INFORMATION TO USERS

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
A spatial dynamic approach to ecological modeling: Simulating fire spread

Ball, George LeRoy, Ph.D.
The University of Arizona, 1990
A SPATIAL DYNAMIC APPROACH TO ECOLOGICAL MODELING:
SIMULATING FIRE SPREAD

by

George LeRoy Ball

A Dissertation Submitted to the Faculty of the
SCHOOL OF RENEWABLE NATURAL RESOURCES
In Partial Fulfillment of the Requirements
For the Degree of
DOCTOR OF PHILOSOPHY
In the Graduate College
THE UNIVERSITY OF ARIZONA

1990
As members of the Final Examination Committee, we certify that we have read the dissertation prepared by George LeRoy Ball entitled A SPATIAL DYNAMIC APPROACH TO ECOLOGICAL MODELING: SIMULATING FIRE SPREAD and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

D. Phillip Guertin

Malcolm Zwoinski

H. Randal Gimblett

William B. Heed

Richard Strauss

Date 3/29/90

Date 3/29/90

Date 3/29/90

Date 3/29/90

Date 3/29/90

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copy of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

D. Phillip Guertin 3/29/90

Dissertation Director D. Phillip Guertin Date
STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: [Signature]
ACKNOWLEDGMENTS

I would to take this opportunity to thank some great friends: Bill Heed and Rich Strauss for encouragement as a biologist; Mal Zwolinski for teaching me about fire; Randy Gimblett for introducing me to AI; and Phil Guertin who has shown me how to survive in academia. I want to thank my friends in the Advanced Resource Technology Program for their support, and the College of Agriculture for supplying the equipment. And most of all I want to thank my mother and sister for all their help, support and a place to live while I pursued what must have seemed like an endless quest.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section/Caption</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF ILLUSTRATIONS</td>
<td>7</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>8</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>9</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>11</td>
</tr>
<tr>
<td>SECTION 1</td>
<td></td>
</tr>
<tr>
<td>1.1 The Form of Ecosystems</td>
<td>15</td>
</tr>
<tr>
<td>1.2 An Examination of Current Models</td>
<td>18</td>
</tr>
<tr>
<td>SECTION 2</td>
<td></td>
</tr>
<tr>
<td>2.1 Representing the Physical Environment</td>
<td>22</td>
</tr>
<tr>
<td>2.1.1 Data Capture</td>
<td>23</td>
</tr>
<tr>
<td>2.1.2 Georeferenced Data</td>
<td>24</td>
</tr>
<tr>
<td>2.1.3 Cell Resolution</td>
<td>25</td>
</tr>
<tr>
<td>2.2 Methods of Data Analysis</td>
<td>25</td>
</tr>
<tr>
<td>2.2.1 Mathematical Surfaces</td>
<td>26</td>
</tr>
<tr>
<td>2.2.2 Analytical Operators</td>
<td>27</td>
</tr>
<tr>
<td>2.3 Models Using GIS Programs</td>
<td>29</td>
</tr>
<tr>
<td>2.4 Criteria for Designing a Simulation Program</td>
<td>30</td>
</tr>
<tr>
<td>2.4.1 First Criterion: Real Numbers</td>
<td>30</td>
</tr>
<tr>
<td>2.4.2 Second Criterion: Iterative Processing</td>
<td>32</td>
</tr>
<tr>
<td>2.4.3 Third Criterion: Flexibility in Data Retrieval</td>
<td>33</td>
</tr>
<tr>
<td>2.4.4 Fourth Criterion: Neighborhood Analysis</td>
<td>33</td>
</tr>
<tr>
<td>SECTION 3</td>
<td></td>
</tr>
<tr>
<td>3.1 Vector and Raster Systems</td>
<td>35</td>
</tr>
<tr>
<td>3.1.1 Vector Suitability</td>
<td>35</td>
</tr>
<tr>
<td>3.1.2 Raster Suitability</td>
<td>36</td>
</tr>
<tr>
<td>3.2 The PROMAP Basic Configuration</td>
<td>38</td>
</tr>
<tr>
<td>3.2.1 File Structure</td>
<td>39</td>
</tr>
<tr>
<td>3.2.2 Memory Structure</td>
<td>40</td>
</tr>
<tr>
<td>3.2.3 Operators</td>
<td>40</td>
</tr>
<tr>
<td>3.2.4 Mathematical Procedures</td>
<td>46</td>
</tr>
<tr>
<td>3.2.5 Programmable Operation</td>
<td>47</td>
</tr>
<tr>
<td>3.2.5.1 Example of the CELL operator</td>
<td>48</td>
</tr>
<tr>
<td>SECTION 4</td>
<td></td>
</tr>
<tr>
<td>4.1 Modeling Biophysical System Processes</td>
<td>51</td>
</tr>
<tr>
<td>4.1.1 Representing the Environment</td>
<td>51</td>
</tr>
<tr>
<td>4.1.2 Q-Morphisms in Ecosystem Models</td>
<td>52</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS CONTINUED

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2</td>
<td>Developing Transition Functions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.2.1 Spatial Transition Functions</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>4.2.2 Timing</td>
<td>55</td>
</tr>
<tr>
<td>5.1</td>
<td>Fire as a Spatial Dynamic Modeling Problem</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>5.1.1 Factors Which Define Fire</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.1.1.1 Fuels</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>5.1.1.2 Fuel Moisture</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>5.1.1.3 Slope and Wind</td>
<td>58</td>
</tr>
<tr>
<td></td>
<td>5.1.2 Dynamic Aspects of Fire</td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>The Rate of Spread Equation</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>5.2.1 Basic Problems with the Fire Equations</td>
<td>61</td>
</tr>
<tr>
<td>5.3</td>
<td>FIREMAP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.3.1 Problems in FIREMAP</td>
<td>63</td>
</tr>
<tr>
<td>5.4</td>
<td>Other Fire Spread Models</td>
<td>64</td>
</tr>
<tr>
<td>6.1</td>
<td>FIREMAP II</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>6.1.1 Algorithm for Fire Spread</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>6.1.1.1 Flanking ROS</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td>6.1.2 The Fire Spread Operators</td>
<td>73</td>
</tr>
<tr>
<td>6.2</td>
<td>Testing Zero State Predictions</td>
<td>76</td>
</tr>
<tr>
<td>6.3</td>
<td>Testing Non-uniform Conditions</td>
<td>80</td>
</tr>
<tr>
<td>7.1</td>
<td>Identified Errors</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>7.1.1 The Adjusted ROS</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>7.1.2 Wind Fields</td>
<td>85</td>
</tr>
<tr>
<td>7.2</td>
<td>Simulation Benchmarks</td>
<td>86</td>
</tr>
<tr>
<td>8.1</td>
<td>An Evaluation of PROMAP/FIREMAP</td>
<td>87</td>
</tr>
<tr>
<td>8.2</td>
<td>Recommendations for Improvement of FIREMAP</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>8.2.1 Wind Fields</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>8.2.2 Rules for Extinguishing Fires</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>8.2.3 Fire Breaks</td>
<td>90</td>
</tr>
<tr>
<td>8.3</td>
<td>Future Implications</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>8.3.1 Fire Ecology</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>8.3.2 Animal Habitat</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>8.3.3 Risk Management</td>
<td>93</td>
</tr>
<tr>
<td>8.4</td>
<td>Conclusion</td>
<td>94</td>
</tr>
<tr>
<td>LITERATURE CITED</td>
<td></td>
<td>96</td>
</tr>
</tbody>
</table>
### List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. PROMAP Memory Structure</td>
<td>41</td>
</tr>
<tr>
<td>2. CELL Operator Script File</td>
<td>49</td>
</tr>
<tr>
<td>3. Results of Script File in Figure 2</td>
<td>50</td>
</tr>
<tr>
<td>4. The Cellular Neighborhood of Fire</td>
<td>70</td>
</tr>
<tr>
<td>5. Fire Spread Zero Wind Speed</td>
<td>78</td>
</tr>
<tr>
<td>6. Fire Spread 2 mph Wind Speed</td>
<td>78</td>
</tr>
<tr>
<td>7. Fire Spread 4 mph Wind Speed</td>
<td>79</td>
</tr>
<tr>
<td>8. Fire Spread 8 mph Wind Speed</td>
<td>79</td>
</tr>
<tr>
<td>9. Fire Spread 4 mph with Wind Shift</td>
<td>81</td>
</tr>
<tr>
<td>10. Fire Spread 8 mph with Barrier</td>
<td>81</td>
</tr>
<tr>
<td>11. Fire Spread 4 mph with 2 Fuels</td>
<td>82</td>
</tr>
<tr>
<td>12. Fire Spread Actual Terrain</td>
<td>82</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. PROMAP Operators</td>
<td>42</td>
</tr>
<tr>
<td>2. Results of Fire Simulations</td>
<td>77</td>
</tr>
</tbody>
</table>
ABSTRACT

The objective of this dissertation is to develop a new research tool, PROMAP, which will allow the construction of models that satisfy the requirements of spatial distribution and hierarchical interactions within a dynamic framework.

An analysis of the form of ecosystems is followed by an examination of current attempts at ecosystem modeling using spatial relationships. An examination of the analytical procedures used in the spatial modeling process, results in a set of criteria that a suitable modeling system should incorporate. These criteria are: the use of real numbers; iterative processing; flexible data retrieval; and neighborhood analytical procedures.

The basic configuration of PROMAP is discussed with an emphasis on the mathematical procedures and the capability for designing cellular automata within the system. The representation of biophysical systems into a set of spatial transition functions is described in relation to the development of nested hierarchies called Q-morphisms.

Having established the design of PROMAP, a suitable test is devised using the simulation of surface fire spread. A model called FIREMAP is developed and the results are compared to expected fire shapes under Zero State Conditions. These conditions are defined as uniform fuel,
zero slope and zero wind with additional factors held constant. Other simulations of fire spread are made by relaxing the conditions to achieve wind driven fires and the response to potential impediments to fire spread. The response of the simulation shows an accurate correspondence between the simulation and the expected fire shape.

As a final test of the model, all restrictions are removed and a simulation is made under actual conditions of complex terrain, and non-uniform fuels using data collected on the San Carlos Apache Indian Reservation in southeast Arizona.

Deficiencies of PROMAP and FIREMAP are discussed as well as future implications for the FIREMAP model as a management tool.
INTRODUCTION

Eldredge and Cracraft (1980) point to the perception of different patterns in nature. Ecological order in the biotic world gives the perception of hierarchical structure, and distributional aspects show patterns in time and space. To develop a realistic model of a natural system, a model must describe the system hierarchically in time and space.

Models are generally constrained by the inability to measure and use all the variables that would be necessary to describe natural systems. To make the modeling process more tractable assumptions are made about some variables, such as "negligible influence" or "held constant". Which variables are considered important and which are not is decided by the modeler and how well the model simulates the real world. Spatial distribution imposes the additional requirements of locational identity and neighborhood interaction. Although the lack of spatial distribution prevents the examination of hierarchical interaction in space, it does not necessarily prevent examination through time.

The objective of this dissertation is to develop a new research tool, PROMAP, which will allow the construction of models that satisfy the requirements of spatial distribution and hierarchical interactions within a dynamic
framework (i.e., spatial dynamic models). PROMAP is a spatial dynamic simulation language which provides the necessary operators to access and manipulate spatially referenced information to simulate the interactions found in natural ecosystems. As a test of the functionality of PROMAP, it has been applied to the modeling of surface fire spread in a natural system.

The spread of surface fire in natural systems has been the subject of much scientific investigation and there exists a large body of literature describing observed patterns as well as empirical data relating fire behavior, fuels and meteorology. The variability within this system, as in any natural system, precludes the ability to make absolute predictions. The application of empirically derived procedures to actual fire situations has shown the efficacy of equations for rate of spread and other components of fire behavior. The test of this dynamic spatial modeling system would be the ability of the model to produce the expected shape of a fire under a set of specified conditions. The fire shapes generated by the model must be derived from known characteristics of fire behavior. The model has been given the name FIREMAP.

Section 1 will discuss the form of ecosystems and examine some current models and their deficiencies in trying to describe natural systems. Section 2, will
examine how biophysical environments can be described as database structures, and what analytical procedures can be applied. This section describes the primary design criteria for PROMAP. Section 3 examines two competing data structures and the PROMAP program. This section outlines the basic components of the PROMAP system, discusses some of its more powerful operators, and gives examples of how the operators are applied in dynamic spatial modeling. Section 4 looks at the major aspects involved in the decomposition of biophysical processes as part of the model design process. Section 5 describes how fire processes can be translated into a spatial dynamic model, and looks at a prior version of the FIREMAP model. This section also examines several other models which represent a range of approaches to fire spread modeling. Section 6 is a discussion of the FIREMAP model as implemented using PROMAP. In addition, this section will describe several tests that can be applied to the model. Section 7 discusses some deficiencies identified by the FIREMAP model. Some indications as to the performance of the model as it relates to computer architectures are also given. Section 8 evaluates the ability of PROMAP and FIREMAP to meet the objectives of the research. Recommendations are made concerning the improvement of the FIREMAP model, and
several areas of possible future research are also discussed.
SECTION 1

1.1 The Form of Ecosystems

To model an ecosystem or even parts of an ecosystem requires the establishment of some mental picture of what is an ecosystem. One definition is "... a natural unit of living and nonliving components which interact to form a stable system in which a cyclic interchange of materials takes place between living and nonliving units..." (Steen, 1971 page 156). An important part of this definition is the concept of component interactions. To look at one biotic component of an ecosystem, e.g. a species, it would be necessary to describe interactions of the species and its environment. Hutchinson (1957) coined the term ecological niche to describe the set of possible environments in which a species might exist. What then is an environment?

Futyuma (1979) considers an environment as a word that describes the total set of factors that influence the activities, achievements, and ultimately the fate, of an animal or plant. The environment is not immutable but subject to constant change. The interaction of animals and plants is not the only source of change in the environment. Physical processes such as weathering, fire, and earthquakes, also contribute to the variation that is
presented to living organisms, both temporally and spatially.

The mental picture of an ecosystem begins to emerge as a spatially distributed set of components that vary through time. Taking a position at some point above a rural community and looking down reveals a pattern of components, the landscape. Forman and Godron (1986) identified certain aspects which all landscapes possess: structure, function, and change.

Structure can be seen as the spatial patterns made up of ecological objects (e.g. animals, plants) and combinations of objects (elements) that comprise the landscape. The patterns which make up the structure of the landscape is appropriately called the landscape mosaic. The flow of objects (which can also include nutrients and energy) between landscape elements defines function. Function also includes emergent properties derived from the hierarchical nature of ecosystems. Change is simply the alterations of the landscape mosaic through time.

Scale and its effects on the perception of the ecosystem is something that must be considered. From the vantage point above the earth it may be possible to see a small community, some agricultural fields, small woodlots, and a stream. All of these elements comprise the mosaic from the current point of view. Although it may not be
possible to see them from this height it can be assumed that animals are moving between the elements. Descending to a position above one of the woodlots, the perception of the ecosystem changes as the number of elements is reduced. Careful observation shows that the woodlot is comprised of several species of trees, bushes and grasses. This is the effect that scale has on the ability to model an ecosystem. The success of modeling a given event is dependent on the scale. From the original vantage point it would not have been possible to observe the interactions at the edge of the woodlot, nor from the current level would it be possible to describe the interactions of the woodlot and the agricultural fields. The scale at which the environment is perceived must be relevant to the events being modeled. The edge of the woodlot may consist of a gradient of vegetation due to microclimatic effects. In order to resolve the gradient, the point of view must be at the proper distance, or in other words, the physical scale of the model must be consistent with the physical scale of the smallest important detail or process of the actual environment.

For a model to simulate a natural ecosystem four criteria have been identified that the model must possess. First it must be able to represent the spatial attributes found in actual landscapes. Second, the model must be able
to handle the flow of objects within the landscape. Third the model must be dynamic rather than static. Fourth, the model must be adaptive to changing scale. Each criterion links together to form the hierarchical structure that is necessary to complete the model. How well do current models meet these criteria?

1.2 An examination of current models

In his review of landscape models, Baker (1989) categorized models as whole models, distributional models, or spatial models. (1) Whole models are those that look at entire regions and examine the variation of a single variable. The scale of whole models is too large to effectively model any but the most large-grained processes. Baker indicates that whole models have found some utility as submodels in spatial modeling. (2) Distributional models examine the distribution of a variable over a portion of the landscape, and have been used effectively in the form of multivariate differential equations. These models are usually called stand dynamics models since their primary use has been the study of forest composition. JABOWA (Botkin et al. 1972) was the predecessor of these models and FORET (Shugart and West 1977), and SILVA (Kercher and Axelrod 1984) are more recent examples. The stand dynamics models track the fate of individual trees
using information about diameter, growth, leaf area index, and other variables for each species in the model. Although highly detailed, the models are effective only at relatively small scales (0.1 ha) and require estimations be made for transfer coefficients. The major shortcoming of both the whole models and the distributional models is the lack of spatial information. These models are designed to simulate the changes in the distribution of one or more variables without regard to the physical location of the elements involved.

(3) Spatial models contain the most detail and deal with the spatial location and configuration of landscape elements. In these models the landscape is generally divided into a two-dimensional grid of equal-area cells. The univariate case of this model is a single plane of cells and the corresponding multivariate model has multiple planes of cells. Baker identifies two types of spatial models: mosaic, in which change in a mosaic of individual subareas is modeled; and element, in which change in individual landscape elements is modeled. The spatial models have the advantage of depicting the heterogeneity of the landscape but, as Baker points out, the choice of cell size is critical to the resolution of the landscape elements. Using Baker's terminology, a subarea can be
considered an individual cell and a mosaic is made up of a group of cells.

In mosaic models the location, configuration, shape and size of landscape elements are not explicitly modeled, but are derived from the configuration of areas comprised of cells with similar values. Mosaic models deal essentially with the form of the landscape. Element models focus on how individual organisms respond to variation in the local environment. This variation may be due to spatial configuration, character, or density of neighbors. The problem with the implementation of models at this scale is the lack of knowledge of the processes involved. A major factor in the development of spatial dynamic models is the uncertainty of scaling processes from micro to macro and vice versa.

The general assumption in these models is homogeneity within individual cells. Although the models allow the influence of both exogenous and endogenous factors they are generally inadequate in providing mechanisms for influences due to neighborhoods of cells and unpredicted variability.

The inherent nature of spatial models allows their linkage to data bases constructed using geographic information systems (GIS). The next section will examine how physical environments are described in GIS data bases.
and the types of operations that would be required in order to improve on current modeling efforts.
SECTION 2

2.1 Representing the physical environment

Burrough (1986) describes geographic information systems (GIS) as "... computer based tools designed to collect, store, retrieve and change at will, manipulate, and display spatial information from the real world for a particular set of purposes." A GIS describes objects from the real world in terms of:

1. Their position with respect to a known coordinate system;
2. Their attributes, such as elevation and soil type, that are unrelated to position; and
3. Their spatial interrelationships with each other (topological relations), which describe how they are linked together or how it is possible to travel between them.

Data are stored in a GIS data base in one of two forms, vector or raster. Vector systems store information in the form of coordinates that define the boundaries of polygons. The information storage scheme usually makes use of some method of reducing file size, such as run-length encoding (storing the number of consecutive occurrences of a value up to the next different value rather than store each occurrence). In raster systems the information for each cell, which represents a given set of coordinates, is
encoded with the information concerning that point in space. Again, file management may use some encoding scheme to reduce file size. Information in vector systems is retrieved by reference to the polygon, and in a raster system by coordinate pairs.

In general, when a data base is developed it is in the form of vectors since the information is stored at the original scale. Although used by some systems, vector data has a serious drawback for spatial modeling: because the data are stored as a series of polygons, it is extremely difficult to derive the value of a neighboring position. Section 3 will discuss other disadvantages to vector systems for dynamic spatial models. For modeling, the vector data is usually transformed into raster or grid-cell form.

2.1.1 Data capture

The data base is derived from cartographic information and entered into the computer by digitizing the maps that represent various attributes. After the information is entered it is rasterized to the desired scale. The use of vectors for data capture provides the ability to use the data for other models by creating a new raster data base at the appropriate scale. This allows multiple use of the data. Once the data are rasterized the information for a particular attribute (e.g. vegetation) is contained on a
single plane or layer, and is stored as a two dimensional matrix. This process continues until all primary information about the system is entered. The completed data base represents the system as a set of individual layers of information in the manner of Baker's (1989) multivariate spatial model.

2.1.2 Georeferenced data

To be useful for spatial models, the individual layers of information must be referenced to a single coordinate system and that coordinate system must be georeferenced to a known location on the earth. For large areas (e.g. Yellowstone National Park) an acceptable method is the use of Universal Transverse Mercator (UTM). In smaller areas the coordinates can be referenced by surveying from a fixed landmark. The use of a georeferenced data base provides two important properties. First, the location of any cell in the matrix is known through the reference to the given coordinate system. This allows knowledge not only of the information concerning the current location within the matrix, but also the information of any other location either adjacent or remote within the data base. Second, the number of cells in the matrix determines the actual real world area encompassed by a single cell and is translated into a cell width measurement. With this information the
actual distance between any locations on the matrix can be determined.

2.1.3 Cell resolution

As mentioned earlier, the amount of abstraction of the physical environment is determined by the grid cell density. Higher cell densities provide increased resolution only up to the point of the resolution of the original information. In practice, the cell density is determined by the model being constructed and data acquisition should be planned accordingly.

2.2 Methods of data analysis

With a data base in place it is necessary to assess how the information can be utilized within the spatial models. The simplest use of the data base is the query. It might be necessary to find all the locations that fulfill certain criteria as in a suitability analysis. To locate all the possible areas in which mountain lions might be found, it would be necessary to decide which attributes would best predict the occurrence of those animals. The intersections where all those attributes coincide within the data base would provide the answer and represents the use of the data base for static modeling.
Another operation would be the redefinition of the environment based on process-related changes using discrete time intervals. In this case, events such as fire or erosion would alter the values within the data base. This would be observed in a dynamic model. The question raised, however, is how attributes are identified or modified within this type of model. Changing values is more than simply altering a number in a matrix. The matrix value contains information about a single geographic location, but also defines a relationship between itself and its neighbors.

2.2.1 Mathematical surfaces

It is helpful to envision the various matrices as mathematical surfaces. The surfaces can be either two-dimensional (binary or discrete data) or three-dimensional (continuous or semi-continuous data). The information contained in the two dimensional surface is in the form of boundaries on the surface. An example of this would be the layer representing vegetation. Areas within the matrix having a specific value may indicate the location of pine trees while other areas represent other vegetation types. The information content is then the location of various polygons and their spatial arrangement.

Three-dimensional surfaces result from information that has a vertical component. Three dimensional surfaces may
represent a natural phenomenon (e.g. elevation or canopy height) or an artificial phenomenon (e.g. travel time).

Operations on the data base essentially change the surface, either by affecting boundaries or by distorting the surface. How these effects are accomplished is dependent on the operators employed in the system.

2.2.2 Analytical operators

Burrough (1986) identifies three types of transformation functions used in GIS systems. The three types are point, region, and neighborhood functions and correspond primarily to the location that the action occurs. A point function operates at a single locus, whereas the regional function relates to properties of regions. The action of the neighborhood transformation occurs within a cell but is based on the values in the surrounding cells.

The classification developed here is similar but is linked to the action in relation to the matrix, e.g. whether the whole matrix is treated in the same way, whether parts of the matrix are involved, or whether the action is dependent on local conditions around a single point. The corresponding terms are global, local and neighborhood influence.

A global operator is applied to all cells in the matrix. Its effect is related to the type of operation but
all cells are evaluated in turn. The local type is applied to only specified locations. With few exceptions, most operators used in GIS are of the global type. A class of operators common to both schemes is the neighborhood type, in which the action is a response to the surrounding neighborhood. The action may be to interpolate the neighborhood and place a value in the central cell, or the value in the center may be used to effect the neighborhood. In a framework being designed for dynamic spatial models this type of operator takes on special importance.

One of Baker's (1989) observations about current spatial models was the lack of transition functions to allow interactions between the cells. Burrough (1986) examined the neighborhood functions used in GIS programs but, as will be explained in Section 3, these GIS implementations have limitations for use in ecosystem modeling. The design of the neighborhood operators is an essential factor in being able to use the database effectively. The ability to respond to the environment of the cellular neighborhood allows the interaction of regions within the system to occur. Currently available operators of this type are called fixed operators, those designed to do one specific task. These operators are technically also global since they must examine all cells during a single operation. The best type of neighborhood operator would be one that could
be programmed to handle different types of problems, or actually "learn" to handle changing environmental variables. Such a "cellular automaton" operator has been proposed by Gimblett (1989) and Ball (1990).

A cellular automaton can be thought of as an engine that travels around the landscape. Its movements and actions are determined by what it encounters during its journey, and perhaps by what it has learned in the past. A very complex system can be created using very simple automata (Wolfram, 1984). By using the ability of the automaton to react to its environment, operators can be designed to handle data that contain unknown variability.

2.3 Models using GIS programs

GIS programs were not designed for dynamic modeling, but specialized models have been developed using GIS databases. Recent articles (Wilke and Finn, 1988; Sklar, et al. 1985) have shown that this approach can provide insight into ecological processes. Although the general concepts used to produce the models are not new, the computer software that can provide the basic tools for this kind of modeling has not been available. This generally implies the writing of software specific for the process under investigation. The availability of a system which could provide the dynamic modeling aspect while still maintaining
the spatial database would provide a very useful tool for investigating ecosystem processes.

2.4 Criteria for Designing a Simulation Program

As was shown by Wilke and Finn (1988), Costanza and Sklar (1985), and Gardner, et al. (1987), the use of matrix-based analysis is an ideal method for the type of spatial problem associated with ecological models. What is needed is a tool that gives the researcher the flexibility of a GIS but with the power of custom designed software. There are certain identifiable criteria that would be desirable in this system.

2.4.1 First Criterion: Real Numbers

The use of real numbers rather than integers is of primary importance. Mathematical operations as used in simulations require that results be maintained as real numbers in order to minimize error. A major limitation of past system designs was computer memory. In the late 1960's a computer with 4 megabytes of memory was considered by many to be a mainframe machine. Today many personal computers have that much available memory. When memory was a major factor, programs which tried to handle data as matrices were generally designed as integer based, representing attributes as whole numbers (on 16-bit machines this means -32768 to
32767). Not only could more information be processed in memory, but there was also an additional speed increase in computation by limiting disk access, and integer arithmetic is generally much faster than floating point.

Application of GIS programs to the question of land suitability, which was the primary reason for their initial development, provided a level of analysis that was unachievable by the older method of physical map overlays. As the power of the GIS programs became apparent to many researchers and academics, this led to more sophisticated applications being attempted. As a result, researchers and academics have slowly become aware of some of the underlying problems of integer-based systems.

Berry (1987) discussed the idea of "map algebra" which treats each of the maps (a matrix layer) as a variable, and operations are then performed by using the maps in mathematical equations. What has been overlooked is that operations that do not yield an integer value (e.g. division with odd numbers) result in a roundoff error. When this is coupled to an iterative process the answer will be much less accurate than the least accurate input map (Vitek, et al., 1984). All cartographic processes have inherent error in the initial data entry procedures and any operation applied to these data must therefore minimize any introduced error. The idea of a map algebra has merit but it needs a better
type of implementation. Although some programs try to circumvent the precision problem by multiplying by a constant (e.g. 100), this is only practical if the resulting value is less than the integer limit. On a 16 bit machine this means the number can be no larger than 327 and no less than -327. In addition, the original value must be a real number or little is gained by right shifting the decimal point. The answer also would need to be stored as a real number otherwise roundoff error (to 2 decimal places for constant of 100) is created and the benefits of the operation are lost.

As mentioned above, the major problem with the use of real numbers is storage in memory. Modern workstations now have a usual minimum of 4 megabytes of memory storage with most machines accessing 8 to 16 megabytes. This allows the use of real numbers for matrices of at least 512 x 512, which is the standard size of most remote sensing images.

2.4.2 Second Criterion: Iterative Processing

There is no ecosystem that can be represented as static. To model a process requires the dynamic component to be retained, and therefore the system must allow for repetitive operations. The design of the operator must therefore take into consideration that the process will be modeled in discrete time steps. To be useful in iterative
procedures the operator must be able to maintain the interim values. For example, if a parameter changes between steps (e.g. wind shift in a fire simulation) the corrections in the simulation must be done without losing the previous calculations. With properly designed operators the simulation would be as nearly continuous as possible while using cellular transitions and discrete time.

2.4.3 Third Criterion: Flexibility in Data Retrieval

In many cases GIS data bases exist concerning the ecosystems in question. If these could be accessed directly, a better simulation could be created with less work spent on data collection. Additionally the system should allow for the creation of synthetic databases with a minimum of effort. It is desirable when working out the transition rules for a system to be able to predict the outcome using a known set of variables.

2.4.4 Fourth Criterion: Neighborhood Analysis

The discussion in Section 2.2.2 pointed to the importance of neighborhoods in modeling ecosystem dynamics. A major criterion for developing an effective research tool must be the capability of performing a wide range of neighborhood operations. The interaction of neighboring points in the landscape is embodied in Forman and Godron's
(1986) idea of function (Section 1.1). Clearly, without the ability to develop transition functions which allow the flow of objects within and between the systems that make up the landscape, the proposed elements would not improve on current methods.

Section 3, will discuss the PROMAP system which embodies the criteria outlined above.
3.1 Vector and Raster Systems

3.1.1 Vector Suitability

In Section 2.1 it was stated that the data are initially stored in vector format. Vector systems store information in the form of points that describe arcs of polygons. Multiple attributes can be attached to the polygons following the data input, allowing the user to have whatever nominative information is necessary for the particular project.

A primary benefit of vector systems is the ability to store the original information at the original scale. Use of the information to produce as output at the original scale is then possible without loss of detail.

Although analysis other than queries can be done with a vector system, the major drawback is the lack of neighborhood associations. As stated in section 2.4.4, the ability to perform neighborhood analyses is a critical criterion for spatial dynamic modeling. Although it might be possible to design a dynamic modeling system using vectors, the algorithms would be very complex and difficult to implement. The utility of such a design would be questionable except for perhaps some highly specific task. As a general research tool it would not be practical.
For the purposes of spatial dynamic modeling, vector systems will be used for initial data capture and hardcopy display.

3.1.2 Raster Suitability

As indicated Section 2.3, the use of raster or cell based systems has been shown to a useful tool in modeling spatial problems. The cell based system has the major benefit of supporting the desired neighborhood functions and the algorithms are easier to write and implement. There are several factors that must be addressed however, concerning cell based design.

There are two primary shapes used in cellular approaches to spatial modeling: square and hexagonal. The hexagonal structure has the main benefit of providing six directions in which to move. The distance of each move along one of the six cardinal directions is identical, which at first glance seems to be easier to handle computationally. The hexagonal structure was used by Frandsen and Andrews (1979) to examine fire behavior in non-uniform fuels.

A major computational problem with hexagonal cells is transition that is not cardinal. Moving through any of the vertices to the center of the next cell results in
traversing the boundary of two adjacent cells. This poses a problem since the boundary along this path is undefined.

Orientation of the cellular array also poses the directional problem of being able to address North and South or East and West. In this type of structure, only two of the four cardinal map directions are aligned for a given orientation. To move toward the non-aligned compass direction requires traversing the intercell boundary, which, being undefined, will produce an error in distance as a function of travel time. The directional transition problem makes reasonable neighborhood algorithms either extremely difficult or applicable in only trivial, non-general cases.

Another major hurdle for hexagonal cells is the difficulty in producing complete coverage of a standard mapping. The boundaries of the hexagonal array do not conform to the rectangular format of cartographic maps. In the hexagonal structure the coordinates for each row of cells will be offset from the row above or below, requiring a more complicated bookkeeping system.

The alternative cell shape is the square, which is the standard of most spatial systems. The square provides excellent overlay of standard cartographic information and the cells are easily georeferenced. World coordinates can be easily calculated from the matrix location.
The square cell provides the directional transition from cell to cell without the boundary traversal problem. There are eight cardinal directions that can be used with a square cell arrangement, but it must be noted that the four diagonal directions result in a distance equal to the square root of two unit dimensions. The design of algorithms using a square cell system is simpler and provides no barriers to the development of the neighborhood transition rules. The choice for data representation in the PROMAP system, based on the above discussion, is the use of a square cell matrix. How this is implemented as both a file structure and in memory, is discussed in the next section.

3.2 The PROMAP Basic Configuration

PROMAP is designed to run on UNIX workstations. It was developed in part on a SUN 386i workstation in the Advanced Resources Technology Program laboratory at the University of Arizona. It is written in the C language and borrows its primary design philosophy from the MAP-PC program (Tomlin, 1986). The name PROMAP connotes the ability for externally programming the system through its operators. Its modular design allows for the addition of other operators if needed and for modifications if required for special applications or refinements.
3.2.1 File structure

The primary file structure of PROMAP is a two dimensional matrix stored on disk as an ASCII text file using the UNIX compress function (or pack on System V UNIX). The choice of this file structure is based on the premise that most GIS programs have the capability of reading and writing text files and that the development of synthetic databases can be easily accomplished. In addition this data structure provides several advantages to the spatial information being accessed by the program.

By using the two dimensional matrix, objects within the system can be described in terms of their position with respect to a known coordinate system, as well as describing attributes that are generally unrelated to position (e.g. elevation, soil type) in relation to spatial distribution. Due to the topological relationship of the cells within the matrix, spatial interrelationships can be examined to identify how cells are linked together and how travel between cells might proceed.

The system can handle real numbers as well as integers so files are identified as binary, discrete or continuous depending on the type of operator used to produce the file. Information concerning the size of the matrix, the cell width (actual real world distance), and number and names of the data files are kept in a separate header file.
3.2.2 Memory Structure

The program handles data in memory in what can be visualized as four matrix levels (Figure 1). Two of the levels are real (floating point) and two are integer. The floating point arrays are of type Float in the C language and on the SUN are 32 bit values, while the integers are stored as 16 bit values. When data are read into an integer level they are rounded off since the original file could contain real numbers.

The reason behind both real and integer value arrays derives both from bookkeeping problems (due to algorithm considerations) and the need to handle operations that are best done in one type or the other. For example, mathematical calculations are considered to be based on real numbers, whereas Boolean operations are integer. The modular design of the program allows the memory structure to be augmented (number or type of layers) based on the needs of the researcher. The only limitation would be the amount of memory available on the machine and the matrix size of the database.

3.2.3 Operators

Some of the primary fixed operators (operators that perform a set function) will be familiar to anyone who has used Tomlin's MAP program (Table 1). Some major changes
Figure 1. PROMAP Memory Structure. Each layer represents a two dimensional matrix in which the data are accessible by the row and column coordinates. All four layers are accessible at the same time.
<table>
<thead>
<tr>
<th>Operator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATH</td>
<td>Allows maps to be treated as variables in mathematical equations. Standard arithmetic operations as well as exponentiation, trigonometric functions, powers, square roots, and logs are available. Equations can be defined in external files. Global operator.</td>
</tr>
<tr>
<td>DRAIN</td>
<td>Calculates the relationship between neighboring cells to determine the downhill connections over a terrain surface. Each cell in the output is given the accumulated value of all cells which drain into it from above. Global operator.</td>
</tr>
<tr>
<td>POINT</td>
<td>Allows a specified value to be placed in specified location or locations. The locations can be named individually or in a contiguous block. Changes can be made to cells in an existing matrix without affecting other cells. Local operator.</td>
</tr>
<tr>
<td>RECODE</td>
<td>Allows specified values in an existing matrix to be given a new value. Global operator.</td>
</tr>
<tr>
<td>SLICE</td>
<td>Creates an output file by dividing a specified range of values on an input file into equal intervals, then replacing each input value with a new value indicating the ordinal position of its interval. Global operator.</td>
</tr>
<tr>
<td>SPREAD</td>
<td>Creates a file by measuring the shortest distance to each point from any of a selected set of points. Distances are measured between point centers and represents the accumulated distance from the nearest starting point. Global operator.</td>
</tr>
<tr>
<td>SCAN</td>
<td>The output of this operator is a statistic which represents a neighborhood of some specified size around the current cell. The neighborhood can be either square or circular. Examples of the statistics are: average, maximum, median, distance. Local operator.</td>
</tr>
</tbody>
</table>
Table 1 continued

<table>
<thead>
<tr>
<th>Operator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>MINMAX</td>
<td>The output file is created by comparing the values of the input files on a point-by-point basis, and assigning to each point the largest or smallest of its input values. Input files are taken in the order presented and the result of the previous operation is used as the comparison file for the next file in the sequence. Global operator.</td>
</tr>
<tr>
<td>BOOL</td>
<td>The operator creates an output file based on a boolean comparison (AND, OR, XOR, NOT) of two or more files. The boolean operation returns zero for false and one for true. Input files are taken in the order presented and the result of the previous operation is used as the comparison file for the next file in the sequence. Global operator.</td>
</tr>
<tr>
<td>CROSS</td>
<td>The output file is created by combining the categories of one file with the categories of another file on a point-by-point basis, then assigning a user-specified value to all points of each combination. Combinations for which no output value is specified are automatically assigned a value of zero. The user specified values are supplied as an external file. Global operator.</td>
</tr>
<tr>
<td>TERRAIN</td>
<td>This operator creates two files based on the input of an elevation file. The files created are SLOPE and ASPECT and the values in the files correspond to the name of the file. An optional file can be created which depicts ridge and drainage areas. Global operator.</td>
</tr>
<tr>
<td>CLONE</td>
<td>This operator is based on SPREAD but takes the value of its starting cell and propagates that value over the surface. Global operator.</td>
</tr>
<tr>
<td>RADIATE</td>
<td>The output file is a binary representation of the viewsheds of points on a 3D surface. Global operator.</td>
</tr>
</tbody>
</table>
### Table 1 continued

<table>
<thead>
<tr>
<th>Operator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CELL</strong></td>
<td>This operator allow the user to create a new function by supplying an external file which describes how the operator will function. This operator is based on cellular automaton principles and can be programmed to respond to local conditions. Local operator, programmable.</td>
</tr>
<tr>
<td><strong>BEHAVE</strong></td>
<td>This operator implements the fire behavior calculations from the DIRECT module of the USFS BEHAVE program. The input and output have been changed to function with the PROMAP system. Global operator.</td>
</tr>
<tr>
<td><strong>FIRE</strong></td>
<td>The operator creates and output file in which the data represent the predicted spread pattern of a fire from one or more ignition points over some specified period of time. Local operator.</td>
</tr>
</tbody>
</table>
have been made based on the premise that this is a simulation system and not a GIS. Although PROMAP can perform all the functions of a GIS it has certain characteristics well suited to ecosystem modeling.

The operators can be one of two types, global or local. A global operator is one in which the operation specified is applied to all cells in the matrix. A local operator is one in which the operation can change from cell to cell. With the exception of a special operator in PROMAP, all operators are global in the same manner as their GIS predecessors.

A second attribute of the operators is that some are designed to make use of information about the surrounding neighborhood. In essence it is desirable in modeling to be able to do three types of operations which involve the neighborhood:

1. accumulate information in a single cell about the neighborhood
2. disseminate information from a single cell to its neighbors
3. inquire as to the status of the neighborhood

Types 1 and 2 always result in some change in status to one or more cells, while type 3 might or might not result in a change.
3.2.4 Mathematical Procedures

PROMAP incorporates two very powerful features of dynamic spatial modeling. The first is a complete mathematical function called MATH. It provides the capability of doing "map algebra" (Berry, 1987) in a strict mathematical implementation by considering the names of the various data layers as variables, and writing them in the form of an equation. For example, to add two maps together (on a cell by cell basis for the entire matrix) and then divide by a third, the syntax of this equation would be:

\[
\text{output} = \left( \frac{\text{input1} + \text{input2}}{\text{input3}} \right)
\]

Since the operator understands the rules of precedence (multiplication before division before addition before subtraction) the parentheses are necessary to keep from performing the division before the addition. The operations are carried out on each cell in turn for the entire matrix, thereby adding the values in the cells (1,1) together and then the values of cells (1,2) together, and so on. An option available with the MATH function is the capability to define external files (macros) to be used as input. These files may contain equations that are used often (e.g. the universal soil-loss equation) or that may be too cumbersome to type on the command line. For equations that are used many times in a program with only the input variables changing, an option allows the user to specify the equation
and then supply only the inputs needed for this step. Most mathematical functions are supported by PROMAP, including trigonometric functions, square root, natural logs, and exponentiation.

3.2.5 Programmable operation

The second and most powerful tool is the CELL operator, which is based on the idea of cellular automata. This operator is programmable by the user through an external script file (ASCII text). It allows the modeler to specify the location of the action (e.g., a specific cell or cells as well as the entire matrix), the influence of the surrounding neighborhood, and the number of iterations desired. The CELL script file is read into the main PROMAP program and becomes a second program running inside PROMAP while the operator is active. Further work is being done on the CELL operator to provide some type of memory structure. With a form of memory the operator will be able to make decisions on the next action based on past experience (adaptive learning). This would give the operator the ability to change its initial instruction set based on what it finds in the environment.
3.2.5.1 Example of the CELL operator

The CELL operator provides the basic implementation of cellular automata into the spatially referenced database of a GIS system. As an example of how this operator is programmed, an example of a CELL script file is shown in Figure 2. The script represents a cellular automaton which uses Conway's Game of Life rules to determine whether a cell is filled (live) or blank (dead) (M. Gardner, 1970). Based on the number of neighbors, a cell will either become live, die, or experience no change in the next generation. In Figure 3 the script file is applied to a particular starting configuration known as a "glider". This type of object will move across the matrix over a set of generations. In this particular example the initial state of the system (Figure 3a) has two gliders facing each other. Over the period of six generations the two gliders combine and form a "pond".
Zero matrices and load file

Begin global operation definition

Begin neighborhood loop
Neighborhood instructions

End of neighborhood loop

Operations based on neighborhood conditions

End of global operation definition

Execute global operation

File output, copy matrix 2 to matrix 1
Repeat as required

Figure 2. CELL Operator Script File. This is a script that implements the Game of Life rules. The annotation explains some of the major structures.
Figure 3. Results of Script File in Figure 2. This figure shows the initial staring configuration and the results of each or the six iterations.
SECTION 4

4.1 Modeling Biophysical System Processes

The design and implementation of PROMAP is only the first step in the development of better ecosystem models. Although many of the restrictions on this type of modeling have been lifted, there still exists the crucial step of how models are constructed.

4.1.1 Representing the Environment

Consider in this design that the environment is a set of "state" variables. An individual matrix layer in the PROMAP system represents a state variable such as vegetation, and the individual cells of the matrix represent the current value of the variable at a specific location in the environment. For the model to accurately represent the ecosystem it must be able to start with the state variable at some initial value at time t, and at time t + 1 have changed the state variable to the appropriate value. The specification of how the states change over time is called the transition function. What is needed is to determine how many state variables and what transition functions are required to model the system.

Holland, et al. (1989) address the problem of levels of approximation in describing systems. They state that "...given the complexity of realistic environments and the
limitations of realistic [modeling] systems, it is unreasonable to expect ...models to be isomorphisms in which each unique state of the world maps onto a unique state in the model" (Holland, et.al., 1989, page 31).

In their book they consider the design of a layered set of transition functions which they call a quasi-homomorphism or simply a q-morphism. The q-morphism is the mapping of many attributes in the real system through a hierarchical set of transition functions in the model. At the current point in development of a model ecosystem, the q-morphism can be considered in the following manner.

4.1.2 Q-Morphisms in Ecosystem Models

Natural systems are hierarchical structures and as such must be treated in a manner which preserves the hierarchical interactions and properties of such systems. Hierarchical structures exhibit emergent properties which Grobstein (1973) says are the "...result of regulated context" (a result of interactive influences). He goes on to say that "...the emergence of new properties in hierarchical systems is closely linked to what we may call the set-superset transition." This transition can be seen in nature by observing how climate affects various areas in the landscape. Higher rainfall coupled with lower temperatures provides a distinctively different floristic composition at
the top of a mountain from a desert area at the base of the mountain. Closer examination reveals further interactions between the plants based on soil conditions, competition for nutrients and light, as well as other factors.

The general tendency when trying to model a system is to describe it mathematically, in other words decompose the system to a set of equations. As Pattee (1973) points out, this causes some problems. He states, "Mathematical descriptions therefore tend to create total decomposition, whereas the essential behavior of real hierarchical systems depends on the partial decomposition of levels. We find then that dynamical systems theory emphasizes holistic, single-level descriptions, avoidance of instabilities, optimization under fixed constraints and artificial isolation of adjacent levels. In contrast to systems theory, hierarchy theory must be formulated to describe at least two levels at a time, it must optimize constraints for a given function, and it must allow interactions between alternative levels." Adopting the q-morphism approach can provide some transition functions which handle the higher level (global) requirements of the system and other transition functions for the lower level (local or neighborhood) requirements.
4.2 Developing Transition Functions

The starting point for designing transition functions is the realization that information is imperfect and the basic data structure assumes homogeneity within a cell. The first step must be to decide what the primary parameters are that influence the system at the level being modeled. Knowing that it is not possible to completely decompose the system, the modeler must choose the information which, when coupled with a suitable transition function, will best describe transition of the state variables. How this is done is not the subject of this thesis, but the later examination of the fire simulation will give the reader a starting point.

Transition functions in spatial dynamic models have two factors that must be considered: the within and between cell actions, and the rates of change of different systems.

4.2.1 Spatial Transition Functions

As mentioned above, the transition functions must allow for both global and local actions. In the implementation of the model, effects of upper level actions will need to be considered at all lower levels. The lower the hierarchical level on which the action is occurring, the more influence neighborhood conditions are going to have, and the more flexible the operator must be to handle the changing
conditions. The development of appropriate neighborhood influence operators must be of prime concern in the model design.

4.2.2 Timing

The complexity of natural systems is compounded by the need to handle more than one system simultaneously. Some timing is synchronous, as in the overall clock time elapsed during the model run. This corresponds, for example, to running the model for 2 model days. Individual events, however, are more likely to be asynchronous. Take for example the difference in the spread of a fire (measured in minutes) versus the change in open field vegetation communities (measured in months or years). If the system being modeled contains more than one event that is running faster than the overall duration of the model (e.g. minutes vs days), then the provisions must be made for handling asynchronous scheduling.
5.1 Fire as a Spatial Dynamic Model

The first step in building a model of fire in a natural ecosystem is outlined in Section 4. How does fire spread through the environment? This dissertation will be concerned only with the spread of surface fire, and will not attempt to examine either ground or crown fire.

Fire is primarily a contact process. As the fire burns it heats the fuels in front of the leading edge. In low intensity fires the distance of preheating may be measured in inches, while in high intensity, wind-driven fires the distance is measured in feet. How long it takes to heat a fuel to the ignition point is a factor of the size of the fuel (measured as cross-sectional diameter) and the amount of moisture contained in the fuel. In modeling this process it is necessary to identify what factors are most important in describing the spread of fire.

5.1.1 Factors Which Define Fire

Fire is a spatial process, defined by the area that is currently burning plus the area already burned. It is very easy to go into a post-fire area and find the boundary which indicates the extent of the fire. Within the area defined by a large fire, there may be zones in which the fire did not intrude. These islands and the overall pattern of the
fire boundary give the characteristic mosaic appearance associated with large fires. The factors that effect the growth and spread of a fire are the factors need to be delineated in the transition functions. Five factors that influence fire are: fuels, moisture, slope, wind speed, and wind direction.

5.1.1.1 Fuels

The single biggest factor in fire is fuel. Without fuel there is no fire, and without the proper type of fuels the fire can not be sustained. The importance of fuels is evident from the literature. Papers such as Brown (1966), Rothermal's seminal paper on fire spread (1972), and Burgan (1987) all point to the importance of understanding fuel composition (available fuels: dead, down, woody), and the associated factors that effect fuels. The size distribution of fuels affects the rate of spread (fine fuels) and intensity (heavy fuels). Total fuel loading, depth, packing density, and mineral content, all influence fire intensity, and the ability of fire to sustain itself. One of the major influencing factors on fuels is moisture.

5.1.1.2 Fuel Moisture

Fuel moisture is a product of the atmospheric moisture, air temperature, wind, days since last precipitation,
diameter of the fuel, and whether the fuel is live or dead. Although moisture can be measured in any fuel, the general approach is to use half inch wooden dowels of known oven dry weight to estimate a fuel moisture value. The higher the moisture content of the fuel, the more heat it takes to bring the fuel to the point of combustion. This influences the fire initially as a function of whether ignition can occur, and later as to whether the fire can be sustained.

5.1.1.3 Slope and Wind

The effects of slope and wind on fire are much the same. A fire will tend to burn uphill and with the direction of the wind. In addition, the higher the wind speed or the greater the slope the greater the fire intensity and the faster the fire will spread.

5.1.2 Dynamic Aspects of Fire

The factors discussed above as well as others are all subject to change over time. As the fire moves across the landscape the fuel composition may change as well as the terrain characteristics. Wind can change in both speed and direction in relatively short time periods. In addition, precipitation, time of day and aspect or exposure can effect the fuel moisture. The matrix structure can handle the changes due to location as the model runs. To compensate
for changes in time the system needs to run in discrete time steps. Between time steps, alterations can be made to variables that have changed.

5.2 The Rate of Spread Equation

Rothermal (1972) succeeded in taking a theoretical basis for fire reaction and coupling it with empirical data from laboratory experiments to develop an equation which would allow a fire behavior analyst to predict certain aspects of fire behavior.

The equation that Rothermal published defines the quasi-steady rate of fire spread as the propagating flux (compensated for wind and slope) divided by the product of the effective bulk density of the fuel and heat of pre-ignition. The basic equation is:

$$R = \frac{(I_p)_o (1 + F_w + F_s)}{r_b E Q_{ig}}$$

Where:

- $R$ = quasi-steady state rate of spread, ft/min
- $(I_p)_o$ = no-wind propagating flux
- $F_w$ = additional propagating flux produced by wind
- $F_s$ = additional propagating flux produced by slope
- $r_b$ = actual bulk density (amount of fuel per unit volume of fuel at the time of ignition), lb/ft$^3$
- $Q_{ig}$ = heat of preignition (the heat required to bring a unit weight of fuel to ignition), B.t.u./lb
E = the effective heating number, defined as the ratio of effective bulk density to actual bulk density, \( r_{bo}/r_b \)

The description of the fire spread process in Section 5.1 can be identified in this equation. The intensity of the fire is found in the numerator of the equation. As can be seen from the variables, slope and wind enter process at this point. This part of the equation will determine how quickly the fire can heat the fuels in front of the leading edge. The denominator contains the information related to the fuel properties. The bulk density will determine how well the fuel can spread from point to point. The heat of pre-ignition determines how much heat is going to be needed to reach the combustion point. Inherent in the heat of pre-ignition is the amount of moisture contained in the fuels. Essentially, by using the five properties outlined in Section 5.1.1 and if necessary the live woody moisture of some fuels, the rate at which the fire will spread through the fuel can be determined.

The intermediate calculations which derive the components of the fire spread equation also supply other information about fire behavior. In addition to rate of spread, the equations can supply two other measures which are important to the management of fire: flame length and fireline intensity. The flame length is important when
deciding how to suppress a fire (hand line or machine) and how well a fire can be contained by natural and manmade breaks. The fireline intensity will provide an indication of possible fire effects, such as overstory mortality. A summary of the basic fire spread equations can be found in Rothermal (1972).

5.2.1 Basic Problems with the Fire Equations

Since Rothermal's equations were based on laboratory conditions it was necessary to verify their usefulness in actual fire situations. Over the years, various improvements have been made to fire behavior prediction. Researchers have worked on spotting (Albini, 1979, 1983), crowning (Albini and Stocks, 1986; Van Wagner, 1977) and most other aspects of fire behavior. In addition the equation must be calibrated for each fire community. The fuel models must be brought into line with actual fire observations for each particular geographic location. This requires that the variables used in the fire spread equation be verified for the particular data base being employed for the model (Rothermal and Rinehart, 1983).

Another problem is that the equations and techniques that are being developed are not designed for dynamic use, they assume homogeneity of all factors, and the output is tabular. The question arises as to whether or not the
equations can be used for spatial dynamic models to give accurate predictions of fire spread.

5.3 FIREMAP

Vasconcelos (1988) was able to demonstrate the feasibility of using a GIS type program to model fire behavior by creating a model called FIREMAP.

She was able to circumvent the limitations of the GIS program she was using by having a second program create the data layers for the rate of spread and other fire effects equations. The ancillary program was a derivation of the U.S. Forest Service BEHAVE fire prediction program (Andrews, 1986). BEHAVE implements the Rothermal equations to produce a set of tables usable by the fire behavior analyst. Vasconcelos modified the DIRECT module of the BEHAVE program to accept matrix input and produce matrix output. In this manner she was able to construct the necessary data layers to allow a test of the fire spread model. Her results were sufficiently encouraging that further work was continued on the project and will be discussed in Section 6. Although Vasconcelos showed that the approach had merit, there were several problems with the implementation of the model that should be reviewed before proceeding.
5.3.1 Problems in FIREMAP

As in any cell-based model the assumption is still made as to uniformity within the cell. FIREMAP, however, provided an example of how a heterogeneous landscape (non-uniform fuels) could be modeled using this approach. The errors in the model are related to the operators in the GIS program.

The fire spread in the FIREMAP model was implemented using the SPREAD command of the MAP-PC program (Tomlin, 1986). Although this is the standard for a dispersion type operator, it was not designed for this type of application. The operator "spreads" either uphill or downhill and not as a consequence of any other factor. To generate the fire spread map, it was necessary to provide a surface over which the operator could proceed, essentially an uphill direction. This restriction meant that the fire was not spreading as a function of cell to cell transition but as a factor of differential rate of spread over an inclined surface. In addition, the changes in moisture, wind and other factors between time steps, results in many cells being "burned" twice. Again this is a problem in the operator and not in the design of the model. Section 6 will discuss how the FIREMAP design was implemented using the PROMAP system.
5.4 Other Fire Spread Models

Other researchers have worked on the problem of simulating fire spread, and it is appropriate at this point to examine some of these models. Kourtz and O'Regan (1971) designed a fire spread model for examining the spread characteristics of smoldering and creeping ground fires. Although not a surface fire, some of the concepts in the model are relevant to the current FIREMAP model.

Kourtz and O'Regan used a square cell array to map the spread of the fire over an area of less than 0.05 acre. The size of the cellular array was restricted to a maximum size of 50 x 50. The spread of the fire was done by calculating the path of least resistance from cell to cell without any dependence on adjacent cells. Although this model provided some useful information about the spread of ground fires, the small area represented by the array does not make it useful for large scale fires. The assumption by Kourtz and O'Regan of non-homogeneous, discontinuous ground fuels however, did have relevance to a later model.

Frandsen and Andrews (1979) were interested in modeling fire behavior in non-uniform fuels. In their model, they chose to use an array of hexagonal cells, each 2 feet across. The cell size was taken in order to provide sufficient resolution of the fuel matrix. The assumption was made that fuel within the cell was uniform. The choice
of hexagonal cells carries with it the problems that were mentioned in Section 3.1.2. The source of the fires used in the model were line fires, and the examination of fire behavior in non-uniform fuels was through the observation of fire spread distribution.

Since the model results were in the form of distribution graphs, no spread patterns were available. A point that should be mentioned, however, is the sensitivity of the model to the cell size. The model assumes that the fire reaches quasi-steady state within the cell, therefore the cell size must be large enough so that the time for the fire to spread through the cell is equal to or greater than the time spent in the cell. This leads to another problem where the size chosen for the simulations restricted the wind speed to less than two miles per hour. If the wind speed were to be increased, the cell size would need to be increased. This is not reasonable for use in simulations involving changing conditions.

An example of a vector approach to fire spread is presented by Eenigenburg (1987). His model compares the rates and directions derived from four or more points as a fire burns across an area. The points are in the form of a grid of electronic sensors. As the fire reaches the sensors the time since ignition is calculated. An algorithm takes the information and transforms it into a plot that indicates
the direction and rate of spread within the fire area. Although an interesting approach, it is only practical in cases where measuring points can be set up prior to the fire arriving at some predetermined position.

The most recent fire simulation model proposed in the literature is from Australia. Green and Gill (1989) have proposed a bushfire simulation which they have called IGNITE. This model uses a raster type grid as in FIREMAP, but is based on fire equations designed by Noble et al. (1980). The program uses input about fuel quantities, vegetation type, topography, and land use. Indirect reference seems to indicate that these files might be obtained from a GIS data base. The model uses an elliptical ignition template to determine what effect the ignited cells have on their neighbors. This template is implemented as a look-up table and specifies the time it takes for a burning cell to ignite its neighbors. The authors state that the model "assumes that fire spread is a simple epidemic process, with the time delay until ignition for any point in the fuel bed being determined by the path of least time from the fire's starting point."

Green and Gill envision this program to be used as a tool for education, training and planning but have not supplied any indication as to the validity of the model under conditions similar to those stated in Section 6.2,
which can provide a measure of accuracy. In addition, the use of the look-up table and the description given by the authors indicates that the fire is assumed to be linked to the source of ignition over the duration of the fire. This is not a reasonable assumption, since only very small fires would still retain some connection with the source. In large fires, especially in very complex terrain, the leading edges are effectively new sources of ignition. This is why the FIREMAP simulation considers the transition from cell to cell as a new event and makes adjustments based on current local conditions.

Although Green and Gill indicate that topography is used in the simulation, all examples shown are based on differences in fire shape as a factor of fuel structure, and no indication is given as to influences due to slope.

Also the authors indicate that the model can handle spotting (randomized ignition), and fire fighting activities such as water trucks, bulldozers and aircraft. They give no indication as to how these parameters are implemented. This poses a problem if the purported use is in management of fire. To say that a fire break drawn on the computer screen is an effective barrier without justification is doing a disservice to the potential user.

The IGNITE program produces interesting graphics, but the authors have not supplied sufficient information to
assume that the model is producing reasonable simulations as
would be encountered in actual fires.

The next section will discuss the implementation of
FIREMAP using the PROMAP system.
SECTION 6

6.1 FIREMAP II

The fire spread problem is an excellent test of PROMAP to handle a dynamic simulation. The literature on the shape of fire under uniform conditions provides a testable hypothesis for the model (Peet, 1967; Van Wagner, 1969; Anderson, 1983). The first step in designing the new model was to examine the deficiencies in the original version.

6.1.1 Algorithm for Fire Spread

As mentioned in section 5, the operator used to simulate the spread of fire across the landscape was not designed for the cell transitions as required for fire.

Placing a simple 3x3 cell window over the data, the question that must be asked is what controls the spread of fire from one cell to another? Figure 4A represents the neighborhood in which the fire is spreading. The letter H is used to indicate that the transition from one cell to another can be considered as a heading fire. This will be discussed in more detail later.

If all the cells have a definite area, consider the fire in the center cell as starting from a point source and spreading outward in a circular manner. This is analogous to the spreading wave in a pond when a pebble is tossed into the center. If all the cells are identical then the
Figure 4. The Cellular Neighborhood of Fire. The letters in each of the cells represent head, flank or back fires, depending on the orientation to the direction of the wind. In A there is no wind and therefore all spread is uniform. In B and C the wind is blowing due north, with a wind shift to the northeast in C. The expected fire shapes are shown next to each cell structure.
transition from the center to the neighboring cells is a function of the time required for the fire to move over the distance. In this case it can be said that the fire has no direction of maximum spread (the spread is uniform in all directions). The shape of the fire will then be a circle as indicated in Figure 4A.

If the fire was directional (due to slope or wind) it would move preferentially in the direction of maximum spread. The leading edge of the fire is then called a heading fire. In Figure 4B the direction of the heading fire is North. As can be seen in the figure, the shape of the fire becomes elongated. The various directional vectors of the fire spread can then be named as "flanking" or "backing" (F or B) depending on the angular degrees from the direction of maximum spread (zero degrees in this example). It is evident that the transition of the fire from cell to cell must be related to the direction of spread. Figure 4C is an example of what might happen if the wind changed direction during the fire. In this case it is blowing due North at the start and then Northeast at some point later. Notice that the values in the cells have changed in accordance with the new direction of spread (Figure 4C1 to Figure 4C2).

Having established a connection between the direction of the spread and the transition between cells, a
generalization can be formulated. Assume that the 3x3 window represents the central cell, and that a similar window represents each of the other cells. Now let each of the windows reflect the conditions of the fuels, moisture, slope, wind and other factors as defined in that geographic location. It is easy to envision that one of the neighboring cells in the examples in Figure 4 could have a direction of maximum spread that is counter to the direction in the center cell. If this were the case, then the fire would not spread as readily into the neighboring cell. This is a simple generalization of the idea behind flanking and backing fires.

At this point the algorithm needed in the fire spread model can be developed using three cell variables: rate of spread (ROS), direction of maximum spread (DMS), and wind speed. The algorithm can be stated as:

1. determine from what direction the fire is entering the cell based on the cell's DMS;

2. with a trigonometric relationship for flanking fires, calculate the adjusted ROS for the cell using the direction found in step 1 and the wind speed for the cell.

6.1.1.1 Flanking ROS Equation

The equation that is used for determining the off-axis rate of spread is taken from the BEHAVE Flanking subroutine
(Andrews, 1986). The adjusted rate of spread equation takes the form:

$$\text{ROS}_{\text{adj}} = \frac{(\text{ROS} \times (1 - \text{el}))}{(1 - (\text{el} \times \cos(\theta)))}$$

Where:

- \( \text{ROS} = \) rate of spread in chains per hour
- \( \text{el} = \sqrt{(\text{rlw}^2 - 1)} / \text{rlw} \)
- \( \text{rlw} = 1 + (\text{ewind} \times 0.25) \)
- \( \text{ewind} = \) effective wind speed in miles per hour
- \( \theta = \) direction of maximum spread in radians

When the wind speed is zero \( \text{ROS}_{\text{adj}} \) is simply set to \( \text{ROS} \). In addition, when the direction of maximum spread is exactly 90 degrees, \( \theta \) is set to 1 to avoid a numerics problem. Once the adjusted \( \text{ROS} \) is known it can be determined how long it would take for the fire to cross the cell based on the cell width.

### 6.1.2 The Fire Spread Operators

To implement the fire spread model it is necessary to develop two operators. One operator will produce the values for \( \text{ROS} \), DMS and wind speed for all the cells. The second operator, using the cellular automaton approach, will examine the local conditions of each neighborhood and
determine which cells will be involved in the fire and how the fire will spread from cell to cell.

The first operator is essentially the DIRECT module of the BEHAVE system designed as a module of PROMAP. The operator is designed on the premise that some conditions may change while the fire is burning. For this reason the operator produces output with specific filenames which are numbered in consecutive order. This allows the user to produce all the necessary input files prior to the actual fire spread simulation. There are two files which are expected by the second operator. One is called HROS and represents the heading rate of spread for each cell.

The second file is called WINDMS and contains an integer which represents the wind speed and the DMS for each cell. The second operator is called FIRE and functions in the following manner. FIRE reads the HROS and WINDMS files appropriate to the current step of the fire simulation. When the operator is first called the user specifies the name of the file which contains the ignition source(s). It is also possible to specify a file which contains information concerning fire impediments such as roads, rivers, and firebreaks. After all the essential information is loaded, the operator begins to scan the matrix using a 3x3 window.
When the operator encounters a "burning" cell in the neighborhood it will then apply the algorithm described in Section 6.1.1 to determine how long the fire will take to move into the cell. The bookkeeping done by the operator accomplishes several things:

1. tracks the time for the cell to be consumed;
2. does not allow more than one cell to be a potential ignition source and knows which cell was the ignition source;
3. if changes are made between steps, adjustments are made to allow for partially consumed cells.

It should be noted that the operator works from the neighbor cell and not from the cell currently burning. Also, since the adjustments between steps allow for partially burned cells, the simulation is as close to continuous as can be expected using discrete time and a cell based array.

The output files contain the values which indicate at what time in the simulation the cell was considered to be consumed by the fire. All cells not consumed at the end of a time step are indicated as time plus one.
6.2 Testing Zero State Predictions

The example shown in Figure 4A indicates the expected shape of a fire under what is called the Zero State Conditions. Zero state is an assumption that the area in which the fire is burning is both homogeneous and uniform as to fuels, and has zero slope and zero wind. As mentioned earlier, the expected shape of the fire would be circular and the radius would be determined by the duration of the fire. For example if the ROS was 10ft/min. and the duration of the fire was 10 minutes, the radius should be 100ft.

Table 2 indicates the expected distance spread and the observed distance in various simulations. The simulation distance is calculated along the principal axis representing North and will differ from the expected value if the ROS is not a whole number. The difference is always within one cell of the true distance. By analogy, all seven of the additional directions are also correctly calculated. The actual simulated fire spread for zero state conditions is shown in Figure 5. Additional relevant conditions for the simulation were:

1. Fuel model 9
2. One hour fuel moisture 2%
3. Live woody moisture 100%
4. Wind direction 0 degrees

The next test is to remove the zero wind restriction and produce a wind-driven fire. The expected shape of a
Table 2: Results of Fire Simulations

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>ROS (ft/min)</th>
<th>Wind (mph)</th>
<th>Expected Distance (ft)</th>
<th>Observed Distance (ft)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.65</td>
<td>0</td>
<td>260</td>
<td>250</td>
</tr>
<tr>
<td>6</td>
<td>2.79</td>
<td>2</td>
<td>1116</td>
<td>1100</td>
</tr>
<tr>
<td>7</td>
<td>7.70</td>
<td>4</td>
<td>3080</td>
<td>3048</td>
</tr>
<tr>
<td>8</td>
<td>23.91</td>
<td>8</td>
<td>6694</td>
<td>6650</td>
</tr>
<tr>
<td>9</td>
<td>7.70</td>
<td>4</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>10</td>
<td>23.91</td>
<td>8</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>11</td>
<td>7.70</td>
<td>4</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>12</td>
<td>n/a</td>
<td>8</td>
<td>n/a</td>
<td>n/a</td>
</tr>
</tbody>
</table>

Burn Duration = 400 minutes (except Figure 8: 280 and Figure 12: 208)

ROS = Rate of Spread in feet/minute
Wind is speed in miles per hour
Wind direction = 0 degrees
(Figure 5 has no wind direction)
(Figure 9 wind direction is 0 degrees and 45 degrees)
Fuel Model = 9
Slope = 0%
1 Hour Fuel Moisture = 2%
Live Woody Moisture = 100%
Expected Distance = ROS x Minutes of Spread
Observed Distance = Value at furthest point on principal axis
Matrix Size = 150 x 153
Cellwidth = 50 feet (Figure 12: 104 feet)
Source Location = 130,75 (except Figures 8 and 10: 140,75)

NOTE: the observed distance is the cell reached by the fire without exceeding the total burn duration.
Figure 5. Fire Spread Zero Wind Speed. Source is at the center of the fire area.

Figure 6. Fire Spread 2 mph Wind Speed. Source is in the lower portion of the fire area.
Figure 7. Fire Spread 4 mph Wind Speed. Source is in the lower portion of the fire area.

Figure 8. Fire Spread 8 mph Wind Speed. Source is in the lower portion of the fire area.
wind-driven fire is shown in Figure 4B. Figures 6, 7, and 8 show the simulation representing wind speeds of 2 mph, 4 mph, and 8 mph respectively. Except for wind speed, all other condition for Figures 6-8 were the same as Figure 5. Also the simulation time for Figure 8 was reduced to 280 minutes to keep the entire image within the boundaries of the matrix. Figure 9 represents the simulation of a wind-driven fire with 4 mph winds. After 400 minutes of simulation time, the wind was shifted from 0 degrees to 45 degrees and the simulation resumed for another 100 minutes.

The analytical models (Anderson 1983, Van Wagner 1969, Peet 1967) do not produce any better approximation of fire shape than the generated shape produced by the simulation. As the complexity of the fuel structure and terrain increases, the simulation should produce better approximations than the analytical models.

6.3 Testing Non-uniform Conditions

The model has been applied to the simulation of fire spread under the zero state conditions as well as under the wind-driven conditions. The zero slope, uniform fuel conditions are not commonly found in nature and therefore the model should be presented with a set of non-uniform conditions. In Figure 10, the fire is presented with a barrier to spread, in this case a lake. The fire is driven
Black : fuel
White : fire area before wind shift
Gray : fire area after wind shift

Figure 9. Fire Spread 4 mph with Wind Shift. Source is in the lower portion of the fire area.

Black : fuel
White : fire area
Gray : lake

Figure 10. Fire Spread 8 mph with Barrier. Source is in the lower portion of the fire area.
Figure 11. Fire Spread 4 mph with 2 fuels. Source is in the lower portion of the fire area.

Black : fuel 1, ROS 7
Gray : fuel 2, ROS 14
White : fire area

Figure 12. Fire Spread Actual Terrain. Source is approximately in the center of the fire area.

Black : Pinyon-Juniper
Dark Gray : Ponderosa Pine
Lt. Gray : Pine-Douglas Fir
White : Fire area
by an 8 mph wind, and over the simulation time of 500 minutes proceeds around the lake and results in two heading fires on the opposite side.

In Figure 11 the fire encounters a second fuel type. The second fuel has a rate of spread exactly twice that of the initial fuel. All other conditions are held constant.

In Figure 12 all restrictions have been removed and the fire is simulated over an actual set of field conditions. In this case the fire is moving across non-uniform terrain and through heterogeneous fuels. The conditions represented in this simulation are as follows:

The data are for a site known as Ivins Canyon, located in the Spotted Mountain area of the White Mountains in east-central Arizona. The topography is semi-mountainous, with altitudes ranging from 5700 to 7000 feet. Annual precipitation is 18.42 inches with an average snow fall of 19.80 inches. The average daily temperature range at during June when the data was collected, is 50 to 80 degrees fahrenheit. The vegetation consists primarily of pine stands (Pinus ponderosa) with variable crown and understory densities, pine-douglas fire stands (Pinus ponderosa - Pseudotsuga menziesii), and pinyon-juniper (Pinus edulis - Juniperus sp.).
SECTION 7

7.1 Identified Errors

The simulations as shown in Figures 5-9 indicate that with appropriately designed operators the PROMAP system can be used to simulate the spread of fire under both uniform and non-uniform conditions. There are two areas within the model that need to be addressed further as a result of experience with the simulations.

7.1.1 The Adjusted ROS

The fire shapes in the simulations show a more angular leading edge than expected. This is a result of the raster data structure which cannot represent the true continuous nature of the actual fire. The characteristic elliptical shape is seen only under the wind-driven conditions as specified in Section 6. The ability to approximate the shape in the cell based system under the uniform conditions can support the contention that the fire spread in the heterogeneous environment is reasonable. There is a problem which arises in the uniform condition, however, that produces a distorted shape. At very low wind speeds (less than 3 mph) such as those examined by Peet (1967), the simulation creates a shape in which the flanking fire is too great at 45 degrees.
The equation that is used for calculating the adjusted ROS is based on a cosine of the off-axis angle. The optimal point of a flanking fire is at 600 but the cellular neighborhood consists of angles at 900 and 450. At the slower wind speeds this results in a bias as seen in Figure 6. As the wind speed increases the bias is lost since the head of the fire moves considerably faster than the flanking fires. Anderson's (1983) examination of the shape of wind-driven fire indicates that at high wind speeds (exceeding 15 mph) the shape is a very eccentric ellipse with very little flanking spread.

Unless the scale of resolution of the data base was such that a large uniform area were present with very low wind speeds, the simulation should provide reasonable results.

7.1.2 Wind Fields

A second area for further study that has been revealed by the simulations is the necessity to develop better wind field approximations. The simulation relies on the direction and speed of the wind in each cell. Movement of an air mass across a complex terrain results in both change of direction and speed at the surface. Most wind measurements are made at 20m above the ground and then interpolated for use at what is called midflame wind speed.
In the FIREMAP simulations the midflame wind speed and direction is estimated in relation to the terrain aspect of the cell. This value is not accurate enough for the precision at which this model is capable and therefore complex terrain simulations should be considered as first order approximations at best.

7.2 Simulation Benchmarks

The FIREMAP simulations have been run on both SUN 386i and SUN Sparcstation 1 workstations with 8 megabytes of memory and no accelerator cards. The size of the data base is 150x153 cells. It takes on average about 20 minutes to calculate the HROS, WINDMS files on the 386i and about 6 minutes on the Sparcstation 1. If other output files such as flame length are requested, there is an increase in calculation time of approximately 10% above the default time for each additional output file.

The actual run time for a 400 minute burn is about 20 minutes of machine time. On the Sparcstation 1 this is reduced to approximately 9 minutes.
SECTION 8

8.1 An Evaluation of PROMAP/FIREMAP

The PROMAP spatial dynamic simulation language has been shown to be a useful tool in the development of spatial dynamic models. There are, however, some features of PROMAP that need to be improved. First is the interface that is presented to the user when running FIREMAP simulations.

Currently the system uses a command line input in which the user types the commands interactively, or the commands are read in from an external file. This type of input requires an expert user if the simulation is to be successful. For this tool to be useful in other than research applications, the interface must be made "user friendly". This would entail constructing an icon driven graphical interface which would present information to the user in a manner that would be both easy to learn and relevant to the users own experiences.

Second is the problem of portability. Although the analytical code can be run on most UNIX workstations, the graphics display output of the simulations is currently specific to the SUN machines. The system should be designed with portable graphics routines and ideally, with a portable windowed interface.

Despite the steep learning curve associated with PROMAP, the FIREMAP model was able to simulate the spread of
surface fire. This version of FIREMAP has made several major advances in fire simulation.

The original FIREMAP model was encumbered by the operator used to generate the spread pattern, and was only able to spread in one direction. In the PROMAP version, the model is now capable of simulating fire spread in all directions. In addition, the algorithm used in the simulation is related to the characteristics of surface fire spread.

The angular nature of the spread patterns under uniform conditions is not as close to the elliptical shape as is seen in the literature. This is primarily due to the nature of the cellular system and the use of the trigonometric function for adjusting rate of spread as noted in Section 7.1.1. Another aspect of this problem arises in the simulation over the non-uniform terrain.

The current version of PROMAP is still restricted to one wind direction. All wind directions in the simulation are simple adjustments of direction as a function of terrain aspect. In this manner the wind direction is changed but not the wind speed. The example using actual terrain and vegetation characteristics is then biased by this error. In addition, the equations used to calculate rate of spread, take into account the vegetation type (fuel model) when the wind speed is greater than zero. The wind speed is
"corrected" before the final ROS is determined. Although the fire simulation moved along the ridge top as would be expected, the overall pattern should not be considered valid at this time. The FIREMAP model does indicate from the uniform conditions tests that with more refinements, the simulation of surface fire spread can be improved.

8.2 Recommendations for Improvement of FIREMAP

The success of the FIREMAP model indicates that with proper design such models have the potential to become useful both as research tools and management tools. There are several areas that will need to be addressed if FIREMAP is going to be accurate enough for field use. These recommendations were developed in cooperation with D. Phillip Guertin (personal communication).

8.2.1 Wind Fields

As mentioned in Section 8.1, it will be necessary to produce better approximations of the influence of topography on wind speed and direction. A promising approach may utilize the KRISSY program developed by Fosberg and Sestak (1986). This program uses a grid cell matrix as used in raster GIS. The information on wind direction, speed, temperature and other wind components are used to produce a
wind field map. The KRISSY program may be able to provide the improved wind characteristics needed for FIREMAP.

8.2.2 Rules for Extinguishing Fires

Another area that will need to be addressed are the rules for extinguishing fires. At the present time, when a cell is marked as burning in FIREMAP, it will burn for the duration of the simulation. Procedures for extinguishing fires (e.g., fire will not spread from that cell) must be developed. This could be done as a simple function of time, in which the fire within a cell is extinguished after a certain time interval. Another method would be to compute the fuel moisture contents through time and extinguish the fire if the moisture of extinction is reached.

8.2.3 Fire Breaks

An effort must be made to examine the effects of fire breaks (e.g., roads, water bodies, cat-lines, rock outcrops, and trails) as barriers to the spread of fire. In Figure 10 the water barrier simply caused the fire to go around the lake. In high intensity, fast moving fires it is possible for the fire to cross the barrier. This can be a matter of flame length, fire line intensity, or spotting. It will be necessary to establish rules for computing the effectiveness
of individual barriers compared to fire intensity. This may be done by converting rules currently used by fire managers.

8.3 Future Implications

The capabilities found in FIREMAP show the potential for the development of a complete fire management tool. Three areas can be used as examples of how ecological models of this type can be used for more advanced research.

8.3.1 Fire Ecology

The capability of FIREMAP to produce a realistic simulation of surface fire spread can be extended to post-fire effects. Since the intermediate calculations of the fire equations provide information concerning fire intensity, the spread map can be altered to depict a map of intensities. Using the map of fire intensity, a model can be generated that would show what percentage of certain types of vegetation would be killed. In addition, the intensities can be used to estimate soil effects. Once this type of information is available, the next step is to consider what changes the vegetation will go through over some period of time (e.g., the next 10 years).

Although there is currently no PROMAP operator that can examine succession, it is something that could be developed. What would need to be incorporated into the
model would be some form of artificial intelligence (AI). This would be needed in order to allow the model to react to the stochastic variability found in a natural ecosystem. The AI component would also provide a means of handling the timing of multiple events, so that interactions between systems would be possible. With this type of structure, a succession model might be developed using the stand dynamics models (Section 1.2) as a starting point. With a succession model, the user would be able to examine the potential change following a fire, or develop prescribed burn policy based on model predictions for certain burn characteristics. As a management tool for modifying vegetation communities, this same model might be used to examine wildlife habitat.

8.3.2 Animal Habitat

A major reason behind the use of prescribed fires is the establishment of animal habitat. By being able to examine the possible effects of fire under many different circumstances, the manager could select the best alternative to achieve the desired result. As a further extension of the model, the inclusion of the AI component would also allow the possible modeling of animal behavior.

Experts in animal behavior would be able to design basic rule systems that could be incorporated in the model. Once these basic rule sets are established the model could
generate additional rules as new situations are encountered. In this manner the animal being modeled could react with the biophysical environment of the model in much the same way as in the real world. The combination of fire ecology and animal behavior would add new dimensions to ecological models. With models of this complexity the use in assessing hazards is an application that can also find great utility.

8.3.3 Risk Management

Even at the level of the current capabilities of FIREMAP, the application of this type of model to risk management is evident. The ability of the fire analyst to anticipate flame lengths, intensity and direction of spread of a fire would provide better utilization of effort and reduce the possible loss of life when managing fires. With the more complex models that incorporate intelligence, the ability to assess various options through the simulation will be possible.

During the Yellowstone fires of 1988 homes and businesses both in and around Yellowstone Park were in danger from the fires. In similar situations where a database is available, it would be possible to examine how effective fire breaks or vegetation structure might be in controlling a fire.
In areas of the urban/rural interface, the use of FIREMAP could provide information on potential property loss of forest areas managed for fuel reduction versus areas that are not managed. This could have a significant effect on insurance rates and on gaining acceptance by the public for prescribed burn policies.

8.4 Conclusion

It has been shown in this dissertation that both key objectives of this research have been achieved. First is the development of a dynamic simulation language that is capable of being used for modeling ecosystem processes. PROMAP is a research program that has sufficient flexibility to handle the complex interactions of dynamic processes both at the global level and at the local or neighborhood level. Designed to be an open system, PROMAP is capable of being expanded as required to perhaps allow even the interaction of animals and the environment through the inclusion of artificial intelligence.

The second objective was the application of PROMAP to the implementation of a model of a very dynamic spatial process. The FIREMAP model was able to simulate the spread of a surface fire, based on fire characteristics, and produce an expected shape under a set of given conditions.
Improvements in PROMAP and in algorithms related to fire and other processes, will provide increasingly better models of ecosystems in the future.
LITERATURE CITED


