

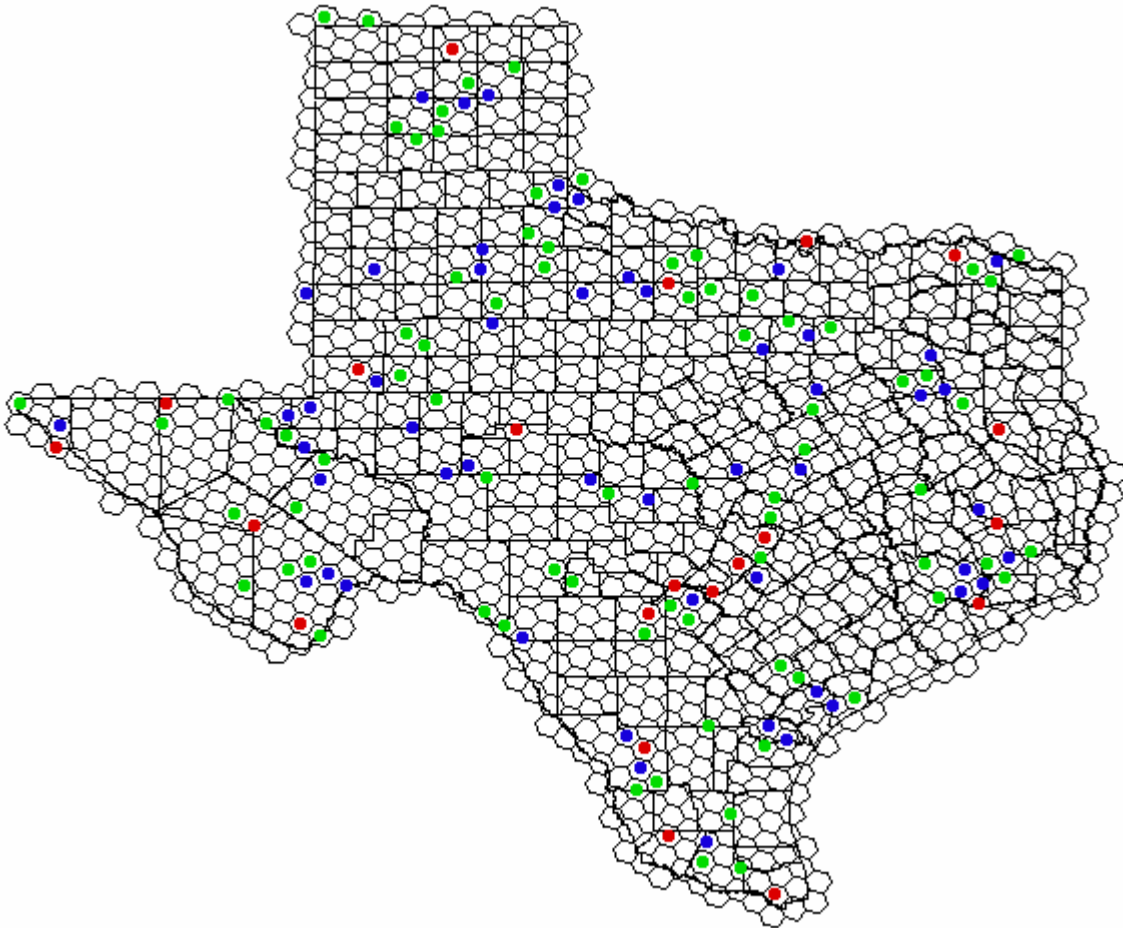
ResNet Manual

Ver 1.2

Available online on 30 October 2002

Last updated on 21 March 2007

J. Garson^{*}, A. Aggarwal^{*}, and S. Sarkar^{*}



^{*} Biodiversity and Biocultural Conservation Laboratory, Section of Integrative Biology, University of Texas at Austin, 1 University Station, C 0930, Austin, TX 78712.

Disclaimer

Although the ResNet 1.2 software package has been tested and run successfully on computer systems at the University of Texas at Austin, no warranty of ResNet 1.2 is expressed or implied.

The software, data, and related materials contained therein are provided "AS IS," without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular task.

ResNet can be downloaded from:

<http://uts.cc.utexas.edu/~consbio/Cons/program.html>.

Cover Graphics: Place prioritization for Texas on a hexagonal grid from Texas-GAP using the modeled distribution of 655 vertebrate species. Red: the cells selected when 5 % of records for each species is targeted for protection; Blue: the additional cells selected when 10 % is set as the target; Green: when 15 % is set as the target.

Biodiversity and Biocultural Conservation Laboratory,
Section of Integrative Biology,
University of Texas at Austin,
1 University Station, C 0930,
Austin, TX 78712 -1180;

<consbio@uts.cc.utexas.edu>;
1 (512) 232 7101.

Contribution No. 2002 -2.

© 2002 J. Garson, A. Aggarwal, and S. Sarkar. Please send comments and questions to <sarkar@mail.utexas.edu>.

Table of Contents

Chapter 1. Introduction.	4
Chapter 2. Algorithms.	9
Chapter 3. The Main Program.	12
Chapter 4. Input File Preparation.	23
Chapter 5. Output Files.	25
References	29
Appendix 1	32

Chapter 1

Introduction

One of the central theoretical tasks of conservation biology is to prioritize places on the basis of their biodiversity value and to devise management strategies to conserve biodiversity in these places. Traditionally, an implicit place prioritization has routinely been performed when places were selected as managed forests, game reserves, national parks, *etc.* However, this process has almost always involved either the use of intuitive judgments of biodiversity value, concern for charismatic or useful species, or even the use of criteria completely extraneous to biodiversity conservation such as scenic value, wilderness quality or, sometimes, mere availability (Pressey *et al.* 1996; Sarkar 1999). From the viewpoint of biodiversity conservation, the last practice in particular leads to *ad hoc* reservation (Pressey 1994). Management practices designed to separate biodiversity from processes that threaten it have also often been based on expert intuition rather than well-tested models.

However, over the last decade an explicit framework for systematic conservation planning has emerged (Margules and Pressey 2000). The ResNet software package implements algorithms that are designed to solve problems that are specified by one part of that framework. Assume that: (i) the biogeographical and other data on which conservation decisions must be made have already been collected for some region; and (ii) explicit conservation goals have been set. Biodiversity planning and management in that region now involves a four-stage process:

(i) using the data set, and keeping the conservation goals in mind, *surrogates* must be selected which will represent whatever it is that is the target of conservation (species, vegetation types, ecosystem types, or other features). This is the “surrogacy problem.” The most common surrogates used are the distributions of some species (usually vertebrates) and environmental parameters (*e. g.*, Nix *et al.* 2000) (average rainfall, average temperature, soil type, aspect, *etc.*) because these are often the only data available (Margules *et al.* 1995). If total species diversity is the conservation goal, whether these surrogates are empirically adequate as representatives of that diversity (in the sense of being good predictors) remains an open and relatively unexplored research question;

(ii) using lists of these surrogates, these places are ordered according to their biodiversity *content* (as opposed to *value*; see [iii] below). This is the “place prioritization problem.” The ResNet software package is designed to solve this problem. This prioritized list provides a basis for the next two stages, which are highly resource-intensive and usually cannot be carried out for all places in a region;

(iii) for each place, the projected futures of the entities of interest (populations, species assemblages, *etc.*) must be estimated. This is the “viability problem.” It is extremely difficult to solve in practice. Viabilities ideally must be estimated for all of the actual

targets of conservation, not just for the surrogates. Knowledge of these viabilities will induce a change in the ranking of places in the prioritized list, for instance, by decreasing the importance of places where viabilities are low and the entities of interest are also found elsewhere. The re-ordered list now reflects the biodiversity *value* of different places. A variety of methods are available for estimating viabilities, for instance, stochastic population viability analysis (PVA) for small populations (Boyce 1992; Burgman *et al.* 1993) and conventional ecological experimentation for larger ones (Caughley and Gunn 1996). Another approach to estimating viability is to predict threats to entire places, for example from conversion to agriculture or forestry (see, e.g. Pressey *et al.* 1996; Cowling *et al.* 1999);

(iv) finally, the problem of devising appropriate management practices for each place can begin, presumably starting with those places with highest biodiversity value. This is the “feasibility problem.” Socio-economic and political factors are typically extremely important at this stage.

While these stages have been listed sequentially, there is feedback from the Stage (iv) to Stage (iii) since management practices can alter viabilities. Furthermore, over time, features such as species may disappear from places, management practices may fail or succeed beyond expectations, and biological systems will evolve in response to environmental and other changes. The goal of “adaptive management” is to take these temporal factors into account and accordingly change management practices and overall conservation policy for a region.

The ResNet software package concerns only Stage 2 of this process. Chapter 2 describes a place prioritization procedure which, in its original form, was developed in the late 1980s. It emphasizes the selection of places containing rare surrogates (the principle of “rarity”) and places which add as many under-represented surrogates as possible to a set of selected places (the principle of “complementarity”). This procedure, which results in a set of related algorithms, is implemented in the ResNet software package. Similar procedures have also been implemented elsewhere (e. g., Margules *et al.* 1988; Vane-Wright *et al.* 1991; Rebelo and Siegfried 1992; Pressey *et al.* 1993). However, ResNet is unique in using *dynamic memory allocation*; thus there is no constraint on the size of the data set.

Recent regional planning applications can be found in Nix *et al.* (2000) and Pressey (1998). The algorithms implemented in ResNet are variations and extensions of one originally proposed by Margules *et al.* (1988) (see also Nicholls and Margules 1993). If a region is divided into a set of places (on the basis of geographical coordinates, ecological boundaries, *etc.*) these algorithms order those places by their biodiversity content. The algorithms implemented in ResNet assume that a definite target has been set in the form of (i) adequate representation of each surrogate, that is, the number of selected places in which that surrogate must be present; (ii) maximum allowed area; or (iii) maximum allowed cost of a proposed set of conserved places. The goal of the algorithms is to achieve the set target efficiently by selecting as few places as possible that together reach the conservation goal (Pressey and Nicholls 1989).

Three principles are incorporated into these algorithms:

(i) *rarity*: first surrogates are ordered inversely by the frequency of their appearance in the data set. Then places are ordered according to whether they contain the rarest surrogate, the next rarest surrogate, and so on, iteratively. (Complete ties are broken by lexical order--see below);

(ii) *complementarity*: places are ordered on the basis of the number of surrogates which have not met the targeted representation (if set) that they contain;

(iii) *richness*: places are ordered on the basis of the number of surrogates present. Richness is potentially used--there are other options--in only one part of the algorithms (the initialization part). This reflects the fact that the use of richness results in inefficient place selection (Williams *et al.* 1996; Csuti *et al.* 1997).

The prioritization consists of two stages with an optional third stage. At Stage 1 there are three choices; at Stage 2 there are two choices; at the optional Stage 3, two further choices. There are thus 18 different algorithms implemented in the ResNet program. Stage 1 is the initialization stage. First, places that cannot be reasonably targeted for conservation measures (for instance, because of a high population density), which are therefore "masked," are removed from the set of potentially selected places. There are now three ways to initialize the prioritization procedure: (i) select the first place by rarity; (ii) select it by richness; or (iii) introduce a set of pre-selected places. In the case of the first two options, ties are broken arbitrarily by selecting the first place on the list (this is called "lexical order"). Thus a unique place is chosen. The third option is the relevant one when a set of reserves is already given and an attempt is being made to build on it systematically. There may be more than one place initially selected through this option.

Stage 2 is the iterative stage. An adjacency constraint can be incorporated at this stage, that is, a place is preferred if one of its neighbors has already been selected. If this option is adopted, larger areas, or groups of areas closer together are more likely to result than if it is not. Given a set of selected places, or an imposed set such as existing reserves, the problem is to find the best new one to add. The method does this by first trying to select the new place using rarity. All places with the rarest under-represented surrogates are identified. If there is only one, it is added to the list. If there is more than one, complementarity is used to try to break ties. If there is still a tie, and if adjacency has been chosen, then adjacency is used to break ties. Final ties are broken by lexical order.

This iteration continues until the target is met, that is, all surrogates are adequately represented, or the maximum allowed area or cost is exceeded. If no explicit target has been set, the procedure continues until all places are selected. The order in which these places are selected produces a ranking of the set of places on the basis of their biodiversity content. Biodiversity content is thus implicitly defined by the algorithm,

and the intuition behind this approach is that diversity is adequately captured by rarity and complementarity. Figure 1.1 shows the flowchart of a typical algorithm. Initialization is by rarity and the adjacency option has not been chosen.

Suppose that an explicit target of surrogate representation has been set, and that a set of places has been selected. This procedure does not guarantee that some of the selected places may not have been made redundant by places selected after them, that is, they may be eliminated without causing the target of representation for any surrogate to fail. The optional Stage 3 checks for redundancy. There is another choice here: all redundant places may be eliminated or only those that are not adjacent to other non-redundant selected places. The second approach, once again, tries to make potentially conserved areas as large as can be justified. First, the algorithms iterate over the list of selected places to find each one that is redundant. Note that while each one is redundant, the entire set of such places need not simultaneously be redundant. Consequently places can be eliminated only one at a time. If there is a unique redundant place, and if adjacency has not been imposed, it is eliminated. If there is no such unique place, the redundant places are ordered by rarity with ties being broken by lexical order. Then, once again after checking for adjacency if required, the first place on the redundant list is eliminated. The entire process is iterated over all the potentially redundant places.

Chapter 3 describes the main program and illustrates how it can be used, step by step. There are two versions, one for Windows (§ 3.1) and one for DOS (§ 3.2). Chapter 4 describes the preparation of the input files in requisite format; Chapter 5 discusses the various output files that may be created. Finally, Appendix 1 contains figures describing a sample run.

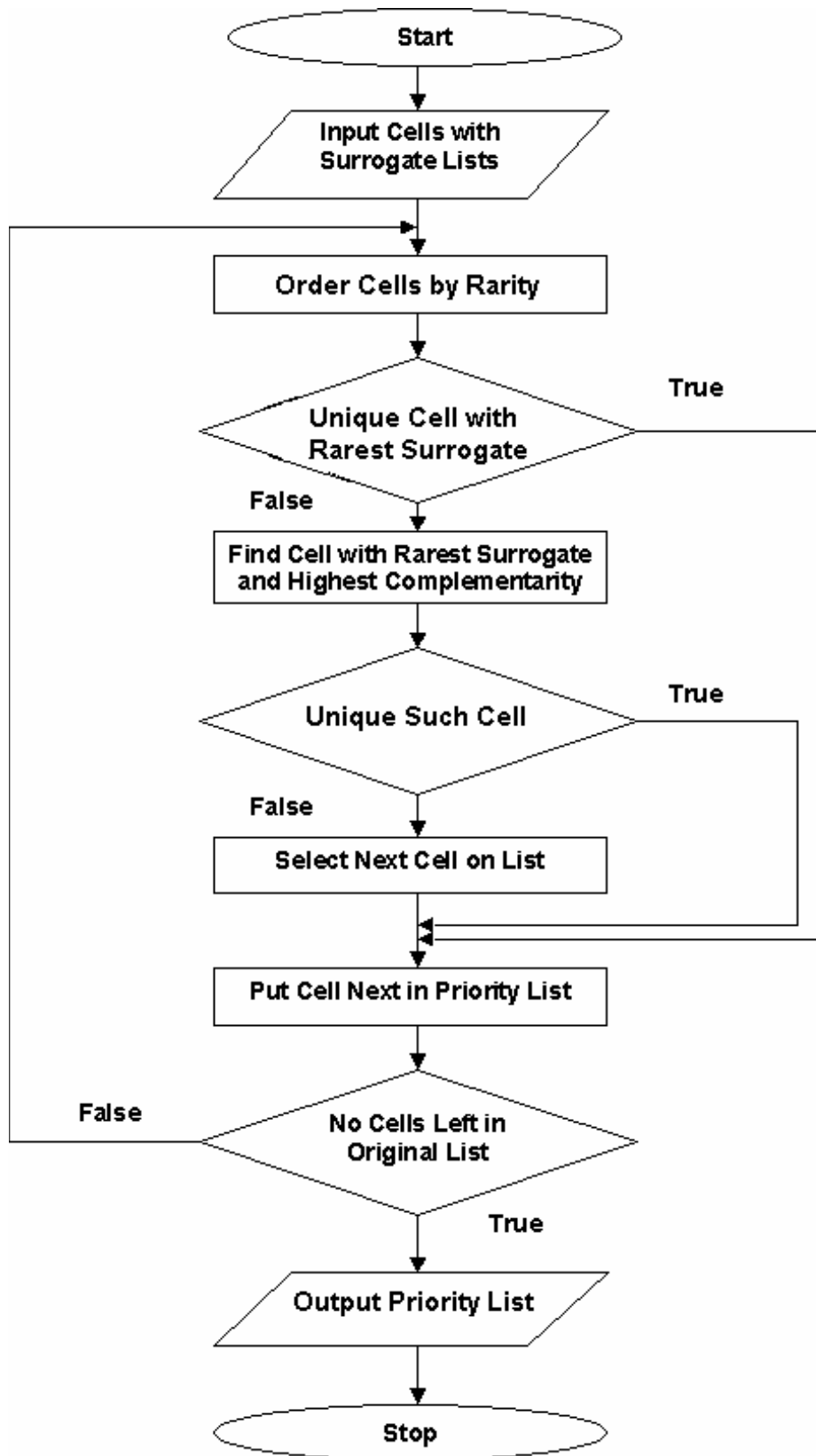


Figure 1.1. Flowchart of the Basic Algorithm

Chapter 2

Algorithms

The purpose of this algorithm is to select cells on the basis of rarity and complementarity until an attribute target is met, or a cost or area target is exceeded, or there are no cells with data left. The algorithm basically consists of two steps, an **initialization** step and an **iteration** step. The initialization step uses rarity; the iteration step uses rarity followed by complementarity to disambiguate potential ties (when the previous steps have chosen more than one cell). The user can also choose (i) to impose an adjacency constraint to disambiguate potential ties after the use of complementarity; and (ii) to remove redundancy, that is, deselect previously selected cells if it is possible to do so without making some attribute to fall below its target or decreasing the number of representations of an attribute that is yet to achieve its target. If redundancy is being removed, the user also has the option to test for adjacency again and not remove a redundant cell if it is adjacent to one that is not redundant. In all steps, if the use of all the relevant rules does not allow disambiguation, lexical ordering is used to break ties as a last resort.

Step 0. Check to see if there is any unmasked cell (that is, a cell which it is possible to select with data. If there is none, exit the program. This is Exit (0).

Step 1. **Initialization.** There are three initialization options:

- (i) start out with an existing set of cells. This is Rule (1-i);
- (ii) start out with the cell with the most number of attributes; this is Rule (1-ii). If there is ambiguity, choose the first cell in the list. This is (Rule 1-ii-l);
- (iii) order the attributes in terms of rarity. Choose the cell with the rarest attribute. If there is no ambiguity, this cell is selected by Rule (1-iii).

There can be ambiguity because more than one cell can contain a rarest attribute. [Note that there can be more than one rarest attribute.] If there is ambiguity, iterate in the following manner over the ambiguous set in the following way:

- (a) choose the rarest attribute *other than the ones previously used to form this set*, and see which cells have it and select that attribute. If there still is ambiguity, repeat the process with a new rarest attribute, and iterate over all attributes. If a unique cell gets selected, this cell is selected by Rule (1-iii-a);
- (b) if, at the end of this iteration over attributes, there is still ambiguity, use lexical order, that is, choose the first cell in the remaining set of cells. This is Rule (1-iii-l).

Step 2. Check to see if: (i) there are unmasked cells with data left; (ii) that the attribute target is not met; (iii) that the area target has not been exceeded; and (iv) that the cost target has not been exceeded. Otherwise, exit the program. These are Exits (2-i), (2-ii), (2-iii) and (2-iv) respectively.

Step 3. Determine which attributes have had their targets met.

Step 4. **Iteration.** Repeat the following process:

(i) Order attributes for which the target has not been met in order of rarity. The use of “rarity” in describing this step (Step 4) is to be interpreted according to the last two sentences¹;

(ii) select the cells with the rarest attribute. If there is a unique such cell, this is Rule (4-ii);

(iii) if there is ambiguity choose the cell that has the next rarest attribute and the most number of attributes for which the target has not been met; this is Rule (4-iii);

(iv) if there is ambiguity, and if the adjacency constraint has been imposed, try to select the unique cell that is adjacent to a cell that has already been selected; this is Rule (4-iv);

(v) if there is ambiguity, select the first cell on the list ; this is Rule (4-i).

(vi) check to see if: (i) there are unmasked cells with data left; (ii) that the attribute target is not met; (iii) that the area target has not been exceeded; and (iv) that the cost target has not been exceeded. If none of these hold, repeat Step 4 (i). If these hold, and the redundancy constraint has not been imposed, exit the program. These are Exits (4-i), (4-ii), (4-iii) and (4-iv) respectively.

Step 5. **Removal of Redundancy.** If the option to remove redundancy has been required, and if attribute targets have not been set, then exit the program. This is Exit (5-o). Otherwise, repeat the following process:

(i) go through the set of remaining selected cells, one by one, and select those cells (if any) the removal of which would not: (a) bring an attribute that has met its target below the target; and (b) would not decrease the representation of an attribute that has not met its target. If there is no such cell, exit the program. This

¹ This means that, if attribute targets have been set, then “rarity” refers to rarity only among those attributes for which the target has not been met.

is Exit (5-i). Otherwise, the potentially redundant cells form the set of redundant cells and the other cells the set of non-redundant cells;

(ii) if there is a unique potentially redundant cell, and if the secondary adjacency constraint has not been imposed, remove this cell and exit the program. This is Exit (5-ii-a). If the secondary adjacency constraint has been imposed, check to see if the cell is adjacent to a selected cell. If so, exit the program. This is Exit (5-ii-b). If not, remove this cell from the set of selected cells and exit the program. This is Exit (5-ii-c);

(iii) if there are several potentially redundant cells, and if the secondary adjacency constraint has been imposed find those cells (if any) that are adjacent to a non-redundant cell. Move these cells to the set of non-redundant cells. Repeat this process until no cells are moved by this process. If there is now no potentially redundant cell left, exit the program. This is Exit (5-iii-a). If there is a unique potentially redundant cell, remove it from the set of selected cells and exit the program. This is Exit (5-iii-b). If there is more than one potentially redundant cell, order them by rarity and, if necessary, lexical order (Step 5 (iv)). If the secondary adjacency constraint has not been imposed, order the potentially redundant cells by rarity and, if necessary, lexical order (Step 5 (iv)). [Note that, in this case, there will be more than one potentially redundant cell because of Step 5 (ii).]

(iv) order the set of remaining potentially redundant cells by rarity. Find the set of cells with the rarest attributes (using the entire data set). If there is no ambiguity, then put the unique cell in the next available rarity rank. If there is ambiguity, iterate over this process always using the next rarest attribute. If there remains ambiguity, use lexical order. When all cells have been ordered, dropped the cell that entered the rarity ranking last and return to Step 5 (i).

Chapter 3

The Main Program

ResNet can be run from Windows (§ 3.1) or from DOS (§ 3.2). Only the former can be downloaded from the web-site. The latter may be requested through e-mail (see Page 2).

3.1. Windows Version.

The Windows version of ResNet is composed of a single software package that contains two components: an interface and a place prioritization algorithm. To run ResNet, click on ResNet.exe. When the interface component has taken all of the necessary data from the user through a series of dialog boxes, it creates a temporary log file (log_file.txt) into which the user's input is written. Then it will automatically execute the algorithm component. A message will appear on the screen when the algorithm has finished execution.

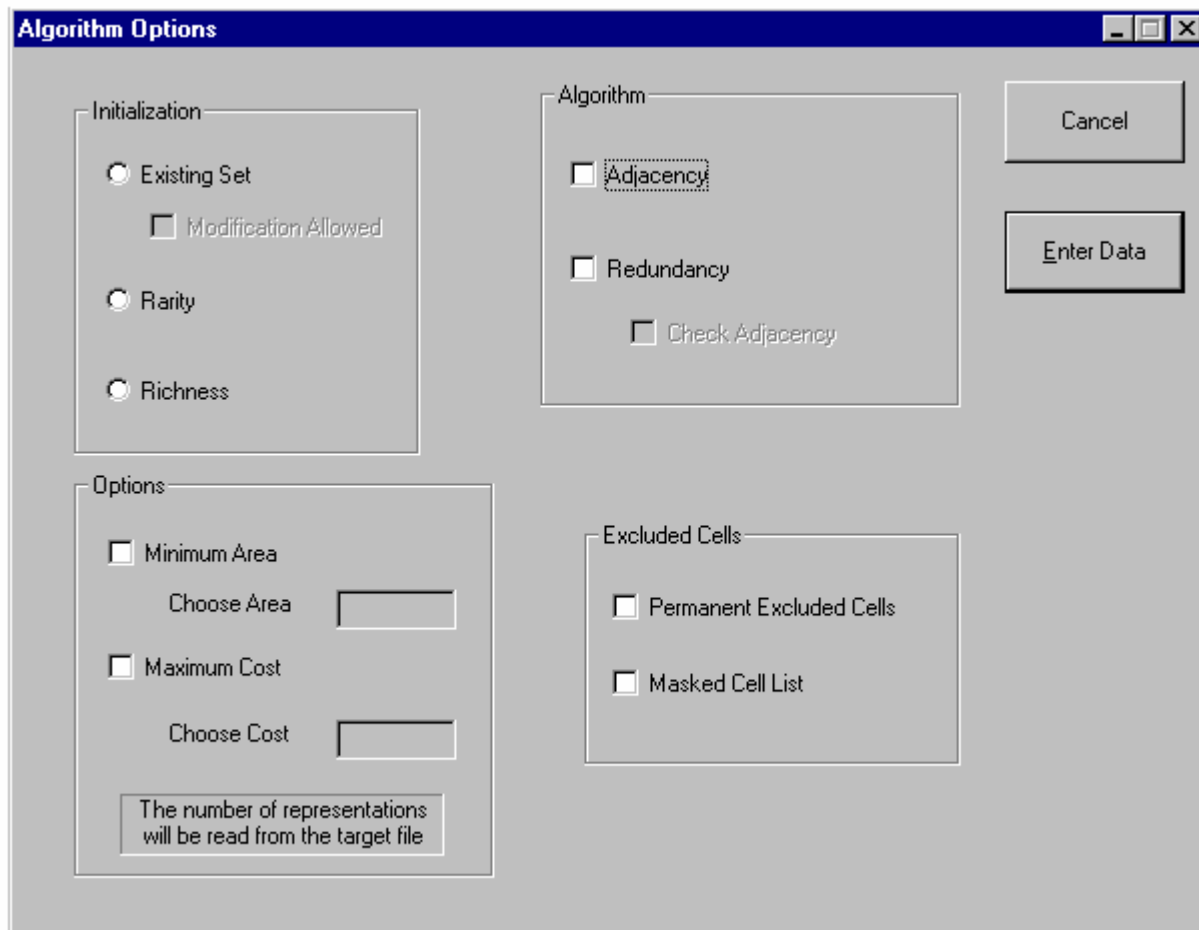


Figure 3.1. Algorithm Options

ResNet.exe may be installed anywhere on the hard drive. To begin the windows version in DOS, run: C://[path_name]//ResNet.exe; in Windows, click on the icon marked "ResNet".

The interface is composed of two main dialog boxes (following the licensing agreement), as well as a number of auxiliary dialog boxes. The first dialog box allows the user to choose the algorithm to be used in cell prioritization, as well as to give constraints on the selection process. The second dialog box prompts the user for the names to the various input and output files that are used by the program. The auxiliary dialog boxes pop up in response to the selection of certain options, and prompt the user for file names that are specific to those options. The two main dialog boxes will be explained in turn.

The first main dialog box is the Algorithm Options box (Figure 3.1). It has four components:

1. Initialization:

The user must choose (only) one of three initialization options. This will determine how the algorithm selects the first cell or set of cells. The default option is rarity.

If “existing set” is selected, an auxiliary dialog box will pop up to prompt the user for the name and location of the input file that contains the existing set of protected cells (Figure 3.2). See Chapter 4 for the format of the existing protected cells set file. (As for all input files, the interface will only allow the user to input a file with “.txt” extension.) The algorithm will employ Rule 1-i for selection (see Chapter 2), “Initialization”. If “richness” is chosen, the algorithm will employ Rule 1-ii. If “rarity” is chosen, the algorithm will employ Rule 1-iii. There is no auxiliary dialog box associated with the last two options.

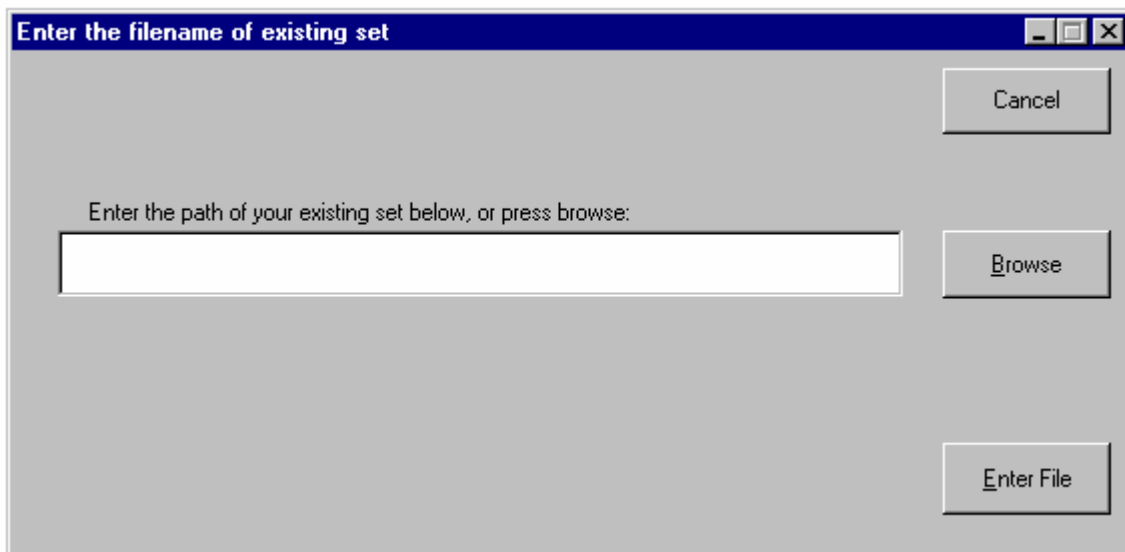


Figure 3.2. The Auxiliary Dialog Box.

2. Iteration:

The “adjacency” and “redundancy” options will condition the manner in which cells will be prioritized after the initial set of cells has been determined (see Chapter 2, Step 4, “Iteration”). The selection of “adjacency” will activate Rule 4-iv in the iteration process. (This will privilege cells for selection that are adjacent to cells already chosen by the algorithm). The selection of “redundancy” will activate Step 5 (if “redundancy” is not checked, the algorithm will terminate at the end of Step 4).

Moreover, the selection of “redundancy” will allow the user the option of a secondary adjacency check (the “Check Adjacency” box will be activated by selection of “Redundancy”). See Chapter 2 (Step 5, iii) for details.

Finally, the selection of any of these three boxes will activate an auxiliary dialog box that will prompt the user for the name of the GIS output file in which the

corresponding results will be written. (If this file does not exist, the program will create it; if it does exist, the program will ensure that it may be overwritten.) Therefore, in addition to the basic GIS output file, which will be created every time the program is executed, the program may also generate up to three additional GIS output files: A GIS output file with the results of using adjacency, a GIS output file with the results of using redundancy, and a GIS output file with the results of using redundancy with a secondary adjacency check.

See Chapter 4 for the format of the various GIS output files.

3. Options:

These options will establish sufficient conditions under which the algorithm will be terminated.

If there is a maximum area that may not be exceeded by the cell selection process, check “Maximum Area”. This will activate the edit box into which the maximum area may be entered. (Note that if “Maximum Area” is de-selected, the edit box will be deactivated, but the number entered by the user will still be visible. However, the program will ignore it unless “Maximum Area” is re-selected.)

Use the same procedure for “Maximum Cost” if there is a maximum cost not to be exceeded.

4. Excluded Cells:

If there is a set of cells that is to be permanently excluded from consideration for selection (e.g., they are in the ocean, or completely built over), check “Permanent Excluded Cells”. An auxiliary dialog box will prompt the user for the file name and location of the set of permanently excluded cells.

The user may also have up to ten separate files for sets of masked cells, that is, cells which are to be strategically excluded from consideration from the selection process. If any such files exist, check “Masked Cell List”. An auxiliary dialog box will prompt the user for the name and location of the first file. If the user has more than one masked cell list, the user may check “More Masked Cell Files?” and press “Enter File”. Another dialog box will appear, and this process begins again. This process will terminate at the end of the tenth cycle, unless the user leaves the “More Masked Cell Files?” box unchecked.

See Chapter 4 for the format of masked cell files.

“Enter Data” may be pressed at any time to open the second dialog box. Moreover, the user retains the option (once having opened the second dialog box) of returning to the first dialog box and making new selections or changing selections already made. The user may also cancel the program at any time; pressing the “Cancel”

button will always activate a message box that will verify the user's intention to terminate the program.

Figure 3.3. Select Input and Output Files

The second necessary dialog box is the Select Input and Output Files box (Figure 3.3). The main purpose of the second dialog box is to gather the names of the four files that *must* be entered in order to proceed with the cell selection process, as well as other information. The files that must be entered are the following (from top to bottom):

1. Input file with cell data / Total number of cells:

Enter the name and location of the input file manually, or press “browse” in order to search the hard drive for the file. See section four for the format of the input file with cell data. Directly underneath the first edit box, enter the total number of cells. The number of cells must be equal to the number of rows in the cell data file.

2. Input file with target and attributes data / Total number of attributes:

Enter the name and location of the target file manually, or press “browse” in order to search the hard drive for the file. See section four for the format of the file with attribute and target data. Directly underneath this control, enter the total number of attributes. This number must be equal to the number of rows in the attribute file. (This will also be equal to the number of columns in the input file – 15).

3. Log file:

The log file must have an “.log” suffix. Enter the name and location of a file, or press “Browse”. See Chapter 5 on the format of the log file that will be generated.

Directly underneath this control, is a box marked “Detailed Log File?” If this box is checked, the log file generated by the program will contain, in addition to the list of selected cells, the order of selection, as well as the rule that was applied in its selection (This option will generate a much longer log file; space constraints should be taken into account).

4. Basic GIS output file:

This file must have a “.txt” suffix. See Chapter 5 on the format of the basic GIS output file that will be generated.

When the “Enter Data” button is pressed, the program will run a series of checks. First, it will ensure that both input files can be opened. Secondly, it will check to ensure that the total number of cells and attributes entered is positive. Finally, it will check to see if the log file, or the basic GIS output file already exist. If one or the other does not exist, it will be created in the folder specified by the user. Otherwise, the computer will ask the user if it may be overwritten.

Once these checks are performed, a dialog box will pop up to inform the user that ResNet is about to begin prioritizing the cells. After pressing “OK”, the place prioritization algorithm will begin, and a second dialog box will pop up to inform the user when it has ended.

The “Back” button will return the user to the previous screen.

3.2. DOS Version.

In DOS, the program and all the input files must be in the same directory. The program is called "ResNet.exe". To run it, type:

>Resnet

[and hit Return].¹ The program will respond:

>Enter the name of the input file with cell data

Type in the name of the input file (which must have a ".txt" suffix) containing the list of cells and the properties of each cell (see Chapter 4 for formats of all input files). If the program does not find the indicated file, it will result in the error message:

>ERROR: unable to open filename to read in cell information

and exit. *The same pattern will be followed for every input file.* Next, the program will prompt:

>Enter the name of the file with attributes and target data

This is the file which has the list of attributes and the target to be met for each. (It must have a ".txt" suffix.) The program will check to make sure that the number of lines in this file is the same as the number of columns containing presence-absence information in the file of the input file with cell data. This will be followed by:

>Enter the name of the log file

The filename must have a ".log" suffix. This file will store the output of this run of the program and will record all important calculations (see Chapter 5 for formats of all output files). The file will ordinarily be stored in the same directory in which the program is run. *The program does not check to see if another file with the same name already exists; if it does, it will be written over and any information in it will be lost. This pattern will be followed for all output files.* This will be followed by:

>Should the log file be detailed? (Y/N)

"Y" means that the log file will contain a list of the selected cells in the order of selection along with the name of the rule that was used to select it; "N" means that this information will not be available. In general the "Y" option will lead to *much* longer log files. If there are significant space constraints, it should not be used.

The program will now prompt:

>Enter the name of the basic GIS output file

¹ "Hit Return" will be implied at each step in this section.

This file must also have a “.txt” format. This is the basic output file that will always be created. It will contain the results of running the algorithm without imposing the adjacency or redundancy constraint. All GIS output files will contain information about the selected cells that can be directly inputted into GIS software packages such as ArcInfo and ArcView for further processing. The program now prompts:

>Enter the total number of cells

This number must be the same as the number of lines in the file with cell data. The program will check for such consistency. If the two parameters are not equal to each other, it will give an error message:

>ERROR: read N_1 lines from cell file, expected N_2 lines

and exit. The next prompt will be:

>Enter the total number of attributes

This number must be the same as the number of lines in the file with the attributes and target data. The program will check for such consistency. If the two parameters are not equal to each other, it will give an error message:

>ERROR: read N_1 lines from attribute file, expected N_2 lines

and exit. The next prompt will be:

>Is there an existing protected cells set (Y/N)?

With the “Y” option, it will prompt:

>Enter the name of the file with the protected cells set

Type in the name of the input file (which must have a “.txt” suffix) containing the list of cells that are already protected. The program will now turn to a specification of the options that the algorithm allows. With the “N” option, the program will turn to other initialization possibilities. It will ask:

>Use rarity for selecting first cell (Y/N)?

With the “Y” option, the first cell to be selected is the one with the rarest attributes. The program will turn to a specification of the options that the algorithm allows. With the “N” option, the program will turn to the final initialization possibility:

>Use richness for selecting the first cell (Y/N)?

With the “Y” option, the first cell to be selected is the one with the most attributes. Exactly one of the last three questions can have a “Y” answer. The program will check for this; if this condition is violated, it will give an error message:

```
>ERROR: Must have either pre-selected set, or use richness or rarity to select starting
>cell
```

and exit.

The program will now turn to a specification of the options that the algorithm allows. It will prompt:

```
>Use adjacency to select cells (Y/N)?
```

“Y” means that, after rarity and complementarity, adjacency will be used to select cells. With the “Y” option, the program will prompt:

```
>Enter the name of the GIS output file with adjacency constraint in use
```

This file must also have a “.txt” format. This is the additional output file that will be created with the results after imposing the adjacency constraint. Thus there are now two GIS output files. With the “N” option, this new file is not created. The program will next prompt:

```
>Use redundancy to reduce the number of selected cells (Y/N)?
```

With “Y”, it will prompt:

```
>Enter the name of the GIS output file after checking for redundancy
```

If the “N” option is chosen, the program will continue to ask about masked cells (see below). If the “Y” option is chosen, the output file just mentioned will always be created. However, the program will then also ask:

```
>Use adjacency to constrain redundancy (Y/N)?
```

With “Y”, it will prompt:

```
>Enter the name of the GIS output file after redundancy and adjacency checks
```

This will lead to the creation of yet another GIS output file. [There are thus potentially four different output files.]

The program will now prompt for the basis for selection. It will first ask:

>Should selection be constrained by targets for attributes (Y/N)?

If the answer is “Y” then it will use the targets in the input file with attributes and target data. If the answer is “N” it will not use these targets. [Internally, the program sets the targets to be artificially high.] The next basis for selection is:

>Should selection be area constrained (Y/N)?

If the answer is “Y” the program will prompt for the area:

>Enter maximum allowed area

The entered parameter must be a real number. The next prompt is:

>Should selection be cost constrained (Y/N)?

If the answer is “Y” the program will prompt for the cost:

>Enter maximum allowed cost

Once again, the entered parameter must be a real number.

Finally, the program will ask for sets of masked cells, that is, cells that will not be available for selection for whatever reason. It will first prompt:

>Is there a file with permanently masked cells (Y/N)?

If the answer is “Y” it will prompt:

>Enter name of file with permanently masked cells

Type in the name of the input file (which must have a “.txt” suffix) containing the list of cells that are permanently masked. Typically these will be cells such as those which are entirely in the ocean or completely built over and so on. All runs of the program will typically use the same set of permanently masked cells. Next the program will allow incorporation of data from as many as 10 files with lists of other masked cells. Typically these will change from run and will allow the exploration of possible solutions when different sets of cells are prevented from being selected. (This is the converse of starting from different sets of previously selected cells.) The prompt will be:

>Is there a file with masked cells (Y/N)?

If the answer is “N” then the program will begin its computations. If the answer is “Y”, it will prompt:

>Enter name of masked cells

In this case, type in the name of the input file (which must have a “.txt” suffix) containing the list of cells that are masked. This routine will be repeated 10 times unless an “N” response is encountered.

Chapter 4

Input File Preparation

The **input file with the cell data** must have the following format:

- (i) the number of lines must be equal to the number of cells that are being analyzed;
- (i) the number of columns must be equal to $15 + n$ where n is the total number of attributes. Each column must consist of the following data:

Column 1: this is the cell identification number and increases sequentially. This must be an integer;

Column 2: this is the longitude of the center of that cell, West is negative, East is positive. This is a real number;

Column 3: this is the latitude of the center of that cell, South is negative, West is positive. This is a real number;

Columns 4 -13: these are reserved for a list of 10 possible neighboring cells identified by their cell numbers (that is, the number in Column 1). This must be an integer; fill these columns with "0" whenever there is no neighbor. If the adjacency constraint is never going to be imposed, these columns may all be filled with "0".

Column 14: this is the area of the cell, a real number. Fill this with "0" if the areas are not available or not relevant for the analysis;

Column 15: this is the cost associated with the cell, a real number. Fill this with "0" if the costs are not available or not relevant for the analysis;

Subsequent columns: the value for each attribute. Currently, this is supposed to be an integer. If the attributes are species, then put "1" for presence and "0" for absence.

The **file with attributes and target data** must have the following format:

- (i) the number of rows is equal to the number of attributes. This number must be equal to the n in the input file with the cell data, that is, the number of attribute columns in that file;
- (ii) there are two columns as follows:

Column 1: this is the attribute identification number and increases sequentially. It must be an integer;

Column 2: this is the target for that attribute. Currently it must be an integer. When a cell is selected, this number is added to the current value for that attribute in the set of selected cells.

The **existing protected cells set** file has the following format:

(i) there is a header column. It does not matter what the content of this column is, because it will be ignored by the place prioritization algorithm;

(ii) the number of rows is equal to the number of protected cells;

(iii) there are 3 columns separated by commas:

Column 1: this is the cell identification number of the protected cell (an integer).

Column 2: this is the longitude of the center of that cell, West is negative, East is positive (a real number);

Column 3: this is the latitude of the center of that cell, South is negative, North is positive (a real number).

This is the same format as that of the GIS output file (see Chapter 5). Thus the results of one run of the program can be used to initiate another run.

Each file with **masked cell data** must have the following format:

(i) the number of rows is equal to the number of cells;

(ii) there is one column which contains the cell identification number of the masked cell (an integer).

Chapter 5

Output Files

There are two types of output file generated by each run of the program, the **log files** and the **GIS output files**. The log file can be **detailed** or not.

The **log file** contains a record of all the most relevant information during a run of the program. All log files will contain the following information about the input to the program for each run:

Input cell file	Filename
Attribute target file	Filename
Log file	Filename
Log file format	Detailed/Not Detailed
GIS output file	Filename
Number of cells	Number of cells
Number of attributes	Number of attributes
Pre-existing set	Yes/No
Pre-existing set file	NA/ Filename of Existing Protected Cells File
Permanently masked cells	Yes/No
Permanent mask file	Filename/NA
Other masked cells	Yes/No
Masked cell file (1)	Filename/NA
Masked cell file (2)	Filename/NA
Masked cell file (3)	Filename/NA
Masked cell file (4)	Filename/NA
Masked cell file (5)	Filename/NA
Masked cell file (6)	Filename/NA

Masked cell file (7)	Filename/NA
Masked cell file (8)	Filename/NA
Masked cell file (9)	Filename/NA
Masked cell file (10)	Filename/NA
Rarity Initialization	Yes/No
Richness Initialization	Yes/No

Note that exactly one of the “pre-existing set”, “Rarity”, and “Richness” can have a “Yes” answer in any one run.

Adjacency constraint	Yes/No
Adjacency File	Filename/NA
Redundancy constraint	Yes/No
Redundancy File	Filename/NA
Redundancy, further Adjacency	Yes/No
Redundancy, Adjacency File	Filename/NA
Attribute target constraint	Yes/No
Area constraint	Yes/No
Area target	Total area/NA
Cost constraint	Yes/No
Cost target	Target cost/NA

There can be a maximum of four such GIS output files, as explained in Chapter 3. The complete information for each (detailed or not) will be recorded here before proceeding to the next one (if there is a next one) in the same order as in which their names were entered when running the program. The pieces of information that will always be recorded are:

--Total number of cells	Total cell number
--Total area of cells	Total cell area

--Total cost of cells	Total cell cost
--Number of permanently masked cells	Permanently masked cell number
--Area of permanently masked cells	Permanently masked cell area
--Cost of permanently masked cells	Permanently masked cell cost
--Number of relevant cells	Relevant cell number
--Area of relevant cells	Relevant cell area
--Cost of relevant cells	Relevant cell cost
--Number of other masked cells	Other masked cell number
--Area of other masked cells	Other masked cell area
--Cost of other masked cells	Other masked cell cost
-- Number of unmasked cells without data	Unmasked cells without data number
-- Area of unmasked cells without data	Unmasked cells without data area
-- Cost of unmasked cells without data	Unmasked cells without data cost

For consistency, the following relations should hold: the relevant cell number = the total cell number - (the permanently masked cell number + the other masked cell number); the relevant cell area = the total cell area - (the permanently masked cell area + the other masked cell area); the relevant cell cost = the total cell cost - (the permanently masked cell cost + the other masked cell cost).

-- Total number of cells selected	Selected cell number
-- Total area of selected cells	Selected cell area
-- Total cost of selected cells	Selected cell cost
-- Reason for stopping	Exit reason

Every log file will also contain the following information for each attribute n .

-- Total attribute representation in selected cells--	
-- Attribute n representation	Number of representations in output
-- Total attribute representation in data set --	

-- Attribute n total presence

Total number of representations

Detailed log files will also contain the following output information for each cell selected.

Cell identification number

Total number of cells currently selected

Percentage of total attributes currently selected

Number of attributes for which target has been met

Percentage of attributes for which target has been met

The **GIS output files all** have the following format:

- (i) there is a header for each column;
- (ii) the number of rows is equal to the number of selected cells (excluding the header);
- (iii) there are 3 columns:

Column 1: this is the cell identification number of the selected cell (an integer).

Column 2: this is the longitude of the center of that cell, West is negative, East is positive (a real number);

Column 3: this is the latitude of the center of that cell, South is negative, North is positive (a real number).

This is the same format as that of the existing protected cells file (see Chapter 4). Thus the results of one run of the program can be used to initiate another run.

References

- Boyce, M. S. 1992. "Population Viability Analysis." *Annual Review of Ecology and Systematics* **23**: 481 -506.
- Burgman, M., Ferson, S. and Akçakaya, H. R. 1993. *Risk Assessment in Conservation Biology*. New York: Chapman and Hall.
- Caughley, G. and Gunn, A. 1996. *Conservation Biology in Theory and Practice*. Boston: Blackwell Science.
- Cowling, R. M., Pressey, R. L., Lombard, A. T., Desmet, P. G. and Ellis, A. G. 1999. "From Representation to Persistence: Requirements for a Sustainable Reserve System in the Species-Rich Mediterranean-Climate Deserts of Southern Africa." *Diversity and Distributions* **5**: 51 -71.
- Csuti, B., Polasky, S., Williams, P. H., Pressey, R. L., Camm, J. D., Kershaw, M., Kiester, A. R., Downs, B., Hamilton, R., Huso, M., and Sahr, K. 1997. "A Comparison of Reserve Selection Algorithms Using Data on Terrestrial Vertebrates in Oregon." *Biological Conservation* **80**: 83 -97.
- Margules, C. R., Nicholls, A. O., and Pressey, R. L. 1988. "Selecting Networks of Reserves to Maximize Biological Diversity." *Biological Conservation* **43**: 63 -76.
- Margules, C. R. and Pressey, R. L. 2000. "Systematic Conservation Planning." *Nature* **405**: 242 -253.
- Margules, C. R., Redhead, T. D., Faith, D. P. and Hutchinson, M. F. 1995. *Guidelines for using the BioRap Methodology and Tools*. Canberra: CSIRO.
- Nicholls, A. O. and Margules, C. R. 1993. "An Upgraded Reserve Selection Algorithm." *Biological Conservation* **64**: 165-169.
- Nix, H. A., Faith, D. P., Hutchinson, M. F., Margules, C. R., West, J., Allison, A., Kesteven, J. L., Natera, G., Slater, W., Stein, J. L. and Walker, P. 2000. *The BioRap Toolbox: A National Study of Biodiversity Assessment and Planning for Papua New Guinea*. Canberra: Centre for Resource and Environmental Studies, Australian National University.
- Pressey, R. L. 1994. "Ad Hoc Reservations: Forward of Backward Steps in Developing Representative Reserve Systems." *Conservation Biology* **8**: 662 -668.

- Pressey, R. L. 1998. "Algorithms, Politics and Timber: An Example of the Role of Science in a Public, Political Negotiation Process over New Conservation Areas in Production Forests. In Eds. R.T. Wills, R.J. Hobbs, R. J. and Fox, M. D. Eds *Ecology for Everyone: Communicating Ecology to Scientists, the Public and Politicians*. Sydney: Surrey Beatty and Sons, pp. 73-87.
- Pressey, R. L., Ferrier, S., Hager, T. C., Woods, C. A., Tully, S. L., Weinman, K. M. 1996. "How Well Protected Are the Forests of North-Eastern New South Wales? - Analyses of Forest Environments in Relation to Tenure, Formal Protection Measures and Vulnerability to Clearing." *Forest Ecology and Management* **85**: 311 -333.
- Pressey, R. L., Humphries, C. J., Margules, C. R., Vane-Wright, R. I. and Williams, P. H. 1993. "Beyond Opportunism: Key Principles for Systematic Reserve Selection." *Trends in Ecology and Evolution* **8**: 124 -128.
- Pressey, R. L. and Nicholls, A. O. 1989. "Efficiency in Conservation Evaluation: Scoring versus Iterative Approaches." *Biological Conservation* **50**: 199 -218.
- Pressey, R. L., Possingham, H. P. and Margules, C. R. 1996. "Optimality in Reserve Selection Algorithms: When Does It Matter and how much?" *Biological Conservation* **76**: 259 -267.
- Rebelo, A. G. and Siegfried, W. R. 1992. "Where Should Nature Reserves Be Located in the Cape Floristic Region, South Africa? Models for the Spatial Configuration of a Reserve Network Aimed at Maximising the Protection of Floral Diversity." *Conservation Biology* **6**: 243 -252.
- Sarkar, S. 1999. "Wilderness Preservation and Biodiversity Conservation--Keeping Divergent Goals Distinct." *BioScience* **49**: 405 -412.
- Williams, P., Gibbons, D., Margules, C., Rebelo, A., Humphries, C., and Pressey, R. 1996. "A Comparison of Richness Hotspots, Rarity Hotspots, and Complementary Areas for Conserving Diversity of British Birds." *Conservation Biology* **10**: 155 -174.
- Vane-Wright, R. I., Humphries, C. J. and Williams, P. H. 1991. What to Protect?-- Systematics and the Agony of Choice. *Biological Conservation* **55**: 235-254.
- Zeidler, J., 1997. "Distribution of Termites (Isoptera) throughout Namibia-- Environmental Connections." MSc Thesis, University of the Witwatersrand, Johannesburg, South Africa.

Zeidler, J. In preparation. "Termite (Isoptera) Spatial Distributions throughout Namibia-- Using a GIS to Map Historic national Distribution Data for Biodiversity, Land-Use and Conservation Planning."

Appendix 1

Sample ResNet Run

The sample ResNet run will utilize termite distribution data for Namibia. The input data consists of presence-only records for 33 genera of termites. The map of Namibia is divided into 1250 cells; each cell is 1° longitude \times 1° latitude. Figure A1.1 is an ArcView image that shows all cells for which there is a record of at least one termite genus.

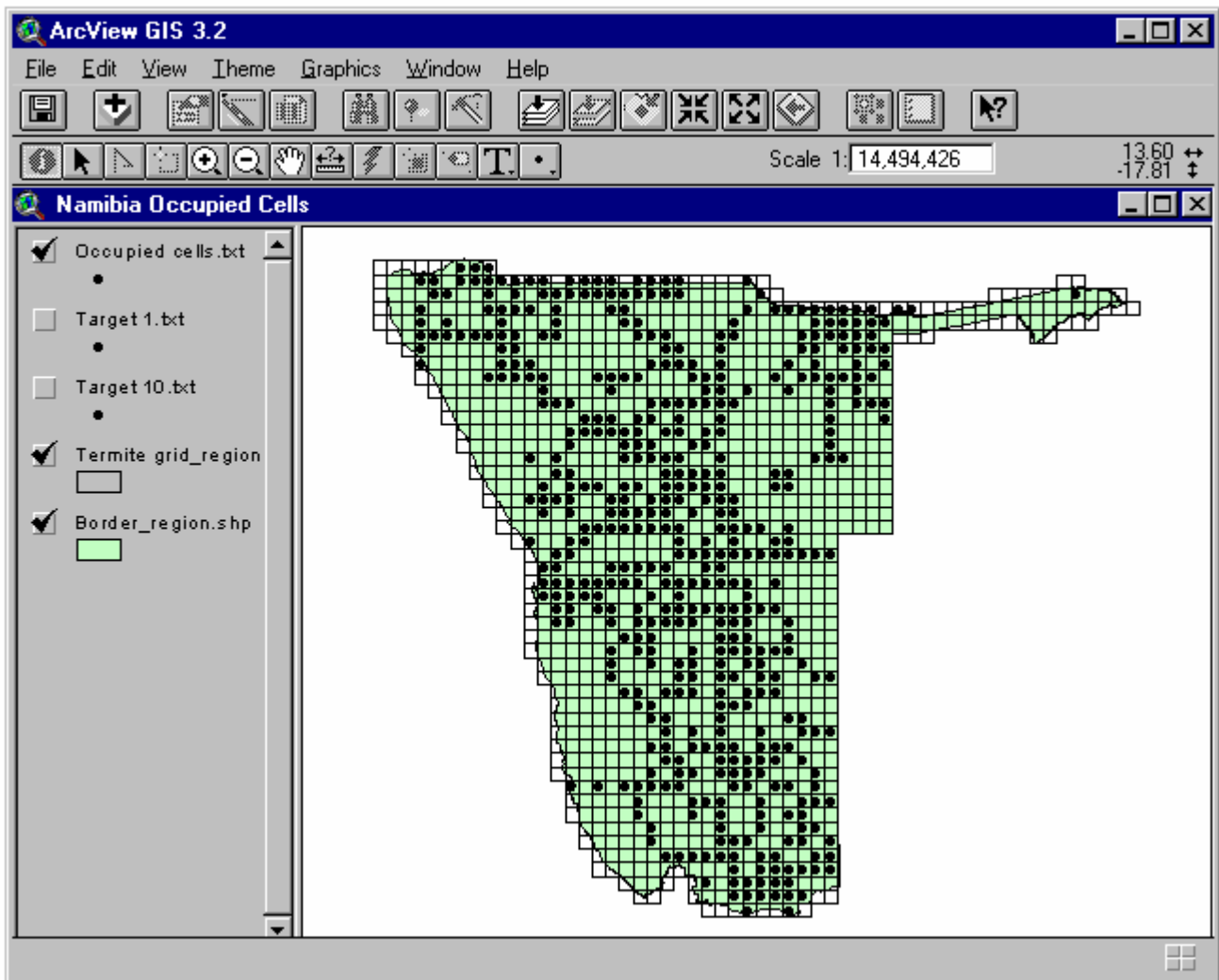


Figure A1.1. Cells in Namibia with Termite Records

The following two ArcView images display the cells that ResNet selects on the basis of the input and target data alone. The first (Figure A1.2) shows the cells selected by ResNet when at least one record is targeted for each termite genus.

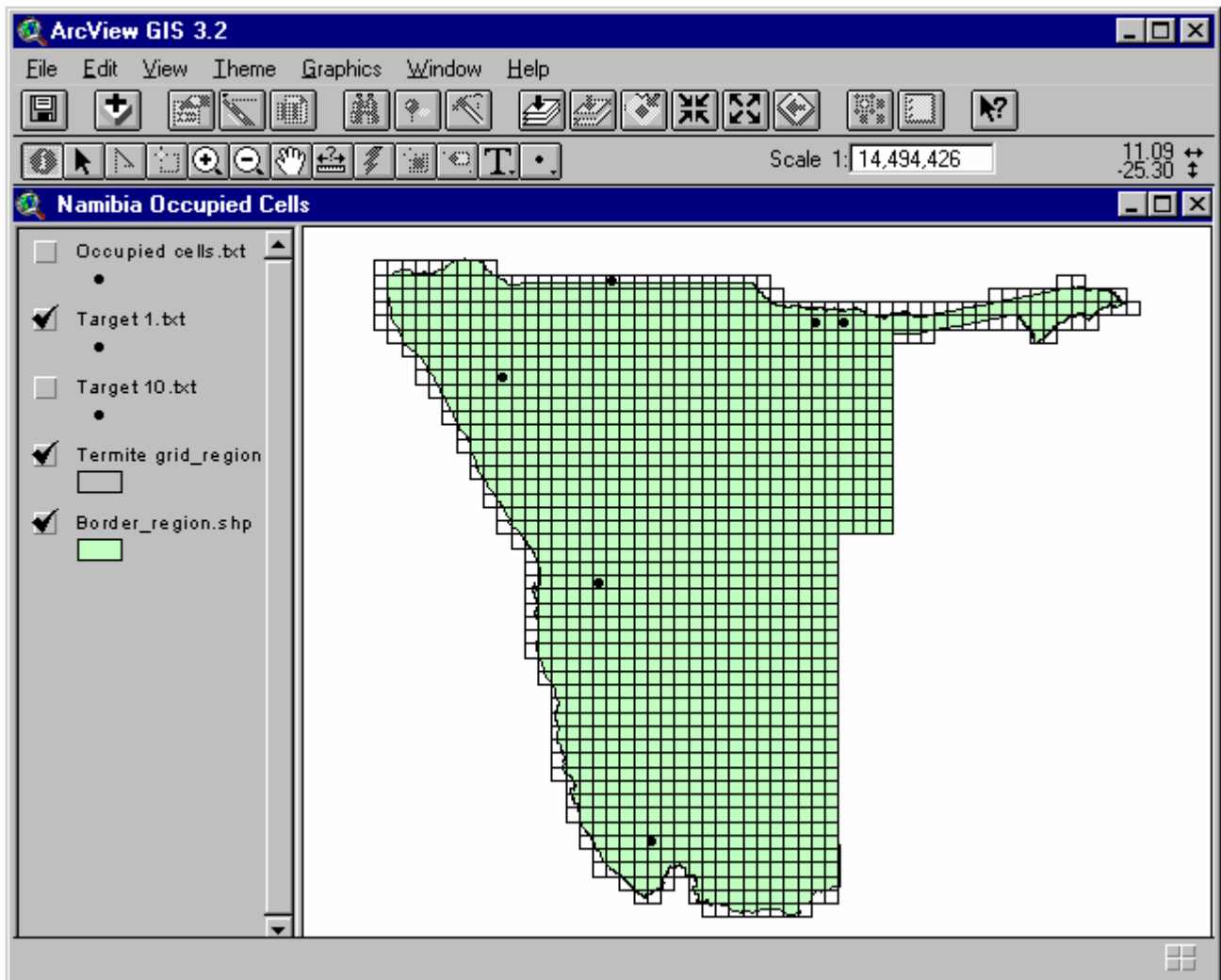


Figure A1.2. ResNet Output (in ArcView) with Target of 1

The second (Figure A1.3) shows the cells selected by ResNet when at least ten records are targeted for each termite genus.

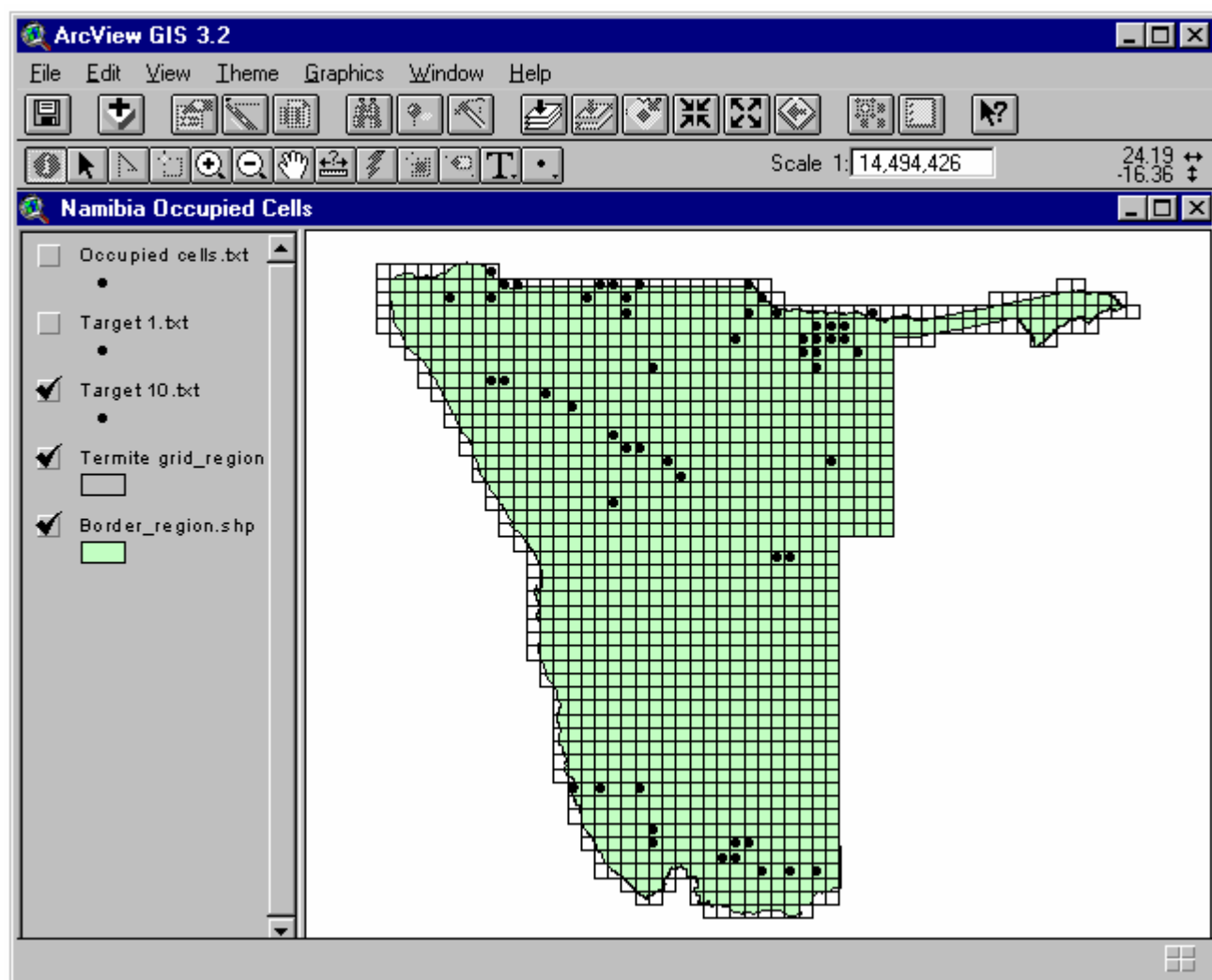


Figure A1.3. ResNet Output (in ArcView) with Target of 10

Below are samples of the various components that enter into a single run of ResNet. There are only two input files that are *necessary* for running ResNet: (i) the input file (Figure A1.4); and (ii) the target file (Figure A1.5).

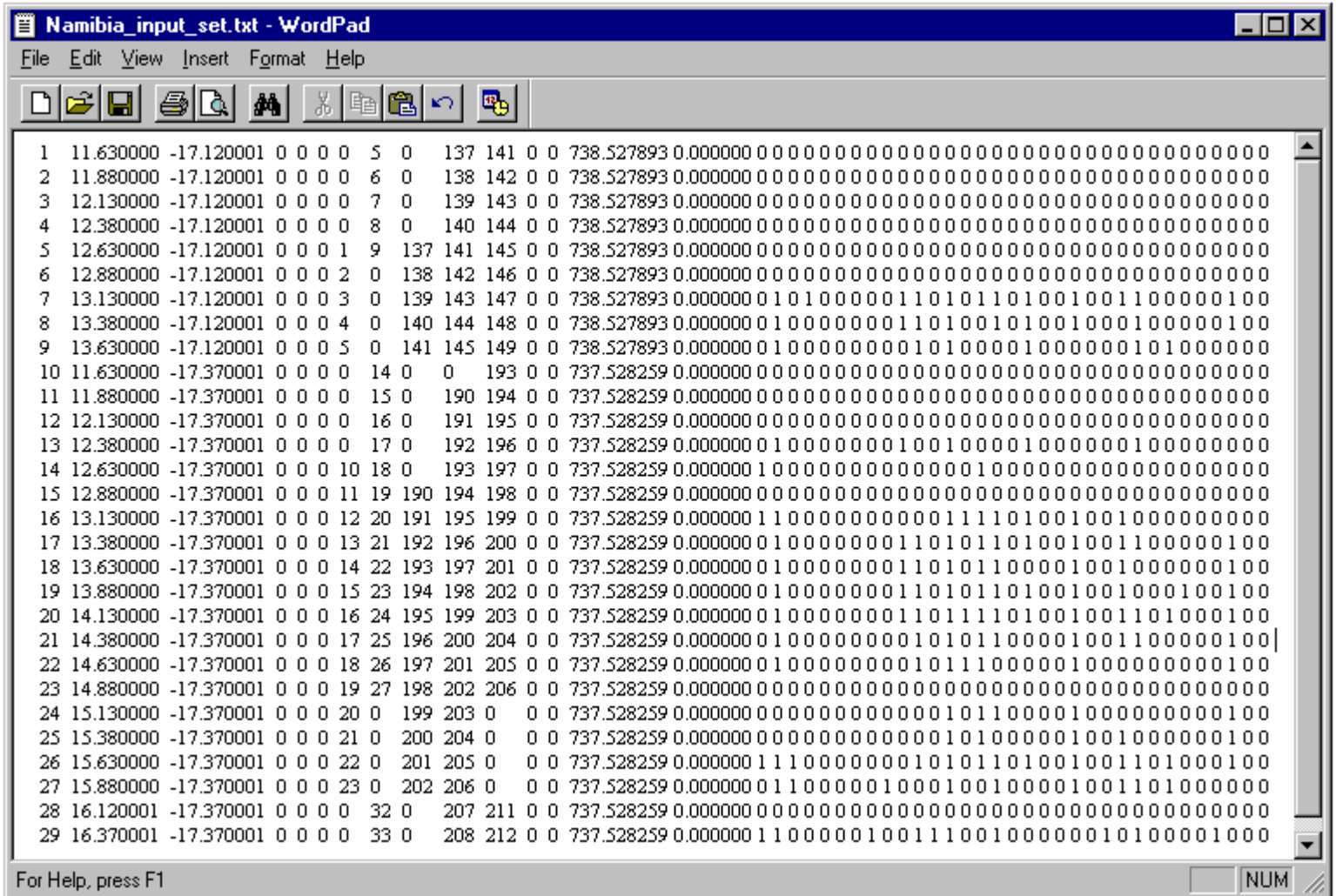


Figure A1.4. The Input File. Not all columns and rows are shown. Interpretation of columns: (i) cell number; (ii) x-coordinate (longitude); (iii) y-coordinate (latitude); (iv) – (xiii) cell numbers for adjacent cells; (xiv) area of cell (in sq. km.); (xv) cost of cells; (xvi) – (xxviii) presence/absence data for 33 genera, 1 indicates presence, 0 indicates absence. There is a corresponding row for each of the 1250 cells.

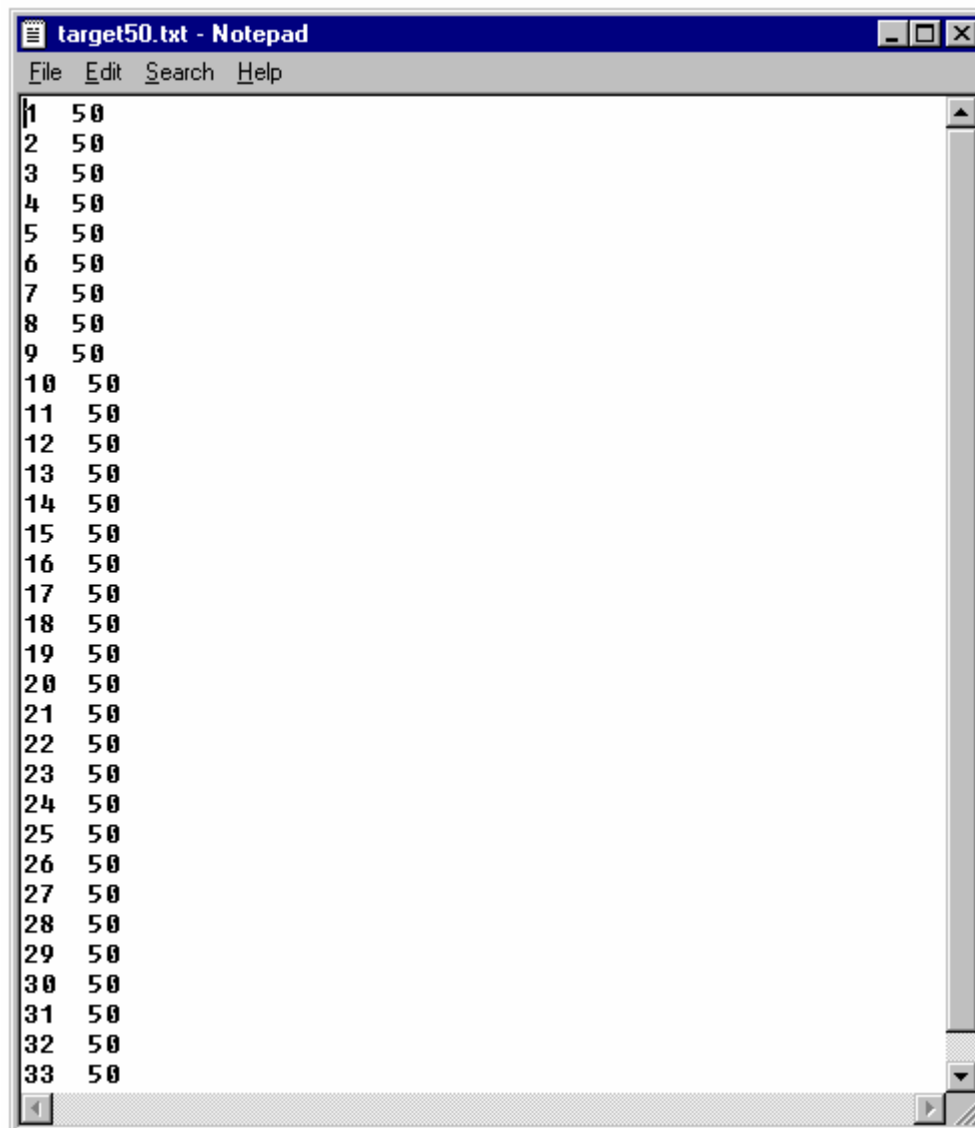


Figure A1.5. Target file. Columns: i) attribute number; ii) desired level of representation for every attribute (the target can also be set differently for each attribute. In this case the target is set at 50.

This is the target set for the 33 species. However, in the following example ResNet will also be set to stop selecting cells when it reaches a maximum area. In this manner, ResNet will stop either when it achieves the level of representation set above (Figure A1.5), or when it reaches the maximum area allowed. Note, however, that even if ResNet is set to stop upon reaching a maximum area or maximum cost, a target file must still be provided. Of course, if there is no particular desired level of representation, then targets may be set arbitrarily high such that the target data does not interfere with the run.

Below is the first dialog box that is encountered (following the licensing agreement), in which ResNet is set to stop selecting cells when an area equivalent to 14% of the total area of Namibia has been selected (Figure A1.6). This run will be initialized by rarity:

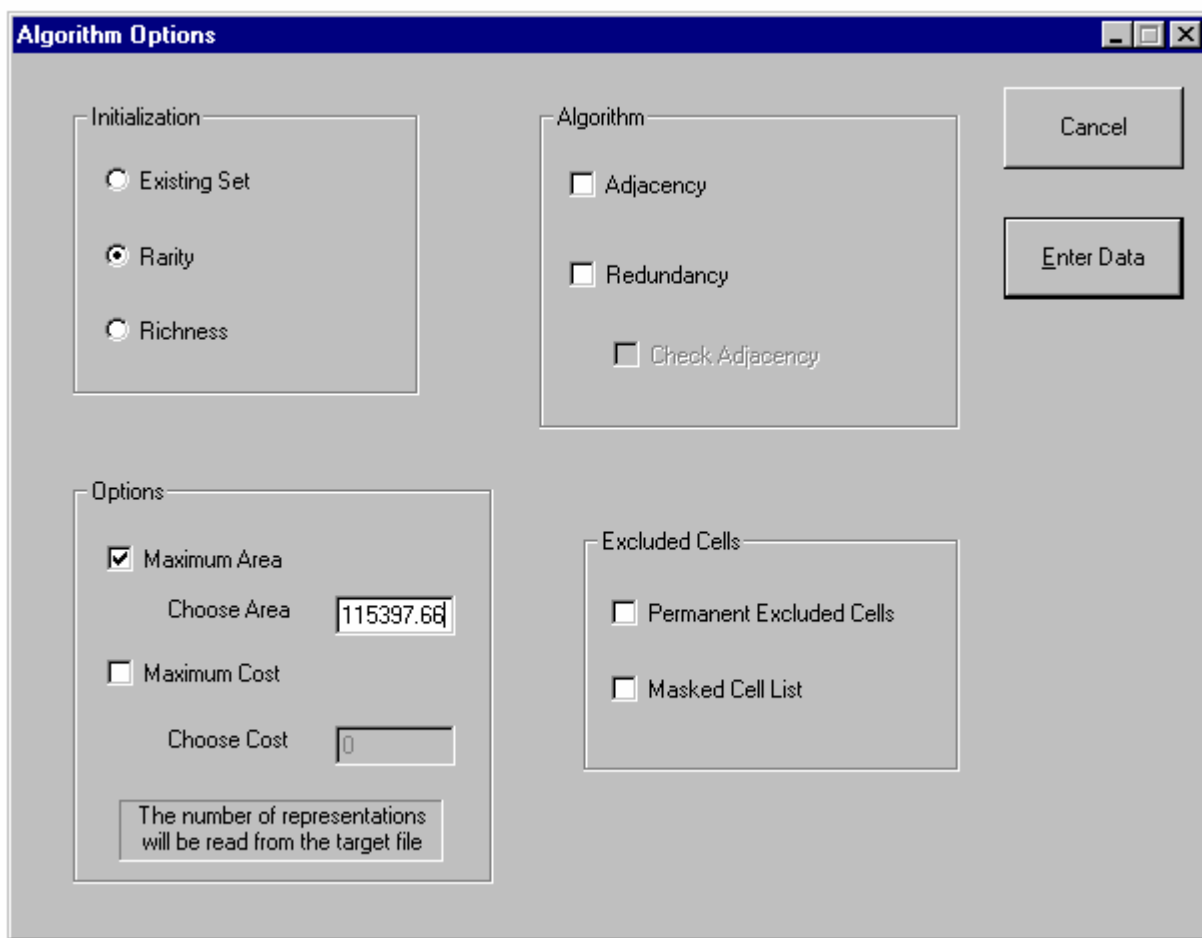


Figure A1.6. Setting ResNet to select 14% of the Total Area of Namibia

Suppose that cells 3, 24, and 625 happen to be heavily built over, and therefore that ResNet should not select these cells. In this situation we create a masked cell file with three entries, of the form (Figure A1.7):



Figure A1.7. Format of Masked Cell File

By checking the “Masked Cell List” box, an auxiliary dialog box appears in which the filename of the masked cell file is inserted (Figure A1.8). Since there is only one such file for this run, the “More Files of Masked Cells?” box is left unchecked.

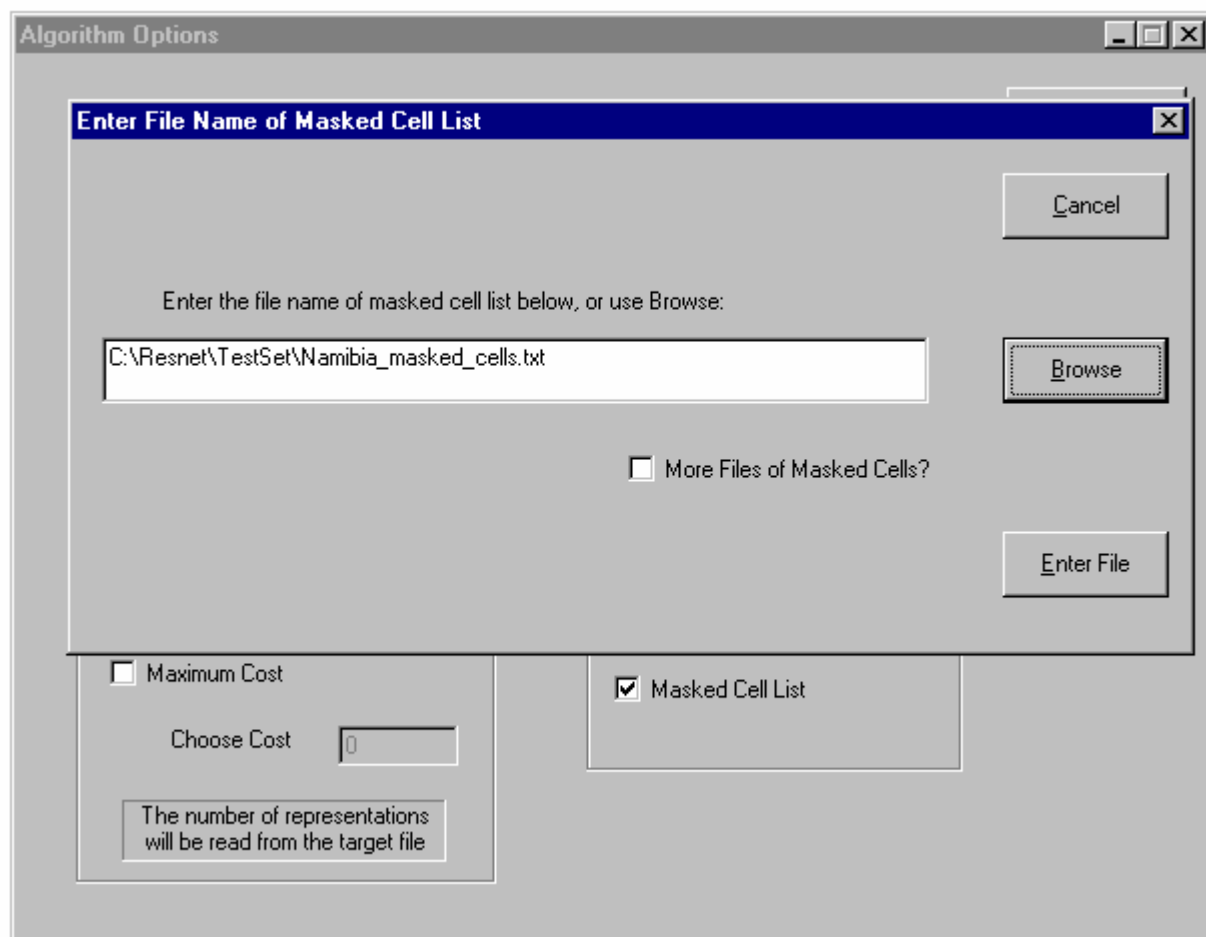


Figure A1.8. Filename of Masked Cell file entered into ResNet

Finally, redundancy will also be used for the sample run. Checking the “Redundancy” box allows the user to create a file for the GIS output file with redundancy (Figure A1.9):

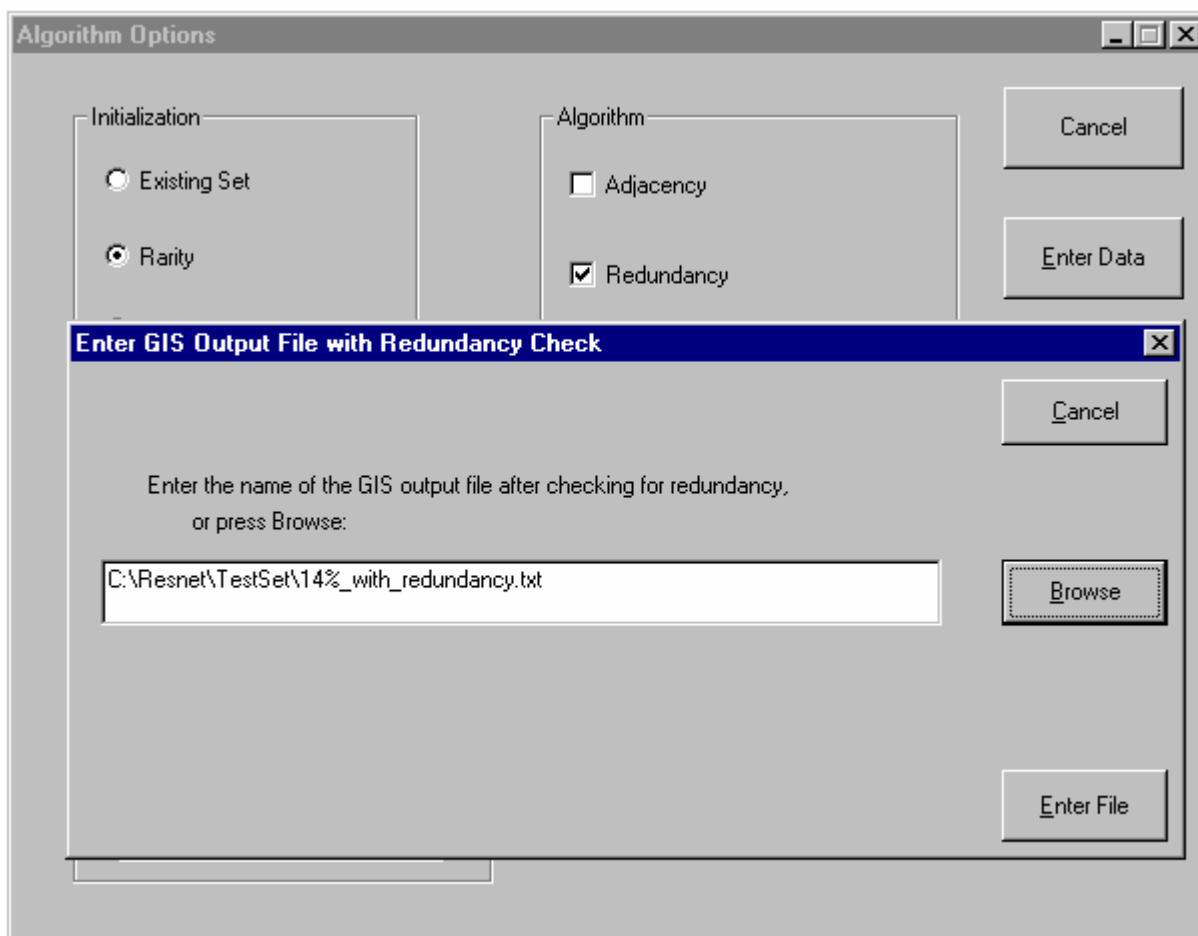


Figure A1.9. Entering the filename of the GIS output file with results of redundancy

Since the “Adjacency” option, “Permanent masked cells” option, and “Existing set” option will not be used for this run, “Enter Data” is pressed and the second dialog box opens.

On the second dialog box, all of the following information must be added, beginning with the filename of the input file. For each filename that must be entered, there is a “Browse” button that can be used (Figure A1.10):

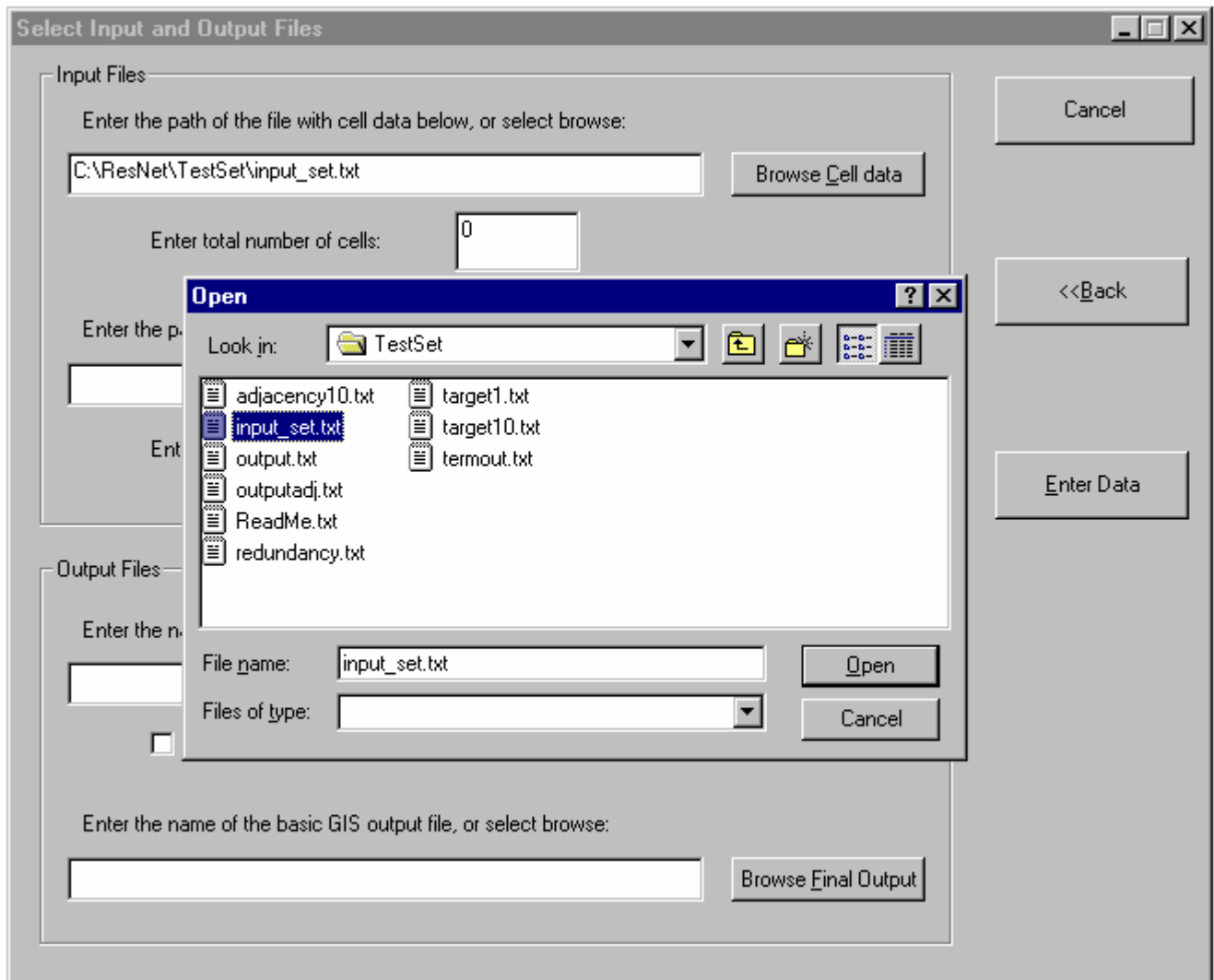


Figure A1.10. “Browse Cell Data” can be used to select input set

Once the remaining input is entered (Figure A1.11), the “Enter Data” button is pushed and the actual algorithm is executed.

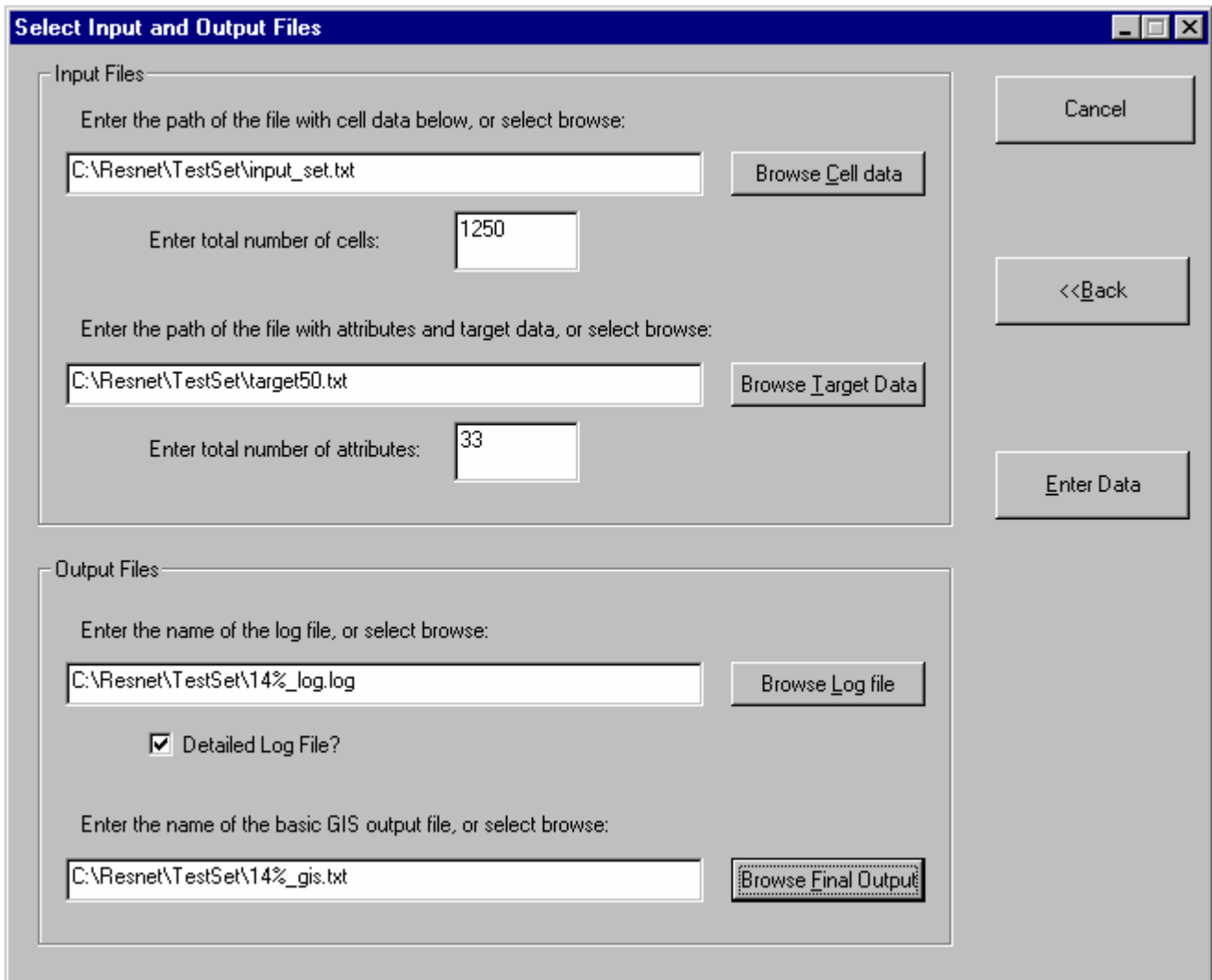


Figure A1.11. The Algorithm is ready to be executed (press “Enter Data”)

When “Enter Data” is pressed, a dialog box will appear to inform the user that the prioritization is about to begin; after the user presses “OK” it will start. Eventually another dialog box will appear to tell the user where the log file can be found (Figure A1.12):

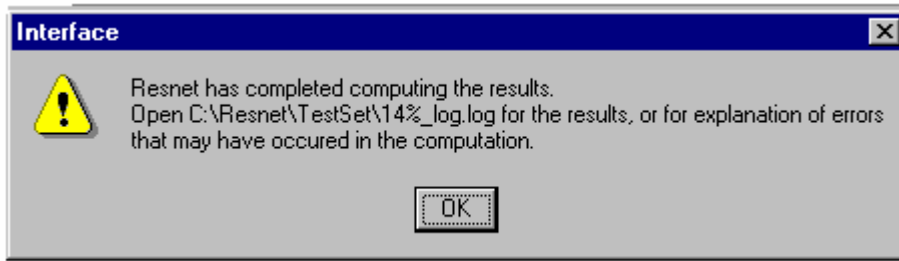


Figure A1.12. The algorithm has finished computing

The format of the GIS output file (“14%_GIS_out.txt”) as well as the GIS output file with adjacency (“14%_with_redundancy.txt”) is as follows (Figure A1.13). (For the GIS output file without redundancy, ResNet selected 161 cells for a total area of 115661.43 square kilometers. The redundancy calculation found one cell to be redundant, selecting 160 cells):

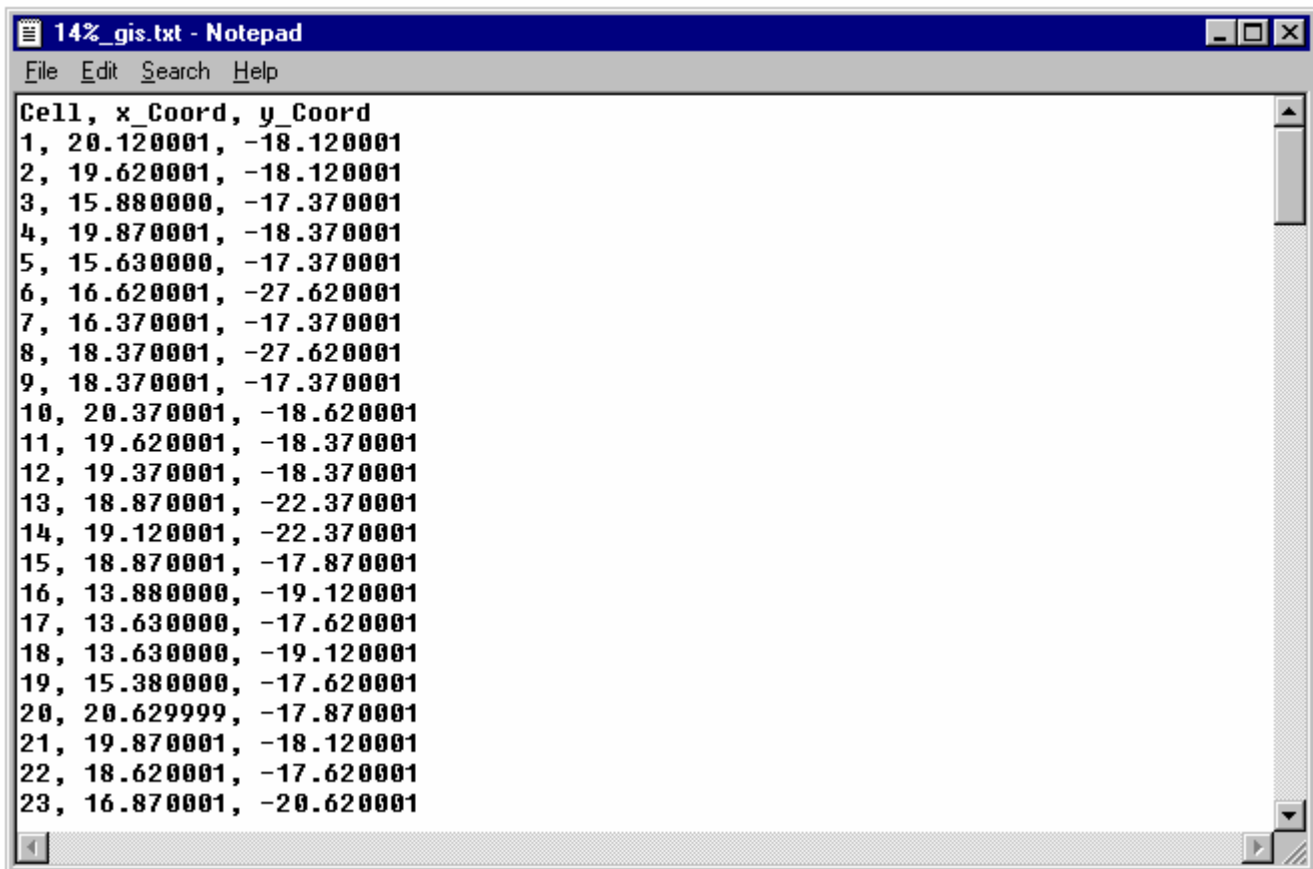


Figure A1.13. Format of GIS output file (without redundancy). The format allows output files to be directly entered into ArcView for projection.

Both of the GIS output files are projected into ArcView. The redundant cell (the black circle with white in the center) is in the upper left-hand corner:

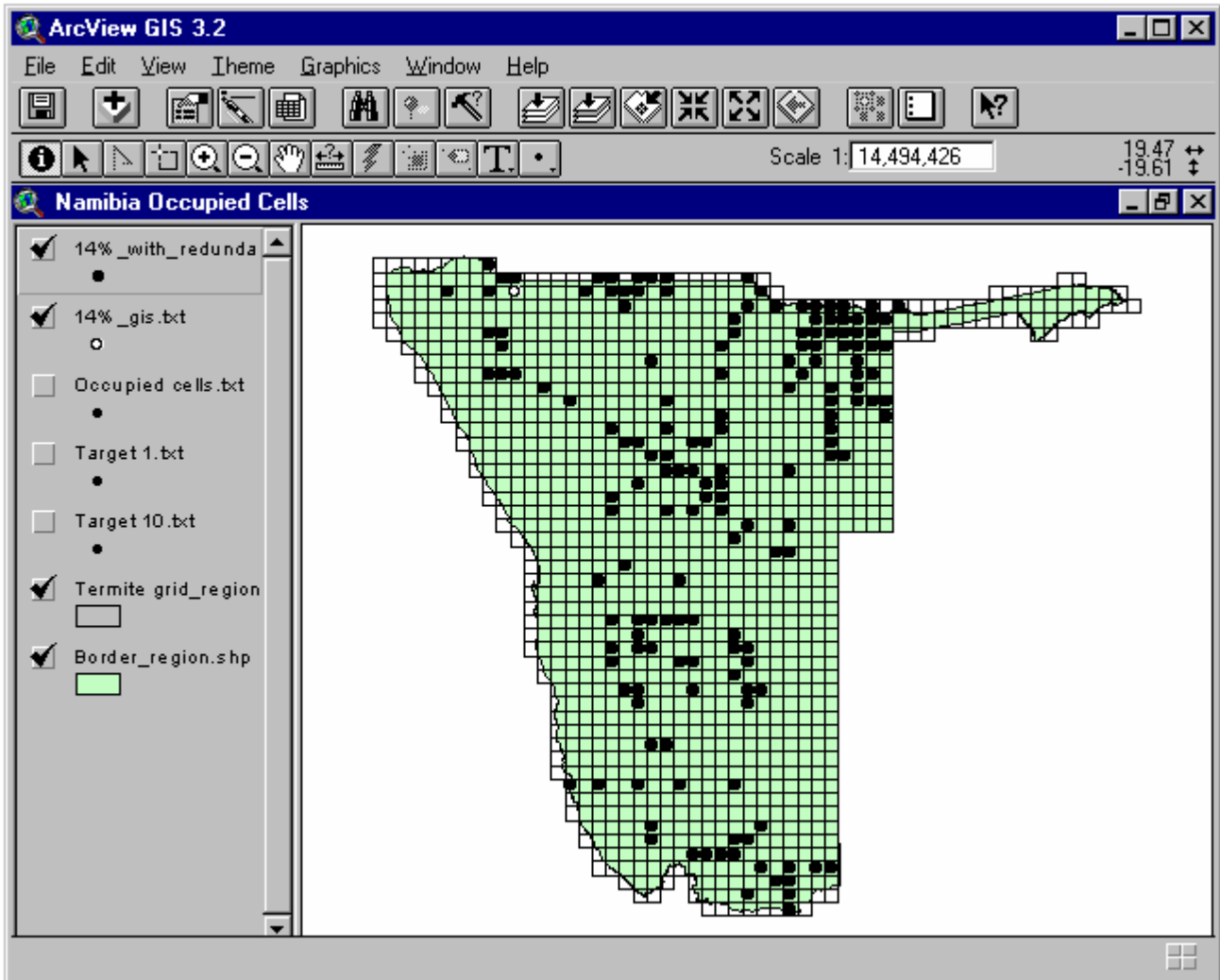


Figure A1.14. GIS output for both redundancy and non-redundancy solutions are projected into ArcView.