DYNAMIC CLASSIFICATION: CONCEPTUAL DEVELOPMENT AND APPLICATIONS IN WILDLIFE MANAGEMENT

by

James Freeman Williamson, Jr.

Dissertation submitted to the Graduate Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements of the degree of
DOCTOR OF PHILOSOPHY
in
Fisheries and Wildlife Sciences

APPROVED:

R. H. Giles, Jr., Chairman

B. S. McGinnes

T. L. Sharik

A. R. Tipton

J. R. Webster

May, 1981

Blacksburg, Virginia

DEDICATION

In their quest for knowledge, many have stepped closer for a better look. This work is gratefully dedicated to those few who have dared to step back to gain a better perspective.

ACKNOWLEDGEMENTS

I wish to thank Drs. Robert H. Giles, Jr. and for serving alternately as committee chairmen. Their guidance and insight were greatly appreciated. Drs. Burd S. McGinnes, Terry S. Sharik, Alan R. Tipton, and Jackson R. Webster served as committee members and provided valuable advice and encouragement.

The staff of Shenandoah National Park, especially , , and , are acknowledged for their assistance in providing necessary information about the Shenandoah National Park study area. Appreciation is due also to personnel of the United States Forest Service, Harrisonburg office, for supplying data on the George Washington National Forest study area. and of the Great Dismal Swamp National Wildlife Refuge are acknowledged for their assistance and courtesy in supplying information about the Dismal Swamp study area.

and are thanked for their assistance, especially in the latter stages of the study. I shall always be grateful for many of my fellow graduate students who have supplied much friendship and abundant intellectual stimulation. My failure to mention them all by name in no way diminishes my respect and esteem for them.

I am also deeply indebted to my wife,   , and my daughter,   , for their patience, confidence, and encouragement.

Financial support for this research was provided by the National Wildlife Federation; United States Forest Service, Southeast Forest Experiment Station; United States Fish and Wildlife Service, Great Dismal Swamp National Wildlife Refuge; and the Department of Fisheries and Wildlife Sciences, Virginia Polytechnic Institute and State University. Cooperating agencies included the National Park Service, Shenandoah National Park; and the United States Forest Service, Harrisonburg office. Each of these agencies is thanked for its assistance in this effort.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter I

INTRODUCTION

Classification, the deliberate aggregation of things based on predetermined similarities, has received wide use in science. Traditionally, classification has been used to reduce an inordinately large collection to a manageable and comprehensible set of groups. Inferences can then be made about this reduced set regarding the structure and function of the whole.

Although classification has received widespread acceptance and use, it suffers from a serious shortcoming. It is a process of generalizing, and once a group is identified, or formed by the aggregation of 2 or more individuals, the "identities" of the constituents of that group are lost in that process. That is, the group becomes the primary object of concern, and the constituents are considered secondary and of no real significance, except as effectively homogeneous components of a group. While the expediency of this process is obvious, it is also obvious that a certain amount of information about the individual members of each group is irretrievably lost.

Typically, the willingness to sacrifice some detail in an effort to make large-scale inferences has been predicated on the assumption that the human mind can accommodate only a finite amount of information over a given time period. Although this assumption is most likely true, it has fostered a notion that only a finite amount of information can be extracted from a set of objects. The falsity of this concept is central to the concept of dynamic classification as postulated in this dissertation.

The premise that an infinitely large amount of knowledge can potentially be extracted from a set of objects is fundamental to dynamic classification. This should not be interpreted as meaning that infinite (all) knowledge can be derived from a single object, or even a set of these objects. Rather, it should be taken to mean that each object contains an infinitely large amount of information about itself. Thus, using a dynamic classification approach, it is possible theoretically to extract whatever information is desired about an entity (or set of entities) from that entity, or set of those entities.

An operational hypothesis of this dissertation was that specific information relative to the objectives of the classification effort is extractable from the infinitely large body of information inherent to objects. The objectives of this study were the following:

1. to articulate a concept of dynamic classification, and to describe a dynamic classification methodology which would permit the extraction and derivation of specific information in a manner unconstrained by the limitations of conventional classification, and
2. to demonstrate the utility and validity of the dynamic classification concept in applications within wildlife management.

Chapter II

LITERATURE REVIEW

FUNDAMENTAL THEORY

Before considering the concept of dynamic classification, the more basic concepts fundamental to it will be analyzed. In the following sections, the ecological concepts of habitat and niche, the basic nature of classification, and the structure and function of the wildlife management system will be examined.

Habitat and Niche

In the broadest sense, habitat may be considered as the living space of an organism (Odum 1963:9). Odum (1971:234) later elaborated on his definition and defined habitat as the physical place where an organism lives, or the physical circumstances under which it is likely to be found. Smith (1974:24) used a similar definition but noted both living and nonliving environmental components within the habitat and substituted the term "population" for organism.

Early researchers found difficulty in describing distributions of species using only habitat parameters as the descriptive criteria. In his discussion of the distribution of lady-beetles, Johnson (1910) noted that distributions of some species were not limited by food supplies only, but by other, unexplained factors in the environment. Although Johnson used the term "niche" in describing distributions, he failed to articulate a formalized concept regarding these "other factors."

In his studies of birds and mammals, Grinnell (1917a, 1917b) proposed the term "niche" to describe the place occupied by a single species in an association. Grinnell (1924:227) later expanded his niche concept and defined niche as the ultimate habitat unit. In a later paper, Grinnell (1928:435) described niche as the "ultimate distributional unit, within which each species is held by its structural and functional limitations."

Grinnell regarded niche as an extension of the habitat concept to explain the distribution of species within a given habitat. A contemporary of Grinnell, Charles Elton, believed that the function, or role, of a species in its habitat would ultimately explain distributions, and that the niche itself was defined by the functional capabilities of the respective species. The following passage from Elton (1927:63-64) summarized succinctly his ideas:

> "It is therefore convenient to have some term to describe the status of an animal in its community, to indicate what it is doing and not merely what it looks like, and the term used is 'niche.' Animals have all means of external factors acting upon them - chemical, physical, and biotic - and the 'niche' of an animal means its place in the

> biotic environment, its relations to food and enemies. ... When an ecologist says 'there goes a badger' he should include in his thoughts some definite idea of the animal's place in the community to which it belongs just as if he had said 'there goes the vicar.'"

Using Elton's concept of the functional niche, Gause (1934:19) expanded upon it by developing the principle of competitive exclusion, which stated that no two species may occupy exactly the same niche. Hardin (1960) restated the principle (although somewhat ambiguously) as "complete competitors cannot coexist." The functional niche concept was articulated further by textbook authors such as Odum (1953:27-30,230-238) and Clarke (1954:468-469).

The ideas of Elton, Grinnell, and Gause were combined in a revolutionary conceptualization of the niche by G. E. Hutchinson. Hutchinson (1958) postulated that niche may be perceived as multidimensional. In his original 1958 paper and in two later works (1965,1967), Hutchinson stated that a species exists within a specified zone of lower and upper limits within a multidimensional set of environmental gradients. With these gradients as axes, each species exists within a theoretically distinct hypervolume. The entire hypervolume of habitable space was called the fundamental niche by Hutchinson (1958,1967:232). However, a species may not be able to use the entire fundamental niche, due to competition from other species inhabiting the same general habitat. Hutchinson (1965:32) defined the realized niche as that portion of the fundamental niche remaining after accounting for the portion lost to competition. Hutchinson's original hypervolume niche concept has been expanded somewhat and placed in more discrete mathematical terms by various authors, including Patten (1962), Vandermeer (1972), and MacArthur (1968).

Theoretically, the concepts of niche and habitat are applicable to any organism. While the concept of niche may be applied equally to plants or animals, it has found limited favor among botanists. Rather, botanists have made extensive use of the term "habitat" to describe everything from broad vegetation types to localized areas containing characteristic plant groupings. In his classic text, Oosting (1956:27) used the term to denote localized areas possessing similar edaphic and environmental conditions, such as a "swamp habitat," or an "upland habitat." Daubenmire (1968:4) removed some ambiguity by equating habitat with ecotope and biotope. He further defined habitat as: "a rather specific kind of living space or environment, i.e., a constellation of interacting physical or biological factors which provide at least minimal conditions for one organism to live or for a group to appear together." Billings (1952) defined the entire array of environmental factors affecting plants as the "environmental complex." Goodall (1963) and Whittaker (1967) used the phrase "habitat hypervolume" to include the set of environmental factors constraining the distribution of plant species. These concepts are not far removed from the fundamental niche concept.

Classification

In general, classification is the process of grouping similar objects and separating dissimilar ones (Nelson et al. 1978). This definition may at first seem trivial; however, the ramificatitons of the concept are profound. In his fundamental analysis of classification, Gilmour (1951) noted that the process of classification has universal applications and that "classification is a prerequisite of all conceptual thought, whatever the subject matter of that thought." Goodall (1966) has argued that classification is essentially a hypothesis-testing process in which the null hypothesis is that individuals or objects under study belong to a single class.

Gilmour (1951) noted that the function of classification is the construction of classes (where the significance of each class is related to a particular purpose) about which inductive generalizations are made. If classes themselves are organized into groups to form different levels, a hierarchical classification results (Good 1965, Cormack 1971). The structure of such systems is consistent with conventional hierarchy theory. Hierarchical classifications are fundamental to biological taxonomy and systematics. In summarizing biological classification, Mayr (1968) noted that classifications are similar to theories in that both possess predictive power.

Cormack (1971) provided an excellent review of conventional classification theory from a statistical point of view. Lance and Williams (1967a,1967b), Wishart (1969), Williams (1971), Goodall (1973), and Pielou (1977:312-331) have discussed classification theory and numerical classification techniques appropriate in ecology, especially plant ecology. Bailey et al. (1978) summarized the purpose of classification by stating that it "creates order out of chaos and reduces to a workably small number the total number of objects ..." with which an analyst or manager must deal.

Vegetation Classification

Much of the early habitat research was performed in the field of plant ecology, with emphasis being placed on the relationships between environmental factors and distributions (presence or absence) of plants. In their efforts to study vegetation and its distribution, botanists devised various classification techniques. Daubenmire (1968:252,254) categorized vegetation classification approaches as being either divisive or agglomerative, and gave appropriate examples. However, the taxonomic problem of "lumping and splitting" had been addressed earlier, as Dansereau (1957:129), citing Villar (1929), noted that environmental classification is based on the criteria of harmony and discrepency.

Daubenmire (1968:249) categorized vegetation classification as being either artificial (i.e., superficial, conceptual) or natural (based on ecological criteria). However, Pfister and Arno (1980) noted that in actuality, all classifications are essentially artificial, being dictated by the specific objectives of the classification process itself. They suggested the use of the term "technical" (cf. Cline 1949) to denote special-purpose, limited objective classification and the term "natural" to signify multiobjective classifications based on natural (ecological) relationships.

Bailey et al. (1978) noted that classifications may be single or multifactor, as well as being single level or hierarchical. In a broader sense, Kessell (1979b) noted that authors of vegetative classifications assume discrete classification groupings, while gradient analysis "deals not with discontinuous classes but rather with continuity and gradient relationships." Kessell (1979b) gave a comprehensive list of the various vegetation classification and gradient analysis approaches, while Bailey et al. (1978), Daubenmire (1968:249-263), and Mueller-Dombois and Ellenberg (1974:153-176) have presented excellent summaries of the major vegetation classifications and the philosophies underlying many of the more notable techniques. Vegetation classification techniques have also been reviewed by Burger (1972), Carmean (1975), Jones (1969), and Rowe (1971).

A goal in vegetation classification is to conceptualize vegetation distribution in an ecologically meaningful framework. The purpose of classification in ecology is not necessarily the classification itself, but to gain further insight into the holistic nature of the environment. A primary utility of classification lies in its ability to be used as a tool for inference (Gilmour 1951). Layser (1974) has noted that following the Braun-Blanquet school of phytosociology (i.e., that habitat and floral community are inseparable and habitat conditions are mirrored by the flora present), certain plant species are diagnostic of the environmental conditions present. Thus, ecologically-based classification may be used to infer or deduce which organisms might be found within a habitat possessing a given set of environmental gradients.

Habitat Classification

Because vegetation is generally accepted as an indicator of environmental conditions (cf. Clements and Goldsmith 1924) and since many animals depend directly on the vegetation substrate for a variety of their essential needs, vegetative characteristics can be useful in making inferences regarding the suitability of an environment for wildlife habitat. In much the same way as vegetation may be analyzed, wildlife habitat (i.e., the physical environment in which wildlife species live) may also be analyzed. Vegetation analysis and wildlife habitat analysis are alike in that both are efforts to examine the relationships between environmental factors and the respective flora or fauna present in an area.

However, vegetation classification and wildlife habitat classification are distinctly different in some respects. Unlike vegetative classifications, in which vegetative groups are categorized based on similarities or differences, wildlife habitat classification usually involves the assignment of areas of habitat into groups based on some preconceived criteria of relative value of these areas to various wildlife species. In short, wildlife habitat classification is an effort to discriminate and categorize areas having similar value for each wildlife species.

Although they may vary in complexity and sophistication, the processes of wildlife habitat analysis and habitat evaluation are precursors to habitat classification. Habitat analysis is a process in which pertinent environmental data are gathered for an area. The goal of this analysis is to determine presence or absence and amounts of the essential needs for wildlife species. Admittedly, value judgements may be used by the habitat analyst. However, value judgements are significant only in determining the relevance of different observational variables with respect to determining or predicting the presence or absence of the plant or animal under study. Conversely, habitat evaluation is an effort to interpret whether and how well the environmental conditions in an area meet the needs (both habitat and niche requirements) of a species (deVos and Mosby 1971). In short, a habitat analysis assembles and processes data into a description or characterization of an area. A habitat evaluation uses the habitat analysis, along with a set of criteria of goodness, to produce an evaluation of managerial usefulness.

THE WILDLIFE MANAGEMENT SYSTEM

Management, including wildlife management, is the process of making decisions and taking actions to achieve objectives. Although the term refers to a process, management itself is part of a larger whole. Considered in this context, management is a system.

Management systems are composed of the four major components of all general systems: inputs, outputs, processes, and feedback. In the case of management systems, inputs are information; outputs are decisions and actions; feedback is the current assessment of system performance along with corrective or stabilizing actions; and the process is the making of decisions and the taking of appropriate actions to achieve specified objectives. Feedforward (cf. Giles 1978:12) is included as a component of some general systems, especially human-oriented ones. Feedforward is anticipatory in nature and may alter system performance in order to accommodate future conditions. Each component is integral and essential for smooth operation of the system. A breakdown of any single component can result in reduced system performance, or even in system failure.

In the field of systems science, all systems are considered to be a part of a hierarchy of other systems (Sutherland 1975:99; Hall and Day 1977:6). Thus, every system is a subsystem and every system has at least 1 subsystem. Within the management system there may be many such subsystems. Because input of information is critical to the management system, information subsystems (systems) have been developed. The function of these systems is to provide for the input of information for the management process.

To understand better this function, discrimination between information and data is important. Data are isolated facts that have real meaning only when placed into a context (Rosove 1967:3). Information is data that have been placed in context. All data may be considered potential information. Information, as required by the management system, is composed of data that have been sifted, sorted, and put into a usable, comprehensible format. Specifically, the function of an information system is to generate information for management from a larger set of data.

Wildlife management has been described as a system (Giles 1978:11), and is a typical example of the general management system described previously. As with management of other resources, information systems have been developed within wildlife management. Some of these systems are merely established techniques for deriving information from raw data. Others are more sophisticated and utilize computers to process data and display information. Like many other management systems, wildlife management is growing in complexity and scale. Although management is often performed at small scales using Leopold's (1933) techniques, many modern wildlife managers have been assigned tasks of management at larger scales. Subsequently, not only are new techniques to be developed, but more and better information will be required for macro-scale management. However, in order to understand better the task facing wildlife managers of the future and possible ways of handling the problem, an analysis of the wildlife management system is in order.

Wildlife, considered collectively, is a resource. The purpose of wildlife management, or the wildlife management system, is to exploit that resource in such a way as to attempt to maximize the utility of the resource. Three general approaches are currently used to effect this end. They are population approaches, sociological approaches, and habitat approaches.

Population Approaches

Wildlife is a renewable resource and therefore, the supply (i.e., the amount of physical resource) may be increased (to an extent) without fear of depletion. Thus, the strategy behind population approaches is to manipulate the resource directly. The populations of desired species or species groups are manipulated in order to meet demand. Population densities are often manipulated by employing management tactics that alter either natality, mortality, migration, or all three (Giles 1978:98,102-114).

Sociological Approaches

Sociological approaches involve managing the resource by managing the resource user. In this approach, the actual amount of the physical resource may neither be increased nor decreased. Rather, the demand and use is changed to a level appropriate to the resource. Public awareness campaigns, law enforcement programs, or other publicity efforts are examples of frequently used techniques involving the sociological approach. This approach may work well, both for conserving scarce resources as well as increasing the utility of existing resources. Sociological approaches are often quite effective in bringing demand to a level commensurate with existing resource levels.

Habitat Approaches

Habitat approaches involve indirect management of wildlife resources. In these approaches, the amount of resource is increased or decreased (usually increased) by manipulation of the "living space" (Odum 1963:9) of the desired species (i.e., the resource component under management). Wildlife resources may be affected significantly by various land uses. For example, practices such as silvicultural treatment may have marked impacts on the amount and quality of habitat for a variety of wildlife species. Assuming that an increase in the amount of suitable habitat or an increase in the suitability of existing habitat will result in a concomitant increase in the population (the resource), the strategy of this management approach is apparent.

THE WILDLIFE MANAGEMENT INFORMATION SYSTEM

A management information system is any functional system that converts data into managerially useful information (Rosove 1967:97). Such systems exist in wildlife management. Although these systems tend to be methodological in nature, they nevertheless exist, and are functional in wildlife management. Wildlife management information systems are useful in population, sociological, and habitat approaches.

Population Information Systems

Various population estimation techniques may be classified as information systems in the functional sense. These techniques range from field procedures such as drive counts for deer (Morse 1943) and strip censuses (Overton 1971) to more analytical techniques of population estimation. The classic Lincoln index (Overton 1971) and its modification (Chapman and Junge 1956), the Schnabel method (Schnabel 1938), and the methods and models proposed by Jolly (1963,1965), Leslie (1945,1948,1952), Leslie and Chitty (1951), and Leslie et al. (1951,1953) serve to convert raw population data into estimates of population size and behavior. Likewise, life tables may provide insight into population dynamics (Eberhardt 1971). More recent population analysis and simulation techniques (Tipton 1975, Coyle 1980, Dean 1972) have been computerized and may provide detailed information on population structure and dynamics.

Sociological Information Systems

Sociological approaches to wildlife management also require information inputs. Various psychometric techniques, such as the Churchman-Ackoff procedure (Churchman and Ackoff 1954), objective weighting (Giles

1978:217-218), and the Delphi technique (Linstone and Turoff 1975) have been used to determine attitudes of both resource users and managers. Beattie (1979) demonstrated the utility of these approaches in deriving basic sociological information from hunters and wildlife managers. Computerized techniques, such as PHAEDRUS (Cason 1980) have been developed to assist decision-makers. Cowles (1979) developed a computer program for determining optimum law enforcement personnel deployments.

Habitat Information Systems

Habitat manipulation has been used traditionally in wildlife management. In his classic book, Leopold (1933) spent considerable effort in describing habitat and habitat manipulation techniques. The literature of wildlife management is replete with techniques for managing wildlife through habitat alteration. However, sound habitat management requires sound habitat information. Fortunately, many formal techniques (systems) exist for obtaining this information. These systems will be categorized as "current status" systems (those that are not capable of supplying predicted information) and "future status" systems (those that can predict habitat conditions).

"Current Status" Systems

Despite their rudimentary nature, habitat analyses were among the first information systems employed for securing basic habitat-related data for wildlife management. Early attempts at wildlife habitat analysis had much in common with techniques used by plant ecologists of the day. Leopold (1933) emphasized the importance of food to wildlife, and many early approaches examined vegetation as a source of food. Aldous (1944) and Wilm et al. (1944) described two early techniques for estimating browse production for deer. A more direct method, the total vegetation clip, was used by Harlow (1955) to determine browse production. Allen and McGinley (1947) and Downs (1944) concentrated on other food sources and described methods for determining mast production.

The above techniques were inherently labor-intensive and site-specific. However, several extensive techniques have also been used. Techniques for producing vegetation cover maps from which wildlife habitat inferences could be made were described by Dalke (1941), Graham (1945), and Arnold (1946). Leedy (1948) described the interpretation of aerial photographs for wildlife management purposes. Other researchers such as Crawford (1946,1950), Byrne and Zeedyk (1965), Schmautz (1970), and Allen et al. (1963) described techniques for delineating wildlife habitat as a function of soil type and soil fertility.

Although some of these techniques are quite sophisticated, the information gleaned from them has little direct management application. While they may be useful in delineating the physical extent of specific habitat parameters such as food, these analytical techniques have been able to produce little more than cursory estimates of overall habitat suitability.

Habitat evaluation techniques can provide information appropriate for intensive wildlife management involving habitat manipulation or requiring estimates of habitat suitability. In habitat evaluation, results of habitat analyses are combined with basic information about the life history (or niche requirements) and habitat requirements of wildlife species to produce qualitative measures of overall habitat suitability for these species. Literally, an assessment is made of the value of the environment to the species.

The concept of habitat evaluation is not new. Leopold (1933:378-380) suggested a technique for determining overall habitat suitability. In his "habitability tally," Leopold based habitat suitability on how well food and cover requirements for game species were met within the specified management area. Other systems have been developed; however, many have been superficial improvements of Leopold's basic technique.

Over the years, various habitat evaluation techniques have been developed. Davenport et al. (1944) developed a technique for estimating carrying capacity for deer based on twig counts. Aldous (1944) claimed his browse survey could be used for rating deer range. Unfortunately, a serious fault of many of the early efforts was the use of a single criterion, such as food availability. Many later techniques have been multifactor approaches, incorporating a variety of parameters into a single suitability index.

Recently, a physiological-anatomical approach to habitat evaluation has been advocated. Kirkpatrick (1975, 1980) proposed use of physiological and nutritional indices as predictors of habitat or environmental suitability. Seal (1977) considered the animal to be an integrator of environmental factors, and advocated use of physiological and anatomical parameters to assess habitat. Pertinent examples are the use of blood parameters, perirenal fat, and femur marrow fat to determine overall physiological condition (Warren 1979) and to infer general habitat condition.

Others have attempted to gather habitat data directly, and to extract from it information for making management decisions. These efforts have ranged from single factor to multifactor approaches. Perkins (1973) and Buckner and Perkins (1974) described a multifactor approach for evaluating wildlife on commercial timber lands. This technique and a similar one by Willis (1975) were field oriented and assumed equal importance among evaluation criteria. Other multifactor approaches have used computers to process and handle data. Katibah and Graves (1978) used preprocessed LANDSAT data to evaluate wild turkey habitat. Daniels et al. (1974) utilized a computer mapping package, SYMAP (Robertson 1967) to display habitat for selected wildlife species.

A refinement in the multifactor habitat evaluation has been to weight individual evaluation criteria. Daniel and Lamaire (1974) developed a weighted multifactor habitat evaluation procedure for use in water resource development in Missouri. This methodology was expanded by Flood et al. (1977) and later incorporated into the Habitat Evaluation Procedures (HEP) used by the U. S. Fish and Wildlife Service (Farmer 1978). The habitat evaluation methods used by the U. S. Army Corps of Engineers (U. S. Army Corps of Engineers 1976) are similar to the HEP, but are less sophisticated.

Even though several of these techniques may be used on extensive areas, all are essentially manual. Several authors have enlisted computer assistance. Roller (1978) described a technique for qualitatively evaluating deer habitat using a computerized, multifactor approach. Taft and Davis (1978), Williamson and Koeln (1980), and Antennucci et al. (1979) have used cellular computer data bases and mapping programs to delineate and display wildlife habitat. Fales (1969) discussed the relative merits of several early computer mapping packages for displaying wildlife information. Practical applications of three recent computer mapping systems for land use planning were examined by Deschene (1977).

To recapitulate, wildlife habitat may be perceived as a multidimensional hypervolume, with all of the factors influencing the animal in the course of performing its life functions, constituting the axes. Habitat analyses involve examination of the components, structure, dynamics, and interactions of habitat. Analyses may also include the interpretation, synthesis, and transformation of observational data into information potentially of use in wildlife decision making.

If habitat analyses are performed with a preconceived notion of value, and with the objective of determining which parts of the animal's environment are of importance, they become evaluations. Habitat evaluation includes an ordering of environmental parameters based on perceived importance to the animal, and identifying habitat components that are beneficial or detrimental to the animal in meeting its life requirements.

An extension of wildlife habitat evaluation is habitat classification. Habitat classification is the process of identifying and ordering areas of similar suitability for a species or species group. Depending on the objectives of the classification, the task may vary from one of identifying areas where total life requirements are met, to one of delineating areas possessing only specific niche requirements.

In short, evaluation is an extension of analysis, and habitat classification is a subset of evaluation. The three may be related in a practical sense, but they are nonetheless functionally separate processes for providing different degrees of structure to observational habitat data. The information from habitat analyses is not necessarily management-oriented due to its lack of contextual structure. Habitat evaluation provides this structure. Therefore, habitat evaluation is management-oriented by virtue of the nature of the information it produces. Although habitat classification does not provide the type of detailed information that evaluation does, classification results are structured in such a way as to be appropriate for upper-level management when extensive areas are involved (Coulombe 1978).

Development of comprehensive multiresource assessments (including wildlife classifications) has been accelerated by the passage of recent legislation including the Forest and Rangeland Renewable Planning Act of 1974, the National Forest Management Act of 1976, the National Land and Resource Policy Act of 1976, and the Soil and Water Conservation Act of 1977 (Nelson et al. 1978). In response to these mandates, a wide variety of natural resource classification techniques has emerged, as evidenced by recent symposia dealing with the topic (cf. Marmelstein 1978, Berger 1978, Lund et al. 1978, and Swanson 1979).

The habitat classification techniques recorded in recent symposia are too numerous to discuss properly here. Some have used remote sensing techniques (Katibah and Graves 1978; Best and Sather-Blair 1978; Traylor and Mealor 1978; Treadwell 1978; Pettinger et al. 1978; Merchant and Waddell 1974). Some methodologies are suitable for extensive areas (e.g., Schmautz 1978, Parker 1978). Thomas (1979) and Thomas et al. (1978) related wildlife habitat suitabiity to vegetation succession. Some techniques (cf. Perkins 1973) have been developed for specific purposes, while others (cf. Buttery 1978, Witmer 1978) are appropriate for obtaining multiresource information at the ecosystem level.

The wildlife habitat classification systems that have appeared to date have been diverse. The functional nature of the techniques varies primarily with the needs of the user and the availability of input data. However, all these techniques are virtually identical in approach. All relate wildlife habitat suitability (expressed in some units of relative goodness) to some environmental parameter or parameter set. These techniques, as sophisticated as they may be, rely on the correlation between directly measured habitat parameters (e.g., food abundance) or calculated parameters, (e.g., diversity and interspersion) and perceived habitat suitability.

"Future Status" Systems

A serious deficiency in many wildlife management information systems (but especially in ones dealing with habitat) is the inability of the system to accommodate the temporally dynamic nature of the resource. This is especially true of habitat information systems. Due to the dynamic nature of vegetation, for example, habitat suitability for particular species may change significantly in a relatively short time. In short, unless environmental changes can be taken into account explicitly, most existing information systems produce dated information, which in time will most certainly become obsolete.

To date, most wildlife information systems have been "current status" systems. "Future status" systems are able to produce scenarios based on expected or predicted changes. Short and Schamberger (1979) and Morris and Nichols (1979) discussed the use of assessment procedures to determine the impact on wildlife habitat caused by land use changes. Mielke et al. (1978) used a computerized forest stand growth model to predict changes in red-cockaded woodpecker habitat resulting from various timber management strategies.

A wide variety of forest models have been developed. Leak (1970) used a simplistic approach and modeled forest composition and succession as a function of the birth (regeneration) and death rates of the tree species present. Horn (1975a,1975b,1976), and Peden et al. (1973) extended this concept and proposed that succession could be modeled using a Markov chain approach. Waggoner and Stevens (1970) have also used Markovian approaches to model changes in stand species composition. A Markov chain approach was used by Bruner and Moser (1973) to predict diameter distributions in an uneven aged forest. The appropriateness and utility of Markovian processes in vegetation modeling have been debated by Slayter (1977), van Hulst (1978,1979), and Culver (1981). Compartment models utilizing differential equations to express species replacement rates have been used by several researchers (Johnson and Sharpe 1976, Bledsoe and Van Dyne 1971, and Shugart et al. 1973). Gutierrez and Fey (1977,1980) and Boyce (1977,1978) used compartment models, but utilized feedback loops to drive successional trends. A compilation of recent forest models may be found in Fries (1974), and Shugart and West (1980) have discussed many of the major forest modeling efforts.

The bulk of the forest stand models in existence have approximated forest dynamics using deterministic models. A few, such as the JABOWA model (Botkin et al. 1972) and its derivatives (Shugart and West 1977) have used some stochastic techniques; however, most of these models have tended to behave deterministically.

Although the specific techniques and approaches used vary considerably, most of the models discussed by Shugart and West (1980) produce similar information. That information is produced by simulating changes in forest composition over time. An entirely different approach to predicting characteristics and composition of on-site vegetation has been taken by Kessell (1976,1979a). Kessell has called the approach gradient modeling.

Gradient modeling is an extension of Whittaker's (1967) concept of gradient analysis. However, in the gradient modeling approach, naturally occurring changes in environmental factors are modeled, and inferences are made with respect to these gradients. Thus, vegetation dynamics may be effectively modeled using the technique. Applications have included modeling forest fire patterns (Kessell 1976) and successional dynamics (Kessell and Potter 1980). Despite its rather radical departure from more conventional means of simulation, Kessell (1979b) has noted that gradient modeling and analysis, and conventional classification approaches should be complementary and integrated whenever possible.

Kessell (1979b) has emphasized the importance of vegetative information in resource management. However, he also has stressed the need for valid classifications, efficient computerized data storage and retrieval systems, and variable-resolution information delivery systems to meet the time- and site-specific information needs of managers. A colleague of Kessell, J. F. Franklin, emphasized the significance of simulation in resource management. Franklin (1979) noted that not only will simulations provide answers to "what if?" type questions, but will also synthesize large data bases into usable information in the form of scenarios and results of sensitivity analyses.

SUMMARY

The wildlife management system has evolved over the years from a simple, intensive-management oriented system into one that is dedicated to broad scale management on extensive scales. This evolutionary process has not stopped; it is continuing and is likely to continue in the future. The scope and nature of wildlife management will change; so will the information needed to support such a management effort.

If wildlife management is to become a more dynamic and effective system, it appears that new ideas, new concepts, and new technology will need to be incorporated into that system. The concepts and technology presented in this work are offered in the hopes that they will be a contribution to that end.

Chapter III

BASIC ASPECTS OF DYNAMIC CLASSIFICATION

INTRODUCTION

In order to prevent misunderstanding of the term "classification," an effort will be made to clarify it, for it will be used throughout this text. Using the terminology of systems science, "classification" denotes both a process and an output or product (cf. Mayr 1968). In the first case, the term refers to the process of identifying, labeling, ordering, or grouping a disaggregate set of objects based on predetermined similarities or differences (Platts 1980). In the latter case of output or product, the term refers to the actual assemblage of objects, the end result following the classifying process. In the former case, the term is synonymous to classifying, and it is in this sense that the term will be used here.

When classification criteria are based on similarities, the process is aggregative in nature. The use of differences as criteria results in a divisive classification approach, or in a "dissection" (Kendall 1966). Regardless of the approach, the objective remains the same. The purpose of classification is to provide conceptual or visual structure to a disaggregate set of entities. In the establishment of such structure, a measure of organization is created. This organization may lead to the formation of information, a key ingredient in the scientific and management processes.

Although classification has received extensive use in the natural sciences, it is becoming a frequent tool of management. In recent years, with the advent of multiresource land management at macro levels, classification (especially land classification) has been applied increasingly. However, it seems the full utility of land classification has not been realized by ecologists or land managers. Fundamental and theoretical problems have plagued ecological classifiers and ecological classifications. Although these problems are not unsolvable, they have created conceptual problems of enormous proportions.

One way to examine these problems is to perceive classification as a subsystem of the general management system. This subsystem possesses the 4 system components common to general systems--namely: input, output, processes, and feedback. Inputs to the classification system are observational data about the things to be classified. The nature of these data is affected directly by the informational needs and the objectives of the system user (i.e., management). System output is information in the form of ordered or organized data. The process component of a classification system involves not only the actual procedures of ordering (grouping or dividing) data or objects, but also the process of labeling, organizing, formatting, and describing the information. Feedback is the system component responsible for altering system behavior or performance so that outputs (information in context) suit the current needs and objectives of the user. Although there are problems inherent in all components of the classification system, it appears that many serious problems have arisen within the process component. Thus, a brief examination of the process of classifying is in order.

In a geometric sense, classification may be perceived as a process of grouping objects along one or more axes. In theory, all classifications are considered to be multicriteria; single-criterion classifications are simply a special case of multicriteria classifications in which only one axis is considered to be relevant. Thus, single-criterion classifications involve the definition of an appropriate interval along a single axis, while multicriteria classifications involve the definition of a finite hypervolume in n-space.

The Boundary Problem

A fundamental problem in the process of classifying, known as the boundary problem, has arisen from the assumption that things to be classified must be placed into discrete and distinct groups. Thus, the boundary problem is a matter of establishing the definitive limits (boundaries) of a grouping. Goodall (1966) noted that individuals or things to be classified comprise a multimodal continuous distribution if they are not all in the same class, whereas unimodality would indicate the presence of a single class. Goodall (1966) described classification as a process of identifying distinct modes and delineating zones of minimum frequency "in order to enable each individual to be allotted to one of the classes (modes) thus distinguished."

Historically, ecological classifiers have used natural groupings or "breaks" in distributions as a basis for establishing class boundaries. A set of diagrams showing the frequency of occurrence of objects along a single attribute continuum is shown in Fig. 1. Distinctive, discrete groups may be apparent. Such naturally occurring groupings may be due to natural factors or may be spurious aggregations occurring at random. In other instances, groupings may be indiscrete or not apparent at all and classifiers may be forced to assign class boundaries

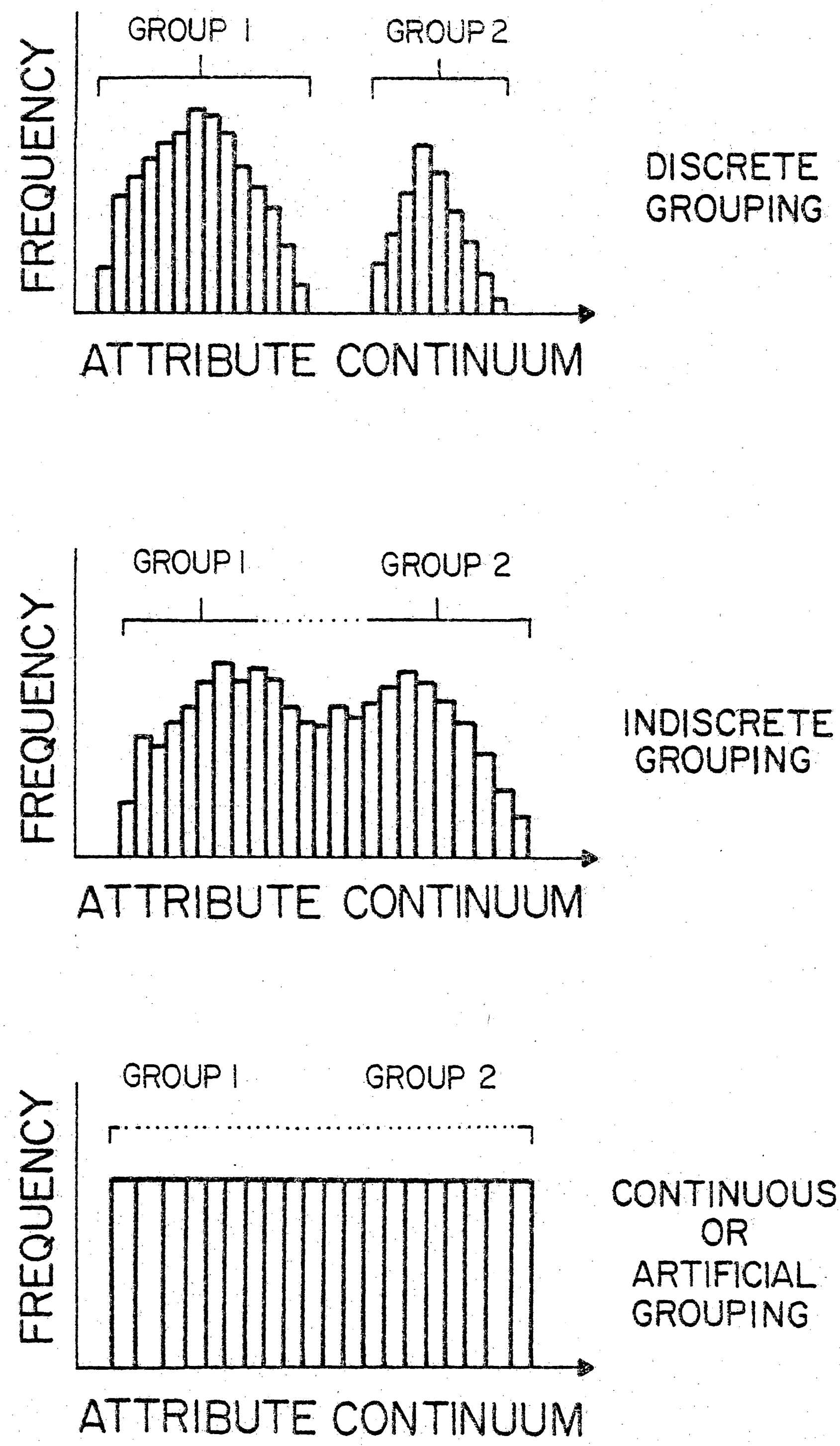


Fig. 1. Types of groupings along a single attribute axis based on frequency distributions.

arbitrarily to reflect the objectives of the classification. It is in establishing these discrete, but artificial, classes that the boundary problem is most manifest.

The boundary problem has been compounded by attempts to form discrete groups from sets of entities that are not discrete. For example, major classification (grouping) problems have developed in plant ecology. Plant ecologists following the ideas of Clements (1916,1928) had little difficulty in conceptualizing and defining vegetative groupings or communities. However, the acceptance of the vegetation continuum concept (Curtis 1955, Curtis and McIntosh 1951, McIntosh 1967) led to the denial of the existence of truly distinct, bordered vegetation communities by some plant ecologists.

An extension of the boundary problem has been the difficulty in categorizing objects possessing attributes intermediate to two groups having nontangential borders along an attribute continuum. These problems are essentially problems of assignment (Cormack 1971). Various numerical techniques have been proposed (cf. Macnaughton-Smith 1963; Sneath 1966; Wishart 1969; and Lance and Williams 1967a,1967b,1967c) to alleviate assignment problems. While some techniques have been useful in certain circumstances, none has been capable of providing a global solution.

The Generalizability Problem

The boundary problem is not the only problem in the process component of the classification system. Another inherent problem is the tendency of otherwise valid classifications to be situation- or site-specific and to become inappropriate when applied to nonrelated situations, although the things being classified may be similar. In short, it is a problem of generalizability of the classification system. Confusion may occur when an attempt is made to classify things that are foreign to the original population and that have attributes unlike those of existing groups. For example, in classifying vegetation, certain species groups are quite often categorized as a distinctive community. However, in geographically different areas, many of the same species may be present, but not together in the same relationships. Thus, the classification has failed due to site specificity.

The Obsolescence Problem

Problems also exist within the input component of the classification system. The entities being classified may themselves change with respect to their attributes. Water chemistry in lakes may change due to eutrophication, or the species composition of plant communities may be altered due to natural succession. Such natural processes may have drastic impacts on the nature of the entities being classified, whether they are lakes, biological communities, or physical objects. Thus, classifiers have been faced with the problem of dynamic, rather than static, sets of objects to be classified. For them, the problem was not actually in the classifiying process itself, but in the information derived from it (i.e., the output) becoming obsolete. An important part of the information derived from a classification is the identification of the appropriate group to which each object belongs. However, if the attributes upon which the classification of an object is based change, information may be lost. If attribute changes among the object set are significant, the entire classification may become meaningless.

The Permanency Problem

Characteristically, groups in a classification process are established under an assumed sense of permanency. Thus, class boundaries are assumed to be both permanent and sufficient. However, this is not always the case. Input in the form of objectives of the classifier may change. This change of objectives may require drastic changes in the classification process, and necessitate the addition, deletion, or substitution of new criteria in the classification.

The Relevancy Problem

Closely allied to the permanency problem is the inadequency of a classification system to supply informational output relative to the needs of various users. Groupings that were adequate and sufficient to one user may not be useful for others. Various users of classification systems may require different levels of information. Groupings may be too general for some purposes, leading to inadequate information relative to some users, while superfluous groupings may lead to confusion among users requiring only general information.

The Need for Dynamic Classification

In summary, classification is a system (as well as a technique) for assigning contextual significance to a set of disaggregate entities. It is a system that should not be static; it should be sensitive to changes over time in (1) the attributes of the things to be classified, (2) the way that things are to be classified, (3) the scale or extent, and (4) the kinds of information desired from it.

The major problems of classification may be perceived as comprising 3 major dimensions. The relevancy, permanency, and obsolescence problems may be collapsed into a single dimension, time. The boundary problem and generalizability comprise 2 other dimensions respectively. Every discrete classification situation may be seen as a point (or small space) within the volume formed by these dimensions (axes). In classification, instantaneous changes may occur along any or all axes; therefore, this point is dynamic and may occur anywhere within this volume. Thus, a classification should be appropriate anywhere within the entire volume, and not just to a particular region. What is needed is a dynamic concept to deal with a dynamic problem.

GENERAL CONCEPT OF DYNAMIC CLASSIFICATION

It would seem that a new conceptualization of the classification process could be useful in overcoming some of the above problems. This new conceptualization is contained within a concept of dynamic classification. This concept is not an extension of conventional classification, but a refinement and formalization of the principles behind conventional classification. As such, conventional classification operates within this general concept. However, dynamic classification is not constrained by many of the problems common to conventional classification procedures and thus may be a more powerful technique.

Like conventional classification, dynamic classification is taxonomic in nature; both serve to assign contextual and informational significance to sets of objects. Whereas conventional classification is a process of placing objects into discrete classes, dynamic classification categorizes along a continuum. Dynamic classification has 5 major characteristics:

1. It de-emphasizes boundaries and is gradient oriented.
2. It is deliberately tentative.
3. It is utilitarian and capable of change based on changeable objectives.
4. It is not hierarchical since any number of attributes may be used as relevant criteria.
5. It is not necessarily linear, hierarchical, phylogenetic, or with evolutionary foundation.

Conventional classification methodology involves examination of objects or entities. (For the sake of simplicity, they will be called objects.) These objects possess certain traits (called the relevant attributes) which may be measured along a series of continua. Similarity among objects is determined (sometimes arbitrarily) and groups are established by splitting the respective continua into segments. For each axis, which is an attribute continuum, objects having attributes within the range of a specific segment are assigned to that segment. A group of objects exists in a segment and is considered to be essentially homogeneous.

In contrast, dynamic classification does not have the group as the fundamental unit. Rather, the set of objects to be classified and the multidimensional set of relevant attributes are fundamental. The objective in dynamic classification is to identify the relevant attributes of each object and to describe the object in terms of these attributes.

The Theoretical Nature of Objects

The term "object" refers to an entity to be classified. In general, objects are usually physical (but may not be), and can be defined with respect to a set of attributes. Objects possess an infinitely large set of descriptive attributes, and under the general concept of dynamic classification an object is defined by that set of descriptive attributes. Although the object to be classified may be a physical reality, it is considered to be dimensionless under the concept of dynamic classification.

That is, the object exists conceptually as an infinitely large set of descriptive attributes. The physical referrents such as weight, color, and texture, that give conceptual meaning to the observer are contained within the attribute set. Therefore, if the observer knows something about the attribute set, he knows something about the object.

The Theoretical Nature of Attributes

The concept of an attribute is central to the concept of dynamic classification. Attributes provide a frame of reference for the description of objects. A given attribute exists as a continuum, with the precise characteristics of a given object occupying a point this continuum. The precise point occupied by an object along an attribute continuum is called the "characteristic" of that object with respect to that attribute. It should be remembered that many attributes may be associated with a given object, and therefore, that object may possess many characteristics with respect to these attributes. However, the number of characteristics is equal to the number of attributes at any given instant in time.

In most cases, an attribute axis may be continuous. However, it is conceivable that this may not always be so;

an axis may be discrete in nature. For example, in one sense, the nature of an electromagnetic charge is not continuous; it is either positive or negative. In such cases, the attribute axis is retained within the concept of dynamic classification, but it is reduced to a line (axis) of only 2 points--a point representing a positive charge, and one representing the negative.

From a theoretical standpoint, attributes are empirical in nature. However, in reality, precise and accurate measurement of some attributes may be difficult if not practically impossible. While the problems of accuracy and precision may be bothersome, they do not detract from the basic premise that the characteristics of a given object are empirical (at least theoretically) and may be represented as a series of points along a series of attribute continua.

Although the total number of attributes possessed by or assignable to an object may be infinitely large, the number used to describe a particular object may be finite. Attributes may be used to describe the physical, functional, or esthetic nature of an object. Examples of physical attributes are weight or volume. Forest site index is an example of a functional attribute, while opportunity cost exemplifies an esthetic attribute.

Attributes may be placed in 2 main groups. They may be labeled as "natural" if the attribute is a naturally occurring characteristic of the object. However, if an attribute exists solely within the context of a particular classification process, it may be called a synthetic attribute. An example of this type would be the suitability of an object for some purpose relevant to the objectives of the classification. In this case, the index of suitability exists only for the purposes of a specific classification scheme, despite the fact that the suitability exhibited by the object for the intended purpose is an inherent characteristic of that object.

Assumptions

Implicit in the general concept of dynamic classification are the following fundamental assumptions:

(1) Physical entities may be classified on the basis of one or more definable attributes or combinations of these attributes;

(2) The total set of attributes possessed by a physical entity is potentially infinitely large;

(3) Some of these attributes may be measured quantitatively with acceptable accuracy;

(4) Attributes may be characterized as continua with ranges from zero (total absence) to infinity (or saturation), or from 0 to 100 percent;

(5) The frequency distribution of characteristics of a finite set of objects with respect to a particular attribute along that attribute continuum may be normal, but is not necessarily so.

(6) Attributes are not fixed and may change over time.

(7) Attributes are not necessarily independent of one another.

(8) A single phenomenon may be perceived with respect to more than one attribute due to differences in role or function in a complex system, or due to objectives of the classification itself.

Limitations

Unfortunately, the dynamic classification concept is not without limitations. In dynamic classification, the potential temporal variability of attributes is acknowledged and dealt with explicitly. However, because performing continuous classifications is technologically and financially impractical, both dynamic and conventional classifications must be done at temporal intervals. However, using simulation techniques, classifications (both dynamic and conventional) may be done at any point in time, whether that point is in the past, present, or future.

Due to the impracticality of defining and measuring an infinite number of attributes for each object (even though the object may have an infinite number of characteristics on which classification may be based), a finite set of attributes must be used. This may be necessitated by the lack of available information on the total attribute set or the infeasibility of obtaining this information. In actuality, it is impossible to have complete and perfect knowledge of an object. Thus, in the real world, attribute sets of a size limited by practicality must be used.

CONVENTIONAL VERSUS DYNAMIC CLASSIFICATION

To demonstrate the fundamental difference between conventional classification and dynamic classification, an analogy will be drawn which involves a set of objects (where the objects may be tangible or intangible). For the sake of simplicity, the example will be restricted to a single dimension. All objects are assumed identical except for a single relevant attribute, light reflectivity. In the set of objects, some are black (or almost black), some are white (or almost so), and some are varying shades of gray. Using conventional classification, an attempt would be made to establish a finite number of groups from the attribute set. However, in dynamic classification, groups are not created. Rather, the task is one of determining where along the attribute continuum each object is to be placed. This is accomplished by accurate quantitative measurement, such as obtaining a light meter reading for each object. Thus, in conventional classification, a number of groups containing similar objects would be established, with the number of groups varying as a function of the information needs of the user. Regardless, the number of groupings established in a single-factor conventional classification will be less than or equal to the total number of objects to be classified. (However, when the number of groups equals the number of objects, part of the utility of conventional classification, namely aggregation, has been lost.)

A primary difference in the two approaches is in the amount of information gained from the classification process. In conventional classification, information consists of (1) a generalization about the attributes of each group, and (2) the constituent members of each group. Beyond the group level, nothing is known about the characteristics of individual group members. In a dynamic classification, generalizations are not made, as the precise attribute characteristics about each object in the set are considered. In effect, each object is a group, except for those groups composed of identical objects (e.g., those having identical light reflectance).

The generalizability problem, that is, the inability of a classification scheme to be universally applicable, does not occur in dynamic classification. In conventional classification, emphasis is placed on determining similarity among objects. An effort is made to determine which are most alike or different. Secondary emphasis (if any) is placed on the significance of the attribute continuum. Frequently, in conventional classification, not only is the entire range of the attribute continuum unknown, but only portions (i.e., those containing groups) are considered relevant. However, in the case of dynamic classification, the attribute continuum is fundamental to the categorization process. Thus, with the a priori assumption that the attribute continuum exists and is definable, the problem of where to place objects unlike any that have been classified previously is reduced to the task of deriving a quantitative measurement of the new object.

In conventional classification, the assumption that the attributes of an object are constant is implicit. This assumption is not made in dynamic classification. Attributes of objects are recognized as potentially changeable; thus, the temporal limitations of a classification are realized. However, in dynamic classification, attributes known to be dynamic over time may be predicted by techniques such as computer simulation, thereby overcoming temporal restrictions.

The fundamental problems of conventional classification mentioned previously are not unique to single-criterion classification, but are also encountered in multifactor classification. Whereas single-factor classification may be perceived as operating in a single dimension, multifactor classification is an attempt to categorize objects in a multidimensional hypervolume. For example, a conventional classification in 3-space (i.e., in 3 dimensions) would entail 3 relevant attributes. With such a classification, a class is represented as a discrete (hyper)volume and the objective is to place objects within appropriate (hyper)volumes. In contrast, in a dynamic classification approach, classes are assumed to exist as geometric points, and objects are categorized based on their respective relative coordinate positions within the hypervolume defined by the attribute continua axes.

In dynamic classification, an infinite set of attributes is assumed explicitly; no such assumption is made in conventional classification. Any number of these attributes may be used in dynamic classification. Herein lies the utility of the dynamic classification concept. By using selected attributes, specific information relative to precise informational needs of the user can be met. Thus, the power of dynamic classification as a dynamic source of precise information is apparent. However, the use of only selected attributes does not preclude or deny the existence of a very large number of other attributes. Rather, the use of selected attributes may reflect specific informational needs of a user.

With conventional classification, a class is expressed as a volume (or hypervolume), with physical dimensions fixed over time. However, in nature, entities rarely remain constant or fixed. Thus, with conventional classification, entities whose characteristics tend to change over time may be said to migrate from (hyper)volume to (hyper)volume, depending on the current status of changeable attributes. In dynamic classification, changes in the characteristics of an object are manifest in the migration of a point (representing the object) through a hyperspace. This migration will follow a trajectory dictated by the rate of change of the characteristics with respect to the entire attribute set. The benefit of dynamic classification in this sense is that it allows the examination of the precise trajectory (or behavior) of individual entities over time, whereas conventional classification allows only an examination of the progression or sequence of classes occupied by an object through time in a steppingstone-like manner.

THE INTERCONNECTEDNESS OF CLASSIFICATION CONCEPTS AND ECOLOGICAL THEORY

In his classic discussion of the theoretical concept of the niche hypervolume, Hutchinson (1958) proposed that a species may exist and persist within a bounded hypervolume. The axes of this hypervolume represent all the environmental gradients operant on the species under study. Upon closer examination, it can be seen that the definition of a niche hypervolume is analogous to the definition of a class in conventional classification. Thus, articulation of a niche hypervolume is in essence a classification effort in which the group (i.e., the niche hypervolume) is a delineation of the suitable conditions under which a given species may exist.

Under conventional classification theory, classes are assumed to be essentially homogeneous (Cormack 1971). That is, the space inside the hypervolume is homogeneous with respect to the objectives of the classification. However, it cannot be said that the space inside a niche hypervolume is homogenous. As Wuenscher (1974) stated succinctly, "Clearly, some regions within the n-dimensional niche volume will be more favorable for ... [some] species than other species." Hutchinson's (1965) concept of realized niche also supports this contention.

Under the premise that certain regions within a niche hypervolume represent conditions more suitable to the species than those of other regions, it must be assumed that an optimum region, or point, exists within that hypervolume. This optimum region may occur anywhere within the niche hypervolume and no assumption of a central tendency should be implied. If this is indeed so, then the location of that region or point may be derived by defining the optimum range of conditions, or optimum point(s), along each of the hypervolume axes.

In conventional habitat classification, a finite number of environmental parameters (axes) are extracted from the niche hypervolume. These parameters are then used to construct axes for a classification unit. Boundaries are set along each axis, and the resultant hypervolume is considered to represent suitable habitat. Considering land areas to be objects, these areas may be categorized as being suitable or unsuitable habitat for particular species. More specific classifications may be performed by creating a series of classes that converge concentrically on the region of optimality with respect to the axes considered. Conventional habitat evaluation utilizes a similar procedure. However, in habitat evaluation, the objective is essentially one of determining how closely the environmental conditions on the site being evaluated approximate the optimal environmental conditions within the niche hypervolume.

In a dynamic classification, an object (e.g., parcel of land) can be described by a point within a hypervolume defined by attribute axes. Using this concept, it is theoretically possible to define an object (here a parcel of land) as a point within the niche hypervolume of a particular species. Similarly, using dynamic classification, it is possible to define that object as a point within a hypervolume defined by a finite subset of niche axes.

Thus, the interconnectedness of classification theory and ecological theory in the form of the niche hypervolume concept is apparent. They are compatible conceptually in that both can be shown to share a common geometric language, as both attempt to explain multivariate relationships in geometric terms.

APPLICATIONS OF DYNAMIC CLASSIFICATION IN WILDLIFE MANAGEMENT

The general concept of dynamic classification has many potential applications. However, the particular use that was examined in this research was in the field of wildlife management.

Management, wildlife management included, thrives on pertinent information. In the early days of wildlife management, major emphasis was placed on studying life histories to derive necessary information for management. As the habitat and niche requirements of more and more wildlife species became known, emphasis was shifted to analyzing habitat in an effort to determine which areas could provide the necessary animal requirements. Habitat evaluations followed and served to synthesize analysis data into information about the suitability of an area for various wildlife species.

However, in recent years, wildlife management efforts have been engaged in more extensive scales than in the past. Many wildlife managers are becoming responsible for a wide variety of species. Others are finding themselves responsible not only for wildlife management, but for a variety of natural resources falling under the umbrella of land management. Along with the expanding area of responsibility has come a need for more extensive information about the land and its suitability for a variety of uses, including use as wildlife habitat.

Fortunately, technological advances have been made to meet some of the information needs of the modern land manager. The recent advent of extensive geographic information systems has changed the perspective of land classification. Governmental efforts, such as LANDSAT and GIRAS (Mitchell et al. 1977) can provide vast amounts of extensive data on the land resource while maintaining high degrees of spatial data resolution. Future projects will undoubtedly produce even higher-resolution data.

Thus, technology is at present capable of providing the land manager with extensive data bases at a resolution appropriate for fairly intensive management efforts. Unfortunately, conventional classifications have been used by land managers to synthesize these data into easily assimilated forms. Inasmuch as conventional classification is a process of generalizing, extensive data often become reduced to information of a general nature. It is therefore proposed that the use of dynamic classification would avoid the inadvertent loss of resolution in information resulting from generalized conventional classification schemes. Specifically, in using dynamic classification to categorize wildlife habitat, it is believed that information may be produced at the same resolution as the baseline data (i.e., the original set of objects to be classified), or when necessary, this information may be reduced to a more generalized form. It will be argued that information derived from dynamic classification is potentially of a finer resolution and is more attuned to the informational needs of the classifier than information derived via conventional means.

Chapter IV

METHODOLOGY

INTRODUCTION

Herein, an attempt was made to propose a concept. An articulation of the fundamental nature of the concept and a declaration of its need were basic aspects of the methodology of the research described. Within this chapter, emphasis will be on demonstrating the utility of the concept and on validating that concept.

UTILITY

In this research, a dynamic classification approach was applied within a computerized wildlife management information system. Specifically, information about wildlife habitat was produced under the contention that increased habitat information could result in increased effectiveness of certain wildlife management efforts. Because of the importance of spatial arrangements within wildlife habitats, information produced by this system was to be displayed as maps.

Thus, a primary goal of this research was to develop a system that (at least in theory) could supply the type of information needed at the resolution required. In order to permit widespread usage of this experimental application, the information system was designed to be deliberately simple and easy to use by potential users lacking extensive computer training and equipment. It was believed that by articulating a methodology for this information system system, the advantages and limitations of dynamic classification might be discovered and made explicit.

Study Areas

Shenandoah National Park

In order to use dynamic classification in a variety of situations, 3 study areas were chosen. One study area was a 8750 ha portion of Shenandoah National Park contained within the McGaheysville Quadrangle. The entire Park is located along a narrow, irregular mountain ridge running from Front Royal, Virginia to Waynesboro, Virginia. Shenandoah has been categorized by Braun (1950) as representative of the Northern Blue Ridge Section of the Oak-chestnut Forest Region within the Eastern Deciduous Forest Formation. Ridge tops and slopes are generally xeric, while the sparse coves are characteristically mesic, and support cove hardwoods. Although overall forest site quality is low, the Park supports large populations of white-tailed deer (*Odocoileus virginianus*) and black bear (*Ursus americanus*), as well as other wildlife species.

George Washington National Forest

A second 1927 ha study area was located on Peters Mountain in the James River Ranger District of the George Washington National Forest, Virginia. Vegetation of the area has been described in detail by Adams (1974). Like Shenandoah National Park, the area is typical of the Appalachian uplands. However, unlike the Shenandoah area, this area is currently managed for timber production, and public hunting is permitted. The area supports populations of a variety of wildlife species; however, numbers of the large game animals are typically lower than in the Shenandoah area, perhaps due to the sanctuary conditions in the former area.

The Great Dismal Swamp

A third area used in this study was the Great Dismal Swamp, located near Suffolk, Virginia. This study area was very different from the previous areas in that it was a coastal plain hardwood swamp. The other areas were mountainous hardwood forests. The Dismal Swamp study area was considerably larger than the other study areas, comprising approximately 84,890 ha (Gammon and Carter 1979). Predominant overstory trees in the Dismal Swamp included black gum (*Nyssa sylvatica*), swamp tupelo (*N. aquatica*), and red maple (*Acer rubrum*) (Gammon and Carter 1979, Levy and Walker 1979). Common mast bearing trees such as oaks (*Quercus* spp.) and hickories (*Carya* spp.) were conspicuously absent due to their intolerance to the periodic flooding common to the Dismal Swamp (Penfound 1952). The Dismal Swamp National Wildlife Refuge presently contains much of the remaining portions of the original Dismal Swamp. The Dismal Swamp currently supports a variety of wildlife species and at least 30 mammalian species are endemic to the area (Meanley 1971). The Dismal Swamp also serves as a refuge for virtually the entire breeding population of black bear in eastern Virginia (Bureau of Sport Fisheries and Wildlife 1974).

Study Species

Several wildlife species were studied in order to provide diverse applications in using dynamic classification to provide wildlife habitat information. White-tailed deer, wild turkey (*Meleagris gallopavo*), and gray squirrel (*Sciurus carolinensis*) were chosen for study in the George Washington National Forest Study area. These 3 species were selected as study species on the George Washington National Forest study area because they were typical of the game species managed on National Forest lands. The black bear has been the source of much concern and management effort by officials in Shenandoah National Park and was therefore chosen as a study species in that study area. The black bear was the single species studied in the Dismal Swamp area, based on specific objectives of a sponsoring agency (the Great Dismal Swamp National Wildlife Refuge).

Selection of study species was admittedly both arbitrary and opportunistic. Regardless, these wildlife species were typical of the many species of wildlife (both game and non-game) being managed on public and private lands and about which information is needed by managers. These species were therefore regarded as representative study species for an application of dynamic classification to wildlife management.

Data Base Construction

Prior to actual data collection, it was necessary to determine exactly what types of data were needed. This began with an articulation of a set of relevant attributes by those parties considered to be the users (i.e., management personnel for the respective study areas).

Further insight into the types of information needed was gained through the administration of questionnaires to experts in the field. Questionnaires were sent to 55 white-tailed deer biologists, 37 black bear biologists, and 7 black bear biologists familiar with bears in the Atlantic coastal region. However, in all cases, the task of data collection was one of collecting data pertinent to the information needs of management.

In some cases where natural attributes were considered among the set of relevant attributes, data about these attributes could be collected directly from existing sources. For example, data on distance to water were recorded directly from maps. In cases where direct measurement of natural attributes was impractical (e.g., hard mast production), data were estimated using known relationships between the estimated attribute and existing data sources (e.g., forest stand type and stand age).

Collection of data relative to the formation of information on synthetic attributes was performed in a similar manner. Where possible, data pertinent to the creation of synthetic attributes were collected directly. When this was impossible or infeasible, estimation was necessary.

Selection of natural, relevant attributes was based primarily on the stated information needs of the managing parties of the respective study areas. Establishment of this attribute set was accomplished by a literature review, personal communication with biologists, through the questionnaires sent to biologists, and from information from officials responsible for the respective study areas.

Data Encoding

From a strict theoretical standpoint, a land area may be considered to be a bounded surface composed of an infinitely large number of parcels, each possessing an infinitely small area. Each of these points of land then may be considered to be an object subject to dynamic classification. Although using an infinite number of infinitely small parcels to describe a land area would maximize the amount of descriptive data about that area, such an approach is obviously impossible, since it would necessitate the generation and manipulation of virtually infinite amounts of data. Thus, a method for systematically sampling the characteristics of a land area was needed.

Fortunately, the underlying theory of dot systems for encoding data (Meyers et al. 1979) is similar to that of dynamic classification. Dot systems are similar to grid cell systems in that regular patterns are overlaid on a base map. However, unlike cell systems, in which the predominant feature or attribute level of a cell is recorded, in dot systems the attribute level at the dot is recorded. Thus, dot systems are essentially systems for sampling attribute levels across a landscape continuum.

Dot grids were arranged in a regular rectangular pattern, with 120 rows and 120 columns per grid. Grids were aligned along the true north-south axis of maps to be encoded. Using the 120 by 120 grid pattern registered to a 7 1/2 min. United States Geological Survey topographic base map, dot centers represented on-the-ground distances of 97.08 m in the north-south direction and 115.66 m in the east-west direction.

The selection of a 120 by 120 grid (data matrix) was strictly a matter of convenience; grids of different dimensions could have been used. However, an array of these dimensions is handled easily by computer programs and represents almost a full page width when displayed by a computer line printer. Selection of the 7 1/2 min. topographic maps as base maps was also somewhat arbitrary. Use of topographic maps as base maps resulted in computer-generated maps in which a printed character represented approximately 1.1 ha.

Data on relevant attributes for each sample point (dot) were collected directly from existing maps, or were derived indirectly, using known relationships of the attribute to map information. A separate map was encoded for each relevant attribute and for those characteristics to be used in generating synthetic attributes.

Data to be used for subsequent computer analysis or display were encoded both manually and through electronic means. In either case, all thematic maps to be encoded were registered to an appropriate topographic base map.

In the manual encoding procedure, a transparent grid sheet of the proper dimensions was overlaid on the map to be coded. Because of specific input requirements of the computer program used to read map data, maps were encoded as left and right halves. That is, each map was encoded as a series of rows, each with 60 columns. Thus, each map was composed of 14,400 data elements in 240 rows consisting of 60 rows each. All thematic data were recorded on standard data coding sheets. Data from these sheets were then keypunched onto 80-column punched cards.

Early into the project, electronic digitizing equipment became available, permitting a rapid method for encoding data. A Numonics Model 237 Graphics Calculator/digitizer (Numonics Corp., Landsdale, Pennsylvania) and a Tektronix 4051 minicomputer (Tektronix Inc., Beaverton, Oregon) were used to encode data electronically. Using this equipment, areas containing similar amounts of a given attribute were digitized as polygons. Thus, an entire map of a particular attribute was recorded as a patchwork-like aggregation of polygons. Modifications of 2 commercially-available computer mapping programs (Federation of Rocky Mountain States 1977) were used to convert this polygon data into a grid format.

Late in the study, a Tektronix 4956 digitizing tablet and the 4051 minicomputer were used to encode data. Using this equipment, data were recorded in cell format and a grid overlay was placed over the map to be coded. The 4051 minicomputer was programmed to allow the data coder to define the perimeter cells of a homogeneous area, and then fill in the interior cells with the desired value or symbol. Thus, in some respects, this procedure was similar to the polygon procedure.

Software Development

Several special-purpose computer programs were developed to assist in converting raw digitized data into a format suitable for further processing into maps. However, due to the peripheral nature of these programs, they will not be discussed further.

The MAP4B Mapping Program

A computerized mapping system, MAP4B, was developed to read and manipulate areal data. Input to the MAP4B system was done via 80-column punched cards, or as card images when the system was run from a remote terminal. Data were stored internally by the program during execution as a series of 120 by 120 matrices, consisting of either numeric or symbolic data. Special program subroutines were used to convert symbolic data into the necessary numeric format prior to data manipulation.

Output from the MAP4B system was in the form of a 30.5 by 38.1 cm computer line printer map consisting of 14,400 printed elements arranged in 120 rows of 120 columns each. Row and column numbers were supplied along map borders for reference. MAP4B was used to display original input data in map format (Fig. 2). This procedure was valuable in checking input data for accuracy, and in detecting coding errors.

Primary utility of the MAP4B system was to manipulate input data matrices to produce composite maps such as the one shown as Fig. 3. These maps were generated by either summing or taking the product of numeric data matrices on a cell-by-cell basis.

MAP4B could be used to provide a display of original input data in map format. However, in situations where data were numeric, or where a composite map was to be produced, a series of 10 contrasting symbols was used to display

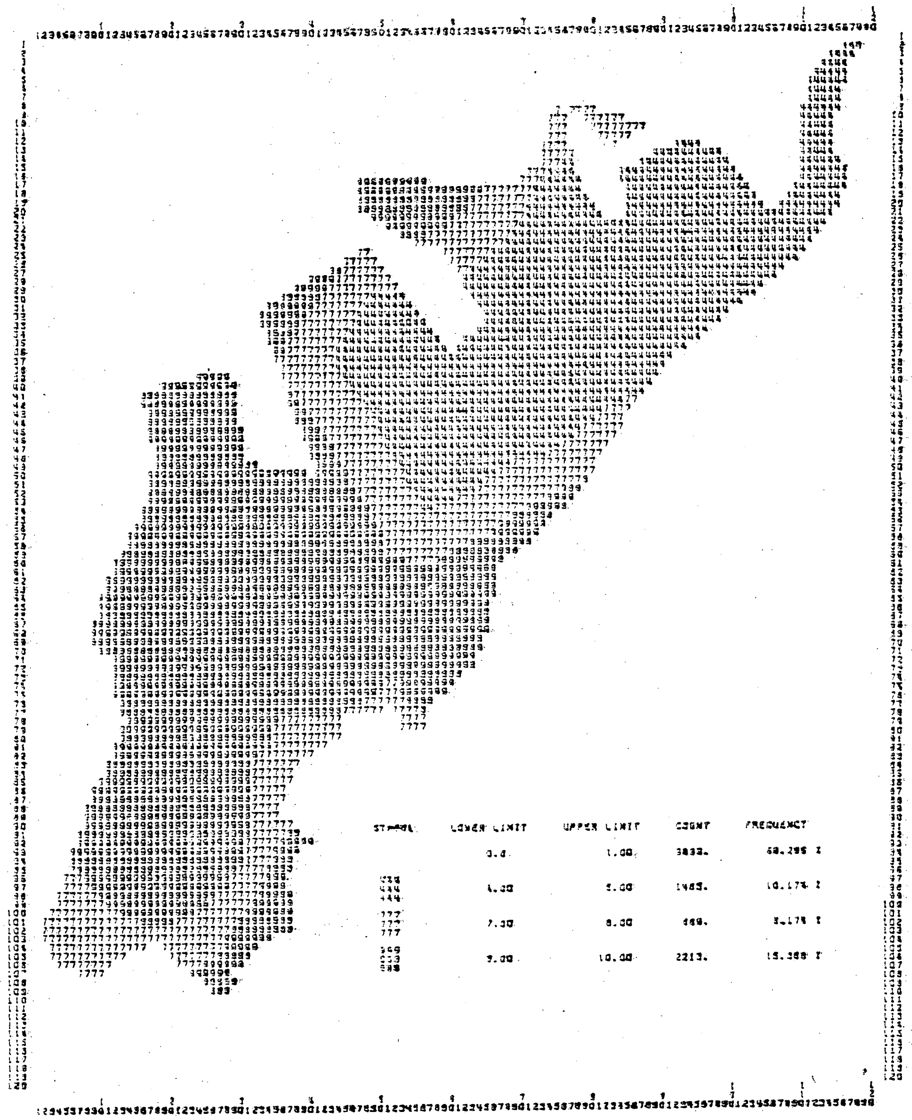
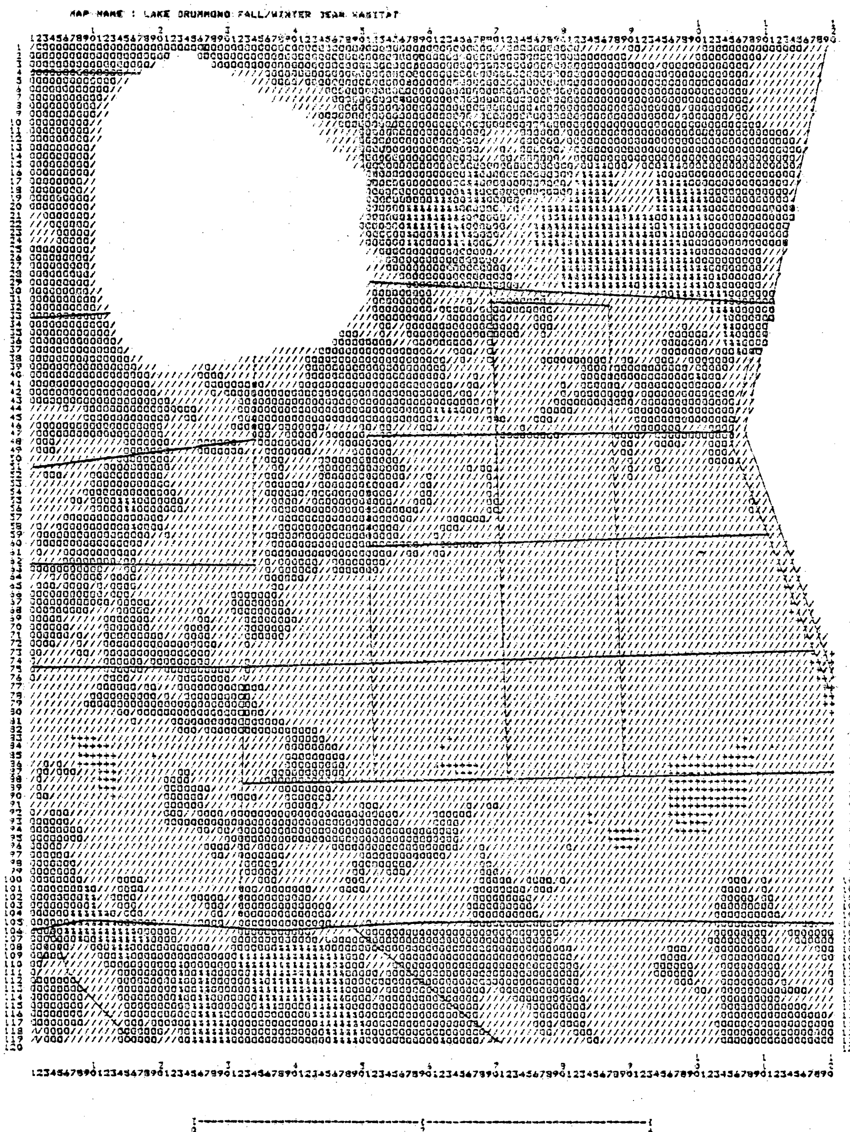


Fig. 2. A computer-generated map of an original data set (index of habitat suitability as a function of distance to free water) for the George Washington National Forest study area, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	290.00	1829.	12.701 %
+++ +++ +++	1450.00	1740.00	184.	1.278 %
/// /// ///	1740.00	2030.00	6550.	45.484 %
000 000 000	2030.00	2320.00	5273.	36.613 %
111 111 111	2320.00	2610.00	564.	3.717 %

Fig. 3. Computer-generated map of bear habitat suitability for the Lake Drummond Quadrangle, Dismal Swamp, Virginia.

characters. In this case, values in the output matrix were scaled linearly by a subroutine and placed into 1 of 10 categories. The value of each element in the display matrix was displayed as an appropriate contrasting symbol.

The MAP4B system will not be dwelt upon here. Specific information concerning the system can be found in the document entitled "MAP4B Users' Guide" which has been included as Appendix A.

The PREDIC Program

It was noted previously that classification problems may result when relevant attributes of objects are subject to change over time. When these changes are deterministic in nature, simulation or prediction may serve to forecast future attribute characteristics.

In this study, direct measurement of some relevant attributes (e.g., hard mast production) was not possible. However, quantitative estimates of such attributes are possible and were derived from available vegetation data such as species composition and age structure of the overstory tree species. In this study, various relevant attributes were derived from vegetation data.

To a large extent, changes in vegetative conditions, such as those caused by succession, may be considered to be deterministic, and thus, predictable. A computer program, PREDIC, was used to forecast specific vegetative conditions which were useful in estimating certain relevant attributes. The PREDIC program was capable of predicting percent species composition and diameter class distribution of major tree species on an area of land. In the application described here, the areas of land about which vegetative predictions were made corresponded to the sample points in the 120 by 120 grid of sample points described previously.

A Markov chain approach (Moser 1978 and Hillier and Lieberman 1980:372-390) was used within the program to predict diameter class distributions. A Boolean approach was used to determine tree species likely to regenerate successfully on the site, based on selected biotic conditions.

A Markov transition matrix (Table 1) was developed for each of 16 major tree species endemic to the Shenandoah National Park and George Washington National Forest study areas. These matrices were based on radial growth tables and information found in Forbes (1955), Fowells (1965), Gingrich (1971), Harlow and Harrar (1969), Ike and Happuch (1968), Nelson et al. (1961), Olson (1959), Schnurr (1937), Smalley and Bailey (1974), and Spaeth (1920). Concurrently, an autecological table (Table 2) representing the occurrence

Table 1. Markov transition matrices used to predict age distributions for 16 overstory tree species.

Species	Origin dbh Class (inches)	Destination dbh Class (inches) Open	0-2	2-5	5-8	>8
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.676	0.324	0.000	0.000
Chestnut oak	2-5	0.000	0.000	0.867	0.133	0.000
	5-8	0.000	0.000	0.000	0.929	0.071
	>8	0.038	0.000	0.000	0.000	0.962
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.700	0.300	0.000	0.000
Scarlet oak	2-5	0.000	0.000	0.858	0.142	0.000
	5-8	0.000	0.000	0.000	0.911	0.089
	>8	0.034	0.000	0.000	0.000	0.966
	Open	0.400	0.600	0.000	0.000	0.000
	0-2	0.000	0.676	0.324	0.000	0.000
White oak	2-5	0.000	0.000	0.867	0.133	0.000
	5-8	0.000	0.000	0.000	0.929	0.071
	>8	0.022	0.000	0.000	0.000	0.978
	Open	0.400	0.600	0.000	0.000	0.000
	0-2	0.000	0.700	0.300	0.000	0.000
Black oak	2-5	0.000	0.000	0.858	0.142	0.000
	5-8	0.000	0.000	0.000	0.911	0.089
	>8	0.037	0.000	0.000	0.000	0.963
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.700	0.300	0.000	0.000
Red oak	2-5	0.000	0.000	0.858	0.142	0.000
	5-8	0.000	0.000	0.000	0.911	0.089
	>8	0.025	0.000	0.000	0.000	0.975
	Open	0.400	0.600	0.000	0.000	0.000
	0-2	0.000	0.750	0.250	0.000	0.000
Bear oak	2-5	0.000	0.000	0.870	0.130	0.000
	5-8	0.000	0.000	0.000	0.926	0.074
	>8	0.038	0.000	0.000	0.000	0.962
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.124	0.876	0.000	0.000
Pitch pine	2-5	0.000	0.000	0.754	0.246	0.000
	5-8	0.000	0.000	0.000	0.900	0.100
	>8	0.050	0.000	0.000	0.000	0.950

Table 1. Markov transition matrices used to predict age distributions for 16 overstory tree species (continued).

Species	Origin dbh Class (inches)	Destination dbh Class (inches) Open	0-2	2-5	5-8	>8
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.124	0.876	0.000	0.000
Virginia pine	2-5	0.000	0.000	0.754	0.246	0.000
	5-8	0.000	0.000	0.000	0.900	0.100
	>8	0.050	0.000	0.000	0.000	0.950
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.124	0.876	0.000	0.000
Table Mountain pine	2-5	0.000	0.000	0.754	0.246	0.000
	5-8	0.000	0.000	0.000	0.900	0.100
	>8	0.050	0.000	0.000	0.000	0.950
	Open	0.400	0.600	0.000	0.000	0.000
	0-2	0.000	0.675	0.325	0.000	0.000
White pine	2-5	0.000	0.000	0.834	0.166	0.000
	5-8	0.000	0.000	0.000	0.908	0.092
	>8	0.020	0.000	0.000	0.000	0.980
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.117	0.883	0.000	0.000
Yellow poplar	2-5	0.000	0.000	0.853	0.147	0.000
	5-8	0.000	0.000	0.000	0.857	0.143
	>8	0.024	0.000	0.000	0.000	0.976
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.716	0.284	0.000	0.000
Black locust	2-5	0.000	0.000	0.853	0.147	0.000
	5-8	0.000	0.000	0.000	0.937	0.063
	>8	0.070	0.000	0.000	0.000	0.930
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.500	0.500	0.000	0.000
Gray birch	2-5	0.000	0.000	0.670	0.330	0.000
	5-8	0.000	0.000	0.000	0.750	0.250
	>8	0.250	0.000	0.000	0.000	0.750
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.117	0.883	0.000	0.000
Sycamore	2-5	0.000	0.000	0.647	0.353	0.000
	5-8	0.000	0.000	0.000	0.857	0.143
	>8	0.020	0.000	0.000	0.000	0.980

Table 1. Markov transition matrices used to predict age distributions for 16 overstory tree species (continued).

Species	Origin dbh Class (inches)	Destination dbh Class (inches) Open	0-2	2-5	5-8	>8
	Open	0.200	0.800	0.000	0.000	0.000
	0-2	0.000	0.716	0.284	0.000	0.000
Red maple	2-5	0.000	0.000	0.853	0.147	0.000
	5-8	0.000	0.000	0.000	0.937	0.063
	>8	0.071	0.000	0.000	0.000	0.929
	Open	0.400	0.600	0.000	0.000	0.000
	0-2	0.000	0.888	0.112	0.000	0.000
Hemlock	2-5	0.000	0.000	0.925	0.075	0.000
	5-8	0.000	0.000	0.000	0.944	0.056
	>8	0.012	0.000	0.000	0.000	0.988

Table 2. Autecological table. The presence of a 1 in a column under a species number indicates that the species is found in conjunction with the indicated environmental factor.

Factor		Species Number[1] 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Slope:	≤5°	0	0	1	0	1	0	0	0	0	1	1	0	0	1	1	1
	6–10°	1	0	1	0	1	0	0	0	0	1	1	1	0	1	1	1
	11–20°	1	0	1	0	1	0	0	1	0	1	1	1	1	0	0	1
	21–45°	1	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1
	46–65°	1	1	0	1	0	1	1	1	1	0	0	1	1	0	1	0
	>65°	0	1	0	1	0	1	1	0	1	0	0	0	0	0	1	0
Streams:	present	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	1
	absent	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0
Catastrophe:	yes	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0
	no	1	1	1	1	1	1	0	0	1	1	0	0	0	0	1	1
Elevation:	≤1000 ft	1	1	1	1	1	1	0	1	0	0	1	1	1	1	1	0
	1001–2000 ft	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0
	2001–3000 ft	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
	3001–4000 ft	1	0	1	1	1	1	0	0	1	1	1	1	0	0	1	1
	4001–5000 ft	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1
	>5000 ft	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Percent open area:	≤1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	1.0–2.99	1	0	1	1	1	1	0	0	0	1	0	0	0	0	0	1
	3.0–6.0	1	1	1	1	1	1	0	0	0	1	0	0	0	1	1	0
	>6.0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
Aspect:	0–89°	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1	1
	90–179°	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0
	180–269°	1	1	0	1	0	1	1	1	1	0	0	1	1	1	1	0
	270–360°	1	0	1	0	1	0	0	0	0	1	1	1	1	1	1	1

[1]Species numbers are as follows: 1 = Chestnut oak; 2 = Scarlet oak; 3 = White oak; 4 = Black oak; 5 = Northern red oak; 6 = Bear oak; 7 = Pitch pine; 8 = Virginia pine; 9 = Table mountain pine; 10 = White pine; 11 = Yellow poplar; 12 = Black locust; 13 = Gray birch; 14 = Sycamore; 15 = Red maple; 16 = Hemlock.

of major tree species as a function of the biotic factors of slope, aspect, elevation, evidence of catastrophe, proximity to running water, and percent open area in the vicinity of the site was prepared based on information from Baker (1949), Fowells (1965), Harlow and Harrar (1969), Putnam et al. (1960), Shelford (1963), and Society of American Foresters (1954). Scientific names of tree species used in this study are provided as Appendix D.

Both the Markov transition matrices and the autecological table were cursory rather than definitive in nature. Since both data sets are used as inputs into the PREDIC program and were not integral within the program proper, necessary modifications and refinements may be made easily by altering the data. In practice, ultimate responsibility for the accuracy of all input data, including these 2 tabular data sets, rests with the user.

Within the program, the autecological table was used to create a list of species suited to the conditions present on the site under consideration. Next, the program was supplied with a series of vectors composed of the percentages of the parcel occupied by the respective tree species, plus any open area (i.e., percent of the area not considered to be under a tree canopy). The percent of the area not under canopy was assumed to be unvegetated and was apportioned equally among those species determined to be suited to the site.

The program was then used to multiply the proportion of the area of the site occupied by each species by the appropriate diameter distribution vector for each species. The set of resultant vectors, along with those portions of the area considered to be subject to regeneration by the respective species, was then subjected to the Markov procedure.

Technically, the results of a Markov procedure are a series of probabilities indicating the probability of an object being in a certain state (i.e., a given diameter size class). For example, a resultant vector such as:

$$A = (.10, .10, .70, .10)$$

would indicate that the probability of being in either the first, second, or fourth state (diameter class) is 0.10, while there is a 70 percent chance of being in the third age class. However, these figures can be translated into proportions of an area occupied by individuals of each diameter size class. That is, it can be said that 70 percent of the area within the parcel (cell) is occupied by individuals within the third diameter size class, with the remainder of the area being comprised equally of individuals in the first, second, and fourth diameter groups.

The vector resulting from the multiplication of the initial-conditions vector for each species present and the appropriate transition matrix (i.e., a table representing the probability of changing over time from one diameter distribution class to another) represented the updated diameter distribution vector. To this vector was added the percent of unvegetated area resulting from tree mortality. At the end of each time step (5 years), the total portion of an area that had no trees (i.e., the total open area) was calculated. This sum was the total open area potentially subject to regeneration in the next time step of the program. This area was then reapportioned, and the process was reiterated for the desired number of time steps.

In order to test the performance of the PREDIC program, a series of 35 sample points within the lower third of Shenandoah National Park was selected. Locations of sample sites were pinpointed on a 1937 vegetation type map of the Shenandoah Park, a 1973 type map of the Park, and on 7 1/2 min. topographic maps. Using the 1937 map as the initial conditions, and with additional data for the autecological table collected from the topographic maps, predictions of vegetative conditions were made for a 40-year horizon. Predicted conditions for each site were then compared to the 1973 map.

Wildlife Habitat Information Production

The classification of a given point with respect to its suitability as wildlife habitat is more than a simple inventory of physical attributes extracted from a set of axes defining a niche hypervolume. A dynamic classification of an area as wildlife habitat and the production of pertinent management information about the suitability of that site as wildlife habitat depended on the creation of certain artificial attributes known as synthetic attributes.

Formulation of Synthetic Attributes

Formulation of a simple synthetic attribute is not difficult. The task is one of determining the relative magnitude of the synthetic attribute as a function of one or more basic attributes.

The strategy was to develop a series of synthetic attributes directly relating specific attributes to some environmental (natural) attributes. For example, consider the relationship between hard mast production of an area and its benefit to squirrels (Fig. 4). In this case a positive correlation exists between the amount of benefit received by the squirrels and hard mast production. Thus, the new synthetic attribute is an index of benefit, and provides a means of describing the "goodness" of a given area for a

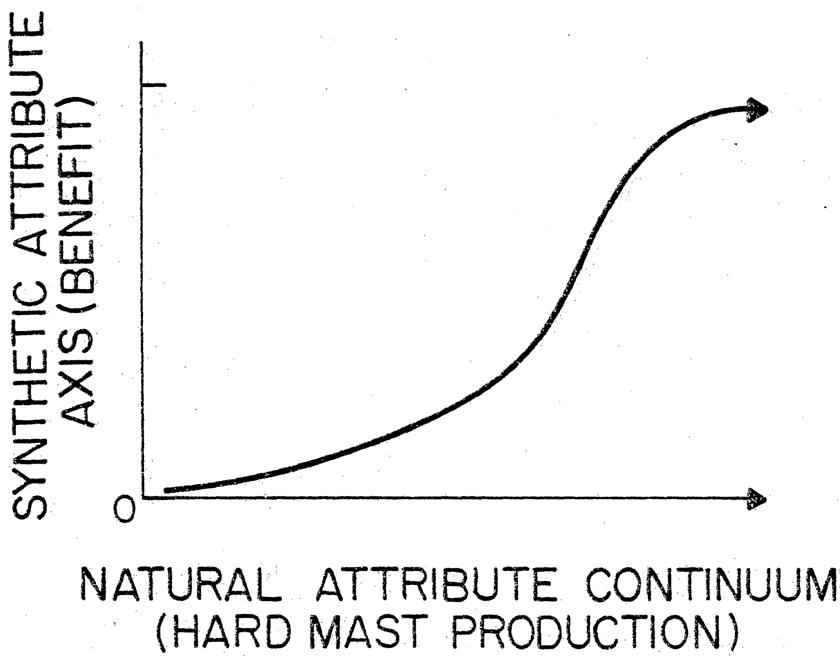


Fig. 4. Functional relationship between a synthetic attribute (benefit) and a natural attribute (hard mast production).

wildlife species (squirrels) as a function of a particular habitat (i.e., natural) attribute.

It is entirely feasible that in some instances, benefit derived by a species from a given attribute is not independent of all other attributes. While at first glance it might appear that interactions could confound derivation of a synthetic benefit attribute, this is not the case. Admittedly, derivation becomes more complicated, but it is nevertheless possible from a theoretical standpoint.

Consider a case involving a 2-way interaction in which the benefit of attribute A is attenuated in the presence of increasing amounts of attribute B. If this relationship is placed in a 3-dimensional context, the interaction between A and B relative to the benefit attribute forms a response surface (Fig. 5).

Interactions involving multiple interactions may be perceived in a similar fashion. However, in a situation with multiple interactions, responses of species are may be described as a many-dimensional hypersurface. The obvious difficulty in this case is determining the shape of that hypersurface. The strategy used in this research was to recognize that a large number of interactions probably exist with respect to the benefit derived from a given attribute. However, only well-established interactions were considered.

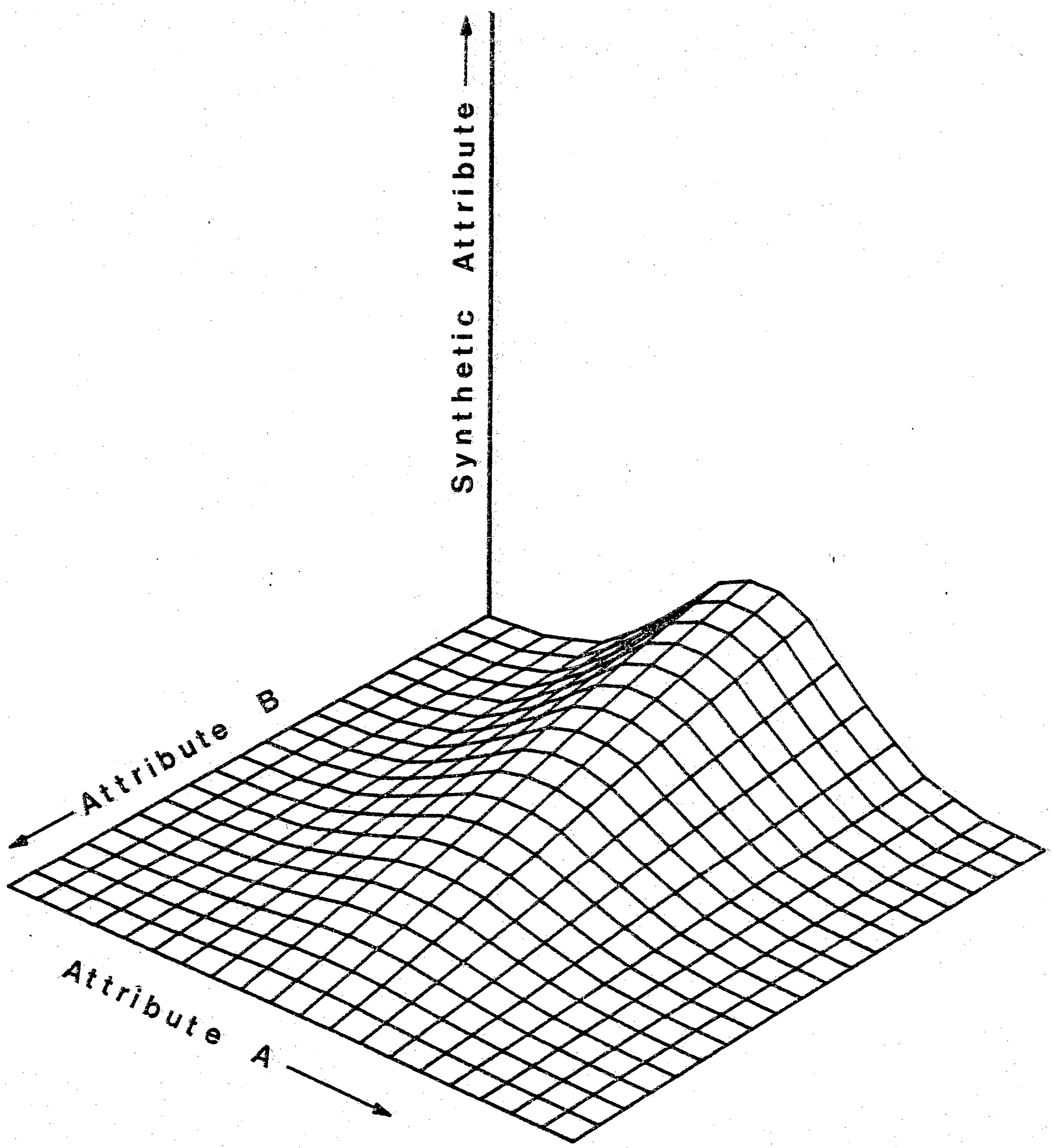


Fig. 5. Response surface resulting from interaction of 2 non-independent attributes.

Where obvious interactions were known, they were recognized and incorporated into the synthetic attributes. In cases where interactions were not well understood, attributes were assumed to exhibit independent relationships with the derived synthetic attributes.

Functional relationships between major habitat factors (attributes) and their respective synthetic benefit attributes were established for wildlife species within their respective study areas. These relationships were determined from direct knowledge of situations within the study areas by personnel on the areas. They were estimated by study-area management personnel based on the best available knowledge, and also obtained from the concensus of biologists within the general region of a study area, or from published information.

Once the functional relationships between habitat factors (i.e., those deemed to be the relevant attributes) were established, wildlife habitat could be described by a set of physical (natural) attributes, or by a set of synthetic attributes. Although both description techniques are useful, synthetic attributes have the potential of providing information more relevant to the objectives of the classification system user.

The technique of creating synthetic attributes is of obvious use in providing specific needed information. However, the true power of this technique lies in its ability to be used in a hierarchical manner. That is, "hybrid" synthetic attributes may be created from first generation synthetic attributes. Synthetic attributes may be used to generate a new synthetic attribute within a single dimension. This procedure was used to produce an index of wildlife habitat suitability. This single attribute was then used to characterize wildlife habitat in a single dimension, i.e., as a single number, for any particular parcel of land. As a unidimensional index, this attribute was amenable to display in map form.

In practice, synthetic attribute data were coded directly for input into the MAP4B system, or were generated from natural attribute data. Thus, synthetic map data were recorded as thematic data, and were input into the MAP4B system along with the appropriate weighting coefficients for each attribute. Within the program, standard matrix addition and scalar multiplication were used to produce a composite matrix representing the synthetic attribute, habitat suitability. Matrices produced in this manner were then displayed as line printer maps.

The raw input data (i.e., the data from U. S. G. S. maps, forest cover maps, etc.) were assumed to be reasonably accurate, and it is assumed that these data were encoded accurately. Data extracted from other data bases (e.g., mast and forage production from forest type and stand age data) were derived in a realistic manner consistent with established procedures (cf. Ehenreich and Crosby 1960, Goodrum et al. 1971, and U. S. D. A. 1971). Similarly, because selection of habitat factors was based on the opinions of experts in the field, it is assumed that the factorial data used were relevant for evaluating wildlife habitat within the respective study areas.

Habitat Suitability Models

Generation of a final habitat index in the work performed was accomplished using 2 models. One algorithm utilized a linear additive model, while the other used a multiplicative model.

The linear additive model

The rationale of the linear additive model was that benefits derived by the species under consideration for the habitat are cumulative, and that some benefits may contribute more to the total welfare of the species than others.

The general mathematical model utilized in the MAP4B system took the form of the equation:

$$V_j = \sum_{i=1}^{n} b_i x_{ij} \tag{1}$$

where V_j = the habitat value of the jth location (cell)

b_i = the relative contribution of the ith synthetic attribute

x_{ij} = standardized value of the ith synthetic attribute at the jth location

$i = 1,2,\ldots,n$

$j = 1,2,\ldots,m$

m = number of locations

n = number of synthetic attributes

In practice, synthetic attribute data were coded directly for input into the MAP4B system, or were derived from natural attribute data. Thus, synthetic data were recorded as map data, and were put into the MAP4B system along with the appropriate weighting coefficients for each attribute. Within the program, standard matrix addition and scalar multiplication were used to produce a composite matrix representing the synthetic attribute, habitat suitability. Matrices produced in this manner were then displayed as line printer maps.

The multiplicative model

A proportional, or volumetric, approach was used in the development of the multiplicative model. Here, the total benefit derived by a species from its habitat was assumed to be the product of the standardized attribute levels. This may be expressed as the following formula:

$$V_j = \prod_{i=1}^{n} x_{ij} \qquad (2)$$

where V_j = the habitat value of the jth location (cell)

x_{ij} = standardized value of the ith synthetic attribute at the jth location

$i = 1,2,\ldots,n$

$j = 1,2,\ldots,m$

m = number of locations

n = number of synthetic attributes

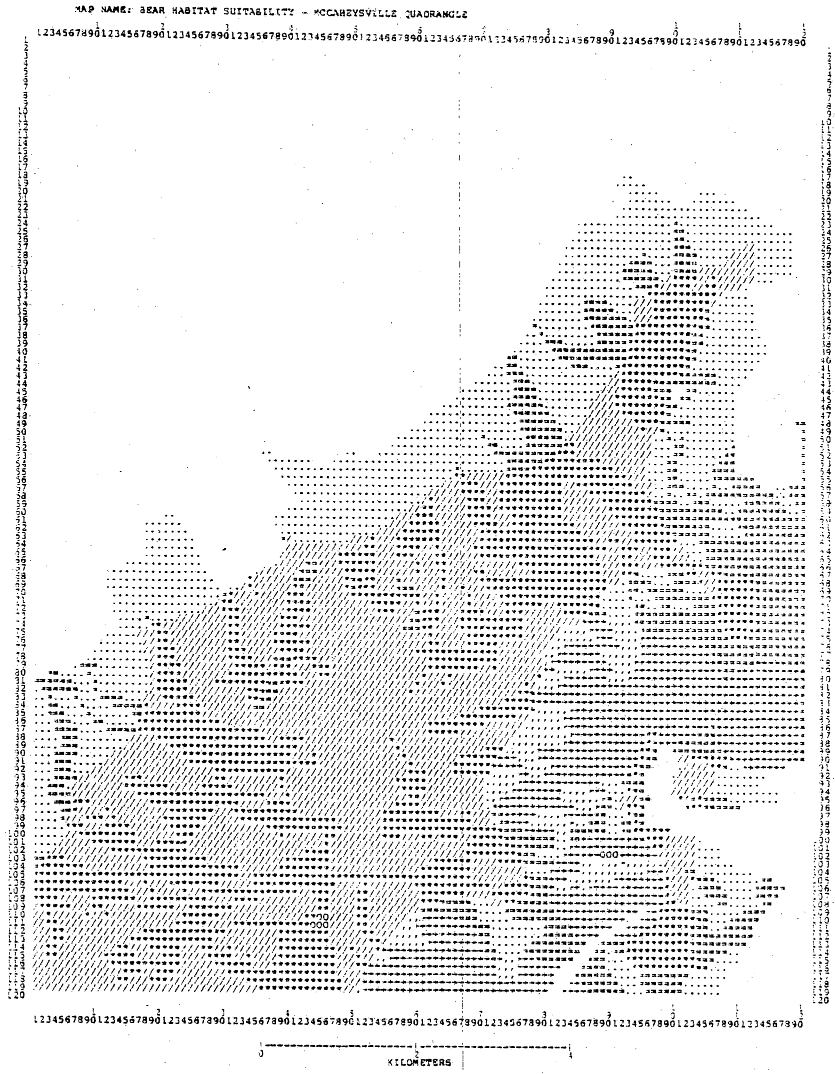
The basic procedure indicated above was to calculate a volumetric index representing the total benefit from the set of synthetic attributes. Since attribute levels were standardized to a maximum value of 1, the maximum total volumetric benefit index was 1.0 to the nth power, or 1.0. Calculated values of less than 1.0 were considered to represent total benefit as a proportion of the maximum potential.

Products

Output from the computerized habitat information system was in the form of 30.5 by 38.1 cm computer line printer maps (Fig. 6). Map sets were produced showing various types of habitat information for each of the 3 study areas. These maps may be categorized as either "current status" maps or "future status" depending on the temporal nature of the information produced. Current status maps displayed habitat information, such as habitat suitability, based on current conditions; future status maps were projections of habitat conditions based on predicted data.

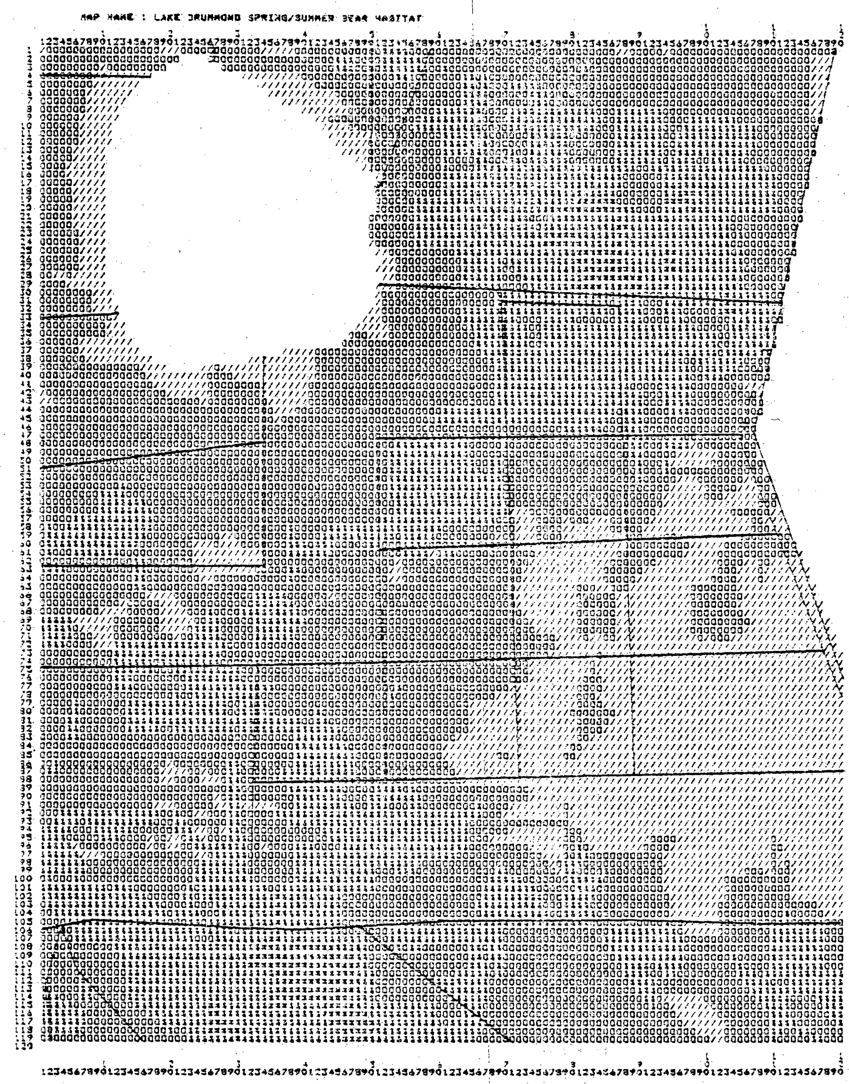
"Current status" maps

Maps depicting the current habitat suitability status for black bears were produced for the Shenandoah National Park and the Dismal Swamp study areas (Figs. 6 and 7). Habitat suitability maps for white-tailed deer, wild turkey, and gray squirrel were produced also for the George Washington National Forest study area. Specific procedures used to derive habitat suitability for these 3 wildlife species were described by Whelan et al. (1979). The process used to assess black bear habitat suitability in the Shenandoah National Park study area has been described in detail by Williamson and Whelan (1980). Similarly, black



SYMBOL	LOWER LIMIT	UPPER LIMIT	COUNT	FREQUENCY
	0.0	16.00	6089.	42.285 %
:::	16.00	32.00	1707.	11.854 %
;;;	32.00	48.00	197.	1.368 %
---	48.00	64.00	0.	0.0 %
===	64.00	80.00	1130.	7.847 %
+++	80.00	96.00	946.	6.569 %
///	96.00	112.00	2338.	16.236 %
000	112.00	128.00	8.	0.056 %
$$$	128.00	144.00	0.	0.0 %
***	144.00	160.00	1985.	13.785 %

Fig. 6. Computer-generated map of current black bear habitat suitability, McGaheysville Quadrangle, Shenandoah National Park, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	255.20	1829.	12.701 %
///	1531.20	1786.40	2457.	17.063 %
000	1786.40	2041.60	6421.	44.590 %
111	2041.60	2296.80	3460.	24.028 %
XXX	2296.80	2552.00	229.	1.393 %

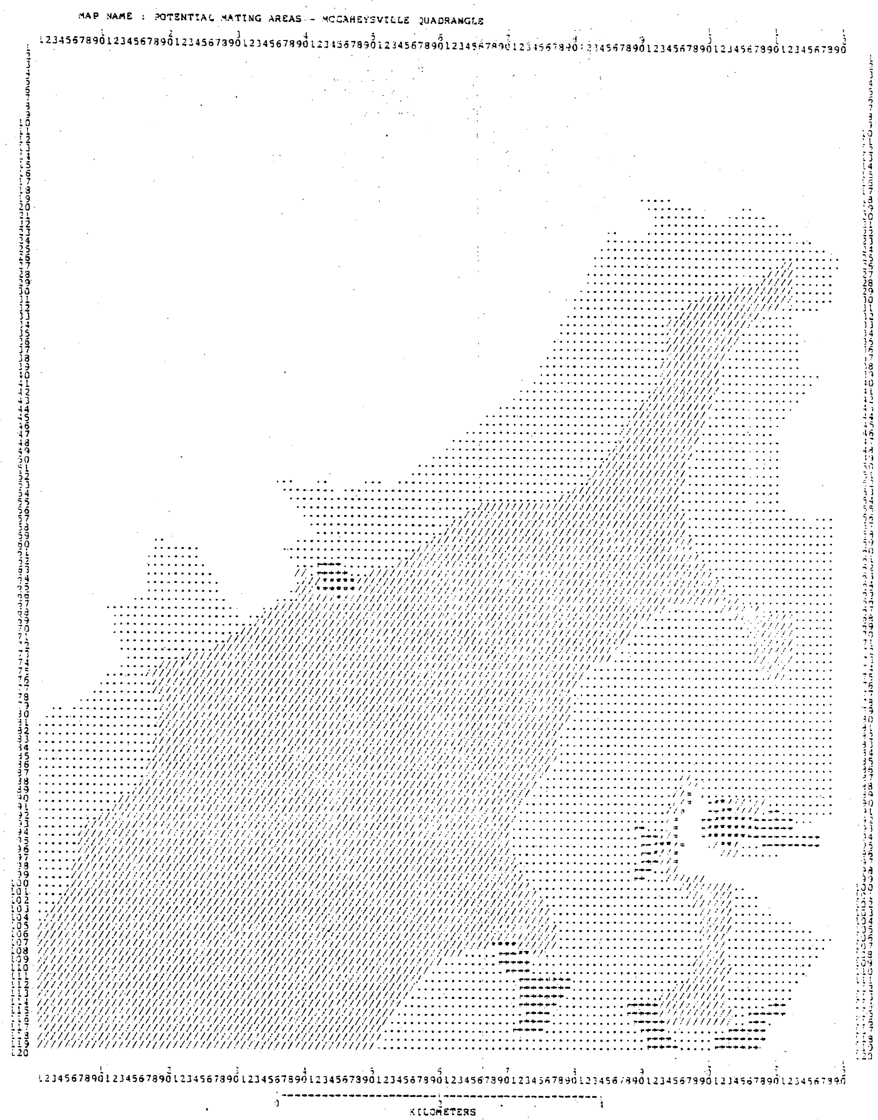
Fig. 7. Computer-generated map of current black bear habitat suitability, Lake Drummond Quadrangle, Great Dismal Swamp, Virginia.

bear assessment procedures used in the Dismal Swamp were discussed by Williamson et al. (1981).

Habitat suitability maps were produced using the compositing operation according to Eq. 1, (i.e., using a linear additive model). Additional maps showing habitat-related information such as potential black bear breeding areas in the Shenandoah Park study area (Fig. 8) and prospective bear denning sites and vegetative interspersion in the Dismal Swamp (Figs. 9 and 10, respectively) were prepared also.

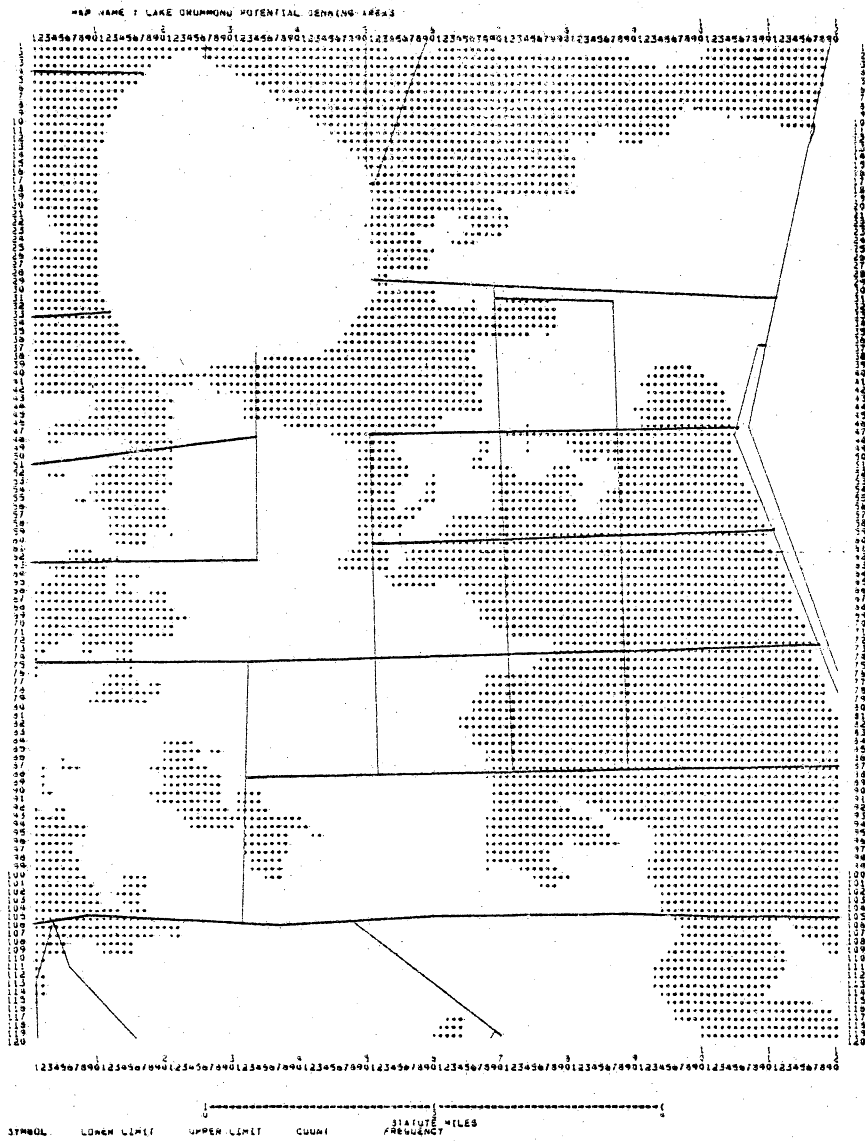
"Future status" maps

The PREDIC program was used to predict future vegetation characteristics on the George Washington National Forest study area. The results of this program were then used to estimate (predict) future habitat suitability for wild turkey using the linear additive model. The algorithm used to calculate future habitat suitability for wild turkey was modified from that of Williamson and Koeln (1980) and Whelan et al. (1979). In the modified algorithm, all factors were assigned equal weights (i.e., "b" values in Eq. 1). Factors and standardized values are given in Table 3. A current status habitat suitability map for wild turkey was prepared using this modified algorithm to permit detection



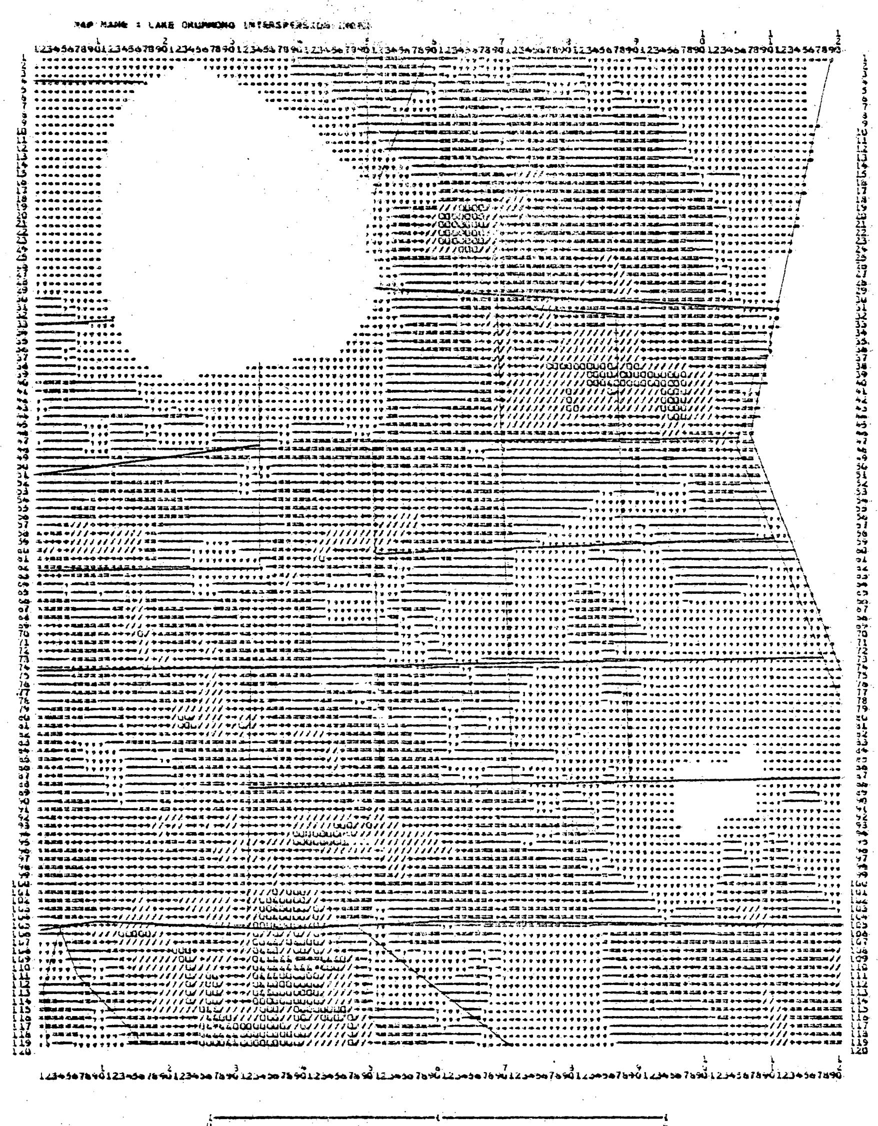
SYMBOL	LOWER LIMIT	UPPER LIMIT	COUNT	FREQUENCY
	0.0	6.00	6024.	41.333 %
:::	6.00	12.00	3800.	26.389 %
===	24.00	30.00	9.	0.063 %
+++	30.00	36.00	183.	1.271 %
///	36.00	42.00	4337.	30.118 %
***	54.00	60.00	47.	0.326 %

Fig. 8. Computer-generated map of potential black bear breeding areas, McGaheysville Quadrangle, Shenandoah National Park, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	1.00	6250.	57.361 %
:::	1.00	2.00	5140.	42.639 %

Fig. 9. Computer-generated map of prospective black bear denning sites, Lake Drummond Quadrangle, Great Dismal Swamp, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.00	1.00	1970.	13.681 %
	1.00	2.00	1317.	9.146 %
	2.00	3.00	2492.	17.628 %
	3.00	4.00	2942.	20.708 %
	4.00	5.00	2635.	18.299 %
	5.00	6.00	1695.	12.160 %
///	6.00	7.00	803.	5.576 %
000	7.00	8.00	292.	2.028 %
	8.00	9.00	48.	0.333 %
	9.00	10.00	6.	0.042 %

Fig. 10. Computer-generated map of vegetative interspersion, Lake Drummond Quadrangle, Great Dismal Swamp, Virginia.

Table 3. Environmental factors and standardized values used to assess wild turkey habitat on the George Washington National Forest study area, Virginia.

Factor	Amount	Value
Mast diversity[1]	>5	9
	5	8
	4	7
	3	5
	2	3
	1	2
	0	0
Water[2]	<0.50	9
	0.50-0.75	7
	0.75-1.50	4
	>1.50	2
Openings[3]	<0.50	6
	>0.50	0
Roosts[4]	<0.50	6
	>0.50	0

[1]Expressed as number of mast-producing trees in the species composition.
[2]Expressed as miles to permanent free-standing water.
[3]Expressed as miles to a site having at least 80% of its area open or in the 0-2 in. dbh class.
[4]Expressed as distance to a site having a pine component of at least 20%.

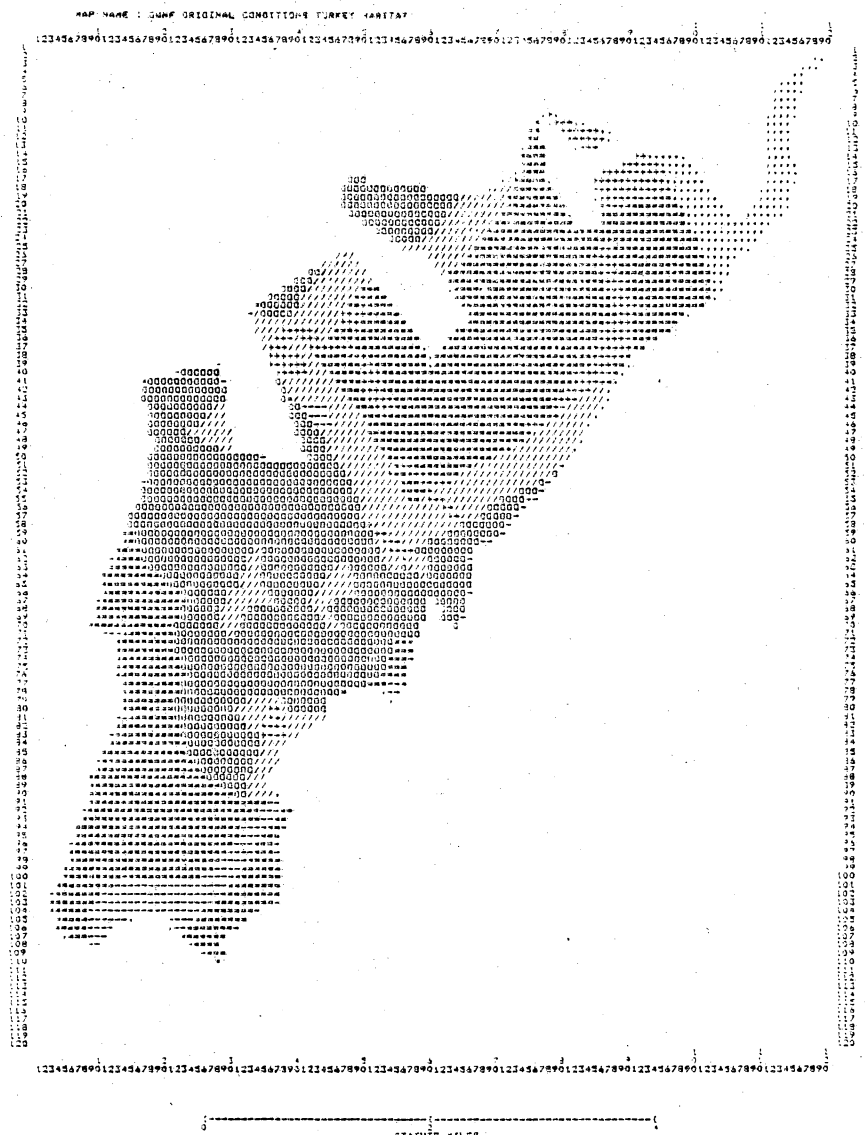
of changing habitat trends over time as compared to original habitat conditions. Habitat suitability predictions were made for periods of 50 and 100 years in the future.

In order to demonstrate the use of such a prediction procedure, habitat suitability forecasts were produced under 2 simulated forest management strategies--a wilderness mode in which no timber harvesting was permitted, and under a timber harvesting regime in which a location (i.e., a cell) was harvested (reset to an open area condition) when 60 percent of the location was occupied by trees in the 8 inch and above dbh (diameter breast high) size class. Habitat suitability forecasts were made for periods 50 and 100 years into the future.

A map of original (current) habitat suitability for wild turkey is shown as Fig. 11. Maps of predicted habitat suitability under the simulated forest management strategies are shown as Figs. 12 ,13, 14, and 15.

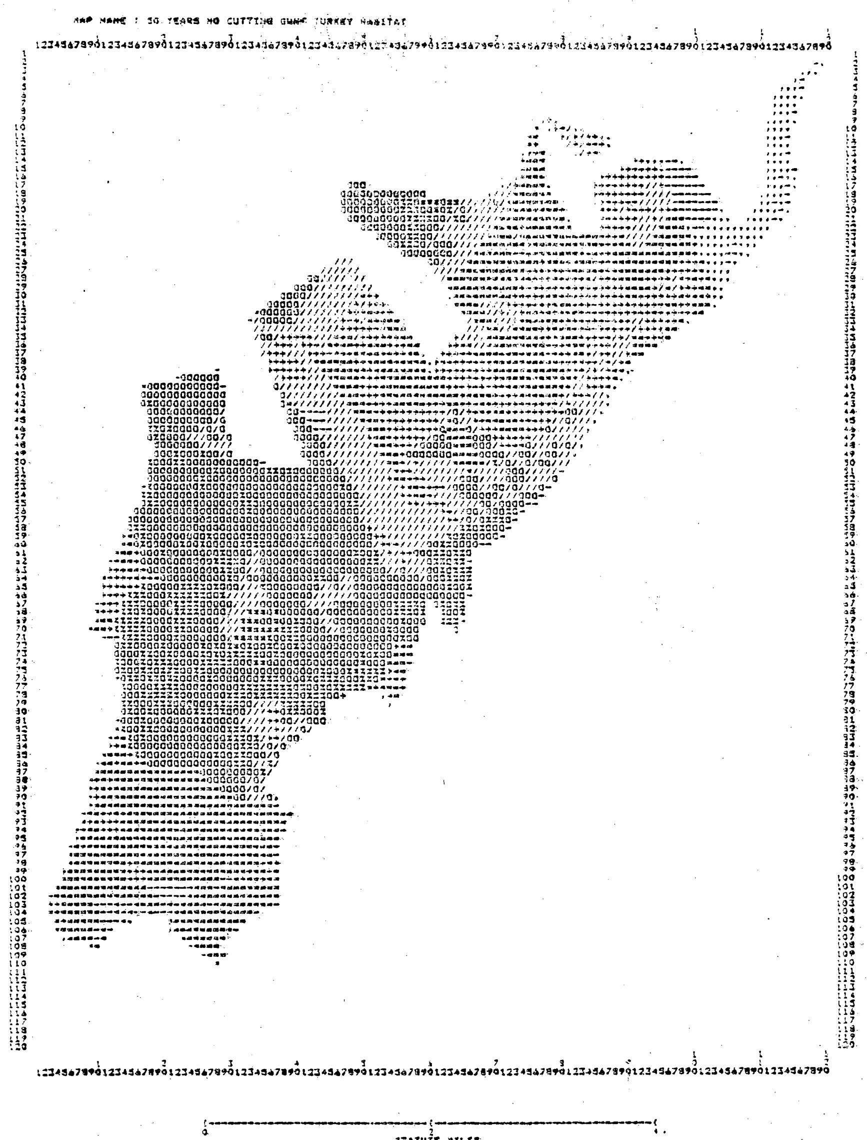
VALIDATION

An objective of this study was to demonstrate the validity of the concept of dynamic classification. To accomplish this task, an epistemological approach was employed.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	14.50	9988.	69.36 %
...	14.50	29.00	33.	0.23 %
,,,	29.00	43.50	197.	1.37 %
---	43.50	58.00	180.	1.25 %
===	58.00	72.50	1780.	12.36 %
+++	72.50	87.00	267.	1.85 %
///	87.00	101.50	601.	4.17 %
000	101.50	116.00	1354.	9.40 %

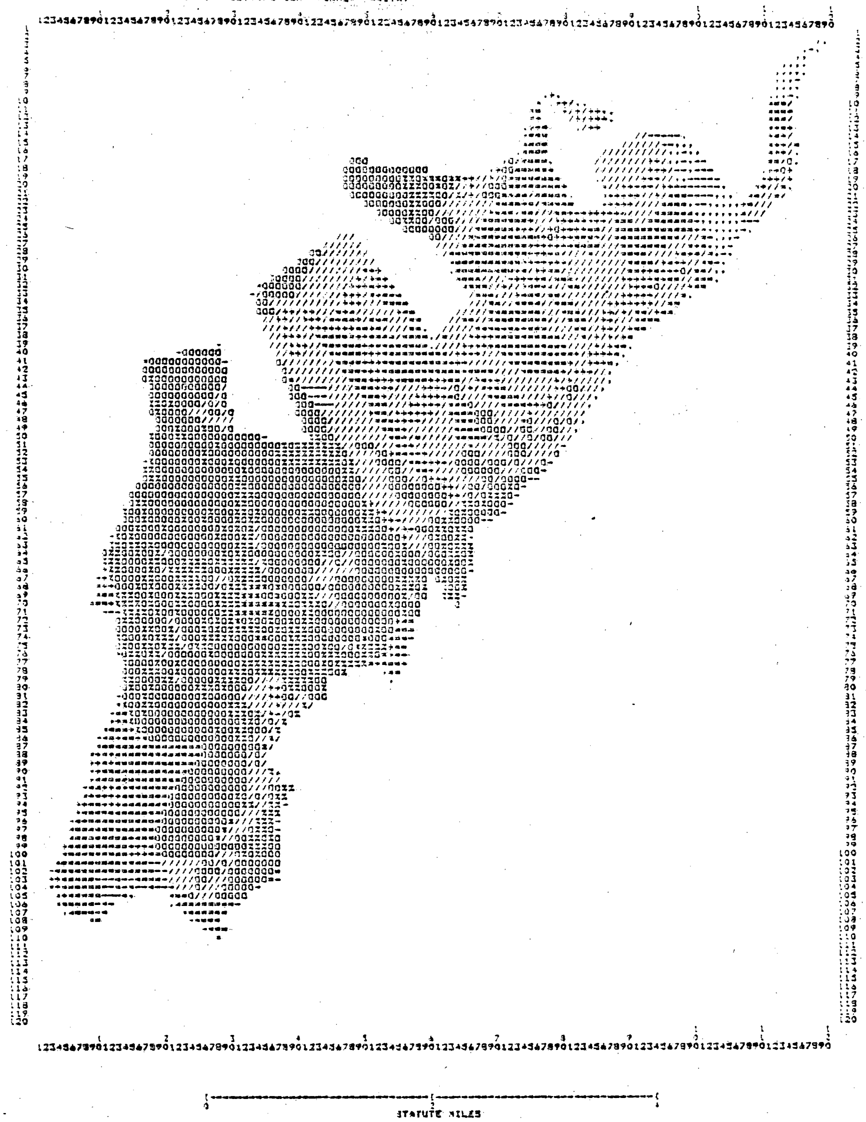
Fig. 11. Computer-generated map of current wild turkey habitat suitability, George Washington National Forest study area, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	14.50	9958.	69.36 %
...	14.50	29.00	33.	0.23 %
,,,	29.00	43.50	137.	0.95 %
---	43.50	58.00	148.	1.03 %
===	58.00	72.50	1134.	7.88 %
+++	72.50	87.00	665.	4.62 %
///	87.00	101.50	617.	4.29 %
000	101.50	116.00	1322.	9.18 %
ZZZ	116.00	130.50	326.	2.26 %
XXX	130.50	145.00	30.	0.21 %

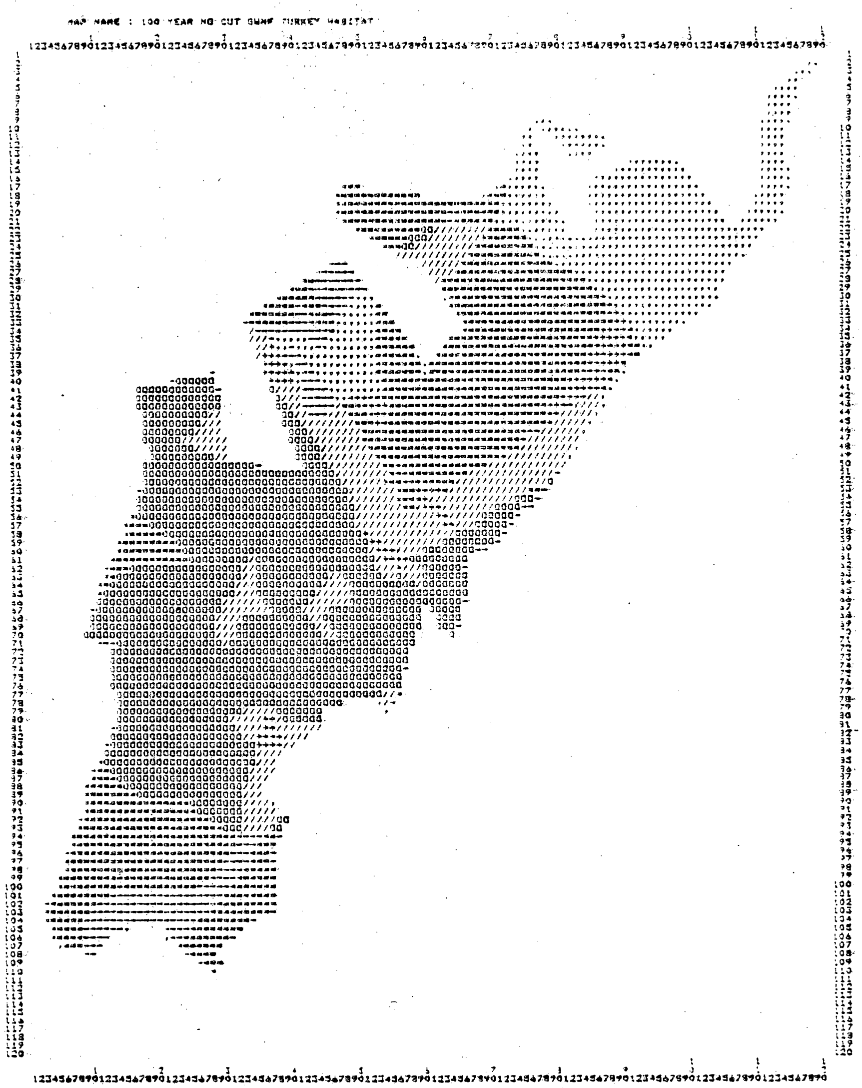
Fig. 12. Computer-generated map of predicted wild turkey habitat suitability at 50 years in the future under a simulated wilderness type forest management strategy, George Washington National Forest study area, Virginia.

MAP NAME : 50 YRS WITH CUTTING GWNF TURKEY HABITAT



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	14.50	9950.	69.34 %
...	14.50	29.00	34.	0.24 %
,,,	29.00	43.50	88.	0.61 %
---	43.50	58.00	120.	0.83 %
===	58.00	72.50	380.	3.11 %
+++	72.50	87.00	411.	2.35 %
///	87.00	101.50	902.	5.26 %
000	101.50	116.00	1489.	10.34 %
ZZZ	116.00	130.50	493.	3.13 %
###	130.50	145.00	35.	0.24 %

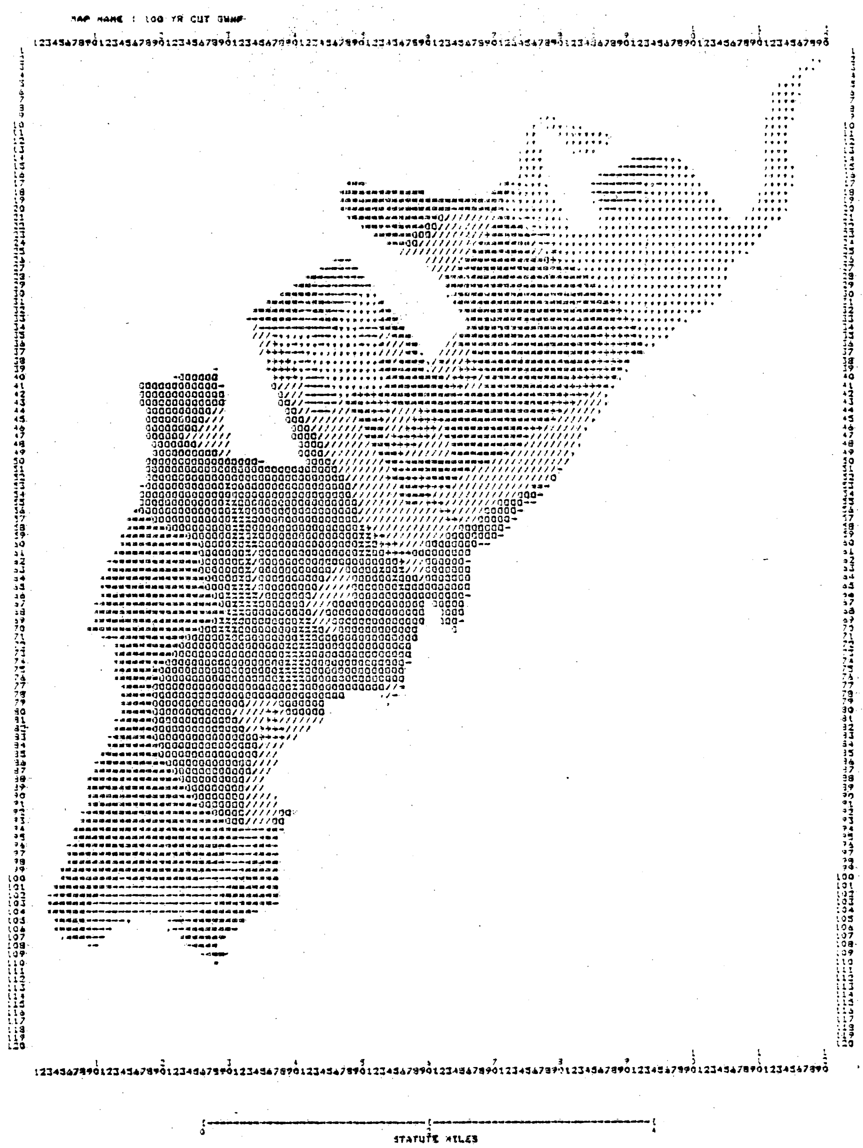
Fig. 13. Computer-generated map of predicted wild turkey habitat suitability at 50 years in the future under a simulated timber harvesting regime, George Washington National Forest study area, Virginia.



0 ——— 3 ——— 6
STATUTE MILES

SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	14.50	7988.	49.36 %
...	14.50	29.00	59.	0.41 %
,,,	29.00	43.50	633.	4.35 %
---	43.50	58.00	252.	1.75 %
===	58.00	72.50	1282.	3.90 %
+++	72.50	87.00	133.	0.82 %
///	87.00	101.50	499.	3.47 %
000	101.50	116.00	1533.	10.44 %

Fig. 14. Computer-generated map of predicted wild turkey habitat suitability at 100 years in the future under a simulated wilderness type forest management strategy, George Washington National Forest study area, Virginia.



SYMBOL	LOWER LIMIT (.GE.)	UPPER LIMIT (.LT.)	COUNT	FREQUENCY
	0.0	14.50	9988.	69.36 %
:::	14.50	29.00	34.	0.24 %
;;;	29.00	43.50	616.	4.28 %
---	43.50	58.00	311.	2.16 %
===	58.00	72.50	1458.	10.13 %
+++	72.50	87.00	171.	1.19 %
///	87.00	101.50	533.	3.70 %
000	101.50	116.00	1211.	8.41 %
ZZZ	116.00	130.50	78.	0.54 %

Fig. 15. Computer-generated map of predicted wild turkey habitat suitability at 100 years in the future under a simulated timber harvesting regime, George Washington National Forest study area, Virginia.

Epistemology is the study of the theory of knowledge (Wartofsky 1968:12-13). In epistemology, questions about the origins of knowledge, how one acquires knowledge, and the forms of valid argument are addressed and analyzed. While the former topics have been the subject of much philosophical controversy, and will not be discussed further, the latter is central to the methodology used in this research.

Various epistemological approaches and bases exist for substantiating a given argument. The process of validating or substantiating may be sensory (experiential), probabilistic (statistical), pragmatic (workable, tenable), authoritative, inductive, or deductive in nature. These are a few of the major epistemological bases in which the validity of an argument or proposition may be rooted. Nevertheless, it is through such bases that people come to accept, to believe, and eventually, to gain knowledge.

Outside the science of mathematics, establishing a definitive and undeniable proof of a proposition is difficult, if not impossible. No attempt has been made to prove the concept of dynamic classification. Rather, an attempt was made to establish the validity of this concept using a variety of epistemological arguments. It will be argued that the fundamental concept of dynamic classification may be substantiated to a degree on each of several epistemological bases. While a single epistemological base may provide sufficient validation, several were used to support the concept, and therefore, better assure its truthfulness.

Inductive Validation

The primary epistemological base used in this study was induction. For obvious reasons, the concept of dynamic classification could not be analyzed under all possible conditions to which it might be applicable. Thus, an application of dynamic classification was used in an experimental situation in order to provide a basis for an inductive substantiation.

Pragmatic Validation

The argument that dynamic classification possesses validity on the epistemological base of pragmatism was investigated. However, the counter argument was used to study the validity of the concept. That argument was that conventional habitat classification systems tend to fail both at small scales and at large, regional scales. At the small scale, groups may be too large, and sufficient information is not produced at the resolution desired. At larger scales, groups may be too small, resulting in a condition of information overload. In this case, results such as maps are said to be "noisy," a term having significance in information science.

Other counter arguments considered were that present habitat information systems cannot accommodate readily outliers (i.e., those situations having conditions unlike any encountered previously); that their results are inherently dated and are therefore of limited use; that some results, especially maps, are not amenable to editing or updating; and that these systems may be use-specific. That is, they may be valid for one use, and one use only.

Contextual Validation

The epistemological question of the consistency of the concept of dynamic classification with current use of the language of science was addressed. Although a contextual foundation is but one base for knowing or for determining the truthfulness of a concept, it is nearly an essential base. The choice of theoretical language was deliberate, and efforts have been made to assure its consistency, as far as possible, with current ecological theory.

Authoritative Validation

Validity may be rooted in authority. That is, credibility, and subsequently, validity may be given to an argument or proposition pending its acceptance by one or more persons regarded as authorities in the field. While it is much too soon for such validation to be feasible here, initial steps have been taken in subjecting the concept of dynamic classification to review by peers, faculty, and prospective users of such techniques.

Deductive Validation

Deduction is a powerful epistemological base. Thus, the matter of the correspondence between dynamic classification and general systems theory; the literature of mathematical classification; statistical decision theory; certain aspects of cluster analysis, factor analysis, and discriminant analysis; and the niche hypervolume concept were studied in an effort to establish a deductive epistemological base.

SUMMARY

The overall methodology for the development of a concept of dynamic classification was (1) to justify the need for such an approach, (2) to articulate the concept itself, (3) to demonstrate the utility of such an approach in a practical application, and (4) and to define the epistemological bases for theoretical validity. These bases were induction, pragmatism, contextual, authority, and deduction. The results of this development are presented and discussed in the following chapter.

Chapter V

RESULTS

INTRODUCTION

A general concept of dynamic classification was developed. Restated briefly, this concept allows for the production of specified information through the precise positioning of the entity to be classified within a hypervolume defined by one or more axes constructed from respective attribute continua. A concerted effort was made to demonstrate the utility of a dynamic classification approach in a wildlife management application and to substantiate and validate the basic concept of dynamic classification using five epistemological approaches.

UTILITY

A primary concern during this research effort was that the physical systems and concepts developed should have practical utility. It was the intent in this effort to have products applicable in a real-world environment.

Due to the infeasibility of determining the utility of research products from extensive user feedback, a determination of their feasibility and utility was undertaken indirectly, through a critical analysis of the assets and liabilities of each research product.

Software

The software developed in this research consisted of the MAP4B mapping system (including auxiliary programs) and the PREDIC program. The MAP4B system was designed to be an independent map generation and display system. The PREDIC program was created specifically to function as a complementary program to the MAP4B system. Although the PREDIC program may be used independently, it was designed to be used in tandem with the MAP4B system to provide projected habitat data as a function of changes in vegetative conditions.

The MAP4B System

An experimental test of the MAP4B system (independent of the PREDIC program) was performed on the Dismal Swamp study area. Although the purpose of the Dismal Swamp study was to produce basic black bear habitat assessment information pertinent to the formulation of a bear management program in the Great Dismal Swamp National Wildlife Refuge (Williamson et al. 1981), this study provided an excellent opportunity to discover any flaws or inadequacies of the mapping system.

Data collection, typically the most laborious and time consuming part of using a computer geographic data system, was extensive in the Dismal Swamp study. However, the availability of digitizing equipment facilitated data capture and encoding. Time required for encoding a block of data (i.e., a map corresponding to the area within a 7 1/2 min. USGS quadrangle sheet) required from less than one-half hour to approximately 4 hours, depending on the complexity of the map. However, no major problems were encountered in the encoding process.

A minor inadequacy of the MAP4B system was the inability to display output as a continuum. Due to the limited amount of visual contrast attainable with standard keyboard characters, display results were limited to 10 character symbols. Such stratification of habitat suitability indices (which were explicitly continuous in nature) resulted in the creation of groupings of similar-valued cells displayed by the same symbol. These groupings are analogous (but not identical) to mapped classes from a conventional classification. In the maps produced, any apparent "classes" (i.e., groupings of similar cells) were strictly an artifact of the display process and should not be interpreted as actually comprising a homogeneous unit.

The linear additive model was used to produce all composite maps in the Dismal Swamp study. No problems were encountered with the MAP4B program or the auxiliary programs, CONVERT and SEARCH, which were used in this study area. Total time required to produce 3 sets of 11 composite habitat suitability maps and to display 2 sets of 11 maps each, showing vegetative interspersion and prospective bear denning sites, required approximately 10 person-hours.

Map products were produced on a high speed computer line printer and on a DECWRITER III (Digital Equipment Corp., Maynard, Mass.) hardcopy terminal. Both devices produced suitable results; however, the DECWRITER produced output having a slightly higher resolution than the line printer. Due to the small cell size, spatial resolution of this output was judged satisfactory by the sponsoring agency and was well within its requirements.

Both the display and composite options (cf. Appendix A for a complete desription of these operations) were utilized in the Dismal Swamp test and in producing maps of the other study areas. Problems were not encountered in using either output option.

The linear additive model was used to produce habitat suitability maps and any other habitat-related maps requiring a compositing operation. No problems were encountered using the additive model. However, problems were encountered in using the multiplicative model. Despite the fact that this model possessed some intuitive theoretical appeal, it was discovered that the inclusion of multiple factors tended to result in most cells having extremely low values. Quite often, the vast majority of the cells in a map would be in the lowest display category (a blank), and an empty map would result. Thus, from a practical standpoint, this particular mathematical model did not appear feasible for producing composite maps with the mapping system developed in this research.

The PREDIC Program

A test of the general utility of the PREDIC program was performed on the George Washington National Forest study area. Forest stand data provided by the U. S. Forest Service were used to produce estimated stand conditions at periods of 50 and 100 years in the future. These projected stand data were then used to estimate future habitat suitability for wild turkey.

Some problems were encountered in obtaining adequate input data for the PREDIC program. A particular problem area was the securing of suitable elevation data from which slope and aspect could be calculated using the auxiliary program SLOPE (Appendix B). Elevation data were extracted from the Digital Terrain Tapes (White 1980) available from the U. S. Geological Survey. From a practical standpoint, it must be admitted that the inability to acquire suitable elevation data in a cost-efficient manner could prove to be a serious impediment to potential users desiring to generate habitat suitability predictions.

Concepts

A definitive determination of the practical utility of the concepts developed in this research could not be made. However, it is believed that the basic concepts postulated here have the potential to provide wildlife managers with a useful tool, even if that tool is a concept.

A dynamic classification approach has the potential of providing information superior to that of other information systems that deal with classification. However, whether or not such an approach will be used, is difficult to establish. Admittedly, potential users such as natural resource managers tend to be conservative. A dynamic classification approach is a radical departure from conventional methodology, and as such may not be accepted readily by conservative users. It is obvious that new methodology for natural resources management is needed; however, the acceptance of such new methods must remain a moot topic.

VALIDITY

Inductive Validity

The information output of the computer system developed in this research reflected reasonably accurate representations of the wildlife habitat suitability in the areas mapped. Validation of habitat suitability based on concomitant population levels was not undertaken, due primarily to the lack of reliable long-term population data for any of the study areas. However, Dismal Swamp Refuge workers have reported a higher-than-average number of bear sightings in the vicinity of Williamson ditch (Ralph Keel, pers. commun.), an area determined to be of superior suitability, lends some credence to the Dismal Swamp habitat maps.

Nevertheless, it must be mentioned that high population levels may not be correlated consistently with high habitat suitability as determined with the procedures used here. Extraneous factors or mechanisms not normally considered to be a part of the natural environment within an area, such as a highly contagious disease, may effectively eradicate animal populations existing in areas having otherwise suitable habitat. Long-term cyclic population fluctuations, such as those experienced by some wildlife species, may confound population estimation attempts. Also, common techniques of population sampling, such as trapping, are not permissible for rare species, nor appropriate for those apparently exhibiting non-uniform dispersal patterns (e.g., bears). Thus, measurement of population levels (especially on a short-term basis) may not be a consistently reliable method for validating estimates of habitat suitability, and it is for this reason that this method was not utilized in this study.

The vegetation prediction program (PREDIC) utilized established, documented information on the relationships between the occurrence of vegetation and edaphic factors, and rates of radial growth. However, empirical validation was accomplished by comparing results of a vegetation prediction (based on 1937 data) for the year 1973 to a forest type map produced that same year. Comparison of predicted vegetation to the type map (see Table 4) revealed that predictions were accurate for all test sites except 2 sites that were previously open land, and 2 sites that were classified as bear oak-pitch pine on the 1937 map. According to the prediction, one of the open area sites (sample stand 86) should have become a mixture of approximately equal proportions of chestnut oak, black locust, yellow poplar, white oak, and gray birch. This site

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park.

Sample Stand Number	Initial Conditions: Stand Type Code[1]	Initial Conditions: Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Predicted Conditions: Species Composition (%)	Predicted Conditions: Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
1	Co	1.0	1.0	13.0	85.0	83.0 Co	2.8	4.1	8.5	67.6	Co
						4.9 Wo	2.1	2.5	0.3	0.0	
						4.9 Wp	2.1	2.4	0.4	0.0	
						7.1 Op					
2	Co	1.0	1.0	13.0	85.0	81.5 Co	2.2	3.3	8.4	67.6	Co
						3.7 Wo	1.6	1.9	0.2	0.0	
						3.7 BlO	1.7	1.7	0.2	0.0	
						3.7 BrO	2.0	1.5	0.2	0.0	
						7.3 Op					
3	Co	85.0	15.0	0.0	0.0	99.3 Co	3.7	44.5	41.7	9.4	Co
						0.6 Op					
4	Co	1.0	1.0	13.0	85.0	83.1 Co	2.8	4.1	8.6	67.6	Co
						4.9 Wo	2.1	2.5	0.3	0.0	
						7.1 Op					
5	Co	85.0	15.0	0.0	0.0	99.3 Co	3.7	44.5	41.7	9.4	Co
						0.7 Op					
6	So	15.0	85.0	0.0	0.0	98.4 So	0.9	31.7	46.9	18.9	So
						1.5 Op					

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

Sample Stand Number	Initial Conditions: Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Predicted Conditions: Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
7	So	85.0	15.0	0.0	0.0	99.3 So	4.9	42.5	40.1	11.7	So
						0.7 Op					
8	So	85.0	15.0	0.0	0.0	99.3 So	4.9	42.5	40.1	11.7	So
						0.7 Op					
9	So	1.0	1.0	13.0	85.0	78.6 So	0.1	0.7	7.0	70.7	So
						4.8 Co	2.6	1.9	0.3	0.0	
						4.8 Ro	2.7	1.8	0.3	0.0	
						3.6 Wo	2.0	1.4	0.2	0.0	
						3.6 Wp	2.0	1.4	0.3	0.0	
						4.6 Op					
21	Ro	1.0	1.0	13.0	85.0	85.8 Ro	1.1	1.7	7.1	75.9	Ro
						2.1 Co	0.9	1.0	0.1	0.0	
						1.5 Wo	0.7	0.8	0.1	0.0	
						1.5 Wp	0.7	0.7	0.1	0.0	
						0.9 Op					
22	Ro	85.0	15.0	0.0	0.0	99.5 Ro	4.9	42.5	40.1	11.9	Ro
						0.5 Op					

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

Sample Stand Number	Initial Conditions: Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Predicted Conditions: Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
23	Ro	1.0	1.0	13.0	85.0	83.7 Ro	0.1	0.7	7.0	75.9	Ro
						2.8 Co	1.2	1.4	0.1	0.0	
						2.1 Wo	0.9	1.0	0.1	0.0	
						9.3 Op					
24	Ro	85.0	15.0	0.0	0.0	99.5 Ro	4.9	42.5	40.1	11.9	Ro
						0.5 Op					
25	Ro	1.0	1.0	13.0	85.0	83.8 Ro	0.1	0.7	7.0	75.9	Ro
						2.8 Co	1.3	1.4	0.1	0.0	
						2.1 Wo	0.9	1.0	0.1	0.0	
						2.1 Op	0.9	1.0	0.1	0.0	
26	BrO	85.0	15.0	0.0	0.0	99.4 BrO	8.5	45.3	37.1	8.5	BrO
						0.1 Co	T	T	0.0	0.0	
						0.5 Op					
27	BrO	85.0	15.0	0.0	0.0	99.4 BrO	8.5	45.3	37.1	8.5	BrO
						0.6 Op					

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

	Initial Conditions					Predicted Conditions					
Sample Stand Number	Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
28	BrO	85.0	15.0	0.0	0.0	99.4 BrO 0.1 Co 0.5 Op	8.5 T	45.3 T	37.1 0.0	8.5 0.0	BrO
29	BrO	85.0	15.0	0.0	0.0	99.4 BrO 0.6 Op	8.5	45.3	37.1	8.5	BrO
30	BrO	85.0	15.0	0.0	0.0	99.4 BrO 0.6 Op	8.5	45.3	37.1	8.5	BrO
31	Pp	20.0	75.0	5.0	0.0	95.8 PP 4.2 Op	0.0	10.7	54.8	30.2	Pp
32	Pp	5.0	5.0	90.0	0.0	90.3 Pp 9.7 Op	0.0	1.2	44.4	44.7	Pp
33	Pp	20.0	75.0	5.0	0.0	95.8 Pp 4.2 Op	0.0	10.7	54.8	30.2	Pp

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

Sample Stand Number	Initial Conditions: Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Predicted Conditions: Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
34	Pp	1.0	4.0	50.0	45.0	79.5 Pp	0.0	0.6	24.3	54.6	Pp
						6.5 Co	4.1	2.2	0.1	0.0	
						4.8 Wo	3.0	1.7	0.1	0.0	
						4.8 Wp	3.0	1.7	0.1	0.0	
						4.4 Op					
35	Pp	1.0	4.0	50.0	45.0	79.5 Pp	0.0	0.6	24.3	54.6	Pp
						6.5 Co	4.1	2.2	0.1	0.0	
						4.8 BlO	3.2	1.6	0.1	0.0	
						4.8 BrO	3.4	1.3	0.1	0.0	
46	Wp	1.0	9.0	45.0	45.0	91.3 Wp	t[6]	2.5	25.8	62.9	Wp
						8.7 Op					
47	Wp	1.0	9.0	45.0	45.0	91.3 Wp	t	2.5	25.8	62.9	Wp
						8.7 Op					
48	Wp	1.0	9.0	45.0	45.0	91.3 Wp	t	2.5	25.8	62.9	Wp
						8.7 Op					
49	Wp	90.0	10.0	0.0	0.0	99.5 Wp	3.9	37.5	44.3	13.9	Wp
						0.5 Op					

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

	Initial Conditions					Predicted Conditions					
Sample Stand Number	Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
50	Wp	1.0	9.0	45.0	45.0	91.3 Wp	0.0	2.5	25.8	62.9	Wp
						8.7 Op					
56	Bl	95.0	5.0	0.0	0.0	99.1 Bl	6.6	43.0	42.1	7.4	Bl
						0.9 Op					
86	Op					19.4 Co	1.6	10.4	6.3	0.9	Bl
						16.4 Bl	1.7	8.0	5.8	0.7	
						16.3 Yp	0.0	6.7	6.5	3.1	
						16.2 Vp	0.0	3.3	9.6	3.2	
						14.6 Wo	1.2	7.8	4.7	0.7	
						14.2 Gb	0.2	2.7	6.6	4.7	
						2.9 Op					
87	Op					18.9 Ro	1.9	9.6	6.1	1.2	Bl
						18.9 Co	1.5	10.1	6.3	1.0	
						16.8 Bl	1.8	8.3	6.0	0.8	
						16.8 Yp	0.0	6.9	6.7	3.1	
						14.1 Wo	1.2	7.5	4.7	0.7	
						14.2 Wp	1.2	6.6	5.3	1.1	
						0.4 Op					

Table 4. Initial stand conditions (based on 1937 data), PREDIC predictions of species compostion and age distribution, and observed stand type (from 1977 data) for 35 sample stands in Shenandoah National Park (continued).

	Initial Conditions					Predicted Conditions					
Sample Stand Number	Stand Type Code[1]	Estimated Age Distribution (%) A[2]	B[3]	C[4]	D[5]	Species Composition (%)	Predicted age Distribution (%) A[2]	B[3]	C[4]	D[5]	Observed Stand Type
26A	BrO/Pp	85.0	15.0	0.0	0.0	49.7 Br	4.3	22.7	18.5	4.2	Pp
						48.5 Pp	0.0	6.9	28.5	13.1	
						1.8 Op					
27A	BrO/Pp	85.0	15.0	0.0	0.0	49.7 Br	4.3	22.7	18.5	4.2	Pp
						48.5 Pp	0.0	6.9	28.5	13.1	
						1.8 Op					

[1]Stand type codes are as follows: Co = Chestnut oak, So = Scarlet oak, Ro = Northern red oak, BrO = Bear oak, Wo = White oak, BlO = Black oak, Bl = black locust, Yp = Yellop poplar, Gb = Gray birch, Pp = Pitch pine, Vp = Virginia pine, Wp = White pine, Op = open.
[2]0-2 inch dbh class
[3]2-5 inch dbh class
[4]5-8 inch dbh class
[5]>8 inch dbh class
[6] Trace

was classified as black locust on the 1973 map. Sample stand number 87, also an open area in 1937 was classified as a black locust stand on the 1973 map. The predicted species was a fairly even mixture of red oak, chestnut oak, black locust, yellop poplar, white oak, and white pine in that order. Discrepencies between observed and predicted conditions were noted with sample stands 26A and 27A. These 2 sites were classified as bear oak-pitch pine on the 1937 map. According to predictions, these sites should have remained bear oak-pitch pine type. However, on the 1973 map, these 2 sites (and all other stands classified as bear oak-pitch pine in 1937) had been classified consistently as pitch pine. It is suspected that although these stands were labeled pitch pine, they may have had a substantial bear oak component since these two species often coexist (Society of American Foresters 1954).

Although the PREDIC system apparently predicted vegetation types with a reasonable degree of accuracy, it utilized a general algorithm which may be refined and expanded as additional knowledge becomes available. The algorithm is also sufficiently general to allow modification for use in different locations or in different edaphic environments.

Nevertheless, methodological validity alone is not sufficient to establish theoretical validity. Additional epistemological evidence is therefore offered to substantiate further the general concept of dynamic classification.

Pragmatic Validity

The concept of dynamic classification also possesses validity from a pragmatic standpoint. Using the concept, apparently valid information pertinent to the needs of wildlife management was produced. It is theoretically possible to obtain whatever information is needed using a dynamic classification approach. The practicality of actually obtaining this information is a function of (1) the ability of users to articulate accurately their information needs, and (2) the ability of the user to define accurately the coordinates of the object along all the necessary attribute axes. The inability of the user to perform either, or both, of these prerequisites may restrict or constrain the utility of the methodology available to the user, but it does not invalidate the fundamental tenets of the overall concept.

It should be noted that similar problems plague conventional classification efforts. However, no methods for coping with such problems exist within the theory of conventional classification. Such problems, coupled with the inherently static and inflexible nature of conventional classification, tend to cause classical methodology to be suboptimal, impractical, inappropriate, or infeasible in many situations.

Contextual Validity

The concept of dynamic classification is consistent with established ecological theory. Describing a site in terms of its position within a niche hypervolume is essentially a process of dynamic classification. However, using dynamic classification, this information may be "collapsed" or reduced into a single measure through the construction of high-information-content synthetic attributes.

Dynamic classification is consistent also with conventional classification theory. Conventional classification is believed to be a special case of dynamic classification.

Authoritative Validity

Currently, direct validation of a dynamic classification approach or its associated methodology as used in this research, from an authoritative standpoint has been limited. Nevertheless, it is believed that since the methodology is based on established ecological relationships and principles, the overall methodology is credible. In time, it is believed that this credibility will increase among authorities.

Deductive Validity

Dynamic classification may be seen as an information-producing system. Inputs into this system are the things to be classified and the objectives specified by the user. The process component is composed of the series of specific procedures used to derive the desired information (i.e., the output).

Noteworthy is the fact that each of these system components is subject to change over time. The inherent nature of objects may change, as discussed previously, as may the nature of the classification objectives. Changes in the procedures used in a classification are necessitated by changes in objectives, or in the type and resolution of the information desired. System output may change from specific information to information of a general nature.

The ideal classification system can be visualized as a 3 dimensional structure with inputs, outputs, and processes comprising the axes. In this general system, the state of a classification system with respect to each of these axes is dynamic; the system must be operable and functional anywhere within the volume defined by these axes. By design, dynamic classification accommodates this dynamic nature of the ideal classification system. Inasmuch as the concept of dynamic classification is consistent with the performance expected of a general classification system, it possesses utility, credibility, and validity as it was used in this research.

Chapter VI

CONCLUSIONS

METHODOLOGY

The MAP4B program, due to its simplicity and ease of use, was considered to be an efficient and effective means of utilizing dynamic classification methodology. Computer time required to execute the system was small, and no special equipment or programming was required to implement the system. Maps produced by the MAP4B system have been deemed suitable and satisfactory in the limited amount of use to which it has been put to date.

Nevertheless, the intent is not to debate the merits of the MAP4B program. Other commercially available programs may have served equally as well as (or better than) the MAP4B program. The MAP4B program was a sufficient and an appropriate vehicle for displaying wildlife habitat information generated by a dynamic classification process.

Limited testing of the validity and accuracy of the vegetation prediction program (PREDIC) showed the algorithm to be reasonably accurate within the time frame used. However, additional testing using more extensive data and longer prediction horizons would be desirable in order to improve the algorithm.

Modification to the PREDIC program to incorporate more factors into the autecological table (see Table 2), or to accommodate combined effects of different factors, would increase the power of the program to discriminate those species suited to a given site. Also, the ability of the PREDIC program to select an appropriate Markov transition table from a family of tables based on different site indices for the respective species (rather than using a single table for each species) would also produce more precise and realistic predictions.

The argument that increased program sophistication would result in increased program fidelity (i.e., precision) reinforces the contention that insufficiencies of this program are technological rather than theoretical in nature.

A serious obstacle in implementing the PREDIC program is the large data base required as input. Prospective users must have appropriate cellular areal data on elevation, species composition, and presence of streams. In addition, users must supply Markov transition matrices that reflect rates of radial growth of the tree species endemic to the locale being studied. While such extensive input data may be an impediment to some users, this should not detract from the overall merit of the computer program itself.

The addage "You can lead a horse to water, but you can't make him drink" is appropriate to the discussion of the implementation of the methodology developed in this research. It is believed that the systems produced in this research effort can be useful tools to those who choose to use them. However, some potential users may not choose to do so for a variety of reasons unrelated to the merits of the tools themselves.

CONCEPTS

It is concluded that the concept of dynamic classification is valid and that such an approach to the classfication of land for wildlife management purposes has utility. It appears to be an appropriate concept and suggests a basis for a methodology for providing information pertinent to wildlife management. It is believed that a dynamic classification approach could be useful in providing information for management of virtually any physical resource, the fundamental attributes of which are quantifiable.

Classification is a common tool of science. It is negentropic in nature, in that it enables the human mind to organize and structure disaggregations, and to put "things" into a context, and to allow, even encourgae, positive feedback. Classification is an information-generating process. Inasmuch as conventional classification is a special case of dynamic classification, it follows that dynamic classification must possess at least as much scientific utility as conventional classification. However, because a dynamic classification process is not specifically or purposefully oriented towards generalizations about the objects being classified, dynamic classification is inherently more precise than conventional classification. Also, since a potentially greater amount of information may be derived from a dynamic classification process than from a conventional classification process, the former has the greater potential utility.

THE APPLICABILITY OF DYNAMIC CLASSIFICATION

In this text, the concept of dynamic classification was applied within the field of wildlife management. A variety of applications is foreseen in allied fields such as agriculture, forestry, and land use planning in general. In these applications, suitability of the land for some specified purpose other than as wildlife habitat would be the appropriate objective.

It is entirely possible that other applications may be found in various natural sciences. It is also possible that a dynamic classification approach would be operationally valid for classifying intangible entities. The dynamic classification concept may indeed have metaphysical applications as well as purely physical ones. However, an adequate discussion of this topic is beyond the scope of this study, and analysis of these possibilities must await further conceptual development and investigation.

Literature Cited

Adams, H. S. 1974. Analysis of vegetation of the south slopes of Peters Mountain, Virginia. Unpubl. Ph. D. Dissert., Virginia Polytechnic Inst. and State Univ., Blacksburg. 192pp.

Aldous, S. E. 1944. A deer browse survey method. J. Mammal. 25(2):130-136.

Allen, D. L. and H. D. McGinley. 1947. A method for the year-to-year measurement of mast yields. J. Wildl. Manage. 11:184-185.

Allen, P. F., L. E. Garland, and R. F. Dugan. 1963. Rating northeastern soils for their suitability for wildlife habitat. Trans. N. Am. Wildl. Conf. 28:247-261.

Antenucci, J., S. A. Miller, and C. R. Brunori. 1979. Maryland wildlife resources information retrieval system. Trans. N. Am. Wildl. Nat. Resour. Conf. 44:446-456.

Arnold, M. C. 1946. Cover mapping for forest and wildlife management in Connecticut. Trans. N. Am. Wildl. Conf. 11:330-338.

Bailey, R. G., R. D. Pfister, and J. A. Henderson. 1978. Nature of land and resource classification - a review. J. For. 76(10):650-655.

Baker, F. S. 1949. A revised tolerance table. J. For. 47:179-181.

Beattie, K. H. 1979. A social psychological investigation of attitudes of Virginia sportsmen toward game laws and regulations. Ph. D. Dissertation, Virginia Polytech. Inst. and State Univ., Blacksburg. 217pp.

Berger, M. E. (general chairman) 1978. Proc. Pecora IV symposium: application of remote sensing data to wildlife management. National Wildlife Federation Scientific and Tech. Ser. 3, 397pp.

Best, R. G., and S. Sather-Blair. 1978. The interpretation of winter wildlife habitat in eastern South Dakota on Landsat imagery. Pages 50-56 in M. E. Berger, general chairman, Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation Scientific and Tech. Ser. 3.

Billings, W. D. 1952. The environmental complex in relation to plant growth and distribution. Quart. Rev. Biol. 27:251-265.

Bledsoe, L. J., and G. M. Van Dyne. 1971. A compartment model simulation of secondary succession. Pages 480-511 in B. C. Patten, ed., Systems analysis and simulation in ecology, Vol. I. Academic Press, Inc. New York.

Botkin, D. B., J. F. Janak, and J. R. Wallis. 1972. Rationale, limitations, and assumptions of a northeastern forest growth simulator. IBM J. Res. Dev. 16(2):101-116.

Boyce, S. G. 1977. Management of eastern hardwood forests for multiple benefits (DYNAST-MB). U. S. D. A. For. Serv. Res. Pap. SE-168. 116pp.

Boyce, S. G. 1978. Management of forests for timber and related benefits (DYNAST-TM). U. S. D. A. For. Serv. Res. Pap. SE-184. 140pp.

Braun, E. L. 1950. Deciduous forests of eastern North America. The Free Press, New York. 596pp.

Bruner, H. D., and J. W. Moser. 1973. A Markov chain approach to the prediction of diameter distributions in uneven-aged forest stands. Can. J. For. Res. 3:409-417.

Buckner, J. L. and C. J. Perkins. 1974. Forest game habitat evaluation by International Paper Company. Proc. 28th Ann. Conf. Southeast. Assoc. Game and Fish Comm. 1974:675-682.

Bureau of Sport Fisheries and Wildlife. 1974. Developmental history and ecology of the Dismal Swamp with recommendations for public ownership and management. Bureau of Sport Fisheries and Wildlife, Northeast Region, Dismal Swamp National Wildlife Refuge, Suffolk, Virginia. 47pp (Mimeograph).

Burger, D. 1972. Forest site classification in Canada. Mitt Vereins Forstl Stankortsk Forstpflz 21:20-36.

Buttery, R. F. 1978. Modified Ecoclass--a Forest Service method for classifying ecosystems. Pages 157-168 in H. G. Lund, V. J. LaBau, P. F. Ffolliott, and D. W. Robinson. (tech. coords.) 1978. Integrated inventories of renewable natural resources: proceedings of the workshop Jan. 8-12, 1978, Tucson, Ariz. U. S. For. Serv. Gen. Tech. Rep. RM-55.

Byrne, J. G. and W. D. Zeedyk. 1965. The application of soil survey information to forest-game habitat management on the Cumberland National Forest. Proc. 19th Ann. Conf. Southeast. Assoc. Game and Fish Comm. 1965:174-181.

Carmean, W. H. 1975. Forest site quality evaluation in the United States. Adv. Agron. 27:209-269.

Cason, T. W. 1980. PHAEDRUS: a system for the preliminary environmental evaluation of large physical projects. M.S. Thesis, Virginia Polytech. Inst. and State Univ., Blacksburg. 305pp.

Chapman, D. G. and C. O. Junge. 1956. The estimation of the size of a stratified animal population. Ann. Math. Stat. 27:375-389.

Churchman, C. W. and R. L. Ackoff. 1954. An approximate measure of value. J. Oper. Res. Soc. Am. 2:172-187.

Clarke, G. L. 1954. Elements of ecology. Wiley, New York. 534pp.

Clements, F. E. 1916. Plant succession. An analysis of the development of vegetation. Carnegie Inst., Washington, D. C. 512pp.

Clements, F. E. 1928. Plant succession and indicators. H. W. Wilson Co., New York. 453pp.

Clements, F. E., and G. W. Goldsmith. 1924. The phytometer method in ecology: plants and community as instruments. The Carnegie Institution of Washington, Washington, D. C. 106pp.

Cline, M. G. 1949. Basic principles of soil classification. Soil Sci. 67:81-91.

Cormack, R. M. 1971. A review of classification. J. R. Statist. Soc., Series A, 134:321-367.

Coulombe, H. N. 1978. Toward an integrated ecological assessment of wildlife habitat. Pages 5-23 in H. G. Lund, V. J. LaBau, P. F. Ffolliott, and D. W. Robinson (tech. coords.) Integrated inventories of renewable natural resouurces: proc. of the workshop. Tucson, Ariz., Jan. 8-12, 1978. U. S. For. Serv. Gen. Tech. Rep. RM-55.

Cowles, C. J. 1979. Optimal deployment of wildlife law enforcement agents with analyses of agent productivity. Ph.D. Dissert., Virginia Polytech. Inst. and State Univ., Blacksburg. 230pp.

Coyle, J. F. 1980. A preliminary methodology for simulating the dynamics of a pine vole population. M. S. Thesis, Virginia Polytech. Inst. and State Univ., Blacksburg. 120pp.

Crawford, B. T. 1946. Wildlife sampling by soil types. Trans. N. Am. Wildl. Conf. 11:357-364.

Crawford, B. T. 1950. Some specific relationships between soils and wildlife. J. Wildl. Manage. 14:115-122.

Culver, D. C. 1981. On using Horn's Markov succession model. Am. Nat. 117:572-574.

Curtis, J. T. 1955. A prairie continuum in Wisconsin. Ecology 36:558-566.

Curtis, J. T. and R. P. McIntosh. 1951. An upland forest continuum in the prairie-forest border regions of Wisconsin. Ecology 32:476-496.

Dalke, P. D. 1941. Development of a statewide system of cover-mapping applicable to Missouri wildlife range. J. Wildl. Manage. 5:103-107.

Daniel, C. and R. Lamaire. 1974. Evaluating effects of water resource developments on wildlife habitat. Wildl. Soc. Bull. 2(3):114-118.

Daniels, R. F., R. Rogers, D. B. Schwendeman, and L. J. Wolgast. 1974. The SYMAP computer program as an aid in evaluating and manipulating wildlife habitat. Trans. Northeast Sect. The Wildlife Society 31:101-111.

Dansereau, P. 1957. Biogeography: an ecological perspective. The Ronald Press Company, New York. 394pp.

Daubenmire, R. 1968. Plant communities: a textbook of plant synecology. Harper & Row, Publishers, New York. 300pp.

Davenport, L. A., W. Shapton and W. C. Gower. 1944. A study of the carrying capacity of deer yards as determined by browse plots. Trans. N. Am. Wildl. Conf. 9:144-149.

Dean, F. C. 1972. POPSID, a FORTRAN IV program for deterministic population studies. Oak Ridge National Laboratory, ORNL-IBP-71-11. 42pp.

Deschene, W. A. 1977. Evaluation of the Wildland Resource Information System for general land-use planning. M. S. Thesis, Univ. of Idaho, Moscow. 115pp.

DeVos, A. and H. S. Mosby. 1971. Habitat analysis and evaluation. Pages 135-172 in R. H. Giles (ed.) Wildlife management techniques, third ed.: revised. The Wildlife Society, Washington, D. C.

Downs, A. A. 1944. Estimating acorn crops for wildlife in the southern Appalachians. J. Wildl. Manage. 8:292-293.

Eberhardt, L. L. 1971. Population analysis. Pages 457-495 in R. H. Giles (ed.) Wildlife management techniques, third ed.: revised. The Wildlife Society, Washington, D. C.

Ehrenreich, J. H., and J. S. Crosby. 1960. Herbage production is related to hardwood crown cover. J. For. 58:564-565.

Elton, C. 1927. Animal ecology. Sidwick & Jackson Ltd, London. 207pp.

Fales, R. R. 1969. Some applications of computer-generated maps to wildlife management. M. S. Thesis, Virginia Polytech. Inst. and State Univ., Blacksburg. 83pp.

Farmer, A. H. 1978. The habitat evaluation procedures. Pages 407-420 in A. Marmelstein (gen. chairman) Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76.

Federation of Rocky Mountain States, Inc. 1977. Composite Mapping System II users' guide. Federation of Rocky Mountain States, Inc., Denver, Colo. 209pp.

Flood, B. S., M. E. Sangster, R. D. Sparrowe, and T. S. Baskett. 1977. A handbook for habitat evaluation procedures. U. S. Fish and Wildl. Serv. Resour. Publ. 132. 77pp.

Forbes, R. D. 1955. Forestry handbook. The Ronald Press Company, New York. vp.

Fowells, H. A. 1965. Silvics of forest trees of the United States. U. S. D. A., For. Serv. Agriculture Handbook No. 271. 762pp.

Franklin, J. F. 1979. Simulation modeling and resource management. Environ. Manage. 3(1):2-5.

Fries, J. (ed.) 1974. Growth models for tree and stand simulation. Res. Notes 30, Dept. For. Yield Res. Royal Coll. For., Stockholm. 379pp.

Gammon, P. T. and V. Carter. 1979. Vegetation mapping with seasonal color infrared photographs. Photog. Eng. and Rem. Sens. 45:87-97.

Gause, G. F. 1934. The struggle for existence. Williams & Wilkins, Baltimore. 163pp.

Giles, R. H. 1978. Wildlife management. W. H. Freeman and Company, San Francisco. 416pp.

Gilmour, J. S. L. 1951. The development of taxonomic theory since 1851. Nature 168:400-402.

Gingrich, S. F. 1971. Management of young and intermediate stands of upland hardwoods. U. S. D. A., For. Serv. Res. Pap. NE-195. 26pp.

Good, I. J. 1965. Categorization of classification. Pages 115-128 In Mathematics and computer science in medicine and biology. H.M.S.O., London.

Goodall, D. W. 1963. The continuum and the individualistic association. Vegetatio 11:297-316.

Goodall, D. W. 1966. Hypothesis-testing in classification. Nature 211:329-330.

Goodall, D. W. 1973. Numerical classification. Pages 575-615 in R. H. Whittaker (ed.) Handbook of vegetation science 5: Ordination and classification of communities. Junk, The Hague, Netherlands.

Goodrum, P. D., V. H. Reid, and C. E. Boyd. 1971. Acorn yields, characteristics, and management criteria of oaks for wildlife. J. Wildl. Manage. 35:520-532.

Graham, S. A. 1945. Ecological classification of cover types. J. Wildl. Manage. 9:182-190.

Grinnell, J. 1917a. Field tests of theories concerning distributional control. Amer. Natur. 51(602):115-128.

Grinnell, J. 1917b. The niche-relationships of the California thrasher. Auk 34:427-433.

Grinnell, J. 1924. Geography and evolution. Ecology 5:225-229.

Grinnell, J. 1928. Presence and absence of animals. Univ. Calif. Chron. 30:429-450.

Gutierrez, L. T., and W. R. Fey. 1977. Simulation of successional dynamics in ecosystems. Pages 73-82 in G. S. Innis (ed.) New directions in the analysis of ecological systems, Part 1. Simulation Councils Proc. Series, Vol. 5, No. 1. Soc. for Computer Simulation (SCI), La Jolla, Calif.

Gutierrez, L. T., and W. R. Fey. 1980. Ecosystem succession: a general hypothesis and a test model of a grassland. The MIT Press, Cambridge, Mass. 231pp.

Hall, C. A. S. and J. W. Day. 1977. Systems and models; terms and basic principles. Pages 6-36 in C. A. S. Hall and J. W. Day (eds.). Ecosystem modeling in theory and practice: an introduction with case histories. John Wiley & Sons, Inc. New York.

Hardin, G. 1960. The competitive exclusion principle. Science 131:1291-1297.

Harlow, R. F. 1955. Preliminary report of a deer browse census based on 100% clipping method. Proc 9th Ann. Conf. Southeast. Assoc. Game and Fish. Comm. 1955:134-156.

Harlow, W. M., and E. S. Harrar. 1969. Textbook of dendrology. Fifth Ed. McGraw-Hill Book Company, New York. 512pp.

Hillier, F. S. and G. J. Lieberman. 1980. Introduction to operations research. Third ed. Holden-Day, Inc., San Francisco. 829pp.

Horn, H. S. 1975a. Forest succession. Sci. Amer. 232:90-98.

Horn, H. S. 1975b. Markovian properties of forest succession. Pages 196-211 in J. Diamond, and M. L. Cody (eds.) Ecology of species and communities. Harvard Univ. Press, Cambridge.

Horn, H. S. 1976. Succession. Pages 187-204 in R. S. May (ed.) Theoretical ecology: principles and applications. W. B. Saunders Co., Philadelphia.

Hulst, R. van. 1978. The dynamics of vegetation: patterns of environmental and vegetational change. Vegetatio 38:65-75.

Hulst, R. van. 1979. On the dynamics of vegetation: Markov chains as models of succession. Vegetatio 40:3-14.

Hutchinson, G. E. 1958. Concluding remarks. Cold Spring Harbor Symp. Quant. Biol. 22:415-427.

Hutchinson, G. E. 1965. The ecological theater and the evolutionary play. Yale University Press, New Haven, Conn. 139pp.

Hutchinson, G. E. 1967. A treatise on limnology, Vol. 2. Introduction to lake biology and the limnoplankton. Wiley, New York. 1,115pp.

Ike, A. F., and C. D. Happuch. 1968. Predicting tree height growth from soil and topographic site factors in the Georgia Blue Ridge Mountains. Ga. For. Res. Council Pap. 54. 11pp.

Johnson, R. H. 1910. Determinate evolution in the color pattern of the lady-beetles. Carnegie Inst. Washington Publ. 122. 104pp.

Johnson, W. C. and D. M. Sharpe. 1976. An analysis of forest dynamics in the northern Georgia piedmont. For. Sci. 22:307-322.

Jolly, G. M. 1963. Estimates of population parameters from multiple recapture data with both death and dilution - deterministic model. Biometrika 50:113-128.

Jolly, G. M. 1965. Explicit estimates from capture-recapture data with low death and immigration - stochastic model. Biometrika 52:315-337.

Jones, J. R. 1969. Review and comparison of site evaluation methods. U. S. D. A. For. Serv. Res. Pap. RM-51. 27pp.

Katibah, E. F., and W. C. Graves. 1978. Remote sensing-aided assessment of wild turkey habitat. Pages 78-81 in M. E. Berger (gen. chairman) Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation Scientific and Tech. Ser. 3.

Kendall, M. G. 1966. Discrimination and classification. Pages 165-185 In P. R. Krishnaiah (ed.) Proc. Symp. Multiv. Analysis. Academic Press, New York.

Kessell, S. R. 1976. Gradient modeling: a new approach to fire modeling and wilderness resource management. Env. Manage. 1(1):39-48.

Kessell, S. R. 1979a. Gradient modeling: resource and fire management. Springer-Verlag, New York. 432pp.

Kessell, S. R. 1979b. Phytosociological inference and resource management. Env. Manage. 3(1):29-40.

Kessell, S. R. and M. W. Potter. 1980. A quantitative succession model for nine Montana forest communities. Environ. Manage. 4:227-240.

Kirkpatrick, R. L. 1975. Applicability of basic physiology and nutrition research to practical wildlife management. Proc. Southeastern Assoc. Game and Fish Comm. 29:476-480.

Kirkpatrick, R. L. 1980. Physiological indices in wildlife management. Pages 99-112 in S. D. Schemnitz (ed.) Wildlife management techniques manual, Fourth edition: revised. The Wildlife Society, Washington, D. C.

Lance, G. N. and W. T. Williams. 1967a. A general theory of classificatory sorting strategies I. hierarchical systems. Comp. J. 9:373-380.

Lance, G. N. and W. T. Williams. 1967b. A general theory of classificatory sorting strategies II. clustering systems. Comp. J. 10:271-277.

Lance, G. N. and W. T. Williams. 1967c. Note on the classification of multi-level data. Comp. J. 9:381-382.

Layser, E. F. 1974. Vegetative classification: its application to forestry in the northern Rocky Mountains. J. For. 72(6):354-357.

Leak, W. B. 1970. Successional change in northern hardwoods predicted by birth and death simulation. Ecology 51(5):794-801.

Leedy, D. L. 1948. Aerial photographs, their interpretation and suggested uses in wildlife management. J. Wildl. Manage. 12:191-210.

Leopold, A. 1933. Game management. Charles Scribner's Sons, New York. 481pp.

Leslie, P. H. 1945. On the use of matrices in certain population mathematics. Biometrika 33:183-212.

Leslie, P. H. 1948. Some further notes on the use of matrices in population mathematics. Biometrika 35:213-245.

Leslie, P. H. 1952. The estimation of population parameters by means of the capture-recapture method. II. The estimation of total numbers. Biometrika 39:363-388.

Leslie, P. H., and D. Chitty. 1951. The estimation of population parameters from data obtained by means of the capture-recapture method. I. The maximum likelihood equations for estimating the death rate. Biometrika 38:269-292.

Leslie, P. H., D. Chitty, and H. Chitty. 1953. The estimation of population parameters from data obtained by means of the capture-recapture method. III. An examination of the practical applications of the method. Biometrika 40:137-169.

Levy, G. F., and S. W. Walker. 1979. Forest dynamics in the Dismal Swamp of Virginia. Pages 101-126 in P. W. Kirk, Jr. (ed.) The Great Dismal Swamp. The University Press of Virginia, Charlottesville.

Linstone, H. A., and M. Turoff (eds.) 1975. The Delphi method: techniques and applications. Addison-Wesley, Reading, Mass. 620pp.

Lund, H. G., V. J. LaBau, P. F. Ffolliott, and D. W. Robinson. technical coords. 1978. Integrated inventories of renewable natural resources: proceedings of the workshop Jan. 8-12, 1978, Tucson, Ariz. U. S. For. Serv. Gen. Tech. Rep. RM-55. 482pp.

McIntosh, R. P. 1967. The continuum concept of vegetation. Bot. Rev. 33:130-187.

MacArthur, R. H. 1968. The theory of the niche. Pages 159-176 in R. C. Lewontin, (ed.) Population biology and evolution. Syracuse University Press, Syracuse, New York.

Macnaughton-Smith, P. 1963. The classification of individuals by the possession of attributes associated with a criterion. Biometrics 19:364-366.

Marmelstein, A., general chairman. 1978. Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76. 604pp.

Mayr, E. 1968. Theory of biological classification. Nature 220:545-548.

Meanley, B. 1971. Great Dismal Swamp mammals. Atlantic Nat. 26:17-18.

Merchant, J. W. and B. H. Waddell. 1974. The use of high altitude photography and ERTS-1 imagery for wildlife habitat inventory in Kansas. Proc. Am. Soc. Photogr. 220-231.

Meyers, C. R., M. Kennedy, and R. N. Sampson. 1979. Information systems for land use planning. Pages 889-907 in M. T. Beatty, G. W. Petersen, and L. D. Swindale (eds.) Planning the uses and management of land. Number 21 in the series Agronomy. Am. Soc. Agron., Crop Sci. Soc. Am., Soil Sci. Soc Am., Inc.

Mielke, D. L., H. H. Shugart, and D. C. West. 1978. Stand model for upland forests of southern Arkansas. ORNL/TM-6225. (Oak Ridge National Laboratory, Oak Ridge, Tenn.) 97pp.

Mitchell, W. B., S. C. Guptill, K. E. Anderson, R. G. Fegas and C. A. Hallam. 1977. GIRAS: a geographic information retrieval and analysis system for handling land use and land cover data. Geological Survey Prof. Pap. 1059. 16pp.

Morris, R. G. and B. E. Nichols. 1979. Wildlife habitat evaluation and mitigation. Pages 369-377 in Swanson, G. A. (tech. coord.) The mitigation symposium: a national workshop on mitigating losses of fish and wildlife habitats. July 16-20, 1979, Fort Collins, Colo. U. S. For. Serv. Gen. Tech. Rep. RM-65.

Morse, M. A. 1943. Technique for reducing man-power in deer drive census. J. Wildl. Manage. 7(2):217-220.

Mueller-Dombois, D., and H. Ellenberg. 1974. Aims and methods of vegetation ecology. John Wiley & Sons, Inc., New York. 547pp.

Moser, J. W. 1978. Discrete time Markov processes. Pages 21-27 in Midwest forest mensurationists meeting. U. S. D. A. For. Serv. Gen. Tech. Rep. NC-46.

Nelson, D., G. A. Harris, and T. E. Hamilton. 1978. Land resource classification - who cares? J. For. 76(10):644-646.

Nelson, T. C., J. L. Clutter, and L. E. Claiken. 1961. Yield of Virginia pine. U. S. D. A., For. Serv., Southeast For. Sta. Pap. 124. 1pp.

Odum, E. P. 1953. Fundamentals of ecology. Saunders, Philadelphia, Pa. 384pp.

Odum, E. P. 1963. Ecology. Holt, Rhinehart and Winston, Inc. New York. 152pp.

Odum, E. P. 1971. Fundamentals of ecology. Third ed. W. B. Saunders Company, Philadelphia, Pa. 574pp.

Olson, D. F. 1959. Site index curves for upland oaks in the Southeast. U. S. D. A., For. Serv., Southeast. For. Exp. Sta. Res. Note 125. 2pp.

Oosting, H. J. 1956. The study of plant communities: an introduction to plant ecology. Second ed. W. H. Freeman and Company, San Francisco, 440pp.

Overton, W. S. 1971. Estimating the numbers of animals in wildlife populations. Pages 403-455 in R. H. Giles (ed.) Wildlife management techniques, third ed.: revised. The Wildlife Society, Washington, D. C.

Parker, H. D. 1978. Integration of operational remote sensing systems with conventional inventory procedures. Pages 543-552 in A. Marmelstein (gen. chairman) Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76.

Patten, B. C. 1962. Species diversity in net plankton of Raritan Bay. J. Mar. Res. 20:57-75.

Peden, L. M., J. S. Williams and W. E. Frayer. 1973. A Markov model for stand projection. For. Sci. 19(4):303-314.

Penfound, W. T. 1952. Southern swamps and marshes. Bot. Rev. 18:413-446.

Perkins, C. J. 1973. A plan of wildlife habitat evaluation. International Paper Company, Southlands Exp. For. For. Res. Lett. No. 18. 12pp.

Pettinger, L. R., A. Farmer, and M. Schamberger. 1978. Quantitative wildlife habitat evaluation using high-altitude color infrared aerial photographs. Pages 335-345 in M. E. Berger (gen. chairman) Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation Scientific and Tech. Ser. 3.

Pfister, R. D. and S. F. Arno. 1980. Classifying forest habitat types based on potential climax vegetation. For. Sci. 26(1):52-70.

Pielou, E. C. 1977. Mathematical ecology. John Wiley & Sons, New York. 385pp.

Platts, W. S. 1980. A plea for fishery habitat classification. Fisheries 5(1):2-6.

Putnam, J. A., G. M. Furnival, and J. S. McKnight. 1960. Management and inventory of southern hardwoods. U. S. D. A., For. Serv. Agr. Handbook No. 181. 102pp.

Robertson, J. C. 1967. SYMAP program for computer mapping. Cartog. J. 4(2):108-113.

Roller, N. E. G. 1978. Quantitative evaluation of deer habitat. Pages 137-146 in M. E. Berger (gen. chairman) Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation, Scientific and Tech. Ser. 3.

Rosove, P. E. 1967. Developing computer-based information systems. John Wiley and Sons, Inc. New York. 384pp.

Rowe, J. S. 1971. Why classify forest land? For. Chron. 47(2):144-148.

Schmautz, J. E. 1970. A model to incorporate soil survey and allotment analysis data into range management planning. Pages 219-220 in Range and wildlife habitat evaluation: a research symposium. U. S. D. A. For. Serv. Misc. Publ. No. 1147.

Schmautz, J. E. 1978. The RPA-80 land classification system. Pages 179-186 in A. Marmelstein (gen. chairman) Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76.

Schnabel, Z. E. 1938. Estimation of the total fish population of lakes. Amer. Math. Monthly 45:348-352.

Schnurr, G. L. 1937. Yield, stand, and volume tables for even-aged upland oak forests. U. S. D. A. Tech. Bull. No. 560. 87pp.

Seal, U. S. 1977. Assessment of habitat condition by measurement of biochemical and endocrine indicators of the nutritional, reproductive and disease status of free-ranging animal populations. Pages 305-329 in A. Marmelstein (gen. chairman) Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76.

Shelford, V. E. 1963. The ecology of North America. University of Illinois Press, Urbana. 610pp.

Short, C., and M. Schamberger. 1979. Evaluation of impacts on fish and wildlife habitat and development of mitigations measures. Pages 331-335 in G. A. Swanson (tech. coord.) The mitigation symposium: a national workshop on mitigating losses of fish and wildlife habitats. July 16-20, 1979, Fort Collins, Colo. U. S. For. Serv. Gen. Tech. Rep. RM-65.

Shugart, H. H., T. R. Crow, and J. M. Hett. 1973. Forest succession models: a rationale and methodology for modeling forest succession over large regions. For. Sci. 19(3):203-212.

Shugart, H. H., and D. C. West. 1977. Development of an Appalachian deciduous forest succession model and its application to assessment of the impact of the chestnut blight. J. Env. Manage. 5:161-179.

Shugart, H. H., and D. C. West. 1980. Forest succession models. Bioscience 30(5):308-313.

Slayter, R. O. (ed.) 1977. Dynamic changes in terrestrial ecosystems: patterns of change, techniques for study and applications to management. MAB Technical Notes 4. UNESCO. 30pp.

Smalley, G. W., and R. L. Bailey. 1974. Yield tables and stand structure for shortleaf pine plantations in Tennessee, Alabama, and Georgia highlands. U. S. D. A., For. Serv. Res. Pap. SO-97. 57pp.

Smith, R. L. 1974. Ecology and field biology. Second ed. Harper & Row, Publishers, New York. 850pp.

Sneath, P. H. A. 1966. A method for curve seeking from scattered points. Comp. J. 8:383-391.

Society of American Foresters. 1954. Forest cover types of North America (exclusive of Mexico). Soc. Am. For., Bethesda, Maryland. 67pp.

Spaeth, J. N. 1920. Growth study and normal yield tables for second-growth hardwood stands in central New England. Harvard For. Bull. No. 2. (cited by Forbes; not seen).

Sutherland, J. W. 1975. Systems: analysis, administration, and architecture. Van Nostrand Reinhold Company, New York. 339pp.

Swanson, G. A. (tech. coord.) 1979. The mitigation symposium: a national workshop on mitigating losses of fish and wildlife habitats. July 16-20, 1979, Fort Collins, Colo. U. S. For. Serv. Gen. Tech. Rep. RM-65, 696pp.

Taft, K. A., and S. E. Davis. 1978. Integrated forest wildlife management through graphical analysis. Eighth World Forestry Congress, Jakarta, Indonesia. 12pp.

Thomas, J. W. (tech. ed.) 1979. Wildlife habitats in managed forests of the Blue Mountains of Oregon and Washington. U. S. D. A., For. Serv. Agr. Handbook No. 553. 512pp.

Thomas, J. W., R. Miller, C. Maser, R. Anderson, and B. Carter. 1978. The relationship of terrestrial vertebrates to plant communities and their successional stages. Pages 281-304 _in_ A. Marmelstein (gen. chairman) Classification, inventory, and analysis of fish and wildlife habitat. Office of Biol. Serv., Fish and Wildl. Serv. FWS/OBS - 78/76.

Tipton, A. R. 1975. A matrix structure for modeling population dynamics. Ph.D. Dissertation, Michigan State Univ., East Lansing. 52pp.

Traylor, C. T., and W. T. Mealor, Jr. 1978. Landsat imagery for wildlife habitat inventory mapping. Pages 159-161 _in_ M. E. Berger (gen. chairman) Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation Scientific and Tech. Ser. 3.

Treadwell, B. D. 1978. Wildlife habitat mapping from color aerial photography in central Arizona. Pages 377-384 in M. E. Berger (gen. chairman) Proc. Pecora IV symposium: application of remote sensing data to wildlife management. Oct. 10-12, 1978, Sioux Falls, South Dak. National Wildlife Federation Scientific and Tech. Ser. 3.

U. S. Army Corps of Engineers. 1976. A tentative habitat evaluation system (HES) for water resources planning. U. S. Army Corps of Eng., Lower Mississippi Valley Div., Vicksburg, Mississippi. vp.

U. S. D. A. 1971. Wildlife habitat management handbook. Southern region. FSH 2609.23R. U. S. D. A. For. Serv. vp.

Vandermeer, J. H. 1972. Niche theory. Ann. Rev. Ecol. Syst. 3:107-132.

Villar, E. H. del. 1929. Geobotanica. Editorial Labor, Barcelona-Buenos Aires, 339pp.

Waggoner, P. E. and G. R. Stephens. 1970. Transition probabilities for a forest. Nature 225:1160-1161.

Warren, R. J. 1979. Physiological indices for the assessment of nutritional status in white-tailed deer. Ph.D. Dissertation, Virginia Polytech. Inst. and State Univ., Blacksburg. 250pp.

Wartofsky, M. W. 1968. Conceptual foundations of scientific thought: an introduction to the philosophy of science. The MacMillan Company, New York. 560pp.

Whelan, J. B., A. R. Tipton, J. F. Williamson, P. R. Johansen, J. P. McClure, and N. D. Cost. 1979. A comparison of three systems for evaluating forest wildlife habitat. Trans. N. Am. Wildl. and Nat. Resour. Conf. 44:392-403.

White, D. L. 1980. USGS Digital Elevation Model data base. U. S. G. S., Nat. Cartog. Info. Center, Reston, Va. 5pp. (mineogr.)

Whittaker, R. H. 1967. Gradient analysis of vegetation. Biol. Rev. 49:207-264.

Williams, W. T. 1971. Principles of clustering. Ann. Rev. Ecol. Syst. 2:303-326.

Williamson, J. F., A. R. Tipton, R. M. Keel, and J. B. Whelan. 1981. An assessment of black bear habitat suitability in the Great Dismal Swamp 1980. Trans. Northeast. Sect. The Wildl. Soc. (In press).

Williamson, J. F., and G. T. Koeln. 1980. A computerized wild turkey habitat evaluation system. Proc. Nat. Wild Turkey Symp. 4:233-239.

Williamson, J. F., and J. B. Whelan. 1980. Computer assisted black bear management in Shenandoah National Park. Proc. Fifth Int. Conf. on Bear Res. and Manage. (In press).

Willis, R. 1975. A technique for estimating potential wildlife populations through habitat evaluations. Kentucky Dept. Fish and Wildl. Resour. Pittman-Robertson Game Manage. Tech. Series No. 23. 12pp.

Wilm, H. G., D. F. Costello and G. E. Klipple. 1944. Estimating forage yield by the double-sampling method. J. Am. Soc. Agron. 36:194-203.

Wishart, D. 1969. Numerical classification method for deriving natural classes. Nature 221:97-98.

Witmer, R. E. 1978. The USGS land use and land cover classification system. Pages 178-182 in H. G. Lund, V. J. LaBau, P. F. Ffolliott, and D. W. Robinson. (tech. coords.) Integrated inventories of renewable natural resources: proceedings of the workshop Jan. 8-12, 1978, Tucson, Ariz. U. S. For. Serv. Gen. Tech. Rep. RM-55.

Wuenscher, J. E. 1974. The ecological niche and vegetation dynamics. Pages 39-45 in B. R. Strain, and W. D. Billings (eds.) Handbook of vegetation science, Part VI, Vegetation and environment. Junk, The Hague.

Appendix A

MAP4B USERS' GUIDE

Foreword

This document is the users' guide for MAP4B, an original computer mapping program developed within the Department of Fisheries and Wildlife Sciences at Virginia Polytechnic Institute and State University. While a familiarity with computers may prove beneficial to some users, this knowledge is not essential to use MAP4B successfully. This users' guide was designed to be informative and to instruct users in the use of the MAP4B system. Examples have been provided where it was felt they were needed. However, if problems or questions do arise, users are encouraged to contact:

J. F. Williamson
Department Fisheries and Wildlife Sciences
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061

Notice to Programmers

The MAP4B program is written in the FORTRAN computer language, and has been successfully compiled on FORTRAN G, H, H-extended, and WATFIV compilers. The main program requires approximately 500k of memory.

Because of differences between individual computing centers, job control language (JCL) has been omitted. Any required JCL must be supplied by users. Users unfamiliar with JCL may obtain this information from their respective computing centers.

The MAP4B program was designed to reduce repetitious map input data, i.e., identical data comprising more that one row of a map. Users may use a row continuation field in the input data to indicate two or more identical rows of map data. A non-zero value in this field activates a program loop that backspaces the read device (unit 4). Users with remote interactive terminal capabilities must designate the input device as unit 4. If actual card input is utilized, appropriate JCL must be provided to designate unit 4 as the input device number.

The version of the program supplied uses device number 4 for input and device number 6 for printed output. Card output produced by auxiliary programs is written to output device number 7. Failure to designate I/O devices to these numbers with necessary JCL may result in program failure.

Equipment Requirements

MAP4B may be operated with a minimum of equipment. Users should have access to a card punch (or some type of input device capable of producing card images) and a line printer. However, these two pieces of equipment are not mandatory as keypunching or data entry can be done by many computing centers on a contractual basis and program output may be mailed to remote users.

The availability of various types of equipment may greatly expedite the use of the MAP4B system. Users having access to digitizing equipment may substantially reduce the time required for encoding map data. Special software packages are available for converting digitized data in a polygon format into the necessary cellular format. Special digitizing programs are available for digitizing data directly in cellular format. Although the programs for converting digitized data into MAP4B-compatible format are general in nature, they may not be compatible with all digitizing equipment, or the data produced by different makes of equipment. Some modifications, in the form of program alterations or additional programming, may be necessary to ensure complete compatibility with user-suppliled equipment.

This Users Guide outlines the use of the MAP4B program in the manual mode. Additional information regarding using MAP4B in other modes is available from the source given in the Foreword.

Introduction

The MAP4B program is a computerized cellular mapping program. Data are read into the program in the form of cards (or card images) and output is in the form of 12 by 15 inch line printer maps. Output maps are composed of 14,400 printed characters arranged in 120 rows of 120 characters each. Details of map production are given in later sections.

The mapping system is capable of producing maps of any single mappable topic, such as forest types, counties, or distance to water. An example of a topic, or thematic, map is provided as Fig. 1. Note that the map is 120 rows long and 120 columns wide with row and column numbers along the margins. The area shown in this map represents that of one 7 1/2 minute topographic quad sheet. Also note that in this map, cells are rectangular, a feature which results in a geometrically accurate representation of the original base map.

Two general types of maps may be produced using MAP4B. These types are (1) symbol maps and (2) numeric maps. Symbol maps are useful where a variety of characteristics are to be displayed on a single map. A total of 57 different symbols may be used for producing symbol maps. By contrast, numeric maps are composed strictly of the integers 0 (zero) through 9.

The MAP4B program has 2 main uses--to produce thematic maps (which may be either symbol or numeric type maps), and to produce composite maps. Composite maps are generated by superimposing a number of individual topic maps. Only numeric maps may be composited; symbol maps CANNOT be composited. However, if symbol maps are first converted into numeric maps using the auxiliary program CONVERT, they may be composited as numeric maps. The final product of the compositing process is called a composite map. Although composite maps are generated from numeric maps, they are a special form of symbol maps.

There are two methods for generating a composite map--an additive procedure and a multiplicative procedure. In the additive procedure, values for each cell of a map are ADDED on a cell-by-cell basis to create a map of sum values. However, in the multiplicative procedure, map cell values

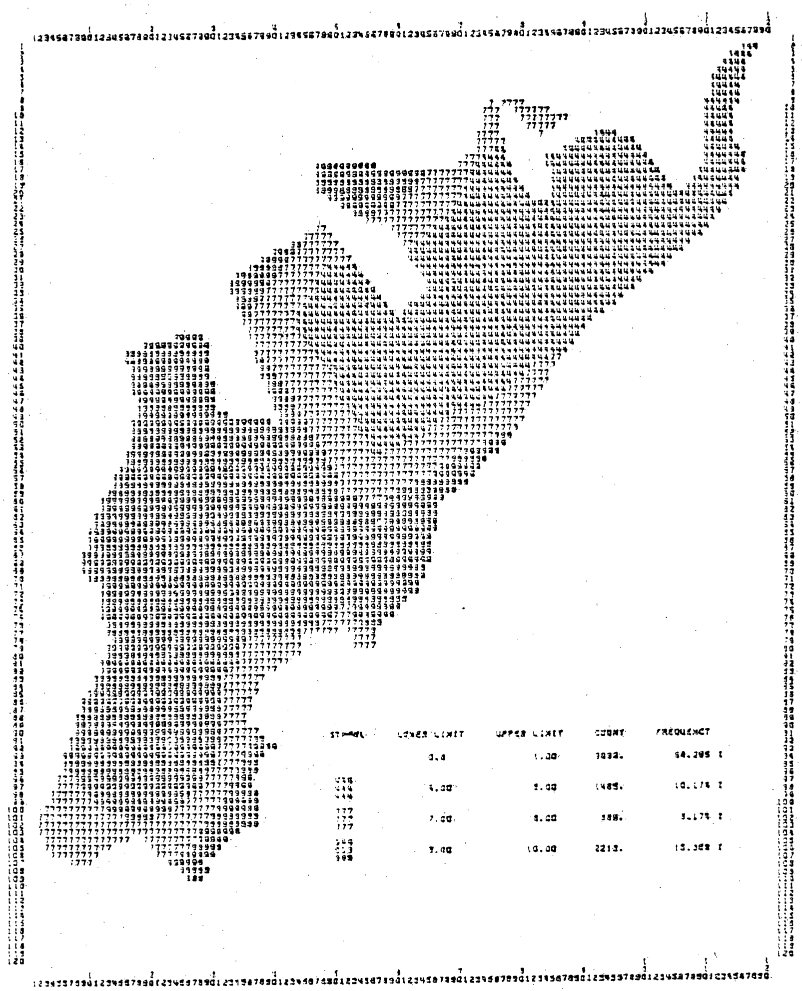


Fig. 1. A topic (thematic) map produced by MAP4B.

are MULTIPLIED in a like manner to create a map of product values.

An example of the additive compositing procedure is given in Fig. 2. Suppose the objective is to overlay two maps, Map A and Map B. These two maps are shown below as 3 by 3 numerical integer matrices.

$$\text{Map A:} \quad 1 \quad \times \quad \begin{matrix} 1 & 1 & 1 \\ 1 & 2 & 2 \\ 2 & 2 & 2 \end{matrix} \quad = \quad \begin{matrix} 1 & 1 & 1 \\ 1 & 2 & 2 \\ 2 & 2 & 2 \end{matrix}$$

$$+$$

$$\text{Map B:} \quad 2 \quad \times \quad \begin{matrix} 3 & 3 & 1 \\ 3 & 3 & 2 \\ 2 & 2 & 2 \end{matrix} \quad = \quad \begin{matrix} 6 & 6 & 2 \\ 6 & 6 & 4 \\ 4 & 4 & 4 \end{matrix}$$

$$\text{Composite matrix:} \quad \begin{matrix} 7 & 7 & 3 \\ 7 & 8 & 6 \\ 6 & 6 & 6 \end{matrix}$$

Fig. 2. Example of the compositing procedure.

Note that both Map A and Map B are numerical matrices composed of single digit integer values. The compositing procedure is identical to regular matrix addition in that corresponding cells in each map are added to produce a final composite map (matrix). If this had been the multiplicative procedure, the map weights would have been ignored, and the map data matrices would have been multiplied instead of being summed. It should be mentioned that although the input maps (matrices) must be comprised of single digit integers, the resultant composite map may have values greater than that of a single integer. A special subroutine within the MAP4B system will stratify large values into a series of categories, and display these categories as pre-specified symbols.

Resolution and Scale

Users must determine the scale that will result in the desired resolution of the output map. Users should remember that computer maps are composed of printed characters. Standard output is normally 10 characters per inch and either 6 or 8 lines per inch. (The Virginia Tech computer center prints 8 lines per inch.) There are 120 rows of 120 columns each per map for a total of 14,400 printed characters per map.

The area represented by a printed character of a map is called a cell. Because the physical dimensions of a character are fixed on the computer printout map, the degree of resolution is determined by altering cell dimensions in the encoding process.

The suggested procedure for manual encoding is to overlay a transparent grid sheet on the base map. This overlay sheet is lined in a grid fashion with each "box" in the grid representing a cell on the computer map. The larger the grid box on the overlay, the coarser the detail of the computer map. Increasingly smaller grid boxes will result in finer and finer resolution of the base map. It should be noted that using very small grid boxes will increase resolution, but the amount of coding effort required will also increase exponentially as grid cell size decreases. Also, for small grid cell sizes, several computer maps may have to be coded to cover the entire base map frame. Thus, it is important that users determine a suitable resolution within practical limits.

For general purposes, it is suggested that a single computer map be used to represent a U.S. Geological Survey (USGS) 7 1/2 minute quad sheet. This results in a single character representing 1.05 hectares on the base map, an acceptable resolution for most users. While this resolution has proven useful, users should remember that the computer map generated at this resolution is not at the same scale as the topo sheets. Thus, overlaying the base map with the computer map is not possible unless the computer map is somehow enlarged, unless the original base map is 12 inches wide and 15 inches long.

Grid Sheet Preparation

Templates for encoding maps at the suggested scale may be prepared by constructing a grid with box dimensions of .18854 (long) by .14921 (wide) inches. Users unable to accomplish this task with the necessary precision may obtain a template from the author at cost. When used with USGS quad sheets, a grid with 10 vertical lines per inch and 8 horizontal lines per inch will result in a map of the same scale as the original base map. While this grid is easy to prepare, it should be remembered that four maps must be encoded for each quad sheet.

Users preparing their own grid sheets are encouraged to prepare a grid with 60 columns and 60 (or 120) rows. Experience has shown that grids with these dimensions are easier to encode.

Map Preparation and Encoding

Data Format

Punch cards contain only 80 data columns but the maps are 120 columns wide; therefore, maps are encoded as two strips, each 60 columns wide. That is, the left half (the first 60 columns) of the map is coded onto 120 cards (one card for each row) and then the right half is coded in a similar manner.

Manual data coding is easier if standard coding forms are used. These forms are printed by IBM and other business machine companies and are generally available from office supply stores, computer centers, and college or university book stores. The formats for coding the right and left sides of the map are different. The formats for coding the left and right sides are given in Tables 1 and 2.

Users are encouraged to designate the pertinent fields (columns) on their coding sheets to expedite coding and to reduce errors. Also, labeling each code sheet with the appropriate map name, date, person coding, and code sheet number may speed the coding process.

A variety of symbols may be selected for use when coding map data. The list of potential symbols includes:

1. the letters A through Z
2. the integer numbers 0 (zero) through 9
3. the blank
4. the following special symbols: period, comma, minus, plus sign, equals sign, slash, percent sign, asterisk, dollar sign, left parenthesis, right parenthesis, semicolon, greater than (>), less than (<), colon, ampersand, exclamation point, at (@), open bracket ({), and double quote sign (").

Table 1. Format for encoding left side of map.

Field Name	Columns	Information
(blank)	1-2	none
row number	3-5	row number
row continuation	6-8	row number of last row in identical sequence
map data	9-68	map data (symbols or integers)
(blank)	69-71	none
sequence number	72-80	identifying data (optional)

Table 2. Right side coding format.

Field Name	Columns	Information
(blank)	1	none
map data	2-61	map data (symbols or integers)
blank	62	none
row number	63-65	row number
row continuation	66-68	row number of last row in identical sequence
(blank)	69-71	none
sequence number	72-80	identifying data (optional)

Users utilizing card input may be restricted in the number of usable symbols, depending on the type of keypunch or input device used. Other problems specific to certain output devices may encountered also, as some special symbols may not be printable on certain output devices. Regardless of the symbols used, it must be remembered that only single-digit characters (either alphabetic, numeric, or special symbols) may be used. Decimal values such as 1.5, 0.75, or 0.2 are not permissible.

Data Coding

Grid sheet alignment:

Once the user has determined which symbols are to be used, grid scale, etc., the actual encoding procedure may begin. To start coding, register the overlay grid along the left margin of the map to be coded. If the 60 by 60 grid sheet is used, the upper left corner of the overlay should coincide with the same corner of the map and the left side of the overlay should cover the left margin exactly. If a 60 by 120 grid is used, it should be placed length-wise along the left margin of the map with the top and bottom of the grid aligning with the map edges, so that the grid is 60 columns wide and 120 rows long. A 120 by 120 grid should, of course, be overlaid exactly on the base map.

The row field:

On the first row of the coding form enter a 001 (zero zero one), or a 1 (one) right justified in the row number field, (columns 3, 4, and 5). This signifies that this is the first row in the map. Leave columns 6, 7, and 8 blank for now; their use will be explained later.

Map data field:

Map data are recorded beginning in column 9 for the left side, and column 2 for the right side. The first 60 boxes in the first row of the grid sheet correspond to the 60 columns of the map data field of the coding sheet. Appropriate symbols are assigned to those boxes having the characteristic being mapped. When a box straddles the border of two characteristics, the user must decide how that cell will be coded. The general rule is to code cells with the symbol of the characteristic occupying the greatest percentage of the grid box. Users are reminded that uncoded boxes will be read by the program as blanks if a symbol map is being read, or as zeros if the input map is numeric.

Sequence field:

Identifying information may be recorded in columns 72 through 80. There is no format for this data, as it is not used by the program and is for the user's convenience in designating specific map decks.

Coding for each row proceeds in this manner until all 120 rows of the left side have been recorded. The coding procedure for the right side of the map is similar, with the data being recorded in the appropriate columns for the right side (Table 2).

Grid sheet realignment:

If a 60 by 60 grid sheet has been used, the grid overlay will have to be repositioned after the first 60 rows of coding. This must be done with care so the top row of the grid, when repositioned, lies precisely where the last row ended. Gaps resulting from improper positioning of the grid may be prevented by marking the position of the last row on the base map. When the grid is moved, exact positioning may be made by re-aligning the grid overlay with the mark of the base map. A similar technique should be used if a 60 column wide grid is used to record both right and left sides of a map.

The shortcut procedure:

If two or more rows of a map are identical, data coding may be reduced by using a shortcut procedure. This is accomplished by using the row continuation field located in columns 6, 7, and 8 for the left side, and columns 66, 67, and 68 for the right side. The first row in a series of identical rows is coded in the normal fashion, with the row number in the row number field. The number of the last row in the series is placed in the row continuation field. Normal coding proceeds with the next row. For example, suppose the left side is being coded and rows 15 through 20 are all identical. The code sheet would appear as follows:

```
column:              1 1 1 1 1 1         7 8
         1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 . . . . 9 0
         -----------------------------------------
             0 1 4       M M M M M M M . . . . 1 4
             0 1 5 0 2 0 N N N N N N N . . . . 1 5
             0 2 1       P P P P P P P . . . . 1 6
```

Running the MAP4B Program

The MAP4B main program is a simple program which subsequently calls (activates) several subprograms, or subroutines. The main program and each of the associated subprograms are discussed in detail below.

The Main Program

The main program reads in 3 important items: (1) the number of maps to be read (MAPNO), (2) a variable (TYPEOP) indicating whether the program should simply display the input data as a map or create a composite map from the input data to be read in, and (3) a variable (ICRD) indicating if optional card output is to be produced. In the event that a compositing process is to be performed, a variable (COMTYP) is read to specify if the compositing is to be additive (indicated by using the word "ADD") or multiplicative (indicated by the word "MULTIPLY").

If a display of data is desired, a card containing the letters "DISPLAY" in columns 1 through 7 is used. If a composite map is to be generated, a similar card containing the letters "COMPOSITE" should be used instead of the "DISPLAY" card. If the program is to be used to display 1 or more maps (rather than creating a composite map), the input sequence for the main program is as follows:

Variable Name	Format	Columns
MAPNO	free	1-2
TYPEOP	A4	1-7[1]
ICRD	free	1-6[2]

[1]must have DISPLAY typed in columns 1-7
[2]must have either CARD or NOCARD typed beginning in column 1

If the program is to be used to generate a composite map, the input sequence for the main program would be as follows:

Variable Name	Format	Columns
MAPNO	free	1-2
TYPEOP	A4	1-9[1]
COMTYP	free	1-7[2]
ICRD	free	1-4[3]
XLIM	free	1-5[4]
NAME	5A4	1-20

[1]must have COMPOSITE typed in columns 1-9
[2]must have ADD or MULTIPLY typed beginning in column 1
[3]must have CARD or NOCARD typed beginning in column 1
[4]must be followed by a decimal

Once the above variables have been read by the main program, various subprograms are activated automatically by the main program. The user, need not be concerned with how these subprograms are activated. The following discussion of the functions of these subprograms (usually called subroutines) is provided so that the user may become familiar wtih the operation and function of the total MAP4B system.

Subroutine ZERO

This subroutine clears all the required registers (storage areas) which are to be used by the main program. This maneuver insures that no "left over" values will be in storage to cause calculation problems. Users with computers that automatically zero out arrays may omit this subroutine; however, leaving it in will not interfere with normal performance of the overall program.

Subroutine READIN

This subroutine is used to read in maps which are to be displayed. A different subroutine is used to read maps that will undergo the compositing procedure. Input sequence for this subroutine consists of (1) the name of the map being displayed, and (2) the actual map data cards. A map name must be specified for each map being read, and a card containing this information must preceed each set of map data. This name may be up to 20 characters long, and may be entered anywhere in the first 20 columns on a card.

Subroutine READ

This subroutine reads in numeric cellular map data for the compositing procedure. Along with the name of the map being read (MAPNAM) the program reads another important variable, MULT, the multiplier (or weight) of the map being read. The variable 'MAPNAM' may be up to 20 characters long (alphabetic or numeric) and must come immediately before the actual cellular map data. The variable 'MULT' comes immediately after the cellular map data and consists of a three-digit number. The format and columnar positions for the subroutine READ are as follows:

Variable Name	Format	Columns
MAPNAM	5A4	1-20
(map data)		
MULT	free	1-3

Subroutine MAPADD

This subroutine performs an additive compositing procedure, and should be used only when the user desires to create a composite map using an ADDITIVE compositing process. This subroutine multiplies the various maps by the appropriate weighting factor and then ADDS these weighted maps together to produce a final composite matrix. No input data are required for this program.

Subroutine MAPMUL

This subroutine is useful for producing composite maps using a multiplicative procedure, and should be used only to produce multiplicative composite maps. In this program, map data ranging in value from zero to nine are standardized from zero to one. Map matrices having these standardized values are then MULTIPLIED to generate a matrix of products. No input data are required for this program, and the input sequence for the main program is the same when using this subroutine. However, the values input for MULT and XLIM are ignored by this program.

Subroutine STRATA

This subroutine categorizes the values of the cells in the composite matrix output from subroutine MAPADD) into one of 10 predetermined symbols. This subroutine also produces the composite map using these symbols. No input data are required by this subroutine.

The categorization procedure was designed to accommodate the situation in which a cell value in the final composite map exceeds the specified upper limit. In the event this happens, cells exceeding the limit value will be displayed as a dollar sign ($) on symbolic maps, or as an asterisk (*) on numeric maps.

Subroutine RULER

This subroutine, which is called up by the previous subroutine, produces a "ruler" at the top and bottom of the output map. This ruler numbers the columns of the map, a feature which is useful in locating specific cells. No input data are required.

Subroutine STATSS

This subroutine is used to produce the legend found at the bottom of each output map. STATSS counts the number of times each symbol appears and calculates the percentage of the entire map frame occupied by each symbol. The upper and lower ranges of the strata represented by each symbol are also produced in the legend. This subroutine requires no input data.

Subroutine CARDS

This subroutine produces card output of a composite map or of a set of input data that is to be displayed as a map. This subroutine is optional and is activated only if the user specifies its use. This subroutine produces MAP4B-compatible card output of original map data or a composite map. Statistical summaries, legends, etc. will not be produced. No user-supplied input is required.

Setting up Card Decks

In order for MAP4B to run properly, it is imperative for the input data to be in the proper format and in the correct order. Input data should be in the following order:

1) the number of maps to be processed (i.e., either added or multiplied to produce a composite map, or to be displayed)
2) the type of operation (i.e., either DISPLAY or COMPOSITE)
3) the type of compositing model to be used (i.e. either additive or multiplicative). To evoke the additive model, input the word ADD; for a multiplicative model, the word MULTIPLY should be entered.
4) the word CARDS or NOCARDS. If optional card output is desired, CARDS should be entered; the word NOCARDS supresses card output.
5) the maximum value limit to be used in the final composite map (i.e., Note: a value of zero for this variable will allow the program to use the maximum possible value that can be attained; a negative value will cause the program to use the maximum realized value as the upper limit; use of a positive value greater than zero will result in the program using the specified value as the upper value limit. -USE WITH COMPOSITE OPERATION ONLY
6) the name of the final composite map (i.e., the variable 'NAME') [USE WITH COMPOSITE OPERATION ONLY]
7) n sets of cellular map data (where n=MAPNO), each consisting of the following:
   a) the name of the map
   b) the coded numerical cellular map data (left and right sides)
   c) the weighting factor (i.e., the variable 'MULT') [USE WITH COMPOSITE OPERATION ONLY].

Below, a sample deck of a run to create a composite map is given. Cards containing similar data are supplied to users with the MAP4B program deck. In this example, two maps are to be added together. The first map, called FACTOR A, has a weighting factor of 2. After being weighted, FACTOR A is added to a second map, FACTOR B, which is weighted by a factor of 1. The input data set is shown below.

```
columns:
           1         2         3 ..... 6         7         8
123456789012345678901234567890 ..... 01234567890123456789O
--------------------------------------------------------------

02                                               CARD 01
COMPOSITE                                        CARD 02
ADD                                              CARD 03
NOCARD                                           CARD 04
0010.                                            CARD 05
COMPOSITE TEST MAP                               CARD 06
FACTOR A                                         CARD 07
  001060111111111111111111111111 ..... 11111111  CARD 08
  061120444444444444444444444444 ..... 44444444  CARD 09
 222222222222222222222222222222 ..... 22 001060  CARD 10
 333333333333333333333333333333 ..... 33 061120  CARD 11
02                                               CARD 12
FACTOR B                                         CARD 13
  001120111111111111111111111111 ..... 11111111  CARD 14
 111111111111111111111111111111 ..... 11 001060  CARD 15
 333333333333333333333333333333 ..... 33 061120  CARD 16
01                                               CARD 17
```

The card number designation in columns 73-80 is optional and is used here for demonstration purposes only. An explantion of each card is provided below.

CARD 01 - number of maps to be read in and added together (2)
CARD 02 - the type of operation. (Here a composite map is to be produced.)
CARD 03 - the compositing model. (Here an additive model was requested.)
CARD 04 - card output option indicator. (Card output was not requested.)
CARD 05 - the maximum value of any cell in the final composite map. This is usually calculated manually by taking the sum of the products of the maximum value for each map and its appropriate weighting factor.
CARD 06 - name of the composite map. (Here it is COMPOSITE TEST MAP.)
CARD 07 - name of the first factor map to be read into the program.
CARD 08 - cellular data for the left side of the first map. Note that the shortcut procedure was used and that the first 60 rows of the map are identical.
CARD 09 - rows 61-120 of the left side of the first map.

CARD 10 - right side data for first map, with rows 1-60 being identical
CARD 11 - rows 61-120 of the right side of the first factor map
CARD 12 - multiplier value for first factor map
CARD 13 - name of second factor map
CARD 14 - left side (rows 1-120) of second factor map
CARD 15 - first 60 rows of right side of second factor map
CARD 16 - rows 61-120 of right side of second factor map
CARD 17 - multiplier for second factor map

It should be noted that these 2 maps (Factor A and Factor B) could have been displayed rather than being composited. If the objective had been to display these maps, a similar card deck would have been required. However, to display these 2 maps Card 02 would have been different (DISPLAY rather than COMPOSITE), and Cards 03, 04, 12, and 17 would have been excluded.

Batch Processing Procedure

At this point, the potential MAP4B user should be familiar with the program and the format for the input data. Users should obtain the details for FORTRAN batch processing from their respective computer centers if batch processing is to be used. Users planning to use MAP4B interactively, such as with the Conversational Monitor System supported on some IBM computers, should consult their respective computer centers for instructions on using their computer interactively.

Auxiliary Programs

A series of auxiliary programs has been developed to expand the utility of the MAP4B program. These programs have been designed to make specific alterations in input data. They are to be run as independent programs to produce altered map data for subsequent processing by MAP4B.

The auxiliary programs may be used to (1) change given map symbols to other symbols, (2) create up to 5 loci of specified radii around a specific cell (or cells) in a map, (3) determine the number of different map symbols encountered within a given search distance from a specified cell, and (4) calculate slope and aspect from elevation data. A brief description of each of these programs is given below.

Program CONVERT

This program is useful for altering symbols on maps and may be used with symbol maps or numeric maps. This program has been designed to be run as an independent program to alter map input data. Input data are in the same format as for the MAP4B program, EXCEPT THAT ONLY THE 'MAP NAME' CARD AND THE MAP DATA CARDS ARE USED. Data cards containing multiplier values, etc., should NOT be used.

This program was designed to change a given map symbol(s) to another symbol. However, it must be stated that all instances of the symbol(s) to be altered will be converted. Selected occurrences of a particular symbol cannot be isolated from other occurrences of the same symbol(s). Alteration of selected symbols at specific locations within the map frame should be performed manually.

Converting a symbol is accomplished by altering the sequence of symbols in the DATA statement in the source program. More than one symbol may be altered within the same run of the program, however all instances of the specified symbols will be converted to the new 'target' symbols. Further instructions for this procedure are given in the source program and user users of the CONVERT program should review a source program listing prior to using this program.

Output for this program is in the form of punched cards (or in card images, if output is returned to a remote terminal). These output cards will be in the proper format for input into the MAP4B program.

Program LOCUS

This program was designed to define a locus of a specified radius about a given cell or series of cells. In other words, if a particular cell in a map frame is specified, along with a radius, then this program will create a circle around that cell. All cells falling within the specified radius will be given a value specified by the program user. Up to 5 different loci, each with a different radius, may be defined with this program. Each of these loci may be displayed with different symbols, which may be any single-digit letter, number, or special character.

This program is especially useful in delineating the shape of a zone surrounding a single cell or group of cells. In this program, a map is scanned for those cells needing a locus definition (i.e., those cells coded with a special "flag" character). This character may be any symbol or a single digit integer value. Once a "flag" is encountered, a circle of a specified radius is described about that cell. This process is then iterated until all cells having "flag" values have been encountered.

A typical example of a use of this program would be to delineate all cells located within a given distance of some mapped feature such as rivers. By using the character for rivers as the "flag" character, a zone having a specified radius may be defined. However, by specifying radii of different lengths, zones of varying distances can be constructed with the program. For example, using radii of 50, 40, 30, and 15 cells, (along with different values for each zone), the program may be used to create concentric zones around a specific cell. When specifying multiple radii, it is imperative to begin with the longest radius (i.e., the outer circle) and then specify radii of decreasing size. The characters used to designate the respective loci formed with these radii should be given in the order corresponding to the radii.

A special feature of this program is the ability to eliminate certain portions of the map frame. That is, a silhouette of the area of interest may be read in as a map, and used as a template with which to mask out all the area outside the silhouetted area. Masked areas are assigned a blank symbol. If no masking is desired, a pair of cards having any character other than a blank in the map data fields should be used. This pair should have "001120" in the row and row continuation fields for both the right and left sides. Also, a background value (or symbol) may be specified to designate all area outside the locus but within the silhouetted area. This background symbol may be any single-digit letter, number, or special character.

Necessary input for this program includes (1) the background symbol, (2) the number of radii to be used to define loci, (3) the radii of the loci to be defined, (4) the special character designated to denote a cell needing a locus, (5) the symbol(s) or value(s) (single digit integer) to be used to fill the circle or locus, (6) the name of the map, (7) the map data containing the cells requiring a locus, and (8) map data of the silhouette of the area of interest. Instructions for using this program, along with necessary format, are given in the source program listing. WARNING: The value or symbol used to fill the circle (i.e., used as the input character for the variable FILL) must not be the same character used as the "FLAG" character. Output from the LOCUS program is in the form of MAP4B compatible cards. Instructions for using this program may be found in the program listing.

Program SEARCH

A third special-purpose, auxiliary program is SEARCH. This program was designed to perform a 4-way search and to determine how many times cells having different symbols or values from the origin cell were found. That is, beginning with an original cell, the program searches upward, downward, and to the left and right of the origin. When the search is complete in all four directions, the number of different symbols (not the total number of symbols) encountered is determined, and the value of the origin cell is reassigned that value. In its current configuration, this program is set up to perform searches from every cell in a map. Primary utility of this program is to generate a map of interspersion (i.e., the amount of patchiness) indices.

Input for this program consists of (1) the value for the search radius (expressed in units of cell height), (2) the map name, and (3) the cards containing the map data. Output is in the form of punched cards in format appropriate for MAP4B. Further instructions for using this program are given in the source program listing.

Program SLOPE

Another special-purpose program is known as SLOPE. SLOPE may be used to produce maps of slope (steepness of terrain), or aspect (the compass direction one faces when looking in the downhill direction) for any cell in a map. However, to use this program, one must have reasonably accurate elevation data for the study area. This data may be encoded manually, like other thematic data, or may be generated from commercially available sources. However, it should be mentioned that elevation data may not be coded as single digit integer values, thus, input data format varies considerably between this program and the other programs. However, regardless of the input format, this program will accept data input having extensive field lengths. Format specifications for reading elevation data must by supplied by the user to comply with the specific structure of the user's data. Regardless of the format, proper performance of this program depends on the elevation data being in a 120 by 120 matrix which should precisely overlay the area of interest. Elevation data may be in feet or meters; a minor program alteration will allow use of either measure.

The Department of Fisheries and Wildlife Sciences has appropriate computer programs for deriving elevation data at a variety of resolutions from elevation data supplied by the United States Geological Survey. However, these programs are designed specifically for the Virginia Tech computer, and may not perform properly on other machines. If users cannot obtain elevation data from any other source, arrangements can be made with the source above.

As with the other auxiliary programs, input data consists of (1) the map name and (2) the map data. Currently, output from this program is in the form of cards compatible with the MAP4B program. Both slope and aspect are internally calculated in terms of degrees. However, in the process of producing cards, values of slope and aspect are stratified into single-digit integer categories. These categories are as follows:

Slope (degrees)	Symbol
0 - 5	1
6 - 10	2
11 - 15	3
16 - 20	4
21 - 25	5
26 - 30	6
31 - 35	7
36 - 40	8
41 - 90	9

Aspect (degrees)	Symbol
0 - 45	1
46 - 90	3
91 - 135	5
136 - 180	7
181 - 225	9
226 - 270	6
271 - 315	4
315 - 359	2

Output from this program is in the form of a deck of MAP4B-compatible cards. Further instructions for using this program may be found in the source program listing.

Appendix B

The MAP4B Mapping System and Auxiliary Programs

```
C       ***********************************************************
C       ***       MAIN PROGRAM FOR PRODUCING COMPUTER MAPS        ***
C       ***********************************************************
C       *****     MAIN PROGRAM INPUT SEQUENCE:                  *****
C       ***     VARIABLE                  FORMAT    COLUMNS       ***
C       ***     NUMBER OF MAPS (MAPNO)     FREE        1-2        ***
C       ***     OPERATION TYPE (TYPEOP)     A1         1-7  (A)   ***
C       ***     COMPOSITING MODEL (COMTYP)  A1         1-10 (B)   ***
C       ***     CARD OUTPUT OPTION (ICRD) FREE         1-2  (C)   ***
C       ***     UPPER VALUE LIMIT (XLIM)  FREE         1-5        ***
C       ***     FINAL MAP NAME (NAME)      5A4         1-20       ***
C       *****     FOR EACH MAP THE INPUT SEQUENCE IS:           *****
C       ***     MAP NAME (MAPNAM)          5A4         1-20       ***
C       ***     (MAP DATA)                                        ***
C       ***     MAP MULTIPLIER (MULT)      FREE        1-2        ***
C       ***                                                       ***
C       ***  (A): THE WORDS "DISPLAY" OR "COMPOSITE" MUST APPEAR ***
C       ***       BEGINNING IN COLUMN 1. (THE PROGRAM READS ONLY ***
C       ***       THE FIRST COLUMN)                               ***
C       ***  (B): THE WORDS "MULTIPLY" OR "ADD" MUST APPEAR       ***
C       ***       BEGINNING IN COLUMN 1. (THE PROGRAM READS ONLY ***
C       ***       THE FIRST COLUMN)                               ***
C       ***  (C): THE WORD "CARDS" MUST APPEAR BEGINNING IN       ***
C       ***       COLUMN IF CARD OUTPUT IS DESIRED. ENTER THE     ***
C       ***       WORD "NOCARDS" IF CARD OUTPUT IS NOT DESIRED.  ***
C       ***********************************************************
      DIMENSION ZLEFT (120,60), ZRIGHT (120,60)
      INTEGER*2 XLEFT (120,60), XRIGHT (120,60)
      INTEGER*2 YLEFT (120,60), YRIGHT (120,60)
      INTEGER*2 COMTYP, MUL/'M'/, TYPEOP, DIS/'D'/, CHAR(11)
      INTEGER*2 ICRD, CARD/'C'/
      INTEGER NAME(5), MAPNAM(5), MULTR(20), MAPLST(20,5)
      DATA NAME/'(NOT',' APP','LICA','BLE)','    '/
C
C      *** SYMBOL ARRAY IS SPECIFIED IN THE STMNT BELOW ***
      DATA CHAR/' ','.',',','-','=','+','/','O','%','*','$'/
C
C      *** NUMERIC SYMBOLS MAY BE GENERATED BY SUBSTITUTING ***
C      *** THE STATEMENT BELOW FOR THE PREVIOUS DATA STMNT  ***
C     DATA CHAR/' ','1','2','3','4','5','6','7','8','9'/
      IFLAG=0
C
C      ** READ THE NUMBER OF MAPS TO BE READ IN **
```

```
      READ (4,*) MAPNO
C
C       ** READ TYPE OF OPERATION TO BE USED **
      READ (4,01) TYPEOP
01    FORMAT(A1)
C
C       ** IF DISPLAY MODE, ENTER SECTION BELOW; IF NOT SKIP **
      IF (TYPEOP.EQ.DIS) GO TO 05
      GO TO 111
C
C       ** DISPLAY ONLY SECTION **
C
C       ** READ IN A MAP **
05    DO 09 IDISPL = 1, MAPNO
      CALL READIN (XLEFT,XRIGHT,IFLAG,MAPNAM)
      IF (IFLAG.NE.0) GO TO 100
C
C       ** COMPILE JOB SUMMARY INFORMATION **
      MARGIN = 0
      MULTR(IDISPL) = 0
      DO 06 JDISPL=1,5
      MAPLST(IDISPL,JDISPL) = MAPNAM(JDISPL)
06    CONTINUE
C
C       ** DISPLAY THE MAP **
      WRITE(6,04) MAPNAM
04    FORMAT(1H1,'MAP NAME: ',5A4)
      CALL RULER
      DO 08 IR = 1,120
      MARGIN = MARGIN + 1
      WRITE(6,03) MARGIN, (XLEFT(IR,IC),IC=1,60), (XRIGHT(IR,IC),IC=1,60
     &), MARGIN
03    FORMAT(1X,I3,1X,120A1,1X,I3)
08    CONTINUE
      CALL RULER
09    CONTINUE
      GO TO 900
C
C       ** READ TYPE OF COMPOSITING TO PERFORM **
111   READ (4,11) COMTYP
11    FORMAT (A1)
C
C       ** READ CARD OUTPUT OPTION **
      READ(4,12) ICRD
12    FORMAT(A1)
C
C       ** COMPOSITE SECTION **
C
C       ** READ MAXIMUM POSSIBLE VALUE FOR A CELL IN THE COMPOSITE MAP **
10    READ (4,*) XLIM
```

```
C
C      ** READ THE NAME OF THE COMPOSITE MAP **
      READ(4,13) NAME
13    FORMAT(5A4)
C
C      *** MAIN PROGRAM LOOP ***
      MAPMAX = 0
      SUMAX1 = 0.0
      SUMAX2 = 0.0
C
      DO 20 IRUN = 1,MAPNO
C
C      ** READ IN A MAP **
      CALL READ (XLEFT,XRIGHT,MULT,IFLAG,MAPNAM)
      IF (IFLAG.NE.0) GO TO 100
C
C      ** SCAN A MAP TO DETERMINE MAXIMUM OBSERVED VALUE **
      MAPMAX = 0
      CALL SCAN (XLEFT,XRIGHT,MAPMAX)
C
C      ** COMPILE JOB SUMMARY INFORMATION **
      DO 15 J = 1,5
15    MAPLST(IRUN,J) = MAPNAM(J)
      MULTR(IRUN) = MULT
C
C      ** COMPOSITING PROCEDURE **
C
C      ** DETERMINE WHICH COMPOSITING PROCEDURE TO USE **
      IF (COMTYP.EQ.MUL) GO TO 17
C
C      ** THE FOLLOWING STMNT EVOKES THE LINEAR ADDITIVE MODEL **
16    CALL MAPADD (ZLEFT,ZRIGHT,XLEFT,XRIGHT,MULT,IRUN,SUMAX1,SUMAX2,MAP
     &MAX)
      GO TO 20
C
C      ** THE FOLLOWING STMNT EVOKES THE MULTIPLICATIVE MODEL **
17    CALL MAPMUL (ZLEFT,ZRIGHT,XLEFT,XRIGHT,XLIM,IRUN,SUMAX1,SUMAX2,MAP
     &MAX)
C
20    CONTINUE
C
C      ** DETERMINE WHICH LIMITING VALUE TO USE **
      IF (XLIM.EQ.0.) XLIM = SUMAX1
      IF (XLIM.LT.0.) XLIM = SUMAX2
      IF (XLIM.GT.0.) XLIM = XLIM
C
C      ** CATEGORIZE CELL VALUES INTO STRATA AND PRODUCE MAP **
      CALL STRATA (XLIM,ZLEFT,ZRIGHT,CHAR,NAME,YLEFT,YRIGHT)
C
C      ** GENERATE THE LEGEND AND STATISTICS FOR COMPOSITE MAP **
```

```
      CALL STATSS (YLEFT,YRIGHT,XLIM,CHAR)
C
C      ** DETERMINE IF CARDS ARE TO BE PRODUCED FOR COMPOSITE MAP **
      IF (ICRD.EQ.CARD) GO TO 30
      GO TO 900
30    CALL CARDS (YLEFT,YRIGHT,NAME)
C
C      ** JOB SUMMARY SECTION **
900   WRITE(6,1000)
1000  FORMAT(1H1,'MAP4B JOB SUMMARY INFORMATION')
      WRITE(6,1001) MAPNO
1001  FORMAT(1X,I3,' MAPS WERE READ IN FOR PROCESSING')
      WRITE(6,1002)
1002  FORMAT(1X,'THE NAMES OF THE MAPS READ IN WERE AS FOLLOWS:')
      DO 1111 ICNT = 1,MAPNO
      WRITE(6,1003) ICNT, (MAPLST(ICNT,J),J=1,5), MULTR(ICNT)
1003  FORMAT(1X,I3,')',2X,5A4,4X,'MULTIPLIER =',I3)
1111  CONTINUE
      WRITE(6,1004) NAME
1004  FORMAT(1X,'NAME OF COMPOSITE MAP WAS: ',5A4)
      IF (TYPEOP.EQ.DIS) GO TO 1008
      WRITE(6,1005) XLIM
1005  FORMAT(1X,'MAXIMUM STATED VALUE FOR A CELL IN A COMPOSITE MAP WAS:
     %',1X,F10.5)
      IF(COMTYP.EQ.MUL) GO TO 1006
      GO TO 1007
1006  WRITE(6,1009)
1009  FORMAT(1X,'THE MULTIPLICATIVE MODEL WAS USED TO COMPOSITE')
      GO TO 1008
1007  WRITE(6,1011)
1011  FORMAT(1X,'THE LINEAR ADDITIVE MODEL WAS USED TO COMPOSITE')
1008  WRITE(6,1010)
1010  FORMAT(1X,'NORMAL TERMINATION OF MAP4B')
C
      GO TO 999
C      ** ERROR SECTION **
100   WRITE(6,101)
101   FORMAT(1X,'MAIN PROGRAM ABORTED DUE TO INPUT DATA ERROR')
C
999   STOP
      END
      SUBROUTINE READ (XLEFT,XRIGHT,MULT,IFLAG,MAPNAM)
C      ***********************************************************
C      ***     THIS SUBROUTINE READS IN AND CHECKS MAPS         ***
C      ***      INPUT -  1) MAP NAME,  COLS. 1-20               ***
C      ***               2) (MAP DATA)                          ***
C      ***               3) MULTIPLIER, COLS. 1-2               ***
C      ***********************************************************
C      *** NOTE: LESS THAN 240 MAP CARDS CAN BE READ USING      ***
C      ***              THE SHORTCUT PROCEDURE                  ***
```

```
C        *********************************************************
         INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
         INTEGER MAPNAM(5), RNUMBR, R2, CHEKL, CHEKR
C
C        ** READ THE MAP NAME **
         READ(4,111) MAPNAM
111      FORMAT(5A4)
C        *****LEFT SIDE READ LOOP*****
         L=1
         CHEKL=1
101      READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01       FORMAT(2X,I3,I3,60I1)
         IF(CHEKL.NE.LNUMBR) GO TO 991
         IF(L2.EQ.0) GO TO 100
         N=L2-LNUMBR
         DO 99 J=1,N
         L=L+1
         CHEKL=CHEKL+1
         BACKSPACE 4
99       READ(4,011) (XLEFT(L,K),K=1,60)
011      FORMAT(8X,60I1)
100      L=L+1
         CHEKL=CHEKL+1
         IF(L.GT.120) GO TO 102
         GO TO 101
C        *****RIGHT SIDE READ LOOP*****
102      L=1
         CHEKR=1
103      READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02       FORMAT(1X,60I1,1X,I3,I3)
         IF(CHEKR.NE.RNUMBR) GO TO 992
         IF(R2.EQ.0) GO TO 200
         N=R2-RNUMBR
         DO 199 J=1,N
         L=L+1
         CHEKR=CHEKR+1
         BACKSPACE 4
199      READ(4,022)(XRIGHT(L,K),K=1,60)
022      FORMAT(1X,60I1)
200      L=L+1
         CHEKR=CHEKR+1
         IF(L.GT.120) GO TO 201
         GO TO 103
201      CONTINUE
C        ** READ WEIGHTING FACTOR **
15       READ(4,*) MULT
C
C        ** ERROR SECTION **
         GO TO 999
991      WRITE (6,901) CHEKL, MAPNAM
```

```
      IFLAG=10
      GO TO 999
992   WRITE (6,902) CHEKR, MAPNAM
      IFLAG=10
901   FORMAT(1H ,'ERROR: SUBROUTINE READ ABORTED. CARD ',I3,' ON LEFT OU
     &T OF ORDER ON MAP NAME ',5A4)
902   FORMAT(1H ,'ERROR: SUBROUTINE READ ABORTED, CARD ',I3,' ON RIGHT O
     &UT OF ORDER ON MAP NAME ',5A4)
999   CONTINUE
      RETURN
      END
      SUBROUTINE READIN (XLEFT,XRIGHT,IFLAG,MAPNAM)
C      * * * * * * * * * * * * * * * * * * * * * * * *
C      ***       SUBROUTINE NAME:  READIN          ***
C      *** THIS SUBROUTINE READS IN AND CHECKS     ***
C      ***       SYMBOL OR NUMERIC MAPS            ***
C      *** INPUT SEQUENCE:                         ***
C      ***         1) MAP NAME,  COLS. 1-20        ***
C      ***         2) (MAP DATA)                   ***
C      * * * * * * * * * * * * * * * * * * * * * * * *
C      *** NOTE: LESS THAN 240 MAP CARDS CAN BE    ***
C      ***        READ USING THE SHORTCUT PROCEDURE ***
C      * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      INTEGER MAPNAM(5), RNUMBR, R2, CHEKL, CHEKR
C
C      ** READ THE MAP NAME **
      READ(4,111) MAPNAM
111   FORMAT(5A4)
C      *****LEFT SIDE READ LOOP*****
      L=1
      CHEKL=1
101   READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01    FORMAT(2X,I3,I3,60A1)
      IF(CHEKL.NE.LNUMBR) GO TO 991
      IF(L2.EQ.0) GO TO 100
      N=L2-LNUMBR
      DO 99 J=1,N
      L=L+1
      CHEKL=CHEKL+1
      BACKSPACE 4
99    READ(4,011) (XLEFT(L,K),K=1,60)
011   FORMAT(8X,60A1)
100   L=L+1
      CHEKL=CHEKL+1
      IF(L.GT.120) GO TO 102
      GO TO 101
C      *****RIGHT SIDE READ LOOP*****
102   L=1
      CHEKR=1
```

```
103   READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02    FORMAT(1X,60A1,1X,I3,I3)
      IF(CHEKR.NE.RNUMBR) GO TO 992
      IF(R2.EQ.0) GO TO 200
      N=R2-RNUMBR
      DO 199 J=1,N
      L=L+1
      CHEKR=CHEKR+1
      BACKSPACE 4
199   READ(4,022)(XRIGHT(L,K),K=1,60)
022   FORMAT(1X,60A1)
200   L=L+1
      CHEKR=CHEKR+1
      IF(L.GT.120) GO TO 201
      GO TO 103
201   CONTINUE
      GO TO 999
991   WRITE (6,901) CHEKL, MAPNAM
      IFLAG=10
      GO TO 999
992   WRITE (6,902) CHEKR, MAPNAM
      IFLAG=10
901   FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED. CARD ',I3,' ON LEFT
     &OUT OF ORDER ON MAP NAME: ',5A4)
902   FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED, CARD ',I3,' ON RIGHT
     & OUT OF ORDER ON MAP NAME: ',5A4)
999   CONTINUE
      RETURN
      END
      SUBROUTINE MAPADD (ZLEFT,ZRIGHT,XLEFT,XRIGHT,MULT,IRUN,SUMAX1,SUMA
     &X2,MAPMAX)
C       * * * * * * * * * * * * * * * * * * * * * * * * * * *
C       ***             SUBROUTINE NAME:  MAPADD           ***
C       ***      THIS SUBROUTINE ADDS THE MAP MATRICES     ***
C       ***           INPUT REQUIRED: NONE                 ***
C       * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION ZLEFT (120,60), ZRIGHT (120,60)
      INTEGER*2 XLEFT (120,60), XRIGHT (120,60)
C
      XVAL = 0.0
      IF (IRUN.EQ.1) SUMAX1 = 0.
      IF (IRUN.EQ.1) SUMAX2 = 0.
      SUMAX1 = SUMAX1 + (9*MULT)
      SUMAX2 = SUMAX2 + (MAPMAX*MULT)
      IF (IRUN.EQ.1) GO TO 05
      GO TO 06
05    CALL ZERO (ZLEFT,ZRIGHT,XVAL)
06    DO 10 L=1,120
      DO 10 K=1,60
      ZLEFT(L,K)= ZLEFT(L,K) + (XLEFT(L,K)*MULT)
```

```
      ZRIGHT(L,K)= ZRIGHT(L,K) + (XRIGHT(L,K)*MULT)
10    CONTINUE
      RETURN
      END
      SUBROUTINE MAPMUL (ZLEFT,ZRIGHT,XLEFT,XRIGHT,XLIM,IRUN,SUMAX1,SUMA
     &X2,MAPMAX)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***            SUBROUTINE NAME:  MAPMUL              ***
C      ***  THIS SUBROUTINE MULTIPLIES THE MAP MATRICES     ***
C      ***             INPUT REQUIRED:  NONE                ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION ZLEFT(120,60), ZRIGHT(120,60)
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      REAL MULT
C
C      ** FORCE ALL VALUES TO A ZERO TO ONE SCALE **
      MULT = 0.1111111
      XVAL = 1.0
      IF (IRUN.EQ.1) SUMAX2 = 1.0
      SUMAX1 = 1.0
      SUMAX2 = SUMAX2 * (MAPMAX*MULT)
      IF (IRUN.EQ.1) GO TO 05
      GO TO 06
05    CALL ZERO (ZLEFT,ZRIGHT,XVAL)
06    DO 10 L=1,120
      DO 10 K=1,60
      XL = XLEFT(L,K)
      XR = XRIGHT(L,K)
      ZLEFT(L,K)= ZLEFT(L,K) * (XL*MULT)
      ZRIGHT(L,K)= ZRIGHT(L,K) * (XR*MULT)
10    CONTINUE
      RETURN
      END
      SUBROUTINE STRATA (XLIM,ZLEFT,ZRIGHT,CHAR,NAME,YLEFT,YRIGHT)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***             SUBROUTINE NAME:  STRATA                 ***
C      ***  THIS SUBROUTINE AUTOSCALES NUMERICAL MAP VALUES     ***
C      ***         INTO 10 USER-DEFINED CATEGORIES              ***
C      ***    THIS SUBROUTINE ALSO PRODUCES A SYMBOL MAP.       ***
C      ***              INPUT REQUIRED: NONE                    ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION  ZLEFT(120,60), ZRIGHT(120,60)
      INTEGER*2 YLEFT (120,60), YRIGHT(120,60), CHAR(11)
      INTEGER NAME(5)
C
      DVAL=XLIM/10.
      DO 30 L=1,120
      DO 30 K=1,60
      I=(ZLEFT(L,K)/DVAL)+1
      YLEFT(L,K)=CHAR(I)
```

```
      I=(ZRIGHT(L,K)/DVAL)+1
30    YRIGHT(L,K)=CHAR(I)
      WRITE(6,50) NAME
50    FORMAT(1H1,10X,'MAP NAME : ',5A4)
      MARGIN = 0
      CALL RULER
      DO 55 L=1,120
      MARGIN = MARGIN + 1
55    WRITE(6,51) MARGIN,(YLEFT(L,K),K=1,60),(YRIGHT(L,K),K=1,60),MARGIN
51    FORMAT(1H ,I3,1X,120A1,1X,I3)
      CALL RULER
      RETURN
      END
      SUBROUTINE STATSS (YLEFT,YRIGHT,XLIM,CHAR)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***                 SUBROUTINE NAME: STATSS                  ***
C      *** THIS SUBROUTINE COUNTS THE FREQUENCY OF EACH MAP SYMBOL ***
C      ***  THIS ROUTINE IS TO BE USED WITH SYMBOL TYPE MAPS ONLY  ***
C      ***                  INPUT REQUIRED:  NONE                   ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION PCT(10), FREQ(10)
      INTEGER*2 CHAR(11), YLEFT(120,60), YRIGHT(120,60)
C
      DATA FREQ/10*0./
      HIGH=XLIM/10.
      ZHIGH=HIGH
      XLOW=0.
      DO 70 I=1,10
      DO 60 L=1,120
      DO 60 K=1,60
      IF(YLEFT(L,K).EQ.CHAR(I)) FREQ(I)=FREQ(I)+1.
      IF(YRIGHT(L,K).EQ.CHAR(I)) FREQ(I)=FREQ(I)+1.
60    CONTINUE
70    PCT(I)=FREQ(I)/144.
      DO 71 M=1,5
71    WRITE(6,61)
61    FORMAT(1H )
C     NOTICE: USE THE SCALE BELOW ONLY FOR USGS QUAD SHEETS
      WRITE(6,611)
      WRITE(6,612)
      WRITE(6,613)
611   FORMAT(1X,T31,'I---------------------------------I----------------
     &-----------------I')
612   FORMAT(1X,T31,'0',T65,'2',T99,'4')
613   FORMAT(1H ,T60,'STATUTE MILES')
      WRITE(6,62)
62    FORMAT(1X)
90    WRITE(6,620)
620   FORMAT(1H ,'SYMBOL',5X,'LOWER LIMIT',5X,'UPPER LIMIT',5X,'COUNT',
     &5X,'FREQUENCY')
```

```
      WRITE(6,621)
621   FORMAT(1H ,T15,'(.GE.)',T32,'(.LT.)')
      DO 65 I=1,10
      IF(FREQ(I).EQ.0) GO TO 641
      WRITE (6,61)
      WRITE(6,63) CHAR(I), CHAR(I), CHAR(I)
63    FORMAT(1H ,T3,3A1)
      WRITE(6,64) CHAR(I),CHAR(I),CHAR(I),XLOW,ZHIGH,FREQ(I),PCT(I)
64    FORMAT(1H ,T3,3A1,7X,F9.2,8X,F9.2,5X,F6.0,5X,F6.2,1X,'%')
      WRITE(6,63) CHAR(I), CHAR(I), CHAR(I)
641   XLOW=XLOW+HIGH
65    ZHIGH=HIGH+ZHIGH
      RETURN
      END
      SUBROUTINE ZERO (ZLEFT,ZRIGHT,XVAL)
C      * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***             SUBROUTINE NAME: ZERO            ***
C      ***  THIS SUBROUTINE INITIALIZES THE 'Z MATRIX'  ***
C      ***       ON THE FIRST PASS IN THE PROGRAM       ***
C      ***             INPUT REQUIRED:  NONE            ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION ZLEFT (120,60), ZRIGHT (120,60)
      DO 10 L=1,120
      DO 10 K=1,60
      ZLEFT(L,K) = XVAL
      ZRIGHT(L,K) = XVAL
10    CONTINUE
      RETURN
      END
      SUBROUTINE RULER
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***            SUBROUTINE NAME:  RULER               ***
C      *** THIS SUBROUTINE PUTS A RULER AT THE TOP AND/OR   ***
C      ***     BOTTOM OF A MAP, BEGINNING IN COLUMN SIX     ***
C      ***             INPUT REQUIRED:  NONE                ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      WRITE (6,21)
21    FORMAT(1X,T105,'1',T115,'1',T125,'1')
      WRITE (6,22)
22    FORMAT(1X,T15,'1',T25,'2',T35,'3',T45,'4',T55,'5',T65,'6',T75,'7',
     &T85,'8',T95,'9',T105,'0',T115,'1',T125,'2')
      WRITE (6,23)
23    FORMAT(1X,T6,'12345678901234567890123456789012345678901234567890',
     &'12345678901234567890123456789012345678901234567890123',
     &'4567890')
      RETURN
      END
      SUBROUTINE SCAN (XLEFT,XRIGHT,MAPMAX)
C      **********************************************
C      ***          SUBROUTINE NAME:  SCAN          ***
```

```
C       ***  THIS SUBROUTINE SCANS A SET OF MAP DATA   ***
C       ***  TO DETERMINE THE LARGEST INTEGER VALUE    ***
C       ***          INPUT REQUIRED:  NONE             ***
C       *************************************************
C
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
C
      DO 10 ISCAN = 1,120
      DO 10 JSCAN = 1,60
      IF(XLEFT(ISCAN,JSCAN).GT.MAPMAX) MAPMAX = XLEFT(ISCAN,JSCAN)
      IF(XRIGHT(ISCAN,JSCAN).GT.MAPMAX) MAPMAX = XRIGHT(ISCAN,JSCAN)
10    CONTINUE
      RETURN
      END
      SUBROUTINE CARDS (YLEFT,YRIGHT,NAME)
C       *******************************************
C       ***       SUBROUTINE NAME: CARDS        ***
C       *** THIS SUBROUTINE PRODUCES A DECK OF  ***
C       *** CARDS OF THE OUTPUT DATA FROM MAP4B ***
C       ***        INPUT REQUIRED: NONE         ***
C       *******************************************
      INTEGER*2 YLEFT(120,60), YRIGHT(120,60)
      INTEGER NAME(5)
C
      WRITE(7,100)
100   FORMAT(1X,'CARD OUTPUT FROM MAP4B')
      WRITE(7,110) NAME
110   FORMAT(1X,'MAP NAME:',5A4)
C
C       ** LEFT SIDE CARD OUTPUT LOOP **
      MARGIN = 0
      DO 10 L = 1,120
      MARGIN = MARGIN + 1
      WRITE(7,120) MARGIN, (YLEFT(L,K),K=1,60)
120   FORMAT(2X,I3,3X,60A1)
10    CONTINUE
C
C       ** RIGHT SIDE CARD OUTPUT LOOP **
      MARGIN = 0
      DO 20 L = 1,120
      MARGIN = MARGIN + 1
      WRITE(7,130) (YRIGHT(L,K),K=1,60), MARGIN
130   FORMAT(1X,60A1,1X,I3)
20    CONTINUE
      RETURN
      END
```

```
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     *              PROGRAM NAME:  LOCUS                                 *
C     *  THIS IS A SUBROUTINE TO CREATE AND FILL A CIRCLE AROUND ANY      *
C     *  GIVEN CELL (AS SPECIFIED BY THE MAIN PROGRAM).  UP TO THREE      *
C     *  CONCENTRIC ZONES MY BE CREATED AROUND POINTS OR LINEAR           *
C     *  FEATURES.   TO DO THIS, INPUT THE RADII IN ORDER OF LARGEST      *
C     *  TO SMALLEST.  FILL VALUES ARE INPUT IN THE ORDER THAT THEY       *
C     *  CORRESPOND TO THE SPECIFIED RADII.                               *
C     *  CELLS OUTSIDE THE AREA OF INTEREST MAY BE SET TO BLANKS BY       *
C     *  READING IN A SILHOUETTE MAP (I.E. A SECOND SET OF MAP CARDS)     *
C     *  WARNING !! DO NOT USE A FILL VALUE EQUAL TO THE SYMBOL USED      *
C     *            AS THE FLAG VALUE !! (I.E., VARIABLE 2 BELOW)          *
C     *       INPUT SEQUENCE:                                             *
C     *            1) SYMBOL TO BE USED AS BACKGROUND (COL. 1)            *
C     *            2) NUMBER OF RADII TO BE USED (FREE FORMAT)            *
C     *            3) RADII (IN UNITS OF CELL HEIGHT) (FREE FORMAT)       *
C     *            4) CHARACTER USED TO DENOTE CELLS REQUIRING LOCUS      *
C     *               (COL. 1)                                            *
C     *            5) CHARACTER(S) USED TO FILL THE LOCUS                 *
C     *               (COLS. 1, 3, 5, 7, & 9)                             *
C     *            6) MAP NAME (SEE SUBROUTINE READIN FOR FORMAT)         *
C     *            7) MAP DATA (SEE SUBROUTINE READIN FOR FORMAT)         *
C     *            8) MAP NAME OF FINAL MAP (COLS. 1-20)                  *
C     *            9) MAP DATA OF SILHOUETTE (SAME AS FOR (7))            *
C     *       OUTPUT: A MAP4B-COMPATIBLE CARD DECK CONTAINING:            *
C     *            1) THE CHARACTER(S) USED TO FILL THE LOCUS             *
C     *            2) CHARACTER USED TO DENOTE CELLS REQUIRING LOCUS      *
C     *            3) VALUES OF RADIUS                                    *
C     *            4) MAP NAME                                            *
C     *            5) MAP DATA (LEFT HALF)                                *
C     *            6) MAP DATA (RIGHT HALF)                               *
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 X(120,120), Z(120,120), XLEFT(120,60), XRIGHT(120,60)
      INTEGER*2 PROFIL(120,120)
      INTEGER MAPNAM(5), RADIUS(5)
      INTEGER*2 FLAG, FILL(5), BLANK/' '/, ZERO/'0'/
C
C      ** READ A BACKGROUND VALUE FOR CELLS OUTSIDE LARGEST LOCUS **
      READ(4,111) IBKGND
C
C      ** CLEAR THE Z ARRAY AND SET THE BACKGROUND **
      DO 1111 ICLX=1,120
      DO 1111 ICLY=1,120
      Z(ICLX,ICLY) = IBKGND
1111  CONTINUE
C
C      ** READ THE NUMBER OF RADII TO BE USED **
      READ(4,*) NUMRAD
C
C      **  READ VALUE(S) FOR RADIUS **
```

```
      READ (4,*)(RADIUS(I),I=1,NUMRAD)
C
C     ** READ THE FLAG SYMBOL **
      READ (4,111) FLAG
111   FORMAT(A1)
C
C     ** READ SYMBOL(S) TO BE USED TO FILL THE LOCUS **
      READ (4,112)(FILL(J),J=1,NUMRAD)
112   FORMAT (5(A1,1X))
C
C      ** READ A SET OF MAP CARDS **
      IFLAG = 0
      CALL READIN (XLEFT,XRIGHT,MAPNAM,IFLAG)
      IF(IFLAG.NE.0) GO TO 997
C
C      ** FORM A 120 BY 120 ARRAY FROM THE INPUT DATA **
      DO 01 IA = 1,120
      DO 01 JA = 1,60
      X(IA,JA) = XLEFT(IA,JA)
      JB = JA+60
      X(IA,JB) = XRIGHT(IA,JA)
01    CONTINUE
C
C      READ IN THE PROFILE OF THE AREA
      CALL READIN (XLEFT,XRIGHT,MAPNAM,IFLAG)
      IF (IFLAG.NE.0) GO TO 997
C
C      FORM ARRAY
      DO 1001 IA = 1,120
      DO 1001 JA = 1,60
      PROFIL(IA,JA) = XLEFT(IA,JA)
      JB = JA + 60
      PROFIL (IA,JB) = XRIGHT(IA,JA)
1001  CONTINUE
C
C      ** SCAN THE ARRAY FOR CELLS NEEDING CIRCLES **
      DO 1002 ICNT = 1,NUMRAD
      DO 1002 I = 1,120
      DO 1002 J = 1,120
      IF(X(I,J).EQ.FLAG) GO TO 10
      GO TO 03
10    CALL LOCUSS (I,J,FILL(ICNT),RADIUS(ICNT),Z)
03    CONTINUE
1002  CONTINUE
C
C      ** MASK OUT ALL BUT AREA OF INTEREST **
      DO 1003 I=1,120
      DO 1003 J=1,120
      IF (PROFIL(I,J).EQ.BLANK.OR.PROFIL(I,J).EQ.ZERO) Z(I,J)=BLANK
C      **** INSERT THE NEXT CARD (REMOVE THE "C") TO ****
```

```
C      **** DISPLAY CELLS REQUIRING LOCI DEFINITION   ****
C     IF (X(I,J).EQ.FLAG) Z(I,J) = FLAG
1003  CONTINUE
C
C     ** CREATE A NEW CARD DECK FROM THE COMPUTED DATA **
      WRITE(7,96)
96    FORMAT(1X,'OUTPUT FROM PROGRAM LOCUS')
      WRITE(7,97) FILL
97    FORMAT(1X,'SYMBOL(S) USED TO FILL CIRCLE: ',A1,',',A1,',',A1,',',A
     &1,',',A1)
      WRITE(7,98) FLAG
98    FORMAT(1X,'SYMBOL USED TO DENOTE CELLS NEEDING LOCI: ',A1)
      WRITE(7,99) RADIUS
99    FORMAT(1X,'RADII OF LOCUS: ',I3,',',I3,',',I3,',',I3,',',I3)
      WRITE(7,100) MAPNAM
100   FORMAT (1X,5A4)
      MARGIN = 0
      DO 30 L = 1,120
      MARGIN = MARGIN + 1
      WRITE (7,101) MARGIN, (Z(L,K),K=1,60)
101   FORMAT (2X,I3,3X,60A1)
30    CONTINUE
      MARGIN = 0
      DO 40 L = 1,120
      MARGIN = MARGIN + 1
      WRITE (7,102) (Z(L,K),K=61,120), MARGIN
102   FORMAT (1X,60A1,1X,I3)
40    CONTINUE
C
C      *** ERROR SECTION ***
      GO TO 999
997   WRITE(6,998)
998   FORMAT(1X,'PROGRAM LOCUS HALTED DUE TO ERROR CONDITION')
999   STOP
      END
      SUBROUTINE LOCUSS (M,N,FILL,RADIUS,Z)
C     *****************************************************************
C     *              SUBROUTINE NAME:  LOCUSS                         *
C     *       VARIABLE LIST:  (FROM CALL STATEMENT)                   *
C     *           M = ROW NUMBER OF POINT CENTER                      *
C     *           N = COLUMN NUMBER OF POINT CENTER                   *
C     *        FILL = SINGLE-PLACE CHARACTER TO BE ASSIGNED IN CIRCLE *
C     *      RADIUS = RADIUS OF CIRCLE                                *
C     *           Z = ARRAY OF CELLS WITHIN ASSIGNED CIRCLE           *
C     *                                                               *
C     *  NOTE: CELLHT, CELWTH, AND RADIUS MUST BE IN COMMON UNITS     *
C     *****************************************************************
      INTEGER*2 Z(120,120), IX(120), IOYY(120)
      INTEGER*2 FILL
      INTEGER RADIUS, XLIM, YLIM
```

```
C      DIMENSION IX(120)
C      DIMENSION IOYY(120)
C
C        ** INITIALIZATION SECTION **
       J = 0
       XLIM = 120
       YLIM = 120
       CELWTH = .8
       CELLHT = 1.0
C
C       ** CALCULATE RADIUS LENGTH IN CELLS AND ROUND OFF RESULT **
       IRAD = RADIUS/CELWTH+0.5
C
C       ** SEARCH NW QUADRANT **
C       ** GO OUT X-AXIS TO LEFT **
       IOY = 0
       IOX = -IRAD
C
C       ** GO UP A CELL **
10     IOY = IOY-1
C
C       ** CALCULATE HYPOTENUSE FOR NEW CELL **
15     UP = IOY*CELLHT
       OUT = IOX*CELWTH
       HYPTNS = SQRT((UP**2)+(OUT**2))
C       ** CHECK TO DETERMINE IF CELL IS INSIDE RADIUS **
       IF (HYPTNS.LT.RADIUS) GO TO 10
C
C       ** CREATE Y-OFFSET ARRAY (ROUND OFF TO INTEGER VALUE) **
       J = J+1
C       ** (IX = ARRAY OF COLUMN LENGTHS) **
       IX(J) = -IOY*2+1
C       ** (IOYY = ARRAY OF OFFSETS FROM X AXIS FOR RIGHT SIDE WORK) **
       IOYY(J) = IOY
C
C      ** SET ENTIRE COLUMN IN LEFT SEMICIRCLE EQUAL TO CHARACTER **
       LDOWN = IX(J)
       DO 20 I = 1,LDOWN
       IA = M+IOY+I-1
       IB = N+IOX
       IF(IA.GT.0.AND.IB.GT.0.AND.IA.LE.YLIM.AND.IB.LE.XLIM)Z(IA,IB)=FILL
20     CONTINUE
C
C       ** GO TO CELL ON RIGHT **
       IOX = IOX+1
       IF (IOX.GT.0) GO TO 30
       GO TO 15
C
C      ** START ON RIGHT SIDE SEMICIRCLE **
30     J = J-1
```

```
      LDOWN = IX(J)
      DO 40 I = 1,LDOWN
      IA = M+IOYY(J)+I-1
      IB = N+IOX
      IF(IA.GT.0.AND.IB.GT.0.AND.IA.LE.YLIM.AND.IB.LE.XLIM)Z(IA,IB)=FILL
40    CONTINUE
      IF (J.EQ.1) GO TO 50
      IOX = IOX + 1
      GO TO 30
50    CONTINUE
C
      RETURN
      END
      SUBROUTINE READIN (XLEFT,XRIGHT,MAPNAM,IFLAG)
C      * * * * * * * * * * * * * * * * * * * * * * * *
C      ***      SUBROUTINE NAME:  READIN           ***
C      *** THIS SUBROUTINE READS IN AND CHECKS     ***
C      ***      SYMBOL OR NUMERIC MAPS             ***
C      *** INPUT SEQUENCE:                         ***
C      ***         1) MAP NAME,  COLS. 1-20        ***
C      ***         2) (MAP DATA)                   ***
C      * * * * * * * * * * * * * * * * * * * * * * * *
C      *** NOTE: LESS THAN 240 MAP CARDS CAN BE    ***
C      ***       READ USING THE SHORTCUT PROCEDURE ***
C      * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      INTEGER MAPNAM(5)
      INTEGER RNUMBR, R2, CHEKL, CHEKR
C
C      ** READ THE MAP NAME **
      READ(4,111) MAPNAM
111   FORMAT(5A4)
C      *****LEFT SIDE READ LOOP*****
      L=1
      CHEKL=1
101   READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01    FORMAT(2X,I3,I3,60A1)
      IF(CHEKL.NE.LNUMBR) GO TO 991
      IF(L2.EQ.0) GO TO 100
      N=L2-LNUMBR
      DO 99 J=1,N
      L=L+1
      CHEKL=CHEKL+1
      BACKSPACE 4
99    READ(4,011) (XLEFT(L,K),K=1,60)
011   FORMAT(8X,60A1)
100   L=L+1
      CHEKL=CHEKL+1
      IF(L.GT.120) GO TO 102
      GO TO 101
```

```
C       *****RIGHT SIDE READ LOOP*****
102    L=1
       CHEKR=1
103    READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02     FORMAT(1X,60A1,1X,I3,I3)
       IF(CHEKR.NE.RNUMBR) GO TO 992
       IF(R2.EQ.0) GO TO 200
       N=R2-RNUMBR
       DO 199 J=1,N
       L=L+1
       CHEKR=CHEKR+1
       BACKSPACE 4
199    READ(4,022)(XRIGHT(L,K),K=1,60)
022    FORMAT(1X,60A1)
200    L=L+1
       CHEKR=CHEKR+1
       IF(L.GT.120) GO TO 201
       GO TO 103
201    CONTINUE
       GO TO 999
991    WRITE (6,901) CHEKL, MAPNAM
       IFLAG=10
       GO TO 999
992    WRITE (6,902) CHEKR, MAPNAM
       IFLAG=10
901    FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED. CARD ',I3,' ON LEFT
      &OUT OF ORDER ON MAP NAME: ',5A4)
902    FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED, CARD ',I3,' ON RIGHT
      & OUT OF ORDER ON MAP NAME: ',5A4)
999    CONTINUE
       RETURN
       END
```

```
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     *               PROGRAM NAME:  SEARCH                         *
C     *   THIS PROGRAM PERFORMS AN ORTHOGONAL 4-WAY SEARCH TO       *
C     *       DETERMINE HOW MANY DIFFERENT CELL VALUES ARE          *
C     *         ENCOUNTERED WITHIN A GIVEN SEARCH RADIUS            *
C     *    NOTE: SEARCHES ARE NOT MADE FROM CELLS HAVING A          *
C     *          A ZERO OR BLANK VALUE.                             *
C     *   INPUT SEQUENCE:                                           *
C     *          1) SEARCH RADIUS (EXPRESSED IN UNITS OF CELL       *
C     *              HEIGHT) (FREE FORMAT)                          *
C     *          2) MAP NAME (SEE SUBROUTINE READIN FOR FORMAT)     *
C     *          3) MAP DATA (SEE SUBROUTINE READIN FOR FORMAT)     *
C     *   OUTPUT: MAP4B-COMPATIBLE CARD DECK CONTAINING:            *
C     *          1) MAP NAME                                        *
C     *          2) MAP DATA (LEFT HALF)                            *
C     *          3) MAP DATA (RIGHT HALF)                           *
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 X(120,120), NEWX(120,120)
      INTEGER*2 XLEFT (120,60), XRIGHT (120,60), ICATS(56)
      INTEGER*2 ZERO/'0'/, BLANK/' '/
      INTEGER MAPNAM(5), RADIUS, RADWTH
C
C      ** SET CELL WIDTH TO CELL HEIGHT RATIO **
      CELWTH = .8
C
C      ** READ SEARCH RADIUS **
      READ (4,*) RADIUS
C
C      ** DETERMINE SEARCH RADIUS IN UNITS OF CELL WIDTH **
      RADWTH = (RADIUS/CELWTH) + 0.5
C
C      ** READ A SET OF MAP CARDS **
      IFLAG = 0
      CALL READIN (XLEFT,XRIGHT,MAPNAM,IFLAG)
      IF (IFLAG.NE.0) GO TO 999
C
C      *** FORM 120 BY 120 ARRAY FROM INPUT DATA ***
      DO 01 IA=1,120
      DO 01 JA=1,60
      X(IA,JA) = XLEFT(IA,JA)
      JB=JA+60
      X(IA,JB) = XRIGHT(IA,JA)
01    CONTINUE
C
C      ** MAIN PROGRAM LOOP **
      DO 60 I=1,120
      DO 60 J=1,120
C      *** CLEAR ICATS ARRAY ***
      DO 03 M=1,56
      ICATS(M)=0
```

```
03    CONTINUE
C
C      ** SKIP CELLS HAVING VALUE OF ZERO OR BLANK **
      IF(X(I,J).EQ.ZERO.OR.X(I,J).EQ.BLANK) GO TO 51
C
C      ** BEGIN ORTHOGONAL SEARCHES **
C
C      *** EASTWARD SEARCH ***
      JE=J
      DO 10 IW=1,RADWTH
      JE=J+IW
      IF (JE.GT.120) GO TO 11
      IF (X(I,JE).EQ.X(I,J)) GO TO 10
      CALL CATN (X,I,JE,ICATS)
10    CONTINUE
C      *** SOUTHWARD SEARCH ***
11    IS=I
      DO 20 IX=1,RADIUS
      IS=I+IX
      IF (IS.GT.120) GO TO 21
      IF (X(IS,J).EQ.X(I,J)) GO TO 20
      CALL CATN (X,IS,J,ICATS)
20    CONTINUE
C      *** WESTWARD SEARCH ***
21    JW=J
      DO 30 IY=1,RADWTH
      JW=J-IY
      IF (JW.LT.0) GO TO 31
      IF (X(I,JW).EQ.X(I,J)) GO TO 30
      CALL CATN (X,I,JW,ICATS)
30    CONTINUE
C      *** NORTHWARD SEARCH ***
31    IN=I
      DO 40 IZ=1,RADIUS
      IN=I-IZ
      IF (IN.LT.0) GO TO 41
      IF (X(IN,J).EQ.X(I,J)) GO TO 40
      CALL CATN (X,IN,J,ICATS)
40    CONTINUE
41    CONTINUE
C      *** SUM THE ICATS ARRAY ***
      ITYPES = 0
      DO 45 ISUM=1,56
      ITYPES = ITYPES + ICATS(ISUM)
45    CONTINUE
50    NEWX(I,J)=ITYPES
      GO TO 60
51    NEWX(I,J)=0
60    CONTINUE
C
```

```
C      *** CREATE NEW CARD DECK FROM COMPUTED DATA ***
      WRITE (7,100) MAPNAM
      MARGIN=0
      DO 70 L=1,120
      MARGIN=MARGIN+1
      WRITE (7,101) MARGIN, (NEWX(L,K),K=1,60)
70    CONTINUE
      MARGIN=0
      DO 80 L=1,120
      MARGIN=MARGIN+1
      WRITE (7,102) (NEWX(L,K),K=61,120), MARGIN
80    CONTINUE
100   FORMAT (1X,5A4)
101   FORMAT (2X,I3,3X,60I1)
102   FORMAT (1X,60I1,1X,I3)
999   STOP
      END
      SUBROUTINE READIN (XLEFT,XRIGHT,MAPNAM,IFLAG)
C      * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***        SUBROUTINE NAME:  READIN           ***
C      ***   THIS SUBROUTINE READS IN AND CHECKS     ***
C      ***          SYMBOL AND NUMERIC MAPS          ***
C      ***  INPUT SEQUENCE:                          ***
C      ***          1) MAP NAME,  COLS. 1-20         ***
C      ***          2) (MAP DATA)                    ***
C      * * * * * * * * * * * * * * * * * * * * * * * * *
C      *** NOTE: LESS THAN 240 MAP CARDS CAN BE      ***
C      ***       READ USING THE SHORTCUT PROCEDURE   ***
C      * * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      INTEGER MAPNAM(5)
      INTEGER RNUMBR, R2
      INTEGER CHEKL, CHEKR
C      ** READ THE MAP NAME **
      READ(4,111) MAPNAM
111   FORMAT(5A4)
C      *****LEFT SIDE READ LOOP*****
      L=1
      CHEKL=1
101   READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01    FORMAT(2X,I3,I3,60A1)
      IF(CHEKL.NE.LNUMBR) GO TO 991
      IF(L2.EQ.0) GO TO 100
      N=L2-LNUMBR
      DO 99 J=1,N
      L=L+1
      CHEKL=CHEKL+1
      BACKSPACE 4
99    READ(4,011) (XLEFT(L,K),K=1,60)
011   FORMAT(8X,60A1)
```

```
100    L=L+1
       CHEKL=CHEKL+1
       IF(L.GT.120) GO TO 102
       GO TO 101
C       *****RIGHT SIDE READ LOOP*****
102    L=1
       CHEKR=1
103    READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02     FORMAT(1X,60A1,1X,I3,I3)
       IF(CHEKR.NE.RNUMBR) GO TO 992
       IF(R2.EQ.0) GO TO 200
       N=R2-RNUMBR
       DO 199 J=1,N
       L=L+1
       CHEKR=CHEKR+1
       BACKSPACE 4
199    READ(4,022)(XRIGHT(L,K),K=1,60)
022    FORMAT(1X,60A1)
200    L=L+1
       CHEKR=CHEKR+1
       IF(L.GT.120) GO TO 201
       GO TO 103
201    CONTINUE
       GO TO 999
991    WRITE (6,901) CHEKL, MAPNAM
       IFLAG=10
       GO TO 999
992    WRITE (6,902) CHEKR, MAPNAM
       IFLAG=10
901    FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED. CARD ',I3,' ON LEFT
      &OUT OF ORDER ON MAP NAME: ',5A4)
902    FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED, CARD ',I3,' ON RIGHT
      & OUT OF ORDER ON MAP NAME: ',5A4)
999    CONTINUE
       RETURN
       END
       SUBROUTINE CATN (X,II,JJ,ICAT)
C       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C       ***              SUBROUTINE NAME :  CATN                    ***
C       ***  THIS SUBROUTINE CREATES AN ARRAY (ICAT) OF DIFFERENT   ***
C       ***     CELL VALUES ENCOUNTERED WITHIN A GIVEN SEARCH       ***
C       *** DIRECTION, AND PASSES THE ARRAY BACK TO THE MAIN PGM    ***
C       ***                  NO INPUT IS REQUIRED                   ***
C       ***    WARNING: THE BLANK SYMBOL IS NOT COUNTED AS A        ***
C       ***             DIFFERENT SYMBOL, AND SHOULD BE RESERVED    ***
C       ***             FOR SIGNIFYING CELLS OUTSIDE THE MAPPED     ***
C       ***             AREA.                                       ***
C       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       INTEGER*2 ICAT(56), IN(56), X(120,120)
C
```

```
C        ** DEFINE THE ARRAY OF POSSIBLE MAP SYMBOLS **
C        ** BLANK CHARACTER IS NOT CONCIDERED, BUT MAY BE ADDED BELOW **
       DATA IN/'A','B','C','D','E','F','G','H','I','J','K','L','M','N','
      &O','P','Q','R','S','T','U','V','W','X','Y','Z','-','=','+','*',',
      &','$','/','(',')','.','%',';','>','<',':','&','@','{','"','!','0',
      &'1','2','3','4','5','6','7','8','9'/
C
       DO 10 I=1,56
       IF (X(II,JJ).NE.IN(I)) GO TO 10
5      ICAT(I)=1
       GO TO 99
10     CONTINUE
99     RETURN
       END
```

```
C       ************************************************************
C       ***              PROGRAM NAME:  CONVERT                  ***
C       *** THIS PROGRAM CONVERTS ALPHANUMERIC MAPS TO NUMERIC MAPS ***
C       ***   INPUT SEQUENCE:                                    ***
C       ***         1) MAP NAME,   COLS. 1-20                    ***
C       ***         2)  (MAP DATA)                               ***
C       ***                                                      ***
C       *** NOTE: ONLY 1 MAP DECK SHOULD BE READ IN FOR EACH RUN ***
C       ***          OF THIS PROGRAM                             ***
C       *** NOTE: LESS THAN 240 MAP CARDS CAN BE READ IN USING THE ***
C       ***          SHORTCUT PROCEDURE                          ***
C       ************************************************************
C
      INTEGER XLEFT(120,60), XRIGHT(120,60)
      INTEGER NEWLFT(120,60), NEWRHT(120,60)
      INTEGER IN(57), OUT(57), MAPNAM(5)
      INTEGER RNUMBR, R2, CHEKL, CHEKR
C       ***   THE VARIABLE 'IN' IN THE FOLLOWING DATA STATEMENT   ***
C       ***    REPRESENTS THE ENTIRE RANGE OF POSSIBLE SYMBOLS    ***
C       ***        WHICH MAY BE USED WITH THE MAP4B SYSTEM        ***
      DATA IN/'A','B','C','D','E','F','G','H','I','J','K','L','M','N','
     &O','P','Q','R','S','T','U','V','W','X','Y','Z','-','=','+','*',',
     &','$','/','(',')','.','%',';','>','<',':','&','@','{','"','!','0',
     &'1','2','3','4','5','6','7','8','9',' '/
C
C       *** OUTPUT SYMBOLS ARE DETERMINED WITH THE FOLLOWING DATA ***
C       *** STATEMENT BY ASSIGNING THE DESIRED INTEGER VALUES IN  ***
C       *** THE PROPER SEQUENCE SO AS TO REPLACE A SYMBOL WITH    ***
C       *** THE DESIRED SINGLE INTEGER VALUE                      ***
C       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C       **** WARNING: THE REPLACEMENT SEQUENCE MUST BE EXACT AND ****
C       **** AND THERE MUST BE A TOTAL OF 45 DATA VALUES IN THE  ****
C       **** 'DATA STATEMENT'                                    ****
C       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DATA OUT/57*0/
C
C       ** READ MAP NAME **
      READ(4,111) MAPNAM
111   FORMAT(5A4)
C
C       ** WRITE MAP NAME **
      WRITE(7,222) MAPNAM
222   FORMAT(1X,5A4)
C
C       *****LEFT SIDE READ LOOP*****
      L=1
      CHEKL=1
101   READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01    FORMAT(2X,I3,I3,60A1)
      IF(CHEKL.NE.LNUMBR) GO TO 991
```

```
      DO 55 K=1,60
      DO 50 I=1,45
      ICHEK=I
      IF ((XLEFT(L,K)).EQ.IN(I)) GO TO 300
50    CONTINUE
      WRITE (7,51)
51    FORMAT (1X,'ERROR: NO CORRESPONDING SYMBOL, CHECK DATA STMNT CARDS
     &')
300   NEWLFT(L,K)=OUT(ICHEK)
55    CONTINUE
C
C      ** LEFT SIDE CARD OUTPUT **
      WRITE (7,555) LNUMBR,L2,(NEWLFT(L,K),K=1,60)
555   FORMAT (2X,I3,I3,60A1)
C
      IF(L2.EQ.0) GO TO 100
      N=L2-LNUMBR
      DO 99 J=1,N
      L=L+1
      CHEKL=CHEKL+1
      BACKSPACE 4
99    READ(4,011) (XLEFT(L,K),K=1,60)
011   FORMAT(8X,60A1)
100   L=L+1
      CHEKL=CHEKL+1
      IF(L.GT.120) GO TO 102
      GO TO 101
C      *****RIGHT SIDE READ LOOP*****
102   L=1
      CHEKR=1
103   READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02    FORMAT(1X,60A1,1X,I3,I3)
      IF(CHEKR.NE.RNUMBR) GO TO 992
      DO 65 K=1,60
      DO 60 I=1,45
      ICHEK=I
      IF (XRIGHT(L,K).EQ.IN(I)) GO TO 400
60    CONTINUE
400   NEWRHT(L,K)=OUT(ICHEK)
65    CONTINUE
C
C      ** RIGHT SIDE CARD OUTPUT **
      WRITE (7,556) (NEWRHT(L,K),K=1,60),RNUMBR,R2
556   FORMAT (1X,60A1,1X,I3,I3)
C
      IF(R2.EQ.0) GO TO 200
      N=R2-RNUMBR
      DO 199 J=1,N
      L=L+1
      CHEKR=CHEKR+1
```

```
      BACKSPACE 4
199   READ(4,022)(XRIGHT(L,K),K=1,60)
022   FORMAT(1X,60A1)
200   L=L+1
      CHEKR=CHEKR+1
      IF(L.GT.120) GO TO 201
      GO TO 103
201   CONTINUE
      GO TO 999
C
C     ** ERROR CONDITION SECTION **
991   WRITE (7,901) CHEKL
      GO TO 999
992   WRITE (7,902) CHEKR
901   FORMAT(1H ,'ERROR: PROGRAM ABORTED DUE TO IMPROPER CARD SEQUENCE.
     &CARD ',I3,' ON LEFT OUT OF ORDER.')
902   FORMAT(1H ,'ERROR: PROGRAM ABORTED DUE TO IMPROPER CARD SEQUENCE.
     &CARD ',I3,' ON RIGHT OUT OF ORDER.')
999   CONTINUE
      STOP
      END
```

```
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C     *                   PROGRAM NAME:  SLOPE                        *
C     *   THIS PROGRAM USES A SIMPLE, GREATEST ELEVATION DIFFERENCE   *
C     *   ALGORITHM TO CALCULATE SLOPE AND ASPECT FOR A GIVEN CELL.   *
C     *    NOTE: DIFFERENCES ARE CALCULATED FROM THE TARGET CELL      *
C     *    INPUT SEQUENCE:
C     *           MAP DATA (SEE READ STATEMENT FOR FORMAT)            *
C     *    OUTPUT (UNIT 7): 2 DECKS OF CARDS CONSISTING OF:           *
C     *           1) 120 LEFT SIDE CARDS OF SLOPE DATA                *
C     *           2) 120 RIGHT SIDE CARDS OF SLOPE DATA               *
C     *           3) 120 LEFT SIDE CARDS OF ASPECT DATA               *
C     *           4) 120 RIGHT SIDE CARDS OF ASPECT DATA              *
C     *    OUTPUT (UNIT 6): ERROR MESSAGES                            *
C     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION X(120,120)
      INTEGER*2 SLOPE(120,120), ASPCT(120,120)
      INTEGER*2 ISLOPE, IASPCT
C
C      ** SET CELL HEIGHT, CELL WIDTH, AND CRFX FOR USGS TOPO SHEETS **
C      ** NOTE : CRFX = 379.45733 IF ELEVATION IS IN FEET **
C      **        CRFX = 115.66 IF ELEVATION IS IN METERS  **
      CELLHT = 1.0
      CELWTH = .8393667
      CRFX = 242.
C
C      ** ENTER DEVIATION (DEGREES CLOCKWISE) IF BASE MAP IS TILTED **
      DEV = 10.
C
      IFLAG = 0
C
C      ** READ THE ELEVATION DATA **
      DO 01 IA=1,120
      READ (4,111) (X(IA,JA),JA=1,120)
111   FORMAT(120F6.0)
01    CONTINUE
C
C      ** MAIN PROGRAM LOOP **
      DO 03 I=1,120
      DO 03 J=1,120
      CALL SLOPA (I,J,X, CELLHT,CELWTH,ISLOPE,IASPCT,CRFX,IFLAG,DEV)
      IF (IFLAG.NE.0) GO TO 999
      SLOPE(I,J) = ISLOPE
      ASPCT(I,J) = IASPCT
C
C      ** SLOPE SORT SECTION **
      IF (SLOPE(I,J).GE.0.AND.SLOPE(I,J).LT.5) SLOPE(I,J) = 1
      IF (SLOPE(I,J).GE.5.AND.SLOPE(I,J).LT.10) SLOPE(I,J) = 2
      IF (SLOPE(I,J).GE.10.AND.SLOPE(I,J).LT.15) SLOPE(I,J) = 3
      IF (SLOPE(I,J).GE.15.AND.SLOPE(I,J).LT.20) SLOPE(I,J) = 4
      IF (SLOPE(I,J).GE.20.AND.SLOPE(I,J).LT.25) SLOPE(I,J) = 5
```

```
      IF (SLOPE(I,J).GE.25.AND.SLOPE(I,J).LT.30) SLOPE(I,J) = 6
      IF (SLOPE(I,J).GE.30.AND.SLOPE(I,J).LT.35) SLOPE(I,J) = 7
      IF (SLOPE(I,J).GE.35.AND.SLOPE(I,J).LT.40) SLOPE(I,J) = 8
      IF (SLOPE(I,J).GE.40.AND.SLOPE(I,J).LT.45) SLOPE(I,J) = 9
      IF (SLOPE(I,J).GE.45) SLOPE(I,J) = 0
C
C      *** TEST FOR ERRORS **
      IF (SLOPE(I,J).LT.0) GO TO 996
      GO TO 998
996   WRITE(6,997) I, J, SLOPE(I,J)
997   FORMAT(1X,'WARNING: CELL ',I3,1X,I3,' HAS A VALUE OF:',I5,' FOR SL
     &OPE')
998   CONTINUE
C
C      ** ASPECT SORT SECTION **
      IF (ASPCT(I,J).GE.0.AND.ASPCT(I,J).LT.45) ASPCT(I,J) = 1
      IF (ASPCT(I,J).GE.45.AND.ASPCT(I,J).LT.90) ASPCT(I,J) = 3
      IF (ASPCT(I,J).GE.90.AND.ASPCT(I,J).LT.135) ASPCT(I,J) = 5
      IF (ASPCT(I,J).GE.135.AND.ASPCT(I,J).LT.180) ASPCT(I,J) = 7
      IF (ASPCT(I,J).GE.180.AND.ASPCT(I,J).LT.225) ASPCT(I,J) = 9
      IF (ASPCT(I,J).GE.225.AND.ASPCT(I,J).LT.270) ASPCT(I,J) = 6
      IF (ASPCT(I,J).GE.270.AND.ASPCT(I,J).LT.315) ASPCT(I,J) = 4
      IF (ASPCT(I,J).GE.315.AND.ASPCT(I,J).LE.360) ASPCT(I,J) = 2
      IF (ASPCT(I,J).GE.360) ASPCT(I,J) = 0
C
C      *** TEST FOR ERRORS WITH ASPECT ***
      IF(ASPCT(I,J).LT.0) GO TO 889
      GO TO 890
889   WRITE(6,887) I, J, ASPCT(I,J)
887   FORMAT(1X,'WARNING: CELL',I3,1X,I3,' HAS A VALUE OF',I5,' FOR ASPE
     &CT')
890   CONTINUE
C
03    CONTINUE
C
C      ** OUTPUT LOOP **
      WRITE (7,100)
100   FORMAT (1X,'OUTPUT FROM PROGRAM SLOPE')
      WRITE (7,101)
101   FORMAT(1X,'SECTION 1: SLOPE OUTPUT')
C      *** LEFT SIDE CARD OUTPUT LOOP ***
      MARGIN = 0
      DO 041 L=1,120
      MARGIN = MARGIN + 1
      WRITE (7,1101) MARGIN, (SLOPE(L,K),K=1,60)
1101  FORMAT (2X,I3,3X,60I1)
041   CONTINUE
C
C      *** RIGHT SIDE CARD OUTPUT LOOP ***
      MARGIN = 0
```

```
      DO 042 L=1,120
      MARGIN = MARGIN + 1
      WRITE (7,1102) (SLOPE(L,K),K=61,120), MARGIN
1102  FORMAT (1X,60I1,1X,I3)
042   CONTINUE
C
      WRITE (7,103)
103   FORMAT(1X,'SECTION 2: ASPECT OUTPUT')
C
C      *** LEFT SIDE CARD OUTPUT LOOP ***
      MARGIN = 0
      DO 051 L = 1,120
      MARGIN = MARGIN + 1
      WRITE (7,1103) MARGIN, (ASPCT(L,K),K=1,60)
1103  FORMAT(2X,I3,3X,60I1)
051   CONTINUE
C
C      *** RIGHT SIDE CARD OUTPUT LOOP ***
      MARGIN = 0
      DO 052 L = 1,120
      MARGIN = MARGIN + 1
      WRITE (7,1104) (ASPCT(L,K),K=61,120), MARGIN
1104  FORMAT (1X,60I1,1X,I3)
052   CONTINUE
C
999   STOP
      END
C
      SUBROUTINE SLOPA(M,N,X,CELLHT,CELWTH,ISLOPE,IASPCT,CRFX,IFLAG,DEV)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ** INPUT VARIABLE LIST (FROM CALLING PROGRAM):                  **
C      **         M = ROW NUMBER OF CELL                               **
C      **         N = COLUMN NUMBER OF CELL                            **
C      **         X = ARRAY OF ELEVATIONS                              **
C      **    CELLHT = HEIGHT OF CELL (DIMENSION OF CODING GRID)        **
C      **    CELWTH = WIDTH OF CELL (DIMENSION OF CODING GRID)         **
C      **      CRFX = CORRECTION FACTOR FOR CONVERTING CELL DIMENSIONS **
C      **               INTO ON-THE-GROUND MEASUREMENTS IN THE SAME    **
C      **               SAME UNITS AS ELEVATION (METERS OR FEET)       **
C      **  OUTPUT VARIABLES:                                           **
C      **    ASPECT = WEIGHTED AVERAGE OF GREATEST DOWNHILL DIRECTION  **
C      **              AND OPPOSITE ANGLE OF GREATEST UPHILL DIRECTION **
C      **              (EXPRESSED IN DEGREES)                          **
C      **     SLOPE = AVERAGE OF GREATEST ELEVATIONAL DIFFERENCE AND   **
C      **              ELEVATION OF TARGET CELL (EXPRESSED IN DEGREES) **
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      INTEGER XLIM,YLIM
      INTEGER*2 ISLOPE, IASPCT
      DIMENSION THETA (9), ELEV (9), X (120,120)
```

```
      REAL MAXUP,MINDN
C
C      ** INITIALIZATION OF MAP SIZE LIMITS **
      XLIM = 120
      YLIM = 120
C
C      CHECK M AND N FOR IMPROPER VALUES
      IF (M.LE.0.OR.M.GT.YLIM) GO TO 110
      IF (N.LE.0.OR.N.GT.XLIM) GO TO 112
C
C      CALCULATE ANGLES TO EACH OF THE ADJACENT 8 CELLS
C        NOTE: THETA(5) IS A DUMMY
C       THETA'S ARE NUMBERED CONSECUTIVELY, LEFT TO RIGHT BEGINNING AT
C        TOP LEFT...
C
      ANGLE = (ATAN(CELLHT/CELWTH))*57.295773
      THETA(2) = 360.
      THETA(4) = 270.
      THETA(6) = 90.
      THETA(8) = 180.
      THETA(1) = THETA(4)+ANGLE
      THETA(3) = THETA(6)-ANGLE
      THETA(5) = 0.
      THETA(7) = THETA(4)-ANGLE
      THETA(9) = THETA(6)+ANGLE
C
C      READ ELEVATIONS LEFT TO RIGHT BEGINNING WITH NW CORNER
C
C      TOP ROW SCAN
      IM = M-1
      IN = N-1
      DO 10 I=1,3
C      CHECK FOR OVERRUN
      IF (IM.LE.0.OR.IN.LE.0.OR.IN.GT.XLIM) GO TO 08
      ELEV(I) = X(IM,IN)
      GO TO 09
08    ELEV(I) = X(M,N)
09    IN = IN+1
10    CONTINUE
C
C      MIDDLE ROW SCAN
      IM = M
      IN = N-1
      DO 20 I=4,6
C      CHECK FOR OVERRUN
      IF (IN.LE.0.OR.IN.GT.XLIM) GO TO 18
      ELEV(I) = X(IM,IN)
      GO TO 19
18    ELEV(I) = X(M,N)
19    IN = IN+1
```

```
20    CONTINUE
C
C      BOTTOM ROW SCAN
      IM = M+1
      IN = N-1
      DO 30 I=7,9
C      CHECK FOR OVERRUN
      IF (IN.LE.0.OR.IN.GT.XLIM.OR.IM.GT.YLIM) GO TO 28
      ELEV(I) = X(IM,IN)
      GO TO 29
28    ELEV(I) = X(M,N)
29    IN = IN+1
30    CONTINUE
C
C      DETERMINE MAXIMUM ELEVATION AND LOCATION (IHIGH)
C
      HTMAX = ELEV(5)
      IHIGH = 5
      DO 40 I=1,9
      IF (ELEV(I).GT.HTMAX) GO TO 41
      GO TO 40
41    HTMAX = ELEV(I)
      IHIGH = I
40    CONTINUE
C
C      DETERMINE MINIMUM ELEVATION AND LOCATION (ILOW)
C
      HTMIN = ELEV(5)
      ILOW = 5
      DO 50 I=1,9
      IF (ELEV(I).LT.HTMIN) GO TO 51
      GO TO 50
51    HTMIN = ELEV(I)
      ILOW = I
50    CONTINUE
C
C      ***** SLOPE COMPUTATION SECTION *****
C
C      COMPUTE SLOPE, WHERE SLOPE IS THE AVERAGE OF THE DIFFERENCES
C       BETWEEN THE CENTRAL ELEVATION AND THE 2 EXTREME ELEVATIONS
C
      IF (IHIGH.EQ.1.OR.IHIGH.EQ.3.OR.IHIGH.EQ.7.OR.IHIGH.EQ.9) DISTUP=
     &(SQRT(CELLHT**2+CELWTH**2))*CRFX
      IF (IHIGH.EQ.2.OR.IHIGH.EQ.8) DISTUP=CELLHT*CRFX
      IF (IHIGH.EQ.4.OR.IHIGH.EQ.6) DISTUP=CELWTH*CRFX
      IF (ILOW.EQ.1.OR.ILOW.EQ.3.OR.IHIGH.EQ.7.OR.ILOW.EQ.9) DISTDN =
     &(SQRT(CELLHT**2+CELWTH**2))*CRFX
      IF (ILOW.EQ.2.OR.ILOW.EQ.8) DISTDN=CELLHT*CRFX
      IF (ILOW.EQ.4.OR.ILOW.EQ.6) DISTDN=CELWTH*CRFX
C
```

```
C        CREATE A DUMMY VALUE FOR KNOB AND/OR BOWL SITUATION
C
      IF (IHIGH.EQ.5) DISTUP=1.
      IF (ILOW.EQ.5.) DISTDN=1.
C
C      CALCULATE UPHILL AND DOWNHILL SLOPE ANGLES
C
      UPSLOP = ATAN((ELEV(IHIGH)-ELEV(5))/DISTUP)*57.295773
      DNSLOP = ATAN((ELEV(5)-ELEV(ILOW))/DISTDN)*57.295773
C
C      **** ACCOMMODATE BOWL OR KNOB SITUATION ****
      IF (IHIGH.EQ.5) GO TO 68
      IF (ILOW.EQ.5) GO TO 69
C      DETERMINE SLOPE ANGLE
      SLOPE = (UPSLOP+DNSLOP)/2.
C
      GO TO 70
68    SLOPE = DNSLOP
      GO TO 70
69    SLOPE = UPSLOP
70    CONTINUE
C      ACCOMMODATE FOR A HORIZONTAL PLANE
      IF (IHIGH.EQ.5.AND.ILOW.EQ.5) SLOPE=0.
C
C      ** SET ISLOPE (AN INTEGER) TO THE VALUE OF SLOPE **
      ISLOPE = SLOPE + 0.5
C
C     ***** ASPECT COMPUTATION SECTION *****
C
C      SET MAXIMUM UP-SLOPE AZIMUTH
      MAXUP = THETA(IHIGH)
C
C      SET MINIMUM DOWN-SLOPE AZIMUTH
      MINDN = THETA(ILOW)
C
C      ACCOMMODATE FOR KNOB SHAPE (I.E. ELEV5 IS HIGHEST)
      IF (IHIGH.EQ.5) GO TO 58
C
C      SET AN AZIMUTH 180 DEGREES FROM MAXIMUM UP-SLOPE
      UPOPP = MAXUP + 180.
C
C      ACCOMMODATE FOR BOWL SHAPE (I.E. ELEV5 IS LOWEST)
      IF (ILOW.EQ.5) GO TO 59
C
C      ACCOMMODATE FOR A PLANE
      IF (MINDN-UPOPP.GT.-.000001.AND.MINDN-UPOPP.LT..00001) GO TO 58
C
C      CREATE WEIGHTING FACTORS BASED ON SLOPE RATIOS
C
      SUMSLP = UPSLOP + DNSLOP
```

```
      UPRTIO = UPSLOP/SUMSLP
      DNRTIO = DNSLOP/SUMSLP
C
C      CALCULATE ASPECT BASED ON ELEVATIONAL (SLOPE) WEIGHTS
C
      CALL VECADD (MINDN,DNRTIO,UPOPP,UPRTIO,ASPECT,XMAG)
C
      GO TO 60
58    ASPECT = MINDN
      GO TO 60
59    ASPECT = UPOPP
60    CONTINUE
C
C      ACCOMMODATE FOR A HORIZONTAL PLANE
61    IF (IHIGH.EQ.5.AND.ILOW.EQ.5) ASPECT = 0.
C
C      ACCOMMODATE FOR NEGATIVE ANGLES
      IF (ASPECT.LT.0.) ASPECT=ASPECT+360.
C
C      ACCOMMODATE FOR ANGLES GREATER THAN 360 DEGREES
      IF (ASPECT.GT.360.) ASPECT=ASPECT-360.
C
C      ** CORRECT FOR TILTED BASE MAP **
      ASPECT = ASPECT - DEV
C
      IF (ASPECT.LT.0.) ASPECT = ASPECT + 360.
C
C      ** CREATE INTEGER VALUE FOR ASPECT **
      IASPCT = ASPECT + 0.5
C
      GO TO 999
110   WRITE (6,111) M
111   FORMAT (1X,'SUBROUTINE SLOPE ABORTED... M HAS VALUE OF',I3)
      IFLAG = 999
      GO TO 999
112   WRITE (6,113) N
113   FORMAT (1X,'SUBROUTINE SLOPE ABORTED... N HAS VALUE OF',I3)
      IFLAG = 999
999   CONTINUE
      RETURN
      END
C
C      *** VECTOR ADDITION SUBROUTINE ***
      SUBROUTINE VECADD(Q1,D1,Q2,D2,Q3,D3)
      CALL PRIANG(Q1,P1,DX,DY)
      RP1=P1/57.295773
      Y1=DY*SIN(RP1)*D1
      X1=DX*COS(RP1)*D1
      CALL PRIANG(Q2,P2,DX,DY)
      RP2=P2/57.295773
```

```
      Y2=DY*SIN(RP2)*D2
      X2=DX*COS(RP2)*D2
      YDIFF=Y1+Y2
      XDIFF=X1+X2
      IF(XDIFF.LT.0.00001.AND.XDIFF.GT.-.00001) XDIFF=.00001
      IF(YDIFF.LT.0.00001.AND.YDIFF.GT.-.00001) YDIFF=.00001
      D3=SQRT(XDIFF*XDIFF+YDIFF*YDIFF)
      RP3=ATAN(YDIFF/XDIFF)
      Q3=RP3*57.295773
C     WRITE(6,60) X1,Y1,X2,Y2,Q3,XDIFF,YDIFF
      IF(XDIFF.GT.0.) Q3=90.-Q3
      IF(XDIFF.LT.0.) Q3=270.-Q3
C     IF (ABS(Q1-Q2).EQ.180.) Q3=Q1+90.
      IF(Q3.GT.360.) Q3=Q3-360.
      IF(Q3.LT.0.) Q3=360.-Q3
C60     FORMAT(5F20.5)
      RETURN
      END
      SUBROUTINE PRIANG(Q,P,DX,DY)
      THETA=Q
      J=(THETA-.0001)/90.
      IF(THETA.LE.90.) ANG=90.-THETA
      IF(THETA.GT.90.) ANG=THETA-90.
      IF(THETA.GT.180.) ANG=270.-THETA
      IF(THETA.GT.270.) ANG=THETA-270.
      P=ANG
      DY=1.
      DX=1.
      IF(J.EQ.1.OR.J.EQ.2) DY=-DY
      IF(J.EQ.2.OR.J.EQ.3) DX=-DX
      RETURN
      END
```

Appendix C

The PREDIC Program

```
C       *** PROGRAM FOR LINKING PREDICTION PGM WITH MAP4B ***
      DIMENSION AGETBL(15,6), VECTOR(16,6), ELEV(120,120)
      REAL MARKOV(16,5,5)
      INTEGER*2 IFACTR(24,16)
      INTEGER*2 BLANK/' '/, ZERO/'0'/, ONE/'1'/
      INTEGER*2 MASTMT(120,120), OPENMT(120,120), ROOSMT(120,120)
      INTEGER*2 MASTDV, IOPENG, IROOST, IASPC
      INTEGER*2 SPPMAP(120,120), AGEMAP(120,120), ISTRM(120,120), ISTR
      INTEGER*2 SLOPE(120,120), ASPECT(120,120), ISLOPE, IASPCT, ISLOP
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      INTEGER*2 SPP1/'A'/, SPP2/'B'/, SPP3/'C'/, SPP4/'D'/, SPP5/'E'/,
     &SPP6/'F'/, SPP7/'G'/, SPP8/'H'/, SPP9/'I'/, SPP10/'J'/, SPP11/'K'/
     &,SPP12/'L'/
      INTEGER SPP(11), IM(3), NAME(16,15), HORIZN
      INTEGER*2 AGE1/'1'/, AGE2/'2'/, AGE3/'3'/, AGE4/'4'/, AGE5/'5'/,
     &AGE6/'6'/, AGE7/'7'/, AGE8/'8'/, AGE9/'9'/, AGE10/'A'/, AGE11/'B'/
     &, AGE12/'C'/, AGE13/'D'/, AGE14/'E'/, AGE15/'F'/
C
C       ** SET INITIAL SETUP CONDITIONS **
      CELLHT = 1.0
      CELWTH = .8393667
      CRFX = 242.
      DEV = 10.
      IFLAG = 0
      ICAST = 2
      HORIZN = 1
C
C       ** READ MARKOV TRANSITION MATRICES **
      DO 200 I = 1,16
      READ(1,201) (NAME(I,INAM),INAM=1,15)
201   FORMAT(15A4)
      DO 200 J = 1,5
      READ(1,*) (MARKOV(I,J,K),K=1,5)
200   CONTINUE
C
C       ** READ FACTOR OCCURRENCE (FSO) TABLE **
      DO 250 I = 1,24
      READ(2,*) (IFACTR(I,J),J=1,16)
250   CONTINUE
C
C       ** READ IN ELEVATION DATA **
      DO 300 IA = 1,120
      READ (3,301) (ELEV(IA,JA),JA=1,120)
```

```
301   FORMAT (120F5.0)
300   CONTINUE
C
C       ** READ IN SPECIES DATA FROM MAP CARDS **
        CALL READIN (XLEFT,XRIGHT,IFLAG)
      IF (IFLAG.NE.0) GO TO 1012
C
C       ** FORM 120 BY 120 WORKING MATRIX FROM SPECIES DATA **
      DO 1001 IA = 1,120
      DO 1001 JA = 1,60
      SPPMAP(IA,JA) = XLEFT(IA,JA)
      JB = JA + 60
      SPPMAP(IA,JB) = XRIGHT(IA,JA)
1001  CONTINUE
C
C       ** READ IN AGE DISTRIBUTION DATA FROM MAP CARDS **
      CALL READIN (XLEFT,XRIGHT,IFLAG)
      IF (IFLAG.NE.0) GO TO 1013
C
C       ** FORM 120 BY 120 WORKING MATRIX OF AGE DATA **
      DO 1002 IA = 1,120
      DO 1002 JA = 1,60
      AGEMAP(IA,JA) = XLEFT(IA,JA)
      JB = JA + 60
      AGEMAP(IA,JB) = XRIGHT(IA,JA)
1002  CONTINUE
C
C       ** READ IN STREAM DATA FROM MAP CARDS **
      CALL READIN (XLEFT,XRIGHT,IFLAG)
      IF (IFLAG.NE.0) GO TO 1014
C
C       ** FORM 120 BY 120 WORKING MATRIX FROM STREAM DATA **
      DO 1003 IA = 1,120
      DO 1003 JA = 1,60
      ISTRM(IA,JA) = XLEFT(IA,JA)
      IF(ISTRM(IA,JA).EQ.ZERO)  ISTRM(IA,JA) = 0
      IF(ISTRM(IA,JA).EQ.BLANK) ISTRM(IA,JA) = 0
      IF(ISTRM(IA,JA).EQ.ONE)   ISTRM(IA,JA) = 1
      JB = JA + 60
      ISTRM(IA,JB) = XRIGHT(IA,JA)
      IF(ISTRM(IA,JB).EQ.ZERO)  ISTRM(IA,JB) = 0
      IF(ISTRM(IA,JB).EQ.BLANK) ISTRM(IA,JB) = 0
      IF(ISTRM(IA,JB).EQ.ONE)   ISTRM(IA,JB) = 1
1003  CONTINUE
C
C       ** CREATE 120 BY 120 SLOPE AND ASPECT MATRICES **
      DO 400 I = 1,120
      DO 400 J = 1,120
401   CALL SLOPA (I,J,ELEV,CELLHT,CELWTH,ISLOPE,IASPCT,CRFX,IFLAG,DEV)
      SLOPE(I,J) = ISLOPE
```

```
      ASPECT(I,J) = IASPCT
      IF (IFLAG.NE.0) GO TO 1015
400   CONTINUE
C
C      ** READ IN TABLE OF AGE DISTRIBUTIONS **
      DO 500 IRW = 1,15
      READ (5,*) (AGETBL(IRW,ICOL),ICOL=1,6)
500   CONTINUE
C
C      ** MAIN PROGRAM LOOP **
      DO 1000 I=1,120
      DO 1000 J=1,120
C
C      ** AVOID UNASSIGNED VALUES RESULTING FROM BYPASSING MAIN LOOP **
      MASTMT(I,J) = 0
      OPENMT(I,J) = 0
      ROOSMT(I,J) = 0
C
C      ** SKIP CELLS NOT WITHIN AREA OF INTEREST **
      IF((SPPMAP(I,J).EQ.BLANK).OR.(SPPMAP(I,J).EQ.ZERO)) GO TO 1000
C
C      ** CLEAR POINTER ARRAY **
      DO 2000 ICLR = 1,3
      IM(ICLR) = 0
2000  CONTINUE
C
C      ** DETERMINE ROW NUMBER OF VECTOR ARRAY TO FILL **
      IF(SPPMAP(I,J).EQ.SPP1) GO TO 01
      GO TO 02
01    IM(1) = 3
      IM(2) = 5
      IM(3) = 11
      GO TO 22
02    IF(SPPMAP(I,J).EQ.SPP2) GO TO 03
      GO TO 04
03    IM(1) = 1
      GO TO 22
04    IF(SPPMAP(I,J).EQ.SPP3) GO TO 05
      GO TO 06
05    IM(1) = 3
      IM(2) = 5
      GO TO 22
06    IF(SPPMAP(I,J).EQ.SPP4) GO TO 07
      GO TO 08
07    IM(1) = 5
      GO TO 22
08    IF(SPPMAP(I,J).EQ.SPP5) GO TO 09
      GO TO 10
09    IM(1) = 7
      GO TO 22
```

```
10    IF(SPPMAP(I,J).EQ.SPP6) GO TO 11
      GO TO 12
11    IM(1) = 6
      IM(2) = 7
      GO TO 22
12    IF(SPPMAP(I,J).EQ.SPP7) GO TO 13
      GO TO 14
13    IM(1) = 3
      GO TO 22
14    IF(SPPMAP(I,J).EQ.SPP8) GO TO 15
      GO TO 16
15    IM(1) = 2
      GO TO 22
16    IF(SPPMAP(I,J).EQ.SPP9) GO TO 17
      GO TO 18
17    IM(1) = 9
      GO TO 22
18    IF(SPPMAP(I,J).EQ.SPP10) GO TO 19
      GO TO 20
19    IM(1) = 8
      GO TO 22
20    IF(SPPMAP(I,J).EQ.SPP11) GO TO 21
21    IM(1) = 11
      GO TO 22
24    IF(SPPMAP(I,J).EQ.SPP12) GO TO 23
C      ** IF SPPMAP DOESN'T MATCH ANY SYMBOL, GO TO ERROR SECTION **
      GO TO 1010
23    IM(I) = 12
      GO TO 22
C      *** END OF CATEGORIZATION FOR SPECIES **
C
22    CONTINUE
C
C      ** DEFINE THE ROW OF THE AGE DISTRIBUTION TABLE TO BE USED **
C
C      ** SORT MAP SYMBOLS TO DEFINE A TABULAR NUMBER FOR THE AGE CLASS
      IROW = 0
100   IF(AGEMAP(I,J).EQ.AGE1) GO TO 101
      GO TO 102
101   IROW = 1
      GO TO 130
102   IF(AGEMAP(I,J).EQ.AGE2) GO TO 103
      GO TO 104
103   IROW = 2
      GO TO 130
104   IF(AGEMAP(I,J).EQ.AGE3) GO TO 105
      GO TO 106
105   IROW = 3
      GO TO 130
106   IF(AGEMAP(I,J).EQ.AGE4) GO TO 107
```

```
      GO TO 108
107   IROW = 4
      GO TO 130
108   IF(AGEMAP(I,J).EQ.AGE5) GO TO 109
      GO TO 110
109   IROW = 5
      GO TO 130
110   IF(AGEMAP(I,J).EQ.AGE6) GO TO 111
      GO TO 112
111   IROW = 6
      GO TO 130
112   IF(AGEMAP(I,J).EQ.AGE7) GO TO 113
      GO TO 114
113   IROW = 7
      GO TO 130
114   IF(AGEMAP(I,J).EQ.AGE8) GO TO 115
      GO TO 116
115   IROW = 8
      GO TO 130
116   IF(AGEMAP(I,J).EQ.AGE9) GO TO 117
      GO TO 118
117   IROW = 9
      GO TO 130
118   IF(AGEMAP(I,J).EQ.AGE10) GO TO 119
      GO TO 120
119   IROW = 10
      GO TO 130
120   IF(AGEMAP(I,J).EQ.AGE11) GO TO 121
      GO TO 122
121   IROW = 11
      GO TO 130
122   IF(AGEMAP(I,J).EQ.AGE12) GO TO 123
      GO TO 124
123   IROW = 12
      GO TO 130
124   IF(AGEMAP(I,J).EQ.AGE13) GO TO 125
      GO TO 126
125   IROW = 13
      GO TO 130
126   IF(AGEMAP(I,J).EQ.AGE14) GO TO 127
      GO TO 128
127   IROW = 14
      GO TO 130
128   IF(AGEMAP(I,J).EQ.AGE15) GO TO 129
C      ** IF AGEMAP DOESN'T MATCH ANY SYMBOL GO TO ERROR SECTION **
      GO TO 1011
129   IROW = 15
C
C      *** END OF CATEGORIZATION FOR AGE ***
130   CONTINUE
```

```
C
C      *** SECTION FOR CONSTRUCTING SPP/AGE DIST. MATRIX ***
C      *   STRUCTURE OF THE MATRIX IS AS FOLLOWS:            *
C      *     I = SPECIES NUMBER (THERE ARE 16 SPECIES)       *
C      *     VECTOR(I,1) = PERCENT OF CELL OCCUPIED BY THE   *
C      *                    ITH SPECIES                      *
C      *     VECTOR(I,2) = PERCENT OF CELL OCCUPIED BY       *
C      *                    OPEN AREA                        *
C      *     VECTOR(I,3) = PERCENT OF CELL OCCUPIED BY       *
C      *                    TREES 0-2 IN. DBH                *
C      *     VECTOR(I,4) = PRECENT OF CELL OCCUPIED BY       *
C      *                    TREES 2-5 IN. DBH                *
C      *     VECTOR(I,5) = PERCENT OF CELL OCCUPIED BY       *
C      *                    TREES 5-8 IN. DBH                *
C      *     VECTOR(I,6) = PRECENT OF CELL OCCUPIED BY       *
C      *                    TREES GREATER THAN 8 IN. DBH     *
C      *  NOTES:                                             *
C      *    VECTOR(I,2) IS UPDATED ON EACH PASS IN THE PGM   *
C      *      BY SUMMING VECTOR(I,1) AND SUBTRACTING THIS    *
C      *      VALUE FROM 1.0.  THIS DIFFERENCE IS THEN       *
C      *      APPORTIONED AMONG THE SPECIES SUITED TO THE    *
C      *      SITE, AS DETERMINED BY THE TRANS SUBROUTINE.   *
C      *    AN ARRAY OF VECTOR(I,2) THRU VECTOR(I,6) IS      *
C      *      PASSED TO THE UPDATE SUBROUTINE WHERE IT IS    *
C      *      MULTIPLIED BY THE APPROPRIATE MARKOV MATRIX.   *
C      *    VECTOR(I,1) IS UPDATED ON EACH PASS IN THE PGM   *
C      *      BY SETTING IT EQUAL TO THE SUM OF VECTOR       *
C      *      (I,3) THRU VECTOR(I,6)                         *
C
C      ** CLEAR VECTOR ARRAY **
      DO 333 K = 1,16
      DO 333 L = 1,6
      VECTOR(K,L) = 0.0
333   CONTINUE
C
C      ** CALCULATE NUMBER OF SPP IN CELL FROM THE SPPMAP(I,J) DATA **
      SUM = 0.
      DO 444 II = 1,3
      IF (IM(II).GT.0) SUM = SUM + 1.
444   CONTINUE
C
C      ** SET UP APPROPRIATE ROWS IN THE VECTOR WITH AGE DATA **
      DO 555 M = 1,3
      IF (IM(M).EQ.0) GO TO 555
      DO 556 N = 1,6
      VECTOR(IM(M),N) = AGETBL(IROW,N)
556   CONTINUE
C      ** RESET FIRST ENTRY IN VECTOR ARRAY TO PERCENT SPP COMP. **
      VECTOR(IM(M),1) = (1.0-VECTOR(IM(M),2))/SUM
555   CONTINUE
```

```
C
C
C      ** CONVERT % FREQUENCY OF VECTOR TO % OCCURRENCE
      DO 560 IDBH = 1,16
      DO 560 JDBH = 3,6
      VECTOR(IDBH,JDBH) = VECTOR(IDBH,1) * VECTOR(IDBH,JDBH)
560   CONTINUE
C
C     ** CALL PREDICTION PROGRAM **
C      ** NOTE: PREDICTION PROGRAM WILL NOT BE CALLED FOR CELLS **
C      **       OUTSIDE THE MAPPED AREA OF INTEREST             **
C
      IF(SPPMAP(I,J).EQ.BLANK.OR.SPPMAP(I,J).EQ.ZERO) GO TO 1000
      ISLOP = SLOPE(I,J)
      IASPC = ASPECT(I,J)
      XELEV = ELEV(I,J)
      ISTR = ISTRM(I,J)
      CALL PRED(HORIZN,MARKOV,VECTOR,NAME,IFACTR,ISLOP,IASPC,ISTR,XELEV,
     &ICAST,MASTDV,IOPENG,IROOST)
C
C      ** FORM 120 BY 120 ARRAYS OF MAST DIVERSITY, OPENINGS, AND **
C      **          ROOST DATA FROM PREDICTON PROGRAM              **
      MASTMT(I,J) = MASTDV
      OPENMT(I,J) = IOPENG
      ROOSMT(I,J) = IROOST
C
1000  CONTINUE
C
C      ** CARD GENERATION SECTION **
      WRITE(7,701) HORIZN
701   FORMAT(1X,'MAST DIVERSITY CARDS AT ',I3,' TIMES 5 YEARS NO CUT ')
      CALL CARDS (MASTMT)
C
      WRITE(7,702) HORIZN
702   FORMAT(1X,'OPENINGS CARDS AT ',I3,' TIMES 5 YEARS NO CUTTING')
      CALL CARDS (OPENMT)
C
      WRITE(7,703) HORIZN
703   FORMAT(1X,'ROOST SITE CARDS AT', I3,' TIMES 5 YEARS NO CUT ')
      CALL CARDS (ROOSMT)
C
      GO TO 9999
C
C      ** ERROR SECTION **
1010  WRITE(6,99) I,J,SPPMAP(I,J)
99    FORMAT (1X,'ERROR: CELL ',I3,1X,I3,' HAS A VALUE OF (',A1,') AND C
     &CANNOT BE ASSIGNED A SPECIES COMPONENT')
      GO TO 9999
1011  WRITE(6,98) I,J,SPPMAP(I,J)
98    FORMAT (1X,'ERROR: CELL ',I3,1X,I3,' HAS A VALUE OF (',A1,') AND C
```

```
     &ANNOT BE ASSIGNED AN AGE COMPONENT')
      GO TO 9999
1012  WRITE(6,97)
97    FORMAT(1X,'MAIN PROGRAM HALTED DUE TO INPUT ERROR WHILE READING SP
     &ECIES DATA MAP')
1013  WRITE(6,96)
96    FORMAT(1X,'MAIN PROGRAM HALTED DUE TO INPUT ERROR WHILE READING AG
     &E DATA MAP')
      GO TO 9999
1014  WRITE(6,95)
95    FORMAT(1X,'MAIN PROGRAM HALTED DUE TO INPUT ERROR WHILE READING ST
     &REAM DATA')
      GO TO 9999
1015  WRITE(6,94)
94    FORMAT(1X,'MAIN PROGRAM HALTED DUE TO ERROR IN SUBROUTINE SLOPE')
C
9999  STOP
      END
      SUBROUTINE READIN (XLEFT,XRIGHT,IFLAG)
C      *********************************************
C      *** THIS SUBROUTINE READS IN AND CHECKS    ***
C      ***      SYMBOL OR NUMERIC MAPS            ***
C      *** INPUT -  1) MAP NAME,  COLS. 1-20      ***
C      ***          2) (MAP DATA)                 ***
C      *********************************************
C      *** NOTE: LESS THAN 240 MAP CARDS CAN BE   ***
C      ***  READ USING THE SHORTCUT PROCEDURE     ***
C      *********************************************
      INTEGER*2 XLEFT(120,60), XRIGHT(120,60)
      INTEGER MAPNAM(5)
      INTEGER RNUMBR, R2
      INTEGER CHEKL, CHEKR
C      ** READ THE MAP NAME **
      READ(4,111) MAPNAM
111   FORMAT(5A4)
C      *****LEFT SIDE READ LOOP*****
      L=1
      CHEKL=1
101   READ(4,01) LNUMBR,L2,(XLEFT(L,K),K=1,60)
01    FORMAT(2X,I3,I3,60A1)
      IF(CHEKL.NE.LNUMBR) GO TO 991
      IF(L2.EQ.0) GO TO 100
      N=L2-LNUMBR
      DO 99 J=1,N
      L=L+1
      CHEKL=CHEKL+1
      BACKSPACE 4
99    READ(4,011) (XLEFT(L,K),K=1,60)
011   FORMAT(8X,60A1)
100   L=L+1
```

```
      CHEKL=CHEKL+1
      IF(L.GT.120) GO TO 102
      GO TO 101
C      *****RIGHT SIDE READ LOOP*****
102   L=1
      CHEKR=1
103   READ(4,02)(XRIGHT(L,K),K=1,60), RNUMBR, R2
02    FORMAT(1X,60A1,1X,I3,I3)
      IF(CHEKR.NE.RNUMBR) GO TO 992
      IF(R2.EQ.0) GO TO 200
      N=R2-RNUMBR
      DO 199 J=1,N
      L=L+1
      CHEKR=CHEKR+1
      BACKSPACE 4
199   READ(4,022)(XRIGHT(L,K),K=1,60)
022   FORMAT(1X,60A1)
200   L=L+1
      CHEKR=CHEKR+1
      IF(L.GT.120) GO TO 201
      GO TO 103
201   CONTINUE
      GO TO 999
991   WRITE (6,901) CHEKL, MAPNAM
      IFLAG=10
      GO TO 999
992   WRITE (6,902) CHEKR, MAPNAM
      IFLAG=10
901   FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED. CARD ',I3,' ON LEFT
     &OUT OF ORDER ON MAP NAME ',5A4)
902   FORMAT(1H ,'ERROR: SUBROUTINE READIN ABORTED, CARD ',I3,' ON RIGHT
     & OUT OF ORDER ON MAP NAME ',5A4)
999   CONTINUE
      RETURN
      END
      SUBROUTINE SLOPA(M,N,X,CELLHT,CELWTH,ISLOPE,IASPCT,CRFX,IFLAG,DEV)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ** INPUT VARIABLE LIST (FROM CALLING PROGRAM):                   **
C      **          M = ROW NUMBER OF CELL                               **
C      **          N = COLUMN NUMBER OF CELL                            **
C      **          X = ARRAY OF ELEVATIONS                              **
C      **     CELLHT = HEIGHT OF CELL (DIMENSION OF CODING GRID)        **
C      **     CELWTH = WIDTH OF CELL (DIMENSION OF CODING GRID)         **
C      **       CRFX = CORRECTION FACTOR FOR CONVERTING CELL DIMENSIONS **
C      **                  INTO ON-THE-GROUND MEASUREMENTS IN THE SAME  **
C      **                  SAME UNITS AS ELEVATION (METERS OR FEET)     **
C      **    OUTPUT VARIABLES:                                          **
C      **     ASPECT = WEIGHTED AVERAGE OF GREATEST DOWNHILL DIRECTION  **
C      **                 AND OPPOSITE ANGLE OF GREATEST UPHILL DIRECTION **
C      **                 (EXPRESSED IN DEGREES)                        **
```

```
C       **    SLOPE = AVERAGE OF GREATEST ELEVATIONAL DIFFERENCE AND    **
C       **              ELEVATION OF TARGET CELL (EXPRESSED IN DEGREES) **
C       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      INTEGER XLIM,YLIM
      INTEGER*2 ISLOPE, IASPCT
      DIMENSION THETA (9), ELEV (9), X (120,120)
      REAL MAXUP,MINDN
C
C       ** INITIALIZATION OF MAP SIZE LIMITS **
      XLIM = 120
      YLIM = 120
C
C       CHECK M AND N FOR IMPROPER VALUES
      IF (M.LE.0.OR.M.GT.YLIM) GO TO 110
      IF (N.LE.0.OR.N.GT.XLIM) GO TO 112
C
C       CALCULATE ANGLES TO EACH OF THE ADJACENT 8 CELLS
C         NOTE: THETA(5) IS A DUMMY
C        THETA'S ARE NUMBERED CONSECUTIVELY, LEFT TO RIGHT BEGINNING AT
C         TOP LEFT...
C
      ANGLE = (ATAN(CELLHT/CELWTH))*57.295773
      THETA(2) = 360.
      THETA(4) = 270.
      THETA(6) = 90.
      THETA(8) = 180.
      THETA(1) = THETA(4)+ANGLE
      THETA(3) = THETA(6)-ANGLE
      THETA(5) = 0.
      THETA(7) = THETA(4)-ANGLE
      THETA(9) = THETA(6)+ANGLE
C
C       READ ELEVATIONS LEFT TO RIGHT BEGINNING WITH NW CORNER
C
C       TOP ROW SCAN
      IM = M-1
      IN = N-1
      DO 10 I=1,3
C       CHECK FOR OVERRUN
      IF (IM.LE.0.OR.IN.LE.0.OR.IN.GT.XLIM) GO TO 08
      ELEV(I) = X(IM,IN)
      GO TO 09
08    ELEV(I) = X(M,N)
09    IN = IN+1
10    CONTINUE
C
C       MIDDLE ROW SCAN
      IM = M
      IN = N-1
```

```
      DO 20 I=4,6
C      CHECK FOR OVERRUN
      IF (IN.LE.0.OR.IN.GT.XLIM) GO TO 18
      ELEV(I) = X(IM,IN)
      GO TO 19
18    ELEV(I) = X(M,N)
19    IN = IN+1
20    CONTINUE
C
C      BOTTOM ROW SCAN
      IM = M+1
      IN = N-1
      DO 30 I=7,9
C      CHECK FOR OVERRUN
      IF (IN.LE.0.OR.IN.GT.XLIM.OR.IM.GT.YLIM) GO TO 28
      ELEV(I) = X(IM,IN)
      GO TO 29
28    ELEV(I) = X(M,N)
29    IN = IN+1
30    CONTINUE
C
C      DETERMINE MAXIMUM ELEVATION AND LOCATION (IHIGH)
C
      HTMAX = ELEV(5)
      IHIGH = 5
      DO 40 I=1,9
      IF (ELEV(I).GT.HTMAX) GO TO 41
      GO TO 40
41    HTMAX = ELEV(I)
      IHIGH = I
40    CONTINUE
C
C      DETERMINE MINIMUM ELEVATION AND LOCATION (ILOW)
C
      HTMIN = ELEV(5)
      ILOW = 5
      DO 50 I=1,9
      IF (ELEV(I).LT.HTMIN) GO TO 51
      GO TO 50
51    HTMIN = ELEV(I)
      ILOW = I
50    CONTINUE
C
C      ***** SLOPE COMPUTATION SECTION *****
C
C      COMPUTE SLOPE, WHERE SLOPE IS THE AVERAGE OF THE DIFFERENCES
C       BETWEEN THE CENTRAL ELEVATION AND THE 2 EXTREME ELEVATIONS
C
      IF (IHIGH.EQ.1.OR.IHIGH.EQ.3.OR.IHIGH.EQ.7.OR.IHIGH.EQ.9) DISTUP=
     &(SQRT(CELLHT**2+CELWTH**2))*CRFX
```

```
      IF (IHIGH.EQ.2.OR.IHIGH.EQ.8) DISTUP=CELLHT*CRFX
      IF (IHIGH.EQ.4.OR.IHIGH.EQ.6) DISTUP=CELWTH*CRFX
      IF (ILOW.EQ.1.OR.ILOW.EQ.3.OR.IHIGH.EQ.7.OR.ILOW.EQ.9) DISTDN =
     &(SQRT(CELLHT**2+CELWTH**2))*CRFX
      IF (ILOW.EQ.2.OR.ILOW.EQ.8) DISTDN=CELLHT*CRFX
      IF (ILOW.EQ.4.OR.ILOW.EQ.6) DISTDN=CELWTH*CRFX
C
C        CREATE A DUMMY VALUE FOR KNOB AND/OR BOWL SITUATION
C
      IF (IHIGH.EQ.5) DISTUP=1.
      IF (ILOW.EQ.5.) DISTDN=1.
C
C       CALCULATE UPHILL AND DOWNHILL SLOPE ANGLES
C
      UPSLOP = ATAN((ELEV(IHIGH)-ELEV(5))/DISTUP)*57.295773
      DNSLOP = ATAN((ELEV(5)-ELEV(ILOW))/DISTDN)*57.295773
C
C       **** ACCOMMODATE BOWL OR KNOB SITUATION ****
      IF (IHIGH.EQ.5) GO TO 68
      IF (ILOW.EQ.5) GO TO 69
C       DETERMINE SLOPE ANGLE
      SLOPE = (UPSLOP+DNSLOP)/2.
C
      GO TO 70
68    SLOPE = DNSLOP
      GO TO 70
69    SLOPE = UPSLOP
70    CONTINUE
C       ACCOMMODATE FOR A HORIZONTAL PLANE
      IF (IHIGH.EQ.5.AND.ILOW.EQ.5) SLOPE=0.
C
C       ** CREATE AN INTEGER*2 VARIABLE FOR PASSING SLOPE BACK TO MAIN **
      ISLOPE = SLOPE + 0.5
C
C     ***** ASPECT COMPUTATION SECTION *****
C
C       SET MAXIMUM UP-SLOPE AZIMUTH
      MAXUP = THETA(IHIGH)
C
C       SET MINIMUM DOWN-SLOPE AZIMUTH
      MINDN = THETA(ILOW)
C
C       ACCOMMODATE FOR KNOB SHAPE (I.E. ELEV5 IS HIGHEST)
      IF (IHIGH.EQ.5) GO TO 58
C
C       SET AN AZIMUTH 180 DEGREES FROM MAXIMUM UP-SLOPE
      UPOPP = MAXUP + 180.
C
C       ACCOMMODATE FOR BOWL SHAPE (I.E. ELEV5 IS LOWEST)
      IF (ILOW.EQ.5) GO TO 59
```

```
C
C        ACCOMMODATE FOR A PLANE
      IF (MINDN-UPOPP.GT.-.000001.AND.MINDN-UPOPP.LT..00001) GO TO 58
C
C        CREATE WEIGHTING FACTORS BASED ON SLOPE RATIOS
C
      SUMSLP = UPSLOP + DNSLOP
      UPRTIO = UPSLOP/SUMSLP
      DNRTIO = DNSLOP/SUMSLP
C
C        CALCULATE ASPECT BASED ON ELEVATIONAL (SLOPE) WEIGHTS
C
      CALL VECADD (NINDN,DNRTIO,UPOPP,UPRTIO,ASPECT,XMAG)
C
      GO TO 60
58    ASPECT = MINDN
      GO TO 60
59    ASPECT = UPOPP
60    CONTINUE
C
C        ACCOMMODATE FOR A HORIZONTAL PLANE
61    IF (IHIGH.EQ.5.AND.ILOW.EQ.5) ASPECT = 0.
C
C        ** CORRECT FOR TILTED BASE MAP **
      ASPECT = ASPECT - DEV
C
C        ACCOMMODATE FOR NEGATIVE ANGLES
      IF (ASPECT.LT.0.) ASPECT = ASPECT + 360.
C
C        ACCOMMODATE FOR ANGLES GREATER THAN 360 DEGREES
      IF (ASPECT.GT.360.) ASPECT = ASPECT - 360.
C
C        ** CREATE AN INTEGER*2 VARIABLE FOR PASSING ASPECT BACK TO MAIN *
      IASPCT = ASPECT + 0.5
C
C        ** ERROR SECTION **
      GO TO 999
110   WRITE (6,111) M
111   FORMAT (1X,'SUBROUTINE SLOPE ABORTED... M HAS VALUE OF',I3)
      IFLAG = 999
      GO TO 999
112   WRITE (6,113) N
113   FORMAT (1X,'SUBROUTINE SLOPE ABORTED... N HAS VALUE OF',I3)
      IFLAG = 999
999   CONTINUE
      RETURN
      END
      SUBROUTINE VECADD(Q1,D1,Q2,D2,Q3,D3)
C        ** SUBROUTINE FOR PERFORMING VECTOR ADDITION **
      CALL PRIANG(Q1,P1,DX,DY)
```

```
      RP1=P1/57.295773
      Y1=DY*SIN(RP1)*D1
      X1=DX*COS(RP1)*D1
      CALL PRIANG(Q2,P2,DX,DY)
      RP2=P2/57.295773
      Y2=DY*SIN(RP2)*D2
      X2=DX*COS(RP2)*D2
      YDIFF=Y1+Y2
      XDIFF=X1+X2
      IF(XDIFF.LT.0.00001.AND.XDIFF.GT.-.00001) XDIFF=.00001
      IF(YDIFF.LT.0.00001.AND.YDIFF.GT.-.00001) YDIFF=.00001
      D3=SQRT(XDIFF*XDIFF+YDIFF*YDIFF)
      RP3=ATAN(YDIFF/XDIFF)
      Q3=RP3*57.295773
      WRITE(6,60) X1,Y1,X2,Y2,Q3,XDIFF,YDIFF
      IF(XDIFF.GT.0.) Q3=90.-Q3
      IF(XDIFF.LT.0.) Q3=270.-Q3
C     IF (ABS(Q1-Q2).EQ.180.) Q3=Q1+90.
      IF(Q3.GT.360.) Q3=Q3-360.
      IF(Q3.LT.0.) Q3=360.-Q3
 60    FORMAT(5F20.5)
      RETURN
      END
      SUBROUTINE PRIANG(Q,P,DX,DY)
C      **  SUBROUTINE FOR CALCULATING PRIMARY ANGLE   **
      THETA=Q
      J=(THETA-.0001)/90.
      IF(THETA.LE.90.) ANG=90.-THETA
      IF(THETA.GT.90.) ANG=THETA-90.
      IF(THETA.GT.180.) ANG=270.-THETA
      IF(THETA.GT.270.) ANG=THETA-270.
      P=ANG
      DY=1.
      DX=1.
      IF(J.EQ.1.OR.J.EQ.2) DY=-DY
      IF(J.EQ.2.OR.J.EQ.3) DX=-DX
      RETURN
      END
      SUBROUTINE PRED(HORIZN,MARKOV,VECTOR,NAME,IFACTR,SLOPE,ASPECT,ISTR
     &M,ELEV,ICAST,MASTDV,IOPENG,IROOST)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      *               SUBROUTINE NAME:  PRED                 *
C      *  THIS SUBROUTINE UPDATES A VEGETATION MATRIX USING   *
C      *  A SERIES OF MARKOV MATRICES TO PREDICT THE STATUS   *
C      *  OF PORTIONS OF A CELL.  CELL PORTIONS REACHING A    *
C      *   A MORTALITY COMPONENT ARE REASSIGNED WITH EQUAL    *
C      *   PROBABILITY AMONG THOSE SPECIES DETERMINED TO BE   *
C      *  SUITED TO THE PARTICULAR EDAPHIC CONDITIONS WITHIN *
C      *            THAT CELL.                                *
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C
      DIMENSION TRANST(16), VECTOR(16,6)
      REAL MARKOV(16,5,5)
      INTEGER HORIZN, NAME(16,15)
      INTEGER*2 SLOPE,ASPECT,ISTRM,MASTDV,IOPENG,IROOST
      INTEGER*2 IFACTR(24,16)
C
      ICNTX = 999
C
      GO TO 15
C      *** TEST OUTPUT OF INITIAL CONDITIONS ***
666   WRITE(6,01) HORIZN
01    FORMAT(1X,'HORIZON = ',I4)
      WRITE(6,02) SLOPE
02    FORMAT(1X,'SLOPE    =',I4,' DEGREES')
      WRITE(6,03) ASPECT
03    FORMAT(1X,'ASPECT   =',I4,' DEGREES')
      WRITE(6,04) ELEV
04    FORMAT(1X,'ELEVATION =',F6.0,' FEET')
      WRITE(6,05) ISTRM
05    FORMAT(1X,'STREAM CATEGORY (1=PRESENT) =',I2)
      WRITE(6,06) ICAST
06    FORMAT(1X,'CATASTROPHE CATEGORY =',I2,' (1=YES;2=NO')
      WRITE(6,07)
07    FORMAT(1X,'  INITIAL CONDITIONS MATRIX')
      DO 10 IJ=1,16
      WRITE(6,08) (VECTOR(IJ,IK),IK=1,6)
08    FORMAT(1X,6F12.5)
10    CONTINUE
15    CONTINUE
C
C     **(INTERATE HERE TO 101 ONCE FOR EACH UNIT IN PLANNING HORIZON)**
C
      ITIME=0
      DO 101 IHOR=1,HORIZN
C
C      CALCULATE AMOUNT OF OPEN AREA IN THE CELL
C
      SUMV1 = 0.0
      DO 25 I=1,16
      SUMV1 = SUMV1 + VECTOR(I,1)
25    CONTINUE
      OPENS = 1.0 - SUMV1
C
C       SECTION FOR CREATING A VARIABLE CONTAINING OPENS+DBH1
      SUMV2 = 0.
      SUMV2A = 0.
      DO 26 I=1,16
      SUMV2 = SUMV2 + (VECTOR(I,3)/2.)
      SUMV2A = SUMV2A + VECTOR(I,3)
```

```
26      CONTINUE
        OPENX = OPENS + SUMV2
        OPENY = OPENS + SUMV2A
C        WRITE (6,27) OPENX
C27      FORMAT (1X,'OPENX =',F10.6)
C        WRITE (6,28) OPENS,OPENY
C28      FORMAT (1X,'PERCENT OPEN AREA =',F10.6,'  OPENS+DBHA =',F10.6)
C
C        CALCULATE TRANSITION PROBABILITIES FOR GOING FROM OPEN TO SPECIES
C
        CALL TRANS (TRANST,SLOPE,ISTRM,ICAST,ELEV,OPENX,ASPECT,IFACTR)
C
C        APPORTION OPENS TO VARIOUS SPECIES GROUPS
C
        DO 30 I=1,16
        VECTOR(I,2) = OPENS*TRANST(I)
30      CONTINUE
C
C        UPDATE THE VEGETATION DATA MATRIX (I.E., VECTOR)
C
        CALL UPDATE(VECTOR, MARKOV)
C
C       CALL CUT (VECTOR,ICNTX)
C
C        UPDATE SPECIES COMPOSITION COLUMN
C
        DO 40 IZ=1,16
        VECTOR(IZ,1) = VECTOR(IZ,3)+VECTOR(IZ,4)+VECTOR(IZ,5)+VECTOR(IZ,6)
40      CONTINUE
C
C        RESET CATASTROPHE CATEGORIZATION AFTER 2 PASSES (10 YEARS)
        ITIME = ITIME + 5
        IF (ICNTX.LE.2) ICAST = 1
        ICNTX = ICNTX + 1
C
C        SKIP OUTPUT
C
C        TEST SECTION FOR OUTPUTTING UPDATED VEGETATION DATA
C        WRITE(6,100) ITIME
C100      FORMAT('1','DATA FOR TIME ZERO PLUS ',I3,' YEARS')
C        DO 88 IVEC=1,16
C        WRITE(6,103) (NAME(IVEC,INAM),INAM=1,15)
C103      FORMAT(1X,'SPP=',15A4)
C        WRITE(6,104)(VECTOR(IVEC,J),J=1,6)
C104      FORMAT(1X,6F11.7)
88      CONTINUE
C
101     CONTINUE
C        *** END MAIN LOOP ***
C
```

```
C      ** SORT RESULTS PRIOR TO RETURNING VALUES TO MAIN PROGRAM **
C
C      ** DETERMINE NUMBER OF MAST PRODUCING SPECIES PRESENT IN CELL **
      ISMAST = 0
      DO 200 IMAST = 1,6
      IF(VECTOR(IMAST,5)+VECTOR(IMAST,6).GE..02) ISMAST = ISMAST + 1
200   CONTINUE
C
C     ** STRATIFY NUMBER OF MAST PRODUCING SPECIES INTO CATEGORIES **
      IF (ISMAST.LE.0) MASTDV = 0
      IF (ISMAST.EQ.1) MASTDV = 2
      IF (ISMAST.EQ.2) MASTDV = 3
      IF (ISMAST.EQ.3) MASTDV = 5
      IF (ISMAST.EQ.4) MASTDV = 7
      IF (ISMAST.GE.5) MASTDV = 9
C
C     ** DETERMINE IF CELL CAN BE CONSIDERED TO BE AN OPENING **
      IOPENG = 0
      IF (OPENY.GE..70) IOPENG = 1
C
C      ** DETERMINE IF CELL CONTAINS ROOST TREE SPP. GT 2 INCHES DBH **
      IROOST = 0
      SUMRST = VECTOR(7,4)+VECTOR(7,5)+VECTOR(7,6)+VECTOR(8,4)+VECTOR(8,
     &5)+VECTOR(8,6)+VECTOR(9,4)+VECTOR(9,5)+VECTOR(9,6)+VECTOR(10,4)+
     &VECTOR(10,5)+VECTOR(10,6)+VECTOR(16,4)+VECTOR(16,5)+VECTOR(16,6)
      IF (SUMRST.GE..20) IROOST = 1
C
99    RETURN
      END
      SUBROUTINE MATMUL (A,B,R,N,M,L)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      ***       SUBROUTINE MATMUL (ALSO GMPRD IN SSP PACKAGE)         ***
C      *** PURPOSE: TO MULTIPLY 2 GENERAL MATRICES- A AND B, TO FORM ***
C      ***             A RESULTANT MATRIX, R                           ***
C      *** USAGE:  CALL MATMUL (A,B,R,N,M,L)                           ***
C      *** PARAMETERS:                                                 ***
C      ***     A - NAME OF FIRST INPUT MATRIX (REAL)                   ***
C      ***     B - NAME OF SECOND INPUT MATRIX (REAL)                  ***
C      ***     R - NAME OF OUTPUT MATRIX (REAL)                        ***
C      ***     N - NUMBER OF ROWS IN MATRIX A (INTEGER)                ***
C      ***     M - NUMBER OF COLUMNS IN A AND ROWS IN B (INTEGER)      ***
C      ***     L - NUMBER OF COLUMNS IN B                              ***
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION A(1), B(1), R(1)
      IR = 0
      IK = -M
      DO 10 K=1,L
      IK = IK+M
      DO 10 J=1,N
      IR = IR+1
```

```
      JI = J-N
      IB = IK
      R(IR) = 0
      DO 10 I=1,M
      JI = JI+N
      IB = IB+1
10    R(IR) = R(IR) + A(JI) * B(IB)
      RETURN
      END
C
      SUBROUTINE UPDATE (VECTOR, MARKOV)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      *    THIS SUBROUTINE UPDATES THE AGE VECTORS FROM   *
C      *             THE PRED SUBROUTINE                    *
C      *  INPUT REQUIRED: NONE                              *
C      * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION DBH(16,5), DBHVEC(5), R(5)
      DIMENSION VECTOR(16,6), MARKML(5,5), MARKOV(16,5,5)
C
C      CREATE WORKING MATRIX OF DBH DISTRIBUTIONS
      DO 10 I = 1,16
      DO 10 J = 2,6
      K = J-1
      DBH(I,K) = VECTOR(I,J)
10    CONTINUE
C
C      CREATE A DBH VECTOR FOR EACH SPECIES(I)
C
      DO 20 I = 1,16
C
C      SET UP A DBH VECTOR
      DO 15 J = 1,5
      DBHVEC(J) = DBH(I,J)
C
C      SET UP THE ACTUAL MARKOV MATRIX (I.E. MARKML)
      DO 15 K = 1,5
      MARKML(J,K) = MARKOV(I,J,K)
15    CONTINUE
C
C     PERFORM MATRIX MULTIPLICATION
      CALL MATMUL (DBHVEC,MARKML,R,1,5,5)
C
C      UPDATE VECTOR
      DO 16 IR = 1,5
C      ****** SET ANY DBH4 VALUE LESS THAT .0002 TO ZERO ******
      IF (VECTOR(I,6).LT..0002) VECTOR(I,6) = 0.
      IV = IR +1
      VECTOR(I,IV) = R(IR)
16    CONTINUE
C
```

```
20    CONTINUE
      RETURN
      END
C
      SUBROUTINE TRANS (TRANSM,SLOPE,ISTRM,ICAST,ELEV,OPENS,ASPECT,
     &IFACTR)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      *  SUBROUTINE FOR DETERMINING TRANSITION PROBABILITIES FOR  *
C      *       OPEN LAND BEING COLONIZED BY THE ITH SPECIES         *
C      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION TRANSM(16)
      INTEGER*2 SLOPE, ASPECT, ISTRM, IFACTR(24,16)
      INTEGER ILIST(16), ICAST
C
C      SLOPE CLASS SECTION
      IF (SLOPE.LE.5.) ISLPCL = 1
      IF (SLOPE.GT.5..AND.SLOPE.LE.10.) ISLPCL = 2
      IF (SLOPE.GT.10..AND.SLOPE.LE.20.) ISLPCL = 3
      IF (SLOPE.GT.20..AND.SLOPE.LE.45.) ISLPCL = 4
      IF (SLOPE.GT.45..AND.SLOPE.LE.65.) ISLPCL = 5
      IF (SLOPE.GT.65.) ISLPCL = 6
C
C      STREAM CLASS SECTION
      IF (ISTRM.NE.0) ISTRCL = 7
      IF (ISTRM.EQ.0) ISTRCL = 8
C
C      CATASTROPHE CLASS SECTION
      ICSTCL = ICAST+8
C
C      ELEVATION CLASS SECTION
      IF (ELEV.LE.1000.) IELVCL = 11
      IF (ELEV.GT.1000..AND.ELEV.LE.2000.) IELVCL = 12
      IF (ELEV.GT.2000..AND.ELEV.LE.3000.) IELVCL = 13
      IF (ELEV.GT.3000..AND.ELEV.LE.4000.) IELVCL = 14
      IF (ELEV.GT.4000..AND.ELEV.LE.5000.) IELVCL = 15
      IF (ELEV.GT.5000.) IELVCL = 16
C
C      PERCENT OPENINGS SECTION
      IF (OPENS.LE..1) IOPNCL = 17
      IF (OPENS.GT..1.AND.OPENS.LE..3) IOPNCL = 18
      IF (OPENS.GT..3.AND.OPENS.LE..6) IOPNCL = 19
      IF (OPENS.GT..6) IOPNCL = 20
C
C      ASPECT CLASS SECTION
      IF (ASPECT.EQ.360) ASPECT = 0.
      IF (ASPECT.LE.90.) IASPCL = 21
      IF (ASPECT.GT.90..AND.ASPECT.LE.180.) IASPCL = 22
      IF (ASPECT.GT.180..AND.ASPECT.LT.270.) IASPCL = 23
      IF (ASPECT.GT.270.) IASPCL = 24
C
```

```
C      SUM FACTORS BY SPECIES
C
      DO 10 J=1,16
      ISUM = IFACTR(ISLPCL,J)+IFACTR(ISTRCL,J)+IFACTR(ICSTCL,J)+
     & IFACTR(IELVCL,J)+IFACTR(IOPNCL,J)+IFACTR(IASPCL,J)
      ILIST(J)=0
      IF (ISUM.EQ.6) ILIST(J) = 1
C     WRITE (6,11) J, ISUM
C11    FORMAT(1X,'ISUM (',I1,') =',I2)
10    CONTINUE
C
C      CALCULATE EQUAL PROBABILITY DISTRIBUTION BASED ON NUMBER
C               OF SPECIES REMAINING IN THE LIST ARRAY
C
C      SUM THE LIST ARRAY
      ISM = 0
      DO 20 K=1,16
      ISM = ISM+ILIST(K)
20    CONTINUE
C
C      PREVENT DIVISION BY ZERO
      IF (ISM.EQ.0) GO TO 30
      PROB = 1./ISM
      GO TO 40
30    PROB = 0.
40    CONTINUE
C
C      DISTRIBUTE PROBABILITIES AND CREATE TRANSITION PROBABILITY ARRAY
C
C     CLEAR ARRAY
      DO 50 IP=1,16
      TRANSM(IP) = PROB
      IF(ILIST(IP).EQ.0) TRANSM(IP) = 0.
50    CONTINUE
C
60    CONTINUE
C
C      SKIP OUTPUT
C      WRITE (6,102)
C102   FORMAT(4X,'ISLPCL ISTRCL ICSTCL IELVCL IOPNCL IASPCL')
C      WRITE (6,103) ISLPCL,ISTRCL,ICSTCL,IELVCL,IOPNCL,IASPCL
C103   FORMAT(1X,6I7)
C      WRITE (6,104) (ILIST(I),I=1,16)
C104   FORMAT(1X,'ILIST ARRAY: ',16I3)
C      WRITE (6,105) (TRANSM(I),I=1,16)
C105   FORMAT(1X,'TRANSITION PROBS:',16F7.4)
99    RETURN
      END
      SUBROUTINE CUT (VECTOR,ICNT)
C      * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C       *             SUBROUTINE NAME:  CUT                          *
C       *  THIS SUBROUTINE CHECKS TO DETERMINE IF A CELL             *
C       *  COULD BE HARVESTED.  IF 60% OR MORE OF THE CELL           *
C       *  IS IN THE 8 INCH DBH CLASS, THE CONTENTS OF THE           *
C       *  CELL ARE SET TO OPEN AREA, AND THE CATASTROPHE            *
C       *  VALUE IS SET TO THE YES CONDITION.                        *
C       *    INPUT REQUIRED: NONE                                    *
C       * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION VECTOR(16,6)
C
C       ** DETERMINE PERCENT OF CELL GREATER THAN 8 INCHES DBH **
      SUMV6 = 0.
      DO 10 I=1,16
      SUMV6 = SUMV6 + VECTOR(I,6)
10    CONTINUE
C
C       ** IF 60% OF THE CELL IS .GT. 8 IN.DBH, CUT THE CELL **
      IF (SUMV6.GE..60) GO TO 20
      GO TO 30
20    DO 25 I=1,16
      VECTOR(I,2)=VECTOR(I,2)+VECTOR(I,4)+VECTOR(I,5)+VECTOR(I,6)
      VECTOR (I,4) = 0.
      VECTOR (I,5) = 0.
      VECTOR (I,6) = 0.
25    CONTINUE
      ICNT = 1
30    CONTINUE
      RETURN
      END
      SUBROUTINE CARDS (X)
C       * * * * * * * * * * * * * * * * * * * * * * * * * *
C       *         SUBROUTINE NAME:  CARDS                       *
C       * THIS SUBROUTINE PRODUCES MAP4B-COMPATIBLE CARDS *
C       *  NO INPUT REQUIRED, ALL INPUT FROM MAIN PROGRAM *
C       * * * * * * * * * * * * * * * * * * * * * * * * * *
      INTEGER*2 X(120,120)
C
C       ** LEFT SIDE CARD OUTPUT LOOP **
      DO 10 I = 1,120
      WRITE(7,01) I, (X(I,J),J=1,60)
01    FORMAT(2X,I3,3X,60I1)
10    CONTINUE
C
C       ** RIGHT SIDE CARD OUTPUT LOOP **
      DO 20 K = 1,120
      WRITE(7,02) (X(K,L),L=61,120), K
02    FORMAT(1X,60I1,1X,I3)
20    CONTINUE
      RETURN
      END
```

Appendix D

List of Common and Scientific Names of Trees

COMMON NAME	SCIENTIFIC NAME
Chestnut oak	(Quercus prinus)
Scarlet oak	(Q. coccinea)
White oak	(Q. alba)
Black oak	(Q. velutina)
Northern red oak	(Q. rubra)
Bear oak	(Q. ilicifolia)
Pitch pine	(Pinus rigida)
Virginia pine	(P. virginiana)
Table Mountain pine	(P. pungens)
White pine	(P. strobus)
Yellow poplar	(Liriodendron tulipifera)
Black locust	(Robinia psuedoacacia)
Gray birch	(Betula populifolia)
Sycamore	(Plantanus occidentalis)
Red maple	(Acer rubrum)
Hemlock	(Tsuga canadensis)

The two page vita has been removed from the scanned document. Page 1 of 2

The two page vita has been removed from the scanned document. Page 2 of 2

DYNAMIC CLASSIFICATION: CONCEPTUAL DEVELOPMENT AND APPLICATIONS IN WILDLIFE MANAGEMENT

by

James Freeman Williamson, Jr.

(ABSTRACT)

Information is a prerequisite for effective management of wildlife habitat. In response to the need for management-oriented information regarding the suitability of an area as wildlife habitat, a new methodology was developed and demonstrated. This methodology involved the conceptual development of a dynamic classification approach.

The proposed methodology sought to avoid many of the problems inherent in conventional classification due to the inflexibility of the latter technique. Using dynamic classification methodology, an entity is described with respect to 1 or more attribute axes relevant to the objectives of the specific classification effort. Attribute axes may represent naturally occurring (i.e., physical and theoretically empirical) attributes or synthetic attributes such as suitability for some purpose.

To demonstrate conceptual utility, an original computerized cellular mapping system was developed to display information graphically. Maps of habitat suitability and other habitat-related information were produced for a total of 5 wildlife species on 3 study areas in Virginia. Within this demonstration, levels of habitat attributes subject to change over time were estimated from forest stand data predicted with a modified Markov chain algorithm. Performance of the prediction program was determined from a 40 year hindcast procedure. Conceptual validity of a dynamic classification approach was examined using epistemological arguments.

The computer mapping package was found to be an effective vehicle for displaying information derived from dynamic classification. The vegetation prediction system appeared to be a feasible technique for predicting certain wildlife habitat attributes which are dynamic over time. It was concluded that dynamic classification was conceptually valid and is an effective methodology for producing information specific to the objectives of a given classification effort. Although conclusions were based on an application of dynamic classification in a wildlife management context, it was speculated that the overall concept of dynamic classification may have additional utility in other fields within the natural sciences.