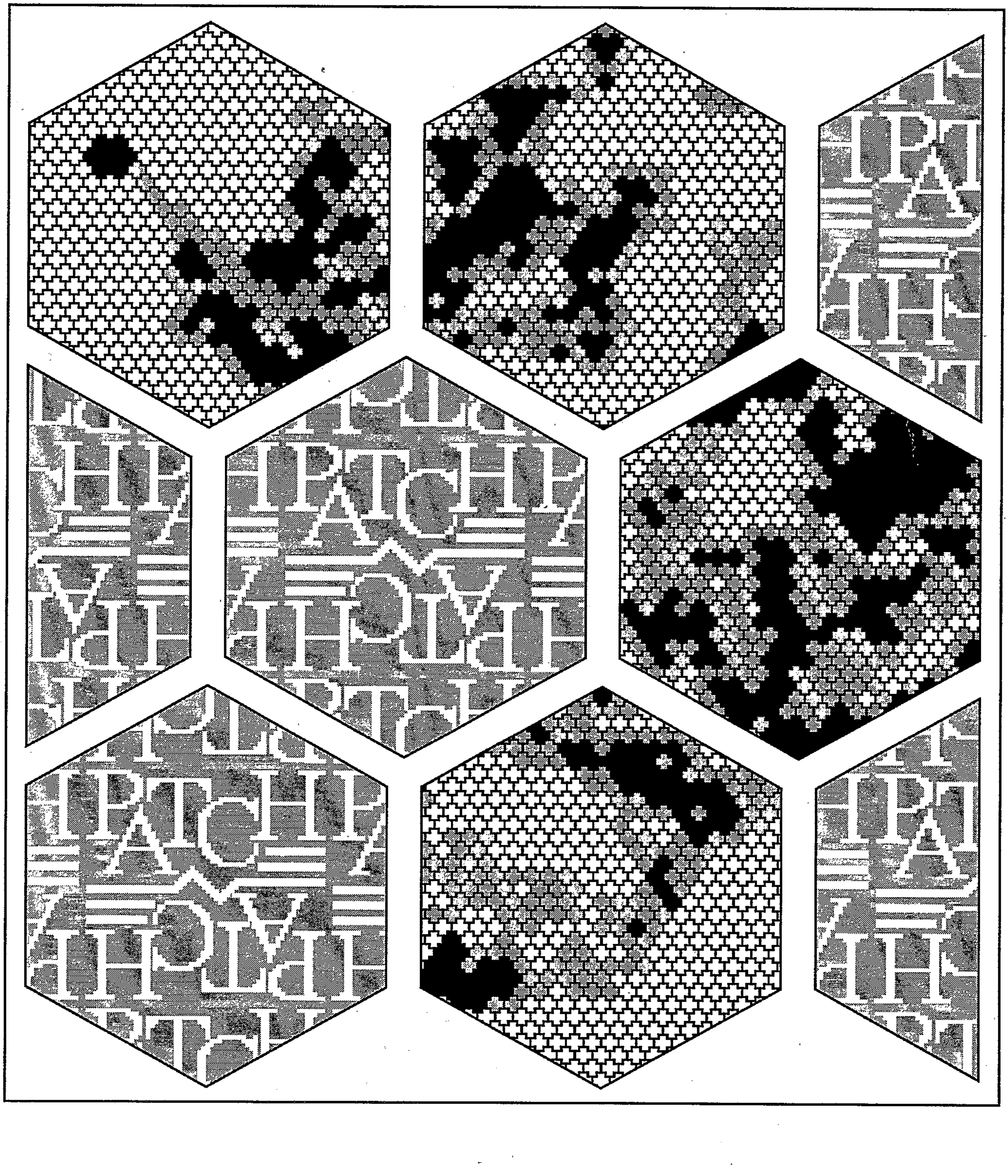
United States
Environmental Protection
Agency

Office of Research and
Development
Washington, DC 20460

EPA/600/R-98/135
September 1998

EPA

A Users Guide to the PATCH Model



EPA/600/R-98/135
September 1998

A Users Guide to the PATCH Model

by

Nathan H. Schumaker
U.S. Environmental Protection Agency
National Health and Environmental Effects Research Laboratory
Western Ecology Division
Corvallis, OR 97333

National Health and Environmental Effects Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency
Research Triangle Park, NC 27711

Notice

The information in this document has been funded in part by the U.S. Environmental Protection Agency. It has been subjected to the Agency's peer and administrative review, and it has been approved for publication as an EPA document. Mention of trade names or commercial products does not constitute endorsement or recommendation for use. The development of PATCH was also supported by the U.S. Forest Service through grant number PNW 90-340, the U.S. State Department through grant number 1753-000574, and the National Science Foundation through grant number BIR9256532.

Preferred Citation:

Schumaker, N.H. *A users guide to the PATCH model.*
EPA/600/R-98/135. U.S. Environmental Protection Agency,
Environmental Research Laboratory, Corvallis, Oregon.

PATCH is an excellent tool that will take a lot of the guesswork out of predicting the impacts of different habitat management scenarios on species' populations. Although the model seems complicated, it is actually the best "pre-packaged" spatial module we have seen. There are no shortcuts in spatially explicit models, and it would be intellectually dishonest and misleading to contrive a model that did not have the depth or flexibility of PATCH. Users will not master PATCH in a few hours, but neither will they need to be computer wizards. PATCH has the potential to stimulate breakthroughs in landscape-based resource management. It will do this because it makes biology and clear thinking the hurdle, not computer programming.

Peter Kareiva
Professor of Zoology
The University of Washington

PATCH is an excellent tool for applied ecologists seeking to predict the effects of land use change on populations of concern. PATCH is a vast improvement over any similar program of which we are aware. PATCH was clearly designed to address and overcome the most serious concerns voiced by critics of spatially-explicit, individually-based models. In our view, PATCH succeeds admirably in this goal, providing a modeling framework that will allow the intelligent and informative use of complicated simulation models to address environmental problems.

Daniel Doak
Associate Professor, Environmental Studies
The University of California at Santa Cruz

We found PATCH very easy to use, the manual to provide good documentation, and the examples to be sufficient to illustrate the most common applications of the model. The model should prove quite useful for exploring spatial solutions to the conservation of threatened and endangered species. Given that most species of concern are at risk due to habitat loss and fragmentation, the focus of the model on linking demography to habitat quality and geometry is highly appropriate. The model's integration with actual landscapes via a GIS interface is a unique strength that makes PATCH particularly relevant to "real world" conservation problems. We believe that many population and conservation biologists will make wide use of PATCH.

Barry Noon
Associate Professor, Fisheries and Wildlife
Colorado State University

Table of Contents

Preface viii

Chapter 1 **PATCH Basics** 1

Getting Started 1
File Name Conventions 1
Numeric and Text Fields 2
Mouse Conventions 2
Selecting the Active Window 3
The Zoom-Box 3
Window Names 3
Control Windows 4
Graphics Windows 4
Using GIS Imagery 5
Loading an Image 6
Creating Images with Arc/Info 7
Caveats and Concerns 7

Chapter 2 **Model Inputs** 8

Introduction 8
GIS Data 8
Habitat Affinities 8
Territory Size 9
Vital Rates 9
Movement Behavior 9

Chapter 3 **Model Outputs** 10

Introduction 10
Landscape Pattern 11
Home Range Analysis 11
Demographic Information 12
Observed Movement Rates 12
Observed Occupancy Rates 12
Source-Sink Analysis 13

Chapter 4 **Patch Identification** 14

Introduction 14
Habitat Weights 14
Adjacency 14
Measurement Conventions 14

Identifying Patches 15
Statistics Output 16
Using the Zoom-Box 17
Displaying the Results 17
Selecting a Patch 17

Chapter 5 **Territory Allocation** 18

Introduction 18
Scores and Breeding Status 18
Territory Min and Max 18
Habitat Sharing 19
Linking to Life History 20
Identifying Territories 20
Using the Zoom-Box 21
Displaying the Results 21
Selecting a Hexagon 22
Adjusting Min and Max Sizes 22
Aligning the Hexagon Grid 22
Hexagon Borders 22
Editing the Hexagons 23
Hexagon Compression 23
Changing the Resample Rate 24

Chapter 6 **The Movement Module** 25

Introduction 25
Movement Behavior 25
Random Walks 26
Walk Linearity 30
Nonrandom Movement 31
Site Fidelity 32
Reflecting Boundaries 33

Chapter 7 **The Life History Module** 34

Introduction 34
Survival and Reproduction 35
Linking to Habitat Quality 36
Demographic Stochasticity 38
Environmental Stochasticity 38
Miscellaneous Parameters 40
Running a Simulation 40
Life History Output Files 41
Using Movement Limits 43
Selecting Starting Sites 45
Time Series Analysis 45

The Parameters Window 46
The Functions Window 47
The Life History Functions 47

Chapter 8 **PATCH Utilities** 49

Introduction 49
Reading and Viewing Imagery 49
Miscellaneous Functions 51
Zoom-Box and Image Window 53
Tracking and Resampling 54
Zoom-Box Placement 54
Habitat Controls Files and Tables 55
Patch Identification 56
Hexagon Editing 58
Territory Construction 60
Life History File Names 62
Time Series Editor 63
Visualization and Initialization 63
Life History Controls 65
Life History Parameters 69
The Projection Matrix 72
Interpolation Functions 73
Movement Functions 75

Example 1 **Patch Identification** 77

Introduction 77
Starting PATCH 77
Reading GIS Data 78
Displaying GIS Data 78
Setting the Zoom-Box Size 78
Using the Zoom Window 78
Moving the Zoom-Box 79
Setting the Legend Weights 80
Identifying Habitat Patches 81
Picking Four or Eight Neighbors 82
Locating Core Habitat Areas 82
Saving the Results 82
Patch Statistics 83
Loading a Patch Map 83

Example 2 **Territory Allocation** 85

Introduction 85
Starting PATCH 85
Reading GIS Data 86

Displaying GIS Data .86
Setting the Zoom-Box Size .86
Using the Zoom Window .87
Setting the Legend Weights .87
Setting the Hexagon Size .87
Building a Territory Map .88
Adding a Border .89
Selecting a Hexagon .89
Territory Min and Max .90
Saving the Results .91
Locating Core Habitat Areas .91
Hexagon Statistics .91
Hexagon Compression .92
Hexagon Grid Alignment .94
Editing Hexagon Scores by Range .94
Editing Hexagon Scores by Hand .95
Randomizing the Territory Map .96
Changing the Resample Rate .97

Example 3 Demographic Simulations .98

Introduction .98
Getting Started .98
The Projection Matrix .100
The Vital Rates Factor .101
Runs, Years, and Utility .103
The Initial Population Size .104
Movement Parameters .104
Visual Feedback .106
Specifying an Output File .107
Running a Simulation .107
The Output Statistics .108
The Main Life History File .109
The ".stats0" File .109
The ".stats1" File .110
The ".stats2" File .110
Displaying the Output Data .111
The Output Imagery .112
The Occupancy Rate Map .112
The Source-Sink Maps .112
Editing Output Imagery .113
Environmental Stochasticity .114
Including Habitat Change .117

Preface

This manual provides instructions for using a spatially explicit demographic simulator called PATCH (a Program to Assist in Tracking Critical Habitat). The model was designed to project populations of territorial terrestrial vertebrate species through time, but it can perform other analyses as well. PATCH is ideal for investigations involving wildlife species that are mobile habitat specialists because it pays careful attention to habitat pattern and quality. Where the details of landscape pattern are relatively unimportant, a different model may be preferable. PATCH is a single species, females only model, thus it cannot track intra-specific interactions, or complications associated with mate finding or sex ratio. In spite of these limitations, PATCH can accommodate a broad range of analyses, and it should be useful for both theoretical and applied investigations.

I began work on the precursor to PATCH while attending a summer school on Patch Dynamics organized by S. A. Levin, T. M. Powell, and J. H. Steele and held at Cornell University in the summer of 1991. I continued the development of the model from 1991 to 1995 at the University of Washington. Funding constraints encountered during this period resulted in the model becoming essentially a life history simulator for the Northern Spotted Owl. From 1995 to 1998, I transformed the spotted owl simulator into the present PATCH model. I wrote PATCH entirely in the C programming language, and worked hard to keep it user friendly and bug free. The source code to the PATCH model is not being released for two reasons. First, the source code consists of about 40,000 lines of text. This code is modular and straightforward, but it would be difficult for anyone other than the author to modify it correctly. Second, having a single home for the PATCH source code guarantees that copies of the model with the identical version numbers are truly the same.

PATCH's data requirements are minimal. It must be supplied with habitat maps, typically from a geographic information system (GIS), and data specifying habitat use (territory size and habitat affinity), vital rates (survival and reproduction), and the movement behavior of the species of concern. PATCH requires that its GIS data be stored in Sun Rasterfiles. The model is guaranteed to run correctly only on a Sun Microsystems computer running the OpenWindows window manager. Problems may arise if it is used with Sun's Common Desktop Environment. This manual assumes some familiarity with both the Sun operating system and the OpenWindows window manager.

Models are never perfect, and PATCH is no exception. Errors in a model's outputs can result from poor or inadequate design, mistakes made during its construction, and from limits on input data quality and assumptions. The PATCH model has been designed to help you investigate and minimize sources of error. PATCH was put through an extensive debugging process in which the performance of every algorithm was carefully validated. This bug checking process was intended to eliminate poor construction as a source of model error. In addition, PATCH is designed to require as few input parameters as possible, and it brings a minimum of complexity to the simulation of organism's life histories. But ultimately, it is up to you to ensure that PATCH is used responsibly. In order to take full advantage of this program it is important that you understand the implications of altering the assumptions behind the simulations. Take the time to conduct and explore sensitivity analyses and alternative parameterizations to get a grasp of how strongly your results are effected by the particular set of assumptions and life history values you choose. The confidence you place in PATCH's projections should reflect the reliability of the data and assumptions you build into its simulations.

PATCH's pattern identification module simply quantifies landscape pattern, and as such is not subject to design flaws. But errors in the GIS imagery can be expected to propagate through the pattern identification process. PATCH's territory allocation module analyzes GIS landscapes and provides input to the demographic subroutine. Errors in the GIS imagery will also be reflected in the territory maps that you construct. The design of the territory allocation module is straightforward, but it restricts the model's use to territorial species. The territory allocation module allows you to carefully test the sensitivity of the model results to uncertainty in its input parameters.

Concerns about the accuracy of the model outputs can be expected to focus primarily on PATCH's life history simulator. But the life history module's simplicity and flexibility should help you to assess your confidence in its results. The life history module is based upon a population projection matrix and, in an optimal landscape with abundant breeding sites, its projections are going to closely match those from a matrix model. Population-level attributes such as density dependence emerge directly from the model's explicit use of space, and are never hard wired in the source code. Other than the vital rates used in a matrix model, you need only supply habitat affinities, territory sizes, and rules defining movement behavior. Uncertainties in these parameters can be explored directly through sensitivity analysis.

The PATCH model is distributed on a compact disk (CD). The contents of the CD include the directories "bin", "man", and "gis". The "bin" directory contains the PATCH model (compiled for Sun Microsystem's OpenWindows and for LINUX) and an S-PLUS macro that is useful for plotting the model output. S-PLUS is a statistics package produced by MathSoft, Inc. The "man" directory contains copies of this manual in PDF (Adobe Acrobat) and PostScript formats. The "gis" directory contains seven maps (Sun Rasterfiles) ready to be used with PATCH. One of these contains imagery of the Clayoquot Sound region of western Vancouver Island in British Columbia, Canada. This image was created by David Leversee of the Sierra Club of Western Canada. A second image, created by Warren Cohen of the U. S. Forest Service, depicts part of the central western Cascade mountain range in Oregon. Three images of the Olympic National Forest (ONF), exhibiting forest conditions in 1940, 1962, and 1982, are also included. These data can be used to test the demography module's time series facility. The ONF data were developed by Peter Morrison and David Leversee for The Wilderness Society. The last two data sets consist of a color spectrum that may be helpful when customizing imagery, and a control file and key for use with PATCH's source-sink output maps.

The early development of the PATCH model was supported by USDA/USFS Grant PNW 90-340, U.S. State Department Grant 1753-000574, and NSF Grant BIR9256532. Subsequent development was supported by the U.S. EPA. I would like to thank Peter Kareiva, Steve West, Stan Gregory, Dixon Landers, Joan Baker, and Roger Blair for guidance and support. The Wilderness Society donated the Olympic Forest imagery. David Leversee and the Sierra Club of Western Canada donated the Clayoquot Sound imagery. Warren Cohen, of the U.S. Forest Service, provided the Oregon Cascades data. The PATCH model and manual were reviewed by Barry Noon, Daniel Doak, Peter Kareiva, David Bigger, Cynthia Hartway, and Jon Belak. Lastly, I'd like to thank Dawn Moyer-Schumaker for tolerating the years of obsession and long hours that characterized the construction of PATCH.

I look forward to collaborating with other investigators and users of the PATCH model. In this context, I will consider proposals that involve modifying the model to address specific research or management questions. I also encourage users to send me suggestions for improving the model and this manual. Lastly, I would like users to send me citations for all published work that incorporates the PATCH model.

Nathan H. Schumaker
U.S. Environmental Protection Agency
200 SW 35th St.
Corvallis, OR 97333
(541) 754-4658
nathan@mail.cor.epa.gov

Chapter 1 PATCH Basics

Getting Started

Sun Microsystem's OpenWindows window manager must be running before the PATCH model can be started (the Common Desktop Environment may not work correctly with PATCH). First, move to the root directory of the CD or to the appropriate location on the hard disk if PATCH has been copied off the CD. At the UNIX prompt, type "bin/PATCH". This calls up the initial window, referred to as the *title window* (Figure 1). Next, click anywhere in the title window, which causes PATCH to display its control windows in an array at the upper left of the screen. This collection of windows is shown in Figure 2.

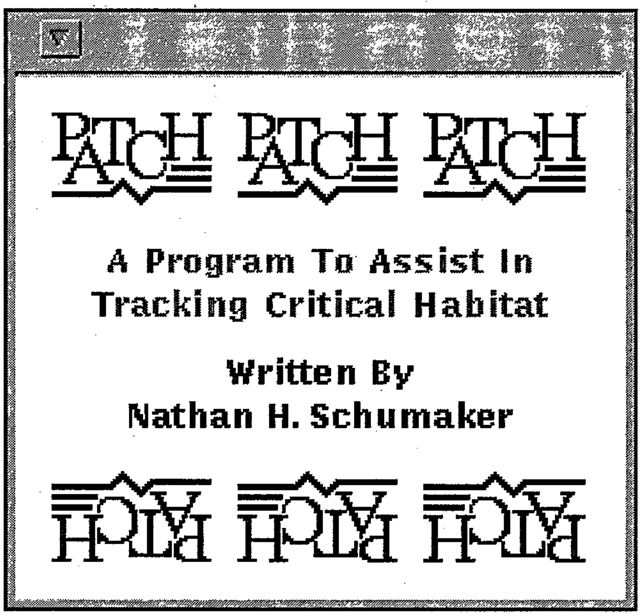


Figure 1. The title window. This window is displayed when PATCH is first started, and it is used to call up all of the other control windows. The header of the title window displays the version of PATCH that you are using (not shown here).

File Name Conventions

PATCH reads from and writes to the location in the file system at which the model was called up, unless absolute path names are used. In some circumstances, PATCH automatically appends extensions to the end of file names. Model output can always be directed to a file. If an output file name is not specified, PATCH writes its data directly to the monitor.

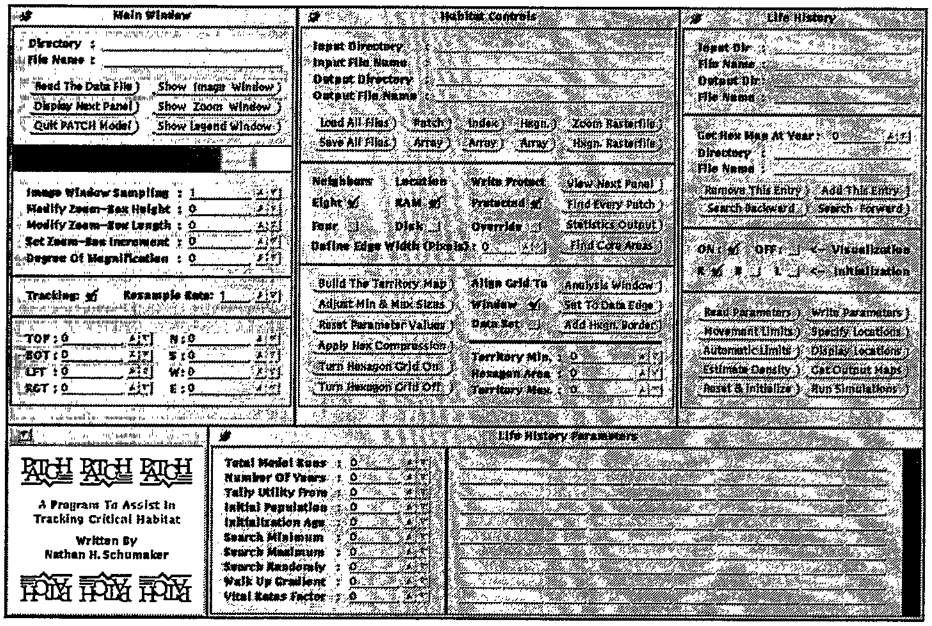


Figure 2. PATCH's control windows. Two additional panels and another control window are hidden behind those displayed above. The hidden panels can be viewed by clicking the "Display Next Panel" and "View Next Panel" buttons in the main and habitat controls windows, respectively. The hidden window can be displayed by clicking the rectangular bar shown in the lower right hand corner. Right clicking in the appropriate area also displays the hidden window or panels.

Numeric and Text Fields

When entering file names into PATCH's text fields, you do not have to type a carriage return at the end of a line. When entering numbers into numeric fields, you must use a carriage return to force the new value to be read by the model. A carriage return is not necessary when a numeric field is being modified using the increment or decrement widgets () or the up and down arrows on the keyboard.

Mouse Conventions

The PATCH model is designed to be used with a 3-button mouse. The following conventions are used throughout this manual when references are made to the mouse:

- The term *click* by itself means push the left mouse button.
- *Double click* means push the left mouse button twice, quickly.
- Use of the middle and right mouse buttons are denoted by *middle click* and *right click*, respectively.

Table 1 displays the various consequences associated with clicking the mouse buttons in PATCH's graphics windows. The features mentioned in Table 1 are discussed in subsequent sections of this manual.

Table 1: Clicking in Graphics Windows

	Double Click	Left Click	Middle Click	Right Click
Legend Window		Increment Weighting	Start Image Editing	Decrement Weighting
Image Window	Paint Window		Move Zoom-Box	Stretch Zoom-Box
Zoom Window	Paint Window	Edit Pixel	Select Pixel	Move Zoom-Box
Analysis Window	Paint Window	Edit Patch, Hexagon	Select Patch, Hexagon	Move Zoom-Box

Selecting the Active Window

Sun Microsystem's OpenWindows and Common Desktop Environment window managers provide you with two mechanisms for selecting the active window. The active window is the window that receives the commands you enter from the keyboard or the mouse. The options for making a window active are to either click on it or move the mouse pointer into it. When using PATCH, you should always make sure that the active window is defined by the location of the mouse pointer. Some of PATCH's utilities do not work correctly when a mouse click is required to specify the active window.

The Zoom-Box

PATCH allows you to perform analyses on any rectangular subset of a GIS image. The portion of the image to be used is specified by the placement of a small rectangular box in the principle graphical window (the *image window*). This rectangular box is referred to as the *zoom-box,* since PATCH allows you to zoom into the data within it. The zoom and analysis windows display only the GIS imagery present within the zoom-box, which can be enhanced using a magnification parameter that appears in a control window called the *main window.*

Window Names

Windows in PATCH that contain buttons and text fields are referred to collectively as *control windows.* Windows that contain images are referred to collectively as *graphics windows*. Every window in PATCH has a name, and these names are displayed in the window headers. However, exceptions to this rule are made for the title and legend windows. The title window is the very first window that comes up when the model is started. The title window header displays the version of PATCH being used. The legend window displays the classes present in the GIS data. The legend window header displays the name of the GIS data set being used. Moving the mouse pointer into the image, zoom, or analysis window causes its header to display the name of the GIS data set being used. When the mouse pointer is removed from the window, the window header redisplays the window name.

Control Windows

Each control window is designed to facilitate a particular class of operations. The title window is used to iconify the entire PATCH interface and to recall other control windows that have been removed from the desktop. The main window permits you to read, display, and modify GIS data. The habitat controls window allows you to quantify landscape pattern and to construct and edit a territory map. The life history windows are used to drive the demographic analysis.

Graphics Windows

The PATCH model contains four graphics windows. The image window displays a subsampled version of the entire GIS image. The size of the image window is set using a numeric field called *Image Window Sampling* in the main control window. A sampling value of n forces the image window to be constructed from every nth row and nth column of the GIS image. Hence smaller values of n result in larger image windows that display more detail. The legend window displays the colors and names of each category present in the GIS image. The size of the legend window is set automatically by the model. The zoom window displays the contents of the zoom-box at a specified magnification (see the *Degree Of Magnification* field in the main control window). The analysis window displays the results of any analysis conducted with the model, and is also linked to the zoom-box and the magnification parameter.

The legend window is always painted automatically by the model. This is not true for the image, zoom, and analysis windows. To repaint these windows, it is necessary to double click within the window. However, if the zoom or analysis window dimensions do not equal the zoom-box size times the magnification value, then a double click resizes the window (this is not the case for the image window, whose size can be altered only by using the *Image Window Sampling* field). A second double click paints the image after a window has been resized. Throughout this manual, the term *up to date* is used to describe a window that is displayed on the screen, has the correct size, and has been painted with an image.

Different portions of your GIS imagery can be viewed in the zoom and analysis windows by changing the size and placement of the zoom-box. The zoom-box size and placement can be modified using any of the many fields in the main control window that specify its location within the GIS data. The edges of the zoom-box can also be moved to the location of the mouse pointer by right clicking in the image window. In addition, the zoom-box location can be altered by middle clicking in the image window. The zoom-box can also be moved by left clicking in either the zoom or the analysis window. In these instances, you specify the direction in which the zoom-box moves by clicking in one of four quadrants, delineated by connecting opposite corners of the window along diagonals (Figure 3). The zoom-box moves a distance equal in pixels to the value of the numeric field in the main control window labeled *Set Zoom-Box Increment*. However, if this field is set to zero, the zoom-box moves exactly its own width, or height.

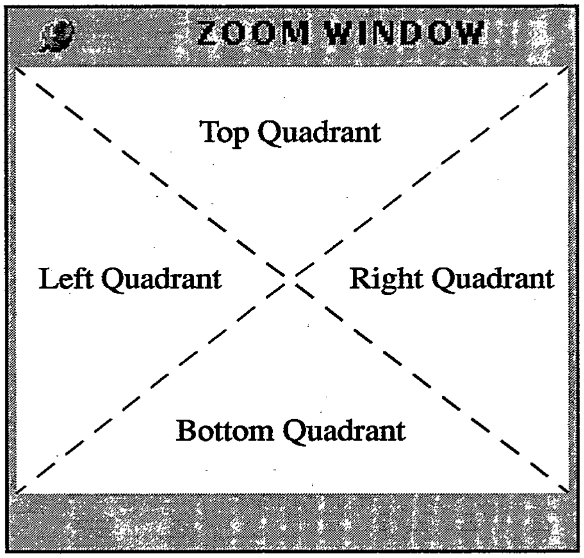


Figure 3. The zoom-box can be moved incrementally by right clicking in either the zoom or the analysis window. The zoom-box moves up, down, left, or right, depending on if the mouse pointer is located in the top, bottom, left, or right quadrant, respectively.

Using GIS Imagery

To begin using PATCH, it is first necessary to load in a GIS image. PATCH only reads images that have been formatted as 8-bit, uncompressed, full color Sun Rasterfiles. However, most standard GIS applications create Sun Rasterfiles of this type, or another raster data format (e.g., GIFF, TIFF) that can be converted to a Sun Rasterfile. PATCH also stores some necessary ancillary data in a text file referred to as a *control file*. If this text file does not exist, PATCH creates one, but you must then edit it to supply meaningful values for the Universal Transverse Mercator (UTM) coordinates associated with the image's edges, the size in meters of the data pixels, and the names of the legend categories. The UTM information is not critical to running the model.

Loading an Image

You can begin by reading one of the sample images provided with the PATCH distribution. To do this, it is necessary to provide a directory and a file name that describe the location of the imagery in the UNIX file system. If the PATCH program is started from the CD, then the sample imagery should be found in a directory called "gis". Load the imagery by entering its directory and name into the main window, and then click on the button labeled *Read The Data File* (Figure 4). Next, click on the button in Figure 4 labeled *Show Image Window.* This window initially is painted black. Double click in this black area to display the GIS data set that you have loaded into the model. This small window can be increased in size by lowering the value of the text field labeled *Image Window Sampling* that appears just below the color palette in Figure 2. You always have to read a GIS data set before running analyses with PATCH.

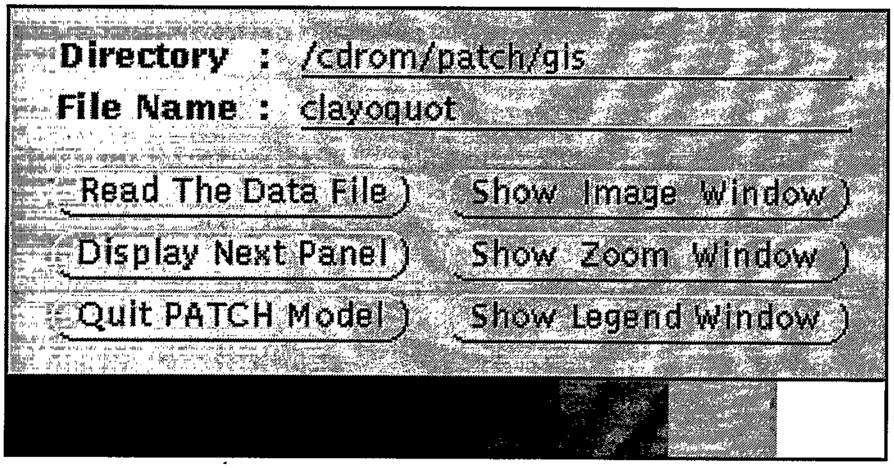


Figure 4. The top panel in PATCH's main window. The buttons on the right side of this panel are used to access three of PATCH's four graphics windows. The "Read The Data File" button loads a GIS image into the model. Clicking on "Display Next Panel" sends this entire panel to the background, and brings a second set of buttons into the foreground (this can also be accomplished by right clicking in the panel). The colored boxes (depicted here as in gray-tones) at the bottom of the panel are used to modify the color of the zoom-box and of the hexagon grid in the zoom window.

Creating Images with Arc/Info

Arc/Info is an extremely popular GIS program written by ESRI. This software package has a raster modeling toolkit called GRID. You can use GRID, and the instructions provided below, to convert an Arc/Info raster coverage into a 8-bit Sun Rasterfile with an imbedded colormap suitable for use with PATCH. You must first construct a table that assigns colors to the pixels in the GRID image. The first column of this table must be a list of the data values present in the GRID image. An 8-bit Sun Rasterfile cannot be created if any of these data values lie outside the range from 0 to 255. The remaining columns in the table specify the red (R), green (G), and blue (B) levels to be associated with each data value. These RGB levels must also lie in the range from 0 to 255, and all of the entries in this table must be separated by spaces. The Arc/Info SAVECOLORMAP command can be used to construct this table. The Arc/Info command you use to generate the Sun Rasterfile is:

GRIDIMAGE <input grid> <color map file> <output image> sunraster

where *input grid* refers to the GRID coverage to be converted, *color map file*, refers to the table of RGB values described above, and *output image* specifies the name of the Sun Rasterfile to be constructed.

Caveats and Concerns

Complex spatially explicit models like PATCH should be used with caution because they can generate convincing results from questionable assumptions and uncertain parameter estimates. Listed below are a few caveats and concerns worth remembering as you work with PATCH. This is only a short list, and the general theme to keep in mind is the importance of exploring the sensitivity of your results to the assumptions and parameter estimates that you make.

- PATCH is a females-only model and it will not exhibit an Allee effect. For this reason, PATCH may underestimate the risk faced by populations that are subject to periods of extremely low density.

- PATCH's simulated organisms necessarily have identical habitat needs throughout their entire life cycles. This constraint is unrealistic for many species that exhibit distinct life stages or that utilize separate summer and winter ranges, etc.

- PATCH is designed for territorial, terrestrial organisms that are sensitive to the details of landscape pattern. Many species are not territorial, and others tend to utilize space at much smaller scales than those typically associated with natural or human disturbances.

- PATCH's environmental stochasticity is correlated across space. In the limit of very large maps, this constraint can become unrealistic.

Chapter 2 Model Inputs

Introduction

This chapter presents a brief description of the data necessary to drive the PATCH model. PATCH can quantify landscape pattern using only GIS data as input. The model can also perform a home range analysis using GIS data and information specifying habitat affinities and territory sizes. Source-sink analyses can be developed from habitat affinities, territory sizes, and estimates of vital rates. In order to conduct demographic simulations, you must also provide PATCH with information describing a species' movement behavior.

GIS Data

The imagery supplied to the PATCH model must come in the form of an 8-bit, uncompressed, Sun Rasterfile with square pixels. This file must be given a name that ends in ".sun" in order for the model to locate it. PATCH stores some ancillary data in a text file that is always assigned the same name as the Sun Rasterfile, minus the ".sun" extension. These text files, hereafter referred to as *control files*, provide a mechanism for you to supply Universal Transverse Mercator (UTM) coordinates for the edges of the data, the pixel size (in meters), and the names and colors associated with each data category. If a control file is missing, PATCH automatically creates one that contains default values for every field except for the colors, which are assigned the values actually present in the Sun Rasterfile.

Every 8-bit, full color Sun Rasterfile contains an embedded colormap. As mentioned previously, when PATCH builds a missing control file, this colormap is used to assign red, green, and blue values to the image categories. This is the only situation in which a rasterfile colormap is accessed by the model. Thus if the colors in the control file are altered, PATCH's graphics windows display this change. In addition, when PATCH creates a new Sun Rasterfile, it constructs its imbedded colormap from the color table present in the control file being used.

Habitat Affinities

Habitat affinities in PATCH are integer weighting values between 0 and 99 associated with each habitat class present in the GIS data. These weights (also referred to as habitat utility indices) are relative measures of a habitat's importance to the species under study. They are assigned by clicking in the legend window. Categories present in the imagery that are assigned weights of zero are totally inhospitable to the species. Habitat classes having the largest of the assigned weights serve as optimal habitat for the species. The data necessary to specify habitat weights can be hard to come by. Ideally, you should specify the weights based upon field data that directly measure a species' preference in different habitat types. An alternative that you can use if field data are not available is to survey experts and request that they rank the importance of available habitats for the species in question. In any case, remember that the habitat weights are relative, not absolute, measures.

Territory Size

The territory size parameter is designed to be set to the typical size of a territory for the species being modeled. In addition, estimates can be provided for the minimum territory size (expected in optimal habitat), and a maximum territory size (expected in marginal habitat).

Vital Rates

PATCH's vital rates are estimates of survival and fecundity (the number of female offspring per female) supplied in the form of a Leslie or Lefkovitch projection matrix. Because PATCH uses projection matrices, you must determine whether the species under study is best described with an age or stage based model. Separate survival and fecundity values are required for each age or stage class identified in the species' life history. The survival rates (or transition probabilities) within any one column of the projection matrix must sum at most to one. PATCH's life history module incorporates a post-breeding census, which means that the first age or stage class always consists exclusively of newly recruited individuals that are not yet of breeding age. Thus the upper left entry in PATCH's projection matrix is always fixed at zero.

Movement Behavior

Movement behavior is defined collectively through a series of parameters. Movement events occur when young of the year (those individuals born in the last breeding pulse) disperse in search of vacant sites in which to set up territories. In addition, floaters (older individuals without territories) must also search the landscape for breeding sites. Territorial individuals may or may not elect to move, and this behavior is controlled by the site fidelity parameter and the quality of the site the individuals presently occupy. The movement process can involve a pseudorandom walk, or it can assign the organisms a complete knowledge of some portion of the landscape and then let them move directly to vacant breeding sites. If a random walk is being employed, you must indicate the species' movement ability and supply parameters that influence the linearity in searcher's movements, and their tendency to move up gradient from low to high quality habitats. Otherwise, you must specify a search radius and whether individuals should attempt to obtain the best or closest available site. Regardless of the movement strategy employed, three degrees of site fidelity can be imposed.

Chapter 3 Model Outputs

Introduction

Outputs from the PATCH model fall into two general categories: pattern-based metrics and demographic analyses. Pattern-based outputs include patch-by-patch descriptions of landscapes, assessments of the number, quality, and spatial orientation of breeding sites, and map-based estimates of the occupancy rate and the source-sink behavior of the breeding habitat. The principal demographic outputs generated by the PATCH model include several measures of population size as a function of time, realized survival and fecundity rates (rates that reflect the limitations on a population imposed by habitat quality and landscape pattern), and assessments of the occupancy rate and source-sink behavior of the breeding sites present in a landscape.

PATCH's pattern-based outputs help you to quantify landscape structure and quality. These outputs vary in the extent to which they are sensitive to species' life history. Patch-based analyses might reflect only species' habitat preferences, while territory-based metrics also consider estimates of territory size. Occupancy rate and source-sink analyses go further to incorporate estimates of species' vital rates. PATCH's demographic outputs include estimates of population size, age structure, and spatial distribution. PATCH's life history module can also help you quantify the impacts of landscape change on population viability, estimate changes in vital rates corresponding to habitat loss or fragmentation, and identify source and sink populations within a landscape. PATCH's source-sink analysis, coupled with its capacity to incorporate landscape change through time, allow you to identify, track, and modify clusters of habitat that play a critical role in the maintenance of a population. This is the model's most unique feature and the reason it is called *a Program to Assist in Tracking Critical Habitat*, or PATCH, for short.

A great deal of interest has recently developed in exploring the performance of indices of landscape pattern as predictors of ecological quality and function. Because PATCH both quantifies landscape pattern and projects population trends, it is ideally suited to this type of investigation. For example, PATCH will make it easy for you to search for measures of landscape pattern that are well correlated with estimates of dispersal success or population persistence.

The ecologies and life histories of most species are poorly known. In the common situation where data are sparse and parameter values less than certain, you will have to find creative ways to make the most of what PATCH can offer. Many approaches are possible. You can begin with very simple habitat-based analyses and add life history and behavioral information iteratively. An assessment of landscape quality, in the simplest case, could consist only of counting the number of pixels of habitat of various qualities. A slightly more detailed analysis might be performed by examining the size structure of the habitat patches present in the landscape. Supplying territory size estimates would allow you to assess the number of high quality breeding sites. Vital rates could be specified and the numbers of source and sink habitats identified. All of this data, and more, can be collected from PATCH without even running a life history simulation. Still, sensitivity analyses should always be performed to examine the consequences of your assumptions and specific model parameterizations.

Landscape Pattern

PATCH can generate a histogram showing the frequency of each habitat type present in any rectangular subset of the landscape. PATCH can also quantify habitat pattern on a patch-by-patch basis. The module that does this will supply you with a table that includes the area, weighted area, core area, and perimeter of each patch. Area is measured as the number of pixels of habitat present. Weighted area is measured as the sum of the weighting values (habitat affinities) assigned to each pixel in a patch. Core area is computed based on a edge width that you provide (in pixels). A patch's core area is equal to the number of pixels separated from the patch's perimeter by a distance equal to at least one edge width. These pattern-based measures can be used (outside of PATCH) to develop indicators of habitat quality, such as perimeter / area ratios or estimates of fractal dimension. In addition to generating these numerical outputs, PATCH also displays the results of the patch identification process in a graphics window.

Home Range Analysis

PATCH's home range analysis complements the assessment of landscape pattern described above. The home range analysis is also an important part of the process of conducting demographic simulations. The key procedure involved in PATCH's home range analysis, termed *territory allocation*, involves intersecting a GIS image with an array of hexagonal cells. PATCH was designed with the intent that the area of these hexagons would equal the typical size of a territory for the species being modeled. In addition to setting the hexagon size, you can also specify a minimum and a maximum territory size. The design of the territory allocation module is such that the minimum territory size is observed in optimal habitat, and the maximum territory size is observed in more marginal habitats.

One output from the territory allocation process is a new raster data set displaying each individual hexagon and its breeding status (suitable vs. unsuitable). A second output resulting from territory allocation is a table that displays every hexagon's score (the mean value of the weights assigned to each of its pixels), the overall amount of habitat it contains, its weighted habitat area, its breeding status, and the amount of core and edge habitat it contains (computed patch by patch).

Demographic Information

When conducting demographic simulations, PATCH compiles a table exhibiting a year-by-year tally of the floater and breeder populations, the number of individuals in each age or stage class, and the overall population size. To facilitate analyses involving multiple replicate simulations, the means and standard deviations of these quantities (taken over every replicate run) are computed as well. PATCH also tracks the effective aggregate survival rates and fecundities, on an age or stage class basis, as mean values computed from every replicate run. These observed vital rates can be used to construct a time series of projection matrices that can together exactly recreate the population trajectory generated by the simulation model.

Observed Movement Rates

For every year of a demographic simulation, PATCH compiles the mean dispersal distance and reports it as both the total distance moved and the net straight-line displacement from the starting point. The term *dispersal* is used to describe only the movements of young of the year (individuals born in the last breeding pulse). The distances moved by older individuals are not reported in any of the output files generated by the PATCH model.

Observed Occupancy Rates

PATCH examines whether or not each breeding site is occupied at the end of the final year of each replicate simulation. These data are stored and used to generate three different types of outputs. First, PATCH constructs a map that displays the relative occupancy rate of each breeding site present in the landscape. This map can be used to glean a general picture of the overall patterns of occupancy exhibited by the model species. Second, the model creates a numeric output that pairs each hexagon's total occupancy rate (again, gathered once per replicate run, at the end of the final model year) with information quantifying its quality. These data can be used to segregate patterns of occupancy based on measures of habitat quality. Third, PATCH allows you to examine the population density that existed in any subset of a landscape. This feature can be used to generate a distribution of occupancy rates built up from multiple replicate simulations. This output is designed to be compared directly to density estimates gathered in the field.

Source-Sink Analysis

Throughout this manual, demographic sources are defined as sites in which reproduction exceeds mortality. If mortality exceeds reproduction, then a site is referred to as a sink. PATCH compiles estimates of both the expected and the observed source-sink behavior of each breeding site present in a landscape.

PATCH can conduct a source-sink analysis using only data describing a species' habitat affinities, territory size, and vital rates. In a typical landscape, survival and reproductive rates change from site to site because habitat quality varies. PATCH uses this information to compute the mean steady-state replacement rate (as the dominant eigenvalue of the associated projection matrix) for the model species in each breeding site. With this information, expected demographic sources and sinks are identified throughout the landscape. This source-sink analysis can be used as an initial estimate of the importance of different regions of the landscape for the model species. Deviations from these expected outcomes result primarily from limitations inherent in an organism's searching behavior.

PATCH also tracks immigration into and emigration from each breeding site, and uses this information to identify observed demographic sources and sinks throughout the landscape. PATCH accomplishes this by incrementing a counter each time an individual leaves a breeding site, and decrementing the same counter each time an individual enters the site. Individuals that simply pass through a site produce no net change in the counter's value. On the other hand, when individuals are born in and subsequently disperse from a breeding site, the counter's net value is incremented. When individuals move into a breeding site and die, the counter's net value is decremented. These counters can be held at zero until a specified number of years has passed (so that transient effects resulting from the initial conditions can die down), and they can also be averaged across replicate simulations.

PATCH's source-sink analysis is presented to you in three files. The first file contains a table that displays the score and the expected and observed source-sink behavior of each breeding site, as described above. The second and third files are Sun Rasterfiles that provide a visual representation of the expected and observed source-sink behavior. Each hexagon in these images is displayed as a 2 x 2 pixel square. Constructed this way, these output images are easier to compare visually to the GIS input data from which they were derived.

Chapter 4 Patch Identification

Introduction

It is often desirable to identify aggregate features of landscape pattern that correlate with measures of ecological quality such as habitat connectivity or population viability. PATCH facilitates this type of analysis because it includes a module devoted specifically to quantifying landscape pattern. PATCH's approach to providing this information is to break a landscape up into a collection of individual patches of habitat, and then to provide a limited amount of information about each one. Many different metrics can then be developed (outside of PATCH) from such measures of landscape pattern.

Habitat Weights

For the purposes of patch identification, PATCH requires that each legend category be assigned a weighting value (i.e., a habitat affinity), which takes the form of an integer between 0 and 99. Every pixel belonging to a category that has been assigned a nonzero weight is treated as habitat, whereas those with zero weights are considered non-habitat. Habitat weights are entered into the legend window by clicking on the different categories displayed there. A weighting value of zero is displayed in the legend as an empty box. The patch identification process treats all habitat classes with nonzero weights identically when computing patch shapes, sizes, perimeters, and core areas. However, the model also computes a weighted area, and this quantity can be used to rank patches based on the amount of high-quality habitat they contain.

Adjacency

A patch consists of any collection of pixels that touch, and that all have been assigned nonzero weights. PATCH provides you with two ways to define what it means for pixels to touch each other. One choice specifies that, for the purpose of patch identification, only a pixel's four adjacent neighbors (top, bottom, left, and right) actually touch it. The other option includes pixels touching at corners as well. Based on the rule applied, PATCH assigns every habitat pixel to one, and only one, patch. The areas, weighted areas, core areas, and perimeters assigned to the landscape's patches can all be affected by adjacency rule used.

Measurement Conventions

PATCH area is measured as the number of pixels of habitat. Weighted area is measured as the sum, taken over every pixel in a patch, of the weighting values assigned to each pixel. Core area is computed based on a edge width that you supply, specified as a number of pixels. A patch's core area is defined as the number of pixels separated from the patch's perimeter by a distance equal to at least one edge width in every direction (regardless of the adjacency rule being used). The actual values for patch area, weighted area, and core area can be obtained by multiplying these values by the area of a pixel (in square meters). Perimeters are measured as the total number of pixel edges present in a patch. Patch perimeters can always be converted from pixels to meters by multiplying by the length of a pixel. This is also true for the edge width parameter.

Many well known indices of habitat pattern can be built up from the measures just described. Examples include perimeter-area ratio, shape index, and estimates of fractal dimension. More importantly, this analysis provides you with the raw data necessary to search for measures of landscape pattern that function well as effective indicators of ecological quality. PATCH itself does not construct such metrics, but many other programs are capable of doing so using PATCH's outputs.

Identifying Patches

A panel in the habitat controls window is devoted to the process of patch identification (Figure 5). This panel allows you to specify whether data pixels are to be treated as having four or eight touching neighbors. This panel also allows you to run the patch identification process entirely in memory (*RAM*) or to store the relevant tables on the hard disk (*Disk*). The data generated by the patch identification process can always be saved to the disk after the process has completed, and the usual reason for setting the location switch to *Disk* is that not enough memory is available to build the tables in RAM. A write protect feature is also provided that helps to prevent data from being unintentionally overwritten. A numeric field is included in the panel for setting the edge width used in computing patch core areas. Core areas are computed automatically any time that the edge width field is set greater than zero, including when the territory allocation process is running.

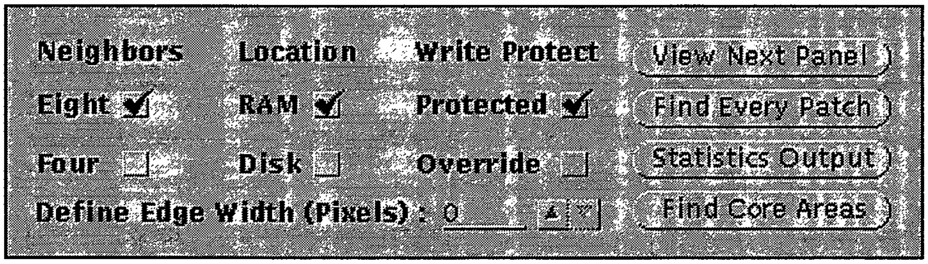
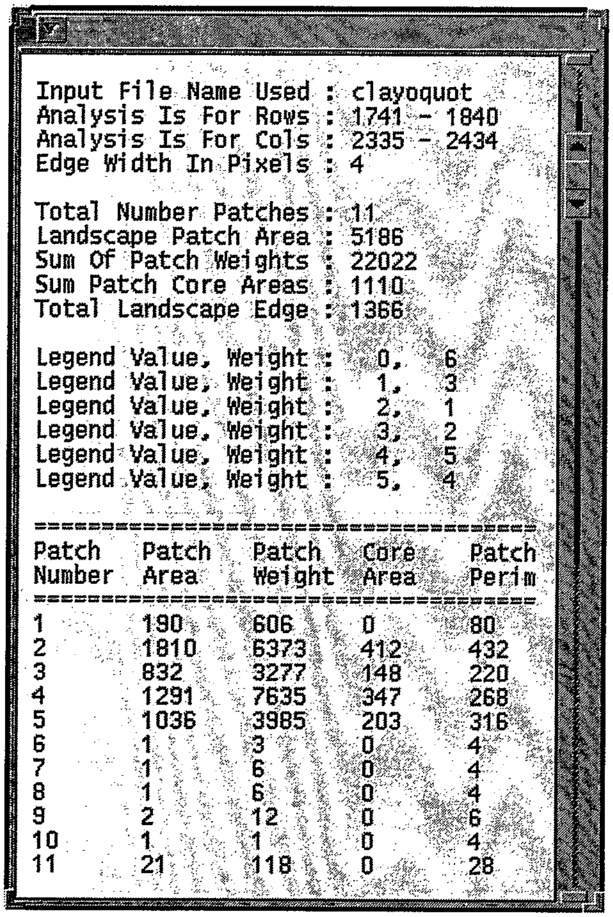


Figure 5. The panel within the habitat controls window that governs the patch identification process. See the text for descriptions of the panel's various fields, settings, and buttons.

Statistics Output

The button in Figure 5 labeled *Statistics Output* is used to generate a summary report describing the results of the patch identification or territory allocation process. These reports differ slightly, depending on whether patches or hexagons are present. The reports contain a list of the settings used to generate the data, and provide a summary of the model's description of each patch or hexagon. Figure 6 provides an example of the statistics output file that results from running the patch identification process. If the *Define Edge Width* field is set greater than zero, then PATCH identifies all of the core habitat present in the landscape prior to generating the statistics output file.



```
Input File Name Used : clayoquot
Analysis Is For Rows : 1741 - 1840
Analysis Is For Cols : 2335 - 2434
Edge Width In Pixels : 4

Total Number Patches : 11
Landscape Patch Area : 5186
Sum Of Patch Weights : 22022
Sum Patch Core Areas : 1110
Total Landscape Edge : 1366

Legend Value, Weight :   0,   6
Legend Value, Weight :   1,   3
Legend Value, Weight :   2,   1
Legend Value, Weight :   3,   2
Legend Value, Weight :   4,   5
Legend Value, Weight :   5,   4
```

Patch Number	Patch Area	Patch Weight	Core Area	Patch Perim
1	190	606	0	80
2	1810	6373	412	432
3	832	3277	148	220
4	1291	7635	347	268
5	1036	3985	203	316
6	1	3	0	4
7	1	6	0	4
8	1	6	0	4
9	2	12	0	6
10	1	1	0	4
11	21	118	0	28

Figure 6. A report generated using the "Statistics Output" button in the habitat controls window. Only legend weights that have been assigned non-zero values are listed in the header. This format of this report changes slightly if it is developed from a territory map instead of a collection of habitat patches.

Using the Zoom-Box

PATCH runs its patch identification process on the subset of the imagery contained within the zoom-box. If the zoom-box is 1000 pixels or more in width or height, then the patch identification process is conducted separately in rectangular blocks between 500 and 1000 pixels on a side. After the analysis has been performed on every one of these blocks, the results are pieced back together into a single data set describing the patches present in the landscape. As individual patches are identified, their bounding rectangles are displayed within the image window (the window that displays the zoom-box). This visual feedback helps you to monitor the progress of the patch identification routine.

Displaying the Results

The results of the patch identification process are displayed in the analysis window. The analysis window is accessed using a button located in the habitat controls window. The analysis window tracks the zoom-box just as the zoom window does, and patches are displayed in this window using a revolving scheme of four colors. There is no significance to the choice of colors assigned to patches in the analysis window, and adjacent patches often have identical colors. When the patch identification process is run on a large part of a GIS image, the results often cannot be displayed all at once because they would then extend beyond the edges of the monitor. However, the zoom-box can be decreased in size and moved around after the patch identification routine has run, and in this way the results can be viewed a bit at a time.

Selecting a Patch

The PATCH model keeps track of each patch located by the patch identification routine. When the results are displayed in the analysis window, you can middle click on a patch to display a summary of its properties in the window footer. When a patch is selected in this way, it is highlighted (painted white) and the analysis window footer displays the patch's number, its area (in pixels), and its perimeter (in pixel edges). The patch number can then be cross-referenced to data supplied in PATCH's output files.

Chapter 5 Territory Allocation

Introduction

Before PATCH's demographic analysis can be conducted, it is necessary to break a landscape into an array of territory-sized units. This process, termed territory allocation, involves intersecting the GIS image with an array of hexagonal cells. PATCH was designed with the intent that each hexagon's area would be set equal to the typical size of a territory for an individual of the species being modeled. PATCH allows you to define minimum and maximum territory sizes as well. Setting the minimum or maximum territory size to a value other than the hexagon size tends to eliminate artifacts associated with the exact placement of the grid. Weighting values (habitat affinities) are used to incorporate habitat quality into the territory allocation process, just as they are in the patch identification routine. The territory allocation module also includes a hexagon editing facility designed to help you explore the consequences for wildlife species of landscape change.

Scores and Breeding Status

When the territory allocation process is run, a territory map is generated and displayed in the analysis window. Each hexagon in this map has two attributes: its score and its breeding status. A hexagon's score is computed as the arithmetic average of the weighting values assigned to each of the data pixels contained within it. The scores are real numbers between zero and the maximum weighting value assigned to any of the categories present in the GIS imagery. A hexagon's breeding status is a binary attribute that dictates whether or not breeding is allowed at the site. Breeding status is determined based on the minimum and maximum territory sizes.

Territory Min and Max

The *Territory Min* and *Territory Max* parameters influence the allocation of breeding sites across a landscape. The minimum size corresponds to the size of a territory in optimal habitat, whereas the maximum territory size is realized in marginal habitats. The territory minimum and maximum sizes do not affect the hexagon areas. Instead, these parameters govern the degree to which habitat is shared across hexagon boundaries, as the model allocates breeding sites throughout a landscape. The territory allocation algorithm proceeds in several steps. Initially, PATCH computes a threshold score using the equation

$$\text{threshold score} = \text{max weighting value} \times \frac{\text{min territory size}}{\text{hexagon size}}.$$

This relationship defines the threshold score as the score assigned to a hexagon containing exactly the minimum territory size worth of optimal habitat, and no other habitat whatsoever. Any hexagon with a score of at least this threshold value is automatically labeled suitable for breeding. Hexagons that do not meet this threshold value still have a chance to be classified as breeding sites, as long as they contain some habitat. However, breeding status can be automatically assigned to every hexagon with habitat simply by setting the minimum territory size to zero. As a rule, hexagons that contain no habitat can never be suitable for breeding. The next step in the territory allocation process is governed by the maximum territory size parameter.

Habitat Sharing

PATCH determines the extent to which habitat can be shared across the hexagon boundaries using the expression

$$\text{expansion} = \frac{\text{maximum territory size}}{\text{hexagon size}} - 1.0\,.$$

The expansion parameter defines the maximum amount of habitat (expressed in fractions of a hexagon) that a hexagon can borrow from its six immediate neighbors. The maximum territory size is never allowed to exceed seven times the size of a single hexagon, and thus the expansion parameter can never exceed the area of a hexagon's six neighbors. After identifying every hexagon that contains enough high-quality habitat to automatically qualify as a breeding site, PATCH builds a list of all remaining sites that have any habitat at all. These hexagons are sorted by score, in decreasing order. The hexagons are then allowed, in turn from best to worst, to borrow habitat from their neighbors up to the limit set by the expansion parameter. For each hexagon in this list, lending continues until the hexagon either meets the suitability threshold, reaches the borrowing limit, or exhausts the habitat available in its neighborhood. Habitat can only be lent once, except that when hexagons are unable to expand up to the suitability threshold, they return any habitat they have borrowed in the process.

The amount of habitat that can be borrowed depends on whether the lending hexagon is suitable for breeding. Suitable hexagons are allowed to lend only what they hold in excess of the threshold score, whereas unsuitable hexagons can lend all of their habitat. Borrowing begins with the neighbor having the largest amount of habitat to lend, and concludes with the neighbor having the least. This process, coupled with the initial sorting of the borrowing hexagons by score, roughly optimizes the allocation of suitable breeding sites across the landscape.

Setting the expansion parameter to 2.5 would tell PATCH that portions of each neighbor could be borrowed until a total of 1.5 hexagons worth of the neighboring habitat had been claimed. The borrowing process is conducted under the assumption that each lending hexagon's habitat is distributed uniformly throughout its area. Though this is not always the case, the use of this assumption allows PATCH to avoid following the borrowing process on a pixel-by-pixel basis within hexagons, and at the same time does not significantly compromise the results. It is important for you to remember that the process of borrowing habitat does not change any features of a territory map other than the classification of hexagons as suitable or unsuitable for breeding.

The additional energetic costs of foraging in, and defending larger territories are approximated through the borrowing process because hexagons labeled suitable for breeding by virtue of having borrowed habitat have lower scores than those that had sufficient habitat on their own. These lower scores typically translate into higher mortality rates and lower reproductive output in the demographic analysis.

Linking to Life History

PATCH's use of territory maps is designed to keep the life history simulator as straightforward and general as possible. The structure of a territory map is simple compared to the GIS imagery from which it is derived. And from the standpoint of demographic modeling, PATCH is concerned only with each hexagon's score, whether or not it is suitable for breeding, and who its neighbors are. Information about habitat patterns within each hexagon is not used by the demography module, and is left out of the internal description of the territory maps. Habitat patterns are still preserved in the visual display of territory maps, as long as they are not compressed. Editing territory maps is easy because it only entails modifying individual hexagon scores. The hexagon editing tools are designed to facilitate the generation of temporal sequences of maps that can be used to investigate demographic consequences of landscape change. A territory map can constitute the final product of an analysis, or it may simply be a necessary step towards performing demographic simulations.

Identifying Territories

The territory allocation controls are accessed through a panel in the habitat controls window (Figure 7). This panel allows you to construct a new territory map. It also makes it possible to alter an existing map (if the territory size settings have been updated) and to compress a territory map. If the current working data set consists of either a patch map or a territory map that has been compressed using PATCH's hexagon compression routine, then clicking on the *Build The Territory Map* button simply removes these existing data. Otherwise, clicking on this button always builds a new territory map.

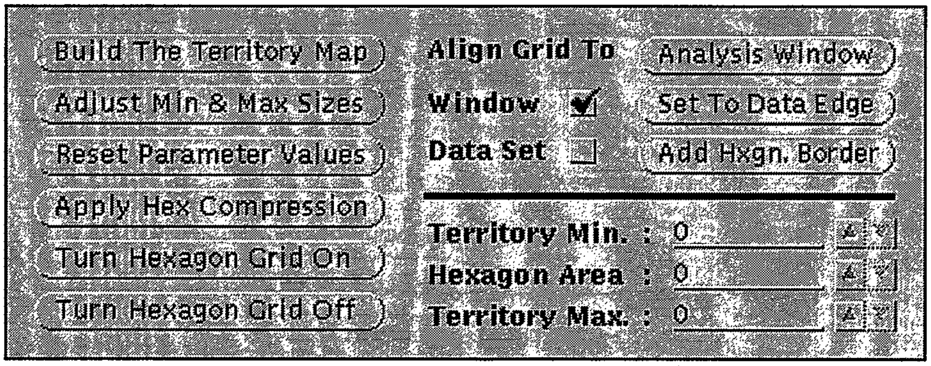


Figure 7. The panel in the habitat controls window that contains the territory allocation routines. See the text for descriptions of the panel's various fields, settings, and buttons.

Using the Zoom-Box

Territory maps are constructed using the subset of the GIS imagery that falls within the zoom-box. If the zoom-box contains a portion of the imagery that is too large to display on the screen at one time, then after constructing the map, it can be reduced and moved around so that the results can be examined a bit at a time.

Displaying the Results

The results of the territory allocation process can be viewed in the analysis window (use the button labeled *Analysis Window* shown in Figure 7). The analysis window is always set to the size of the zoom-box, given the current magnification value. By displaying the analysis and zoom windows side by side, you can compare a territory map directly to the GIS data used in its construction.

The result of the territory allocation process is a map of hexagonal cells displayed in one of three colors, depending on the amount of high-quality habitat in their vicinity. Hexagons that contain no habitat (no pixels with a weighting value greater than zero) are painted black. Hexagons that contain some habitat, but that are not suitable for breeding, are painted gray. Hexagons suitable for breeding are painted green. This visual presentation is designed to help you quickly grasp the character of the results. Each hexagon is also assigned a score, and you should keep in mind that a hexagon's color (black, gray, or green) does not necessarily reveal a lot about its score.

PATCH locates core areas within a territory map just as it does with a patch map. Core areas are defined relative to an edge width that you supply. They are computed on a patch-by-patch basis, since the hexagon boundaries do not typically fall along habitat edges.

Selecting a Hexagon

Once the territory allocation process has been run, individual hexagons can be selected and a few of their properties observed. To select a hexagon, simply middle click on or near it in the analysis window. The hexagon is highlighted (painted white), and its number, area (in pixels of habitat), and score are displayed in the analysis window footer. When the mouse button is released, the hexagon returns to its original color and the analysis window footer displays its standard information.

When a hexagon is selected, the threshold score and the mean distance separating neighboring hexagons are displayed in the footer of the habitat controls window. The mean distance between neighboring hexagons is computed as

$$\frac{(4 \times \text{Horiz}) + (2 \times \text{Diag})}{6}$$

where "Horiz" and "Diag" refer to the horizontal and diagonal distance between neighbors, respectively. You will need to know this mean distance when parameterizing PATCH's life history simulator.

Adjusting Min and Max Sizes

The button labeled *Adjust Min & Max Sizes* (Figure 7) updates the territory map if the territory minimum or maximum value is altered after the map has been constructed. This does not change individual hexagon scores, only their designation as suitable or unsuitable for breeding. When this button is used, the analysis window is repainted.

Aligning the Hexagon Grid

The *Align Grid To* setting allows you to align the hexagon grid to the zoom-box, and hence to the zoom window (the *Window* option), or to the edges of the current territory map (the *Data Set* option). This feature is useful when the dimensions of the zoom-box are not equal to those of the territory map. In such a situation, the hexagon grid should be aligned to the edges of the entire hexagon data set, not to the edges of the zoom-box. But if a new map is to be constructed later using a different zoom-box size or placement, the hexagon grid must then be aligned to the zoom-box. Otherwise, the new map will be constructed using the zoom-box settings from the previous patch or territory map.

Hexagon Borders

The *Set To Data Edge* button sets the zoom-box to exactly the size and placement of the current patch or territory map. The *Add Hxgn Border* button expands the zoom-box slightly beyond the edges of a hexagon data set in order to add a small border around the sides of the map. However, if this feature is used with a patch map, it simply sets the zoom-box to the data edges.

Editing the Hexagons

The PATCH model contains an editor that you can use to alter a territory map (Figure 8). Using this tool, alternative future landscapes can be developed, or "what if" questions addressing the consequences of specific habitat modifications can be pursued. PATCH also makes it possible to randomize the placement of hexagons within a territory map, and you can use this feature to test hypotheses about the importance of a particular orientation of habitat across a landscape. The hexagon editor is accessed by clicking on the button labeled *View Next Panel* in Figure 5, or by right clicking in the panel.

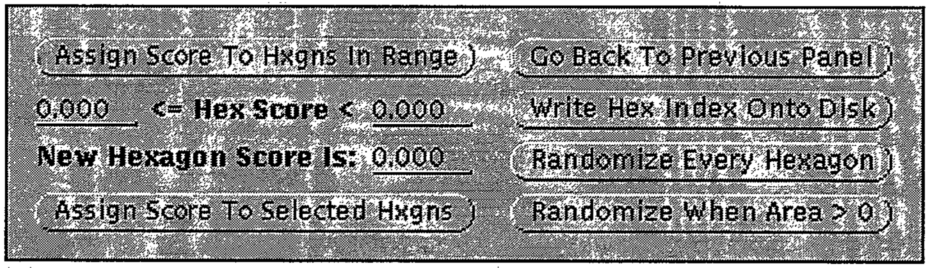


Figure 8. The hexagon editor. The fields and buttons on the left side of the panel are used to modify hexagon scores. The buttons on the right side of the panel are used for randomizing hexagons and for saving results obtained using the hexagon editor.

Hexagon Compression

The hexagon compression routine converts a standard territory map into a new map composed entirely of 12-pixel hexagons. All of the routines involved in running PATCH's demographic simulations work correctly with compressed territory maps. There are three reasons for you to take advantage of this routine.

First, many territory maps cannot be displayed in full on the computer's monitor because they simply extend beyond the screen's borders. If such a map is composed of large territories (much greater than 12 pixels), then the result of running the hexagon compression routine is a new map with dimensions in pixels significantly smaller that those of the uncompressed map. A compressed territory map often fits entirely on the computer screen. The second reason for using the hexagon compression facility is that it is a prerequisite to running the hexagon randomization routines. Finally, when working with large territory maps, it is often desirable to use the compression facility to save disk space. When a territory map is saved to the disk, the size of the file containing the map itself (the ".patch" file) is approximately 4 bytes × # rows × # columns. Saving such files can quickly consume large amounts of hard disk space. If a territory map is compressed first, the demands on storage space can be reduced considerably. The difference will be proportional to the ratio of the compressed hexagon size (12 pixels) to the original hexagon size. Since some territory maps take a long time to build, you might want to first save the original data to the hard disk and compress it using a UNIX compression routine.

When the hexagon compression routine is used, a new territory map is constructed and substituted for the original one. The new map is composed of 12-pixel hexagons, but the scores and breeding status of the new hexagons are exactly the same as their counterparts in the uncompressed map. All habitat pattern is lost in the compressed territory map, and its hexagons are painted entirely black, gray, or green. In compressed territory maps there is no within-hexagon habitat pattern to track. This is why only compressed maps can be randomized.

When a territory map is compressed, the one-to-one spatial correspondence between the GIS habitat coverage and the territory map is lost. To avoid creating misleading visual cues, the zoom window is disabled when a territory map is compressed. This also applies when a compressed territory map is loaded into the model. The image window and zoom-box still work when a compressed territory map is used.

PATCH can be taken out of this mode by clicking the *Build The Territory Map* button. This removes the compressed territory map from PATCH's memory and reestablishes the one-to-one correspondence between the zoom and analysis windows. A second click builds a new uncompressed territory map. When the *Build The Territory Map* button is used to remove a compressed territory map, the zoom-box size and location, the legend weights, and the hexagon and territory sizes are all set to the values used to construct the original uncompressed map.

Running the hexagon compression routine on a 12-pixel territory map is allowed in spite of its having no advantage from a display or storage standpoint. This is permitted because only compressed territory maps can be used with the hexagon randomization routines. It can also be easier to see some of the patterns in compressed territory maps since every hexagon is painted a solid color. When a territory map is compressed, the hexagon grid conforms to the pixel boundaries. This is intended to serve as a visual reminder that the data are compressed.

Changing the Resample Rate

PATCH contains a mechanism, the *Resample Rate* parameter, that you can use to split data pixels into finer units. This feature is useful primarily when the desired territory size cannot be obtained using the original pixel dimensions. If the resample rate is set to n (an arbitrary integer), every pixel in the GIS data set is split into an $n \times n$ array of pixels. The *Resample Rate* parameter is located in the main window.

Chapter 6 The Movement Module

Introduction

Three different movement routines are available within the PATCH model. In addition, three choices are available for defining site fidelity behavior (the likelihood that an individual remains on a breeding site from one year to the next). The options for simulating movement include a directed random walk, selection of the best available site within a search radius, and selection of the closest available site within a search radius. These movement routines require that you specify a minimum and a maximum movement ability, in terms of the total number of steps that can be taken from a hexagon to one of its six neighbors. It is also necessary to define just how random the movement is when a random walk is taken. The degree of randomness is determined by parameters that control the amount of linearity and the attraction to high quality habitat exhibited during random walks. The options for site fidelity are termed high, medium, and low. High site fidelity implies that organisms possessing territories never relinquish them. Low site fidelity dictates that every territorial individual must search yearly for a new breeding site. If site fidelity is set to medium, then the decision regarding whether or not to remain on a territory is made based upon the quality of the habitat at that site. This chapter discusses all of these features.

There is no mortality associated with movement in PATCH. This was done because the data necessary to link mortality to movement is almost never available. However, PATCH does link survival to habitat quality, and individuals dispersing into poor habitats will experience lower survival rates than those settling in high quality sites. If population-wide estimates of dispersal mortality exist, you can always approximate this loss by lowering a species' fecundity values.

Movement Behavior

The movement routine is called twice per year. It is used in the fall to control the dispersal of the year's new recruits (referred to as *young of the year* or *juveniles*) and in the spring to drive the movements of older organisms (referred to as *adults*) just prior to the breeding pulse. The implementation of the movement routine differs slightly depending on whether juveniles or adults are involved. Every juvenile is obliged to disperse from its natal site, whereas adults may or may not elect to move. Decisions regarding adult movement are made based upon whether the individual is territorial or a floater, the quality of habitat currently being occupied (for territorial individuals), and the site fidelity parameter. The success of the dispersal process is evidenced in part by the number of floaters in a population. PATCH keeps track of the number of floaters and reports this information on a yearly basis.

The edges of PATCH's territory maps act as reflecting boundaries. In addition, you can force any nonbreeding hexagon to behave as a reflecting boundary. This can be done on a hexagon by hexagon basis, but PATCH also has a routine that performs this function automatically along edges present in the landscape. This facility is designed to help you restrict movement paths in the presence of large bodies of water, mountain ranges, etc. The PATCH model does not make use of wrapping or absorbing boundaries.

Every juvenile and every floater is required to undergo a yearly search for a territory. Further, because the minimum movement distance is always at least one hexagon, individuals who qualify for movement are obligated to actually do so. The only exception to this rule is that floaters are allowed to take over the site at which they are presently located in the event that the owner of the territory dies or leaves. (This is technically possible for juveniles too, but it never happens because there is no opportunity for the parent to move or die in the period separating the breeding pulse and the juvenile dispersal event.) Regardless of the movement behavior being specified, individuals are not allowed to settle until they have gone at least the minimum distance you have specified. Throughout the movement process, reflecting boundaries are respected and movement paths are adjusted accordingly. Movement paths are not affected by the presence of other individuals.

As a whole, this scheme provides you with the flexibility necessary to apply the model to a variety of organisms using only a small number of parameters. Figure 9 depicts the sequence of decisions that are made when the movement routine is called.

Random Walks

When a random walk is used for the movement routine, individual organisms take a series of steps, from the hexagon currently occupied, to one of its six neighbors. The direction of the walk is influenced by the *Search Randomly* and *Walk Up Gradient* parameters. Individual's tendency to move up gradient is in turn influenced by the quality of the habitat within which the movement is taking place. Before taking a step, individuals performing a random walk first look to see if a neighboring hexagon is both suitable for breeding and available. If so, they determine whether or not to settle in that site. This decision is based upon the site's quality and on the distance already moved. The better the quality of the site, the more likely an individual is to settle in it. At the same time, individuals become less selective as their ability to

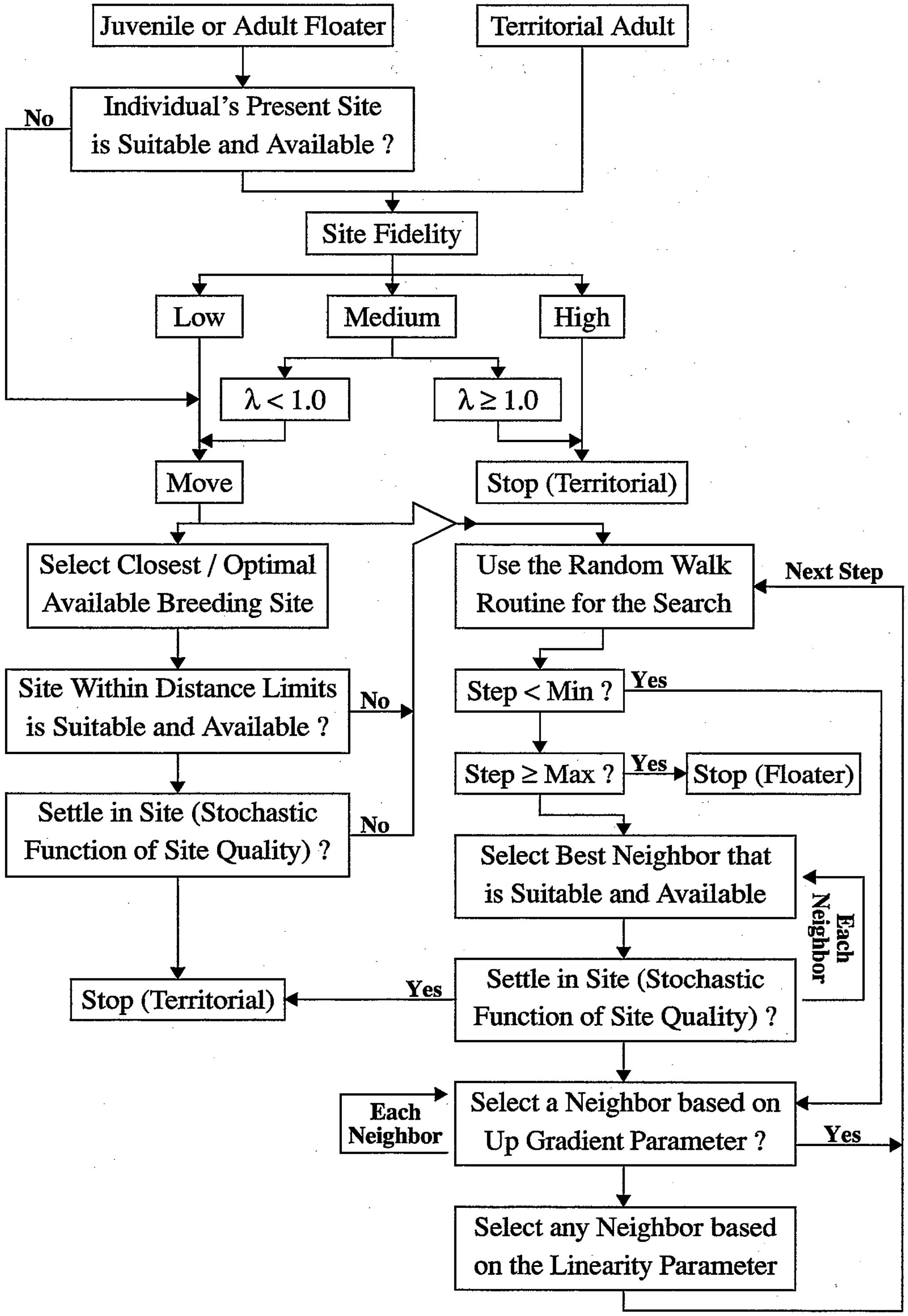


Figure 9. A schematic of the movement routine. The term "juvenile" refers to individuals born in the previous breeding pulse. The symbol λ denotes the dominant eigenvalue of the projection matrix associated with a hexagon. The random walk routine's "Settle in Site" box is passed over if there are no available breeding sites to be colonized. Notice that a path connects the exhaustive search strategies (closest and optimal) to the random walk routine.

continue searching diminishes. If more than one neighbor is suitable for breeding and unoccupied, the highest quality site is considered first for settling. Ties are handled randomly.

PATCH uses a settling probability to determine whether an individual taking a random walk should stop and occupy an available breeding site. The settling probability is computed based on the quality of the target hexagon, the maximum possible site quality, the distance already moved, and the maximum movement distance. A random number between zero and one is then selected, and if the random number is less than the settling probability, the site is colonized. Otherwise it is passed up and the individual continues to search. The settling probability is

$$P_{\text{settling}} = \frac{\text{hexagon score}}{\text{max score}} \times \left(1 - \frac{\text{step number}}{\text{max steps}}\right) + \frac{\text{step number}}{\text{max steps}}$$

Early in a random walk, a searcher's probability of colonizing an available breeding site is determined almost entirely by the breeding sites scores. But the likelihood of settling increases (eventually to 100%), as the step number approaches the maximum number of steps, regardless of breeding site quality. In the absence of unoccupied breeding habitat, individuals taking a random walk select a neighbor in a more or less random fashion. The extent to which the choice of neighbors is made at random is influenced by a general tendency to move towards, but not necessarily to remain in, higher quality habitats (the *Walk Up Gradient* parameter), and by the amount of linearity built into the movement paths (the *Search Randomly* parameter).

The probability that an individual elects to move up gradient from lower to higher quality habitat is proportional to the difference between the score of the currently occupied hexagon and the neighbor being considered. A decision to move up gradient is made if

$$r < \Gamma(x, z),$$

where

$$\Gamma(x, z) = \begin{cases} x \cdot z/50 & : \quad 0 \le z \le 50 \\ 50 \cdot x/(100 - z) & : \quad 50 < z < 100 \\ 1.0 & : \quad \text{otherwise,} \end{cases}$$

and

$$r = \text{a random number between 0 and 1}$$

$$x = \frac{\text{(neighboring score)} - \text{(current score)}}{\text{maximum score}}$$

$$z = \text{the Move Up Gradient parameter}$$

Thus, the attraction to a neighboring site grows stronger as the difference increases between its score and that of the site presently occupied. This process always considers a hexagon's highest quality neighbors first so that individuals are most likely to move into the best adjacent site. The equations governing up gradient movement generate a family of curves. Each curve corresponds to a value of the *Move Up Gradient* parameter (Figure 10). Individuals never elect to move up gradient when this parameter is set to zero. The probability that an individual moves up gradient increases smoothly as the *Move Up Gradient* parameter is increased (Figure 10). Individuals always move up gradient when the parameter is set to 100%.

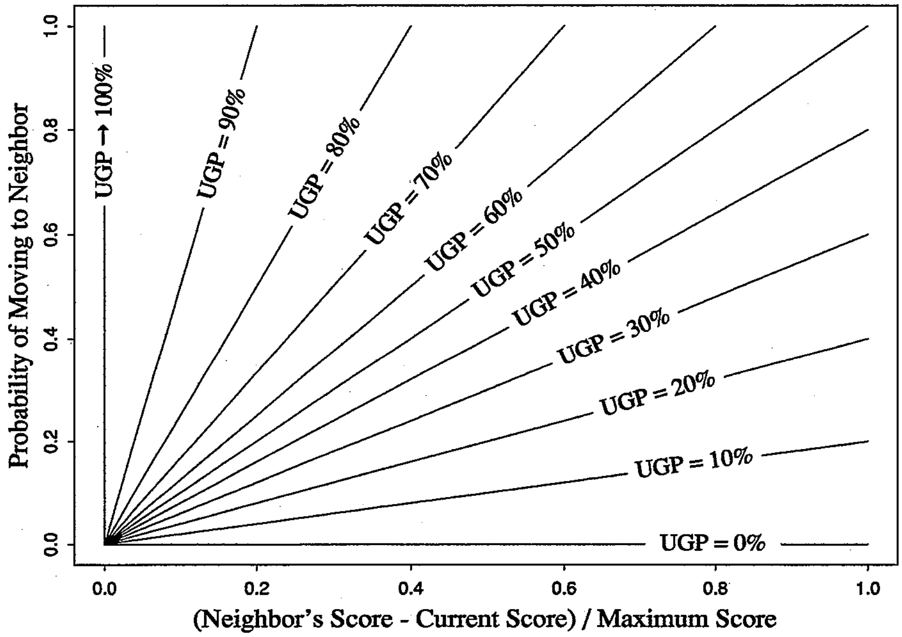


Figure 10. The family of curves corresponding to different values of the *Move Up Gradient* parameter (UGP). The horizontal axis measures the relative increase in quality associated with movement to a specified neighbor. The vertical axis represents the probability that an individuals decides to make such a move.

If an individual does not elect to move up gradient, then the choice of movement direction is made as randomly as possible within the constraints imposed by the *Search Randomly* parameter and given the presence of any reflecting boundaries.

Walk Linearity

The amount of linearity in a random walk is controlled by the parameter labeled *Search Randomly.* This parameter takes on values between 0 and 100%. When *Search Randomly* is set to 100%, the choice of which of a hexagon's six neighbors to move into (Figure 11) is made completely randomly (in the absence of reflecting boundaries and decisions to move up gradient). When this parameter is set to zero, an individuals always moves in the direction of the previous step (again, in the absence of boundaries and up gradient motion). The resulting movement paths then become completely linear.

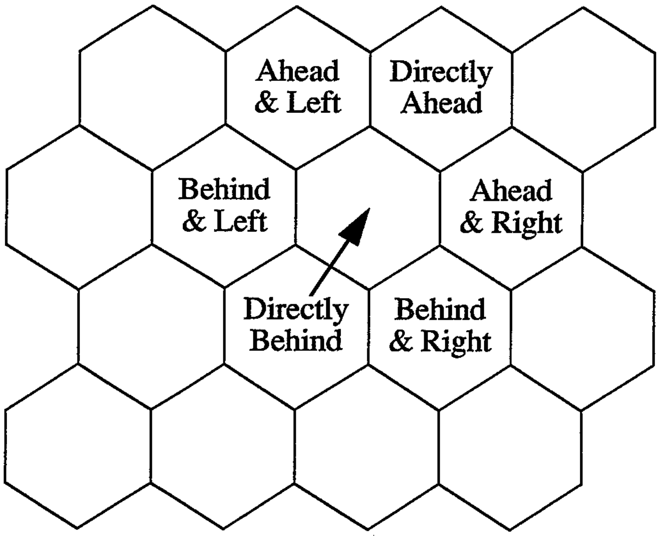


Figure 11. The naming conventions that PATCH uses to describe the movement choices available to individuals in motion. The direction of the previous step is indicated by the arrow. The labeled hexagons indicate the directions corresponding to the six movement choices that can be made in the subsequent move. The abbreviations used for these directions, moving clockwise from "Directly Ahead", are: DA, AR, BR, DB, BL, and AL.

The influence of the *Search Randomly* parameter on a searching individual's movement path is mediated through a series of probabilities that govern the likelihood of moving in each of the different available directions. Let the six possible movement directions be referred to as directly ahead (DA), ahead and left or right (AL, AR), behind and left or right (BL, BR), and directly behind (DB). These naming conventions are always relative to the previous move (Figure 11). Each time an individual in motion prepares to take a step, probabilities are assigned to each of the six possible directions, and this in turn governs the likelihood of each neighbor being selected. The rules used for assigning these probabilities are:

$$P_{AL} = P_{AR} = \frac{x(2-x)^{2.467}}{6},$$

$$P_{BL} = P_{BR} = \frac{x^2}{6},$$

$$P_{DB} = \frac{x^4}{6},$$

and $$P_{DA} = 1 - P_{AL} - P_{AR} - P_{BL} - P_{BL} - P_{DB},$$

where $$x = \frac{\text{the Search Randomly parameter}}{100\%}.$$

The probabilities P_{DA}, P_{DB}, P_{AL}, P_{AR}, P_{BL}, and P_{BR}, are all smoothly varying curves that together sum to one, as long as the *Search Randomly* parameter is held between zero and 100% (Figure 12). When the *Search Randomly* parameter is set to 100%, each of the six possible movement directions has an equal probability of being chosen. As this parameter decreases, the direction selected becomes steadily more biased toward forward movement (Figure 12). When the *Search Randomly* parameter is set to 50%, $P_{DA} = P_{AL} + P_{AR}$ (hence the odd exponent assigned to P_{AL} and P_{AR}, above). When the *Search Randomly* parameter is set to zero, searchers not interacting with a reflecting boundary or moving up gradient always elect to travel directly ahead.

The behavior of the *Search Randomly* parameter can also be characterized by its affect on the observed diffusion rate in the absence of influences due to habitat quality or reflecting boundaries. Figure 13 displays the relative diffusion rate of individuals as a function of the *Search Randomly* parameter. Relative diffusion rate is measured here as an individual's displacement from its starting point divided by the total distance moved. The data points used to construct the curve in Figure 13 were generated from a sample of 100 individuals each taking exactly 100 steps without reflecting boundaries or up gradient motion.

Nonrandom Movement

When individuals searching for breeding sites are instructed to select the best or closest available site within a search radius, the movement process takes on an appearance different from that of a random walk. The search radius becomes the annulus, centered on the current site, defined by the minimum and maximum movement abilities. If the best site is to be selected, the quality and availability of every hexagon within the search radius is examined. If the closest available site is to be

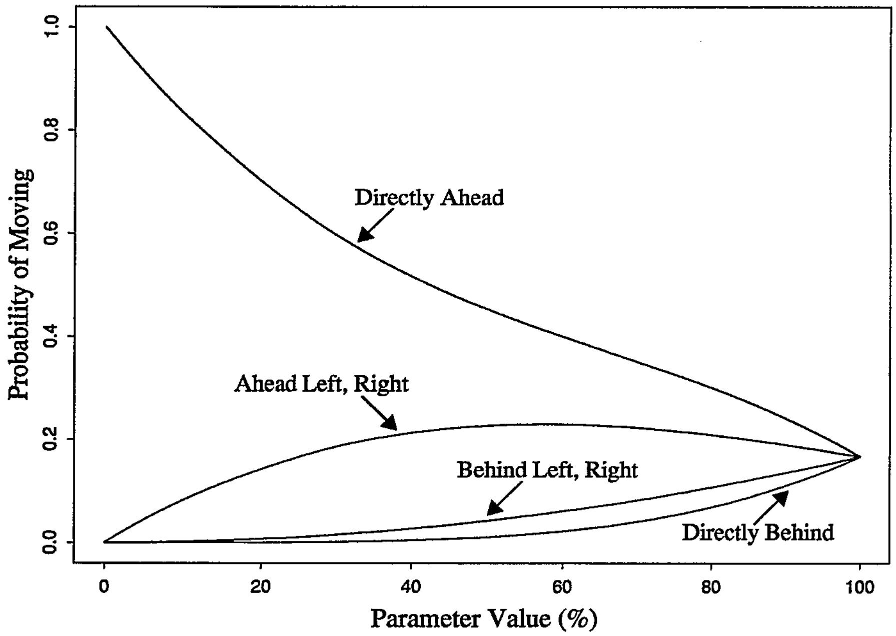


Figure 12. The influence of the *Search Randomly* parameter on the probabilities governing choice of movement direction. The probability of moving ahead (behind) and left equals the probability of moving ahead (behind) and right. When the *Search Randomly* parameter is set to 50%, the probability of moving directly ahead equals the sum of the probabilities of moving ahead to the left and ahead to the right.

selected, then the search radius is expanded iteratively from the minimum to the maximum until a suitable hexagon is located. In either case, if more than one suitable site is found at the same distance, then the best site is always taken. If two equidistant sites have the same score, the decision of which to occupy is made randomly. Reflecting boundaries are always respected, and if the searching individual is unable to locate a suitable site, a random walk must be taken. For this reason, the *Search Randomly* and *Walk Up Gradient* parameters should to be supplied even if a random walk is not being explicitly used.

Site Fidelity

The behavior of PATCH's movement algorithm is also controlled by the site fidelity parameter. Site fidelity governs the probability that a territorial adult (or a floater poised to take over a site who's owner has died) elects to abandon its breeding site in search of one of higher quality. If the site fidelity parameter is low, then every adult abandons its territory (if it holds one) and searches for another site. If site fidelity is high, individuals holding territories remain on them indefinitely.

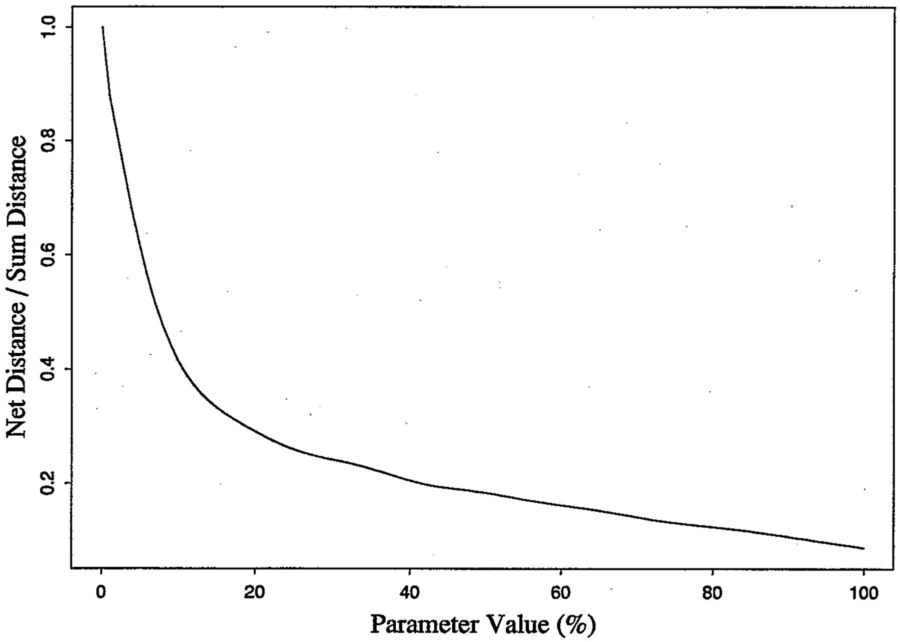


Figure 13. The relative diffusion rate as a function of the Search Randomly parameter. The net movement distance is the straight line distance connecting the start and end of an individual's movement path. The sum movement distance is the actual length of the movement path, measures as a series of jumps between hexagons. The data points used to construct this curve were generated from a sample of 100 individuals each taking exactly 100 steps in the absence of reflecting boundaries or up gradient motion.

When the site fidelity parameter is set to medium, individuals move away from demographic sinks, but remain on demographic sources. In this context, demographic sinks and sources refer to sites having dominant eigenvalues (λ values) that are less than 1.0 and greater than or equal to 1.0, respectively.

Reflecting Boundaries

Individuals in motion reflect off the edges of PATCH's territory maps. In addition, you can force any nonbreeding hexagon to behave as a reflecting boundary. This feature allows you to prohibit movement beyond a coastline or over a mountain range, etc. You can enter reflecting boundaries by hand (using the life history window's *Movement Limits* button) or they can be located automatically at interfaces present within the landscape (using the life history window's *Automatic Limits* button). The PATCH model does not allow you to insert wrapping or absorbing boundaries.

Chapter 7 The Life History Module

Introduction

The life history simulator is the centerpiece of the PATCH model. It is designed for use with a range of wildlife species. The life cycle is modeled as a series of discrete events taking place on a yearly basis. Each year a cohort of new recruits is generated during a breeding pulse. PATCH has a post-breeding census, so these new recruits (subsequently referred to as *juveniles*) are not of breeding age when they are initially censused. All of the older (and potentially reproductive) individuals are referred to for simplicity as *adults*. PATCH is a single-sex model, and as such it cannot simulate an Allee effect. For this reason, PATCH may underestimate the risk experienced by populations that are subject to periods of extremely low density.

PATCH's life cycle begins in the winter with an assessment of the year's mortality. Next comes spring and the optional movement of adults, which is followed by a summer breeding pulse. Juveniles disperse in the autumn, and finally the census is taken. The cycle then repeats starting with the winter of the following year. For simplicity, all mortality is collapsed into the yearly survival event. There is no mortality associated with the movement process. As mentioned above, the timing of events in PATCH's life history simulator is designed to mimic a projection matrix with a post-breeding census. However, the census is placed after the dispersal event so that juveniles are included in the counts of floaters and breeders. The absence of mortality in the movement process ensures that the census still accurately records the number of new recruits. PATCH's life cycle is displayed in Figure 14.

PATCH's life history simulator was designed with the intent that it would carefully incorporate spatial pattern into its predictions, while being as parsimonious as possible. None the less, you may not have the data necessary to supply all of the parameters with confidence, particularly the interpolation functions, the vital rates factor, and the parameters controlling random walk behaviors. But all of these parameters emerge directly from the spatially explicit nature of the model. If, for example, the interpolation routines were not selectable from the model interface, PATCH's connection to GIS maps would dictate that their functionality still existed in some form. By providing you with control over these parameters, instead of simply hard-wiring them into the model code, PATCH makes it possible for you to conduct sensitivity analyses and test the various assumptions that you will inevitably end up making.

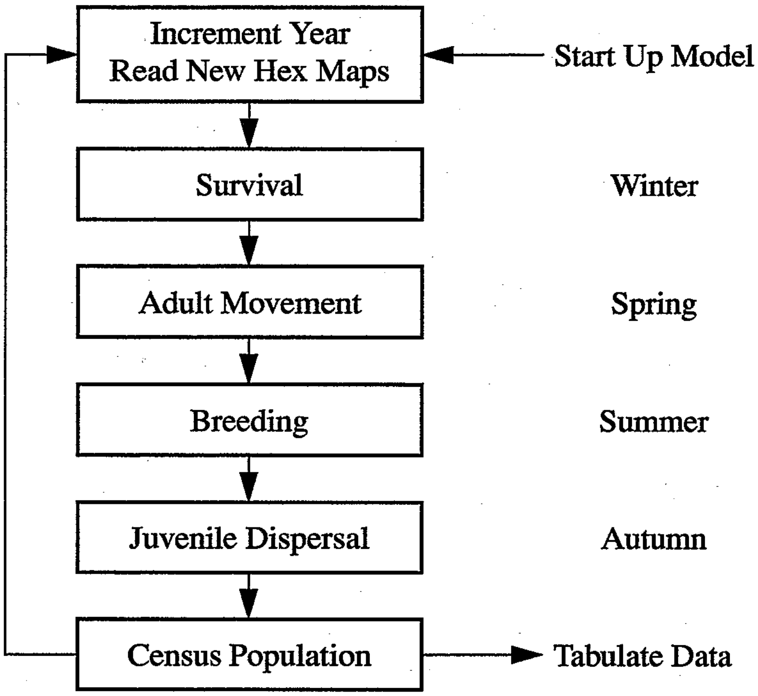


Figure 14. The timing of events in the life history simulator. The census is placed between the breeding pulse and the survival routine so that it records the actual number of new recruits to the population. The dispersal event is placed before the census so that juveniles are included in the counts of floaters and breeders. The movement routines do not subject individuals to any additional mortality.

Survival and Reproduction

Survival and reproductive rates are loaded into PATCH in the form of a projection matrix. If an entire landscape consists of optimal habitat, and if breeding sites are unlimited, then PATCH's projected population sizes will agree almost perfectly with those obtained using a matrix model. However, to the extent that high-quality habitat is limiting, PATCH's results will differ from those of a projection matrix. Differences arise because PATCH is individual based, spatially explicit, and because its survival, reproduction, and movement modules all incorporate some stochasticity. Decisions regarding survival, reproduction, and movement behavior are made on an individual basis using a random number generator. These events can be significantly influenced by the quality of habitat present at an animal's current location. Variability in the survival and reproductive rates resulting from the use of a random number generator approximates demographic stochasticity. PATCH also allows you to incorporate environmental stochasticity into its simulations.

Linking to Habitat Quality

PATCH does not assume that survival and reproductive rates scale linearly with habitat quality. Instead, you are provided with a set of six generic functions (linear, logistic, concave, convex, constant, and piecewise constant) that can be used to compute species' vital rates from habitat quality. But PATCH cannot carry out these computations unless you also tell it what quality of habitat is to be associated with the vital rates supplied as input parameters. That is, PATCH needs to know if the vital rates in its input matrix are to be realized in the best habitat, or the worst, or some intermediate value. This information is brought to PATCH by the *Vital Rates Factor* parameter. Of course, you also need to select specific survival and reproduction interpolation functions. The six interpolation functions are described mathematically in Table 2. In each equation, hexagon quality is introduced through the relation

$$x \equiv \frac{\text{Hexagon Score}}{\text{Maximum Hexagon Score}}.$$

The value of each interpolation function is represented in Table 2 by the symbol "y". The minimum survival and reproductive rates always equal zero, and they are realized in hexagons having scores of zero. The maximum survival and reproductive rates are realized in hexagons having scores equal to the largest weighting value specified in the legend. These maximum vital rates can be calculated from the input data using the relationship

$$\text{maximum rate} = \frac{\text{vital rate entered into matrix}}{\text{INTERPOLATE}\left\{\dfrac{\text{Vital Rates Factor}}{100\%}\right\}}$$

where "INTERPOLATE" is replaced with the appropriate interpolation function from Table 2. The argument of the interpolation function represents the extent (expressed as a number between zero and one) to which the best possible hexagon score exceeds the score associated with the vital rates entered into the matrix. For example, you might specify that the yearly adult survival probability is 0.60 in hexagons that have scores equal to 75% of the highest possible value (i.e., vital rates factor = 75). If the interpolation function is linear, then the value of adult survival ranges between zero, in hexagons having scores of zero, and

$$\frac{0.60}{\text{LINEAR}\{0.75\}} = \frac{0.60}{0.75} = 0.80$$

in hexagons having the best possible score. However, if the interpolation function is concave, then this maximum value equals

$$\frac{0.60}{\text{CONCAVE}\{0.75\}} = \frac{0.60}{(0.75)^3} = 1.44.$$

Because this represents a survival rate greater than one, PATCH would refuse to run until you either lowered the matrix value for adult survival, raised the quality of habitat associated with the input matrix, or changed the interpolation function used for survival. To make this sort of assessment easier, PATCH displays the maximum vital rates any time you middle click in the Life History Parameters window.

Table 2: The Interpolation Functions

Name	Mathematical Description
Linear	$y = x$
Logistic	$y = \frac{1}{1 + e^{14\left(\frac{1}{2} - x\right)}}$
Concave	$y = x^3$
Convex	$y = 1 - (1 - x)^3$
Constant	$y = 1$
Piecewise Constant	$y = 0 \quad x \in [0, 1/3)$ $y = 1/2 \quad x \in [1/3, 2/3]$ $y = 1 \quad x \in (2/3, 1]$

The above rules governing survival probabilities are modified slightly for floaters. Floaters are adults or post-dispersal juveniles that do not possess a territory. By definition, floater reproductive rates are fixed at zero. The survival rates for floaters are computed the same as for territorial individuals, except that floaters' maximum survival rate is capped at the threshold score. The threshold score is given by:

$$\text{threshold score} = \text{max weighting value} \times \frac{\text{min territory size}}{\text{hexagon size}}.$$

The logic behind this is simple. Floaters in the vicinity of high-quality sites occupied by territorial individuals cannot be expected to reap the same benefits from these habitats (i.e., they should have higher mortality) as those who are breeding there. At the same time, floaters in habitats that cannot support breeding do not necessarily experience intra-specific competition, and in those situations it might be inappropriate to lower their survival rates. The threshold score is the break-point between habitat that can and cannot support breeding (in the absence of habitat sharing between sites) and as such it provides a natural vehicle for fixing an upper bound on the survival rate of floaters.

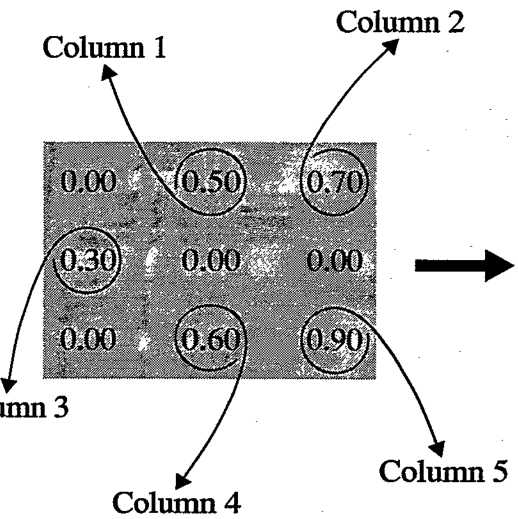
Demographic Stochasticity

Life history simulations conducted with PATCH always incorporate demographic stochasticity, the random fluctuations in survival and reproductive rates that occur between individuals. This results from the model's use of a random number generator to evaluate the outcome of specific survival and reproduction decisions. In the case of survival, a uniform random number between zero and one is selected. If this number is less than the sum of the probabilities of making a transition between the current stage (or age) class and every other class, then the individual dies. Reproduction is stochastic only to the extent that a random number is selected to force the brood sizes take on integer values. Let the expected rate of reproduction for a particular hexagon be represented by the real number x. Refer to brood sizes equal to the closest integers less than x and greater than x as n^- and n^+, respectively. The probability of producing a brood of size n^- is set to $n^+ - x$. Similarly, the probability of producing a brood of size n^+ equals $x - n^-$. Only broods of these two sizes exist as possible outcomes for individuals breeding in this hexagon. This scheme for reproduction is simplistic, but it only a single parameter per stage (age) class.

Environmental Stochasticity

Changes to the mean survival and reproductive rates resulting from fluctuations in environmental conditions (good and bad years, catastrophes, etc.) can be assumed to influence every member of a population in a similar way. This type of variability is referred to as environmental stochasticity, and it frequently plays a significant role in the dynamics of wildlife populations. PATCH allows you to build environmental stochasticity into its demographic simulations. You accomplish this by constructing a table of vital rates that reflect the level of environmental stochasticity to be simulated. PATCH can upload the data in this table as long as the data are formatted correctly and contained in a file that has an appropriate name. Each column of the table signifies a specific survival or reproduction rate. Every row of the table must contain the entire set of survival and reproductive rates necessary to construct a projection matrix.

Matrix elements set to zero in the model interface cannot be represented in the table of vital rates, and as such they must remain zero indefinitely. Each year the model is run, a row from the table is selected at random and used to fill the projection matrix for the species being simulated. Thus, you can include any amount of correlation between age or stage classes by constructing the table's rows appropriately. The name of the file holding this table has to be built up from two parts. The first part must be identical to the name of a data file containing the other demographic parameters describing the species. The second part must contain the string ".random". Figure 15 provides an illustration of the format that PATCH expects for tables of vital rates.



0.42	0.62	0.22	0.52	0.82
0.44	0.64	0.24	0.54	0.84
0.46	0.66	0.26	0.56	0.86
0.48	0.68	0.28	0.58	0.88
0.50	0.70	0.30	0.60	0.90
0.52	0.72	0.32	0.62	0.92
0.54	0.74	0.34	0.64	0.94
0.56	0.76	0.36	0.66	0.96
0.58	0.78	0.38	0.68	0.98
		⋮		

Figure 15. The data format for entering environmental stochasticity into the PATCH model. Each nonzero entry in the projection matrix (shown on the left) defines a column of the data table (shown on the right). The columns in the data table are numbered in increasing order from left to right. A row from the data table is selected randomly at the start of each year, and is used to define a new projection matrix to govern that year's survival and reproduction decisions. You can create as many rows as desired, but the survival values within a single column of any potential matrix must never sum to more than one. This calculation includes the influence of the vital rates factor.

Miscellaneous Parameters

Other than specifying the model organism's movement behavior, you must provide only a few remaining parameters. It is necessary to specify the duration of each simulation in years, and the number of replicate simulations to be conducted. PATCH must also be told where to locate the initial population of organisms, and what age or stage class to assign to them. You can also make specific nonbreeding hexagons function as reflecting boundaries. Lastly, a transient period can be established during which information about the emigration and immigration into breeding sites is not to be compiled.

Running a Simulation

The PATCH model can be used to conduct demographic simulations after a territory map has been installed and the parameters in the life history windows have been specified. A simulation is started by clicking the *Run Simulations* button in the bottom panel of the life history window (Figure 16). If the visualization toggle is on (Figure 17), and if the analysis window is up to date, then the simulation is displayed on the screen as it progresses. It is possible for one simulation to start up at the point where the previous one left off, but this can be done only when the number of runs is set to one, and only if the *Tally Utility From* field is set to zero. Otherwise, the run number and year are automatically reset to zero at the start of each new simulation. Altering any model parameter except the number of years or the visualization setting causes PATCH to reset the run number and year to zero. This design allows you to run the model for a large number of years with the visualization check-box turned off, and then subsequently for a few more years with the visualization on.

The *Write Parameters* button (Figure 16) can be used to write a copy of the current set of demographic parameters to the computer's monitor, or to a file. The *Read Parameters* button loads demographic data from a file and sets the parameters in the life history windows. This utility can be used to read output generated from the *Write Parameters* button, or it can read output from a previous run of the model. This works because every time a demographic simulation is conducted, the model begins by printing out the full set of life history parameters currently being used.

While a simulation is being conducted, PATCH tracks the number of individuals in each age or stage class, the number of floaters and breeders, and the mean values of both the net and sum dispersal distances (for juveniles only). The net dispersal distance is the length of the straight line connecting the starting and ending points on the movement path. The sum dispersal distance is the actual distance moved, measured as a series of straight lines from one hexagon center to another. PATCH also tracks the effective survival and reproductive rates and it provides you with a year-by-year record of the dominant

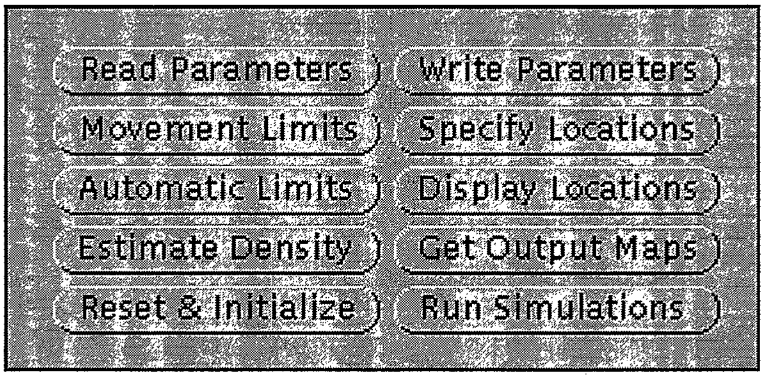


Figure 16. The panel in the life history window that is used to control PATCH's demographic simulations.

eigenvalue (the anticipated steady-state growth rate, or λ) of the projection matrix derived from these vital rates. An example of the data generated by running the model with a two-stage projection matrix is provided in Figure 18. The results shown are averages taken over 10 replicate runs. Only part of the model output is shown in Figure 18.

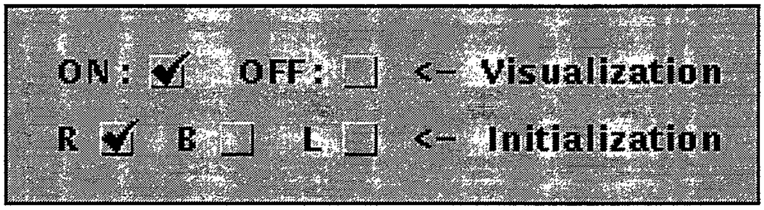


Figure 17. The panel in the life history window that controls the visualization and initialization options. The *R, B,* and *L* refer to the random, best, and locked options of the routine used to locate the initial breeding population within the landscape.

Life History Output Files

If an output file name has been specified in the life history window, then model results are written to this location. If a file name has not been provided, then the output is sent to the monitor. The output data include a run-by-run, year-by-year summary of the population. If an output file has been provided, and if the number of runs is greater than one or the *Track Utility From* parameter is not zero, a binary output file ending in ".stats0" is generated. This file contains the data on occupancy rates. If the *Track Utility From* parameter is not zero, two more text files are also generated. One of these ends in ".stats1", and contains the means and standard deviations of various measures of the population computed from the collection of replicate runs. The second file ends in ".stats2" and stores occupancy rates and source-sink data.

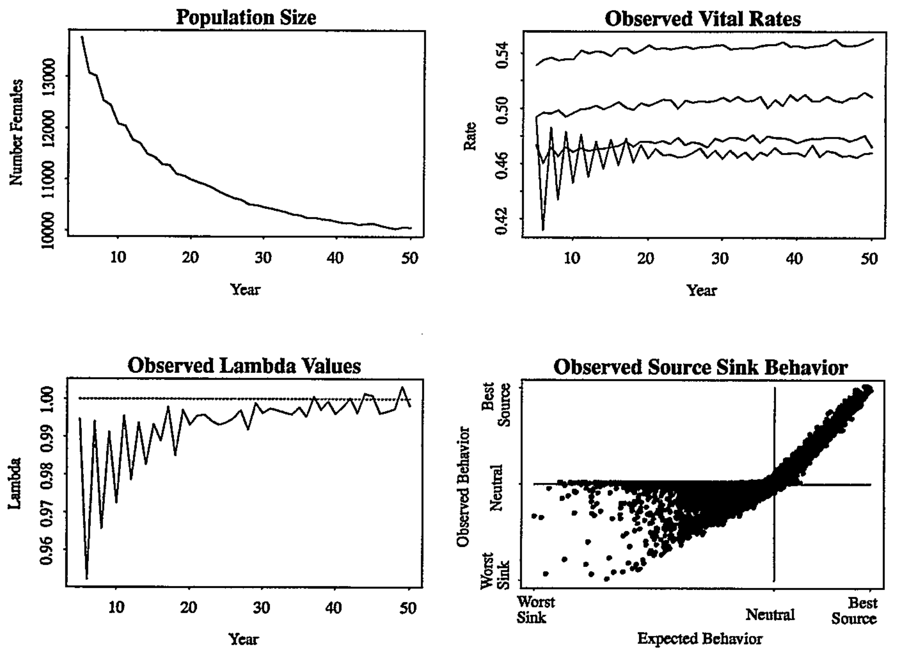


Figure 18. Example output, averaged over 10 replicate simulations, generated using a two-stage projection matrix. Each replicate simulation was run for 1000 years, but only years 5 through 50 are shown in the figure. Every breeding site was initially filled, and the population is shown declining to a stable size. The source-sink data are averages for years 500-1000. PATCH does not create these graphs, it only generates the data that can be used to construct them.

Three additional output files containing occupancy rate and source-sink information can also be generated by clicking on the *Get Output Maps* button in Figure 16. The first of these ends in ".image0" and it displays the observed occupancy rates. These occupancy rates are gathered from the final year of a model simulation, and are summed across every replicate model run. Each breeding site is assigned an occupancy register that is incremented by one if the site is occupied at the end of a given replicate run. When the occupancy rate map is generated, the occupancy register totals are collected and rescaled to lie between zero and ten. The more frequently occupied sites are assigned the higher values. Breeding sites that were used the least are displayed in red in the occupancy maps. The sites that were used the most are displayed in blue. Intermediate rates of occupancy are displayed using a gradient of colors that shift smoothly from red to blue.

The second output map ends in ".image1", and it displays the expected source-sink properties of each breeding site. This image is constructed directly from the territory map and the survival and reproductive rates supplied. It does not incorporate any results from a simulation. Instead, it provides an anticipated source-sink distribution for the landscape. In contrast, the third output map (ending in ".image2") displays the observed source-sink properties that actually emerged from a simulation. In both cases, the source-sink evaluations are made only for the breeding sites present in a landscape. The expected source sink map is built from the dominant eigenvalues (λ's) of the population projection matrices associated with each breeding site, given its score. The observed source-sink map is generated from a tally of the immigration and emigration events observed at each breeding site. The actual data values used to generate these maps are stored in a text file (the ".stats2" file) that is automatically generated by the model.

The expected and observed source-sink maps are constructed by breaking the original continuous source-sink data into discrete color classes. The maps are designed to distinguish between sources, sinks, neutral sites (neither a source nor a sink), and sites that are unsuitable for breeding. Sources and sinks are further broken down into 100 classes each. Every one of these classes represents an interval, one percent in width, within the observed range for all sources or all sinks. The scheme used to construct these images is illustrated in Table 3. Initially, all of the sources are displayed in green, and all of the sinks are displayed in red. But you can easily change these color schemes and identify sources and sinks based on their severity. A new copy of the expected source-sink map can always be generated by loading a territory map, entering the vital rates, the vital rates factor, the survival and reproduction interpolation functions, and then clicking the *Get Output Maps* button. A new copy of the observed source-sink map can be generated by loading a territory map and a demographic output data set that includes a ".stats2" file. Once PATCH has uploaded the data contained in the ".stats2" file, you can click the *Get Output Maps* button to recreate an observed source-sink map.

Using Movement Limits

PATCH allows you to insert reflecting boundaries into a territory map. These boundaries are defined on a hexagon-by-hexagon basis, and are used to prevent movement across bodies of water or high mountains, etc. You can start this process by clicking on the *Movement Limits* button (Figure 16), but this only works if the analysis window is up to date. Once the button has been clicked, and after the mouse pointer has been moved into the analysis window, reflecting boundaries can be inserted by clicking on any hexagon that is painted either black or gray. The assignment of boundaries works like a toggle, hence a second click removes a boundary that has previously been added. This process ends when the mouse pointer is removed from the analysis window.

Table 3: Source-Sink Map Structure

Pixel	Interpretation
0	1.00 × Worst Sink ≥ Value > 0.99 × Worst Sink
1	0.99 × Worst Sink ≥ Value > 0.98 × Worst Sink
2	0.98 × Worst Sink ≥ Value > 0.97 × Worst Sink
⋮	⋮
98	0.02 × Worst Sink ≥ Value > 0.01 × Worst Sink
99	0.01 × Worst Sink ≥ Value > 0.00 × Worst Sink
100	Neither Source Or Sink
101	0.00 × Best Source < Value ≤ 0.01 × Best Source
102	0.01 × Best Source < Value ≤ 0.02 × Best Source
⋮	⋮
198	0.97 × Best Source < Value ≤ 0.98 × Best Source
199	0.98 × Best Source < Value ≤ 0.99 × Best Source
200	0.99 × Best Source < Value ≤ 1.00 × Best Source
201	Hexagons That Do Not Contain Any Habitat
202	Non-Breeding Hexagons Containing Habitat

The life history window also has a button labeled *Automatic Limits* that inserts reflecting boundaries into the landscape for you. Recall that hexagons with no habitat are painted black. Hexagons with habitat are painted green if they are suitable for breeding, and gray otherwise. If no reflecting boundaries are present, clicking the *Automatic Limits* button initially places a boundary at every black hexagon that has a gray or green neighbor. Clicking on this button a second time places a boundary at every gray hexagon that has a neighbor has a black or green neighbor. A third click locates a boundary at every site that qualified under either of the two scenarios just described. A fourth click removes all of the reflecting boundaries. The *Automatic Limits* feature only works if the analysis window is up to date. The locations of reflecting boundaries are stored in the life history parameters files.

Selecting Starting Sites

PATCH provides three mechanisms for specifying the size and locations of the initial breeding population. PATCH cannot be initialized with nonbreeders. The initial population can be distributed randomly across the breeding sites, it can be placed preferentially into the highest quality neighborhoods (a neighborhood is a hexagon plus its six neighbors), or it can be located by hand using the mouse. In addition, any configuration of initial locations can be locked into place, and thus applied to every one of a series of replicate simulations. These three options (random, best, and locked) correspond to the *R*, *B*, and *L* check-boxes found in Figure 17.

If the random or best neighborhood options are being used, then it is necessary to specify only the size of the initial population. If the starting sites are to be selected by hand, then the *Specify Locations* button must be used. Click on this button, move the mouse pointer into the analysis window, and then click on each of the sites that is to be initialized. For this to work, the analysis window must be up to date, and the sites selected must be suitable for breeding. Once the mouse pointer is moved outside of the analysis window, the procedure ends. This process can be used in combination with either of the other two methods of selecting starting sites. Any time you set a starting location by hand, the *Initialization* parameter (Figure 17) is automatically locked. You can always view the current placement of initial breeders in the analysis window by clicking on the *Reset & Initialize* and *Display Locations* buttons. If you have already run a simulation, you may have to press this button twice to generate a new initial population.

Time Series Analysis

PATCH provides you with a way to alter the landscape while a demographic simulation is running. You do this by telling the model to load new territory maps at specific years. The life history window contains a time series editor (Figure 19) that exists for this purpose. To use the time series editor, first select the desired year. If a territory map has already been entered for that year, then the name of map appears in the panel's text fields. Otherwise, enter the location and name of the map you want to use. An existing map can be removed using the *Remove This Entry* button. A new map can be installed using the *Add This Entry* button. The list of active maps can be examined using the *Search Backward* and *Search Forward* buttons. If multiple replicate simulations are being conducted, you must install a territory map at year zero. This prevents the model from starting a new replicate using the ending map from the previous replicate. The life history window footer informs you about any actions taken by the time series editor.

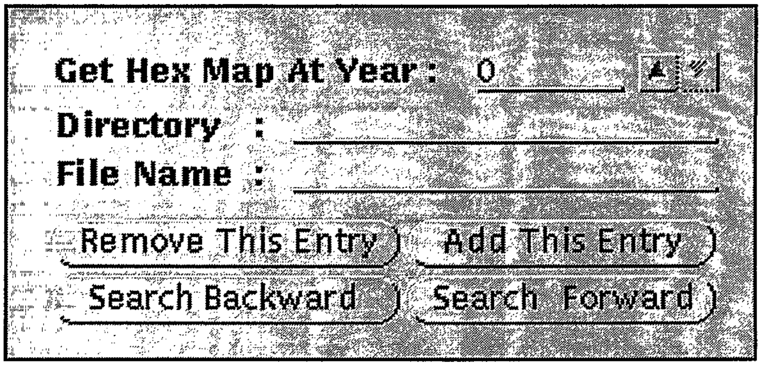


Figure 19. PATCH's time series editor. The utilities in this panel allow you to alter the landscape while a simulation is running. PATCH accomplishes this by loading new territory maps at specified times.

The Parameters Window

The majority of the information used to describe a species is entered into PATCH through the life history parameters window (Figure 20). The life history parameters window contains two panels. The left panel holds ten different model parameters and the right panel holds a projection matrix. The ten parameters include the number of runs and number of years per run to be used in the simulations. It is also where you specify the starting population size, the age or stage class of the starting population, and the minimum and maximum movement ability of the species. This panel holds two parameters (*Search Randomly* and *Walk Up Gradient*) that govern the behavior of random walks. It is also where you set the year in which the tabulation of source-sink data for the species begins (the *Tally Utility From* parameter). Lastly, this is where you define the percent of the maximum habitat quality that is to be associated with the vital rates entered into the projection matrix. You specify this using the field labeled *Vital Rates Factor.*

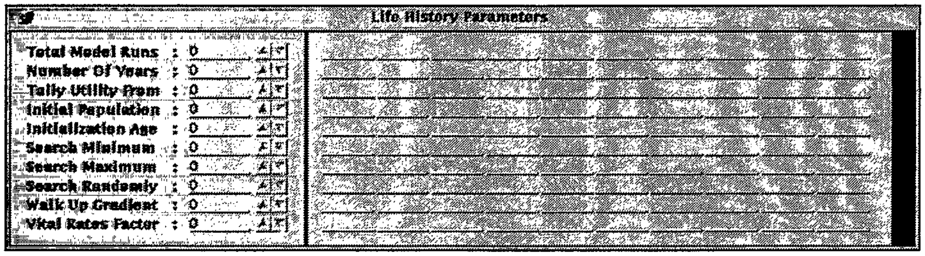


Figure 20. The life history parameters window. The left panel holds a variety of parameters that describe a species and set up the properties of a simulation. The right panel holds a 10 x 10 projection matrix.

The right panel of Figure 20 is where the population projection matrix is supplied to the model. This panel can accommodate up to a 10 x 10 matrix. The rows and columns of this matrix, and hence the age or stage classes, are numbered 0 - 9. You must provide a projection matrix with at least two age or stage classes, but it is not necessary to fill the entire 10 x 10 matrix present in the model. PATCH's life history module incorporates a post-breeding census, which means that the first age or stage class always consists exclusively of newly recruited individuals that are not yet of breeding age. Thus the upper left entry in PATCH's projection matrix is necessarily fixed at zero. The vertical bar on the right hand side of Figure 20 is used to display the life history functions window (Figure 21). You can also right click in the life history parameters window to call up the life history functions window.

The Functions Window

PATCH uses interpolation functions to compute the expected survival and reproductive rates associated with hexagons of an arbitrary quality. These functions are accessed through the life history functions window (Figure 21). The functions provided are linear, logistic, concave (cubic growth), convex (cubic decay), constant, and piecewise constant. Hexagons with a score of zero are always assigned survival and reproductive rates of zero. Hexagons having a score equal to that specified by the *Vital Rates Factor* parameter are assigned the survival and reproductive rates appearing in the projection matrix. Survival and reproductive rates for hexagons with all other scores are obtained from the interpolation functions.

If the constant function is used, then the survival and reproductive rates appearing in the projection matrix are applied to every hexagon that has a score greater than zero. Hexagons with scores of zero are assigned vital rates of zero. The break points for the piecewise constant function are at 1/3 and 2/3 of the maximum hexagon score. Hexagons with less than 1/3 of the maximum score are assigned survival and reproductive rates of zero. Hexagons with more than 2/3 of the maximum score are assigned the survival and reproductive rates entered into the projection matrix. All other hexagons are assigned vital rates equal to 1/2 of the values entered into the projection matrix. The maximum hexagon score is always equal to the maximum weight assigned to any of the image categories in the legend, even if no hexagon actually has a score that high. A hexagon would have to consist entirely of the habitat that was assigned this maximum weight in order to attain the maximum score.

The Life History Functions

The life history interpolation functions are included in PATCH to give you control over the way survival and reproductive rates scale with habitat quality. Most spatially explicit models ignore this issue by assuming a constant or piecewise constant functional form. While

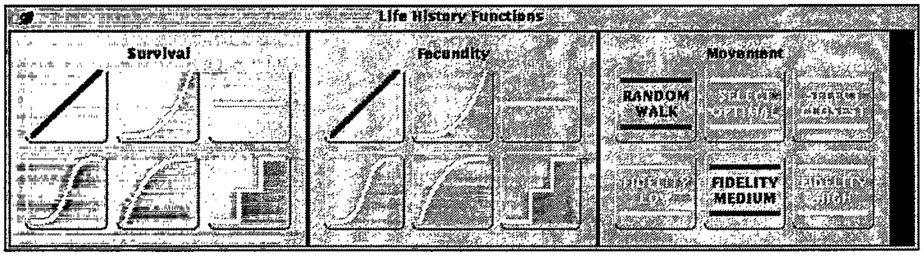


Figure 21. The life history functions window. The left and center panels allow you to select interpolation functions to be used with the survival and reproductive routines. The right panel is used to select a movement strategy and the site fidelity behavior.

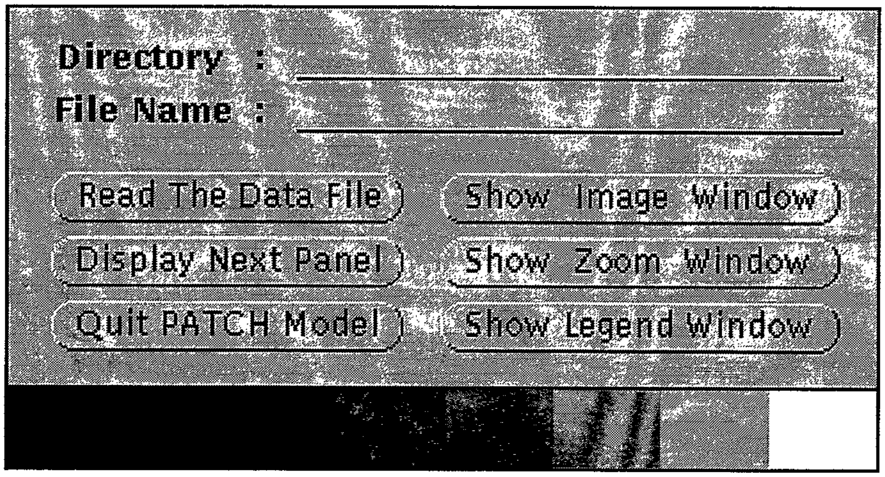
PATCH's additional flexibility in this area may seem like a burden, there are at least three approaches you can take to selecting the interpolation functions. Ideally, field data can be used to fit survival and reproductive rates directly to estimates of habitat quality. If this is possible, the interpolation functions can be chosen that most closely match the curves fitted to the field data. A second approach is to use the interpolation functions as free parameters to help you tune the observed population-wide vital rates. The observed vital rates reflect the influence of habitat quality on the population as a whole. You might, for instance, supply PATCH with biological upper limits for the survival and reproductive rates of a species, and then associate these values with optimal habitat using the *Vital Rates Factor*. Interpolation functions can then be chosen so that the observed aggregate vital rates or λ's agree with field-based estimates. A third alternative is simply to use the life history interpolation functions in sensitivity analyses. Some of the functions may be ruled out in advance using plausibility arguments. The remainder can be examined to see how significant a role they play in controlling the results of your life history simulations.

Chapter 8 PATCH Utilities

Introduction

This chapter contains an exhaustive list of the utilities present in the PATCH model. Each section begins with an image of one of PATCH's control panels. The remainder of the section describes the function of each button and field located within the panel.

Reading and Viewing Imagery



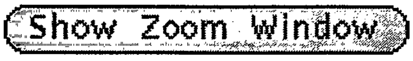
This is the first panel you interact with. The directory and file name fields are used for loading GIS imagery, and also by PATCH's histogram routine. This panel also provides you with access to three of PATCH's graphics windows, and is where you exit the program.

This button is used to load a GIS data set into PATCH. Pushing this button also resets most of the model parameters to their default values. You cannot conduct any analyses until a GIS data has been successfully loaded.

This button calls up a hidden panel that contains 10 additional buttons.

This button exits PATCH. You will not be given the option to cancel this command.

This button calls up the image window. The image window displays a subsampled version of the entire GIS data set. The level of subsampling is set using the *Image Window Sampling* field. When this field is set to n (an arbitrary integer), every nth row and every nth column from the GIS data set are shown in the image window. As n decreases, the window gets larger and more data is displayed.

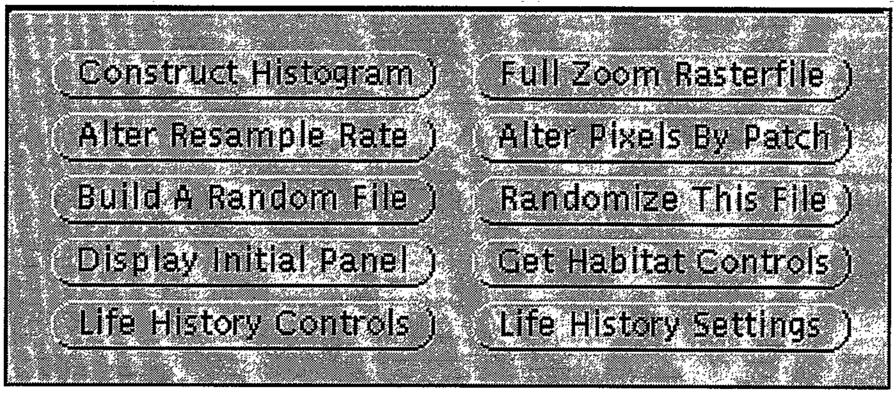


This button calls up the zoom window. The zoom window displays the contents of the zoom-box. The zoom-box is shown in the image window, and can be moved around and resized in a variety of ways. The data displayed in the zoom window are shown in their entirety (unlike the image window), and they can be magnified using the *Degree Of Magnification* field. Only the data within the zoom window are used when the patch identification or territory analysis routines are run.

This button calls up the legend window. The legend window displays the category names and colors present in the control file. Entries in the control file that do not contain a category name are skipped in the legend. The legend window also contains a facility for assigning weights (habitat affinities) to classes in the GIS data. These weights are needed by the patch identification and territory allocation routines. Habitat weights are assigned by clicking on the legend categories. A left click increments a weighting value, while a right click decrements it. If the left and right buttons are used together, the weighting value is set to zero. Note that a weight of zero is displayed as an empty box in the legend window. Middle clicking on a legend category inserts the editing arrow. The editing arrow is used to define a target habitat class for use with PATCH's pixel and patch editing procedures. The editing arrow can only be inserted if the GIS file has UNIX write permission.

The color bar (shown throughout the manual in gray-tone) is used to change the color of the zoom-box and of the hexagon grid in the zoom window. You click on the desired color, and it is automatically applied to both features. However, selecting the white color cell removes both the hexagon grid from the zoom window and the zoom-box from the image window. If the size of the zoom-box exceeds that of the territory or patch map, use of the white color cell also produces a white box in the zoom window that shows you the extent of the map.

Miscellaneous Functions



This panel is obtained by clicking on the main window's *Display Next Panel* button. The buttons labeled *Get Habitat Controls*, *Life History Controls*, and *Life History Settings* simply call up other control windows. The button labeled *Display Initial Panel* recalls the panel that was originally displayed in this part of the main window. The remaining utilities have more complex functions.

This button causes PATCH to generate a histogram showing the number of occurrences of each habitat class within the area enclosed by the zoom-box. The histogram includes a record of the row and column values of the zoom-box edges used in its construction. It also includes a list all of the habitat classes, the class names, and a tally of the number of pixels and hectares of each class found in the data. The histogram is written to the computer screen unless a file name is provided in the main window. The histogram routine never overwrites an existing file.

This button generates a new Sun Rasterfile containing only the image present in the zoom window. This routine is identical to the one accessed with the *Zoom Rasterfile* button (located in the habitat controls window), except that this utility creates a rasterfile whose colormap contains every entry present in the legend window. In contrast, the *Zoom Rasterfile* button strips out legend colors not used in the image. This utility works only if an output file name has been provided in the habitat controls window.

The resample rate provides a mechanism for splitting the data pixels into finer units. This is useful primarily when the desired territory size cannot be obtained using the original pixel dimensions. When the resample rate is set to n (an arbitrary integer), every pixel in the GIS data set is split into an $n \times n$ array of pixels. Patch or territory maps constructed using the same GIS data but different resample rates are not identical. At a minimum, their extent in pixels is different. For this reason, PATCH must store the resample rate that was used to construct a territory or patch map. This information is recorded in the ".patch" file. When you attempt to load a patch or territory map, PATCH compares the resample rate stored in the ".patch" file to that currently set through the model interface. If the two values do not match, then PATCH alerts you and aborts the load.

In some very special cases, you might want to alter the value of the resample rate as it is stored in the ".patch" file. This can be done by setting the resample rate field in the main window to the value desired, entering the name of an existing patch or territory map in the habitat controls window (use the input fields but do not load the data), and clicking on the *Alter Resample Rate* button. In order for the edited patch or habitat map to be useful, you must ensure that its row and column boundaries do not lie outside the range defined by the GIS imagery and the new resample rate.

Alter Pixels By Patch This is one of the tools available for using PATCH to edit GIS imagery. This utility only works when the GIS file being used has UNIX write permission, and when the patch identification routine has been run. To proceed, you must first insert the editing arrow at one of the legend entries by clicking the middle mouse button on the desired category. When the *Alter Pixels By Patch* button is pushed, every pixel in every patch present in the landscape is changed to the legend value containing the editing arrow. This change is made directly to the GIS imagery, so it is permanent. The patch map is discarded after the process has finished, since the underlying data from which it was built has been altered.

Build A Random File This utility builds a new Sun Rasterfile based on the image present in the zoom window. The new image has the same extent as the zoom-box, and contains only the habitat classes actually found the zoom window. The habitat classes are assigned randomly to the pixels in the new image, so the class frequencies in the new image all end up roughly identical. You must provide an output file name in the habitat controls window.

Randomize This File This utility is similar to the *Build A Random File* routine described above. It also constructs a new Sun Rasterfile based on the image present in the zoom window. The new image has the same extent as the zoom-box, and contains only the habitat classes actually found in the zoom window. The new image becomes a randomized version of the old image, and the numbers of pixels in each habitat class are preserved. You must provide an output file name in the habitat controls window.

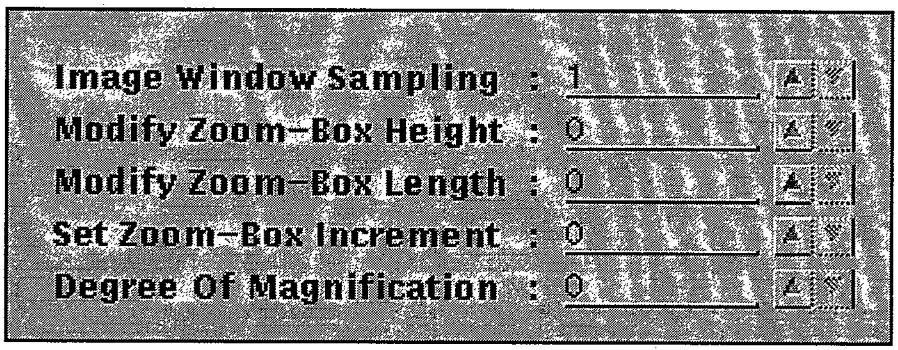
This button simply redisplays the panel that originally appeared at this location.

This button recalls the habitat controls window if it has been removed.

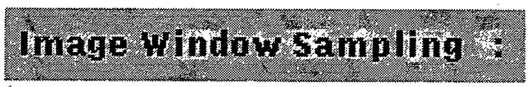
This button recalls the life history controls window if it has been removed.

This button recalls the life history parameters and functions windows.

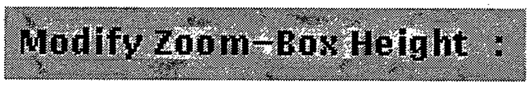
Zoom-Box and Image Window



The process of viewing GIS data and the results of PATCH's analyses requires that you make extensive use of the fields in this panel. You should become familiar with these tools as soon as possible.

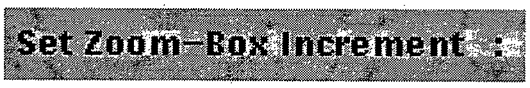


This field changes the size of the image window. The image window subsamples the input data and constructs a dithered version of the entire GIS image. This field sets the subsampling rate used in the process. When this field is set to n (an arbitrary integer), every nth row and every nth column of the GIS data are displayed in the image window. As n decreases, the image window gets larger and more of the GIS data become visible.



This field sets the zoom-box height to the value provided.

This field sets the zoom-box length to the value provided.



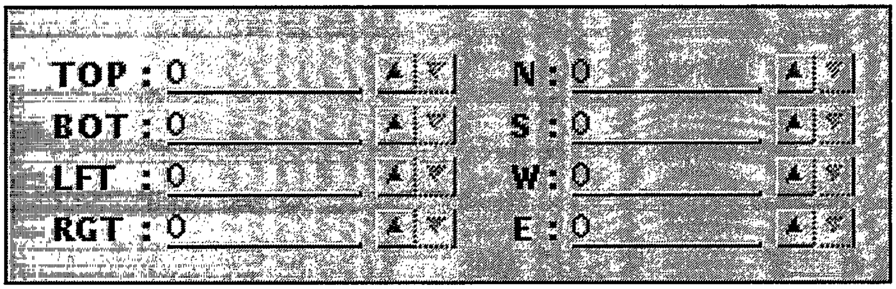
This field sets the distance in pixels that the zoom-box moves when you right click in the zoom or analysis window. If this field is set to zero, the zoom-box moves a distance equal to its current height or width. This feature makes it easy for you to move quickly through a large GIS data set. Since the analysis window also tracks the zoom-box, this technique can also be used to efficiently search large patch or territory maps.

This sets the magnification value to be used with the zoom and analysis windows.

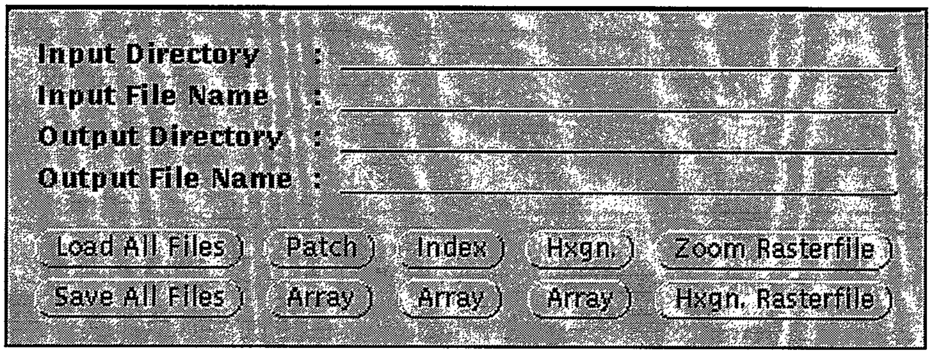
Tracking and Resampling

When the zoom window is sized correctly, it is repainted every time the zoom-box is moved. At times, you may want to move the zoom-box but not wait for PATCH to repaint the zoom window. This can be accomplished by removing the check mark from the *Tracking* check-box. When tracking is off and the zoom-box is moved, the zoom window simply goes blank. A second click replaces the check mark and turns tracking back on again.

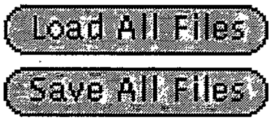
Zoom-Box Placement



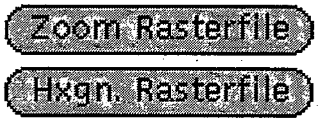
The *TOP*, *BOT*, *LFT*, and *RGT* fields are used to set the zoom-box's top, bottom, left, and right edges to specific row and column values within the GIS data set. The *N*, *S*, *W*, and *E* (north, south, west, and east) fields are used to set the zoom-box's top, bottom, left, and right edges to specific Universal Transverse Mercator (UTM) coordinate values. The UTM coordinates are associated with the centers of the data pixels. The fields specifying UTM information are disabled if the UTM coordinates for the edges of the GIS data set have not been entered into the control file. All of these fields are disabled if a compressed territory map is being used. Otherwise, these fields are updated any time the zoom-box is moved. Thus you can record the placement of the zoom-box at any time by reading these values off of the panel.



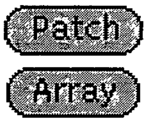
The ability to save patch and territory maps to the hard disk and load them in later is central to the PATCH model. This panel contains the tools used to perform these tasks. The input and output file name fields are used to specify the names of existing patch or territory maps to be loaded, or of new maps to be saved to the hard disk. These fields are also used by a few other routines and are mentioned elsewhere in this manual. The three pairs of buttons in the center of this panel are unlikely to be of much interest to you. Much of the information they generate can be obtained in a more concise format using the *Statistics Output* button (also located in the habitat controls window).



These buttons load existing patch or territory maps, or to save new maps as files on the hard disk. Every patch and territory map consists of a ".patch" and a ".index" file. Territory maps also include a ".hexgn" file. These files are all binary, so you cannot read them directly. The write protect feature present in the habitat controls window can be used to prevent the *Save All Files* utility from overwriting an existing data set.



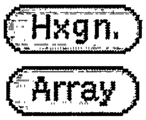
The *Zoom Rasterfile* button generates a new Sun Rasterfile containing the image present in the zoom window. This routine strips out any habitat classes that do not appear in the zoom window's image. The *Hxgn Rasterfile* button is used to generate a new Sun Rasterfile that contains the portion of a territory map present in the analysis window. This feature does not work with patch maps. The hexagon grid is not included in either of the output images. Core areas are not displayed in the image created using the *Hxgn Rasterfile* button. You must provide an output file name in the habitat controls window.



These buttons print out ascii versions of the image data imbedded in a patch or territory map. The button labeled *Patch* prints a summary, while the button labeled *Array* prints the full data array. These utilities read from a

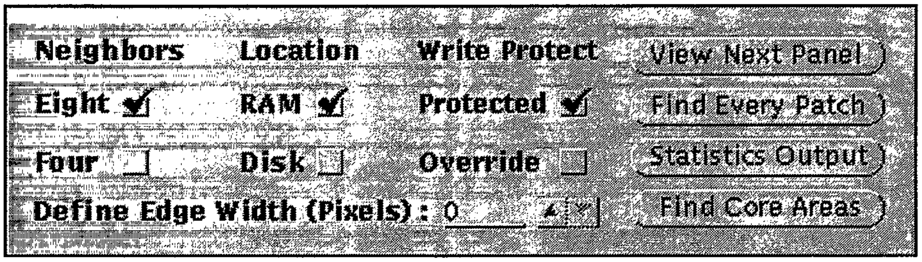
".patch" file if a valid input file name is provided in the habitat controls window. Otherwise they read the data stored in memory, if any exists. These utilities write to an output file if one is provided in the habitat controls window. Otherwise, they write to the screen.

These buttons print out ascii versions of the location and properties data imbedded in a patch or territory map. The button labeled *Index* prints a summary, while the button labeled *Array* prints the full data array. These utilities read from a ".index" file if a valid input file name is provided in the habitat controls window. Otherwise they read the data stored in memory, if any exists. These utilities write to an output file if one is provided in the habitat controls window. Otherwise, they write to the screen.



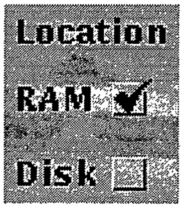
These buttons print out ascii versions of the hexagon specific data imbedded in a territory map. The button labeled *Hxgn* prints a summary, while the button labeled *Array* prints out the full data array. These utilities read from a ".hexgn" file if a valid input file name is provided in the habitat controls window. Otherwise they read the data stored in memory, if any exists. These utilities write to an output file if one is provided in the habitat controls window. Otherwise, they write to the screen.

Patch Identification

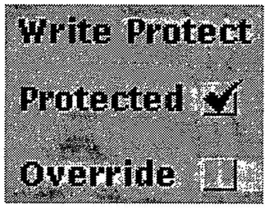


This panel is where the routines controlling the patch identification process are located. It is not necessary for you to master these utilities before using PATCH's life history simulator.

These check-boxes specify the rule used to group collections of pixels into patches. Setting this field to *Eight* implies that every pixel has eight neighbors (four adjacent and four diagonal pixels). Setting it to *Four* implies that every pixel has only four touching neighbors (the four adjacent pixels).



This check-box specifies the location of the image data generated during patch identification or territory allocation. This utility is designed to help you construct patch or territory maps that are too large to be stored in the computer's memory. Setting the check-box to *RAM* forces the image data to be stored in the computer's memory. If an attempt to build a patch or territory map in memory produces an error message stating *Memory Allocation Failed - Use Disk File*, then the location check-box should be set to *Disk*. This forces the image data to be written directly to the hard disk (as a ".patch" file). PATCH uses disk files that are too large to fit in memory by treating them as virtual memory. When the *Location* check-box is set to *Disk*, the ".index" and ".hexgn" files are also automatically written to the hard disk. PATCH's write protect feature can be used to prevent this utility from overwriting an existing data set.



This check-box controls PATCH's write protect feature. It is provided to help you avoid accidentally overwriting patch or territory maps when using *Save All Files* or building a map directly on the hard disk. Existing data are safe when the check mark is placed on *Protected*.

This field sets the edge width used to identify core habitats. Edge widths are measured in pixels, and core (interior) habitat pixels must be separated from non-habitat by a distance of at least one edge width in every direction. Core areas are identified and displayed any time this field is greater than zero, including when territories are being constructed.

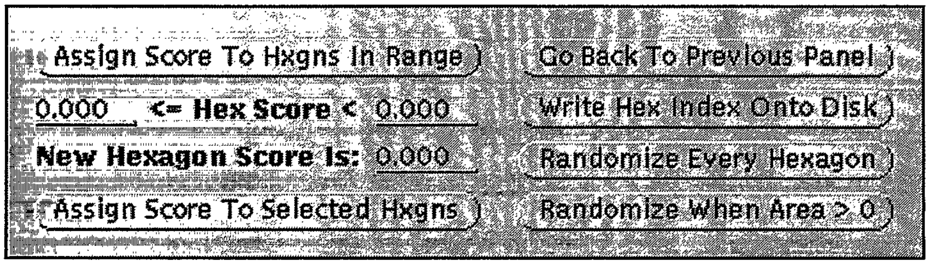
Clicking on this button displays the panel containing the hexagon editing routines.

This button starts the patch identification routine. When each new patch is identified, its bounding rectangle is displayed in the image window (if the window is up to date). This feature helps you follow the routine's progress. The patch identification algorithm breaks very large landscapes up into blocks that are between 500 and 1000 pixels in width and height. It then proceeds within each of these subregions, and reconnects the results when it has finished. The patch identification routine is quite fast, and you should not hesitate to use it on large GIS data sets.

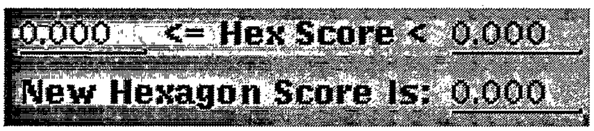
This button prints a report that describes a patch or territory map. The report contains a subset of the information found in the ".patch", ".index", and ".hexgn" files. The format of the statistics output varies a bit depending on whether patches or territories are present. This utility writes to a file if a valid file name is provided in the habitat controls window. Otherwise, it writes to the computer screen. If the *Define Edge Width* field is set greater than zero, then PATCH identifies all of the core habitat present in the landscape prior to generating the statistics report.

This button makes PATCH locate the core (interior) habitat present within a patch or territory map. PATCH recognizes core areas as habitats separated from non-habitat by a distance of at least one edge width in every direction. These core areas are displayed in cyan if they are associated with a patch map, and in yellow if they are associated with a territory map.

Hexagon Editing



These utilities allow you to edit territory maps. You can not alter the scores of hexagons with no habitat. The randomization utilities can only be used on territory maps that have been compressed using PATCH's hexagon compression utility.



These fields allow you to select a target range of hexagon scores, and to specify a new score to be applied when editing hexagons. The target range includes its lower bound but not its upper bound. The *New Hexagon Score Is* field cannot be set higher than the largest weight used to construct a territory map.

Pushing this button edits every hexagon that lies entirely within the zoom-box and that has a score falling within the target range (see discussion immediately preceding). Each of these hexagons is assigned the new score specified in the *New Hexagon Score Is* field. After the hexagon scores have been changed, the habitat sharing routine is rerun. This makes PATCH re-evaluate the breeding status of every hexagon in the landscape that contains some habitat.

Assign Score To Selected Hxgns This button allows you to modify a territory map. After this button is pushed, move the mouse pointer inside the analysis window. The window header should read *CHANGE HEXAGON SCORES*. You can then select (left click) hexagons to modify them. Selected hexagons are painted blue and their scores are changed to the value of the *New Hexagon Score Is* field. These hexagons can be reset to their original value by clicking on them a second time. When the mouse pointer is moved outside the analysis window, the window header returns to normal, the editing process terminates, and the habitat sharing routine is rerun. This causes PATCH to re-evaluate the breeding status of every hexagon in the landscape that contains some habitat.

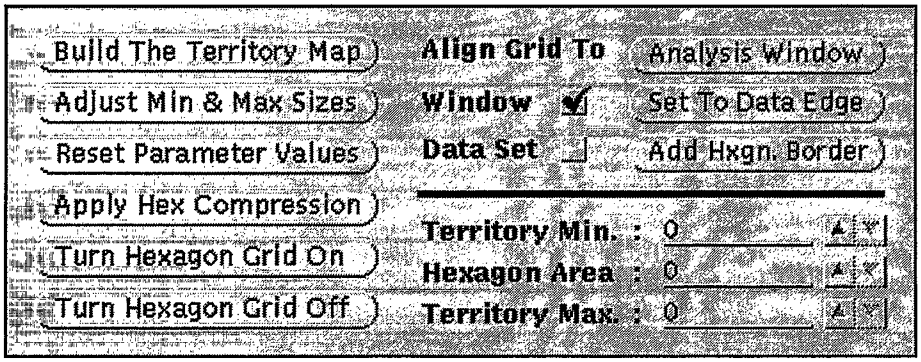
Go Back To Previous Panel This button redisplays the panel that was originally located here.

Write Hex Index Onto Disk This button writes a new ".hexgn" file to the hard disk. You may want to do this after editing a territory map to make the changes permanent. Saving just the ".hexgn" file is much quicker than rewriting an entire territory map to the disk. You must provide an output file in the habitat controls window.

Randomize Every Hexagon This routine randomly arranges all of the hexagons in a territory map. The locations of the hexagons change, but their areas and scores stay the same. After the territory map has been randomized, the habitat sharing routine is rerun. This causes PATCH to re-evaluate the breeding status of every hexagon in the landscape that contains some habitat. This randomization routine works only with compressed hexagons.

Randomize When Area > 0 This routine randomly arranges all of the hexagons in a territory map that contain some habitat (those painted gray and green). The locations of the hexagons change, but their areas and scores stay the same. If you want to preserve features such as a coastline, but randomize everything else, then this routine should be selected over the one labeled *Randomize Every Hexagon*. After the territory map has been randomized, the habitat sharing routine is rerun. This causes PATCH to re-evaluate the breeding status of every hexagon in the landscape that contains some habitat. This randomization routine works only with compressed hexagons.

Territory Construction



This panel contains the utilities used to construct territory maps. It is important to become familiar with all of these features unless PATCH is to be used only for displaying GIS data and constructing patch maps.

These fields allow you to specify the hexagon size and the range of territory sizes used in constructing a territory map. You cannot alter the hexagon area after a territory map has been constructed. In contrast, the territory minimum and maximum can be changed at any time. When hexagons are incremented (decremented) in size, PATCH increases (decreases) their width by two pixels at a time in order to keep them symmetric. This means that the hexagon area can not be set to any arbitrary value.

The smallest hexagon that PATCH constructs has an area of 12 pixels. The hexagons then increase in size to 36, 90, 168, 270 pixels, and so on. If you cannot obtain the desired hexagon size, the resampling rate can be increased until PATCH produces the intended value. But this strategy has a cost: setting the resampling rate to 2 is the same as working with a data set four times as large. Setting it to 3 produces a data set nine times as large, and so on. Under such scenarios, PATCH's analyses run more slowly, and its output files are larger.

The number of hexagons and the hexagon size in hectares are shown in the habitat controls window footer. If the territory minimum or territory maximum is not equal to the hexagon size, then these quantities are displayed in the footer as well. The territory minimum cannot be changed until at least one legend weighting value has been provided.

The *Hexagon Area* field gets locked any time a territory map is compressed, or if a compressed map is loaded into the model. To remove PATCH from this state, you must click once on the button labeled *Built The Territory Map*. This causes the compressed map to be discarded but does not construct a new territory map.

Build The Territory Map This button makes PATCH build a new territory map. However, if either a patch map or a compressed territory map currently resides in PATCH's memory, clicking on this button simply removes the patch or territory data. In these situations, a second click builds a new territory map. If a compressed territory map is removed in this way, the zoom-box size and location are likely to change.

Adjust Min & Max Sizes If the minimum or maximum territory size is altered after a territory map has been constructed, pushing this button propagates these changes through the map. Running this utility causes PATCH to re-evaluate the breeding status of every hexagon that has some habitat.

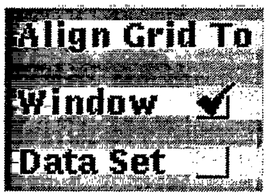
Reset Parameter Values This button resets the hexagon size, territory minimum and maximum, and the habitat weights to the values used to construct a territory map.

Apply Hex Compression This button runs the hexagon compression procedure. This utility only works when an uncompressed territory map is present in the model. The result of running this utility is a new territory map constructed entirely from hexagons 12 pixels in area. From the standpoint of the life history simulator, the new map is identical to the old one. But, if the original map was built from hexagons that were greater than 12 pixels in area, then the compressed map takes up much less space in memory and on the computer screen. Every hexagon in the compressed map is painted a solid color (either black, gray, or green) since the patch structure present in the original map is lost.

Large territory maps can sometimes take a long time to build, and if they are compressed the original data are lost. Therefore, you may want to save uncompressed territory maps before running the compression routine. If the uncompressed data are taking up too much disk space, they can always be compressed with a UNIX file compression routine.

Turn Hexagon Grid On This displays the hex grid in the zoom and analysis windows.

Turn Hexagon Grid Off This removes the hex grid from the zoom and analysis windows.



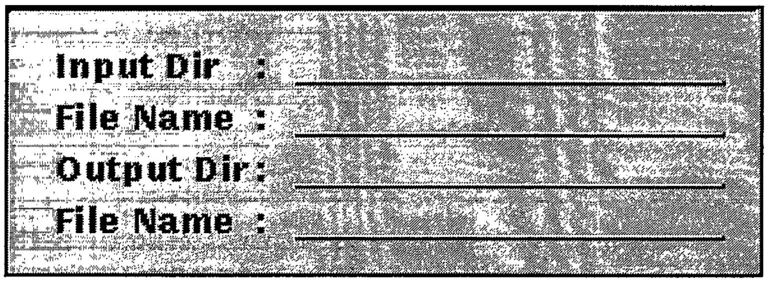
This feature is useful when the zoom-box size is not equal to the size of a territory map being viewed. When you examine a territory map, you want the hexagon grid tied to the territory data, regardless of the size and location of the zoom-box. Clicking on the *Data Set* check-box aligns the hexagon grid to the territory map's boundaries. But if you are going to construct a new map, you typically want to link the hexagon grid to the zoom-box (and hence the zoom window) instead of an old territory map. This can be accomplished by clicking on the *Window* check-box. If a new territory map is constructed with the *Data Set* box checked, it is built from the region used to construct the previous map.

If a patch or a territory map is present in memory, then clicking on this button sets the edges of the zoom-box to the edges of that map. This provides you with an automatic way to zoom out and view an entire patch or territory map (assuming it fits on the screen). When this button is pushed, the *Align Grid To* check-box is set to *Window.*

If a patch map is present, this button works the same as the one labeled *Set To Data Edge*. However, if a territory map is present, clicking on this button sets the edges of the zoom-box slightly beyond the edges of the map (as long this does not exceed the limits of the GIS data). This tends to make it a little easier to see the territory map. When this button is pushed, the *Align Grid To* check-box is set to *Data Set.*

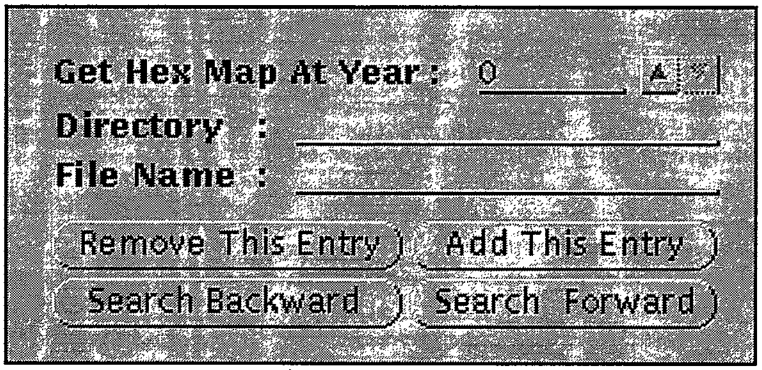
Clicking this button causes the analysis window to be displayed.

Life History File Names

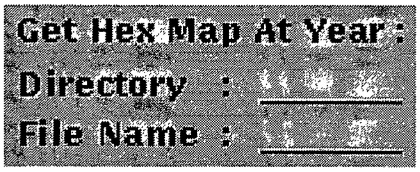


The fields in this panel allow you to specify the locations of life history parameter files to be stored or retrieved. These fields are used frequently when demographic simulations are being conducted.

Time Series Editor



The time series editor is used to introduce landscape change to PATCH's demographic simulations. PATCH approximates landscape change by installing different territory maps at specified points in time.



These utilities allow you to specify territory maps to add or remove from PATCH's time series facility. When a year is selected, any territory map to be installed at that time is shown in the directory and file name fields. As the model runs, the time series editor's year field tracks the model year. If a file name linked to that year comes up in the editor, the model stops and loads the territory map with that name. Once the new map has been installed, the model continues the simulation where it left off.

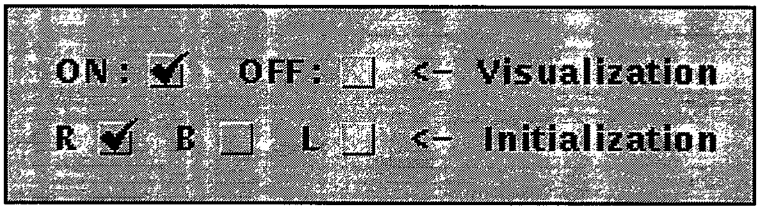
This button removes a territory map that has been placed in the time series editor.

This button loads a territory map into the time series editor at a specified year.

This makes the time series editor search backward for a year with an installed map.

This makes the time series editor search forward for a year with an installed map.

Visualization and Initialization

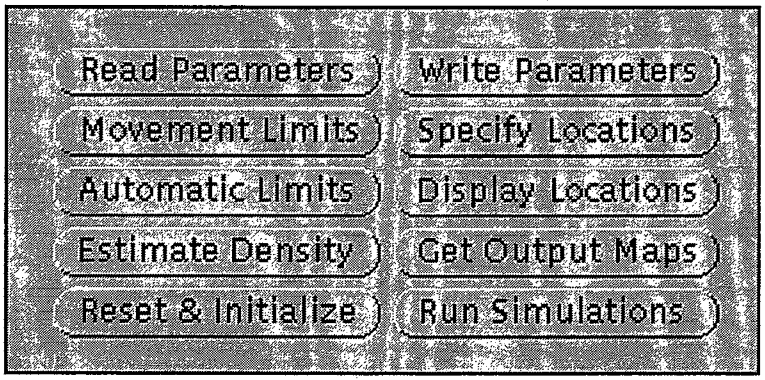


PATCH's life history simulator can display the location and movements of individual members of a population as the model runs. The visualization check-box provides a mechanism for you to turn this feature on or off. The model runs much more slowly if the visualization is on, so you probably want to keep it off most of the time. If a single replicate is being conducted, and if the *Tally Utility From* parameter is set to zero, you can conduct part of a model run with the visualization parameter off, and part of it with the option set on. If the visualization parameter is on, the current population, and any reflecting boundaries, are displayed every time that the analysis window is repainted.

The initialization check-box allows you to control how the starting population is distributed throughout a landscape. This feature is used with the *Initial Population* and *Initialization Age* parameters. The initial population always consists exclusively of breeders. You can force PATCH to reselect the starting population at any time by clicking the *Reset & Initialize* button. You can make PATCH display the locations of the initial population in the analysis window by clicking the *Display Locations* button. If the initialization check-box is set to *R* (random), PATCH selects the locations of the initial population randomly from the breeding sites present in the landscape. If this check-box is set to *B* (best), PATCH places the initial population in the landscape's best neighborhoods. For this purpose, PATCH defines a neighborhood as a breeding site and its six immediate neighbors. The best neighborhoods are those with the largest cumulative scores.

You can also manually enter the initial population. This is done using the *Specify Locations* button, and this feature can be used to modify any existing initial distribution. In addition, it can be used to modify a final population distribution resulting from a simulation. When the *Specify Locations* button is pushed, the initialization check-box is set to *L* (locked), and the present distribution of breeding individuals is locked into place. While this lock is on, you cannot modify the initial population except by using the *Specify Locations* utility. Setting the lock on also ensures that every model run begins with the same initial population. Any starting locations entered by hand are lost if the initialization lock is taken off. Therefore, if you have put a lot of time into selecting starting locations by hand, it is a good idea to save this data to a life history parameters file using the *Write Parameters* button.

Life History Controls



The tools in this panel control PATCH's life history simulations. These utilities are accessed every time the model is run, and you should become familiar with them as soon as possible.

Read Parameters Clicking on this button makes PATCH load a life history parameters file. An input file name must first be entered in the life history window. Feedback and error messages are sent to the life history window footer. If a ".stats2" file is present, and if its prefix is the same as the name of the life history parameters file being read, then PATCH loads the data contained in it as well. This feature allows you to generate occupancy rates and an observed source-sink map from the results of a previous simulation. If a ".random" file is present, and if its prefix has the name of the life history parameters file being read, then PATCH loads this data too. This feature allows you to add environmental stochasticity to a simulation.

Write Parameters This button makes PATCH write a life history parameters file. If an output file name has been entered into the life history window, then these data are written to that file. Otherwise, the data are sent to the computer screen.

Movement Limits This button allows you to enter reflecting boundaries by hand. After pushing this button, move the mouse pointer inside the analysis window. PATCH is ready to assign reflecting boundaries if the analysis window header changes to *SELECT REFLECTING SITES*. A reflecting boundaries can be applied by selecting (left clicking) a hexagon. A reflecting boundary can be removed by selecting the hexagon a second time. Sites that are suitable for breeding cannot be assigned reflecting status. This function only works if the analysis window is up to date. The locations of reflecting boundaries are stored in the life history parameters files.

Automatic Limits This button allows you to automatically locate reflecting boundaries throughout a territory map. If no reflecting boundaries are present, clicking this button initially places a reflecting boundary at every black hexagon (those having no habitat) that has a neighbor with some habitat (the gray or green hexagons). Clicking on this button a second time places a reflecting boundary at every gray hexagon (those having habitat but not suitable for breeding) with a neighbor that is either black (no habitat) or green (suitable for breeding). A third click locates a reflecting boundary at every site that received one as a result of either the first or second clicks. A fourth click removes all of the reflecting boundaries. This feature only works if the analysis window is up to date. The locations of reflecting boundaries are stored in the life history parameters files.

This button allows you to enter the locations of the starting population by hand. After pushing this button, move the mouse pointer inside the analysis window. PATCH is ready to assign reflecting boundaries if the analysis window header changes to *SELECT STARTING SITES*. You can add a member of the starting population by selecting (left clicking) a hexagon. Selecting the hexagon a second time will remove the individual. Only sites that are suitable for breeding can be assigned a member of the starting population. This function only works if the analysis window is up to date. The locations of the starting population are not stored in the life history parameters files unless the *Initialization* check-box is set to *L* (locked). However, PATCH sets the *Initialization* check-box to *L* any time you set a starting location by hand.

This button makes PATCH display the location of every member of the current population. If a simulation has not yet been run, or if the life history module has been reset, you are instead shown the locations of the starting population. In either case, the locations of all reflecting boundaries entered into the landscape are shown as well.

This button prints a distribution of ending population densities for the portion of the landscape contained within the zoom-box. When this button is pushed, PATCH first identifies every hexagon that is both suitable for breeding and contained entirely within the zoom-box. These hexagons are painted blue (in the analysis window) so you can see which ones are included in the tabulation. Next, PATCH reads the ".stats0" file with the prefix specified in the life history window's input file name field. If a ".stats0" file with this name does not exist, then density estimates are not made. Otherwise, PATCH counts up the total number of breeders that were present in the target hexagons at the end of every replicate run, and reports this data on a run-by-run basis. PATCH also writes out the row and column values of the zoom-box's edges, and the mean and standard deviation of the density values, computed across the collection of replicate runs. Lastly, PATCH writes out the statistics present in a ".stats2" file, but for just the breeding sites contained entirely within the zoom-box. This information is provided to help you summarize the importance of a specific collection of breeding sites. If an output file name is provided, PATCH sends all of this information to a file with that prefix, but that ends in ".stats3". If an output file name is not provided, PATCH sends the data to the computer's monitor. Changes to the image in the analysis window made by this routine are not permanent. You can restore the original image using the window manager's refresh feature, or using the *Turn Hexagon Grid On* button.

This button makes PATCH write three output maps to the hard disk. These maps are placed in files that end in ".image0", ".image1", and ".image2". The first of these maps (".image0") displays the occupancy rates observed at the end of each replicate simulation, but averaged across all of the replicate runs. The second and third maps (".image1" and ".image2") contain the expected and observed source-sink distributions, respectively. The expected source-sink distribution can be generated without first running a simulation. A simulation must be conducted before the occupancy rate or observed source-sink maps can be compiled. To generate an expected source-sink map, you must supply a territory map and specify the vital rates, the vital rates factor, and the interpolation functions. The data used in building the observed source-sink map are compiled only if the *Tally Utility From* field is set greater than zero. The data used in building the occupancy rate map are compiled if the *Tally Utility From* field is greater than zero, or if the number of runs is greater than one. You can recreate the observed source-sink or occupancy rate maps from the data stored in the ".stats2" output file. PATCH automatically uploads this data any time a life history parameter file is read, provided a valid ".stats2" files exists. Once these data have been uploaded, you can push the *Get Output Maps* button to recreate the observed source-sink and occupancy maps.

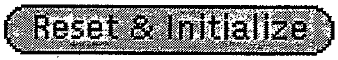
The occupancy rate maps are Sun Rasterfiles composed of 4-pixel "hexagons" that are derived from territory maps. Each hexagon in an occupancy rate map is assigned to a color class. Hexagons with no habitat are displayed in white. Hexagons that have habitat but are not suitable for breeding are displayed in black. All other hexagons (the breeding sites) are assigned a color from a spectrum that shifts from red to blue. The color assigned to a hexagon depends on its relative occupancy rate. Sites with a relative occupancy rate of zero (the minimum) are displayed in 100% red, and those with a relative occupancy rate of ten (the maximum) are shown in 100% blue. The colors vary smoothly for sites in-between. The relative occupancy rates are constructed from tallies of the breeders present in the final year of each (or a single) replicate simulation. At the end of each replicate, a breeding site's occupancy register is incremented if the site is occupied, and is left unchanged if it is not. When the occupancy map is built, the values in these registers are rescaled to lie between zero and ten.

The expected source-sink map is a Sun Rasterfile composed of 4-pixel "hexagons" that is derived from a territory map. Each hexagon in an expected source-sink map falls into one of five classes. Hexagons may have no habitat, have habitat but be unsuitable for breeding, or they can be classified as sinks, sources, or as neutral sites. The source or sink designation (which applies only to breeding sites) is based on the dominant eigenvalue (λ) of the projection matrix associated with each hexagon. This projection matrix is constructed from the vital rates entered into the model interface, the vital rates factor, the interpolation functions you selected, and from a hexagon's score. If this value is less than one the site is classified as a sink. If it is greater than one the site is listed as a source. Sites that have a dominant eigenvalue of exactly one are considered neutral. This source-sink designation is not affected by the presence or absence of the data tables used to introduce environmental stochasticity into the model. The data used to create the expected source-sink map are stored in the ".stats2" file under the heading *Lambda*. A new expected source-sink map can be always be built by loading a territory map and a set of life history parameter files that include a ".stats2" file. The expected source-sink map is not affected when a simulation is conducted.

The observed source-sink map is also a Sun Rasterfile composed of 4-pixel "hexagons" that is derived from a territory map. Each hexagon in an observed source-sink map falls into one of five classes. Hexagons may have no habitat, have habitat but be unsuitable for breeding, or they can be classified as sinks, sources, or as neutral sites. The source or sink designation (which applies only to breeding sites) is based on the number of immigration and emigration events observed at a site during a simulation. Any time an individual enters a breeding hexagon, an immigration event is recorded for that site. Similarly, an emigration event is recorded whenever an individual exits a site. These tallies are made on a hexagon-by-hexagon basis. The process of tallying immigration and emigration events can be started at a specific year using the *Tally Utility From* field in the life history parameters window.

When the number of immigration events is greater than the number of emigration events, a site is classified as a sink. If emigration events exceed immigration events, a site is considered a source. When immigration and emigration are exactly balanced, a site is labeled neutral. Most neutral sites fall into this category because immigration and emigration both equal zero. The data used to create the observed source-sink map are stored in the ".stats2" output file under the heading *Utility*. The utility values are computed as the emigration rate minus the immigration rate. Thus, sites with utility values less than zero are sinks, while those greater than zero are sources. If the *Tally Utility From* field is set to zero, or to a value greater than the number of model-years, then immigration and emigration tallies are not made. In these cases, every site's utility value equals zero. The immigration and emigration tallies are summed across replicate model runs.

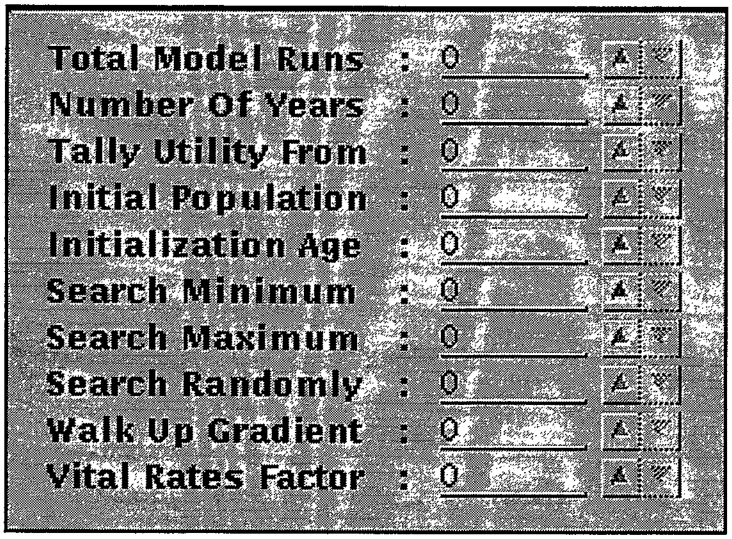
Sinks and sources in both the expected and observed source-sink maps can be distinguished based on their severity. PATCH makes this possible by taking the observed range of the sinks, and the observed range of the sources, and splitting each into 100 uniform intervals. Each of these intervals is one percent in width, and is represented by a unique class in the source-sink map's color table. Hexagons are assigned to these classes when the maps are constructed. The ranges, and hence the interval sizes, used for the sources and sinks are usually not the same. All sinks are painted red and all sources are painted green. Only hexagons that are suitable for breeding can qualify as a source, sink, or neutral site. Hexagons that do not contain any habitat are painted white. Hexagons are painted black if they have habitat but are not suitable for breeding. Only the hexagons that are suitable for breeding are recorded in the ".stats2" files.



This button sets the run number and year to zero, and inserts the starting population.

Pushing this button initiates a simulation.

Life History Parameters



This panel is used to supply many of PATCH's demographic parameters. Familiarity with an organism is not required to specify the panel's top three fields. Simulation results are often insensitive to the initial population size and age, so you may be able to justify picking these values arbitrarily. The *Search Randomly* and *Walk Up Gradient* parameters can only be estimated, and PATCH's animation facility is designed to help you perform this task. Model results tend to be sensitive to the remaining parameters, particularly the vital rates factor.

Total Model Runs : This field specifies the number of replicate model runs to be conducted. If this field is greater than one, the run number and year are set to zero when a new simulation is initiated.

Number Of Years : This field sets the duration in years of each model run. This parameter can be altered without resetting the life history module. Altering any other parameter, except the visualization check-box, causes PATCH to reset the run number and year to zero. If the number of runs is set to one, and the *Tally Utility From* field is set to zero, then you can make PATCH conduct a simulation in increments. For instance, the model could be run for 100 years with the visualization check-box set off. Then you could turn the visualization on and run the model for an additional 10 years.

Tally Utility From : This field specifies the model year in which the immigration and emigration data are first recorded. These data make up the observed source-sink map, and you should use this parameter to delay tallying them until any transient behaviors resulting from the initial conditions have had a chance to die down. If the *Tally Utility From* parameter is set to zero, then the immigration and emigration data are not collected at all. If this parameter equals one or more, the run number and year are automatically reset to zero every time you initiate a demographic simulation. PATCH's occupancy rate data are only collected only if the *Tally Utility From* parameter is set greater than zero, or if the number of replicate runs is greater than one.

Initial Population : This field sets the size of the starting population. However, this field gets locked if the initialization check-box is set to *L*. This parameter is incremented or decremented if you add or remove starting individuals using the *Specify Locations* button.

Initialization Age : Starting populations consist of a single cohort of breeding individuals.

This field is used to specify the age or stage class of this cohort. PATCH numbers its ages and stage classes from zero to nine.

Search Minimum : This parameter specifies the minimum distance an individual must move before a vacant breeding site can be colonized. If the movement strategy selected does not involve a random walk, this parameter defines a disk about the starting site inside which hexagons are not available for colonization. Individuals taking a random walk are obligated to move a number of steps that is at least equal to the value of this parameter. You must specify the minimum search distance as a number of steps from a hexagon to one of its immediate neighbors. It is usually convenient to know the distance between hexagons in meters when assigning a value to this parameter. PATCH displays the mean distance between neighboring hexagons (the step size) in meters any time you middle click on a hexagon in the analysis window. This information is sent to the habitat controls window footer. The step size is reported as a mean value because the distances separating each hexagon's horizontal and diagonal neighbors are slightly different.

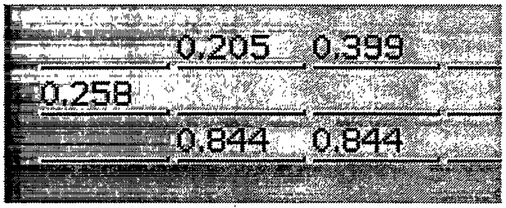
Search Maximum : This parameter limits the maximum distance any individual can move in search of a vacant breeding site. If the movement strategy does not involve a random walk, then this parameter defines a disk about the starting site outside which a hexagon is not available for colonization. In such cases, if a vacant site does not exist within the search radius, a random walk is taken. Individuals taking a random walk move a number of steps that is at most equal to the value of this parameter. You must specify the maximum search distance as a number of steps from a hexagon to one of its immediate neighbors. It is usually convenient to know the distance between hexagons in meters when assigning a value to this parameter. PATCH displays the mean distance between neighboring hexagons (the step size) in meters any time you middle click on a hexagon in the analysis window. This information is sent to the habitat controls window footer. The step size is reported as a mean value because the distances separating each hexagon's horizontal and diagonal neighbors are slightly different.

Search Randomly : This parameter influences the behavior of individuals taking a random walk. Decreasing its value makes movement paths more linear. As this parameter is increased, movement paths become more random. If this parameter is set to zero, individuals travel in straight lines. If this parameter is set to 100%, individuals select movement directions entirely at random. However, the influence of this parameter is overridden by an individual's tendency to move up gradient from lower to higher quality hexagons. If you want to simulate a true random walk, you must set this parameter to 100% and set the *Walk Up Gradient* parameter to zero. Extremely low values of this parameter may produce unrealistic movement paths.

Walk Up Gradient : This parameter influences the behavior of individuals taking a random walk. It governs the tendency to move up gradient from lower to higher quality hexagons. When this parameter is set to zero, individuals never attempt to move into better quality sites. As the value of this parameter is raised, individuals become increasing likely to move into higher quality habitats. When this parameter is set to 100%, individuals always select the best neighboring site to move into, as long as that sites' score is higher than the hexagon currently occupied. Extremely high values of this parameter produce unrealistic movement paths.

Vital Rates Factor : The vital rates factor determines the quality of habitat to be associated with the survival and reproductive values you supply. Hexagons with scores of zero are always assigned survival and reproductive rates of zero. Hexagons having the score indicated by the vital rates factor are assigned the survival and reproductive rates entered into the model's projection matrix. Survival and reproductive rates for hexagons with all other scores are obtained using PATCH's interpolation functions. The vital rates factor must be set to a percentage of the maximum possible hexagon score. The maximum score is always equal to the largest weighting value used in the legend window, even if no hexagon actually has a score this high. For instance, a vital rates factor of 80% implies that the survival and reproductive rates you entered are to be realized in hexagons having a score equal to 80% of the maximum value. The vital rates factor also governs the survival and reproductive rates experienced by individuals when environmental stochasticity is included in a simulation.

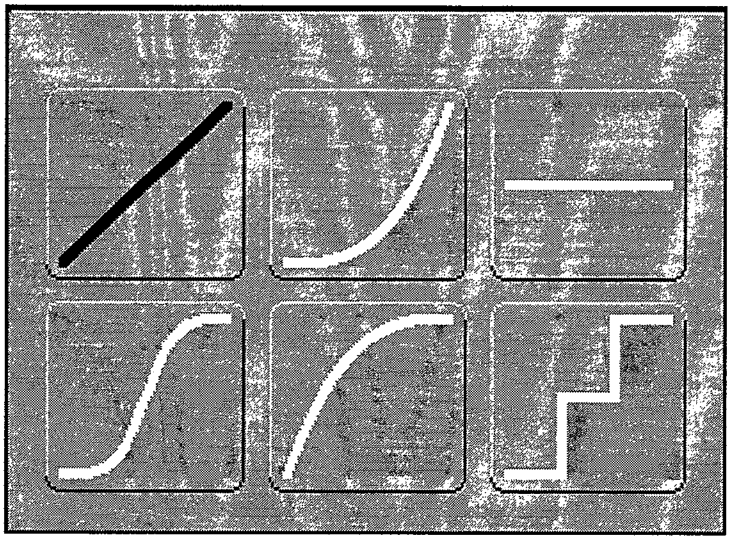
The Projection Matrix



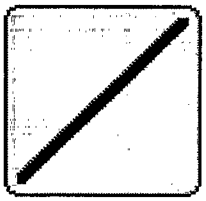
$$\longleftrightarrow \begin{bmatrix} 0 & 0.205 & 0.339 \\ 0.258 & 0 & 0 \\ 0 & 0.844 & 0.844 \end{bmatrix}$$

Population projection matrices are entered into PATCH through the panel on the right side of the life history parameters window. This panel contains a 10 × 10 array of fields that can each hold a single vital rate. Rates set to zero are not displayed. These survival and reproductive rates are associated with hexagons having scores specified by the vital rates factor. The maximum survival and reproductive rates (those observed in hexagons having the highest possible score) are displayed any time you click the middle mouse button in the projection matrix. The survival and reproductive rates experienced by an individual are influenced as well by the interpolation functions you select. The survival rates present in any column of PATCH's projection matrix must sum at most to one, and this is also true of the matrix used for survival and reproductive decisions in the best habitats. PATCH's life history module incorporates a post-breeding census, which means that the first age or stage class always consists exclusively of newly recruited individuals that are not yet of breeding age. For this reason, the upper left entry in PATCH's projection matrix is always fixed at zero. If a table of vital rates has been supplied for simulating environmental stochasticity, then PATCH randomly selects a projection matrix from this table at the start of each model year. In such cases, the randomly selected vital rates are displayed in PATCH's projection matrix for the duration of that model year. When the simulation ends, PATCH redisplays the vital rates you entered into the model interface.

Interpolation Functions

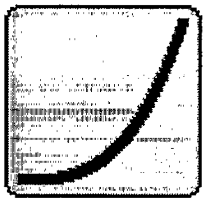


PATCH allows you to specify the relationships linking the survival and reproductive rates to habitat quality. The influence of habitat quality can be ignored by selecting the constant function, or it can be minimized by picking the piecewise constant function. The simplest nontrivial option is a linear relationship. You can also choose between logistic, concave, and convex functions. The mathematical definitions of these interpolation functions are provided in Table 2.

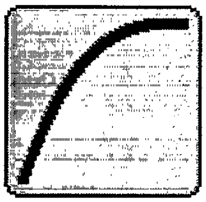


This option makes either survival or fecundity a linear function of habitat quality. A minimum rate of zero is realized in hexagons with a score of zero. The maximum rate is realized in hexagons having a score equal to the largest habitat weight.

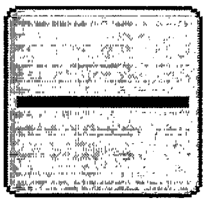
This option makes either survival or fecundity a logistic function of habitat quality. A minimum rate of zero is realized in hexagons with a score of zero. The maximum rate is realized in hexagons having a score equal to the largest habitat weight. It is possible for no hexagon to have a score this high.



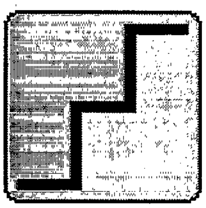
This option makes either survival or fecundity a concave function of habitat quality. A minimum rate of zero is realized in hexagons with a score of zero. The maximum rate is realized in hexagons having a score equal to the largest habitat weight. It is possible for no hexagon to have a score this high.



This option makes either survival or fecundity a convex function of habitat quality. A minimum rate of zero is realized in hexagons with a score of zero. The maximum rate is realized in hexagons having a score equal to the largest habitat weight. It is possible for no hexagon to have a score this high.

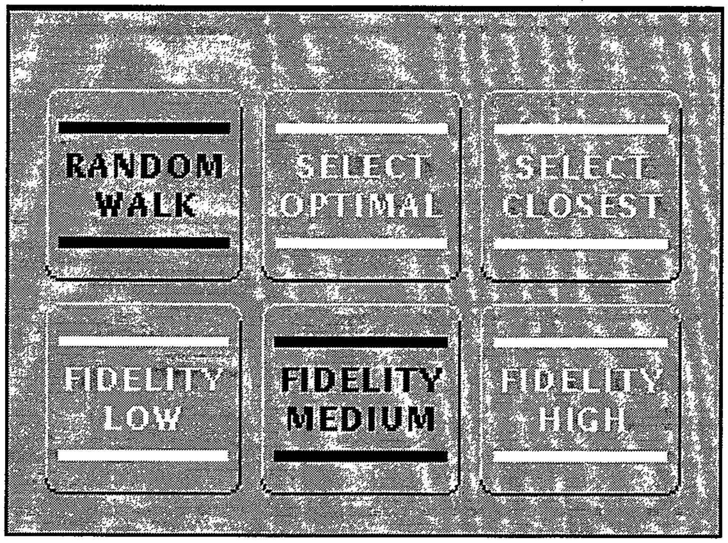


This option assigns the survival or fecundity values entered into the model's projection matrix to every hexagon with a score greater than zero. Survival or fecundity are set to zero in every hexagon that has a score of zero. The role of the vital rates factor is suspended when this function is used.



This option assigns a survival or fecundity value of zero to every hexagon with a score less than one-third of the largest habitat weight. In hexagons having scores greater than two-thirds of the largest habitat weight, survival or fecundity are set to the values entered into the model's projection matrix. In all other hexagons, survival or fecundity are set to one-half of the values entered into the model's projection matrix. The role of the vital rates factor is suspended when this function is used.

Movement Functions



PATCH allows you to choose between three different movement strategies. The button labeled *RANDOM WALK* forces individuals to search for available breeding sites using a pseudo-random walk. The button labeled *SELECT OPTIMAL* allows individuals to travel immediately to the best available breeding site within a search neighborhood. The button labeled *SELECT CLOSEST* allows individuals to take the closest available breeding site within a search neighborhood. In latter two cases, if more than one option exists at the same distance, then the best of these is colonized. Ties are handled randomly. A random walk is always used by individuals that cannot locate an available breeding site with one of the other strategies.

PATCH also provides you with three options for specifying site fidelity. The site fidelity parameter controls the likelihood that territorial individuals (or a floater poised to take over a site who's owner has died) remain in the breeding sites they have colonized. When site fidelity is high, individuals never give up their territories. When site fidelity is low, each individual gives up its territory and relocates every year. At the intermediate value of site fidelity, decisions regarding whether or not to remain in a territory are made based upon hexagon quality.

This button forces individuals searching for available breeding sites to use a pseudo-random walk. The behavior exhibited during a random walk is influenced by the minimum and maximum movement ability, the linearity in searcher's movements, and by their tendency to travel up gradient.

This choice of movement allows individuals to move directly to the best available site falling within an annulus specified by the minimum and maximum movement distances. This is the slowest of the three available movement routines.

This choice of movement allows individuals to move directly to the closest available site falling within an annulus specified by the minimum and maximum movement distances. This option is faster than the optimal site selection algorithm, but it is still slower than a random walk.

When site fidelity is set to low, each individual gives up its breeding site every year, regardless of the site's quality. The individual is then obligated to search for a new site.

When site fidelity is set to medium, individuals relinquish breeding sites that can be expected to function as sinks, and retain breeding sites expected to behave neutrally or as sources.

When site fidelity is set to high, territory holders always remain on their sites indefinitely.

Introduction

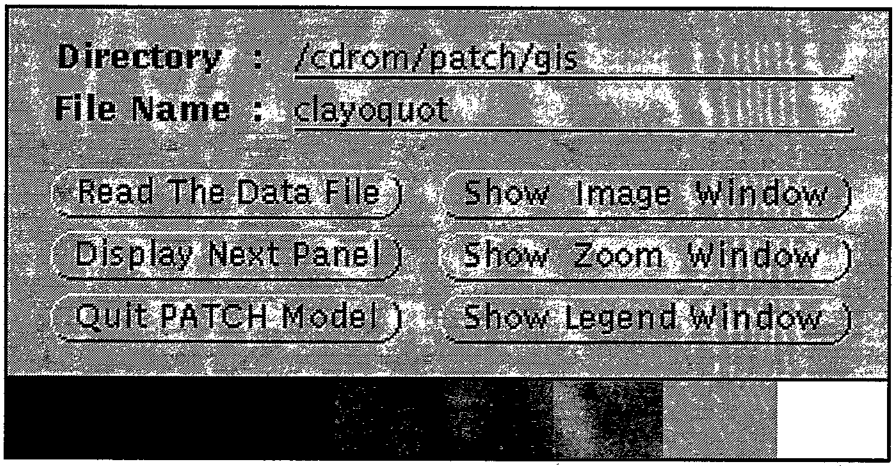
This example introduces the patch identification module. It is also intended to help you become familiar with PATCH's utilities for reading and writing data, and for displaying imagery and results.

Windows in PATCH that contain buttons and text fields are referred to collectively as *control windows*. Windows containing images are referred to as *graphics windows*. Every window in PATCH has a name, displayed in the window header, except for the title and legend windows. The title window is the first window that comes up when the model is started, and its header displays the version of PATCH being used. The legend window displays the classes present in the GIS data. Its header displays the name of the GIS data set being used. Moving the mouse pointer into the image, zoom, or analysis window causes its header to display the name of the GIS data set being used.

Many of PATCH's control and graphics windows also have footers. Footers, always located at the bottoms of the windows, display data and error messages generated PATCH's utilities. It is important for you to get into the habit of monitoring PATCH's various window footers.

Starting PATCH

Begin by starting PATCH and clicking on the title window. This makes all of the control windows visible. Enter the location of PATCH's sample GIS data in the main window *Directory* field. Enter "clayoquot" into the main window's *File Name* field.



Reading GIS Data

Click on the button labeled *Read The Data File*. This makes PATCH read the GIS data set. If the clayoquot image has been successfully loaded, the main window footer displays the number of rows and columns present in the data. The clayoquot image contains 3731 rows and 4301 columns. If PATCH has trouble reading the clayoquot data set or its control file, it writes error messages to the main window footer.

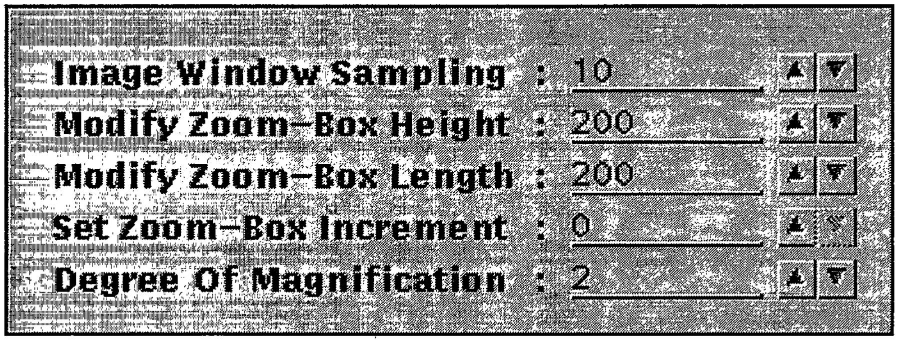
Displaying GIS Data

Call up the image window by clicking on the button labeled *Show Image Window*. Now double click in the image window. PATCH displays every 28th row and 28th column of the clayoquot data set. The image window is designed to subsample the GIS data and generate a dithered version of the entire image. In contrast, the zoom window is designed to display part, or all, of the GIS image without dithering.

Change the *Image Window Sampling* field from 28 to 10 (remember to use the "Return" key). This expands the size of the image window, and paints it black. Double click on the image window to display every 10th row and every 10th column of the clayoquot data. Drag the image window to a convenient location on the desktop.

Setting the Zoom-Box Size

Change the fields labeled *Modify Zoom-Box Height* and *Modify Zoom-Box Length* to 200 pixels each. Notice how the zoom-box size changes (in the image window) as this is done. Make sure that the field labeled *Set Zoom-Box Increment* is set to zero. Lastly, set the *Degree Of Magnification* field to 2 (remember to use the "Return" key). Click on the cyan color cell in the main window. It is just to the left of the white color cell. This changes the color of the zoom-box.

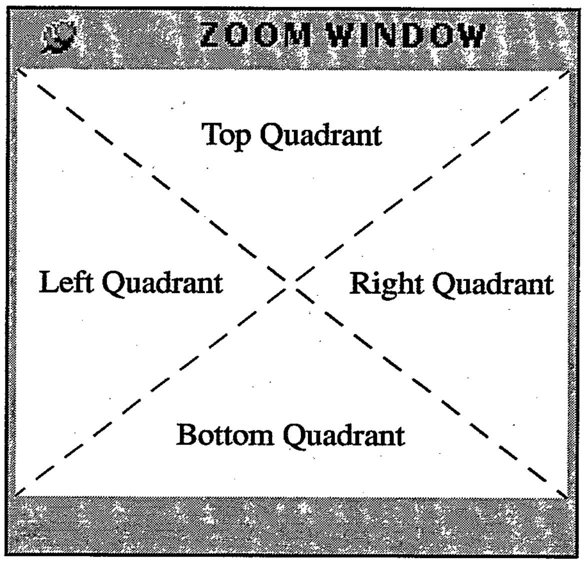


Using the Zoom Window

Call up the zoom window by clicking on the button labeled *Show Zoom Window*. Double click once in the zoom window to set its size. Double click a second time to display the data within the zoom-box. Drag the zoom window to a convenient location on the desktop. Select a data pixel by clicking and holding the middle mouse button anywhere in the zoom window. While the mouse button is held down, the selected pixel is highlighted (painted white) and the zoom window footer displays its row and column values and indicates which legend entry it belongs to.

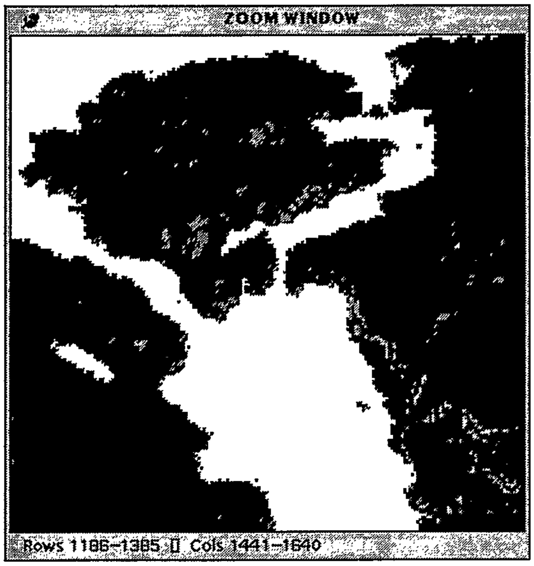
Moving the Zoom-Box

Imagine breaking the zoom window into four quadrants by connecting its opposite corners with two diagonal lines. For the purpose of this discussion, label the resulting triangular regions *top*, *bottom*, *left*, and *right*. When the right mouse button is clicked in one of these quadrants, the zoom-box moves incrementally. The increment size is specified in pixels, and is set by the *Set Zoom-Box Increment* field. If the *Set Zoom-Box Increment* field is set to zero, however, the increment size equals either the height or width of the zoom-box.

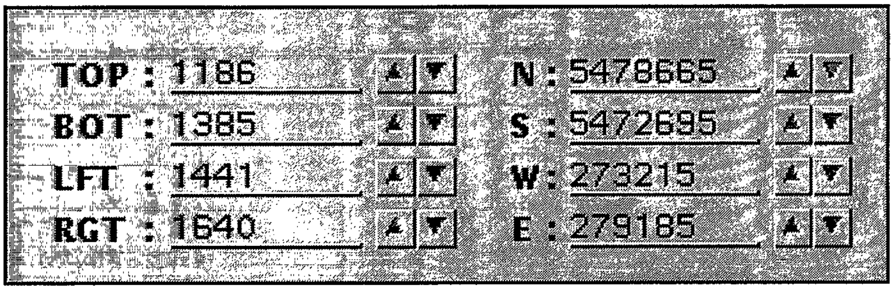


Place the mouse pointer over the zoom window's top quadrant and click the right mouse button three times. Because the *Set Zoom-Box Increment* field is set to zero, each click moves the zoom-box vertically a distance equal to its height (200 pixels). Now place the mouse pointer over the zoom window's left quadrant, and right click three times. This moves the zoom-box a distance of 600 pixels (three times the zoom-box's width) to the left. The zoom window is repainted each time the zoom-box is moved. This happens only when the tracking switch is on and the zoom window is sized correctly.

Now change the field labeled *Set Zoom-Box Increment* to 10 (remember to use the "Return" key). Again, place the mouse pointer over the left quadrant of the zoom window. Click right once. Then click right twice in the bottom quadrant of the window. A small island should be visible in the upper half of the zoom window.



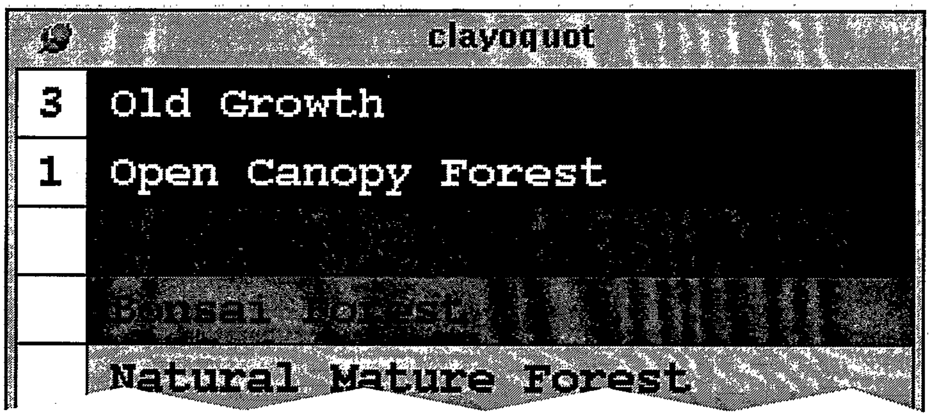
The zoom window footer should read *Rows 1186-1385 [] Cols 1441-1640*. If this is not the case, modify the location of the zoom-box directly by altering the fields in the main window labeled *TOP*, *BOT*, *LFT*, and *RGT*. Change these fields to 1186, 1385, 1441, and 1640.



Setting the Legend Weights

Display the legend window by clicking on the button labeled *Show Legend Window*. Place the mouse pointer on the category labeled *Old Growth* and click the left button three times. The uppermost white box within the legend window should now contain a 3. Move the mouse pointer to *Open Canopy Forest* and click left once. Old growth and open canopy forest should now have weights of 3 and 1, respectively. Every other habitat class should have a weight of 0. Weights that are set to 0 are displayed in the legend window as empty boxes.

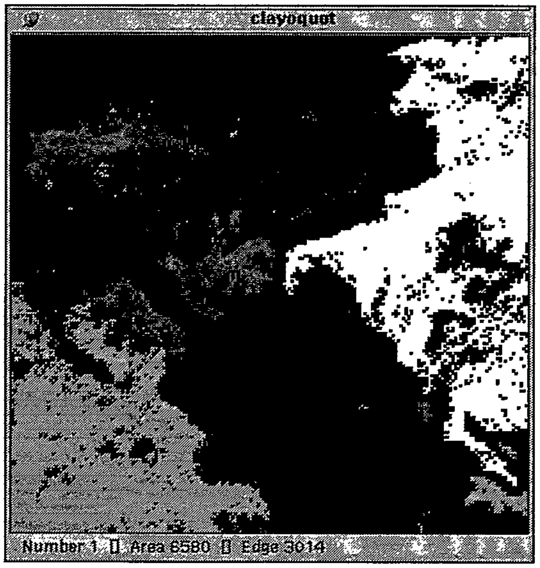
Move the mouse pointer to the habitat controls window. Call up the analysis window by clicking on the button labeled *Analysis Window*. Double click in the analysis window to update its size. Drag the analysis window to a convenient location on the desktop.



Identifying Habitat Patches

Click the button in the habitat controls window labeled *Find Every Patch.* This makes PATCH locate every patch within the zoom-box composed of old growth, open canopy forest, or both. These patches are displayed in the analysis window, and the analysis window footer reads *Number Of Patches: 160.* Take a look at the zoom-box while the patch counting routine is running. As each separate patch is located, its bounding rectangle is drawn within the zoom-box. This feature is designed to help you follow the progress of the patch counting routine.

Move the mouse pointer into the analysis window. Select the large patch covering most of the right hand side of the image by clicking the middle mouse button on it. As long as the mouse button is held down, the patch is highlighted (painted white), and the analysis window footer should read *Number 1 [] Area 6580 [] Edge 3014.*



Picking Four or Eight Neighbors

Locate the check-box labeled *Neighbors* in the habitat controls window. When this check-box is set to *Four*, only adjacent pixels are considered to be touching. When it is set to *Eight*, neighbors at diagonals are assumed to touch as well. Patches are built up from collections of pixels that touch and that have all been assigned nonzero weighting values. The value of the *Neighbors* check-box controls PATCH's definition of what it means for pixels to touch, thus it can greatly influence the results of the patch counting routine.

Change the *Neighbors* check-box from *Eight* to *Four*. Click again on the button labeled *Find Every Patch*. The analysis window footer should show that 364 separate patches have now been identified. Select patch number 1 again. Its area has decreased from 6580 pixels to 6437.

Locating Core Habitat Areas

Reset the *Neighbors* check-box to *Eight*, and set the field labeled *Define Edge Width* to 1 (remember to use the "Return" key). Again, click on the button labeled *Find Every Patch*. A large quantity of core habitat should become visible in the analysis window. Core habitat identified during patch counting is always painted cyan.

Now increase the edge width to three pixels, and click on the button labeled *Find Core Areas*. PATCH removes the previously displayed core habitats and replaces them with the subset that is at least thee pixels from an edge in every direction.

Saving the Results

Enter a directory and file name into the habitat controls window *Output Directory* and *Output File Name* fields. For the sake of this discussion, let's assume you have entered "/tmp" and "test" for the directory and file name fields, respectively. Set the write protect feature to *Override* if a patch or territory map with this name already exists.

Now click on the button labeled *Save All Files*. PATCH attempts to save the results of the patch identification process. If it is successful, the habitat controls window footer reads *test Successfully Written.* This causes PATCH to create two binary (hence not readable) files named "/tmp/test.patch" and "/tmp/test.index".

Patch Statistics

Click on the button labeled *Statistics Output*. This creates a file named "/tmp/test", assuming you have supplied the output directory and file name fields suggested above. This file contains a general description of the results of the analysis run thus far. The first few lines of this statistics file should contain the text displayed here.

```
Input File Name Used : clayoquot
Analysis Is For Rows : 1186 - 1385
Analysis Is For Cols : 1441 - 1640
Edge Width In Pixels : 3

Total Number Patches : 160
Landscape Patch Area : 17660
Sum Of Patch Weights : 41228
Sum Patch Core Areas : 2526
Total Landscape Edge : 10532

Legend Value, Weight :   0,    3
Legend Value, Weight :   1,    1

=====================================
Patch   Patch   Patch   Core    Patch
Number  Area    Weight  Area    Perim
=====================================
1       6580    16078   1219    3014
2       3493    8045    301     2198
3       1       1       0       4
4       1       1       0       4
5       1       1       0       4
```

Loading a Patch Map

Quit PATCH by clicking on the button in the main window labeled *Quit PATCH Model*. Then call PATCH up again and enter the location of PATCH's sample GIS data in the main window *Directory* field. Enter "clayoquot" into the main window *File Name* field. Load the GIS image using the button in the main window labeled *Read The Data File*. Next, set the *Input Directory* and *Input File Name* fields in the habitat controls window to the names used for storing the patch map that was built earlier in this example (e.g. "/tmp" and "test").

Reload the patch map by clicking on the *Load All Files* button in the habitat controls window. All of the information describing the original patch map, except for the core areas, are restored from the ".patch" and ".index" files when the data set is loaded. If the edge width is nonzero, then the core areas are recomputed. If the zoom and analysis windows are sized correctly, they are automatically repainted when the patch map is loaded.

PATCH does not automatically fit the zoom-box to the boundaries of an imported map. Therefore, you may want to adjust the zoom-box size and location after a patch map is loaded. Fortunately, you do not have to set the zoom-box size and location by hand. Instead, you can simply click on the button in the habitat controls window labeled *Set To Data Edge*. This alters the zoom-box size and location to match the extent of the patch map. Click on this button, and then change the magnification parameter to 2. Next, call up the zoom and analysis windows. Double click on each window once to set its size, and a second time to paint it. Examine the patch map that has just been loaded. It should be identical to the map constructed earlier in this example.

Example 2 Territory Allocation

Introduction

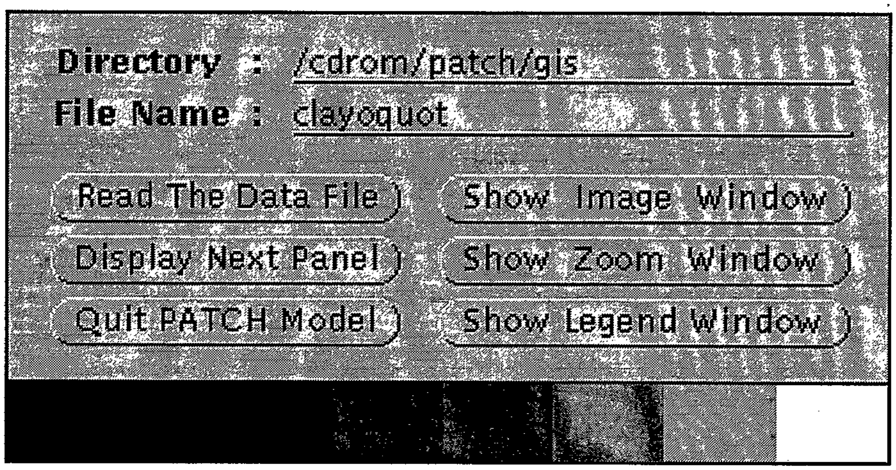
This example introduces the territory allocation module. It is also intended to help familiarize you with PATCH's utilities for reading and writing data, and for displaying imagery and modeling results.

Windows in PATCH that contain buttons and text fields are referred to collectively as *control windows.* Windows that contain images are referred to as *graphics windows*. Every window in PATCH has a name, displayed in the window header. However, there are exceptions to this rule for the title and legend windows. The title window is the very first window that comes up when the model is started. The title window header displays the version of software being used. The legend window displays the classes present in the GIS data. The legend window header displays the name of the GIS data set being used. Also, moving the mouse pointer into the image, zoom, or analysis window causes its header to display the name of the GIS data set being used.

Many of PATCH's control and graphics windows also have footers. Footers, always located at the bottom of the windows, are used to display data and error messages generated by the model's utilities. You should get into the habit of monitoring PATCH's window footers.

Starting PATCH

Begin by starting PATCH and clicking on the title window. This makes all of the control windows visible. Enter the location of PATCH's sample GIS data in the main window *Directory* field. Enter "clayoquot" into the main window's *File Name* field.



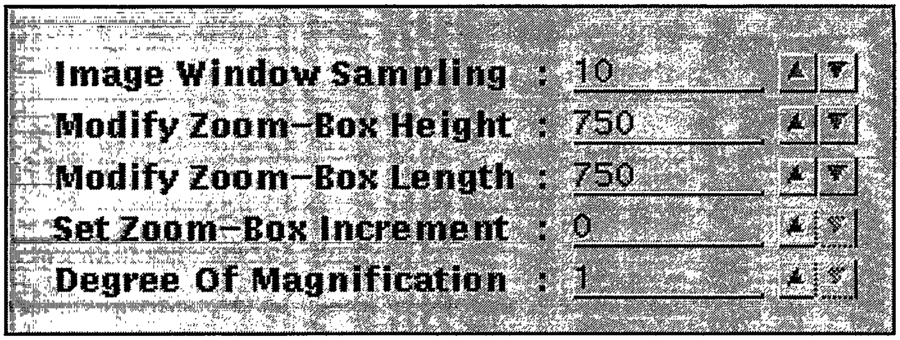
Reading GIS Data

Click on the button labeled *Read The Data File*. This makes PATCH read the GIS data set. If the clayoquot image has been successfully loaded, the main window footer displays the number of rows and columns present in the data. The clayoquot image contains 3731 rows and 4301 columns. If PATCH has trouble reading the clayoquot data set or its control file, it writes error messages to the main window footer.

Displaying GIS Data

Call up the image window by clicking on the button labeled *Show Image Window*. Now double click in the image window. PATCH displays every 28th row and 28th column of the clayoquot data set. The image window is designed to subsample the GIS data and generate a dithered version of the entire image. In contrast, the zoom window is designed to display part, or all, of the GIS image without dithering.

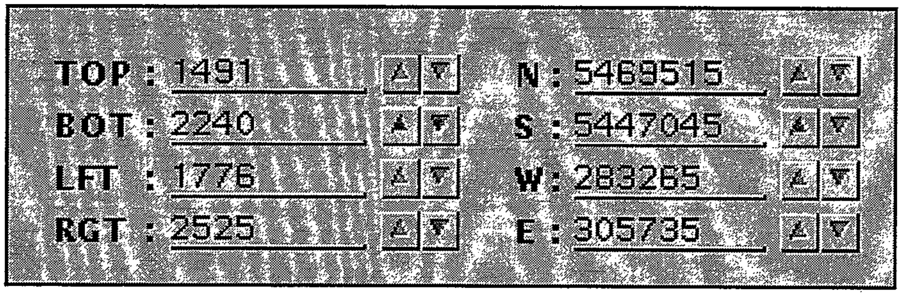
Change the *Image Window Sampling* field from 28 to 10 (remember to use the "Return" key). This expands the size of the image window, and paints it black. Double click on the image window to display every 10th row and every 10th column of the clayoquot data. Drag the image window to a convenient location on the desktop.



Setting the Zoom-Box Size

Change the fields labeled *Modify Zoom-Box Height* and *Modify Zoom-Box Length* to 750 pixels each. Notice how the zoom-box size changes (in the image window) as this is done. Then, set the *Degree Of Magnification* field to 1 (remember to use the "Return" key). Click on the cyan color cell in the main window. It is just to the left of the white color cell. This changes the color of the zoom-box.

The zoom window footer should read *Rows 1491-2240 [] Cols 1776-2525*. If this is not the case, modify the location of the zoom-box directly by altering the fields in the main window labeled *TOP*, *BOT*, *LFT*, and *RGT*. Change these fields to 1491, 2240, 1776, and 2525, respectively. Remember to use the *Return* key.



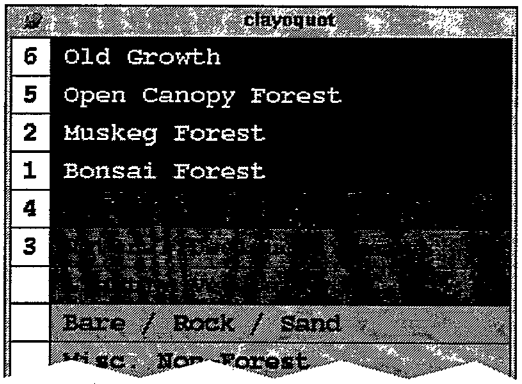
Using the Zoom Window

Call up the zoom window by clicking on the button labeled *Show Zoom Window*. Double click once in the zoom window to set its size. Double click a second time to display the data within the zoom-box. Drag the zoom window to a convenient location on the desktop.

Select a data pixel by clicking and holding the middle mouse button anywhere in the zoom window. While the mouse button is down, the selected pixel is highlighted (painted white), and the zoom window footer displays its row and column values and indicates which legend category it belongs to. Close the zoom window to get it out of the way.

Setting the Legend Weights

Display the legend window by clicking on the button labeled *Show Legend Window*. Place the mouse pointer on the category labeled *Old Growth* and click the left button six times. Move the mouse pointer down to *Open Canopy Forest* and click left five times. Then click twice on *Muskeg Forest*, once on *Bonsai Forest*, four times on *Natural Mature Forest*, and three times on *Natural Deciduous Forest*. The first six classes present in the legend should now be assigned nonzero habitat weights. The other classes should all have weights of zero.



Setting the Hexagon Size

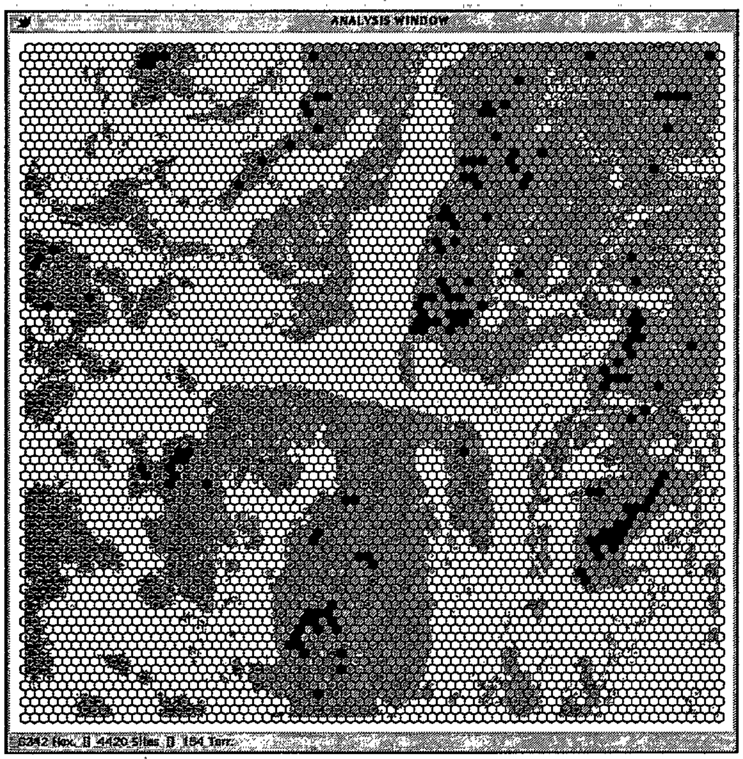
Display the zoom window again by clicking on the *Show Zoom Window* button. Move to the habitat controls window and click on the button labeled *Turn Hexagon Grid On*. The hexagon grid should appear on top of the image in the zoom window. Each hexagon should be 12 pixels in size and difficult to make out at this magnification.

With the zoom window visible, increase the hexagon area considerably (e.g., to 10044 pixels or larger), then decrease it all the way down to 90 pixels. Use the increment or decrement widgets ( ) or the up and down arrow keys on the keyboard. (The cursor must be inserted in the *Hexagon Area* field in order for the up and down arrow keys to work.) Each time the hexagon area is altered, its new value is automatically assigned to the *Territory Min* and *Territory Max* fields.

The habitat controls window footer should now read *6342 Hexagons* and *8.10 Hectares*. This indicates that 6342 hexagons, each with an area of 8.10 hectares (or 90 pixels), can fit within the portion of the image specified by the zoom-box.

Building a Territory Map

Now click on the button labeled *Build The Territory Map*. PATCH constructs a territory map and aligns the hexagon grid to its edges. To see the results, call up the analysis window using the button (also in the habitat controls window) labeled *Analysis Window*. Double click once on the analysis window to set its size. Double click a second time to display the new territory map.



The analysis window footer should read *6342 Hex. [] 4420 Sites [] 154 Terr.*, indicating that the map has a total of 6342 hexagons, that 4420 of them contain at least some habitat (are painted either gray or green), and that 154 of them are territories (painted green), which implies that they are suitable for breeding.

Adding a Border

Now click on the button labeled *Add Hxgn Border*. Pushing this button expands the zoom-box size slightly. Consequently, the analysis window is no longer up to date, and PATCH paints it black. Double click on the analysis window once to resize it, and a second time to repaint the territory map. The territory map and hexagon grid should now be entirely visible within the analysis window. A small black border should separate the outer edge of the hexagon grid from the window edges. This border region lies outside of the portion of the image used to generate the territory map.

Selecting a Hexagon

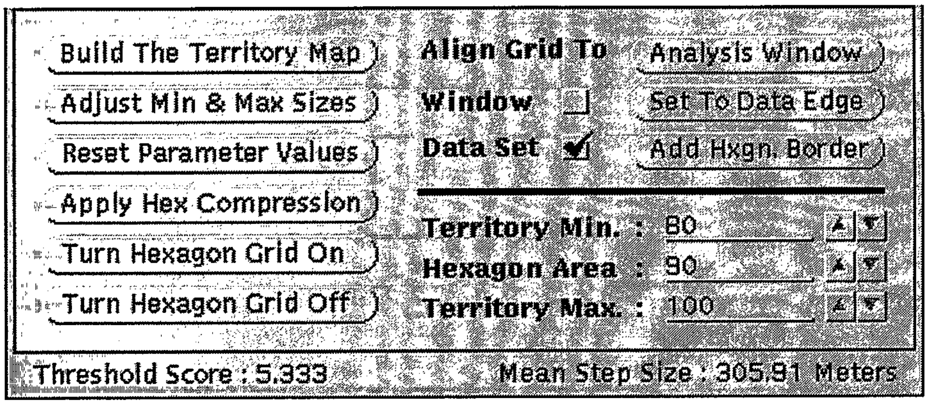
Place the mouse pointer over one of the hexagons present in the analysis window. Select the hexagon by pressing and holding the middle mouse button. As long as the mouse button is held down, the hexagon is highlighted (painted white) and the analysis window footer displays its number, area in pixels, and score. Hexagon numbers are simply assigned sequentially from top to bottom, and left to right. A hexagon's area is equal to the number of pixels of habitat it contains. Habitat is defined as any legend class that has been assigned a nonzero weight. A hexagon's score is equal to the mean value of the weights assigned to every pixel it contains.

When a hexagon is selected, the habitat controls window footer changes to display the threshold score associated with the territory map. This value is the same, regardless of which hexagon is selected. The threshold score is the minimum score a hexagon must have to qualify as suitable for breeding. Remember that hexagons can borrow habitat from their neighbors in order to attain this threshold value. The threshold score for this territory map is equal to 6.0 (the largest weight entered into the legend) because the *Territory Min* field has been set equal to the hexagon area.

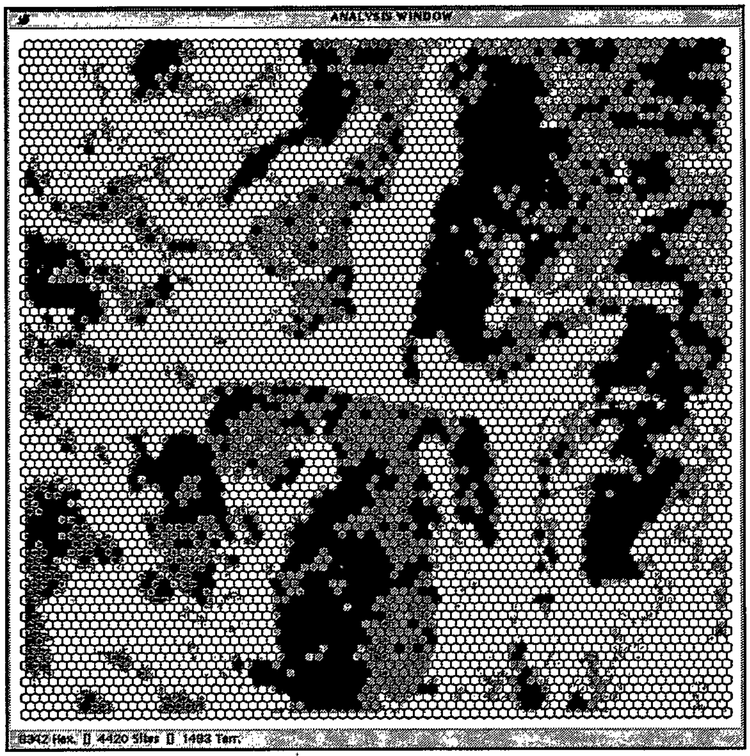
When a hexagon is selected, the habitat controls window footer also displays the mean step size. This is the average center-to-center distance separating a hexagon and each of its six neighbors. The reason for reporting the step size as a mean value is that PATCH's raster hexagons are not entirely symmetric. The distances between horizontal and diagonal neighbors differ slightly. Knowing the mean step size is useful when parameterizing the life history simulator. The mean step size is always reported in meters (305.91 meters in this case).

Territory Min and Max

Now change the field called *Territory Min* to 80. The habitat controls window footer changes to indicate that the minimum territory size has been set to 7.20 hectares. Click on the button labeled *Adjust Min & Max Sizes*. This changes the threshold score from 6.00 to 5.33, and thus increases the number of breeding sites present in the landscape from 154 to 1079. This change should be apparent in the analysis window.



Now change the field called *Territory Max* to 100. The habitat controls window footer changes to indicate that the territory maximum has been set to 9.00 hectares. Click the button labeled *Adjust Min & Max Sizes.*



This further increases the number of breeding sites present in the landscape from 1079 to 1493. This change should also be easy to spot in the analysis window. The *Territory Min* and *Territory Max* fields control the number of hexagons designated as suitable for breeding. Altering them does not change the hexagon scores or sizes.

Saving the Results

Enter a directory and file name into the habitat controls window *Output Directory* and *Output File Name* fields. For the sake of this discussion, let's assume you have entered "/tmp" and "test" for the directory and file name fields, respectively. Set the write protect feature to *Override* if a map with this name already exists. Now click on the button labeled *Save All Files.* PATCH attempts to save the results of the territory allocation process. If it is successful, the habitat controls window footer reports *test Successfully Written.* This causes PATCH to create three binary (hence not readable) files named "/tmp/test.patch", "/tmp/test.index", and "/tmp/test.hexgn".

Locating Core Habitat Areas

Set the field labeled *Define Edge Width* to 10 (remember to use the *Return* key), and click on the button called *Find Core Areas.* A small quantity of core habitat becomes visible in the analysis window. The process of locating core habitat in patch and territory maps is the same, and the presence of the hexagon boundaries does not affect the results. But the core habitats identified within a territory map are reported on a hexagon-by-hexagon basis. This makes it possible for you to rank hexagons based on the quantity of core habitat they contain. Core habitat identified during territory allocation is always painted yellow.

Hexagon Statistics

Now decrease the edge width to six pixels, and click on the button labeled *Statistics Output.* PATCH recomputes the core areas before it generates the output file. Because the edge width was decreased, a large number of additional core habitat is added to that previously displayed. When PATCH is done identifying core habitat, it creates a file named "/tmp/test" that contains a general description of the results of the analysis run thus far.

This statistics file is similar to those generated from patch maps. It begins with a header indicating the GIS data set being used. The rows and columns corresponding to the boundaries of the zoom-box are also included. The file then records the edge width used to make the core area computations. Summary statistics describing the landscape as a whole are displayed next, followed by data characterizing the overall properties of the hexagons contained within the landscape. The header ends with a description of the legend weights you provided. The remainder of the statistics table consists of a record of the attributes of every hexagon present in the landscape that has a score exceeding zero. The first few lines of this statistics file contain the text displayed here.

```
Input File Name Used     : clayoquot
Analysis Is For Rows     : 1491 - 2240
Analysis Is For Cols     : 1776 - 2525
Edge Width In Pixels     : 6
Landscape Patch Area     : 285416
Sum Of Patch Weights     : 1390538
Sum Patch Core Areas     : 103281
Total Landscape Edge     : 45840

Hexagon Size (Pixels)    : 90
Hexgn Threshold Value    : 5.333
Hexgn Expansion Value    : 0.111
Dist Between Hexagons    : 305.91
Total Nbr Of Hexagons    : 6342
Hexagons With Habitat    : 4420
Number Breeding Sites    : 1493

Legend Value, Weight :   0,   6
Legend Value, Weight :   1,   5
Legend Value, Weight :   2,   2
Legend Value, Weight :   3,   1
Legend Value, Weight :   4,   4
Legend Value, Weight :   5,   3
```

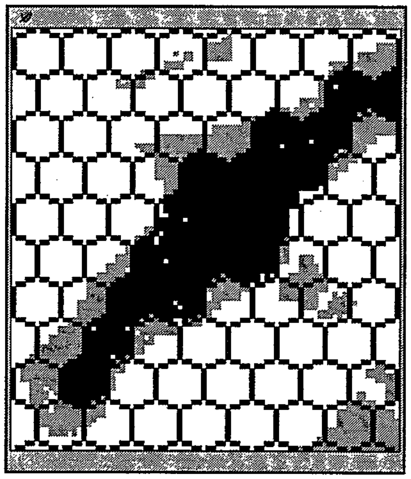
Hexgn Number	Hexgn Score	Hexgn Area	Hexgn weight	Core Area	Hexgn Perim	Breed In Hex
13	3.400	51	306	0	17	NO
14	4.878	74	439	10	10	YES
15	4.933	74	444	14	10	YES
16	4.911	74	442	14	10	YES
17	4.911	74	442	14	10	YES
18	3.922	74	353	8	10	NO
19	0.778	20	70	0	14	NO
20	0.011	1	1	0	4	NO

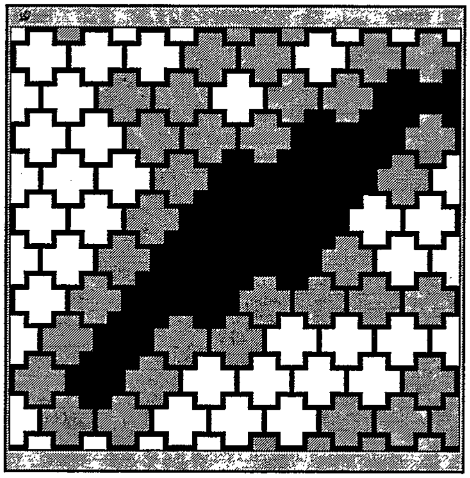
The table above begins with hexagon number 13 because the statistics file lists only hexagons that have some habitat. The score, area, weight, core area, and perimeter of empty hexagons always equals zero. Empty hexagons also never qualify as breeding sites. Now set the edge width to zero. This makes PATCH stop locating core areas and consequently speeds up any subsequent analysis.

Hexagon Compression

The hexagon compression routine is useful for shrinking the size of a territory map. PATCH's compression routine works by constructing a new territory map from 12-pixel hexagons. It then copies the scores and breeding status from each hexagon present in the original territory map over to the corresponding site in the compressed map. The compressed and uncompressed maps are identical to PATCH's life history simulator, but compressed maps are easier to display, take up less space on the hard disk, and can be randomized with the hexagon editor.

Compress the territory map by clicking on the button labeled *Apply Hex Compression*. When this is done, set the magnification parameter in the main window to 2 (remember to use the *Return* key). Double click in the analysis window once to change its size, and again to display the compressed territory map. The compressed hexagons are each 12 pixels in size, and the cells in the hexagon grid now have a slightly different shape. Also, the hexagons in the compressed map are all painted a solid color. The within-hexagon habitat pattern present in the uncompressed map is lost when the compressed image is constructed.





Notice that the uncompressed hexagon size (90 pixels) is still displayed in the *Hexagon Area* field, even though the hexagons themselves have shrunk to 12 pixels each. In order for this feature to work, the *Hexagon Area* field must be locked any time that a territory map is compressed, or if a compressed map is loaded into the model. To remove PATCH from this state, you must click once on the button labeled *Build The Territory Map* (don't do this now). The *Territory Min* and *Territory Max* fields can still be modified when a compressed territory map is being used. The habitat controls window footer displays the uncompressed hexagon and territory sizes.

The original territory map was 750 pixels tall and 750 pixels wide. The compressed map is 251 pixels tall and 300 pixels wide. The zoom-box appears unchanged even though its height and width have become much smaller. PATCH accomplishes this by translating the zoom-box coordinates used to display the compressed map into the corresponding values for the uncompressed map. The zoom-box is displayed with the uncompressed coordinates while the analysis window is displayed using the compressed values. The algorithm that does this cannot map the contents of the zoom-box onto the zoom window, thus the zoom window is disabled when a compressed territory map is being used.

Move the zoom-box around by clicking the middle mouse button in the image window. This alters the image displayed in the analysis window. The zoom-box can be moved only as far as the edges of the territory map (if this is done, the analysis window is painted almost entirely black). This limitation is imposed by the routine that translates between new (compressed) and old (uncompressed) zoom-box coordinates. When you are done, click on the button labeled *Add Hxgn Border* to center the image in the analysis window.

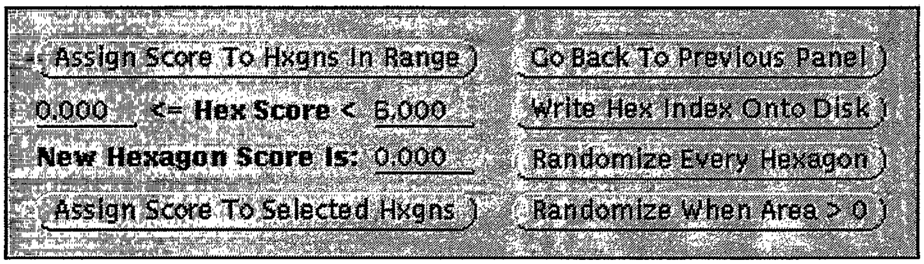
Hexagon Grid Alignment

Move the *Align Grid To* check-box from *Data Set* to *Window* and back again. Notice how the habitat controls window footer changes to reflect the grid alignment. The *Align Grid To* feature allows you to link the hexagon grid to the territory map or to the zoom-box.

Now change both the zoom-box's height and its length to 50 pixels by editing the *Modify Zoom-Box Height* and *Modify Zoom-Box Length* fields in the main window. Set the *Degree Of Magnification* field to 10 (remember to use the "Return" key). Use the middle mouse button to move the zoom-box around in the image window. Every time the zoom-box is moved, the analysis window is repainted. Change the grid alignment to *Window* and then back to *Data Set*. Lastly, turn the hexagon grid off and back on again, using the buttons labeled *Turn Hexagon Grid Off* and *Turn Hexagon Grid On.*

Editing Hexagon Scores by Range

Set the *Degree Of Magnification* field back to 2, and click on the button labeled *Add Hxgn Border.* Then double click on the analysis window once to resize it, and a second time to display the entire compressed territory map. Now click on the button in the center left of the habitat controls window labeled *View Next Panel* (alternatively, right click anywhere in this panel). This displays the hexagon editor.



Locate the two numeric fields in the hexagon editor that are placed adjacent to the text <= *Hex Score* <. These fields are used to define a target set of hexagons that are to be altered as a group. Change the numeric fields so that the lower end of this range equals zero, and the upper end of the range equals six. None of the hexagons in the territory map can have a score greater than six because the largest weighting value was set to six in the legend window. Now set the field labeled *New Hexagon Score Is* to 0.0. Once this is done, click on the button labeled *Assign Score To Hxgns In Range*. PATCH identifies every hexagon lying entirely within the zoom-box (all of the hexagons in this case) that has a score greater than or equal to zero, but less than six. The model then changes the scores of these hexagons to zero.

Examine the results as they are displayed in the analysis window. Only 154 sites remain suitable for breeding. These are the hexagons that originally had (and still have) a score of exactly six. Only hexagons having a score less than six were modified in the preceding step. Select a few of these hexagons, using the middle mouse button, to confirm that they really do have a score of six. Examine some other hexagons as well. They should all have scores of zero. Notice that hexagons containing habitat are never painted black, even after their scores have been changed to zero.

This territory map has the same spatial distribution of breeding sites as was present in the original map, when the territory minimum and maximum were equal to the hexagon size. In both cases, only hexagons with scores of exactly six qualified as breeding sites. Now increase the upper end of the range of target hexagon scores to seven. The hexagon editor's range fields should read *0.0 <= Hex Score < 7.0*. Leave the *New Hexagon Score Is* field at zero, and click again on the button labeled *Assign Score To Hxgns In Range*. The territory map should no longer contain any sites suitable for breeding.

Click the middle mouse button on a hexagon in the analysis window. While the mouse button is held down, the habitat controls window footer displays the threshold score. This value is 5.333, and is equal to the minimum score a hexagon must achieve in order to be suitable for breeding. Hexagons can borrow habitat from their neighbors to reach the threshold score, but the amount that can be lent is limited by the maximum territory size.

Editing Hexagon Scores by Hand

Now change the value of the field labeled *New Hexagon Score Is* to 5.334. Click on the button labeled *Assign Score To Selected Hxgns* and move the mouse pointer into the analysis window. The analysis window header should change to read *CHANGE HEXAGON SCORES*. This indicates that PATCH is ready for you to modify hexagon scores by hand. Click left on any of the gray hexagons. They should turn blue, indicating they have been selected for modification. If you click on a blue hexagon, PATCH removes it from the list of sites to be modified and restores its original color. The scores of hexagons that do not contain any habitat (those painted black) can not be changed with this routine, or by using the *Assign Score To Hxgns In Range* button.

Next, move the mouse pointer out of the analysis window. PATCH applies the new score to all of the hexagons that were selected, and reruns the habitat sharing algorithm. The only breeding sites present in this modified landscape should be the sites that were just altered using the *Assign Score To Selected Hxgns* button. No other hexagons can achieve breeding status by borrowing habitat, since the *New Hexagon Score Is* field is set right at the threshold score.

Randomizing the Territory Map

Now type the name of the saved territory map into the habitat controls window input file location. If you used the file name suggested earlier in this example, then this means you should place "/tmp" into the directory field, and "test" into the file name field. Now click on the button labeled *Load All Files*. This loads the uncompressed territory map that was constructed previously.

Now click on the *Apply Hex Compression* button. When the process finishes, the entire compressed territory map should be displayed in the analysis window. Next, click on the button labeled *Randomize When Area > 0*. This randomizes the location of every hexagon that has at least some habitat. The black regions in the image do not change. Lastly, click on the button labeled *Randomize Every Hexagon*. This randomizes every hexagon in the territory map.

When a territory map is randomized, every hexagon's location changes, but the scores stay the same. However, the randomization process can change the number of territories present in a landscape because each hexagon's ability to share habitat is influenced by the quality of its neighbors. Click the *Randomize Every Hexagon* button a few more times and watch the analysis window footer. The number of territories present in the landscape should change with the landscape pattern. Now set the territory maximum to the hexagon area (90 pixels) and click the *Adjust Min & Max Sizes* button. Randomize the landscape a few more times and watch the analysis window footer. The number of territories should stay constant because habitat sharing has been precluded.

Changing the Resample Rate

Reset PATCH's fields to their default values by clicking on the *Read The Data File* button in the main window. The territory map is discarded. Go to the habitat controls window and increment the value of the *Hexagon Area* field using the increment or decrement widgets ( ) or the up and down arrow keys on the keyboard. Each time the hexagon size is increased to a new value, its size in hectares is displayed in the habitat controls window footer. You should observe a sequence of hexagon sizes beginning with 1.08 hectares at 12 pixels per hexagon. Next you should see 3.24 hectare hexagons composed of 36 pixels, then 8.10, 15.12, 24.30 hectares, and so on.

Now change the *Resample Rate* field in the main window to 2. Repeat the process of incrementing the hexagon size and observing the corresponding areas in hectares. The sequence described above changes so that it now progresses through 0.27, 0.81, 2.02, 3.78, and 6.08 hectares. You may want to repeat this process for other values of the resample rate parameter to see how the sequence of hexagon sizes is further modified.

The resample rate is designed to help you obtain a particular hexagon size (in hectares). Use of the resample rate field forces PATCH to divide each pixel in the GIS data set into an array of smaller subpixel units. When the resample rate is set to 2, each data pixel is divided into a 2-by-2 array of subpixels. A resample rate of 3 causes PATCH to split each pixel into a 3-by-3 array, and so on. Change the resample rate a few times and notice the number of rows and columns reported in the main window footer. These values grow in a predictable manner as the resample rate is increased.

With the resample rate set greater than one, open and display both the image and zoom windows. Now move the zoom-box until the image displayed in the zoom window includes an isolated pixel of a single color. Left click on this pixels to observe the effect of the resample rate parameter. Repeat this process one more time with an even larger resample rate.

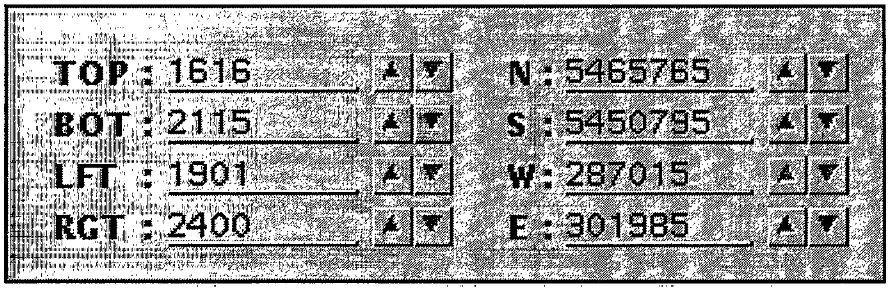
Example 3 Demographic Simulations

Introduction

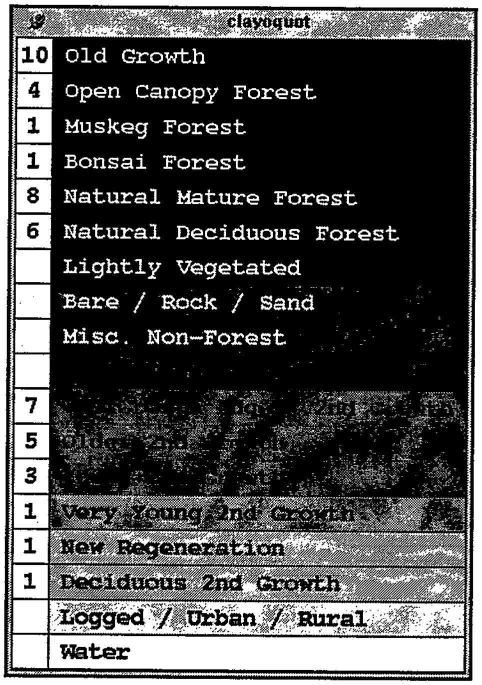
This example is designed to introduce PATCH's life history simulator. It is assumed that the reader has worked through the previous two examples and thus already possesses a basic understanding of the organization and use of the PATCH model. This example does not come close to illustrating every possible approach to working with PATCH's life history simulator. Many other applications remain for you to identify and explore. The utilities discussed here are flexible and much more can be done with them than is illustrated. For better or worse, the possibilities are pretty much limitless.

Getting Started

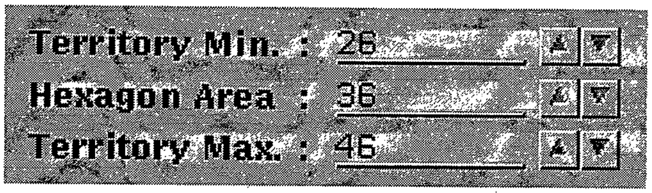
Begin by loading the clayoquot GIS data set (this is one of the sample images included with the PATCH distribution). Then set the zoom-box size to 500 × 500 pixels. If the zoom-box location has not been altered, then its top, bottom, left, and right edges (in pixels) should be equal to 1616, 2115, 1901, and 2400, respectively. If this is not the case, then make the appropriate changes to the *TOP*, *BOT*, *LFT*, and *RGT* fields in the main window.



Once the zoom-box is in place, call up the legend window and specify the habitat weights. Working from the top of the legend down to the bottom, set the weights to 10, 4, 1, 1, 8, 6, 0, 0, 0, 0, 7, 5, 3, 1, 1, 1, 0, 0. As was the case in the previous two examples, these legend weights are completely arbitrary. In a real investigation, you should specify the legend weights based upon field data that directly measure a species' preferences in different habitat types. An alternative approach that you can take if field data are not available is to survey experts and request that they rank the importance of available habitats for the species in question. Recall that the left mouse button increments the weighting values and the right button decrements them.



Next, set the hexagon size to 36 pixels, and the territory minimum and maximum to 26 and 46 pixels, respectively. Then build a territory map.



The territory map should contain 7056 hexagons, of which 2310 should be suitable for breeding. Click on the *Add Hxgn Border* button (found in the habitat controls window) and change the magnification level to 1 (found in the main window). Then call up the analysis window and double click in it once to resize it, and a second time to paint it. This displays the territory map that was just constructed.

Place the mouse pointer over one of the hexagons present in the analysis window. Select the hexagon by pressing and holding the middle mouse button. As long as the mouse button is held down, the hexagon is highlighted (painted white) and the analysis window footer displays its number, area in pixels, and score. When a hexagon is selected, the habitat controls window footer changes to display the threshold score associated with the territory map. The footer also displays the mean step size when a hexagon is selected.

Now that a territory map has been created, you can begin to use PATCH's life history simulator. But before going on, save the territory map you just built. Choose a directory and name for the map. Later on in the example, this territory map is referred to as "old_map", so you may want to specify that name now.

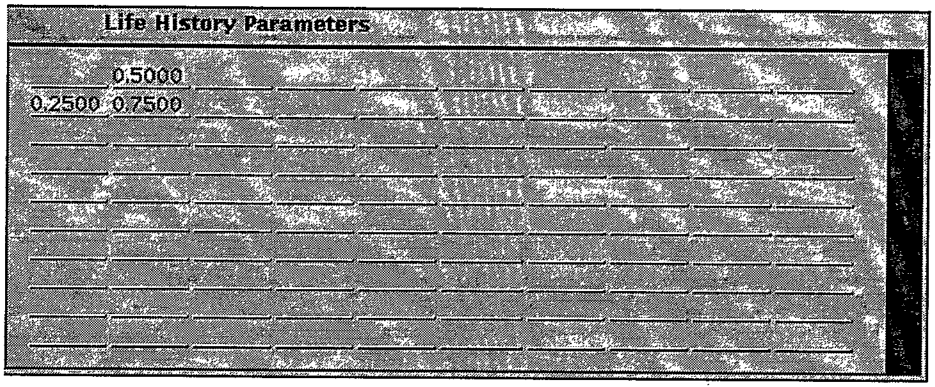
The Projection Matrix

If a life history parameters file already exists, you can simply upload the data and immediately run a simulation. Otherwise, it is best to begin preparing for a simulation by specifying the parameters in the life history parameters window. There are 10 parameter fields in this window, plus a 10×10 population projection matrix. PATCH's life history module simulates a post-breeding census, which means that its first age or stage class always consists exclusively of newly recruited individuals that are not yet of breeding age. Thus the upper left entry in PATCH's projection matrix is always fixed at zero.

Start preparing to run a simulation by entering a projection matrix into the panel at the right of the life history parameters window. Remember that the survival rates present in any column of PATCH's projection matrix must not sum to more than one. More importantly, this must also be true of the corresponding matrix used to make survival and reproductive decisions in the best habitats.

Enter the simple 2 × 2 population projection matrix, shown to the right, into the life history parameters window. You can think of this matrix as defining a two-stage system of juveniles and adults. According to this matrix, juveniles (who, by definition, do not reproduce) have a relatively high rate of mortality. Adults (who do reproduce at a mean rate of 0.5 females per female per year) experience a relatively low mortality rate. These vital rates were simply made up for this example.

$$\begin{bmatrix} 0 & 0.50 \\ 0.25 & 0.75 \end{bmatrix}$$



The Vital Rates Factor

The vital rates factor determines the quality of habitat to be associated with the survival and reproductive values you supply. Hexagons with scores of zero are always assigned survival and reproductive rates of zero. Hexagons having the score indicated by the vital rates factor are assigned the survival and reproductive rates entered into the model's projection matrix. PATCH uses the interpolation functions to find survival and reproductive rates for hexagons with other scores. As the vital rates factor is lowered, the vital rates associated with the best habitats increase. You can examine the projection matrix associated with the best habitats by holding down the middle mouse button in the panel containing PATCH's projection matrix. Of course, holding down the middle mouse button will not change the values displayed in PATCH's projection matrix unless the vital rates factor is set to a value less than 100%. Set the vital rates factor to 100% (the default value), but don't supply values to any of the other life history parameters present in the window.

Next call up the life history functions window by clicking on the yellow bar on the left (alternatively, right click anywhere in the window). Select the linear options for the survival and fecundity interpolation functions (these are the default options). Then bring back the life history parameters window. Provide an output file name in the life history controls window and click on the button labeled *Get Output Maps*. Pushing this button after a simulation has been conducted would generate three different image files. But two of these (".image0" and ".image2") are built from the results of a simulation and are not printed if a simulation hasn't been run (unless a ".stats2" file containing utility and density information has been loaded into the life history module).

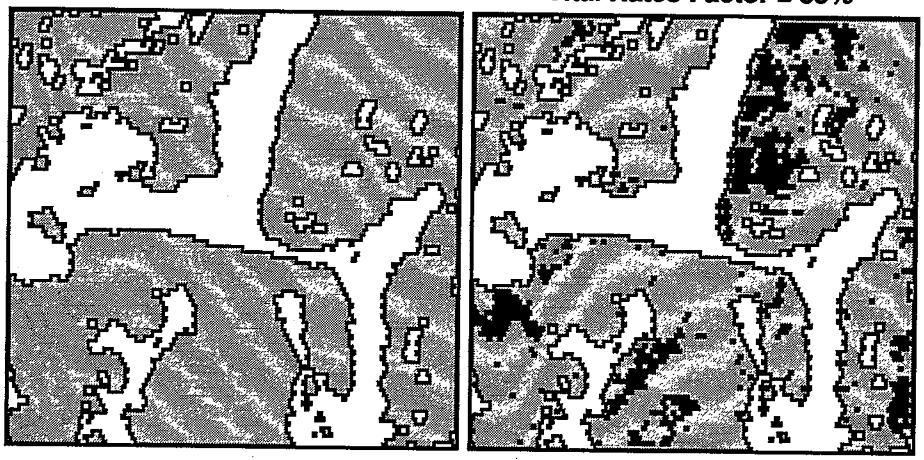
At present, you should obtain only a single image file ending in ".image1". This is the expected source-sink map. PATCH also prints out a ".stats2" file (unless one with the specified prefix already exists). This is where PATCH stores the raw data used to construct its output image files. Examine the expected source-sink map (".image1") that has just been created. You can do this by changing its name so that it ends in ".sun" and then reading it into PATCH. Alternatively, you can examine the file using other raster image viewing packages such as *XV*, *XloadImage*, or Sun's *snapshot* or *imagetool* programs.

Non-habitat is displayed in white on the source-sink map. Sites that have habitat but are not suitable for breeding are displayed in black. Breeding sites are displayed in red if they can be expected to behave as sinks and in green if they are expected to behave as sources. Notice that all of the breeding habitat displayed in this image is colored red, which alerts you that the result of combining the current territory map with the population projection matrix, the vital rates factor, and the interpolation functions specified above, is a landscape containing no source sites whatsoever. You can be confident that any model species is going to decline rapidly to extinction in a landscape containing only sinks. Remember that sources and sinks here are evaluated on a hexagon-by-hexagon basis.

If the species being modeled is not experiencing a rapid decline in numbers, then the above analysis should be sufficient to alert you that something is already amiss. Assume the survival and reproduction values being used have been taken from the literature and are reasonable for the species. Assume also that the GIS map is accurate and that the selection of habitat weights is appropriate. You may then conclude that the vital rates factor is set too high. This simple analysis thus suggests that if the model species is to have any chance at persistence, the survival and reproductive rates specified above must be realized in suboptimal habitat. When the vital rates factor is set to 100%, altering the interpolation functions does not improve the species' performance in the best habitats.

Now set the vital rates factor to 85%. This tells PATCH to associate the survival and reproductive rates entered above with hexagons that have scores of 85% of the maximum value of 10, or 8.5. As a result, the vital rates experienced (ignoring stochasticity) in hexagons with scores greater than 8.5 exceed those entered into PATCH's projection matrix. This will allow some of these sites to function as demographic sources.

Leave the interpolation functions as they are (both linear), and click and hold the middle mouse button on PATCH's projection matrix. The adult fecundity, juvenile survival, and adult survival should change to read 0.5882, 0.2941, and 0.8824 respectively. These are the survival and reproductive rates now realized in the highest quality habitats. Click on the button labeled *Get Output Maps* and examine the new copy of the expected source-sink map (".image1") that PATCH constructs. Many of the breeding sites previously classified as sinks (red) should now be classified as sources (green). This combination of parameters is more likely to produce a stable population.



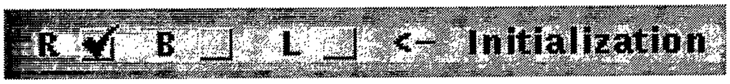
Runs, Years, and Utility

After supplying the projection matrix, specify the remaining parameters present in the life history parameters window. It is usually a good idea to begin by setting the number of runs to 1, and the number of years to a small value such as 10. Otherwise, if the model population begins growing rapidly, the simulation can take a long time to complete. Use the field labeled *Tally Utility From* to specify a year for PATCH to begin tracking the observed source-sink data. It is reasonable to begin by setting this field to zero.

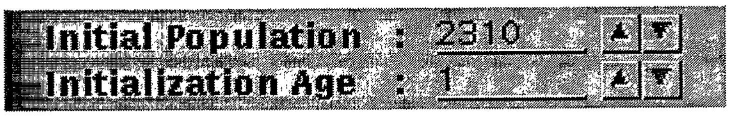
The life history parameters selected for this example are designed to produce simulations that progress relatively quickly. So for the time being, ignore the cautions mentioned above and set the number of runs to 10, the number of years to 200, and the *Tally Utility From* field to 101. This instructs PATCH to perform 10 replicate runs of 200 years each, and to tally the observed source-sink properties of each breeding site for the final 100 years of each simulation.

The Initial Population Size

The initial population size can be set with the *Initial Population* field. But this parameter must be used in conjunction with the *Initialization* check-box present in the life history window. If the initialization check-box is set to *R* (random) or *B* (best neighborhoods), then you can directly enter the initial population size. However, if the Initialization check-box is set to *L* (locked), then the *Initial Population* field is frozen and you must manually add or remove members of the initial population by clicking in the analysis window. It is easiest to begin a simulation with the initialization check-box set to *R* or *B*. Manual placement of the initial population is not discussed in this example.



The *Initialization Age* field determines the age or stage class of the initial population. Remember that age and stage classes in PATCH are numbered from 0 to 9. For example, a 3 × 3 projection matrix corresponds to stage classes numbered 0, 1, and 2. Set the initialization check-box to *R* (the default value) and the initial population size to 2310. This equals the number of breeding sites, and is therefore the largest possible initial population size. Set the initialization age to 1. This instructs PATCH to construct the initial population from adult (as opposed to juvenile) organisms.

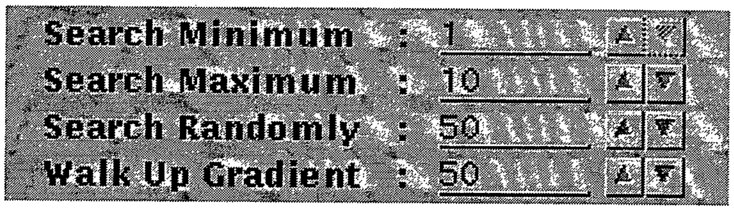


Movement Parameters

The search minimum specifies the minimum number of steps that must be taken before an available breeding site can be settled. This field is often left at its default value of 1 (its lowest possible value). The search maximum limits the total distance that can be traveled. The search minimum and maximum are specified as a number of steps from a hexagon to one of its immediate neighbors. This step size is displayed in kilometers in the habitat controls window footer when a hexagon is selected (using the middle mouse button) in the analysis window. This mechanism is provided to help you translate actual dispersal estimates into numbers of steps within a territory map.

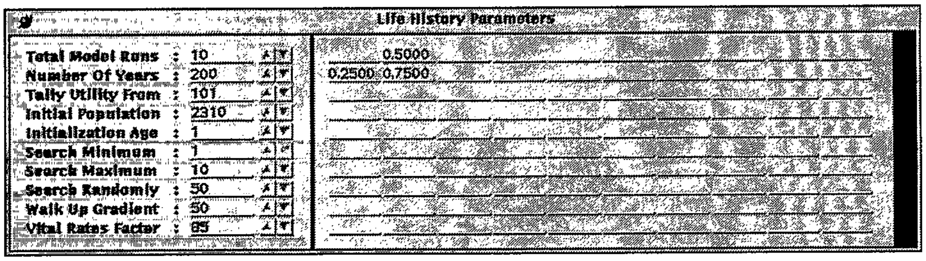
The parameters labeled *Search Randomly* and *Walk Up Gradient* are used only when the search strategy involves a random walk. But remember that a random walk is always taken by individuals that fail to locate an available breeding site using the alternative search strategies. Thus the Search Randomly and *Walk Up Gradient* parameters should be specified regardless of the search strategy being implemented. Both of these parameters range between 0 and 100%. If the *Search Randomly* parameter is set to 0, movement paths become straight lines. If this parameter is set to 100%, the choice of movement direction is made randomly. Movement paths vary continuously from linear to completely random as this parameter is increased from 0 to 100%.

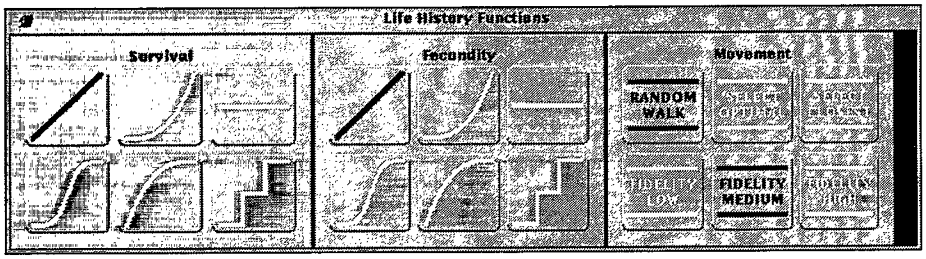
When the parameter labeled *Walk Up Gradient* is set to 0, individuals taking a random walk ignore habitat quality all together. The character of the random walks is then controlled entirely by the *Search Randomly* parameter. When the parameter labeled *Walk Up Gradient* is set to 100%, individuals always attempt to move to the best neighboring hexagon (as long as it is better than the current site). As this parameter is increased from 0 to 100%, searchers exhibit an increased propensity to move up gradient to better habitats. Decisions based on the *Walk Up Gradient* parameter always take precedence over decisions based on the *Search Randomly* parameter. For this example, set the search minimum and maximum to 1 and 10, respectively. Then set the *Search Randomly* and *Walk Up Gradient* parameters both to 50% (their default values).



Life History Functions

Values have now been assigned to all of the fields in the life history parameters window. Next call up the life history functions window by clicking on the yellow bar that appears to the left of the projection matrix (alternatively, right click anywhere in the window). The life history functions window lets you specify the interpolation functions for survival and reproduction, the movement strategy, and the level of site fidelity. The interpolation functions for survival and reproduction were both set to the linear option earlier in this example. Leave these the way they are, but set the movement routine to *Random Walk* and the site fidelity to medium (these are both the default settings).

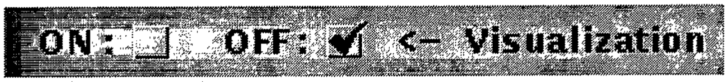




This forces every individual to use a random walk when searching for an available breeding site. In addition, setting the site fidelity to medium obligates territorial individuals occupying sinks to search yearly for a new breeding site. Territorial individuals occupying sources remain on their sites indefinitely. When the site fidelity parameter is set to low, every territorial individual searches yearly for a new site. When it is set to high, all territorial individuals remain on their sites indefinitely.

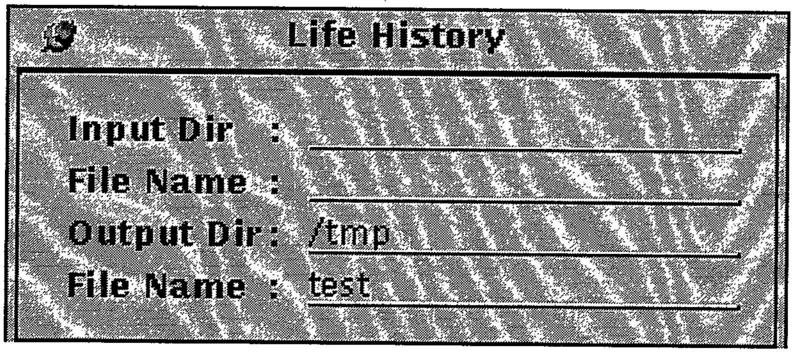
Visual Feedback

PATCH can display the locations and movements of every member of the simulated population in the analysis window. This visual feedback is informative, but the process of displaying it makes the model run much more slowly. You can turn these graphics off and on using the *Visualization* check-box present in the life history window. It is usually best to begin with the visualization check-box off.



Specifying an Output File

The last thing to consider before running a simulation is whether or not to save the results to a disk file. The life history window contains output file name fields that can be used to save the life history parameters and model results. This window also contains input file name fields that can be used to upload life history parameter files that have previously been saved. It is usually not necessary to enter an output file name in the early stages of a simulation. PATCH simply sends the output data directly to the computer's monitor. However, the life history simulator can generate seven additional files (".stats0", ".stats1", ".stats2", ".stats3", ".image0", ".image1", and ".image2"), that are only constructed if an output file name is supplied. For now, select an output directory and file name and enter them in the life history window. Let's assume you have selected the file name "/tmp/test" for this purpose.

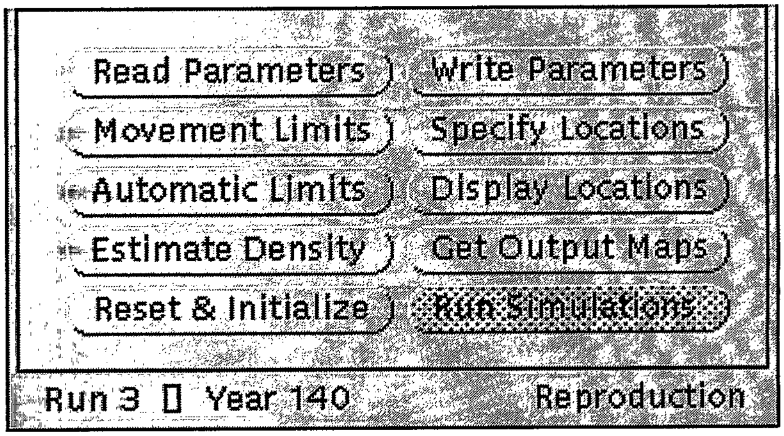


Running a Simulation

Before starting a simulation, call up and paint the analysis window. Then go to the life history window and push the *Reset & Initialize* button. The life history window footer should change to read *Selected Initial Sites.* If it does not do this, push the reset button again. Next click on the button labeled *Display Locations.* This displays every member of the initial population in the analysis window. These individuals are all territorial, and are shown in red. A territorial individual should appear in every breeding site because the initial population was set equal to the number of breeding sites. Check that the vital rates factor is at 85%. If this parameter is mistakenly set to 100%, the population can be expected to die out in about 50 years.

Now push the button labeled *Run Simulations.* This starts a model run. The left side of the life history window footer displays the run number and year as the simulation progresses. The right side of the footer displays the function currently being implemented. The simulation results are sent to the specified output file, or to the UNIX command window from which PATCH was called up.

If the visualization check-box is set on and the analysis window is up to date, you can observe the simulation superimposed on the territory map. Dispersing juveniles are displayed as white lines. The movements of floaters, if there are any, are displayed as yellow lines. Floaters themselves are displayed as yellow borders surrounding the hexagons in which they are located.

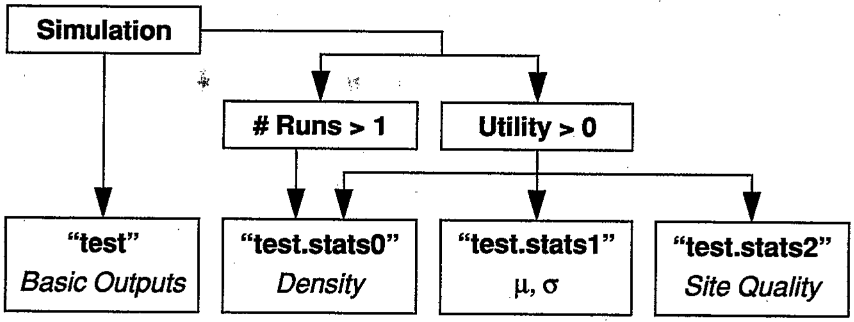


The Output Statistics

When a simulation begins, PATCH constructs two output files (as long as an output file name has been specified in the life history window). These include the main life history output file and a file ending in ".stats0". The main life history file contains the basic model output, tallied on a run-by-run and year-by-year basis. The ".stats0" file contains breeding site occupancy data designed for use with the *Estimate Density* button. A ".stats0" file is created only if the number of runs is greater than 1 or the *Tally Utility From* parameter exceeds 0.

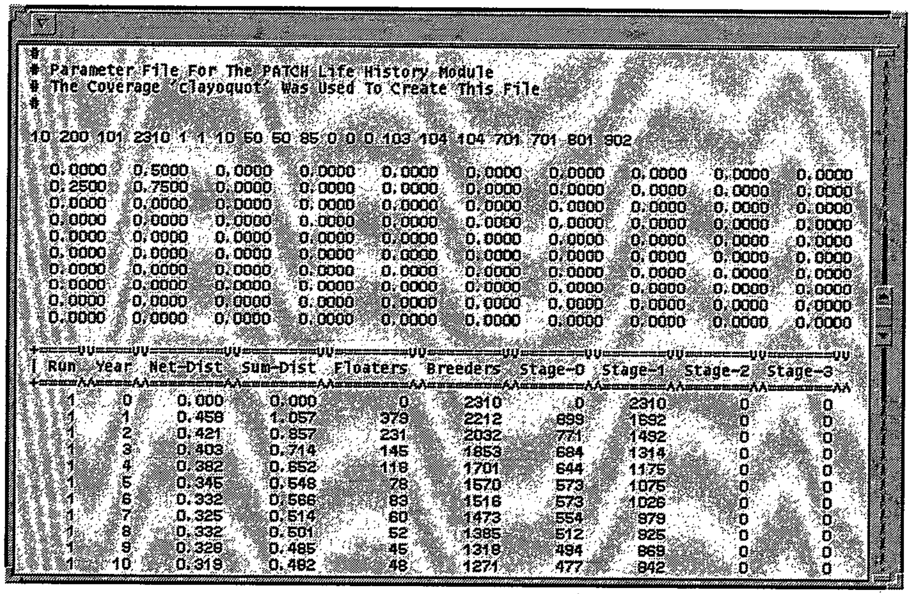
If the *Tally Utility From* parameter is greater than 0, then when a simulation is complete, PATCH creates two additional output files. These files end in ".stats1" and ".stats2". The ".stats1" file contains the means and standard deviations of various measures of the population, taken over every replicate model run. This file also contains the year-by-year effective vital rates and values of lambda (the dominant eigenvalue of the corresponding projection matrix). The ".stats2" file contains the expected and observed source-sink data and additional occupancy rate information. The ".stats2" file includes data only on hexagons suitable for breeding.

The figure below provides a list of the output data files that PATCH creates when a life history simulation is performed. As few as one, and as many as four of these files are constructed when the model is run.



The Main Life History File

Assuming the above naming scheme was used, the file named "test" contains the parameters provided to the life history module, and the simulation results on a run-by-run and year-by-year basis. This file can be used to display the net and sum movement distances, the number of floaters and breeders, and the number of individuals in each stage class.



```
#
# Parameter File For The PATCH Life History Module
# The Coverage 'clayoquot' Was Used To Create This File
#

10 200 101 2310 1 1 10 50 50 85 0 0 0 103 104 104 701 701 801 902

  0.0000  0.5000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.2500  0.7500  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
```

Run	Year	Net-Dist	Sum-Dist	Floaters	Breeders	Stage-0	Stage-1	Stage-2	Stage-3
1	0	0.000	0.000	0	2310	0	2310	0	0
1	1	0.458	1.057	379	2212	899	1692	0	0
1	2	0.421	0.857	231	2032	771	1492	0	0
1	3	0.403	0.714	145	1853	684	1314	0	0
1	4	0.382	0.652	118	1701	644	1175	0	0
1	5	0.345	0.548	78	1570	573	1075	0	0
1	6	0.332	0.566	83	1516	573	1026	0	0
1	7	0.325	0.514	60	1473	554	979	0	0
1	8	0.332	0.501	52	1385	512	925	0	0
1	9	0.329	0.485	45	1318	494	869	0	0
1	10	0.319	0.492	48	1271	477	842	0	0

The ".stats0" File

The file called "test.stats0" contains a record indicating whether or not each breeding site was occupied at the end of the final year of every replicate simulation. The ".stats0" files are written in a binary format, and because of this you cannot read them directly. PATCH accesses the data in a ".stats0" file when the button labeled *Estimate Density* is used. When this button is pushed, PATCH extracts the density information corresponding to the breeding sites currently enclosed in the zoom-box, and it generates a list showing the number of breeders present in that region at the end of each simulation. PATCH highlights the breeding sites being used in the calculation so you can make sure that the occupancy data being gathered actually correspond to the intended region. When this utility is used, the appropriate file name must appear in the life history window's input file name field.

The ".stats1" File

The file called "test.stats1" can be used to display the yearly means and standard deviations of the population size, the number of floaters and breeders, and the number of individuals in each stage class. This file also records the mean effective survival and reproductive rates, presented as elements of a projection matrix. PATCH also records the dominant eigenvalue (λ) corresponding to each effective projection matrix. Each value in a ".stats1" file is computed using the entire collection of replicate runs generated during a simulation.

Year	PopSize + StdDev		Floater + StdDev		Breeder + StdDev	
0	2310.00	0.00	0.00	0.00	2310.00	0.00
1	2538.30	38.13	349.00	25.05	2189.30	15.68
2	2229.00	37.34	216.50	25.68	2012.50	24.12
3	1993.20	46.97	144.70	25.28	1848.50	24.71
4	1815.20	37.78	101.70	21.39	1713.50	24.57
5	1673.40	25.00	80.90	14.48	1592.50	24.33
6	1598.40	22.53	70.60	6.70	1527.80	22.52
7	1521.30	18.93	55.90	10.50	1465.40	15.71
8	1448.00	29.86	48.00	13.48	1400.00	22.92
9	1402.70	34.81	48.30	13.01	1354.40	26.60
10	1341.30	35.43	38.90	12.09	1302.40	30.65

Stage-0 + StdDev		Stage-1 + StdDev		[0][0]	[0][1]	[1][0]	[1][1]	Lambda
0.00	0.00	2310.00	0.00	0.0000	0.5000	0.2500	0.7500	0.8904
859.20	25.95	1679.10	20.53	0.0000	0.3719	0.0000	0.7269	0.7269
767.00	20.32	1462.00	24.54	0.0000	0.4568	0.2002	0.7683	0.8730
695.10	30.76	1298.10	26.38	0.0000	0.4754	0.2042	0.7808	0.8899
635.90	22.03	1179.30	18.35	0.0000	0.4899	0.2168	0.7924	0.9092
590.00	15.46	1083.40	15.59	0.0000	0.5003	0.2206	0.7997	0.9197
571.80	9.34	1026.60	16.93	0.0000	0.5278	0.2290	0.8229	0.9501
541.00	14.05	980.30	13.94	0.0000	0.5270	0.2385	0.8220	0.9538
517.40	15.49	930.60	21.13	0.0000	0.5278	0.2292	0.8228	0.9501
506.30	17.43	896.40	24.17	0.0000	0.5441	0.2348	0.8327	0.9651
476.00	15.53	865.30	24.05	0.0000	0.5310	0.2285	0.8362	0.9623

The ".stats2" File

The file called "test.stats2" can be used to display the breeding site scores, their expected and observed source-sink properties, and their observed occupancy rates. Nonbreeding sites do not appear in ".stats2" files. The observed source-sink properties are compiled for every year of every replicate run that is greater than or equal to the value of the *Tally Utility From* parameter. In contrast, the observed occupancy rates are gathered strictly from the population distributions present at the final model year of each replicate simulation.

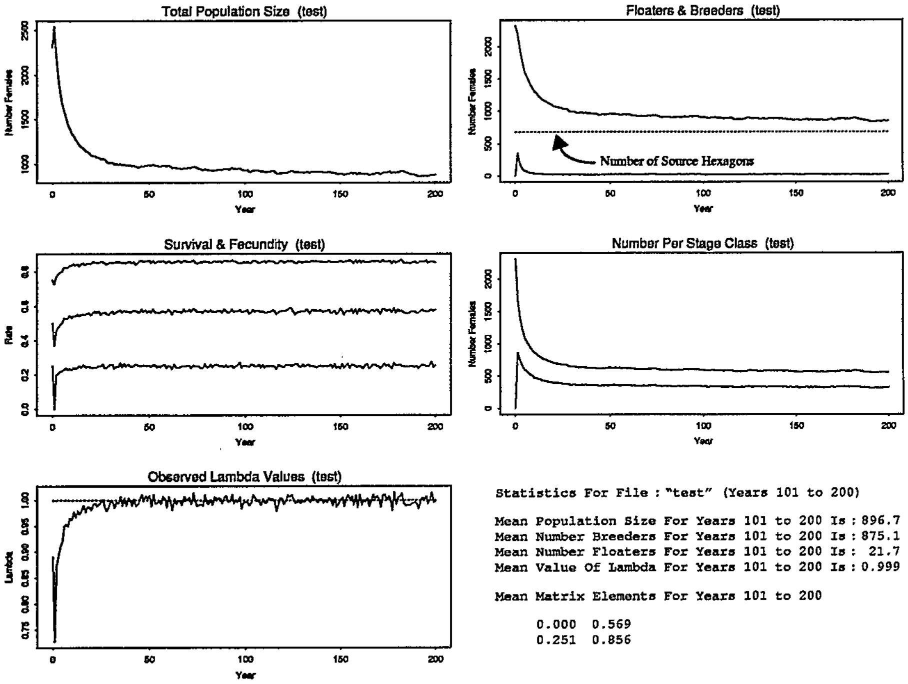
The first two columns in a ".stats2" file contain identifying numbers unique to each hexagon. PATCH uses the numbers in the first column when a ".stats2" file is uploaded into the life history module. You can ignore these values. The numbers in the second column correspond to the numbers displayed in the analysis window footer when a hexagon is selected (using the middle mouse button). The next two columns contain the hexagon scores and lambda values. The lambda values are computed from the scores using the life history parameters you supply, and they indicate each hexagon's expected source-sink behavior. The next column contains utility values, which describe each hexagon's observed source-sink behavior. The last column in a ".stats2" file displays the number of times each hexagon was occupied by a breeder on the final year of each replicate model run.

Patch-#	Hex-#	Scores	Lambda	Utility	Occupancy
17	20	8.3333	0.8729	-253	5
18	21	8.0000	0.8380	-88	3
19	22	8.3333	0.8729	-138	3
20	23	8.3333	0.8729	-74	3
21	24	8.3333	0.8729	-38	0
22	25	6.7222	0.7042	-30	0
35	44	8.0833	0.8372	0	0
40	58	8.1667	0.8555	-127	4
41	59	8.3333	0.8729	-327	5
42	60	8.3333	0.8729	-250	4
43	61	8.3333	0.8729	-313	6
44	62	8.3333	0.8729	-260	5
45	63	8.3333	0.8729	-283	8
46	64	7.7500	0.8118	-273	2
47	65	8.0000	0.8380	-311	3
48	66	8.3333	0.8729	-188	4
49	67	8.3333	0.8729	-163	7
50	68	8.3333	0.8729	-250	5
51	69	8.0833	0.8467	-157	4
52	70	8.1667	0.8555	-150	2
53	71	4.7500	0.4976	-92	2
54	72	5.5000	0.5761	-69	2
55	73	8.3333	0.8729	-59	1
56	74	8.3333	0.8634	-13	0
57	75	8.1111	0.8401	-2	0
58	76	7.7778	0.8147	0	0
59	77	7.6667	0.8031	0	0
86	104	9.5556	1.0010	83	5
87	105	9.7500	1.0213	223	9
88	106	9.5000	0.9951	-194	7
89	107	10.0000	1.0476	342	9
90	108	9.1667	0.9602	-83	2
103	128	9.1667	0.9602	0	0
104	129	9.0000	0.9428	0	0
107	142	8.8889	0.9311	-265	6
108	143	10.0000	1.0476	333	9
109	144	10.0000	1.0476	378	9
110	145	10.0000	1.0476	325	9

Displaying the Output Data

PATCH does not generate graphical displays of any of its output data. Instead, a S-PLUS macro called "plot" has been provided specifically for this purpose. S-PLUS is a statistics package produced by MathSoft, Inc. The S-PLUS macro reads and plots all of the data present in the main life history output file as well as the ".stats1" and ".stats2" files.

The S-PLUS macro is complex, but it is easy to use. It prompts you for all of the information it needs. You must open a graphics window in S-PLUS (e.g., issue a *motif* command) before using the macro. If possible, use the S-PLUS macro to view the results from the simulation you just conducted. The S-PLUS macro accesses the data stored in the ".stats1" file if you ask it to plot replicate number 0. Otherwise, it displays only the data corresponding to a specific model replicate (data contained in the main life history output file). The S-PLUS macro is also useful for generating data sets that can be used to introduce environmental stochasticity into PATCH's simulations.



The Output Imagery

Next, push the button in the life history window labeled *Get Output Maps*. This makes PATCH print out three output image files, assuming that the appropriate data have been compiled during the simulation, and also that a file name has been specified in the life history window. These three image files end in ".image0", ".image1", and ".image2".

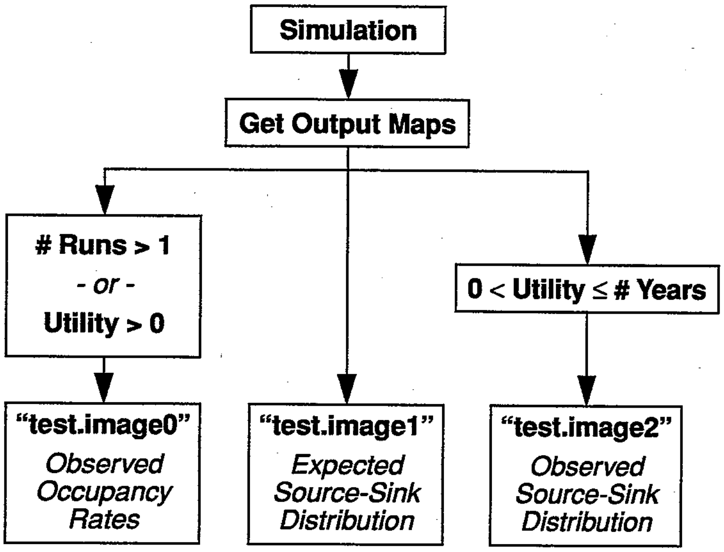
The Occupancy Rate Map

The ".image0" file displays the occupancy rate of breeding habitat throughout the landscape. The data displayed in the occupancy map are gathered during the final year of each replicate simulation. However, this information is compiled only if the number of runs conducted was greater than 1 or the *Tally Utility From* parameter was set greater than 0. In the construction of this map, the occupancy rate data are rescaled to lie in the range from 0 to 10. The resulting occupancy rates are then displayed using a color spectrum that shifts from red to blue. The least used sites appear in pure red, and the most frequently used sites appear in pure blue. The data used to create this image are essentially a summary of the information stored in the ".stats0" file. But PATCH does not consult the ".stats0" file when it constructs a ".image0" map.

The Source-Sink Maps

The ".image1" and ".image2" files contain the expected and observed source-sink maps. As discussed earlier, the ".image1" file (the expected source-sink map) can be generated before a simulation has been run. In contrast, the ".image2" file (the observed source-sink map) can be created only after a simulation has been performed. But the data used to construct the ".image2" file will be compiled only if the *Tally Utility From* parameter is set greater than 0 and less than or equal to the number of simulation years. All sinks in these maps are painted red, and all sources are painted green. But you can alter this color scheme because both images contain 100 sink and 100 source classes.

The figure below provides a list of the output image files that PATCH can construct before and after a life history simulation is performed.



The occupancy and source-sink data are stored in the ".stats2" file, and can be uploaded along with a life history parameters file using the *Read Parameters* button. This feature can be used to generate new copies of the three output images at any time after a simulation has been conducted. To rebuild the image files, you need only load a life history output file paired with an existing ".stats2" file. When a ".stats2" file is successfully imported, the life history window footer adds the text *+ stats2* to the message that would otherwise be displayed. At this point, you can click on the button labeled *Get Output Maps* to generate new copies of PATCH's output images.

Editing Output Imagery

PATCH's source-sink maps are designed to be customized. The expected and observed source-sink maps both use the same color scheme. Sources and sinks are broken into 100 classes each, with every sink initially painted red, and every source painted green. You can alter this color scheme to highlight the best sources or the worst sinks. A key to this color scheme is included with PATCH's sample GIS data in the files "source_sink" and "source_sink.sun". The control file ("source_sink") can be copied and used with one of PATCH's output source-sink images. Because the control file has been designed for this purpose, the resulting legend categories are meaningful.

A second option exists for customizing an observed source-sink map. The key to this method lies in the fact that PATCH can upload a ".stats2" file along with a life history parameters file, and the data stored in it can then be used to generate a new observed source-sink map. The observed source-sink map can be customized by editing the ".stats2" file so that it contains only the information you want to display in the ".image2" file. Hexagons corresponding to lines removed from the ".stats2" file are considered neutral and painted blue. Using this method, you can select specific hexagons or groups of sources and sinks to be highlighted in the observed source-sink map.

Environmental Stochasticity

PATCH allows you to build environmental stochasticity into its demographic simulations. You accomplish this by constructing a table of vital rates that reflect the level of environmental stochasticity to be simulated. PATCH can upload the data in this table as long the table is arranged correctly and contained in a file that has an appropriate name. Each column of the table holds a specific survival or reproductive rate. Every row of the table must contain the entire set of survival and reproductive rates necessary to construct a projection matrix.

Matrix elements that have been set to zero in the model interface cannot be represented in the table of vital rates, and as such they remain zero throughout a simulation. Nonzero elements in the matrix are read from top to bottom and left to right, and (in this order) they define the columns of the table used to simulate environmental stochasticity. Each year a simulation is run, a row from the table is selected at random and used to build the projection matrix for the model species. Remember that survival rates present in any column of PATCH's projection matrix must sum at most to one, and this is also true of the matrix used for survival and reproductive decisions in the best habitats. Every matrix contained in a table of vital rates used to simulate environmental stochasticity is subject to these constraints. If any of the matrices fail to meet these conditions, the entire table is discarded.

PATCH expects a table of vital rates to have the same name as a life history parameter file, except that it must end in ".random". If a table of vital rates is successfully loaded (using the *Read Parameters* button), PATCH indicates this in the life history window footer by adding *+random* to the message that would otherwise be displayed. You can practice constructing and using tables of vital rates by building these data sets by hand. A valid table of vital rates may have as little as one line. There is no upper limit to the number of lines a table may contain.

The S-PLUS plotting program included with the PATCH distribution includes a flexible tool for constructing tables of vital rates. If you enter the text “random” when the S-PLUS macro asks for the replicate run number, the program’s vital rates generator is initiated.

Add environmental stochasticity to the simulation conducted earlier in this example by constructing a table of vital rates. If the life history output file created earlier was named “test”, then this table should be called “test.random”. The table of vital rates has three columns corresponding to the three nonzero entries used in PATCH’s projection matrix. The first column specifies the adult fecundity. The second and third columns specify the juvenile and adult survival rates, respectively. A copy of the table of vital rates intended to be used with this example is displayed below. This table is not intended to be realistic.

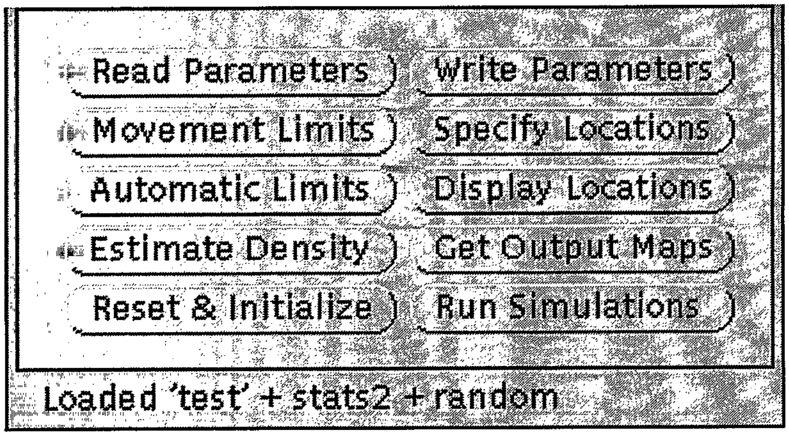
0.25	0.00	0.50
0.30	0.05	0.55
0.30	0.05	0.55
0.35	0.10	0.60
0.35	0.10	0.60
0.35	0.10	0.60
0.40	0.15	0.65
0.40	0.15	0.65
0.40	0.15	0.65
0.40	0.15	0.65
0.45	0.20	0.70
0.45	0.20	0.70
0.45	0.20	0.70
0.45	0.20	0.70
0.45	0.20	0.70
0.50	0.25	0.75
0.50	0.25	0.75
0.50	0.25	0.75

continued on right...

continued from left...

0.50	0.25	0.75
0.50	0.25	0.75
0.50	0.25	0.75
0.55	0.30	0.80
0.55	0.30	0.80
0.55	0.30	0.80
0.55	0.30	0.80
0.55	0.30	0.80
0.60	0.35	0.85
0.60	0.35	0.85
0.60	0.35	0.85
0.60	0.35	0.85
0.65	0.40	0.85
0.65	0.40	0.85
0.65	0.40	0.85
0.70	0.45	0.85
0.70	0.45	0.85
0.75	0.50	0.85

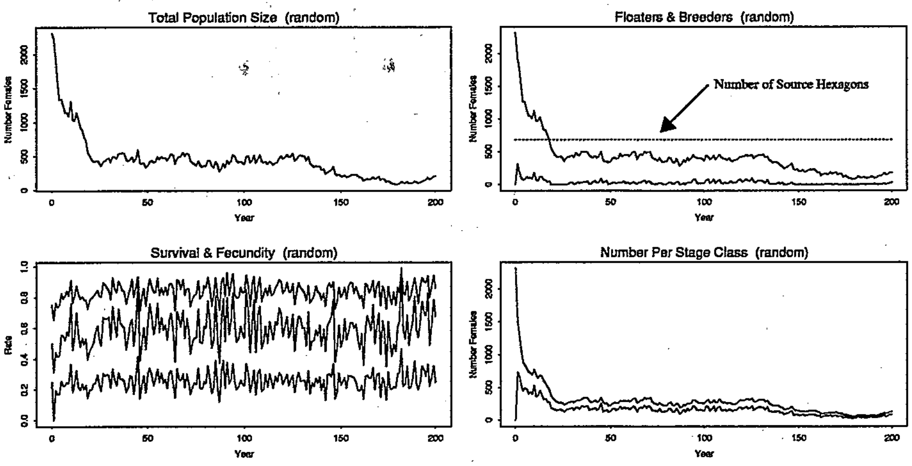
After constructing the table of vital rates, load it and the existing demographic output files into PATCH’s life history module by clicking the *Read Parameters* button. But first, be sure to type the correct name into the life history window input file name field. Because a simulation has already been conducted, a “.stats2” file should be present with the same prefix as the “.random” file that was just created. PATCH attempts to upload this file, along with the main life history parameter file and the table of vital rates. If PATCH successfully reads these files, a corresponding message appears in the life history window footer.

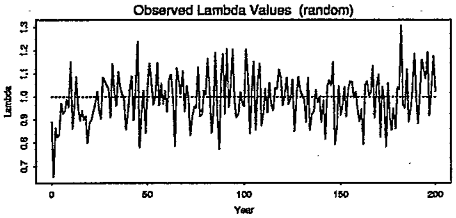


Select a new name to be used for the life history output files that include environmental stochasticity. Let's assume here that you have chosen the name "random" for this purpose. Enter the new name in the life history window output file name field. Assuming the *Read Parameters* operation was successful, PATCH is now ready to conduct a new simulation in which everything is the same as before except for the addition of environmental stochasticity. Start this new simulation by clicking the button labeled *Run Simulations*.

At the beginning of each model year, PATCH selects a row at random from the table of vital rates. This information is used to construct a projection matrix for that year, which is then placed into PATCH's graphical interface. As a result, the vital rates displayed in PATCH's projection matrix change constantly as a simulation progresses. This feature makes it easy to verify the presence of environmental stochasticity in the model. When the simulation has finished, compare the statistics files to those generated without environmental stochasticity. A great deal more variability should be present in the model outputs, and the various measures of population size most likely have decreased as a result.

Environmental variation in PATCH's vital rates is always going to be synchronized across a landscape. This simplification becomes less realistic as the size of your landscape increases. You can work around this spatial synchronization by creating a time series of territory maps that exhibit the patterns and ranges of environmental variability desired. The time series of territory maps can then be substituted for PATCH's traditional approach to environmental stochasticity (the use of ".random" files), or the two techniques can be combined.






```
Statistics For File : "random" (Years 101 to 200)

Mean Population Size For Years 101 to 200 Is : 278.4
Mean Number Breeders For Years 101 to 200 Is : 260.1
Mean Number Floaters For Years 101 to 200 Is :  18.3
Mean Value Of Lambda For Years 101 to 200 Is : 1.000

Mean Matrix Elements For Years 101 to 200

     0.000  0.598
     0.261  0.843
```

Including Habitat Change

A principal feature of the life history module is its ability to simulate landscape change through time. PATCH models landscape change by loading different territory maps at specified points in time. A tool called the time series editor is available in the life history window for this purpose. The time series editor allows you to create a schedule of territory maps to be installed at specific times during a simulation. The process of loading territory maps during a simulation works correctly whether or not multiple replicate runs are being conducted. However, if the number of replicates is greater than one, PATCH insists that you install a territory map at year zero. PATCH loads the year zero landscape at the beginning of each new replicate and inserts the initial population into this landscape.

Any reflecting boundaries that have been entered into the landscape (or uploaded along with a life history parameters file) are replaced each time a new territory map is installed. However, hexagons suitable for breeding cannot be made into reflecting boundaries. Thus, if suitable breeding sites are gained when a new territory map is installed, then it is possible to lose some reflecting boundaries. On the other hand, if suitable breeding sites are lost, reflecting boundaries may appear where they previously did not exist. This feature can be used to simulate the creation or destruction of obstacles to movement.

In order to take advantage of PATCH's ability to simulate landscape change, you must first create a sequence of territory maps. You have at least two options for accomplishing this. PATCH's habitat controls window contains a hexagon editor. With this tool, you can quickly generate a series of territory maps that illustrate any number of subtle or dramatic landscape changes. The resulting territory maps may then be imported directly into PATCH's time series editor. The second option for creating a time series involves editing the GIS imagery and then rebuilding the territory maps. A GIS image can always be modified using the program that created it. But it is also possible to modify a GIS image from within PATCH. You can change the habitat class assigned to a pixel, a patch, or a collection of patches. These utilities are straightforward to use, but they work only on GIS files that have UNIX write permission.

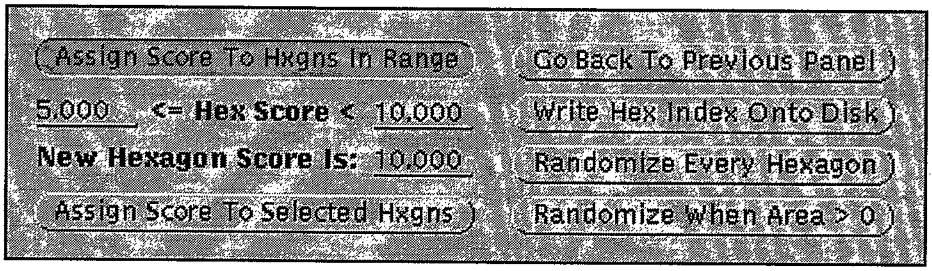
To edit individual pixels, first identify the class in the legend window that is to be applied to the modified cells. Next, click the middle mouse button on this legend window category. If the GIS file is writable, the editing arrow appears in the box next to the specified legend category. Once the editing arrow has been placed, you can modify individual pixels by left clicking on them in the zoom window. Any pixel selected in this way is changed to the legend class indicated by the editing arrow. The editing arrow can be relocated as often as desired, and the process continued. Click either the right or left mouse button anywhere in the legend window to remove the arrow and end the editing session.

To edit patches, a patch map must first be constructed. You can then place the editing arrow in a legend category and left click on any of the patches present in the analysis window. Each time a patch is selected, every pixel in it is changed to the habitat class specified by the editing arrow. You can also automatically modify every habitat patch present in the landscape by clicking on the button labeled *Alter Pixels By Patch.* This button is in the hidden panel at the top of the main window.

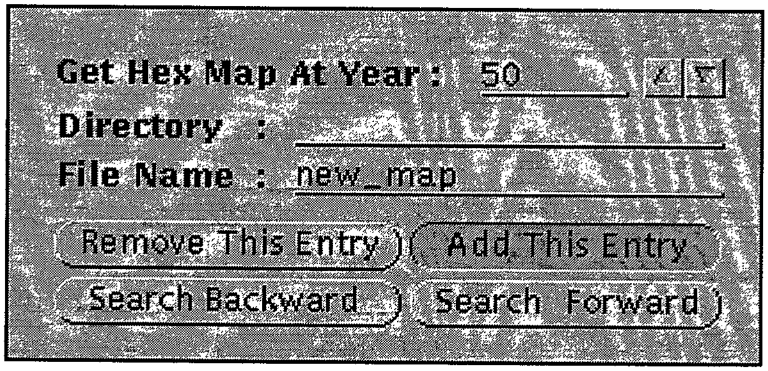
This example now illustrates the use of the hexagon editor for creating new territory maps. Recall that a territory map was constructed in the beginning of this example, and it has been used in all of the simulations conducted thus far. Let's assume here that this original territory map was named "old_map" (i.e., it consisted of the files "old_map.hexgn", "old_map.index", and "old_map.patch"). If this territory map has not been loaded, do so before proceeding.

The legend weights used in constructing the original territory map ranged between 0 and 10. All of the hexagon scores therefore also fall within this range. Call up the hexagon editor by clicking on the *View Next Panel* button located in the center of the habitat controls window. Alternatively, right click anywhere in the habitat controls window middle panel. Set the lower end of the range of target hexagon scores to

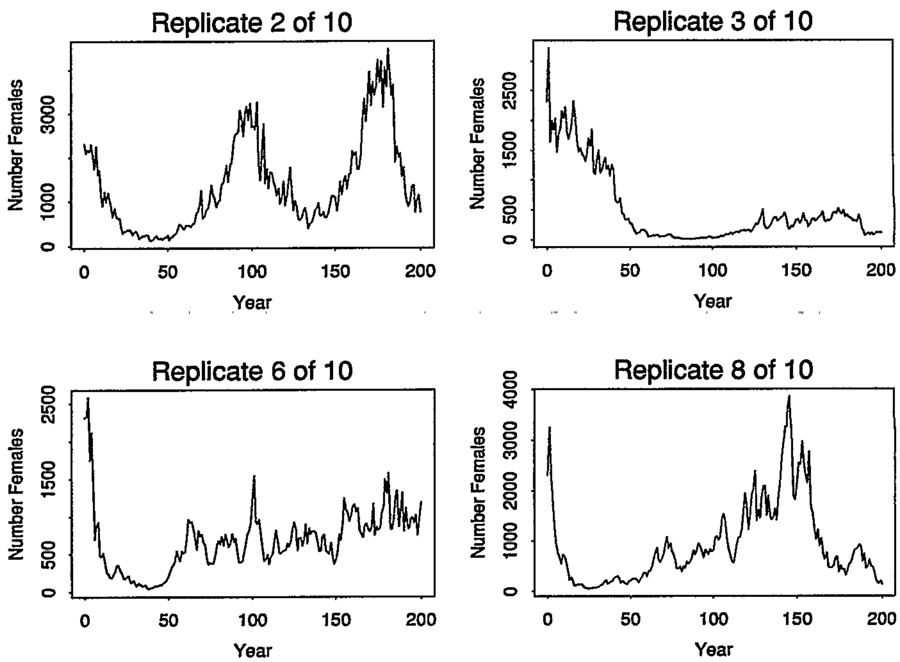
5, and the upper end to 10. Then set the *New Hexagon Score Is* field to 10. Next, make sure that the entire territory map is visible in the analysis window. You accomplish this by clicking on the button in the habitat controls window labeled *Add Hxgn Border* (the *Set To Data Edge* button also suffices). Then click on the button labeled *Assign Score To Hxgns In Range*. This causes PATCH to identify every hexagon contained entirely within the zoom-box that has a score between 5 and 10. It assigns each of these hexagons a new score of 10. This modified territory map should contain 2658 breeding sites.



Next, save the resulting territory map using the name "new_map". Do so by entering this name in the habitat controls window output file name field and then pushing the button labeled *Save All Files*. Many of the hexagons in the territory map named "new_map" have higher scores than their counterparts in the map named "old_map". Now upload the demographic output files generated earlier ("test", "test.stats2", "test.random") into PATCH's life history module. Make sure that PATCH successfully reads the table of vital rates ("test.random"). Then insert the original territory map, presumably named "old_map", into the life history window time series editor at year 0. Next, insert the territory map named "new_map" into the time series editor at year 50. To add an entry into the time series editor, first set the year, then specify the directory and file name information, and finally click on the *Add This Entry* button. PATCH provides all of the relevant feedback in the life history window footer.



Leave all of the other life history parameters as they are. Lastly, enter a new file name in the life history window output file name field. Let's assume here that you have selected the name "old_+_new" for this purpose. Then begin the new simulation by clicking on the button labeled *Run Simulations*. The new model output incorporates both environmental stochasticity and landscape change. In the absence of environmental stochasticity, the model population could be expected to improve dramatically after the new landscape is installed at year 50. However, a considerable amount of environmental stochasticity is represented in the table of vital rates created earlier. As a consequence, these new results are likely to be highly variable.



The changes to the landscape that take place at year 50 should be dramatic enough to quickly and positively influence the model population in most of the replicate simulations. At the same time, the fluctuations in population size resulting from environmental stochasticity most likely keep the extinction probability from dropping to zero. On average, the population size will probably remain larger than it did when it was confined to the original landscape.