)    INFO EXCHANGE CALL
(126)  MON, OCT 16 1989
(127)    INFO 9.42 12/ 1/86 52.74.63*
(128)    COPYRIGHT 1986 HENCO SOFTWARE, INC.
(129)    PROPRIETARY TO HENCO SOFTWARE, INC.
(130)    US GOVT AGENCIES SEE USAGE RESTRICTIONS IN HELP FILES (HELP RESTRICTIONS)
(131)    ENTER USER NAME>ARC
(132)
(133)    ENTER COMMAND >SELECT LUSE.SAM.PAT
(134)      856 RECORD(S) SELECTED
(135)
(136)    ENTER COMMAND >RELATE LUSE.POP.PAT UNIQUE
(137)
(138)    ENTER COMMAND >RESELECT CATEGORY = 'MIXED'
(139)      179 RECORD(S) SELECTED
(140)
(141)    ENTER COMMAND >CALC $1CELLNO = CELL
(142)
(143)    ENTER COMMAND >CALC $1CHOSEN = CHOSEN

```

```

(144)
(145) ENTER COMMAND >Q STOP
(146) Preparing plot file: #LUSE.MIXED
(147) (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
(148) All Rights Reserved Worldwide
(149) ARCPLLOT Version 5.0.1
(150)
(151) Enter Plot filename #LUSE.MIXED
(152) LUSE.SAM points : 12 of 656 selected.
(153) Leaving ARCPLLOT...
(154)
(155) Saving locations of selected sites...
(156) Submitting command RESUME NAWSOFT>SAMP>FINAL>CATRPT.RUN
(157) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(158) Coverage of population is LUSE.SAM
(159) Category-variable name is CATEGORY
(160) Category is MIXED
(161) Cell-variable name is CELL
(162) Projecting to latitude-longitude...
(163) Locations projected.....formatting report...
(164) Submitting command RESUME NAWSOFT>SAMP>FINAL>FCATRP.RUN
(165) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(166) Category is MIXED
(167) Number of primary sites is 4
(168) Number of alternate sites is 2
(169)
(170) Report is ready in the file: SITE.REPORT.MIXED
(171)
(172) Site-selection software main menu
(173)
(174) Code    Option
(175)
(176)      1    Define population
(177)      2    Select sites from population
(178)      3    Exit from site-selection software
(179)
(180) Enter option code: 3
(181) Arc:

```

All of the operations during the simple random site-selection sample session are performed in this sample; however, some additional operations are performed. As before, the user's specifications are checked (lines 37-39). The next operation aggregates the subareas into equal-area cells (lines 40-60). The results of computations for the desired areas of strips and cells are displayed (lines 49-52). Next, the actual areas of the cells prepared by aggregating the subareas are shown (lines 53-60). The next group of operational messages (lines 61-72) are displayed while the software determines the cell in which each potential site is located. A coverage of the cells is created (lines 73-81).

Next, one primary and two alternate sites are selected from each cell. This operation is repeated for the user-specified number of primary sites (lines 82-90, 91-99, 100-108, and 109-117). The category-index file is

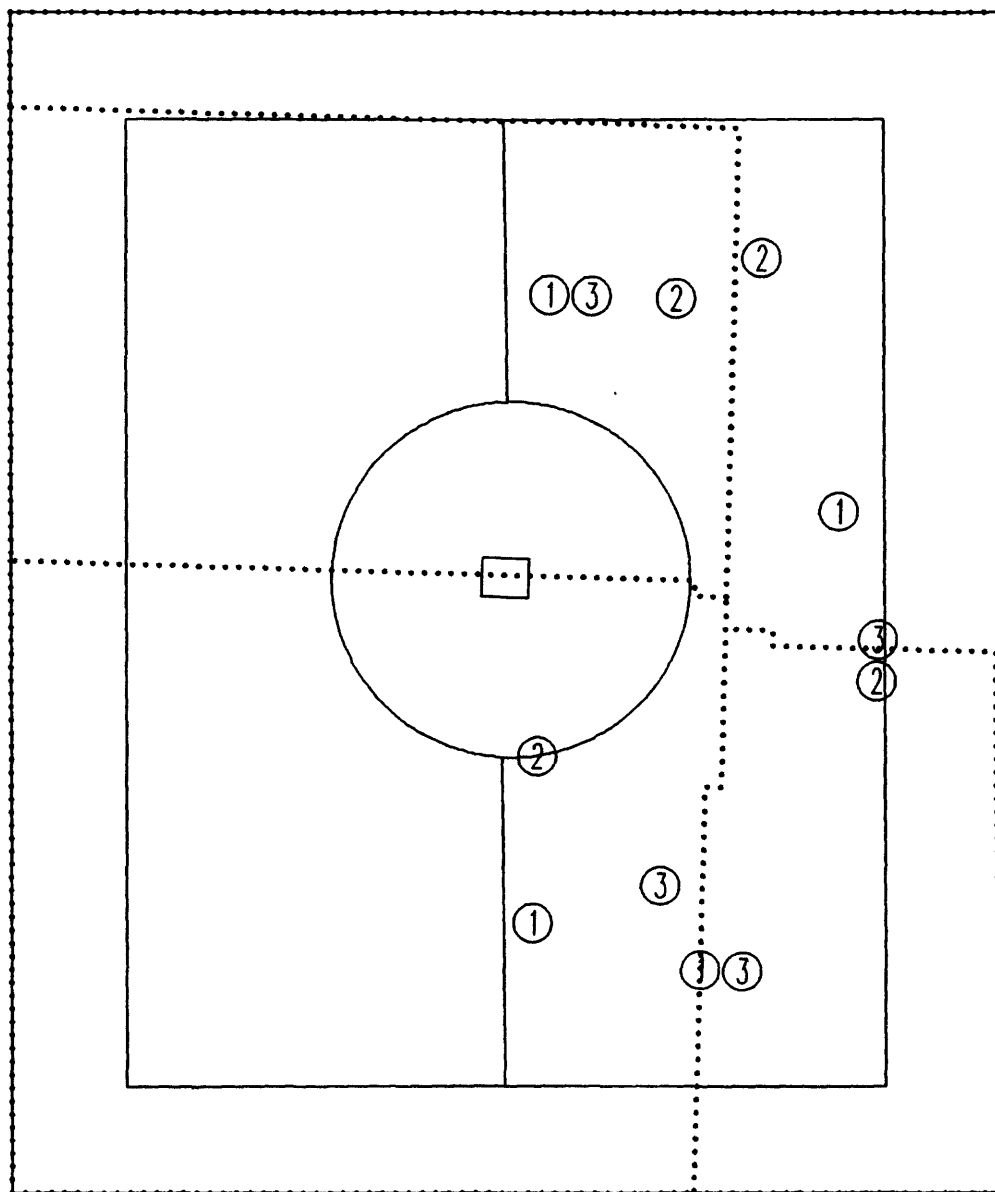


updated (lines 118-122), the results of the site selections are saved in the population coverage (lines 123-145), a plot file is prepared (lines 146-153), and a site-selection report is written (lines 154-170). Finally, the main menu is displayed again (lines 171-180). The cells created and the sites selected for category "MIXED" are shown in figure 12.

### Sample run using random selection with iterating grids

This sample shows use of the random-selection with iterating grids method to select four sites from category "RURAL" (lines 1-35). Again, two alternate sites are specified for each primary site selected (lines 35-36). As with the two previous samples, the user's specifications are checked (lines 37-39). Next, the three iteration-control variables are entered (lines 40-42).

```
( 1) Arc: &R SAMPLE
( 2)
( 3) Site selection may be performed using one of three methods:
( 4)
( 5)   Code    Method
( 6)
( 7)       1    Simple random selection
( 8)       2    Random selection with equal-area distribution
( 9)       3    Random selection with iterating grids
(10)
(11) Enter method code: 3
(12) Enter name of areal-subset coverage: LUSE
(13) Enter name of category variable: CATEGORY
(14)
(15) Checking areal-subset coverage specifications
(16) Submitting command RESUME NAWSOFT>SAMP>FINAL>ITEXST.RUN
(17) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(18) INFO user name is ARC
(19) INFO file name is LUSE.PAT
(20) Variable name is CATEGORY
(21)
(22) Site-selection software main menu
(23)
(24)   Code    Option
(25)
(26)       1    Define population
(27)       2    Select sites from population
(28)       3    Exit from site-selection software
(29)
(30) Enter option code: 2
(31)
(32) Ready for site selection...
(33)
(34) Enter value of CATEGORY for site selection (CR = QUIT): RURAL
(35) Enter number of primary sites to be selected when CATEGORY = RURAL: 4
(36) Enter number of alternate sites (CR = 2): <CR>
```



EXPLANATION

- ... CELL BOUNDARY
- SELECTED SITE, ROUND NUMBER SHOWN INSIDE CIRCLE

Figure 12.--Results of selecting sites from the "MIXED" category using random selection with equal-area distribution.

```

( 37) Submitting command RESUME NAWSOFT>SAMP>FINAL>CHKIDX.RUN
( 38) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 39) Category is RURAL
( 40) Enter fraction of cells to attain 3 sites for convergence, (CR = .75): <CR>
( 41) Enter maximum number of iterations, (CR = 8): 10
( 42) Enter damping factor, (CR = .90): 1.5
( 43) Making an initial guess at grid characteristics...
( 44) Submitting command RESUME NAWSOFT>SAMP>FINAL>NFLATE.RUN
( 45) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 46) Category is RURAL
( 47) Number of primary sites is 4
( 48) Number of alternate sites is 2
( 49) Defining cell characteristics...
( 50) Submitting command RESUME NAWSOFT>SAMP>FINAL>GRIDIM.RUN
( 51) Boundary used from the coverage LUSE
( 52) Method code is 1
( 53) Approximate number is 8
( 54)
( 55) Constructing a grid with 12 cells...
( 56) (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
( 57) All Rights Reserved Worldwide
( 58) [GENERATE Version 5.0.1]
( 59)
( 60)
( 61) Externalling BND and TIC...
( 62)
( 63) Building polygons...
( 64) Killed LUSE.SAM
( 65) Producing identity of LUSE.POP with LUSE.CELL.RURAL to create LUSE.SAM
( 66) Overlaying points...
( 67) Creating LUSE.SAM.PAT...
( 68) ** Item "AREA" duplicated, Join File version dropped **
( 69)
( 70) ** Item "PERIMETER" duplicated, Join File version dropped **
( 71)
( 72) ** Item "AREA" duplicated, Join File version dropped **
( 73)
( 74) ** Item "PERIMETER" duplicated, Join File version dropped **
( 75)
( 76) Checking for convergence...
( 77) Submitting command RESUME NAWSOFT>SAMP>FINAL>SAMINF.RUN
( 78) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
( 79) Coverage of iteration results is LUSE.SAM
( 80) Cell-variable name is LUSE.CELL.RURAL-ID
( 81) Category-variable name is CATEGORY
( 82) Category is RURAL
( 83) Number of primary sites is 4
( 84) Current number of cells is 12
( 85) Number of alternate sites is 2
( 86) Percentage required for convergence is .75
( 87) Damping factor is 1.5

```

```

( 88) *****
( 89) Iteration failed to converge using    12 cells
( 90) Average populated cell contained 18.6 sites
( 91)    16 cells contained at least    3 sites.
( 92) Next iteration will use at least      3 cells
( 93) *****
( 94) Convergence failed on iteration #1
( 95) Defining cell characteristics...
( 96)
.
.
.
(134)
(135) *****
(136) Iteration failed to converge using     4 cells
(137) Average populated cell contained 33.1 sites
(138)    9 cells contained at least    3 sites.
(139) Next iteration will use at least      2 cells
(140) *****
(141) Convergence failed on iteration #2
(142) Defining cell characteristics...
(143)
.
.
.
(181)
(182) *****
(183) Iteration failed to converge using     4 cells
(184) Average populated cell contained 49.7 sites
(185)    6 cells contained at least    3 sites.
(186) Next iteration will use at least      3 cells
(187) *****
(188) Convergence failed on iteration #9
(189) Defining cell characteristics...
(190)
.
.
.
(228)
(229) *****
(230) Iteration failed to converge using     4 cells
(231) Average populated cell contained 33.1 sites
(232)    9 cells contained at least    3 sites.
(233) Next iteration will use at least      2 cells
(234) *****
(235) Convergence failed on iteration #4
(236) Defining cell characteristics...
(237) Submitting command RESUME NAWSOFT>SAMP>FINAL>GRIDIM.RUN
(238) Boundary used from the coverage LUSE
(239) Method code is 1
(240) Approximate number is 2

```

(241)  
 (242) *Constructing a grid with 2 cells...*  
 (243) Killed LUSE.CELL.RURAL  
 (244) (C) 1988 - 1990 Environmental Systems Research Institute, Inc.  
 (245) All Rights Reserved Worldwide  
 (246) [GENERATE Version 5.0.1]  
 (247)  
 (248)  
 (249) Externalling BND and TIC  
 (250)  
 (251) Building polygons...  
 (252) Killed LUSE.SAM  
 (253) Producing identity of LUSE.POP with LUSE.CELL.RURAL to create LUSE.SAM  
 (254) Overlaying points...  
 (255) Creating LUSE.SAM.PAT...  
 (256) \*\* Item "AREA" duplicated, Join File version dropped \*\*  
 (257)  
 (258) \*\* Item "PERIMETER" duplicated, Join File version dropped \*\*  
 (259)  
 (260) \*\* Item "AREA" duplicated, Join File version dropped \*\*  
 (261)  
 (262) \*\* Item "PERIMETER" duplicated, Join File version dropped \*\*  
 (263)  
 (264) *Checking for convergence...*  
 (265) Submitting command RESUME NAWSOFT>SAMP>FINAL>SAMINF.RUN  
 (266) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO  
 (267) Coverage of iteration results is LUSE.SAM  
 (268) Cell-variable name is LUSE.CELL.RURAL-ID  
 (269) Category-variable name is CATEGORY  
 (270) Category is RURAL  
 (271) Number of primary sites is 4  
 (272) Current number of cells is 2  
 (273) Number of alternate sites is 2  
 (274) Percentage required for convergence is .75  
 (275) Damping factor is 1.5  
 (276) *Site selection completed....Updating index file...*  
 (277) Submitting command RESUME NAWSOFT>SAMP>FINAL>UPTIDX.RUN  
 (278) Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO  
 (279) Category is RURAL  
 (280) Number of primary sites is 4  
 (281) *Recording results in population coverage*  
 (282) Submitting command INFO  
 (283) INFO EXCHANGE CALL  
 (284) MON, OCT 16 1989  
 (285) INFO 9.42 12/ 1/86 52.74.63\*  
 (286) COPYRIGHT 1986 HENCO SOFTWARE, INC.  
 (287) PROPRIETARY TO HENCO SOFTWARE, INC.  
 (288) US GOVT AGENCIES SEE USAGE RESTRICTIONS IN HELP FILES (HELP RESTRICTIONS)  
 (289) ENTER USER NAME>ARC  
 (290)

```

(291)  ENTER COMMAND >SELECT LUSE.SAM.PAT
(292)      656 RECORD(S) SELECTED
(293)
(294)  ENTER COMMAND >RELATE LUSE.POP.PAT UNIQUE
(295)
(296)  ENTER COMMAND >RESELECT CATEGORY = 'RURAL'
(297)      298 RECORD(S) SELECTED
(298)
(299)  ENTER COMMAND >CALC $1CELLNO = LUSE.CELL.RURAL-ID
(300)
(301)  ENTER COMMAND >CALC $1CHOSEN = CHOSEN
(302)
(303)  ENTER COMMAND >Q STOP
(304)  Preparing plot file: #LUSE.RURAL
(305)  (C) 1988 - 1990 Environmental Systems Research Institute, Inc.
(306)  All Rights Reserved Worldwide
(307)  ARCPLLOT Version 5.0.1
(308)
(309)  Enter Plot filename #LUSE.RURAL
(310)  LUSE.SAM points : 12 of 656 selected.
(311)  Leaving ARCPLLOT...
(312)
(313)  Saving locations of selected sites...
(314)  Submitting command RESUME NAWSOFT>SAMP>FINAL>CATRPT.RUN
(315)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(316)  Coverage of population is LUSE.SAM
(317)  Category-variable name is CATEGORY
(318)  Category is RURAL
(319)  Cell-variable name is LUSE.CELL.RURAL-ID
(320)  Projecting to latitude-longitude...
(321)  Locations projected.....formatting report...
(322)  Submitting command RESUME NAWSOFT>SAMP>FINAL>FCATRP.RUN
(323)  Directory of INFO data base is NAWSOFT>SAMP>FINAL>INFO
(324)  Category is RURAL
(325)  Number of primary sites is 4
(326)  Number of alternate sites is 2
(327)
(328)  Report is ready in the file: SITE.REPORT.RURAL
(329)
(330)  Site-selection software main menu
(331)
(332)      Code      Option
(333)
(334)          1      Define population
(335)          2      Select sites from population
(336)          3      Exit from site-selection software
(337)
(338)  Enter option code: 3
(339)  Arc:

```

The characteristics of the first grid are calculated (lines 43-53), the grid coverage is prepared (lines 54-74), and the grid is compared to the user-specified convergence criteria (lines 75-87). Convergence is not

attained on the first iteration and informative messages are displayed (lines 88-94).

The amount of time required to select sites and number of lines displayed on the terminal can be large because the site-selection process with this method is iterative. A satisfactory grid was found for this sample problem on the fifth iteration. To avoid unnecessary repetition in this report, lines 96-134, 143-181, and 190-228 were omitted. These lines contain operational and informative messages for the second, third, and fourth iterations similar to those printed for the first iteration (lines 50-87).

The operational and informative messages displayed during the fifth iteration (lines 236-275) are similar to messages displayed during the first iteration. The remainder of the sample session is similar to the previous sample. The category-index file is updated (lines 276-280), the site selections are stored in the population coverage (lines 281-303), the plot file is written (lines 304-311), and the site-selection report is prepared (lines 312-328), and the main menu is displayed (lines 330-338).

The cells created and the sites selected during this sample session are shown in figure 13.

#### Site Selection for the Central Oklahoma Aquifer Study Region

The site-selection software has been tested and applied to several hydrologic studies. Water-quality site selection for the Central Oklahoma aquifer NAWQA project (Christenson and Parkhurst, 1987) was performed using the software. Areal subsets were identified by hydrogeologic unit. An equally spaced population of potential sites was created for the study region and sites were selected using the random-selection with equal-area distribution method. The existing well suitable for sampling nearest the selected location was found in the field and used for water-quality sampling. This technique was workable because the population of existing wells in central Oklahoma is large and evenly distributed throughout the study region.

A three-dimensional network was designed for the Central Oklahoma NAWQA study region by using three areal-subset coverages. The first areal-subset coverage identified alluvium and terrace deposits. The second areal-subset coverage contained shallow and medium depth hydrogeologic units, and the third coverage contained deep hydrogeologic units. Site selection was performed separately with each coverage.

Alluvium and terrace deposits overlie approximately one third of the study area. An equally spaced population of potential sites (fig. 14) was created using the site-selection software. Sixty primary sites for the alluvium and terrace deposits were selected. Two alternate sites were selected for each primary site. Thus, 3 sites were selected from each of 60 equal-area cells (fig. 15).

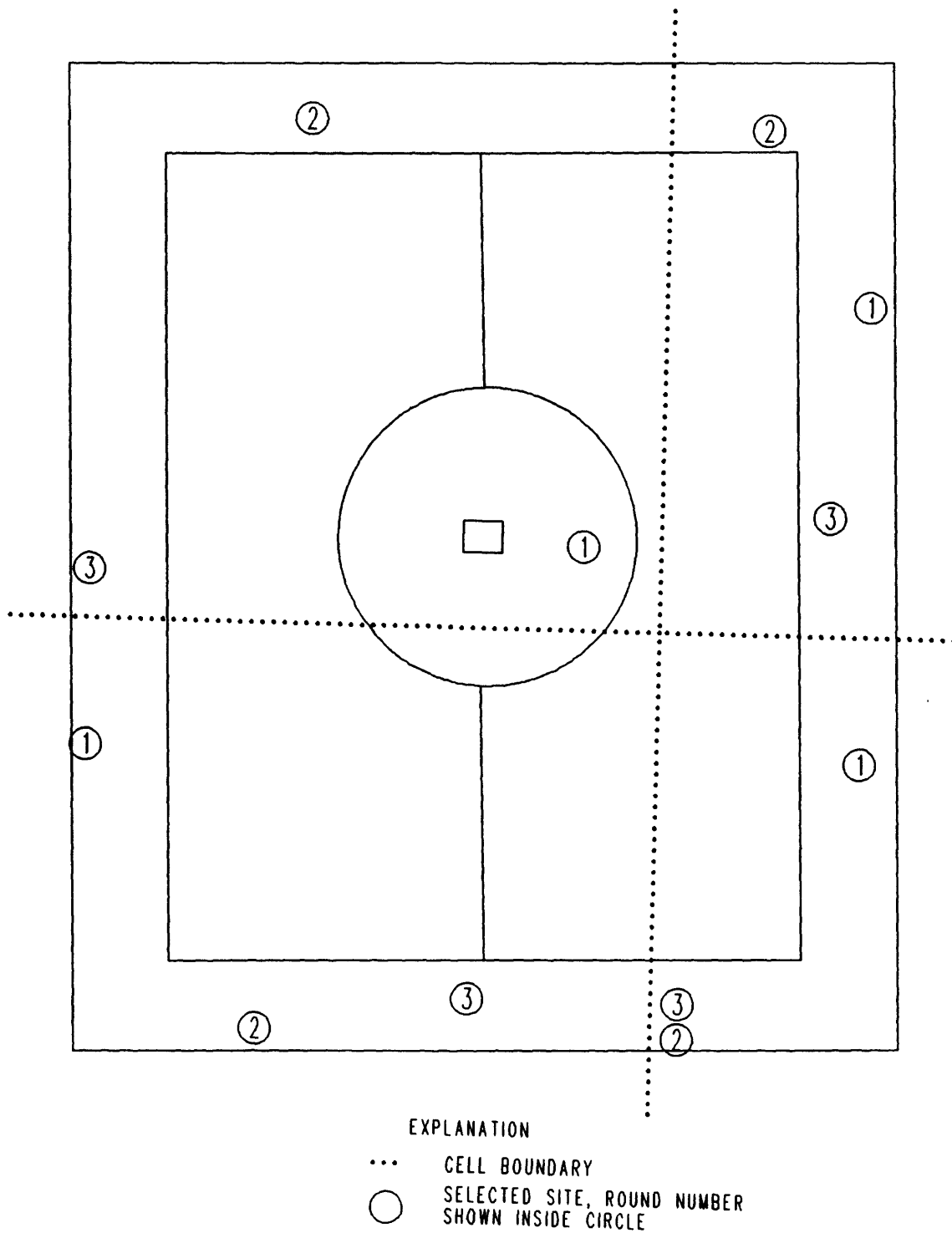


Figure 13.--Results of selecting sites from the "RURAL" category using random selection with iterating grids.



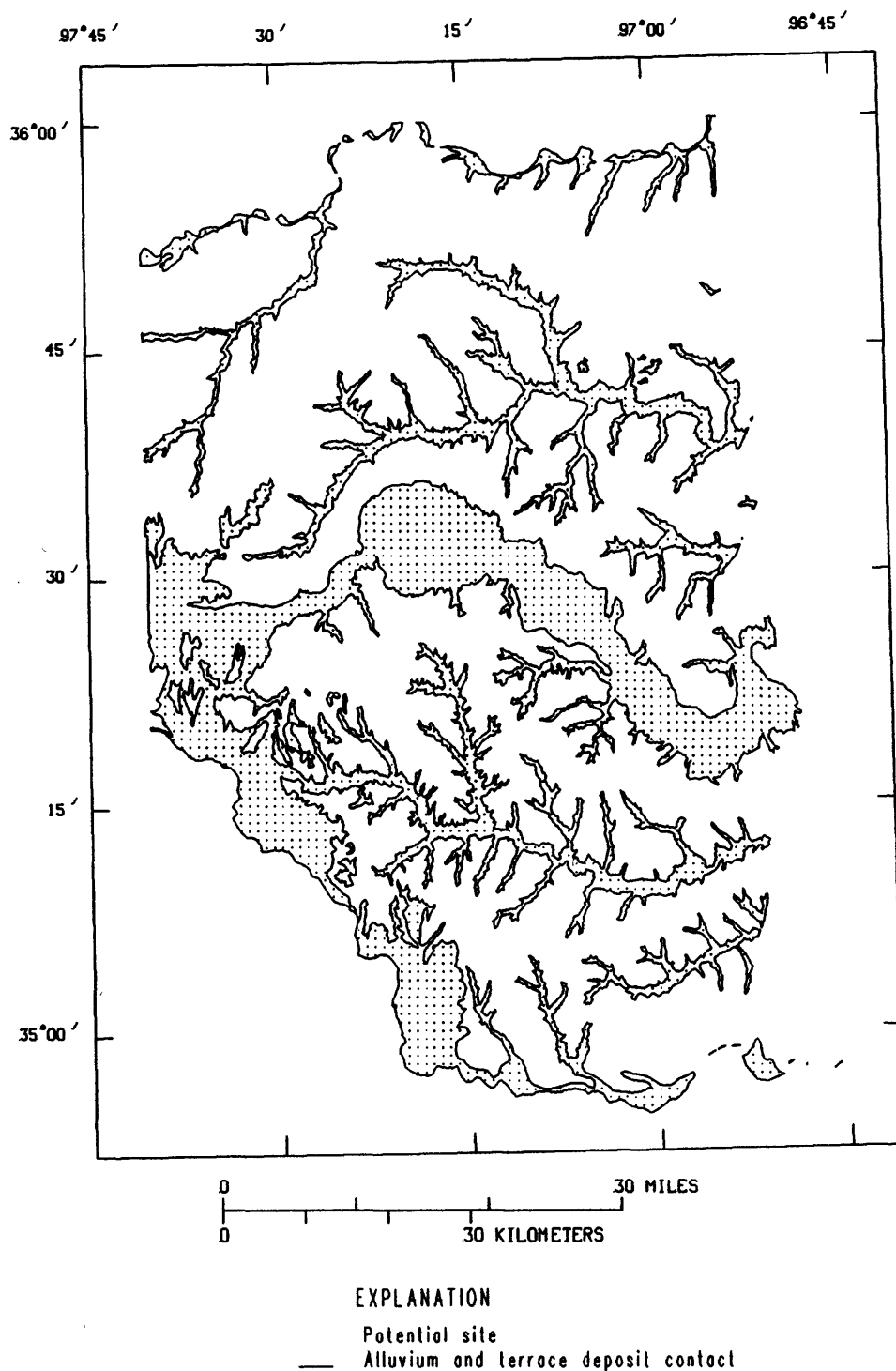


Figure 14.--Equally spaced population of potential sites in the alluvium and terrace deposits of the Central Oklahoma aquifer study unit.

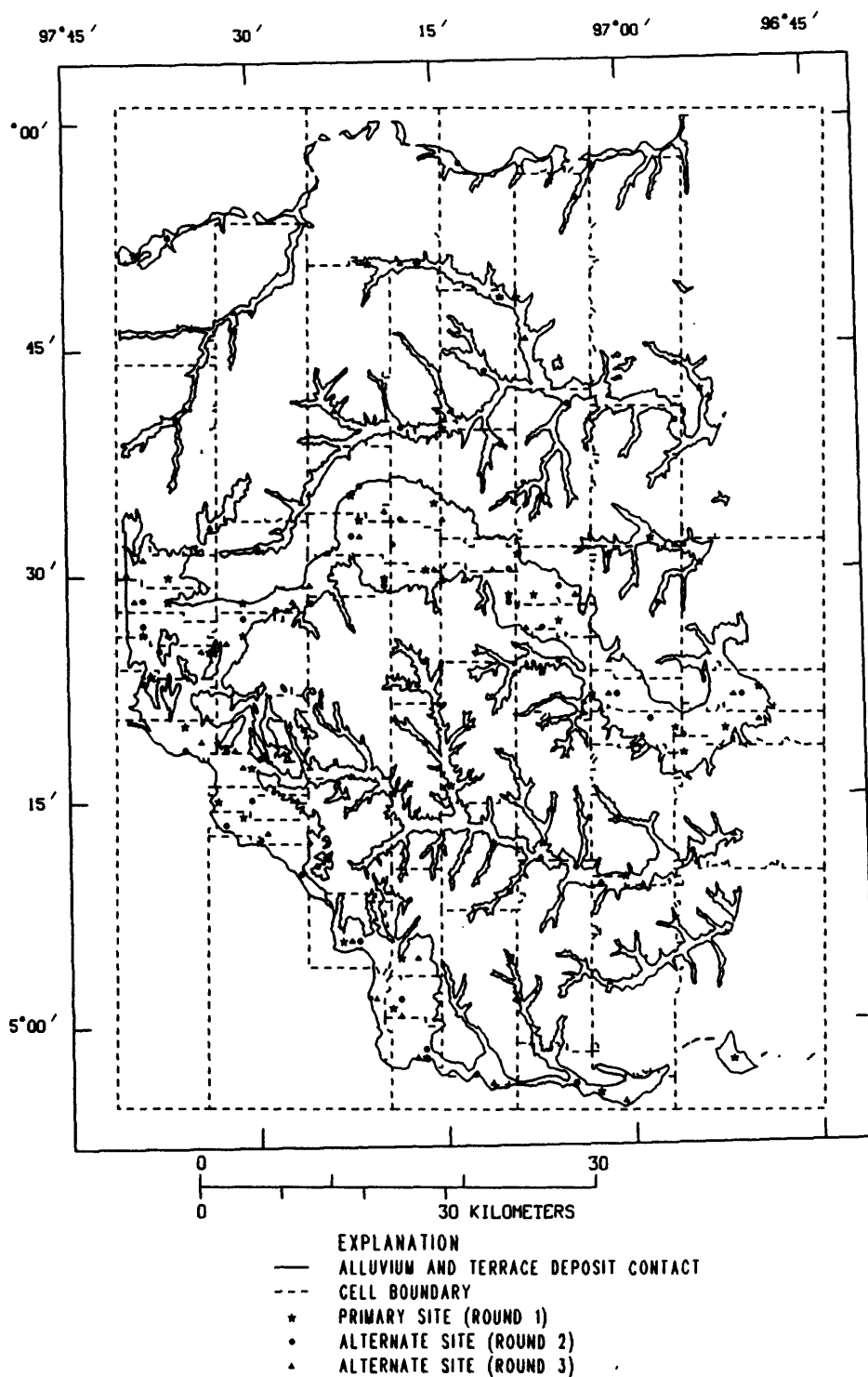


Figure 15.--Site selections in the alluvium and terrace deposits for the Central Oklahoma aquifer study unit.

## SOFTWARE DESIGN

### Installation of the Software

The SAMPLE.AML program uses 11 Fortran programs to perform various operations during the population-definition and site-selection processes (table 3). Four of the Fortran programs contain the \$INSERT compiler directive to insert the statements presented in attachment B. The inserted statements define five integer variables that limit the maximum sizes of arrays used in the software. The values of the five variables may need to be increased in order to use the software for some problems. The values for the first four variables can be determined by examining the characteristics of the site-selection problem. There is no general method for determining in advance the required size for the fifth variable.

If any of the array limits are reached when the software is running, an error message is printed and the program stops. If this occurs, the value of the corresponding array limit should be increased, and the affected Fortran programs should be recompiled and linked. Then the site-selection software can be run again.

All 11 Fortran programs should be compiled and linked before the site-selection software is used on a computer system. A Command Procedure Language (CPL) (Landy, 1982) program for compiling and linking the programs on a Prime computer system is shown below. To use the program, the statements in attachment B need to be stored in a file named SAMPLE.INSERT, the subroutines in attachment C need to be stored in a file named SUBS.F77, and the programs in attachments D-M need to be stored in files named by joining together the name of the program and the characters ".F77".

```
F77 SUBS
&DO PROG &LIST ITEXST CATRPT DEFIDX EQARSA FCATRP GRIDIM NFLATE PICKEM ~
          SAMINF UPTIDX CHKIDX
F77 %PROG%
&DATA BIND
  LO %PROG%
  LO SUBS
  LI ARC>LIB>ARCLIB
  LI VAPPLB
  LI VSRTLI
  LI
  FILE
&END
DELETE %PROG%.BIN
&END
&RETURN
```

Table 3.--*Descriptions of Fortran programs used in site-selection software*

[An "X" in the "Inserted statements" column indicates the programming statements in attachment B are inserted into the program.]

Program name	Attachment	Inserted statements	Purpose
ITEXST	D		Check if INFO file for areal-subset coverage can be opened, and if the user-specified category-variable name exists.
GRIDIM	E		Calculate variables for preparation of a grid of points (equally-spaced population-definition methods) or polygons (random-selection with equal-area distribution and random-selection with iterating grids methods).
DEFIDX	F	X	Define an INFO category-index file (all population-definition methods).
CHKIDX	G		Check user-specified category in category-index file (all site-selection methods).
PICKEM	H	X	Randomly select sites (simple random-selection and random-selection with equal-area distribution methods).
EQARSA	I	X	Aggregate subareas into equal-area cells (random-selection with equal-area distribution method).
NFLATE	J		Calculate the initial guess for preparing a grid (random-selection with iterating grids method).
SAMINF	K	X	Determine if convergence criteria have been met and randomly select sites or compute another guess (random-selection with iterating grids method).
UPTIDX	L		Store number of sites selected in category-index file (all site-selection methods).
CATRPT	M		Retrieve and sort locations of selected sites (all site-selection methods).
FCATRP	N		Prepare site-selection report (all site-selection methods).

It will be necessary to modify the site-selection software to install it on a non-Prime computer system. The Fortran code contains four dependencies on the Primos computer-operating system, as follows.

- (1) A computerized function named RND is used to obtain a random number (attachments E, H, and K). The RND function is a Prime extension of standard Fortran 77.
- (2) To obtain a value to initialize RND, a Primos subroutine named TIMDAT (Spencer, 1986) is called to retrieve the number of seconds since midnight from the computer-system clock. The call to TIMDAT is made from a subroutine called IRSEED (attachment C).
- (3) The Primos subroutine named SUBSRT (Frost, 1986) is called from within ISRTFL (attachment C) to sort the contents of a file.
- (4) The other non-standard language feature in the software is the Prime \$INSERT compiler directive (Johnson, 1983, p. 3-15). The \$INSERT directive causes the compiler to include the contents of a named file at the location of the directive. This allows repetition of a statement or a group of statements in several programs in the software. The \$INSERT directive is used in the site-selection software to repeat PARAMETER statements that define variables for sizing of arrays. Other compiler vendors provide a similar compiler directive for Fortran 77. Often, the corresponding compiler directive is called INCLUDE. The user of a compiler that does not have this capability can insert the appropriate statements into the program at the locations of the applicable \$INSERT directive. The programming statements in attachment B are inserted into the programs and subroutines in attachments F, H, I, and K.

Several installation-dependent variables are defined at the beginning of program SAMPLE.AML. The four variables, TOLER, MAXITER, PERCENT, and DAMPER contain values for (1) the maximum number of iterations, (2) number of alternate sites, (3) the fraction of cells needed to contain the user-specified number of alternate sites for secondary convergence, and (4) the damping factor. All four of these variables can be modified by the user while the program is running. The values in the program are the values used when the prompting message is displayed for the variable, and the user enters a carriage return without entering a value. The last three variables pertain only to the random-selection with iterating grids method.

The next two variables, EXECUTE and INFOPATH, store the operating-system command for running an executable Fortran program, and the location of the INFO files for an ARC/INFO workspace.

The next two variables, BOLD\_ON and BOLD\_OFF, store the terminal-specific codes for enabling and disabling the display of bold characters on a terminal.

The next eleven variables: CHEK\_PGM, GRID\_PGM, DCAT\_PGM, CHEK\_PGM, PICK\_PGM, EQAR\_PGM, INFL\_PGM, MTHR\_PGM, UCAT\_PGM, RPT1\_PGM, and RPT2\_PGM, store the locations of each of the executable Fortran programs. The contents of these variables should be modified to contain the correct locations for the local computer system.

The final variable, PROJ\_DES, stores the location of a file that contains ARC/INFO commands. The file is used for converting the coordinates from the map projection used for the areal-subset coverage to latitude and longitude (Environmental Systems Research Institute, 1987b). The example projection-file listing (attachment N) is used to convert coordinates in zone 14 of the Universal Transverse Mercator map projection to latitude and longitude. When creating a customized projection file, only the lines affecting the input map projection should be changed. Thus, only the following lines should be changed.

PROJECTION UTM  
ZONE 14  
UNITS METERS

Additional lines describing the input map projection may be added, if necessary.

If more than one person is using the site-selection software on a computer system, it may be necessary for each person to copy and modify the SAMPLE.AML program. Each user may want to modify the values of the variables: TOLER, PERCENT, MAXITER, DAMPER, BOLD\_ON, BOLD\_OFF, and PROJ\_DES. The Fortran programs do not need to be copied, unless the values of the array sizes (attachment B) have to be changed.

When revisions to the ARC/INFO GIS are made, some of the subroutines used in the site-selection software may be changed. This may create a need to modify some subroutine call statements in the site-selection software. Table 4 describes the ARC/INFO subroutines that are used in each of the eleven Fortran programs.

### Data Used by the Software

The site-selection software uses the areal-subset coverage provided by the user and a population coverage, if supplied. The software makes copies of these coverages, so the original data are not modified. Other files and coverages are created while the software is running. Some of these are products of the software, as described previously in this report.

However, many files and coverages are created and deleted by the software during normal operation. Because most users do not need to be concerned with these "temporary" files and coverages, this report does not contain detailed descriptions. However, the users needs to be aware of the names because the software will delete any existing file or coverage with the same name.

Table 4.—Locations of call statements and descriptions of ARC/INFO subroutines

[An "X" beneath a program name indicates the subroutine is called within the program.]

Subrou- tine name	Program name											Description
	I T E X S T	G R I D I M	D E F I D X	C H K I D X	P I C K E M	E Q A R S A	N F L A T E	S A M I N F	U P T I D X	C A T I R P	F C A R P	
ACLOSE	X	X		X		X	X	X		X	X	Close logical unit: sequential access
ACREAT	X	X		X		X	X	X		X	X	Create and open logical unit: sequential access
ADELET						X						Delete file: sequential access
AEXIT	X	X	X	X	X	X	X	X	X	X	X	Stop program and return to AML program
AOPEN						X					X	Open logical unit: sequential access
ANAME										X		Construct file name: ARC
BOXGET		X										Get minimum and maximum coordinates for a coverage
ERROR				X	X	X	X	X	X	X	X	Print message, stop program, and stop AML program
INFADI				X								Add field to file: INFO
INFCLE				X				X	X		X	Clear expression: INFO
INFCLS	X			X	X	X	X	X	X	X	X	Close logical unit: INFO
INFDEC				X	X		X	X		X	X	Decode record: INFO
INFDEF				X								Create file and open logical unit: INFO
INFENC				X		X		X	X			Encode record: INFO
INFERS				X								Delete file: INFO
INFEXF					X							Determine existence of file: INFO
INFEXI	X			X	X	X	X	X	X	X	X	Describe field: INFO
INFGET				X	X	X	X	X	X	X	X	Read record: INFO
INFINT	X			X	X	X	X	X	X		X	Initialize subroutines: INFO
INFNAM					X					X		Construct file name: INFO
INFOPN	X			X	X	X	X	X	X	X	X	Open logical unit: INFO
INFORM	X			X		X		X				Display message with carriage return and line feed
INFPUT				X		X		X	X			Write record: INFO
INFSEL				X	X	X	X	X	X	X	X	Determine if record matches expression: INFO
INFTRN				X	X	X	X	X	X	X	X	Encode expression: INFO
LABOPN										X		Open file: label-point coordinates
LABRDR										X		Read direct-access: label-point coordinates
LABCLS										X		Close file: label-point coordinates
LUNINI	X	X	X	X	X	X	X	X	X	X	X	Initialize subroutines: logical units
MESINI	X	X	X	X	X	X	X	X	X	X	X	Initialize subroutines: messages
MINIT	X	X	X	X	X	X	X	X	X	X	X	Initialize subroutines: direct access
PRMSTR	X	X	X	X	X	X	X	X	X	X	X	Display message without carriage return and line feed
VINIT										X		Initialize subroutines: variable-length records

During definition of the population, regardless of the method selected, the SAMPLE.AML program creates a copy of the areal-subset coverage. The name of the copy is formed by joining together the name of the areal-subset coverage and the characters ".SUB".

The Fortran program, GRIDIM, creates a file that contains data used by the SAMPLE.AML program with the ARC GENERATE command (Environmental Systems Research Institute, 1987b). The GRIDIM program is used by the SAMPLE.AML program in three places for three different purposes. The first use of the GRIDIM program occurs during preparation of an equally spaced population. The second use of the GRIDIM program occurs during preparation of a subarea coverage. In both these cases, a temporary file is created and named by joining together the name of the areal-subset coverage and the characters ".SUB.GRD". The third use of the GRIDIM program, when using the random-selection with iterating grids method, creates a file containing the characteristics of the grid for current iteration. The name of this file is formed by joining together the name of the areal-subset coverage and the characters ".GRD".

During many operations performed by the software, a temporary coverage is needed. The name of the temporary coverage is formed by joining together the name of the areal-subset coverage and the characters ".GRID".

One other temporary coverage is needed when using the random-selection within cells approach. This temporary coverage is named by joining together the name of the areal-subset coverage and the characters ".SAM".

During preparation of a site-selection report, three temporary files are created. These files are named SAMPTS, SAMSRT, and SAMSRT.GEO. Two temporary files, named SUBA.ASCII and SUBA.SORT, are created during the aggregation of subareas into cells when the random-selection with equal-area distribution method is used. A temporary file named INFLATED is created when the random-selection with iterating grids method is used.

### Error Messages

This section provides an alphabetical listing of many error messages that may be displayed by the software, the name of the program(s) that displays the message, and possible corrective actions that may be taken by the user. Some error messages in this section have been described previously in this report, but are included here for easy reference.

The software is capable of producing other error messages that are not listed in this section. These messages are not included because it is unlikely that they will be displayed during normal use of the software. The capability to display these messages was incorporated into the software during initial coding and testing and was not removed from the published version of the code. If the software is revised or moved to another computer-operating system, these undocumented messages may be useful to the programmer.



---

Message: \_\_ sites have already been selected from the category: \_\_\_\_  
Do you want to reselect sites from this category (CR=NO)?  
Source: SAMPLE.AML  
Action: If you type an affirmative answer, the previous site selections  
will be removed, and site selection will be performed again.  
If you type a negative answer, the prompting message for the  
name of the category will be displayed again.

---

Message: A population must be defined before site selection  
Source: SAMPLE.AML  
Action: Select option one (Define population) or option three (Exit)  
when the main menu is displayed again.

---

Message: Areal-subset coverage does not exist  
Source: SAMPLE.AML  
Action: Enter the name of an existing areal-subset coverage when the  
prompting message is displayed again.

---

Message: Category value: \_\_\_\_ not found in category-index file  
Source: SAMPLE.AML  
Action: Enter a valid category value when prompting message is displayed  
again.

---

Message: Coverage not found  
Source: SAMPLE.AML  
Action: Enter the name of an existing point coverage when the prompting  
message is displayed again.

---

Message: Damping factor should not be less than or equal to zero  
Source: SAMPLE.AML  
Action: Enter a valid damping factor when the prompting message is  
displayed again.

---

Message: Fraction of cells must be a value between zero and one  
Source: SAMPLE.AML  
Action: Enter a number between zero and one when the prompting message  
is displayed again.

---

---

Message: If you continue, the coverage containing site selections is deleted  
Population coverage already exists, continue (CR=NO)?  
Source: SAMPLE.AML  
Action: If an affirmative answer is entered, a new population coverage is  
created. If a negative answer is entered, the main menu is  
displayed again.

---

Message: INFO file does not exist  
Source: ITEXST  
Action: Enter the name of an existing areal-subset coverage when the  
prompting message is displayed again.

---

Message: Insufficient population ( ) to select \_\_\_ sites.  
Source: NFLATE  
Action: For the specified category value, the number of potential sites  
is less than the number of sites to be selected. Prompting  
messages beginning with the category value are displayed again.  
Enter a different category value, number of sites, number of  
alternate sites, or QUIT.

---

Message: Integer value required  
Source: SAMPLE.AML  
Action: Enter an integer when the prompting message is displayed again.

---

Message: Invalid method code: \_\_\_  
Source: SAMPLE.AML  
Action: Enter a valid method code when the menu is displayed again.

---

Message: Invalid option code: \_\_\_  
Source: SAMPLE.AML  
Action: Enter a valid option code when menu is displayed again.

---

Message: No exclusion criteria entered...  
Source: SAMPLE.AML  
Action: Either disable geometric exclusion, or enter exclusion criteria  
after the population-definition options menu is displayed again.

---

---

Message: No sites in population  
Source: PICKEM  
Action: The program stops. If a fixed population is being used, there are no potential sites in the category for the population. If an equally-spaced population is being used, the areal subsets for the category are small and the distance between potential sites is relatively large. Either create a new population, or do not select sites from the same category when the software is run again.

---

Message: Number of alternate sites must be greater than or equal to zero  
Source: SAMPLE.AML  
Action: Enter a non-negative integer when the prompting message is displayed again.

---

Message: Numeric value required  
Source: SAMPLE.AML  
Action: Enter a number when the prompting message is displayed again.

---

Message: Numeric value required, redisplaying menu...  
Source: SAMPLE.AML  
Action: Enter desired method code and required numeric data when the menu is displayed again.

---

Message: Positive number required  
Source: SAMPLE.AML  
Action: Enter a positive number when the prompting message is displayed again.

---

Message: Positive value required, redisplaying menu...  
Source: SAMPLE.AML  
Action: Enter desired method code and required positive number when the menu is displayed again.

---

Message: Resizing needed for MAXCAT  
Source: DEFIDX  
Action: The program stops. Alter the value of MAXCAT in the inserted file or in the DEFIDX program, recompile, relink, and run the software again.

---

---

Message: Resizing needed for MAXCEL  
Source: SAMINF  
Action: The program stops. Alter the value of MAXCEL in the inserted file or in the SAMINF program, recompile, relink, and run the software again.

---

Message: Resizing needed for MAXPNT  
Source: SAMINF or PICKEM  
Action: The program stops. Alter the value of MAXPNT in the inserted file or in the SAMINF (including the RNDSEL subroutine) and PICKEM programs, recompile, relink, and run the software again.

---

Message: Resizing needed for MAXPOP  
Source: SAMINF  
Action: The program stops. Alter the value of MAXPOP in the inserted file or in the SAMINF program, recompile, relink, and run the software again.

---

Message: Resizing needed for MAXSUB  
Source: EQARSA  
Action: The program stops. Alter the value of MAXSUB in the inserted file or in the EQARSA program, recompile, relink, and run the software again.

---

Message: Selection quantity exceeds cell population  
Source: PICKEM  
Action: The program stops. In at least one cell, the total number of sites to be selected is greater than the number of potential sites. Define a new population, or reduce the number of primary or alternate sites. A previous informative message will have displayed the number of potential sites in the cell where the problem was detected.

---

Message: Unable to select from an empty subset  
Source: PICKEM  
Action: The program stops because there are no potential sites to be selected in the cell. If using the simple-random selection method, the user may either not select sites from the category or create a new population. If using the random-selection with equal-area distribution method, the user may either reduce the number of primary sites or create a new population.

---

---

Message: Variable name does not exist  
Source: ITEXST  
Action: Re-enter category-variable name when prompting message is  
displayed again.

---

#### REFERENCES

- American National Standards Institute, 1978, American national standard programming language FORTRAN: New York, American National Standards Institute.
- , 1979, Additional codes for use with the American national standard code for information interchange: New York, American National Standards Institute.
- Cochran, W.G., 1977, Sampling techniques (3rd ed.): New York, John Wiley, p. 89-92.
- Christenson, S.C., and Parkhurst, D.L., 1987, Ground-water quality assessment of the Central Oklahoma Aquifer, Oklahoma--Project Description: U.S. Geological Survey Open-File Report 87-235, 30 p.
- Environmental Systems Research Institute, Inc., 1987a, ARC/INFO the geographic information system software, v. 1: Redlands, Calif., Environmental Systems Research Institute, Inc.
- , 1987b, ARC/INFO the geographic information system software, v. 2: Redlands, Calif., Environmental Systems Research Institute, Inc.
- , 1987c, ARC Macro Language and user interface tools: Redlands, Calif., Environmental Systems Research Institute, Inc.
- , 1987d, ARC/PLT Map display and query: Redlands, Calif., Environmental Systems Research Institute, Inc.
- , 1987e, Users guide ARC/INFO plot system, Prime version 4.0: Redlands, Calif., Environmental Systems Research Institute, Inc.
- , 1989, ARC/INFO programmer manual, Version 5.0, v. 1-3: Redlands, Calif., Environmental Systems Research Institute, Inc.
- Frost, Dick, 1986, Subroutines reference guide, v. 4: Natick, Mass., Prime Computer, Inc., p. 17-40 through 17-41.
- Gilbert, R.O., 1987, Statistical methods for environmental pollution monitoring: New York, Van Nostrand Reinhold Co., p. 46-47.

- Henco Software, Inc., 1983, INFO reference manual, Revision 9.0: Waltham, Mass., Henco Software, Inc.
- Hirsch, R.M., Alley, W.M., and Wilber, W.G., 1988, Concepts for a National Water-Quality Assessment Program: U.S. Geological Survey Circular 1021, 42 p.
- Johnson, E.C., 1983, FORTRAN 77 reference guide (3rd ed.): Framingham, Mass., Prime Computer, Inc.
- Landy, Alice, 1982, CPL user's guide: Framingham, Mass., Prime Computer, Inc.
- Rennick, W.L., 1986, Implementation of geographic information systems in the Water Resources Division of the United States Geological Survey, in Hydrologic applications of space technology: International Association of Hydrological Sciences Publication no. 160, p. 469-473.
- Spencer, Debra., 1986, Subroutines reference guide, v. 3: Natick, Mass., Prime Computer, Inc., p. 2-34 through 2-35.
- Steel, R.G., and Torrie, J.H., 1960, Principles and procedures of statistics: New York, McGraw-Hill, p. 420.
- Snyder, J.P., 1987, Map projections--A working manual: U.S. Geological Survey Professional Paper 1395, 383 p.; *supercedes* Snyder, J.P., 1982, Map projections used by the U.S. Geological Survey: Geological Survey Bulletin 1532, 313 p.

## GLOSSARY

### Terms

Alternate sites.--Alternate sites are water-quality sampling locations selected to provide in advance for potential difficulties that may be encountered when visiting sites. For example, it may be impossible to drill at a selected site. When such a difficulty occurs, the hydrologist will have one or more alternative sites to visit if more than one alternate site is specified during site selection. Zero, one, or more alternate sites may be selected for each primary site.

Areal subsets.--An areal subset is a part of the study region that is contiguous and consistent with respect to some spatial characteristic (fig. 1). The characteristic can be based on land use, topography, geology, or some other spatial variable. A study region may be composed of many areal subsets, and commonly several distinct areal subsets may represent those parts of a region having a particular value (or range of values) of the characteristic.

Category.--A category is a value of a spatial characteristic that identifies a group of similar areal subsets (fig. 1). Within a study region, there may be more than one discrete areal subset in the same category. The number of different categories determines the amount of stratification of the study region. Categories often are referred to as "strata" in statistics texts.

Cell.--A cell is a polygon that subdivides the study region (fig. 5). A cell is used in the random selection within cells approach to define a group of sites that may be selected from a part of the study region. Cells sometimes are referred to as "blocks" in statistics texts.

Convergence.--Convergence is a condition that occurs when the solution to a problem is found at the end of a series of iterations.

Convergence criteria.--Convergence criteria are a set of user-specified conditions that are tested at the end of each iteration. The convergence criteria are used to determine if a solution to a problem has been found.

Coverage.--A coverage is a computerized representation of the locations and characteristics of spatial data. Spatial data stored in coverages include points, lines, and polygons.

Damping factor.--The damping factor is a coefficient for controlling the change in cell size between successive iterations when selecting sites using the random-selection with iterating grids method.

Fixed population of potential sites.--A fixed population is a group of existing locations that may be selected. Commonly, the fixed population consists of existing water wells that might be used for water-quality sampling.

Grid.--A grid is a collection of cells used for the areal distribution of sites during the selection process. The grid of cells completely covers the study region.

Iteration.--An iteration is a single attempt to find a solution using a sequence of computations. Normally, several iterations are needed to find a solution.

Pass.--A pass is defined as the site-selection process for a single category (fig. 3). The number of passes necessary to select sites for the study region is equal to the number of categories that will have at least one site selected. The total number of sites selected during a pass is the number of site-selection rounds multiplied by the number of primary sites selected. (See "primary site" and "round".)

Polygon.--A polygon is a two-dimensional shape that defines an area.

Population.--The population is the group of all sites in the study region that have the potential to be selected (fig. 2).

Potential site.--A potential site is a location in the study region that may be selected for the collection of a water-quality sample.

Primary site.--A primary site is the first choice made by the site-selection software for the collection of a water-quality sample. If the primary site is not suitable for sampling, and alternate sites have been selected, the alternate site selected during round-number two should be visited next.

Random.--In general, random means without statistical bias. With respect to this computer program, random selection denotes making a choice among arbitrarily numbered alternatives using a computerized method for generating a random number, scaled to the range of alternatives.

Round.--A round is part of the site-selection process for a category (fig. 3). Site selection is performed by rounds. The number of rounds in a selection pass is one plus the user-specified number of alternate sites. Sites are identified by the round number during which the sites were selected. Primary sites are selected during round-number one. Alternate sites are selected during rounds two, three, and so on.

Stratified.--Stratified means organized into internally consistent groups. A stratified study region is subdivided into areal subsets, each of which is homogenous internally with respect to some characteristic (fig. 1).

Study region.--The study region is the two-dimensional spatial extent of the part of the earth's surface that is being investigated during a project.



**Subarea.**--A subarea is a polygon that contains a small part of the study region. Each areal subset is subdivided into one or more subareas when site selection is performed using the random-selection with equal-area distribution method. Subareas are aggregated into the cells that are used for the selection of sites. Each subarea contains area for a single category (fig. 4).

### Symbols

Symbol	Definition
$A$	Total area of the study region
$A_c$	Total area for a category
$C$	Number of cells containing parts of an area
$\delta$	Number of water-quality samples collected within a cell
$f$	User-specified damping factor
$f_a$	User-specified fraction of cells that must contain $\eta_r$ sites
$f_i$	Damping factor for an iteration
$G_i$	Guess for the number of cells needed for the current iteration
$\eta_a$	User-specified number of alternate sites to be selected for a category
$\eta_p$	User-specified number of primary sites to be selected for a category
$\eta_r$	Number of site-selection rounds ( $1 + \eta_a$ )
$\eta_t$	Total number of sites to be selected for a category ( $\eta_p + \eta_r$ )
$N$	Number of potential sites in a category
$N_a$	Number of potential sites in cells that contain less than $\eta_r$ sites
$N_{ct}$	Number of cells containing $\eta_r$ potential sites
$N_i$	Number of potential sites inside an area
$N_c$	Number of potential sites inside a cell
$N_o$	Number of potential sites outside an area and within a cell
$P_a$	Probability of selecting a site from within a specific area
$P_s$	Probability of selecting a specific site
$\Pi$	A mathematical operator indicating all terms in a sequence are multiplied together

## APPENDIX. SELECTION PROBABILITIES

This discussion of selection probabilities presents equations for calculation of the probability that a site or an area will be selected by the software described in this report. These equations are included to (1) explain mathematically how the selection methods differ, (2) provide a method for determining the randomness of site selections using the random selection within cells approach, and (3) establish a framework for additional studies of ground-water sampling-network design.

In the discussion that follows, the mathematical expressions are based on the assumption that only one water-quality sample is collected from each set of primary and alternate sites.

Sites in the same cell have equal probabilities of being selected. During selection of the first site, the probability of selecting a specific site ( $P_s$ ) is equal to the reciprocal of the total number of sites available for selection ( $N$ ). After a site is selected, it is not available for selection again (referred to in statistics texts as "sampling without replacement"). Therefore, the number of sites available for selection decreases by one each time a site is selected. With multiple selections from the same cell, the probability that a site will be chosen is one minus the product of the probabilities that the site will not be chosen during each selection.

$$P_s = 1 - \prod_{i=1}^{\delta} \frac{N - i}{N - i + 1}$$

$$= \delta / N \quad (5)$$

where  $P_s$  = the probability of selecting a specific site,

$N$  = the number of sites available for selection,

$\delta = \begin{cases} \text{the number of water-quality samples collected within a cell,} \\ \eta_p \text{ for the simple random-selection approach, and} \\ 1^p \text{ for the random selection within cells.} \end{cases}$

The simple random-selection approach always provides a statistically random selection of sites from the population because only one cell is used. Thus, every site in a category has an equal probability of being selected. The random-selection within cells approach does not provide a strictly random selection because different numbers of potential sites may be present in each of the cells from which sites are selected. The departure from randomness may be determined using equation 5 by calculating the selection probability for the simple random-selection approach, and comparing it to the selection probability calculated for each cell created during the random-selection within cells approach.

It also may be useful to calculate the probability that a site will be selected from a specific area of the study region ( $P_a$ ). The area may be the limits of some known contamination problem, or the area may be one of the areal subsets used for categorizing the study region. The following equations can be used to perform this calculation.

When the area is completely contained within one cell,  $P_a$  is determined by the number of sites within the area and the number sites within the cell. When selecting one site, the probability of selecting a site from an area ( $P_a$ ) in the cell is equal to the number of sites located within the area ( $N_i$ ) divided by the total number of sites available for selection in the cell. When more than one site is selected, the probability of selecting at least one site within an area is equal to one minus the product of the probabilities of not choosing a site from the area during each selection.

$$P_a = 1 - \prod_{i=1}^{\delta} \frac{N_c - N_i - i + 1}{N_c - i + 1}$$

$$= 1 - \frac{(N_c - N_i)! (N_c - \delta)!}{(N_c - N_i - \delta)! N_c!}$$

where  $P_a$  = the probability of selecting a site within a specific area, and

$N_i$  = number of sites located within the area.

Let  $N_o = N_c - N_i$ , the number of sites in the cell that are outside of the area.

$$P_a = 1 - \frac{N_o! (N_c - \delta)!}{(N_o - \delta)! N_c!} \quad (6)$$

When an area is contained in more than one cell, the probability that at least one site within the area will be chosen is one minus the product of the probabilities that a site within the area will not be chosen from each cell.

$$P_a = 1 - \prod_{i=1}^C \frac{N_{o,i}! (N_{c,i} - \delta_i)!}{(N_{o,i} - \delta_i)! N_{c,i}!}$$

where  $C$  = the number of cells that contain parts of the area.

The simple random-selection method uses one cell for the entire study area; therefore, the above equation does not pertain to the method. The random selection within cells approach selects  $\eta_r$  sites from each cell, but because only one water-quality sample is collected per cell ( $\delta = 1$ ), the above equation simplifies to the following one.

$$P_a = 1 - \prod_{i=1}^C \frac{N_{o,i}}{N_{c,i}} \quad (7)$$

Some area-based approximations to these equations are possible, depending on the methods chosen for defining the population of potential sites, and for selecting sites. These approximations are discussed below. If the requirements for the approximations are not met, the above equations should be used to calculate selection probabilities.

Using the simple random-selection approach, each site in the same category has an equal probability of being selected. If the population of potential sites is spaced equally, the probability of selecting at least one site from an area is equal approximately to the ratio of the size of the area to the total area of areal subsets in the same category.

Using the random-selection with equal-area distribution method, each cell contains approximately the same category area. If the population of potential sites is spaced equally, the following two statements are true: (1) Sites in different cells have approximately equal probabilities of being selected, and (2) the probability of selecting at least one site from an area is equal approximately to the ratio of the size of the area to the total area for the category.

Using the random-selection with iterating grids method, sites within the same cell have equal probabilities of being selected. However, no further conclusions regarding selection probabilities can be made until a grid is found that satisfies the convergence criteria. After a grid has been found, equations 5, 6, and 7 should be used to calculate selection probabilities for sites and areas.

# ATTACHMENTS

## Attachment A.--Arc macro language program SAMPLE.AML listing

```

/* SAMPLE.AML - RANDOMLY SELECT SAMPLING SITES FROM A STRATIFIED STUDY REGION
/*      USING ONE OF THREE METHODS
/*
/* Use of U.S. Geological Survey program SAMPLE.AML is described in U.S.
/* Geological Water-Resources Investigations Report 90-4101, by Jonathon C.
/* Scott. This program is written in the Arc Macro Language (AML). The
/* program uses the following proprietary software: the INFO file-management
/* system and the ARC/INFO geographic information system. The program runs
/* various programs written in Fortran 77. This program was last modified
/* and run on a Prime 9955-II minicomputer running revision 22 of the PRIMOS
/* operating system on July 26, 1990.
/*
/* Although this computer software has been used by the U.S. Geological Survey,
/* no warranty, expressed or implied, is made by the USGS as to the accuracy
/* and functioning of the program and related program material nor shall the
/* fact of distribution constitute any such warranty, and no responsibility
/* is assumed by the USGS in connection therewith.
/*
/*
/* THE FOLLOWING FILE-SYSTEM NAMES ARE USED:
/*
/*      NAME          TYPE      DISPOSITION  METHOD      CONTENTS
/*
/*      NAME          TYPE      DISPOSITION  METHOD      CONTENTS
/*      1 2 3
/* cover             coverage   input/kept   X X X      stratification polygons
/* cover.SUB          coverage   work/deleted X X X      incover after geom. exclusion
/* cover.GRD          file       work/deleted X X X      gridding data
/* cover.GRID         coverage   work/deleted X X X      generated sampling points
/* SUBA.ASCII         file       work/deleted X          subarea centroid coordinates
/* SUBA.SORT          file       work/deleted X          sorted SUBA.ASCII
/* INFLATED           file       work/deleted X          "guess" for next iteration
/* SAMPTS             file       work/deleted X X X      selected coordinates
/* SAMSRT             file       work/deleted X X X      sorted selected coordinates
/* SAMSRT.GEO         file       work/deleted X X X      projected selected coordinates
/* TSTINF             file       work/deleted X X X      logical record (.TRUE./.FALSE.)
/* cover.POP          coverage   output/kept X X X      sampling point population
/* cover.SUBA         coverage   output/kept X          area-resolution polygons
/* cover.CELL         coverage   output/kept X X        sampling cell polygons
/* cover.SAM          coverage   work/deleted X X        selected points
/* CATIDX             INFO-file  output/kept X X X      category index/sampling data
/*
/* STORE CURRENT DIRECTORY AND SET DEFAULT VALUES FOR VARIABLES
&S HOME      := [DIR [PATHNAME *]]
&S TOLER      := 2
&S MAXITER    := 8
&S PERCENT    := .75
&S DAMPER     := .90
/*
/* SET OPERATING-SYSTEM-DEPENDENT VARIABLES
&S EXECUTE    := RESUME          /* COMMAND TO RUN A PROGRAM
&S INFOPATH   := %HOME%>INFO    /* PATHNAME OF INFO DATABASE
/*
/* SET TERMINAL-DEPENDENT VARIABLES (ANSI)
&S BOLD_ON    := '[1m'          /* TURN ON BOLD CHARACTERS
&S BOLD_OFF   := '[0m'          /* TURN OFF BOLD CHARACTERS
/*
/* SET INSTALLATION-DEPENDENT VARIABLES
&S ITCK_PGM   := <GIS>NAWSOFT>SAMP>R5>ITEXST.RUN /* INFO-FILE CHECK
&S GRID_PGM   := <GIS>NAWSOFT>SAMP>R5>GRIDIM.RUN /* CELL SIZE ALGORITHM
&S DCAT_PGM   := <GIS>NAWSOFT>SAMP>R5>DEFIDX.RUN /* INDEX FILE CREATION
&S CHEK_PGM   := <GIS>NAWSOFT>SAMP>R5>CHKIDX.RUN /* CHECK INDEX FILE
&S PICK_PGM   := <GIS>NAWSOFT>SAMP>R5>PICKEM.RUN /* RANDOM SELECTION
&S EQAR_PGM   := <GIS>NAWSOFT>SAMP>R5>EQARSA.RUN /* RANDOM SELECTION (METHOD II)
&S INFL_PGM   := <GIS>NAWSOFT>SAMP>R5>NFLATE.RUN /* INITIAL GUESS (METHOD III)
&S MTHR_PGM   := <GIS>NAWSOFT>SAMP>R5>SAMINF.RUN /* CONVERGENCE CHECK (METHOD III)
&S UCAT_PGM   := <GIS>NAWSOFT>SAMP>R5>UPTIDX.RUN /* INDEX FILE UPDATING
&S RPT1_PGM   := <GIS>NAWSOFT>SAMP>R5>CATRPT.RUN /* SAMPLING LOCATION EXTRACTION
&S RPT2_PGM   := <GIS>NAWSOFT>SAMP>R5>FCATRP.RUN /* REPORT GENERATION

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&S PROJ_DES := <GIS>NAWSOFT>SAMP>R5>UTM.2.LL /* PROJECTION FILE (MAP->LAT-LONG UNITS)
/*
/* DETERMINE SITE-SELECTION METHOD
/*
&LABEL SELECT_METHOD
&TYPE
&TYPE Site selection may be performed using one of three methods:
&TYPE
&TYPE ' Code Method'
&TYPE
&TYPE ' 1 Simple random selection'
&TYPE ' 2 Random selection with equal-area distribution'
&TYPE ' 3 Random selection with iterating grids'
&TYPE
&S METHOD := [RESPONSE 'Enter method code']
&SELECT %METHOD%
&WHEN 1,2,3
&DO
&GOTO STRATIFY
&END
&OTHERWISE
&DO
&TYPE 'Invalid method code: '%METHOD%'
&TYPE ' '
&GOTO SELECT_METHOD
&END
&END
/*
/* DESCRIBE STRATIFYING POLYGONS
/*
&LABEL STRATIFY
&S INCOVER := [RESPONSE 'Enter name of areal-subset coverage']
&S CATITEM := [RESPONSE 'Enter name of category variable']
/*
/* CHECK THAT COVERAGE, INFO FILE, AND VARIABLE NAME EXIST
/*
&TYPE
&TYPE %BOLD ON%checking areal-subset coverage specifications%BOLD_OFF%
&IF ^ [EXISTS %INCOVER% -COVER] &THEN &DO
&TYPE Areal-subset coverage does not exist
&GOTO STRATIFY
&END
&DATA %EXECUTE% %ITCK_PGM%
%INFOPATH%
ARC
%INCOVER%.PAT
%CATITEM%
&END
/*
/* READ RESULTS OF TEST FROM THE OUTPUT FILE WRITTEN BY THE FORTRAN PGM.
/*
&S AMLU := [OPEN TSTINF OPENERR -READ]
&S OK := [READ %AMLU% EOF]
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE TSTINF]
&IF ^ %OK% &THEN &GOTO STRATIFY
/*
/* MAIN MENU FOR CHOOSING PROCESSING OPTIONS...
/*
&LABEL SELECT_OPTION
&S DEFPPOP := [EXISTS %INCOVER%.POP -COVER]
&TYPE
&TYPE 'Site-selection software main menu'
&TYPE
&TYPE ' Code Option'
&TYPE
&TYPE ' 1 Define population'
&IF %DEFPPOP% &THEN ~
&TYPE ' 2 Select sites from population'

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&TYPE ' 3 Exit from site-selection software'
&TYPE
&S OPTION := [RESPONSE 'Enter option code' 3]
/*
&SELECT %OPTION%
&WHEN 1
&DO
&S GO := .TRUE.
&S PROMPT := 'Population coverage already exists, continue (CR=NO)'
&IF %DEFPOP% &THEN &DO
&TYPE 'If you continue, the coverage containing site selections is deleted!'
&S GO := [QUERY %PROMPT% .FALSE.]
&END
&IF %GO% &THEN &GOTO DEFINE_POP
&GOTO SELECT_OPTION
&END
&WHEN 2
&DO
&IF [EXISTS %INCOVER%.POP -COVER] &THEN &GOTO SAMPLE
&TYPE
&TYPE A population must be defined before site selection
&GOTO SELECT_OPTION
&END
&WHEN 3
&RETURN
&OTHERWISE
&DO
&TYPE 'Invalid option code: '%OPTION%'
&TYPE
&GOTO SELECT_OPTION
&END
&END
/*
/* -----> DEFINE POPULATION <-----
/* CREATE A NEW COVERAGE TO BE USED AS THE POPULATION. THE NEW COVERAGE
/* IS A COLLECTION OF POINTS OVERLAID ONTO THE AREAL SUBSETS.
/*
&LABEL DEFINE_POP
/*
/* OPTION FOR GEOMETRIC EXCLUSION OF AREAL SUBSETS
/*
&TYPE
&S EXC := [QUERY 'Do you wish to exclude areal subsets using geometry (CR=NO)' ~
.FALSE.]
/*
&IF [EXISTS %INCOVER%.SUB -COVER] &THEN KILL %INCOVER%.SUB
COPY %INCOVER% %INCOVER%.SUB
&IF %EXC% &THEN &DO
&LABEL GET_MINAREA
&S MINAREA := [RESPONSE 'Enter minimum area (CR=no minimum)' 0]
&IF [TYPE %MINAREA%] > 0 &THEN &DO
&TYPE Numeric value required
&GOTO GET_MINAREA
&END
&LABEL GET_MINRATIO
&S MINRATIO := [RESPONSE 'Enter minimum area/perimeter ratio (CR=no minimum)' 0]
&IF [TYPE %MINRATIO%] > 0 &THEN &DO
&TYPE Numeric value required
&GOTO GET_MINRATIO
&END
&IF %MINAREA% = 0 & %MINRATIO% = 0 &THEN &DO
&TYPE No exclusion criteria entered...
&GOTO DEFINE_POP
&END
/* END IF ERROR BRANCH
&IF %MINRATIO% > 0 &THEN ~
ADDITEM %INCOVER%.SUB.PAT %INCOVER%.SUB.PAT APRATIO 4 12 F 3
/*
/* REMOVE POLYGON CATEGORY VALUE FROM GEOMETRICALLY EXCLUDED AREAL SUBSETS
/*

```

Attachment A.--Arc macro language program SAMPLE.AML. listing--Continued

```

&WORKSPACE %INFOPATH%
&DATA INFO
ARC
SELECT %INCOVER%.SUB.PAT
&S EXPRESS := RESELECT
&IF %MINAREA% > 0 &THEN &S EXPRESS := %EXPRESS% AREA LE %MINAREA%
&IF %MINRATIO% > 0 &THEN &DO
    CALC APRATIO = AREA / PERIMETER
    &IF %MINAREA% > 0 &THEN &S EXPRESS := %EXPRESS% OR
    &S EXPRESS := %EXPRESS% APRATIO LE %MINRATIO%
&END
&EXPRESS%
MOVE ' ' TO %CATITEM%
QUIT STOP
&END
&WORKSPACE %HOME%
&END
/*
/* OPTIONS FOR DEFINING POTENTIAL SITES
/*
&LABEL DEFINE_SIZE
&TYPE
&TYPE You may define a population using any of three methods:
&TYPE
&TYPE ' Code    Method'
&TYPE
&TYPE '    1    Specify the approximate number of potential sites'
&TYPE '    2    Specify the distance between each potential site'
&TYPE '    3    Specify an existing point coverage'
&TYPE
&S OPT := [RESPONSE 'Enter a method code']
/*
&SELECT %OPT%
&WHEN 1
    &S NPTS := [RESPONSE 'Enter approximate number of sites (CR=10,000)']
    10000
&WHEN 2
    &S NPTS := [RESPONSE 'Enter distance between each site']
&WHEN 3
    &DO
        &LABEL GET_PCOVER
        &S PCOVER := [RESPONSE 'Enter name of point coverage']
        &IF ^ [EXISTS %PCOVER% -COVER] &THEN &DO
            &TYPE Coverage not found
            &GOTO GET_PCOVER
        &END
        &GOTO OVERLAY
    &END
&OTHERWISE
    &DO
        &TYPE 'Invalid method code:  '%OPT%
        &GOTO DEFINE_SIZE
    &END
&END
&IF [TYPE %NPTS%] > 0 &THEN &DO
    &TYPE Numeric value required, redisplaying menu...
    &GOTO DEFINE_SIZE
&END
&IF %NPTS% <= 0 &THEN &DO
    &TYPE Positive value required, redisplaying menu...
    &GOTO DEFINE_SIZE
&END
/*
/* WHEN USING METHOD ONE OR TWO:
/* CALCULATE CELL SIZE & PASS BACK THE COVERAGE LIMITS.
/* RESULTS ARE WRITTEN TO THE DISK FILE:  incover.SUB.GRD
/*
&TYPE %BOLD ON%Defining population characteristics...%BOLD OFF%
&IF [EXISTS %INCOVER%.SUB.GRD] &THEN &S DELETER := [DELETE %INCOVER%.SUB.GRD]

```



Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&DATA %EXECUTE% %GRID_PGM%
%INCOVER%.SUB
%OPT%
%NPTS%
&END
/*
/* READ THE RESULTS FROM THE FORTRAN PGM IN THE OUTPUT FILE.
/*
&S AMLU := [OPEN %INCOVER%.SUB.GRD OPENERR -READ]
&S LINE := [TRIM [READ %AMLU% EOF]]
&DO VAR &LIST XMIN XMAX YMIN YMAX XDIM YDIM NCELL
    &S TEMP := [BEFORE %LINE% ' ']
    &S %VAR% := %TEMP%
    &S LINE := [TRIM [AFTER %LINE% %TEMP%]]
&END
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE %INCOVER%.SUB.GRD]
/*
/* GENERATE A COVERAGE WITH LABEL POINTS.
/*
&TYPE %BOLD ON%Building a population with %NCELL% sites...%BOLD OFF%
&IF [EXISTS %INCOVER%.GRID -COVER] &THEN KILL %INCOVER%.GRID
GENERATE %INCOVER%.GRID
COPYTICS %INCOVER%
GRID LABELS
%XMIN%, %YMIN%
%XMIN%, %YMAX%
%XDIM%, %YDIM%
0, 0
%XMAX%, %YMAX%
QUIT
/*
/* CREATE POINT TOPOLOGY FOR THE COVERAGE.
/*
BUILD %INCOVER%.GRID POINT
&S PCOVER := %INCOVER%.GRID
/*
/* ALL THREE POPULATION-DEFINITION METHODS BRANCH TO HERE.
/* OVERLAY THE POINTS ONTO THE AREAL SUBSETS.
/*
&LABEL OVERLAY
&IF [EXISTS %INCOVER%.POP -COVER] &THEN KILL %INCOVER%.POP
CLIP %PCOVER% %INCOVER% %INCOVER%.POP POINT 0.001
KILL %PCOVER%
RENAME %INCOVER%.POP %PCOVER%
IDENTITY %PCOVER% %INCOVER%.SUB %INCOVER%.POP POINT 0.001
/*
/* CLEAN-UP THE WORK DATA SETS, AND ADD ITEMS FOR SAVING SELECTED POINTS,
/* UNIQUELY IDENTIFYING POINTS, AND CELLS
/*
&IF [EXISTS %INCOVER%.GRID -COVER] &THEN KILL %INCOVER%.GRID
ADDITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT CHOSEN 4 5 B
ADDITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT UNIQUE 4 5 B
ADDITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT CELLNO 4 5 B
DROPITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT %INCOVER%.SUB#
DROPITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT %INCOVER%.SUB-ID
DROPITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT %PCOVER%#
DROPITEM %INCOVER%.POP.PAT %INCOVER%.POP.PAT %PCOVER%-ID
&IF %METHOD% NE 2 &THEN KILL %INCOVER%.SUB
/*
/* STORE A UNIQUE IDENTIFIER FOR EACH POINT IN THE POPULATION
/*
&WORKSPACE %INFOPATH%
&DATA INFO
ARC
SELECT %INCOVER%.POP.PAT
CALC UNIQUE = $RECNO
Q STOP
&END

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&WORKSPACE %HOME%
/*
/* DEFINE AN INFO-FILE NAMED 'CATIDX' TO KEEP AN INDEX OF CATEGORIES
/*
&TYPE
&TYPE %BOLD ON%Preparing category-index file, CATIDX, in INFO%BOLD OFF%
&DATA %EXECUTE% %DCAT_PGM%
%INFOPATH%
%INCOVER%
%INCOVER%.POP
%CATITEM%
&END
/*
/* DEFINITION OF A POPULATION FOR SITE-SELECTION METHODS ONE AND THREE IS DONE.
/* SITE-SELECTION METHOD TWO NEEDS A COVERAGE OF SUBAREAS FOR RESOLVING
/* EQUAL-AREA CELLS.
/*
&IF %METHOD% ^= 2 &THEN &GOTO SELECT_OPTION
/*
/* DETERMINE HOW SUBAREAS WILL BE DEFINED...
/*
&LABEL DEFINE_SUBA
&TYPE %BOLD ON%Defining subareas...%BOLD OFF%
&TYPE
&TYPE You may define subareas using either of two methods:
&TYPE
&TYPE ' Code   Method'
&TYPE
&TYPE '   1   Specify the approximate number of subareas, or'
&TYPE '   2   Specify the width of subareas'
&TYPE
&S OPT := [RESPONSE 'Enter a method code']
/*
&SELECT %OPT%
&WHEN 1
&S NPTS := [RESPONSE 'Enter approximate number of subareas (CR=10,000)']~
10000]
&WHEN 2
&S NPTS := [RESPONSE 'Enter width of each subarea']
&OTHERWISE
&DO
&TYPE 'Invalid method code:  '%OPT%
&GOTO DEFINE_SUBA
&END
&END
&IF [TYPE %NPTS%] > 0 &THEN &DO
&TYPE Numeric value required, redisplaying menu...
&GOTO DEFINE_SUBA
&END
&IF %NPTS% <= 0 &THEN &DO
&TYPE Positive value required, redisplaying menu...
&GOTO DEFINE_SUBA
&END
&END
/*
/* CALCULATE SUBAREA SIZE & PASS BACK THE COVERAGE LIMITS.
/* RESULTS ARE WRITTEN TO THE DISK FILE:  incover.SUB.GRD
/*
&TYPE %BOLD ON%Defining subarea characteristics...%BOLD OFF%
&IF [EXISTS %INCOVER%.SUB.GRD] &THEN &S DELETER := [DELETE %INCOVER%.SUB.GRD]
&DATA %EXECUTE% %GRID_PGM%
%INCOVER%.SUB
%OPT%
%NPTS%
&END
/*
/* READ THE RESULTS FROM THE FORTRAN PGM IN THE OUTPUT FILE.
/*
&S AMLU := [OPEN %INCOVER%.SUB.GRD OPENERR -READ]
&S LINE := [TRIM [READ %AMLU% EOF]]

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&DO VAR &LIST XMIN XMAX YMIN YMAX XDIM YDIM NCELL
&S TEMP := [BEFORE %LINE% ' ']
&S %VAR% := %TEMP%
&S LINE := [TRIM [AFTER %LINE% %TEMP%]]
&END
&S CLOSERR := [CLOSE %AML%]
&S DELETER := [DELETE %INCOVER%.SUB.GRD]
/*
/* GENERATE A COVERAGE OF SUBAREAS WITH LABEL POINTS.
/*
&TYPE %BOLD_ON%Preparing %NCELL% subareas...%BOLD_OFF%
/*
&IF [EXISTS %INCOVER%.SUBA -COVER] &THEN KILL %INCOVER%.SUBA
GENERATE %INCOVER%.SUBA
COPYTICS %INCOVER%
GRID LABELS
%XMIN%, %YMIN%
%XMIN%, %YMAX%
%XDIM%, %YDIM%
0, 0
%XMAX%, %YMAX%
QUIT
/*
/* CREATE POLYGON TOPOLOGY FOR THE SUBAREAS, REMOVE SUBAREAS OUTSIDE STUDY
/* REGION, OVERLAY SUBAREAS ON AREAL SUBSETS, ADD VARIABLES FOR CELL AND
/* STRIP NUMBER, REMOVE UNNEEDED VARIABLES. DELETE UNNEEDED COVERAGES.
/*
BUILD %INCOVER%.SUBA POLY
&S SACOVER := %INCOVER%.SUBA
&IF [EXISTS %INCOVER%.GRID -COVER] &THEN KILL %INCOVER%.GRID
CLIP %INCOVER%.SUBA %INCOVER% %INCOVER%.GRID POLY 0.001
KILL %INCOVER%.SUBA
RENAME %INCOVER%.GRID %INCOVER%.SUBA
UNION %INCOVER%.SUB %INCOVER%.SUBA %INCOVER%.GRID 0.001
ADDITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT CELL 4 5 B
ADDITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT VSTRIP 4 5 B
DROPIITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT %INCOVER%.SUB#
DROPIITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT %INCOVER%.SUB-ID
DROPIITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT %INCOVER%.SUBA#
DROPIITEM %INCOVER%.GRID.PAT %INCOVER%.GRID.PAT %INCOVER%.SUBA-ID
KILL %INCOVER%.SUB
KILL %INCOVER%.SUBA
RENAME %INCOVER%.GRID %INCOVER%.SUBA
&GOTO SELECT_OPTION
/*
/* -----> SAMPLE POPULATION <-----
/*
/* POPULATION HAS BEEN DEFINED, READY TO RANDOMLY SELECT FROM ONE CATEGORY.
/*
&LABEL SAMPLE
&TYPE
&TYPE %BOLD_ON%Ready for site selection...%BOLD_OFF%
&TYPE
/*
/* GET SELECTION CONSTRAINTS COMMON TO ALL METHODS
/*
&S PROMPT := 'Enter value of '%CATITEM%' for site selection (CR = QUIT)'
&S CATVAL := [RESPONSE %PROMPT% QUIT]
&IF %CATVAL% = QUIT &THEN &RETURN
/*
&LABEL GET_NOSAMP
&S PROMPT := 'Enter number of primary sites to be selected when '%CATITEM%' = '%CATVAL%'
&S NOSAMP := [RESPONSE %PROMPT% 0]
&IF [TYPE %NOSAMP%] ^= -1 &THEN &DO
&TYPE Integer value required
&GOTO GET_NOSAMP
&END
&IF %NOSAMP% <= 0 &THEN &RETURN
/*

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&LABEL ALTERNATE
&S PROMPT := 'Enter number of alternate sites (CR = '%TOLER%')'
&S TOLER := [RESPONSE %PROMPT% %TOLER%]
&IF [TYPE %TOLER%] ^= -1 &THEN &DO
    &TYPE Integer value required
    &GOTO ALTERNATE
&END
&IF %TOLER% < 0 &THEN &DO
    &TYPE
    &TYPE Number of alternate sites must be greater than or equal to zero
    &GOTO ALTERNATE
&END
/*
/* CHECK CATEGORY-INDEX FILE, DETERMINE IF CATEGORY EXISTS OR ALREADY SELECTED
&DATA %EXECUTE% %CHEK_PGM%
%INFOPATH%
%CATVAL%
&END
&S AMLU := [OPEN T%NOSAMP OPENERR -READ]
&S NDOONE := [TRIM [READ %AMLU% EOF]]
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE T%NOSAMP]
&IF %NDOONE% = -1 &THEN &DO
    &TYPE Category value: %CATVAL% not found in category-index file
    &GOTO SAMPLE
&END
&IF %NDOONE% > 0 &THEN &DO
    &TYPE %NDOONE% sites have already been selected from the category: %CATVAL%
    &S PROMPT := 'Do you want to reselect sites for this category (CR=NO)'
    &S GO := [QUERY %PROMPT% .FALSE.]
    &IF ^ %GO% &THEN &GOTO SAMPLE
&END
/*
/* BRANCH TO APPROPRIATE CODE PERTAINING TO SITE-SELECTION METHOD
&SELECT %METHOD%
&WHEN 1
    &DO
    &GOTO METHOD_ONE
    &END
&WHEN 2
    &DO
    &GOTO METHOD_TWO
    &END
&WHEN 3
    &DO
    &GOTO METHOD_THREE
    &END
&END
/*
/* -----
/* METHOD ONE IS SIMPLE RANDOM SELECTION OF THE STRATIFIED POPULATION.
/* -----
&LABEL METHOD ONE
&DATA %EXECUTE% %PICK_PGM%
%INFOPATH%
%INCOVER%.POP
%CATITEM% EQ [QUOTE %CATVAL%]
%NOSAMP%
%TOLER%
&END
/*
&S CELTM :=
&GOTO REPORT
/*
/* -----
/* METHOD TWO IS RANDOM SELECTION OF EQUAL-AREA CELLS OF THE STRATIFIED
/* POPULATION.
/* -----

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&LABEL METHOD_TWO
/*
/*  CREATE AN ASCII FILE OF LABEL POINTS FOR THE SUBAREA POLYGONS
/*
&IF [EXISTS SUBA.ASCII -FILE] &THEN &S DELETER := [DELETE SUBA.ASCII]
UNGENERATE POINT %INCOVER%.SUBA SUBA.ASCII
/*
/*  PREPARE EQUAL-AREA CELLS
/*
&TYPE
&TYPE %BOLD_ON%Creating %NOSAMP% equal-area cells%BOLD_OFF%
&DATA %EXECUTE% %EQAR_PGM%
%INFOPATH%
%INCOVER%.SUBA
%CATITEM%
%CATVAL%
%NOSAMP%
%TOLER%
&END
/*
&S DELETER := [DELETE SUBA.ASCII]
/*
/*  OVERLAY THE POINT POPULATION ONTO THE EQUAL-AREA CELLS
/*
&IF [EXISTS %INCOVER%.SAM -COVER] &THEN KILL %INCOVER%.SAM
IDENTITY %INCOVER%.POP %INCOVER%.SUBA %INCOVER%.SAM POINT 0.001
/*
/*  CREATE POLYGONS OF THE CELLS FOR ILLUSTRATIVE PURPOSES
/*
&TYPE %BOLD_ON%Preparing coverage of equal-area cells%BOLD_OFF%
&S CCOVER := %INCOVER%.CELL.%CATVAL%
&IF [EXISTS %CCOVER% -COVER] &THEN KILL %CCOVER%
DISSOLVE %INCOVER%.SUBA %CCOVER% CELL POLY
/*
/*  RANDOMLY SELECT SITES FROM EACH OF THE EQUAL-AREA CELLS
/*
&DO I := 1 &TO %NOSAMP%
&TYPE %BOLD_ON%Selecting sites from cell %I%BOLD_OFF%
&TYPE
&DATA %EXECUTE% %PICK_PGM%
%INFOPATH%
%INCOVER%.SAM
CELL = %I% AND %CATITEM% EQ [QUOTE %CATVAL%]
1
%TOLER%
&END
&END
/* END OF DATA TO PICKEM
/* END OF SELECTION LOOP
/*
&S CELITM := CELL
&GOTO REPORT
/*
/* -----
/* METHOD THREE IS RANDOM SELECTION FROM CELLS DEFINED BY AN ITERATIVE
/* AREAL DISTRIBUTION ALGORITHM.
/* -----
&LABEL METHOD_THREE
&S PROMPT := 'Enter fraction of cells to attain '~
[CALC %TOLER% + 1]' sites for convergence, (CR = '%PERCENT%')'
&S PERCENT := [RESPONSE %PROMPT% %PERCENT%]
&IF [TYPE %PERCENT%] > 0 &THEN &DO
&TYPE Numeric value required
&GOTO METHOD_THREE
&END
&IF %PERCENT% < 0 | %PERCENT% > 1 &THEN &DO
&TYPE
&TYPE Fraction of cells must be a value between zero and one
&GOTO METHOD_THREE
&END
/*

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&LABEL GET_MAX
&S PROMPT := 'Enter maximum number of iterations, (CR = '%MAXITER%')'
&S MAXITER := [RESPONSE %PROMPT% %MAXITER%]
&IF [TYPE %MAXITER%] ^= -1 &THEN &DO
    &TYPE Integer value required
    &GOTO GET_MAX
&END
&IF %MAXITER% <= 0 &THEN &DO
    &TYPE Positive number required
    &GOTO GET_MAX
&END
/*
&LABEL GET_DAMP
&S PROMPT := 'Enter damping factor, (CR = '%DAMPER%')'
&S DAMPER := [RESPONSE %PROMPT% %DAMPER%]
&IF [TYPE %DAMPER%] > 0 &THEN &DO
    &TYPE Numeric value required
    &GOTO GET_DAMP
&END
&IF %DAMPER% <= 0 &THEN &DO
    &TYPE
    &TYPE Damping factor should not be less than or equal to zero
    &GOTO GET_DAMP
&END
/*
/* FOR AN INITIAL GUESS AT THE ITERATION PARAMETERS:
/* SET THE CELL POPULATION TO ACCOUNT FOR THE FRACTION OF THE AREA WHICH
/* CONSISTS OF THE DESIRED POLYGON CATEGORY VERSUS THE TOTAL STUDY AREA.
/*
&TYPE %BOLD_ON%Making an initial guess at grid characteristics...%BOLD_OFF%
&S ITER := 0
/*
&DATA %EXECUTE% %INFL_PGM%
%INFOPATH%
%CATVAL%
%NOSAMP%
%TOLER%
&END
/*
&S AMLU := [OPEN INFLATED OPENERR -READ]
&S INFLATE := [UNQUOTE [TRIM [READ %AMLU% EOF]]]
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE INFLATED]
/*
&IF %INFLATE% = 0 &THEN &GOTO SAMPLE
/*
/* PREPARE A POLYGON GRID USING THE CURRENT ITERATION PARAMETERS.
/*
&LABEL ITERATE
/*
&TYPE %BOLD_ON%Defining cell characteristics...%BOLD_OFF%
&S ITER := [CALC %ITER% + 1]
&DATA %EXECUTE% %GRID_PGM%
%INCOVER%
1
%INFLATE%
&END
/*
/* READ THE RESULTS FROM THE FORTRAN PGM IN THE OUTPUT FILE
/*
&S AMLU := [OPEN %INCOVER%.GRD OPENERR -READ]
&S LINE := [TRIM [READ %AMLU% EOF]]
&DO VAR &LIST XMIN XMAX YMIN YMAX XDIM YDIM INFLATE
    &S TEMP := [BEFORE %LINE% ' ']
    &S %VAR% := %TEMP%
    &S LINE := [TRIM [AFTER %LINE% %TEMP%]]
&END
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE %INCOVER%.GRD]

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

/*
/* PREPARE A GRIDDED COVERAGE WITH LABEL POINTS.
/*
&TYPE
&TYPE %BOLD_ON%Constructing a grid with %INFLATE% cells...%BOLD_OFF%

&S CCOVER := %INCOVER%.CELL.%CATVAL%
&IF [EXISTS %CCOVER% -COVER] &THEN KILL %CCOVER%
GENERATE %CCOVER%
COPYTICS %INCOVER%
GRID LABELS
%XMINT%, %YMIN%
%XMINT%, %YMAX%
%XDIM%, %YDIM%
0, 0
%XMAX%, %YMAX%
QUIT
/*
/* CREATE POLYGON TOPOLOGY FOR THE CELL COVERAGE.
/*
BUILD %CCOVER% POLY
&S CELITM := [UNQUOTE [SUBSTR %CCOVER%-ID 1 16]]
/*
/* OVERLAY THE CELLS ONTO THE POINT POPULATION.
/*
&IF [EXISTS %INCOVER%.SAM -COVER] &THEN KILL %INCOVER%.SAM
IDENTITY %INCOVER%.POP %CCOVER% %INCOVER%.SAM POINT 0.001
/*
/* CHECK FOR CONVERGENCE, IF CONVERGED, THEN SELECT SITES...
/*
&TYPE %BOLD_ON%Checking for convergence...%BOLD_OFF%
&DATA %EXECUTE% %MTHR_PGM%
%INFOPATH%
%INCOVER%.SAM
%CELITM%
%CATITEM%
%CATVAL%
%NOSAMP%
%INFLATE%
%TOLER%
%PERCENT%
%DAMPER%
&END
/*
&S AMLU := [OPEN INFLATED OPENERR -READ]
&S INFLATE := [UNQUOTE [TRIM [READ %AMLU% EOF]]]
&S CLOSERR := [CLOSE %AMLU%]
&S DELETER := [DELETE INFLATED]
/*
/* IF CONVERGENCE WAS NOT ATTAINED, ITERATE
/*
&IF %INFLATE% ^= -99998 &THEN &DO
    &TYPE %BOLD_ON%Convergence failed on iteration #%ITER%%BOLD_OFF%
    &IF %ITER% < %MAXITER% &THEN &GOTO ITERATE
    &TYPE %BOLD_ON%'Maximum number of iterations exceeded'%BOLD_OFF%
    &GOTO SAMPLE
&END
/*
/* -----
&LABEL REPORT
/*
/* UPDATE THE INFO CATEGORY INDEX FILE: CATIDX.
/*
&TYPE %BOLD_ON%Site selection completed....Updating index file...%BOLD_OFF%
&DATA %EXECUTE% %UCAT_PGM%
%INFOPATH%
%CATVAL%
%NOSAMP%
&END

```

Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

/*
/* SAVE RESULTS OF SITE SELECTION IN THE POPULATION COVERAGE FOR METHODS TWO AND
/* THREE. METHOD ONE RECORDED ITS RESULTS DIRECTLY IN THE POPULATION COVERAGE.
/*
&IF %METHOD% ^= 1 &THEN &DO
&TYPE %BOLD_ON%Recording results in population coverage%BOLD_OFF%
/*
&WORKSPACE %INFOPATH%
&DATA INFO
ARC
SELECT %INCOVER%.SAM.PAT
RELATE %INCOVER%.POP.PAT UNIQUE
RESELECT %CATITEM% = [QUOTE %CATVAL%]
CALC %ICELLNO = %CELITM%
CALC %ICHOSEN = CHOSEN
Q STOP
&END
&WORKSPACE %HOMEX%
KILL %INCOVER%.SAM
&END
/*
/* RECORD SITE SELECTION NETWORK IN A PLOT FILE.
/*
&TYPE %BOLD_ON%Preparing plot file: %INCOVER%.%CATVAL%%BOLD_OFF%
  ARC PLOT
  DISP 1039
  %INCOVER%.%CATVAL%
  &TYPE %INCOVER%.%CATVAL%
  PAGESIZE 7.5 10.0
  MAPEXTENT %INCOVER%
  MAPPOSITION CEN CEN
  POLYS %INCOVER%
  LINECOLOR 3
  &IF %METHOD% ^= 1 &THEN POLYGONS %CCOVER%
  MARKERSET WATER.MRK
  MARKERSYM 109
  TEXTFONT 3
  RESELECT %INCOVER%.POP POINTS %CATITEM% = [QUOTE %CATVAL%] AND CHOSEN > 0
  POINTS %INCOVER%.POP
  POINTTEXT %INCOVER%.POP CHOSEN CC
  QUIT
&IF [EXISTS %INCOVER%.GRID -COVER] &THEN KILL %INCOVER%.GRID
/*
/* PREPARE SORTED LIST OF CHOSEN SITES.
/*
&TYPE
&TYPE %BOLD_ON%Saving locations of selected sites...%BOLD_OFF%
&IF [EXISTS %SAMSRT] &THEN &S DELETER := [DELETE %SAMSRT]
&IF [EXISTS %SAMPST] &THEN &S DELETER := [DELETE %SAMPST]
&DATA %EXECUTE% %RPT1_PGM%
%INFOPATH%
%INCOVER%.POP
%CATITEM%
%CATVAL%
%CELITM%
&END
/*
&TYPE %BOLD_ON%Projecting to latitude-longitude...%BOLD_OFF%
&IF [EXISTS %SAMSRT.GEO] &THEN &S DELETER := [DELETE %SAMSRT.GEO]
PROJECT FILE %SAMSRT %SAMSRT.GEO %PROJ_DES%
/*
/* FORMAT REPORT SHOWING THE RESULTS OF THE SITE SELECTION
/*

```



Attachment A.--Arc macro language program SAMPLE.AML listing--Continued

```

&TYPE %BOLD_ON%Locations projected.....formatting report...%BOLD_OFF%
&DATA %EXECUTE% %RPT2_PGM%
%INFOPATH%
%CATVAL%
%NOSAMP%
%TOLER%
&END
&S DELETER := [DELETE SAMPTS]
&S DELETER := [DELETE SAMSRT]
&S DELETER := [DELETE SAMSRT.GEO]
&TYPE
&TYPE %BOLD_ON%Report is ready in the file:  SITE.REPORT.%CATVAL%%BOLD_OFF%
/*
&GOTO SELECT_OPTION

```

Attachment B.--Statements inserted into several Fortran programs

```

C SAMPLE.INSERT - PROBLEM-DEPENDANT ARRAY LIMITS
C
C   MAXIMUM NUMBER OF POTENTIAL SITES IN A CATEGORY
C   PARAMETER (MAXPOP = 10000)
C
C   MAXIMUM NUMBER OF SUBAREAS IN A CATEGORY
C   PARAMETER (MAXSUB = 20000)
C
C   MAXIMUM NUMBER OF CATEGORIES IN THE STUDY REGION
C   PARAMETER (MAXCAT = 100)
C
C   MAXIMUM NUMBER OF CELLS
C   PARAMETER (MAXCEL = 1000)
C
C   MAXIMUM NUMBER OF POTENTIAL SITES IN A CELL
C   PARAMETER (MAXPNT = 10000)

```

Attachment C.--Subroutines used in several Fortran programs

```

      SUBROUTINE IRSEED (SSMID)
C
C INTERLUDE TO PRIMOS ROUTINE TO RETRIEVE NUMBER OF SECONDS SINCE MIDNIGHT
C (FOR USE AS A SEED FOR RANDOM NUMBER GENERATION):  TIMDAT
C
      INTEGER*2 TDARRAY(28)
      CALL TIMDAT (TDARRAY,INTS(28))
      SSMID = TDARRAY (5)
      RETURN
      END

      SUBROUTINE ISRTFL (FILEIN,FILOUT,NKEY,KEYSTR,KEYEND,NPASS,NITEM)
C
C INTERLUDE TO PRIMOS FILE-SORTING ROUTINE:  SUBSRT
C
      INTEGER*2 LENIN          /* LENGTH OF INPUT FILE (IN BYTES)
      INTEGER*2 LENOUT         /* LENGTH OF OUTPUT FILE (IN BYTES)
      INTEGER*2 NUMKEY          /* NUMBER OF SORT KEYS
      INTEGER*2 NSTART (9)     /* VECTOR OF SORT KEY START BYTES
      INTEGER*2 NEND (9)       /* VECTOR OF SORT KEY END BYTES (output)
      INTEGER*2 NPASS2          /* NUMBER OF PASSES FOR SORT (output)
      INTEGER*4 NPASS           /* FOUR BYTE COPY OF NPASS2
      INTEGER*4 NITEM           /* NUMBER OF ITEMS SORTED
      INTEGER*4 NKEY            /* FOUR BYTE COPY OF NUMKEY
      INTEGER*4 NKEYSTR (NKEY) /* FOUR BYTE COPY OF NSTART
      INTEGER*4 NKEYEND (NKEY) /* FOUR BYTE COPY OF NEND
      CHARACTER*128 FILEIN      /* INPUT FILENAME (UNSORTED)
      CHARACTER*128 FILOUT      /* OUTPUT FILENAME (SORTED)

```

Attachment C.--Subroutines used in several Fortran programs--Continued

```

C
  LENIN = INTS (LENGTH (FILEIN,128))
  LENOUT = INTS (LENGTH (FILOUT,128))
  NUMKEY = NKEY
  DO 30 I=1,MIN (NKEY,9)
    NSTART(I) = KEYSTR(I)
    NEND (I) = KEYEND(I)
30  CONTINUE
C
  CALL SUBSRT (FILEIN,LENIN,FILOUT,LENOUT,NUMKEY,NSTART,NEND,
&    NPASS2,NITEM)
  NPASS = NPASS2
  RETURN
  END

  INTEGER FUNCTION LENGTH (STRING,LEN)
C
C  FUNCTION TO DETERMINE THE LENGTH OF A STRING
C
  CHARACTER*(*) STRING
  DO 10 I=LEN,1,-1
10  IF (STRING(I:I) .NE. ' ') GOTO 20
  I = 0
20  LENGTH = I
  RETURN
  END

```

Attachment D.--Fortran program ITEXST listing

```

  PROGRAM ITEXST
C
C  TEST IF AN ITEM EXISTS IN AN INFO FILE
C  IF YES: WRITE .TRUE. TO OUTPUT FILE
C  IF NO: WRITE .FALSE. TO OUTPUT FILE
C  JONATHON SCOTT, LAST MODIFIED OCTOBER 24, 1989
C
  CHARACTER DIRECT*128, FILE*32, NAMITM*16, USER*4
  INTEGER ITEMAR(4)
  PARAMETER (ITWO=2)
C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
  CALL MESINI
  CALL LUNINI
  CALL MINIT
  CALL INFINT
C
C2 -- GET DATA FROM CALLING PROGRAM
  CALL PRMSTR ('Directory of INFO data base is ',31)
  READ (*,'(A)') DIRECT
  CALL PRMSTR ('INFO user name is ',18)
  READ (*,'(A)') USER
  CALL PRMSTR ('INFO file name is ',18)
  READ (*,'(A)') FILE
  CALL PRMSTR ('Variable name is ',17)
  READ (*,'(A)') NAMITM
C
C3 -- OPEN THE INFO FILE, TEST FOR ITEM NAME, CLOSE INFO FILE
  CALL ACREAT (LUN,'TSTINF',IER)
  CALL INFOPN (FILE,DIRECT,USER,ITWO,INUM,NUMREC,IEN,IER)
  IF (IER .NE. 0) THEN
    CALL INFORM ('INFO file does not exist',0)
    WRITE (LUN,'(A)') '.FALSE.'
    GOTO 900
  ENDIF
  CALL INFEXI (INUM,NAMITM,ITEMAR,IEXIST)
  CALL INFCLS (INUM)

```

Attachment D.--Fortran program ITEXST listing--Continued

```

C
C4 -- WRITE RESULTS TO OUTPUT FILE
      IF (IEXIST .EQ. 0) THEN
        CALL INFORM ('Variable name does not exist',0)
        WRITE (LUN,'(A)') '.FALSE.'
      ELSE
        WRITE (LUN,'(A)') '.TRUE.'
      ENDIF
C
C5 -- FINISHED, CLOSE FILES
900  ENDFILE      (LUN)
      CALL ACLOSE (LUN)
      CALL AEXIT
      END

```

Attachment E.--Fortran program GRIDIM listing

```

      PROGRAM GRIDIM
C
C GRIDIM COMPUTES GRID DATA FOR USE WITH THE ARC/INFO GENERATE COMMAND
C JONATHON SCOTT, LAST MODIFIED JULY 25, 1990
C
      CHARACTER COVER*128, FILEN*32
      DOUBLE PRECISION BOX(4)
      EXTERNAL MINIT, LUNINI, BOXGET
C
C1 -- INITIALIZE THE ARC/INFO SUBROUTINE LIBRARIES
      CALL MESINI
      CALL MINIT
      CALL LUNINI
C
C2 -- GET DATA FROM THE CALLING PROGRAM
      CALL PRMSTR ('Boundary used from the coverage ',32)
      READ (*,'(A)') COVER
      CALL PRMSTR ('Method code is ',15)
      READ (*,'(I20)') IFLAG
      IF (IFLAG .EQ. 1) THEN
        CALL PRMSTR ('Approximate number is ',25)
      ELSE
        CALL PRMSTR ('Distance is ',15)
      ENDIF
      READ (*,'(I20)') NPTS
C
C3 -- GET THE BOUNDARIES OF THE STUDY REGION
      CALL BOXGET (COVER,BOX,IERR)
      XMIN = BOX(1)
      YMIN = BOX(2)
      XMAX = BOX(3)
      YMAX = BOX(4)
C
C4 -- CALCULATE: LENGTH OF AXES & TOTAL AREA WITHIN THE COVERAGE BOUNDARIES
      XLEN = XMAX - XMIN
      YLEN = YMAX - YMIN
      AREA = XLEN * YLEN
C
C5 -- DEPENDING UPON THE METHOD CHOSEN, SET THE DISTANCE BETWEEN POINTS
      IF (IFLAG .EQ. 1) THEN
C
C6A -- WHEN IFLAG = 1: NPTS = APPROXIMATE NUMBER OF POINTS TO BE GENERATED
        CELL = AREA / NPTS
        SIDE = SQRT (CELL)
        XDIM = SIDE
        YDIM = SIDE
      ELSE

```

Attachment E.--Fortran program GRIDIM listing--Continued

```

C
C6B -- WHEN IFLAG = 2: NPTS = THE DISTANCE BETWEEN POINTS
      XDIM = NPTS
      YDIM = NPTS
      ENDIF
C
C8 -- SEED RANDOM NUMBER GENERATOR
      CALL IRSEED (SSMID)
      RAND = RND (SSMID)
C
C7 -- GET A RANDOM NUMBER AND USE IT TO SET THE ORIGIN OF THE COORDINATES
      RAND = RND (0.)
      XMIN = XMIN - RAND * XDIM
      RAND = RND (0.)
      YMIN = YMIN - RAND * YDIM
C
C8 -- CALCULATE THE NUMBER OF POINTS TO BE GENERATED IN EACH DIMENSION
      XNUM = (XMAX - XMIN) / XDIM + 0.5
      YNUM = (YMAX - YMIN) / YDIM + 0.5
C
C9 -- LENGTHEN MAXIMA TO GET AROUND BUG IN GENERATE
      XMAX = XMAX + XDIM
      YMAX = YMAX + YDIM
C
C10 - CALCULATE THE TOTAL NUMBER OF POINTS
      NCELL = INT(XNUM) * INT(YNUM)
C
C11 - PLACE THE COMPUTED DATA INTO THE RETURN-ARGUMENT TEMPORARY FILE
      FILEN = COVER (1:LENGTH(COVER,128)) // '.GRD'
      CALL ACREAT (LUN,FILEN,IER)
      WRITE (LUN,'(6F15.3,I10)') XMIN,XMAX,YMIN,YMAX,XDIM,YDIM,NCELL
      ENDFILE (LUN)
      CALL ACLOSE (LUN)
      CALL AEXIT
      END

```

Attachment F.--Fortran program DEFIDX listing

```

      PROGRAM DEFIDX
C
C DEFIDX IS RUN ONCE TO CREATE A POLYGON CATEGORY INDEX (CATIDX) INFO FILE
C THE CATIDX FILE IS USED TO KEEP INFORMATION ABOUT AREAL SUBSET CATEGORIES
C THERE IS ONE RECORD IN THE FILE FOR EACH CATEGORY
C JONATHON SCOTT, LAST MODIFIED OCTOBER 23, 1989
C
$INSERT SAMPLE.INSERT
      CHARACTER DIRECT*128, POLCOV*32, PIDCOV*32, FILE*32, CATIDX*32
      CHARACTER CATITM*16, ALTERN*16, NAMITM*16, AFTER*16
      CHARACTER USER*4, STRING*16
      CHARACTER INFEXP*80, MSG*80
      CHARACTER QUOTE*1
      CHARACTER*16 CATS(MAXCAT)
C
      INTEGER ITEMAR(4), INREC(1024)
      INTEGER EVLEXP(2,80), EXPTYP
      DOUBLE PRECISION REALLY, AREA, TOTARE
C
      PARAMETER (USER='ARC',CATIDX='CATIDX',ALTERN='',AFTER='')
      PARAMETER (ISIX=6,IFIVE=5,IFOUR=4,ITHREE=3,ITWO=2,IONE=1,IZERO=0)
      PARAMETER (NEGONE=-1)

```

Attachment F.--Fortran program DEFIDX listing--Continued

```

C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
    CALL MESINI
    CALL LUNINI
    CALL MINIT
    CALL INFINT
    QUOTE = ''''

C
C2 -- GET DATA FROM THE CALLING PROGRAM
    CALL PRMSTR ('Directory of INFO data base is ',31)
    READ (*,'(A)') DIRECT
    CALL PRMSTR ('Coverage of areal subsets is ',29)
    READ (*,'(A)') POLCOV
    CALL PRMSTR ('Coverage of population is ',26)
    READ (*,'(A)') PIDCOV
    CALL PRMSTR ('Category-variable name is ',26)
    READ (*,'(A)') CATITM

C
C3 -- IF THE INDEX INFO FILE DOES EXIST DELETE IT, THEN CREATE & DEFINE IT...
    CALL INFERS (CATIDX,DIRECT,USER,IONE,IER)
    CALL INFDEF (CATIDX,DIRECT,USER,IZERO,IDXNUM,IER)
    IF (IER .NE. 0) CALL ERROR ('ERROR DURING INFDEF',0)

C
C3A -- DEFINE: CATEGORY,16,C
    NAMITM = 'CATEGORY'
    IWIDTH = 16
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,ITWO,IWIDTH,
1              IWIDTH,NEGONE,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)
    IF (IER .NE. 0) THEN
        CALL INFCLS (IDXNUM)
        CALL ERROR ('Error adding item',0)
    ENDIF

C
C3B -- DEFINE: AREA,4,12,F,3
    NAMITM = 'AREA'
    IWIDTH = 12
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,ISIX,IFOUR,
1              IWIDTH,ITHREE,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)

C
C3C -- DEFINE: NUMPOLY,4,5,B
    NAMITM = 'NUMPOLY'
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,IFIVE,IFOUR,
1              IFIVE,NEGONE,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)

C
C3D -- DEFINE: NUMPTS,4,5,B
    NAMITM = 'NUMPTS'
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,IFIVE,IFOUR,
1              IFIVE,NEGONE,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)

C
C3E -- DEFINE: FRACTION,4,12,F,4
    NAMITM = 'FRACTION'
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,ISIX,IFOUR,
1              IWIDTH,IFOUR,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)

C
C3F -- DEFINE: NOSAMP,4,5,B
    NAMITM = 'NOSAMP'
    CALL INFADI (IDXNUM,IZERO,NAMITM,ALTERN,AFTER,IFIVE,IFOUR,
1              IFIVE,NEGONE,IFOUR,IZERO,NEGONE,NEGONE,NEGONE,
2              NEGONE,IZERO,IER)
    IDXLEN = 36

```

Attachment F.--Fortran program DEFIDX listing--Continued

```

C
C4 -- OPEN THE STUDY REGION AREAL SUBSET (PAT) INFO-FILE
      FILE = POLCOV (1:LENGTH(POLCOV,32)) // '.PAT'
      CALL INFOPN (FILE,DIRECT,USER,ITWO,NFPOLY,NUMPOL,IPOLLN,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (IDXNUM)
        CALL ERROR ('Unable to open polygon attribute table',0)
      ENDIF

C
C5 -- OPEN THE SITE POPULATION (PAT) INFO-FILE
      FILE = PIDCOV (1:LENGTH(PIDCOV,32)) // '.PAT'
      CALL INFOPN (FILE,DIRECT,USER,IONE,NFPID,NUMPNT,IPNTLN,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (IDXNUM)
        CALL INFCLS (NFPOLY)
        CALL ERROR ('Unable to open point attribute table',0)
      ENDIF

C
C6 -- CHECK TO MAKE SURE THAT THE POLYGON CATEGORY-ITEM NAME EXISTS
      CALL INFEXI (NFPOLY,CATITM,ITEMAR,IEXIST)
      IF (IEXIST .EQ. 0) THEN
        CALL INFCLS (IDXNUM)
        CALL INFCLS (NFPID)
        CALL INFCLS (NFPOLY)
        CALL ERROR ('Category-variable name does not exist',0)
      ENDIF
      LENCAT = ITEMAR(2)

C
C8 -- LOOP THROUGH POLYGONS & KEEP TRACK OF ALL THE DIFFERENT CATEGORIES
C      (SKIP THE 1ST POLYGON, BECAUSE IT IS THE EXTERIOR 'POLYGON')
      NUMCAT = 1
      DO 20 IPOLY=1,NUMPOL
        CALL INFGET (NFPOLY,IPOLY,INREC,IER)
        CALL INFDEC (INREC,IPOLLN,ITEMAR,REALLY,STRING,IER)
        IEXIST = 0
        IF (NUMCAT .GT. 1) THEN
          DO 10 I=1,NUMCAT-1
            IF (CATS(I) .EQ. STRING) IEXIST = 1
10          CONTINUE
          ENDIF
          IF (IEXIST .EQ. 0) THEN
            CATS (NUMCAT) = STRING
            NUMCAT = NUMCAT + 1
            IF (NUMCAT .GT. MAXCAT) THEN
              CALL INFCLS (NFPOLY)
              CALL INFCLS (NFPID)
              CALL INFCLS (IDXNUM)
              CALL ERROR ('Resizing needed for MAXCAT',0)
            ENDIF
          ENDIF
        ENDIF
20      CONTINUE
      NUMCAT = NUMCAT - 1
      TOTARE = 0.0

C
C9 -- LOOP THROUGH THE POLYGON CATEGORIES & DESCRIBE EACH ONE
      DO 50 ICAT = 1, NUMCAT
        INFEXP = CATITM // ' EQ ' // QUOTE //
1        CATS(ICAT)(1:LENCAT) // QUOTE
        CALL INFTRN (NFPOLY,INFEXP,EVLEXP,EXPTYP,IER)
        AREA = 0.0
        NPOLY = 0
      END DO

```

Attachment F.--Fortran program DEFIDX listing--Continued

```

C
C9A -- LOOP THROUGH ALL THE POLYGONS & FIND OUT TOTAL AREA AND
C      NUMBER OF POLYGONS IN EACH CATEGORY (IGNORING EXTERIOR POLYGON)
      NAMITM = 'AREA'
      CALL INFEXI (NFPOLY,NAMITM,ITEMAR,IEXIST)
      DO 30 IPOLY = 1,NUMPOL
        CALL INFSEL (NFPOLY,IPOLY,EVLEXP,IEXIST)
        IF (IEXIST.EQ. 1) THEN
          CALL INFGET (NFPOLY,IPOLY,INREC,IER)
          CALL INFDEC (INREC,IPOLLN,ITEMAR,REALLY,STRING,IER)
          IF (REALLY.GT. 0.) THEN
            AREA = AREA + REALLY
            NPOLY = NPOLY + 1
          ENDIF
        ENDIF
      ENDIF
30    CONTINUE
      TOTARE = TOTARE + AREA
C
C9B -- LOOP THE SITES & FIND OUT THE NUMBER OF SITES IN EACH CATEGORY
      NPTS = 0
      CALL INFCLE (EVLEXP)
      CALL INFTRN (NFPID,INFEXP,EVLEXP,EXPTYP,IER)
      DO 40 IPNT = 1,NUMPNT
        CALL INFSEL (NFPID,IPNT,EVLEXP,IEXIST)
        IF (IEXIST.NE. 0) NPTS = NPTS + 1
40    CONTINUE
      CALL INFCLE (EVLEXP)
C
C9C -- WRITE TO THE CATIDX FILE THE INFORMATION ABOUT THE CATEGORY
      NAMITM = 'CATEGORY'
      STRING = CATS(ICAT)
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
      NAMITM = 'AREA'
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,AREA, STRING,IDXLEN,INREC,IER)
      NAMITM = 'NUMPOLY'
      REALLY = NPOLY
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
      NAMITM = 'NUMPTS'
      REALLY = NPTS
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
      NAMITM = 'NOSAMP'
      REALLY = 0.
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
C
      CALL INPPUT (IDXNUM,ICAT,INREC,IER)
50    CONTINUE
C
C10 - COMPUTE & STORE THE AREA-WEIGHTED FRACTIONS
      DO 60 ICAT=1,NUMCAT
        NAMITM = 'AREA'
        CALL INFGET (IDXNUM,ICAT,INREC,IER)
        CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
        CALL INFDEC (INREC,IDXLEN,ITEMAR,REALLY,STRING,IER)
        REALLY = REALLY / TOTARE
        NAMITM = 'FRACTION'
        CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
        CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
        CALL INPPUT (IDXNUM,ICAT,INREC,IER)
60    CONTINUE

```

Attachment F.--Fortran program DEFIDX listing--Continued

```

C
C11 - FINISHED
      WRITE (MSG,'(A,I4)') 'Number of categories defined = ',NUMCAT
      CALL INFORM (MSG,0)
      CALL INFCLS (NFPID)
      CALL INFCLS (NFPOLY)
      CALL INFCLS (IDXNUM)
      CALL AEXIT
      END

```

Attachment G.--Fortran program CHKIDX listing

```

      PROGRAM CHKIDX
C
C CHKIDX IS RUN EACH TIME A VALUE IS ENTERED FOR THE CATEGORY VARIABLE.
C THE PROGRAM READS THE CATEGORY INDEX FILE (CATIDX) IN INFO
C THE NUMBER OF SITES SELECTED FOR THE CATEGORY (NOSAMP) IS RETURNED.
C NEGATIVE ONE IS RETURNED IF THE CATEGORY IS NOT FOUND IN THE INDEX.
C JONATHAN SCOTT, LAST MODIFIED OCTOBER 11, 1989
C
      CHARACTER DIRECT*128, CATIDX*32, FNAME*32
      CHARACTER CATITM*16, NAMITM*16, CATNAM*16, STRING*16
      CHARACTER USER*4
      CHARACTER INFEXP*80
      CHARACTER QUOTE*1
C
      INTEGER ITEMAR(4), INREC(1024)
      INTEGER EVLEXP(2,80), EXPTYP
      DOUBLE PRECISION REALLY
C
      PARAMETER (USER='ARC',CATIDX='CATIDX',ITWO=2,QUOTE='''')
      PARAMETER (CATITM='CATEGORY',FNAME='T$NOSAMP')
C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
      CALL MESINI
      CALL LUNINI
      CALL MINIT
      CALL INFINT
C
C2 -- GET DATA FROM CALLING PROGRAM
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATNAM
C
C3 -- OPEN THE CATEGORY INDEX INFO FILE (CATIDX)
      CALL INFOPN (CATIDX,DIRECT,USER,ITWO,IDXNUM,NUMCAT,IDXLEN,IER)
      IF (IER .NE. 0)
        1 CALL ERROR('Unable to open category-index file',0)
C
C4 -- FIND THE RECORD FOR THE CATEGORY
      INFEXP = CATITM // ' EQ ' // QUOTE // CATNAM(1:LENGTH(CATNAM,16))
      1 // QUOTE
      CALL INFTRN (IDXNUM,INFEXP,EVLEXP,EXPTYP,IER)
      DO 50 ICAT = 1,NUMCAT
        CALL INFSEL (IDXNUM,ICAT,EVLEXP,IEXIST)
        IF (IEXIST .NE. 0) GOTO 100
50 CONTINUE
      NOSAMP = -1
      GOTO 200

```



Attachment G.--Fortran program CHKIDX listing--Continued

```

C
C6 -- READ FROM THE CATIDX FILE THE NUMBER OF SITES SELECTED FOR THE CATEGORY
100 CALL INFGET (IDXNUM,ICAT,INREC,IER)
    NAMITM = 'NOSAMP'
    CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
    CALL INFDEC (INREC,IDXLEN,ITEMAR,REALLY,STRING,IER)
    NOSAMP = REALLY
C
C6 -- FINISHED.
200 CALL ACREAT (LUN,FNAME,IER)
    WRITE (LUN,'(I6)') NOSAMP
    ENDFILE (LUN)
    CALL ACLOSE (LUN)
    CALL INFCLS (IDXNUM)
    CALL AEXIT
    END

```

Attachment H.--Fortran program PICKEM listing

```

SUBROUTINE PICKEM
C
C RANDOMLY SELECTS SITES FROM A GROUP DEFINED BY A SELECTION EXPRESSION
C JONATHON SCOTT, LAST MODIFIED MARCH 27, 1990
C
    CHARACTER FILE*32, DIRECT*128, USER*8, ITEM*16, INSTR*1
    CHARACTER COVER*32, INFEXP*320, MSG*80
    INTEGER FNUM, NUMREC, RECLEN, IER, EXISTS, IZERO
    INTEGER RECNUM, REC(1024), ITEMAR(4)
    INTEGER ETYPE, EVLEXP(2,80)
    DOUBLE PRECISION REALLY
C
$INSERT SAMPLE.INSERT
    INTEGER CHOSEN(MAXPNT), RECORD(MAXPNT)
    COMMON CHOSEN, RECORD
    EXTERNAL MINIT, LUNINI, INFINT, INFNAM, INFOPN, INFEXI
    EXTERNAL INFENC, INFGET, INFPUT
C
    PARAMETER (USER='ARC',ITEM='CHOSEN',INSTR='')
    PARAMETER (ITYPE=1,IACCESS=2)
C
C1 -- INITIALIZE THE ARC/INFO LIBRARIES
    CALL MESINI
    CALL MINIT
    CALL LUNINI
    CALL INFINT
C
C2 -- GET COVERAGE NAME, PATH NAME, ITEM NAME, & ITEM VALUE
C    GENERATE ARC/INFO FILENAME FOR POINT ATTRIBUTE TABLE
    CALL PRMSTR ('Directory of INFO data base is ',31)
    READ (*,'(A)') DIRECT
    CALL PRMSTR ('Coverage of population is ',26)
    READ (*,'(A)') COVER
    CALL PRMSTR ('Selection expression is ',24)
    READ (*,'(A)') INFEXP
    CALL INFNAM (COVER,ITYPE,FILE)
C
C3 -- CHECK TO MAKE SURE THE POINT ATTRIBUTE TABLE CAN BE FOUND, THEN OPEN IT
    CALL INFEXF (FILE,DIRECT,USER,EXISTS)
    IF (EXISTS .EQ. 0)
    * CALL ERROR ('Attribute table not found for population',0)
    CALL INFOPN (FILE,DIRECT,USER,IACCESS,FNUM,NUMREC,RECLEN,IER)
    IF (IER .NE. 0)
    * CALL ERROR ('Unable to open population coverage PAT',0)

```

Attachment H.--Fortran program PICKEM listing--Continued

```

C
C4 -- MAKE SURE THERE ARE SOME POINTS IN THIS COVERAGE
      IF (NUMREC .EQ. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('No sites in population',0)
      ENDIF
C
C5 -- FIND OUT HOW MANY SITES AND ALTERNATE SITES ARE TO BE SELECTED
      CALL PRMSTR ('Number of primary sites is ',27)
      READ (*,*) NOSAMP
      CALL PRMSTR ('Number of alternate sites is ',29)
      READ (*,*) NDUP
      NDUP = NDUP + 1
      NUMSEL = 0
C
C6 -- TRANSLATE THE SELECTION EXPRESSION INTO ARC/INFO ENCRYPTION
      CALL INFTRN (FNUM,INFEXP,EVLEXP,ETYPE,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Invalid selection expression',0)
      ENDIF
C
C7 -- CHECK EVERY RECORD IN THE POINT COVERAGE FOR THE SELECTION CRITERIA
      DO 100 I=1,NUMREC
        CALL INFSEL (FNUM,I,EVLEXP,EXISTS)
        IF (EXISTS .EQ. 1) THEN
C
C7A --      MAKE SURE THAT THE PROGRAM HAS SUFFICIENT SPACE ALLOCATED
              NUMSEL = NUMSEL + 1
              IF (NUMSEL .GT. MAXPNT) THEN
                CALL INFCLS (FNUM)
                CALL ERROR ('Resizing needed for MAXPNT',0)
              ENDIF
C
C7B --      SAVE THE RECORD NUMBER OF RECORDS MEETING THE CRITERIA
              RECORD (NUMSEL) = I
              CHOSEN (NUMSEL) = 0
            ENDIF
          100 CONTINUE
          WRITE (MSG,'(I5,A)') NUMSEL, ' sites in the cell'
          CALL INFORM (MSG,0)
C
C8 -- CHECK THAT A LEGITIMATE SUBSETTING EXPRESSION WAS ENTERED
      IF (NUMSEL .EQ. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Unable to select from an empty subset',0)
      ENDIF
C
C9 -- CHECK THAT SUFFICIENT SITES EXIST TO PERFORM THE DESIRED SELECTION
      NTOTAL = NOSAMP * NDUP
      IF (NTOTAL .GT. NUMSEL) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Selection quantity exceeds cell population',0)
      ENDIF
C
C10 - SEED RANDOM NUMBER FUNCTION
      CALL IRSEED (SSMID)
      RAND = RND (SSMID)
C
C11 - SELECT FOR THE SUBSET FOR THE SPECIFIED NUMBER OF SITES AND ALTERNATES
      DO 400 IROUND = 1, NDUP
        NFOUND = 0
C
C11A - GET A RANDOM NUMBER FUNCTION & SCALE RESULTANT
      300 RAND = RND (0.)
        RAND = RAND * FLOAT(NUMSEL) + 1.0
        IF (RAND .GT. FLOAT(NUMSEL)) RAND = NUMSEL
        IRAND = RAND

```

Attachment H.--Fortran program PICKEM listing--Continued

```

C
C11B - CHECK TO MAKE SURE WE HAVEN'T ALREADY CHOSEN THAT ONE
      IF (CHOSEN (IRAND) .GT. 0) GOTO 300
C
C11C - GOT ONE, SAVE IT...
      NFOUND = NFOUND + 1
      CHOSEN (IRAND) = IROUND
      IF (NFOUND .LT. NOSAMP) GOTO 300
400  CONTINUE
C
C12 - GET THE ITEM DESCRIPTION ARRAY FOR THE SELECTION-ROUND NUMBER
      CALL INFEXI (FNUM,ITEM,ITEMAR,EXISTS)
C
C13 - WRITE TO THE POINT ATTRIBUTE TABLE THE ROUND IN WHICH EACH SITE
C      IN THE SELECTED SUBSET WAS SELECTED FOR SELECTION (0 = NOT SELECTED)
      DO 500 I=1,NUMSEL
C
C13A -- RETRIEVE THE RECORD FOR THE SITE
      CALL INFGET (FNUM,RECORD(I),REC,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Error retrieving site record',0)
      ENDIF
C
C13B -- ENCODE THE SELECTION-ROUND NUMBER INTO THE RECORD
      REALLY = CHOSEN (I)
      CALL INFENC (ITEMAR,REALLY,INSTR,RECLN,REC,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Error encoding site record',0)
      ENDIF
C
C13C -- WRITE THE RECORD BACK INTO THE POINT ATTRIBUTE TABLE
      CALL INFPUT (FNUM,RECORD(I),REC,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Error writing site record',0)
      ENDIF
500  CONTINUE
C
C14 - CLOSE THE POINT ATTRIBUTE TABLE AND EXIT
      CALL INFCLS (FNUM)
      CALL AEXIT
      END

```

Attachment I.--Fortran program EQARSA listing

```

      PROGRAM EQARSA
C
C EQARSA IS RUN TO RANDOMLY SELECT SITES FROM A STUDY REGION USING CELLS
C OF EQUAL AREA. THE USER CONTROLS THE CATEGORY, THE NUMBER OF
C CELLS, AND THE NUMBER OF SITES TAKEN FROM EACH CELL.
C JONATHON SCOTT, LAST MODIFIED SEPTEMBER 13, 1989
C
      CHARACTER DIRECT*128, COVER*32
      CHARACTER FILE*32, FILEIN*128, FILEOUT*128
      CHARACTER NAMITM*16, CATITM*16, CATVAL*16
      CHARACTER USER*4, STRING*16
      CHARACTER INFEXP*80
      CHARACTER QUOTE*1
      LOGICAL FORWRD, DBG
      INTEGER ITEMAR(4), ITEMID(4), ITEMVS(4), ITEMML(4)
      INTEGER INREC(1024),KEYSTR(2), KEYEND(2)
      INTEGER EVLEXP(2,80), EXPTYP
      DOUBLE PRECISION REALLY, AREA, TOTARE
      *INSERT SAMPLE.INSERT

```

Attachment I.--Fortran program EQARSA listing--Continued

```

      INTEGER    ID (MAXSUB), KSTRIP (MAXSUB)
      COMMON     ID,          KSTRIP

C
C1 -- DEFINE VARIABLES, INITIALIZE ARC LIBRARY ROUTINES
      PARAMETER (USER='ARC')
      PARAMETER (ITWO=2,IONE=1)
      DBG = .FALSE.
      QUOTE = ''''
      CALL MESINI
      CALL LUNINI
      CALL MINIT
      CALL INFINT

C
C2 -- GET PATHNAME, COVER NAMES, & ITEM-NAME OF POLYGON CATEGORY VARIABLE
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Coverage of subarea polygons is ',32)
      READ (*,'(A)') COVER
      CALL PRMSTR ('Category-variable name is ',26)
      READ (*,'(A)') CATITM
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATVAL
      CALL PRMSTR ('Number of primary sites is ',27)
      READ (*,*)      NOSAMP
      CALL PRMSTR ('Number of alternate sites is ',29)
      READ (*,*)      NDUPS
      NDUPS = NDUPS + 1

C
C3 -- OPEN THE POLYGON ATTRIBUTE TABLE INFO FILE
      FILE = COVER (1:LENGTH(COVER,32)) // '.PAT'
      CALL INFOFN (FILE,DIRECT,USER,ITWO,NFPOLY,NUMPOL,IPOLLN,IER)
      IF (IER .NE. 0) CALL ERROR ('Unable to open PAT',0)

C
C4 -- CHECK TO MAKE SURE THAT THE POLYGON CATEGORY ITEM-NAME EXISTS
      CALL INFEXI (NFPOLY,CATITM,ITEMAR,IEXIST)
      IF (IEXIST .EQ. 0) THEN
        CALL INFCLS (NFPOLY)
        CALL ERROR ('Category-variable name does not exist',0)
      ENDIF
      LENCAT = ITEMAR(2)

C
C5 -- COMPOSE SELECTION EXPRESSION FOR THE CATEGORY
      INFEXP = CATITM // ' EQ ' // QUOTE //
      *      CATVAL (1:LENGTH(CATVAL,16)) // QUOTE
      CALL INFTRN (NFPOLY,INFEXP,EVLEXP,EXPTYP,IER)

C
C6 -- EXAMINE ALL POLYGONS AND CALCULATE THE TOTAL AREA & NUMBER OF POLYGONS
      (SKIP THE EXTERIOR 'POLYGON')
      TOTARE = 0.0
      NPOLY = 0
      NAMITM = 'AREA'
      CALL INFEXI (NFPOLY,NAMITM,ITEMAR,IEXIST)
      DO 30 IPOLY = 1,NUMPOL
        CALL INFSEL (NFPOLY,IPOLY,EVLEXP,INCAT)
        IF (INCAT .LE. 0.) GOTO 30
        CALL INFGET (NFPOLY,IPOLY,INREC,IER)
        CALL INFDEC (INREC,IPOLLN,ITEMAR,REALLY,STRING,IER)
        TOTARE = TOTARE + ABS (REALLY)
        NPOLY = NPOLY + 1
30    CONTINUE
      IF (DBG) WRITE (*,'(A)') 'CATEGORY AREA CALCULATED'

C
C7 -- CALCULATE:  NVSTRP = NUMBER OF VERTICAL STRIPS
C                ARSTRP = AREA OF EACH VERTICAL STRIP
C                ARCELL = AREA OF EACH CELL

```

Attachment I.--Fortran program EQARSA listing--Continued

```

ARSTRP = TOTARE / SQRT (FLOAT (NOSAMP))
NVSTRP = INT ( TOTARE / ARSTRP + 0.5 )
ARCELL = TOTARE / FLOAT (NOSAMP)
PCT = ARCELL / TOTARE * 100.
WRITE (*,'(A,I17,2(/,A,F20.2),/,A,F20.2,2H (,F4.1,2H%))')
*      'Number of vertical strips = ', NVSTRP,
*      '      Total area      = ', TOTARE,
*      '      Strip area      = ', ARSTRP,
*      '      Equal area for cell = ', ARCELL, PCT
ISTRIP = 1
STRIPA = 0.

C
C8 -- SORT FILE OF COORDINATES BY HORIZONTAL POSITION ON THE MAP
FILEIN = 'SUBA.ASCII'
FILOUT = 'SUBA.SORT'
KEYSTR(1) = 11
KEYEND(1) = 25
CALL ISRTFL (FILEIN,FILOUT,1,KEYSTR,KEYEND,NPASS,NITEM)
IF (DBG) WRITE (*,'(A)') 'LABELS SORTED BY HORIZONTAL POSITION'

C
C9 -- OPEN & SORTED LABEL POINT FILE, AND GET ITEM DESCRIPTIONS FOR:
C      COVER-ID, AREA, AND STRIP NUMBER
CALL AOPEN (LUN,FILOUT,IER)
CALL ACREAT (LOUT,FILEIN,IER)
NAMITM = COVER (1:LENGTH (COVER,32)) // '-ID'
CALL INFEXI (NFPOLY,NAMITM,ITEMID,IEXIST)
NAMITM = 'AREA'
CALL INFEXI (NFPOLY,NAMITM,ITEMAR,IEXIST)
NAMITM = 'VSTRIP'
CALL INFEXI (NFPOLY,NAMITM,ITEMVS,IEXIST)
READ (LUN,'(A)') FILE

C
C10 -- READ THE LABEL POINTS IN SORTED ORDER AND ACCUMULATE INTO VERTICAL STRIPS
21  CONTINUE
      READ (LUN,'(I10,15X,F15.6)',END=28) LABID, YCOORD
      LABSTR = LABID
      LABEND = NUMPOL

C
C10A -- GET THE RECORD NUMBER OF POLYGON
      CALL FNDLAB
*      (LABID,LABSTR,LABEND,NFPOLY,NUMPOL,IPOLLN,ITEMID,ILAB,INREC)

C
C10B -- DETERMINE IF POLYGON IS IN THE DESIRED CATEGORY,
C      IF NOT: INCLUDE IN VERTICAL STRIP, BUT DON'T INCLUDE IN STRIP AREA.
CALL INFSEL (NFPOLY,ILAB,EVLEXP,INCAT)
IF (INCAT .EQ. 0) GOTO 223

C
C10C -- GET AREA OF THE POLYGON
CALL INFDEC (INREC,IPOLLN,ITEMAR,REALLY,STRING,IER)

C
C10D -- IGNORE EXTERIOR POLYGON
IF (REALLY .LE. 0.0) GOTO 27

C
C10E -- ADD POLYGON TO CURRENT VERTICAL STRIP, OR START A NEW STRIP
WITH = ABS (STRIPA + REALLY - ARSTRP)
WITHO = ABS (STRIPA - ARSTRP)
IF (WITH .LE. WITHO) THEN
  STRIPA = STRIPA + REALLY
ELSE
  STRIPA = REALLY
  IF (ISTRIP .LT. NVSTRP) ISTRIP = ISTRIP + 1
ENDIF

C
C10F -- PLACE THE VERTICAL STRIP NUMBER INTO THE POLYGON ATTRIBUTE RECORD,
C      ALSO WRITE TO ASCII FILE: RECORD NUMBER, STRIP NUMBER, AND Y-COORDINATE

```

Attachment I.--Fortran program EQARSA listing--Continued

```

223      CONTINUE
        REALLY = FLOAT (ISTRIP)
        CALL INFENC (ITEMVS,REALLY,STRING,IPOLLN,INREC,IER)
        CALL INFPUT (NFPOLY,ILAB,INREC,IER)
        WRITE (LOUT,'(2(I10,1X),F15.6)') ILAB, ISTRIP, YCOORD
27       CONTINUE
        GOTO 21
C
C11 - ALL SUBAREA LABEL POINTS HAVE BEEN ASSIGNED TO A VERTICAL STRIP;
C      CLOSE SORTED LABEL POINT FILE, AND DELETE IT; CLOSE ASCII FILE.
28      CONTINUE
        CALL ACLOSE (LUN)
        CALL ADELET (FILOUT,IER)
        CALL ACLOSE (LOUT)
        IF (DBG) WRITE (*,'(A)') 'VERTICAL STRIPS COMPLETED'
C
C12 - SORT ASCII FILE OF COORDINATES BY STRIP NUMBER AND Y-COORDINATE
        KEYSTR(1) = 12
        KEYEND(1) = 21
        KEYSTR(2) = 23
        KEYEND(2) = 37
        CALL ISRTFL (FILEIN,FILOUT,2,KEYSTR,KEYEND,NPASS,NITEM)
        IF (DBG) WRITE (*,'(A)')
*      'SORT BY VERTICAL STRIP & VERTICAL POSITION COMPLETED'
C
C13 - READ THE SORTED STRIP NUMBER AND RECORD NUMBER INTO MEMORY;
C      CLOSE AND DELETE ASCII FILES
        IF (NITEM .GT. MAXSUB) THEN
            CALL INFCLS (NFPOLY)
            CALL ERROR ('Resizing needed for MAXSUB',0)
        ENDIF
        CALL AOPEN (LUN,FILOUT,IER)
        DO 101 I=1,NITEM
101      READ (LUN,'(I10,1X,I10)') ID(I), KSTRIP(I)
        CALL ADELET (FILEIN,IER)
        CALL ACLOSE (LUN)
        CALL ADELET (FILOUT,IER)
C
C14 - INITIALIZE LOOP VARIABLES FOR THE ASSIGNMENT OF CELL NUMBERS
        NAMITM = 'CELL'
        CALL INFEXI (NFPOLY,NAMITM,ITEMLL,IEXIST)
        CELLA = 0.0
        NOUT = 0
        ICELL = 1
        IPNT = 1
        LSTRIP = 1
        NDONE = 0
        FORWRD = .TRUE.
        IF (DBG) WRITE (*,'(A)') 'Starting assignment of cells...'
        WRITE (*,440)
440      FORMAT (/,10X,' Equal-area grid cells:',
*              //,10X,'Cell      Area      Percentage')
C
C15 - GET A POLYGON RECORD CORRESPONDING TO THE RECORD NUMBER OF THE NEXT -ID
42      CONTINUE
        CALL INFGET (NFPOLY,ID(IPNT),INREC,IER)
C
C15A -- CHECK IF THE SUBAREA BELONGS TO THE CATEGORY
C      IF NOT, THEN ENCODE THE CELL NUMBER W/O INCLUDING IN THE AREA
C      FOR THE CELL
        CALL INFSEL (NFPOLY,ID(IPNT),EVLEXP,INCAT)
        IF (INCAT .EQ. 0) GOTO 443
C
C15B -- GET AREA AND CHECK IF IT BELONGS IN THIS CELL
        CALL INFDEC (INREC,IPOLLN,ITEMAR,REALLY,STRING,IER)
        ARPOLY = REALLY
        IF (ARPOLY .LE. 0.0) WRITE (*,'(A)') 'FATAL LOGIC ERROR'

```

Attachment I.--Fortran program EQARSA listing--Continued

```

      WITH = ABS (CELLA + ARPOLY - ARCELL)
      WITHO = ABS (CELLA - ARCELL)
      IF (WITH .LE. WITHO) THEN
        CELLA = CELLA + ARPOLY
      ELSE
        PCT = 100. * CELLA / TOTARE
        NOUT = NOUT + 1
        WRITE (*,'(10X,I3,G15.6E3,3X,A,F4.1,A)')
          *      ICELL, CELLA, '(',PCT,')'
        CELLA = ARPOLY
        IF (ICELL .NE. NOSAMP) ICELL = ICELL + 1
      ENDIF

C
C15C -- SAVE THE STRIP NUMBER OF THE CURRENT POLYGON
443  CONTINUE
      LSTRIP = KSTRIP (IPNT)

C
C15D -- STORE THE CELL NUMBER WITH THE POLYGON ATTRIBUTES
      REALLY = FLOAT (ICELL)
      CALL INFENC (ITEMLL,REALLY,STRING,IPOLLN,INREC,IER)
      CALL INFPUT (NFPOLY,ID(IPNT),INREC,IER)

C
C15E -- IF FINISHED THE ENTIRE STUDY AREA, THEN EXIT THE LOOP
      NDONE = NDONE + 1
      IF (NDONE .EQ. NITEM) GOTO 50

C
C15F -- FIND THE NEXT POLYGON NUMBER TO BE CHECKED
43  CONTINUE
      IF (FORWRD) THEN
        IPNT = IPNT + 1

C
C15F1 -- WHEN MOVING UP & HIT THE TOP OF STRIP, GO TO TOP OF NEXT STRIP &
C        MOVE DOWNWARD THROUGH THE LIST OF POLYGONS
      IF (KSTRIP (IPNT) .GT. LSTRIP) THEN
44      IPNT = IPNT + 1
        IF (KSTRIP (IPNT) .EQ. KSTRIP (IPNT-1) ) GOTO 44
        IPNT = IPNT - 1
        FORWRD = .FALSE.
        IF (DBG) WRITE (*,'(A)') 'MOVING DOWN'
      ENDIF
    ELSE
      IPNT = IPNT - 1

C
C15F2 -- WHEN MOVING DOWN & HIT THE BOTTOM OF STRIP, GO TO BOTTOM OF NEXT
C        STRIP & MOVE UPWARD THROUGH THE LIST OF POLYGONS
      IF (KSTRIP (IPNT) .LT. LSTRIP .OR. IPNT .EQ. 1) THEN
46      IPNT = IPNT + 1
        IF (KSTRIP (IPNT) .EQ. KSTRIP (IPNT+1) ) GOTO 46
        IPNT = IPNT + 1
        FORWRD = .TRUE.
        IF (DBG) WRITE (*,'(A)') 'MOVING UP'
      ENDIF
    ENDIF
      GOTO 42

C
C16 - ALL POLYGONS HAVE BEEN ASSIGNED TO A CELL
50  CONTINUE
      IF (NOUT .NE. NOSAMP) THEN
        PCT = 100. * CELLA / TOTARE
        WRITE (*,'(10X,I3,G15.6E3,3X,A,F4.1,A)')
          *      ICELL, CELLA, '(',PCT,')'
      ENDIF
      CALL INFCLS (NFPOLY)
      CALL AEXIT
      END

```

Attachment I.--Fortran program EQARSA listing--Continued

```

C
  SUBROUTINE FNDLAB (LABID,LABSTR,LABEND,NFPOLY,NUMPOL,IPOLLN,
*                  ITEMID,ILAB,INREC)
  INTEGER ITEMID(4), INREC(1024)
  CHARACTER STRING*16
  DOUBLE PRECISION REALLY
  DO 22 ILAB = LABSTR,LABEND
    CALL INFGET (NFPOLY,ILAB,INREC,IER)
    CALL INFDEC (INREC,IPOLLN,ITEMID,REALLY,STRING,IER)
    IF (INT(REALLY) .EQ. LABID) GOTO 25
22  CONTINUE
  CALL INFCLS (NFPOLY)
  CALL ERROR ('FNDLAB LOGIC FAILING ',0)
  IF (LABSTR .NE. 1) THEN
    DO 23 ILAB = 1, LABSTR-1
      CALL INFGET (NFPOLY,ILAB,INREC,IER)
      CALL INFDEC (INREC,IPOLLN,ITEMID,REALLY,STRING,IER)
      IF (INT(REALLY) .EQ. LABID) GOTO 25
23  CONTINUE
  ENDIF
  CALL INFCLS (NFPOLY)
  CALL ERROR ('Search for label point failed during FNDLAB',0)
25  CONTINUE
  RETURN
  END

```

Attachment J.--Fortran program NFLATE listing

```

      PROGRAM NFLATE
C
C  CALCULATE AN INITIAL GUESS FOR THE NUMBER OF CELLS WHEN SELECTING SITES
C  WITH ITERATING GRIDS.  THE INITIAL GUESS IS BASED UPON THE RATIO OF CATEGORY
C  AREA TO STUDY UNIT AREA SCALED BY THE NUMBER OF SITES TO BE SELECTED.
C  NORMALLY, THIS GUESS IS TOO HIGH.
C  JONATHAN SCOTT, LAST MODIFIED MARCH 27, 1990
C
  CHARACTER DIRECT*128, CATIDX*32
  CHARACTER CATVAL*16, NAMITM*16
  CHARACTER USER*4, STRING*16
  CHARACTER INFEXP*80
  CHARACTER QUOTE*1, MSG*80
C
  INTEGER ITEMAR(4), INREC(1024)
  INTEGER EVLEXP(2,80), EXPTYP
  DOUBLE PRECISION REALLY
C
  PARAMETER (USER='ARC',CATIDX='CATIDX',QUOTE='''',IONE=1)
C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
  CALL MESINI
  CALL LUNINI
  CALL MINIT
  CALL INFINT
C
C2 -- GET DATA FROM THE CALLING PROGRAM
  CALL PRMSTR ('Directory of INFO data base is ',31)
  READ (*,'(A)') DIRECT
  CALL PRMSTR ('Category is ',12)
  READ (*,'(A)') CATVAL
  CALL PRMSTR ('Number of primary sites is ',27)
  READ (*,*) NOSAMP
  CALL PRMSTR ('Number of alternate sites is ',29)
  READ (*,*) ITOL
  ITOL = ITOL + 1

```



Attachment J.--Fortran program NFLATE listing--Continued

```

C
C3 -- OPEN THE POINT ATTRIBUTE TABLE INFO FILE
      CALL INFOFN (CATIDX,DIRECT,USER,IONE,NFCAT,NUMCAT,ICATLN,IER)
      IF (IER .NE. 0) CALL ERROR ('Unable to open category-index',0)
C
C4 -- COMPOSE THE RECORD SELECTION EXPRESSION
      INFEXP = 'CATEGORY EQ ' // QUOTE // CATVAL // QUOTE
      CALL INFTRN (NFCAT,INFEXP,EVLEXP,EXPTYP,IER)
C
C5 -- FIND THE RECORD THAT DESCRIBES THE CATEGORY
      NAMITM = 'FRACTION'
      CALL INFEXI (NFCAT,NAMITM,ITEMAR,IEXIST)
      DO 10 ICAT = 1,NUMCAT
        CALL INFSEL (NFCAT,ICAT,EVLEXP,IEXIST)
        IF (IEXIST .EQ. 1) THEN
          CALL INFGET (NFCAT,ICAT,INREC,IER)
          CALL INFDEC (INREC,ICATLN,ITEMAR,REALLY,STRING,IER)
          FRACTN = REALLY
          NAMITM = 'NUMPTS'
          CALL INFEXI (NFCAT,NAMITM,ITEMAR,IEXIST)
          CALL INFDEC (INREC,ICATLN,ITEMAR,REALLY,STRING,IER)
          NLEAST = ITOL * NOSAMP
          IF (REALLY .LT. NLEAST) THEN
            WRITE(MSG,'(A,2(A,I8),A)') 'Insufficient ',
1          'population (' ,REALLY,') to select ',NLEAST,' sites.'
            CALL INFORM (MSG,0)
            GOTO 15
          ENDIF
          GOTO 20
        ENDIF
      CONTINUE
      CALL ERROR ('Category does not exist',0)
15    FRACTN = 1
      NOSAMP = 0
C
C6 -- WRITE THE INFLATED NUMBER OUT FOR THE GRIDDING PROGRAM
20    CALL INFCLS (NFCAT)
      NOSAMP = INT (FLOAT (NOSAMP) / FRACTN)
      CALL ACREAT (LUN,'INFLATED',IER)
      WRITE (LUN,'(I10)') NOSAMP
      ENDFILE (LUN)
      CALL ACLOSE (LUN)
      CALL AEXIT
      END

```

Attachment K.--Fortran program SAMINF listing

PROGRAM SAMINF

```

C
C SAMINF IS RUN ITERATIVELY TO DETERMINE IF THE CORRECT GRID HAS
C BEEN PREPARED FOR AN AREALLY-DISTRIBUTED RANDOM SELECTION OF SITES.
C WHEN THE CORRECT GRID HAS BEEN PREPARED, THE SELECTION IS PERFORMED.
C THE VARIABLE 'CHOSEN' IN THE PAT FILE IS SET TO THE SELECTION-ROUND
C FOR SELECTED SITES AND SET TO ZERO FOR SITES NOT SELECTED.
C AN OUTPUT FILE IS WRITTEN WITH ONE RECORD INDICATING A NEW TARGET GRID
C POPULATION WHEN ITERATION IS NEEDED, OR A ZERO IF SELECTION HAS BEEN
C PERFORMED BECAUSE THE ALGORITHM CONVERGED.
C JONATHON SCOTT, LAST MODIFIED SEPTEMBER 27, 1989
C
      CHARACTER DIRECT*128, PIDCOV*32, FILE*32
      CHARACTER NAMITM*16, GRDITM*16, CATVAL*16, CATITM*16
      CHARACTER USER*4, STRING*16
      CHARACTER INFEXP*80, MSG*80
      CHARACTER QUOTE*1

```

Attachment K.--Fortran program SAMINF listing--Continued

```

C
      INTEGER   ITEMAR(4),   INREC(1024)
      INTEGER   EVLEXP(2,80), EXPTYP
      DOUBLE PRECISION REALLY
C
      PARAMETER (USER='ARC',QUOTE='''',ITWO=2)
C
$INSERT SAMPLE.INSERT
      INTEGER   NPOP   (MAXCEL)
      INTEGER   SAMREC (MAXPOP), INGRID (MAXPOP)
      INTEGER   CHOSEN (MAXPNT), ISAMPL (MAXPNT)
      COMMON    CHOSEN, ISAMPL, SAMREC, INGRID, NPOP
C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
C
      CALL MESINI
      CALL LUNINI
      CALL MINIT
      CALL INFINT
C
C2 -- GET DATA FROM THE CALLING PROGRAM
C
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Coverage of iteration results is ',33)
      READ (*,'(A)') PIDCOV
      CALL PRMSTR ('Cell-variable name is ',22)
      READ (*,'(A)') GRDITM
      CALL PRMSTR ('Category-variable name is ',26)
      READ (*,'(A)') CATITM
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATVAL
      CALL PRMSTR ('Number of primary sites is ',27)
      READ (*,*)   NOSAMP
      CALL PRMSTR ('Current number of cells is ',27)
      READ (*,*)   INFSAM
      CALL PRMSTR ('Number of alternate sites is ',29)
      READ (*,*)   ITOL
      ITOL = ITOL + 1
      CALL PRMSTR ('Percentage required for convergence is ',39)
      READ (*,*)   PERCNT
      CALL PRMSTR ('Damping factor is ',18)
      READ (*,*)   DAMPER
C
C3 -- DESCRIBE CELL POPULATIONS IN THE CURRENT GRID
C
C3A -- OPEN THE (PAT) INFO-FILE DESCRIBING THE POPULATION
C
      FILE = PIDCOV (1:LENGTH(PIDCOV,32)) // '.PAT'
      CALL INFOPN (FILE,DIRECT,USER,ITWO,NFPID,NUMPNT,IPNTLN,IER)
      IF (IER .NE. 0)
        * CALL ERROR ('Unable to open population coverage PAT',0)
C
C3B -- COMPOSE CATEGORY-SELECTION EXPRESSION, DESCRIBE CELL ITEM,
C      AND INITIALIZE POPULATION OF EACH CELL (NPOP) TO ZERO.
C
      INFEXP = CATITM // ' EQ ' // QUOTE // CATVAL // QUOTE
      CALL INFTRN (NFPID,INFEXP,EVLEXP,EXPTYP,IER)
      CALL INFEXI (NFPID,GRDITM,ITEMAR,IEXIST)
      DO 20 IGRID = 1,MAXCEL
        NPOP (IGRID) = 0
20    CONTINUE
C
C3C -- FOR SITES IN THE CATEGORY:
C      COUNT THE TOTAL NUMBER OF SITES           (ISAM,NUMSAM)
C      RECORD NUMBER IN INFO-FILE                (SAMREC)
C      WHICH GRID CELLS THEY ARE LOCATED IN      (INGRID)
C      NUMBER OF SITES IN EACH CELL              (NPOP)
C

```

Attachment K.--Fortran program SAMINF listing--Continued

```

    IGRIDE = 0
    ISAM = 0
    DO 30 IPNT = 1, NUMPNT
        CALL INFSEL (NFPID, IPNT, EVLEXP, IEXIST)
        IF (IEXIST .EQ. 1) THEN
            CALL INFGET (NFPID, IPNT, INREC, IER)
            CALL INFDEC (INREC, IPNTLN, ITEMAR, REALLY, STRING, IER)
            ISAM = ISAM + 1
            IF (ISAM .GT. MAXPOP) THEN
                CALL INFCLS (NFPID)
                CALL ERROR ('Resizing needed for MAXPOP', 0)
            ENDIF
            SAMREC (ISAM) = IPNT
            INTGER = REALLY
            INGRID (ISAM) = INTGER
            IF (INTGER .GT. MAXCEL) THEN
                CALL INFCLS (NFPID)
                CALL ERROR ('Resizing needed for MAXCEL', 0)
            ENDIF
            NPOP (INTGER) = NPOP (INTGER) + 1
            IF (INTGER .GT. IGRIDE) IGRIDE = INTGER
        ENDIF
    30 CONTINUE
    NUMSAM = ISAM
    CALL INFCLE (EVLEXP)

C
C4 -- FIND OUT HOW MANY CELLS HAVE AT LEAST 'ITOL' POINTS          (NGDPOP),
C    HOW MANY SITES ARE IN GRID CELLS W/ LESS THAN 'ITOL'        (LSTPOP),
C    AND AVERAGE NUMBER OF SITES IN POPULATED CELLS              (AVGPOP).
C
    NGDPOP = 0
    LSTPOP = 0
    AVGPOP = 0
    IAVG = 0
    DO 40 IGRID = 1, IGRIDE
        IF (NPOP (IGRID) .GE. ITOL) THEN
            NGDPOP = NGDPOP + 1
        ELSE
            LSTPOP = LSTPOP + NPOP (IGRID)
        ENDIF
        IF (NPOP (IGRID) .NE. 0) THEN
            AVGPOP = AVGPOP + NPOP (IGRID)
            IAVG = IAVG + 1
        ENDIF
    40 CONTINUE
    AVGPOP = AVGPOP / FLOAT (IAVG)

C
C5 -- CONVERGENCE IS DETERMINED BASED UPON POPULATIONS OF THE GRID CELLS
C
    ICHECK = NOSAMP * PERCNT
    ILEAST = ITOL * (NOSAMP - NGDPOP)
    IF (NGDPOP .EQ. NOSAMP) THEN

C
C5A -- AT LEAST ITOL POINTS IN EXACTLY NOSAMP CELLS
C
        IREM = 0
        ELSEIF (NGDPOP .GE. ICHECK .AND. LSTPOP .GE. ILEAST .AND.
        1 NGDPOP .LT. NOSAMP) THEN

C
C5B -- AT LEAST PERCNT, BUT NO MORE THAN NOSAMP, CELLS HAVE ITOL POINTS;
C    AND THERE ARE SUFFICIENT REMAINING SITES FOR SELECTION
C
        IREM = NOSAMP - NGDPOP
    ELSE

C
C5C -- MUST ITERATE TO GET BETTER GRID
C

```

Attachment K.--Fortran program SAMINF listing--Continued

```

PORTON = FLOAT (NGDPOP) / FLOAT (NOSAMP)
IF (PORTON .LT. 0.60 .OR. PORTON .GT. 1.4) DAMPER = 1.0
ITRY = DAMPER * (NOSAMP-NGDPOP) * INFSAM / NGDPOP
ITRY = ITRY + INFSAM
WRITE (MSG,'(80(1H*))')
CALL INFORM (MSG,80)
WRITE (MSG,'(A,I6,A)') 'Iteration failed to converge using',
1 INFSAM, ' cells'
CALL INFORM (MSG,0)
WRITE (MSG,'(A,F5.1,A)') 'Average populated cell contained',
1 AVGPDP, ' sites'
CALL INFORM (MSG,0)
WRITE (MSG,'(I4,A,I4,A)') NGDPOP, ' cells contained at least',
1 ITOL, ' sites.'
CALL INFORM (MSG,0)
WRITE (MSG,'(A,I6,A)') 'Next iteration will use at least ',
1 ITRY, ' cells'
CALL INFORM (MSG,0)
WRITE (MSG,'(80(1H*))')
CALL INFORM (MSG,80)
GOTO 900
ENDIF
C
C8 -- CONVERGENCE HAS OCCURRED, PROCEEDING WITH SITE SELECTION
C
ITRY = -99998
C
C8A -- SELECT SITES FOR EACH GRID CELL THAT HAS ITOL POINTS
C
DO 300 IGRID = 1, IGRIDE
  IPICK = 0
  IPNT = 0
  ISAMSZ = NPOP (IGRID)
  IF (ISAMSZ .GT. MAXPNT) THEN
    CALL INFCLS (NFPID)
    CALL ERROR ('Resizing needed for MAXPNT',0)
  ENDIF
  IF (ISAMSZ .LT. ITOL) GOTO 300
C
C8A1 -- LOAD THE ISAMPL ARRAY WITH THE RECORD NUMBERS OF THE SITES
C
DO 110 ISAM = 1, NUMSAM
  IF (INGRID(ISAM) .EQ. IGRID) THEN
    IPNT = IPNT + 1
    IF (IPNT .GT. MAXPNT) THEN
      CALL INFCLS (NFPID)
      CALL ERROR ('Resizing needed for MAXPNT',0)
    ENDIF
    ISAMPL (IPNT) = SAMREC (ISAM)
  ENDIF
110 CONTINUE
  IPICK = ITOL
C
C8A2 -- RANDOMLY PICK ITOL SITES FROM THE SAMREC ARRAY
C
120 CALL RNDSEL (ISAMSZ,IPICK)
C
C8A3 -- UPDATE THE POINT ATTRIBUTE TABLE TO SET CHOSEN TO ROUND NUMBER
  NAMITM = 'CHOSEN'
  CALL INFEXI (NFPID,NAMITM,ITEMAR,EXISTS)
  DO 200 ISAM = 1, ISAMSZ
    CALL INFGET (NFPID,ISAMPL(ISAM),INREC,IER)
    REALLY = FLOAT (CHOSEN (ISAM))
    CALL INFENC (ITEMAR,REALLY,STRING,IPNTLN,INREC,IER)
    CALL INFPUT (NFPID,ISAMPL(ISAM),INREC,IER)
  200 CONTINUE
300 CONTINUE

```

Attachment K.--Fortran program SAMINF listing--Continued

```

C
C6B -- PICK THE REMAINING SITES FROM THE SPARSELY POPULATED CELLS
C
      NOSAMP = ITOL*IREM
      IPNT = 0
C
C6B1 -- LOAD THE ISAMPL ARRAY WITH THE LEFT-OVER SITES' RECORD NUMBERS
C
      DO 400 ISAM = 1,NUMSAM
        IF ( NPOP (INGRID (ISAM) ) .LT. ITOL) THEN
          IPNT = IPNT + 1
          IF (IPNT .GT. MAXPNT) THEN
            CALL INFCLS (NFPID)
            CALL ERROR ('Resizing needed for MAXPNT',0)
          ENDIF
          ISAMPL (IPNT) = SAMREC (ISAM)
        ENDIF
      400 CONTINUE
      IF (LSTPOP .GT. MAXPNT) THEN
        CALL INFCLS (NFPID)
        CALL ERROR ('Resizing needed for MAXPNT',0)
      ENDIF
C
C6B2 -- SELECT SITES FROM THE LEFT-OVERS
C
      CALL RNDSEL (LSTPOP,NOSAMP)
C
C6B3 -- SAVE ROUND NUMBER IN THE CHOSEN ITEM IN THE PAT
C
      DO 500 ISAM = 1, LSTPOP
        CALL INFGET (NFPID,ISAMPL(ISAM),INREC,IER)
        REALLY = FLOAT (CHOSEN (ISAM))
        CALL INFENC (ITEMAR,REALLY,STRING,IPNTLN,INREC,IER)
        CALL INFPUT (NFPID,ISAMPL(ISAM),INREC,IER)
      500 CONTINUE
C
C7 -- WRITE RESULTS OUT TO FILE TO BE READ BY CALLING PROGRAM
C
      900 CALL INFCLS (NFPID)
      CALL ACREAT (LUN,'INFLATED',IER)
      WRITE (LUN,'(I9)') ITRY
      ENDFILE (LUN)
      CALL ACLOSE (LUN)
      CALL AEXIT
      END
C
      SUBROUTINE RNDSEL (NUMSEL,NOSAMP)
C
C SUBROUTINE TO PERFORM RANDOM SELECTION...
C PICKS 'NOSAMP' SITES FROM 'ISAMPL' CHOICES RETURNING PICKS INTO 'CHOSEN'
C FROM A POPULATION OF 'NUMSEL'. CHOSEN = 0 -> NOT SELECTED
C
      $INSERT SAMPLE.INSERT
      INTEGER CHOSEN (MAXPNT)
      COMMON CHOSEN
C
C1 -- SEED RANDOM NUMBER FUNCTION, & INITIALIZE THE CHOSEN ARRAY
C
      CALL IRSEED (SSMID)
      RAND = RND (SSMID)
      DO 200 I=1,NUMSEL
        CHOSEN (I) = 0
      200 IF (NOSAMP .EQ. 0) RETURN
      NFOUND = 0
C
C2 -- CALL PRIMOS RANDOM NUMBER FUNCTION & SCALE RESULTANT
C

```

Attachment K.--Fortran program SAMINF listing--Continued

```

300  RAND = RND (0.)
      RAND = RAND * FLOAT(NUMSEL) + 0.5
      IRAND = RAND
      IF (IRAND .LT. 1) IRAND = 1
C
C3 -- CHECK TO MAKE SURE WE HAVEN'T ALREADY CHOSEN THAT ONE
C
      IF (CHOSEN (IRAND) .NE. 0) GOTO 300
C
C4 -- GOT ONE, SAVE ROUND NUMBER
C
      NFOUND = NFOUND + 1
      CHOSEN (IRAND) = NFOUND
      IF (NFOUND .LT. NOSAMP) GOTO 300
      RETURN
      END

```

Attachment L.--Fortran program UPTIDX listing

```

      PROGRAM UPTIDX
C
C UPTIDX IS RUN EACH TIME SITES ARE SELECTED FOR A CATEGORY
C THE PROGRAM UPDATES THE CATEGORY INDEX FILE (CATIDX) IN INFO
C THE NUMBER OF SITES SELECTED FOR THE CATEGORY IS STORED IN NOSAMP
C JONATHON SCOTT, LAST MODIFIED MARCH 27, 1990
C
      CHARACTER DIRECT*128, CATIDX*32
      CHARACTER CATITM*16, NAMITM*16, CATNAM*16, STRING*16
      CHARACTER USER*4
      CHARACTER INFEXP*80
      CHARACTER QUOTE*1
C
      INTEGER ITEMAR(4), INREC(1024)
      INTEGER EVLEXP(2,80), EXPTYP
      DOUBLE PRECISION REALLY
C
      PARAMETER (USER='ARC',CATIDX='CATIDX',ITWO=2,QUOTE='''')
      PARAMETER (CATITM='CATEGORY')
C
C1 -- INITIALIZE ARC LIBRARY ROUTINES
      CALL MESINI
      CALL LUNINI
      CALL MINIT
      CALL INFINT
C
C2 -- GET DATA FROM CALLING PROGRAM
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATNAM
      CALL PRMSTR ('Number of primary sites is ',27)
      READ (*,'(I10)') NOSAMP
C
C3 -- OPEN THE CATEGORY INDEX INFO FILE (CATIDX)
      CALL INFOPN (CATIDX,DIRECT,USER,ITWO,IDXNUM,NUMCAT,IDXLEN,IER)
      IF (IER .NE. 0)
        1 CALL ERROR('Unable to open category-index file',0)
C
C4 -- FIND THE RECORD FOR THE CATEGORY
      INFEXP = CATITM // ' EQ ' // QUOTE // CATNAM(1:LENGTH(CATNAM,16))
      1 // QUOTE
      CALL INFTRN (IDXNUM,INFEXP,EVLEXP,EXPTYP,IER)
      DO 50 ICAT = 1,NUMCAT
        CALL INFSEL (IDXNUM,ICAT,EVLEXP,IEXIST)
        IF (IEXIST .NE. 0) GOTO 100
50  CONTINUE

```

Attachment L.--Fortran program UPTIDX listing--Continued

```

      CALL INFCLE (EVLEXP)
      CALL INFCLS (IDXNUM)
      CALL ERROR ('Category not found in category-index',0)
C
C6 -- WRITE TO THE CATIDX FILE THE NUMBER OF SITES SELECTED FOR THE CATEGORY
100 CALL INFGET (IDXNUM,ICAT,INREC,IER)
      NAMITM = 'NOSAMP'
      REALLY = NOSAMP
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFENC (ITEMAR,REALLY,STRING,IDXLEN,INREC,IER)
      CALL INFPUT (IDXNUM,ICAT,INREC,IER)
C
C8 -- FINISHED.
      CALL INFCLS (IDXNUM)
      CALL AEXIT
      END

```

Attachment M.--Fortran program CATRPT listing

```

      PROGRAM CATRPT
C
C PREPARES DATA FOR A REPORT ON SITES WHICH HAVE BEEN SELECTED
C JONATHON SCOTT, LAST MODIFIED JULY 26, 1990
C
      CHARACTER DIRECT*128,COVER*32,FILE*32,USER*8,QUOTE*1,CATITM*16
      CHARACTER ITEM*16,POLNAM*16,STRING*1,INFEXP*80,CELITM*16
      CHARACTER FILNAM*128,CATVAL*16,SRTIN*128,SRTOUT*128
      INTEGER FNUM,NUMREC,RECLN,ITEMAR(4),INREC(1024),CHOSAR(4)
      INTEGER LABBUF(8),EVLEXP(2,80)
      INTEGER NSTART(2),NEND(2)
      DOUBLE PRECISION REALLY,XCOORD,YCOORD
C
      EQUIVALENCE (XCOORD,LABBUF(3)),(YCOORD,LABBUF(5))
      PARAMETER (USER='ARC',QUOTE=''',SRTIN='SAMPTS',SRTOUT='SAMSRT')
      PARAMETER (IONE=1,ITWO=2)
      DATA NSTART(1),NEND(1) / 1,10 /
      DATA NSTART(2),NEND(2) / 11,20 /
C
C1 -- INITIALIZE ARC/INFO LIBRARIES
      CALL MESINI
      CALL MINIT
      CALL LUNINI
      CALL INFINT
      CALL VINIT
C
C2 -- GET DATA FROM CALLING PROGRAM
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Coverage of population is ',26)
      READ (*,'(A)') COVER
      CALL PRMSTR ('Category-variable name is ',26)
      READ (*,'(A)') CATITM
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATVAL
      CALL PRMSTR ('Cell-variable name is ',22)
      READ (*,'(A)') CELITM
C
C3 -- OPEN THE SITE POPULATION INFO-FILE AND THE SITE COORDINATE FILE
      CALL INFNAM (COVER,IONE,FILE)
      CALL INFOPN (FILE,DIRECT,USER,ITWO,FNUM,NUMPNT,RECLN,IER)
      CALL INFEXI (FNUM,CATITM,ITEMAR,IEXIST)
      LENCAT = ITEMAR(2)
      CALL LABOPN (NFLAB,COVER,IONE,IONE,IER)
      IF (IER.NE.0) THEN
        CALL INFCLS (FNUM)
        CALL ERROR ('Unable to open LAB file.',24)
      ENDIF

```

Attachment M.--Fortran program CATRPT listing--Continued

```

C
C4 -- OPEN SEQUENTIAL FILE TO CONTAIN THE COORDINATES OF THE SELECTED SITES
      FILNAM = 'SAMPTS'
      CALL ACREAT (LUN,FILNAM,IER)
C
C5 -- COMPOSE AND TRANSLATE SELECTION EXPRESSION, DESCRIBE CHOSEN ITEM
      INFEXP = CATITM // ' EQ ' // QUOTE //
      1    CATVAL(1:LENCAT) // QUOTE // ' AND CHOSEN > 0 '
      CALL INFTRN (FNUM,INFEXP,EVLEXP,IEXTYP,IER)
      IF (IER .NE. 0) THEN
        CALL INFCLS (FNUM)
        CALL ACLOSE (LUN)
        CALL LABCLS (NFLAB)
        CALL ERROR ('Error occurred during INFTRN',0)
      ENDIF
      CALL INFEXI (FNUM,CELITM,ITEMAR,IEXIST)
      ITEM = 'CHOSEN'
      CALL INFEXI (FNUM,ITEM,CHOSAR,IEXIST)
C
C6 -- EXAMINE EVERY SITE IN THE POPULATION
      ISAM = 0
      DO 20 IPNT=1,NUMPNT
        CALL INFSEL (FNUM,IPNT,EVLEXP,IEXIST)
C
C6A -- WHEN THE SITE IS IN THE CATEGORY AND SELECTED...
        IF (IEXIST .NE. 0) THEN
          ISAM = ISAM + 1
C
C6B -- GET THE CELL AND ROUND NUMBER, AND COORDINATES OF SITE
          CALL INFGET (FNUM,IPNT,INREC,IER)
          IF (CELITM .NE. '') THEN
            CALL INFDEC (INREC,RECLN,ITEMAR,REALLY,STRING,IER)
            ICELL = REALLY
          ELSE
            ICELL = 0
          ENDIF
          CALL INFDEC (INREC,RECLN,CHOSAR,REALLY,STRING,IER)
          ICHOSE = REALLY
          CALL LABRDR (NFLAB,IPNT,LABBUF,IER)
C
C6C -- RECORD THESE DATA IN THE SEQUENTIAL FILE
          WRITE (LUN,'(2I10,2F16.3)') ICELL, ICHOSE, XCOORD, YCOORD
        ENDIF
      20 CONTINUE
C
C7 -- CLOSE SEQUENTIAL FILE, INFO-FILE, COORDINATE FILE
      ENDFILE (LUN)
      CALL INFCLS (FNUM)
      CALL ACLOSE (LUN)
      CALL LABCLS (NFLAB)
C
C8 -- SORT THE SITES BY THEIR CELL AND ROUND NUMBERS, THEN EXIT
      CALL ISRTFL (SRTIN,SRTOUT,ITWO,NSTART,NEND,NPASS,NITEM)
      CALL AEXIT
      END

```

Attachment N.--Fortran program FCATRP listing

```

      PROGRAM FCATRP
C
C FCATRP FORMATS A REPORT CONTAINING THE CHOSEN SITES FOR
C EACH POLYGON LISTING THE POINTS IN LAT-LONG. THIS PROCESS IS
C PRECEDED BY EXTRACTING THE POINTS, AND PROJECTING THEM TO LAT-LONG
C ONE REPORT IS PREPARED FOR EACH CATEGORY.
C JONATHON SCOTT, LAST MODIFIED APRIL 2, 1990
C

```



Attachment N.--Fortran program FCATRP listing--Continued

```

      CHARACTER CATVAL*16, FILE*32, LINE*40, STRING*16, NAMITM*16
      CHARACTER DIRECT*128, CATIDX*32, USER*4, QUOTE*1
      CHARACTER INFEXP*80
      INTEGER ICOORD(6), EVLEXP(2,80), EXPTYP, INREC(1024), ITEMAR(4)
      REAL COORD(6)
      DOUBLE PRECISION REALLY
      PARAMETER (CATIDX='CATIDX',USER='ARC',ITWO=2,QUOTE='''')

C
C1 -- INITIALIZE ARC/INFO SUBROUTINE PACKAGES
      CALL MESINI
      CALL LUNINI
      CALL MINIT
      CALL INFINT

C
C2 -- GET DATA FROM THE CALLING PROGRAM
      CALL PRMSTR ('Directory of INFO data base is ',31)
      READ (*,'(A)') DIRECT
      CALL PRMSTR ('Category is ',12)
      READ (*,'(A)') CATVAL
      CALL PRMSTR ('Number of primary sites is ',27)
      READ (*,*) NOSAMP
      CALL PRMSTR ('Number of alternate sites is ',29)
      READ (*,*) ITOL

C
C3 -- GET RECORD OF POLYGON CATEGORY FROM THE 'INFO' INDEX FILE
      CALL INFOPN (CATIDX,DIRECT,USER,ITWO,IDXNUM,NUMCAT,IDXLEN,IER)
      IF (IER .NE. 0) CALL ERROR ('Unable to open category-index',0)
      INFEXP = 'CATEGORY EQ ' // QUOTE // CATVAL(1:LENGTH(CATVAL,16))
      * // QUOTE
      CALL INFTRN (IDXNUM,INFEXP,EVLEXP,EXPTYP,IER)
      DO 44 ICAT = 1,NUMCAT
        CALL INFSEL (IDXNUM,ICAT,EVLEXP,IEXIST)
        IF (IEXIST .NE. 0) GOTO 66
44      CONTINUE
        CALL INFCLE (EVLEXP)
        CALL INFCLS (IDXNUM)
        CALL ERROR ('Unable to find specified category in index',0)
66      CONTINUE
        CALL INFGET (IDXNUM,ICAT,INREC,IER)
        CALL INFCLE (EVLEXP)

C
C4 -- READ THE CATEGORY DESCRIPTORS: NUMBER OF POLYGONS, TOTAL AREA,
C      NUMBER OF SITES IN THE CATEGORY
      NAMITM = 'NUMPOLY'
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFDEC (INREC,IDXLEN,ITEMAR,REALLY,STRING,IER)
      NPOLY = REALLY
      NAMITM = 'AREA'
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFDEC (INREC,IDXLEN,ITEMAR,REALLY,STRING,IER)
      AREA = REALLY
      NAMITM = 'NUMPTS'
      CALL INFEXI (IDXNUM,NAMITM,ITEMAR,IEXIST)
      CALL INFDEC (INREC,IDXLEN,ITEMAR,REALLY,STRING,IER)
      NPTS = REALLY
      CALL INFCLS (IDXNUM)

C
C5 -- OPEN REPORT FILE AND WRITE A HEADING DESCRIBING THE CATEGORY
      FILE = 'SITE.REPORT.' // CATVAL
      CALL AOPEN (LUIN,'SAMSRT.GEO',IER)
      CALL ACREAT (LUOUT,FILE,IER)
      WRITE (LUOUT,1001)
1  'RESULTS OF RANDOM SELECTION FOR CATEGORY ', CATVAL,
2  '      Total area of category          = ',AREA,
3  '      Number of areal subsets         = ',NPOLY,
4  '      Number of potential sites        = ',NPTS,
5  '      Number of primary sites selected = ',NOSAMP,
6  '      Number of alternate sites selected = ',ITOL

```

Attachment N.--Fortran program FCATRP listing--Continued

```

1001 FORMAT (1H1,A,1H',A,1H',/,1X,A,G15.0E3,4(/,1X,A,I10))
      IPNT = 1
      ICELL = 0
C
C6 -- LOOP THROUGH THE COORDINATE DATA FROM THE PRE-REPORT FILE
      IMAX = ITOL + 1
      NTOT = NOSAMP * IMAX
      DO 100 ISAM = 1,NTOT
        READ (LUIN,'(A)') LINE
C
C6A -- FOR A NEW CELL: WRITE A HEADING SHOWING THE SITE NUMBER
      IF ( IPNT .EQ. 1) THEN
        ICELL = ICELL + 1
        WRITE (LUOUT,1002) ICELL
1002   FORMAT (/,1X,13(1H-),' SITE #',I4,1X,13(1H-),
      *        /,1X,' ROUND      LATITUDE      LONGITUDE')
      ENDIF
C
C6B -- READ, FORMAT AND PRINT THE LOCATIONS SELECTED IN EACH GRID CELL
      READ (LINE,'(2(F5.0,F3.0,F8.2))') (COORD(I),I=1,6)
      COORD(3) = COORD(3) + 0.5
      COORD(6) = COORD(6) + 0.5
      DO 35 I = 1, 6
        ICOORD(I) = COORD(I)
35     CONTINUE
        CALL RNDSDMS (ICOORD(1),ICOORD(2),ICOORD(3))
        CALL RNDSDMS (ICOORD(4),ICOORD(5),ICOORD(6))
        WRITE (LUOUT,'(I5,7X,3I3.2,8X,3I3.2)') IPNT, (ICOORD(I),I=4,6),
      1      (ICOORD(I),I=1,3)
        IPNT = IPNT + 1
        IF (IPNT .GT. IMAX) IPNT = 1
100   CONTINUE
C
C7 -- ALL DONE: CLEAN HOUSE
      CALL ACLOSE (LUIN)
      ENDFILE      (LUOUT)
      CALL ACLOSE (LUOUT)
      CALL AEXIT
      END

      SUBROUTINE RNDSDMS (IDEG,IMIN,ISEC)
      IF (ISEC .GE. 60) THEN
        ISEC = ISEC - 60
        IMIN = IMIN + 1
      ENDIF
      IF (IMIN .GE. 60) THEN
        IMIN = IMIN - 60
        IDEG = IDEG + 1
      ENDIF
      RETURN
      END

```

Attachment O.--Example projection-file listing

```

INPUT
PROJECTION UTM
FORMAT '(20X,2F15.3)'
ZONE 14
UNITS METERS
PARAMETERS
OUTPUT
PROJECTION GEOGRAPHIC
QUADRANT NW
UNITS DMS
PARAMETERS
END

```