CANADIAN THESES

THÈSES CANADIENNES

THE UNIVERSITY OF ALBERTA

Effects of Cartographic Error on Polygon Overlay Results with Geomorphic Data

by

James W. F. Smith

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF Master of Science

Geography

EDMONTON, ALBERTA

Fall 1986

ISBN 0-315-32305-1

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR James W. F. Smith

TITLE OF THESIS Effects of Cartographic Error on Polygon Overlay Results with Geomorphic Data

DEGREE FOR WHICH THESIS WAS PRESENTED Master of Science

YEAR THIS DEGREE GRANTED Fall 1986

Permission is hereby granted to THE UNIVERSITY OF ALBERTA LIBRARY to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

(SIGNED) James W. F. Smith

PERMANENT ADDRESS:

5251 TERI ST.

ZACHARY, LOUISIANA

USA 70791

DATED July 21 1986

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled Effects of Cartographic Error on Polygon Overlay Results with Geomorphic Data submitted by James W. F. Smith in partial fulfilment of the requirements for the degree of Master of Science.

....
Supervisor

....

....

Date June 27, 1986

Aknowledgements

First, I would like to thank the members of the committee: Dr. Jean-Claude Muller, for his guidance in organizing the thesis and for discussion along the way; Ian Campbell, for his enthusiasm in the research project, and for helpful comments and suggestions; and Dr. Stan Cabay of Computing Science for reviewing the thesis.

There have been several people directly involved in different aspects of this thesis to whom I express gratitude for their help. Our system manager, Martin Scott, saw to it that things ran as smoothly as possible and taught me a lot about computers. Val Johnson wielded her editing skills on a majority of the chapters and provided much-needed feedback. Sharon Fowler gave advice on Textform and typed part of the manuscript. I would especially like to aknowledge Kimm Harty for laboriously digitizing the maps of the study area that were used for the research.

The approach taken for error measurement and analysis of the results are based on the previous research of Dr. Nicholas Chrisman, now at the University of Wisconsin. His Ph.D thesis provided a general framework for the research and valuable background information.

My friends, colleagues, and peers, inside and outside the Department of Geography, have given me moral support, comaraderie, and choice tidbits of enlightenment. I thank them for making this time in Edmonton an unforgettable experience.

Finally, I wish to express heartfelt thanks to my parents and all of my family for their continuous support, understanding, and patience.

Abstract

Overlay is considered to be one of the fundamental analytical capabilities of a spatial data handling system. With Boolean set operations such as intersection, union, and exclusion, overlay is used to determine areas that meet pre-established criteria such as coincidence of pairs of attributes between two classifications. However, the quality of the cartographic data used with this technique is frequently ignored, resulting in composite map data which are unreliable in the relationships they describe, due to accumulation of cartographic error. This thesis examines the effects of cartographic error on the accuracy of overlay results obtained with thematic maps of geomorphic data. Maps describing seven classes of surface features and four of slopes were digitized as polygons in vector format. The intersection of polygons of these two classifications were derived using a polygon overlay software system.

A model for measuring cartographic error as variability in polygon boundary location was applied to determine error in the maps before and after overlay. The error model is an extension of point error models used to calculate positional accuracy, assigning a measure of uncertainty, epsilon, to line segments of boundaries within the map. The distance epsilon defines a band of error about the line which provides upper and lower bounds on polygon areas and measures the total area within the map that is error-prone. Calculation of epsilon distance is based on known error effects introduced during the translation of areas on the ground to the map and subsequently to digital format. Three values of epsilon of 0.3, 0.5, and 0.7 meters were calculated for the two categorical maps, reflecting different assumptions about error present at distinct stages in the cartographic process.

The overlay of 369 and 476 polygons in the original maps resulted in a composite map of 2723 polygons, with high frequencies of very small polygons. Distinct trends in the cross-classification of surface features and slopes were noted. Results of the epsilon band error measurements show the relationships between polygon size, compactness, and error with increasing epsilon distance. Total map area calculated to be error-prone due to variability in boundary placement is: 13.7, 22.7, and 31.7 percent, at epsilon distances of 0.3, 0.5, and 0.7 meters, respectively.

Perimeter makes up most of the error band measured for a polygon, and thus accounts for the majority of error area within the map. The accumulation of error in the overlay is expected, as a result of combining the perimeters of the two sets of categorical polygons. Error in the composite, overlaid map has increased significantly in the combination of the two polygon coverages, but is less than the combined error of the two separate maps, primarily because of a reduction in total perimeter within the composite map. Total perimeter within the map area is influenced by the heterogeneity of the categories and by complexity and definition of the natural boundaries between them. It is felt that the epsilon value of 0.7 meters is the more realistic of the three calculated, and may well be an underestimate.

The error measurements illustrate the potential for significant error in maps produced from polygon overlay analyses, as a result of the accumulation of cartographic error in the source maps. The epsilon distance model provides a realistic means to quantify this error and produce a statement of reliability for the users of such products.

Table of Contents

Chapter Page

1. Polygon Overlay and Geographic Data Analysis 1
 1.1 Introduction 1
 1.2 Geographic Data Processing 2
 1.3 Map Overlay 3
 1.4 Automated Polygon Overlay in a GIS Environment 4
 1.5 Polygon Overlay in the Presence of Cartographic Error 6
 1.6 Introduction to the Research 6
  1.6.1 Objectives 6
  1.6.2 The Data Set and Approach to Research 7
2. Literature Review 9
 2.1 Introduction 9
 2.2 Data Structures 9
  2.2.1 Grid data structures 10
  2.2.2 Vector Data Structures 13
 2.3 Polygon Overlay with Vector Data Structures 17
  2.3.1 Pre-processing 17
  2.3.2 Intersection 18
  2.3.3 Reconstruction 19
 2.4 Problem Areas in Polygon Overlay 20
  2.4.1 Operational Problems 20
  2.4.2 The Spurious Polygon Problem 21
  2.4.3 Effects of Data Type on Spurious Polygons 23
 2.5 Cartographic Error 24
 2.6 The Epsilon Distance Model: A Method of Measuring Cartographic Error 26
  2.6.1 Background and Definition 26
  2.6.2 Construction of Epsilon Error Bands 30

2.7 Implications for the Application and Analysis 32

3. Methodology of Polygon Overlay and Epsilon Band Measurement 33

3.1 Introduction 33

3.2 The Intergraph System 33

3.3 Graphic Polygon Processing Utilities (GPPU) 34

3.4 The Database 36

3.5 Preparation of the Data 38

3.5.1 Organization of the Source Maps 38

3.5.2 Digitizing 40

3.5.3 Editing and Data Organization 41

3.5.4 Overlay Processing 45

3.6 Measurement of Epsilon Bands 47

3.7 Estimation of Epsilon Distance 48

4. Presentation and Interpretation of Overlay and Error Measurement Results 52

4.1 Introduction 52

4.2 Original Categories 52

4.3 Overlay Results 56

4.4 Upper and Lower Bounds on Polygon Area 63

4.5 Relationships Between Percent Error and Polygon Area 70

4.6 Calculation of Total Epsilon Band Area 75

4.7 Simple Perimeter Estimates of Error 78

4.8 Summary of Overlay Results and Error Measurements 81

5. Discussion 83

5.1 Introduction 83

5.2 Relationships Found through Application of the Epsilon Model 83

5.2.1 Predictability of Results 83

5.2.2 The Consequences of a Geometric Form of Error 84

5.2.3 Polygon Size and Reliability 84
5.3 Influences on Perimeter and Epsilon 85
5.3.1 Classification and Scale 85
5.3.2 Influence of Landscape on Mapping Accuracy 85
5.3.3 A Realistic Measure of Epsilon 87
5.4 Assessment of the Epsilon Model 87
5.4.1 The Epsilon Model as a Component of Digital Cartographic Data Quality ..87
5.4.2 Epsilon and Spurious Polygons 88
5.4.3 Strengths and Limitations 89
6. Conclusions 91
6.1 Summary and Conclusions 91
6.2 Future Perspectives 92
References 93
Personal Communications 98
Appendix 1: Documentation and Source Code for Program MERGER 99
Appendix 2: Documentation and Source Code for Program EBAND 109
Appendix 3: Example Output of Program EBAND 122

List of Tables

Table | Page

3.1 Mapped classes of the study area, before and after aggregation. 39

4.1 Summary of frequency and dimensions of classes of the original categories of surface feature and slope. 53

4.2 Polygon frequency, area, and perimeter of resultant classes 57

4.3 Areas of resultant polygons as a percent of each constituent class and the total map area 58

4.4 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.3 meters 64

4.5 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.3 meters 65

4.6 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.5 meters 66

4.7 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.5 meters 67

4.8 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.7 meters 68

4.9 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.7 meters 69

4.10 Summary of total epsilon band areas 76

4.11 Epsilon band areas calculated by simple perimeter estimates 80

List of Figures

Figure		Page
1.1	Examples of logical operations on polygons	5
2.1	Grid data model and data structure	11
2.2	Intersection of binary grid cell data	12
2.3	Sequential list polygon encoding	14
2.4	The POLYVRT data model and data structure	16
2.5	Spurious polygons generated by different versions of the same feature and near-coincidence of different features	22
2.6	Normal distribution of error about a point, with standard deviation σ	27
2.7	Construction of recursive bands for a line string	27
2.8	Epsilon locus for a line and a polygon	29
2.9	Effects of intersegment angles on area of epsilon bands	31
2.10	Complete overlap of inside epsilon bands	31
3.1	Database hierarchy of entity files.	37
3.2	Digitized map of surface feature polygons.	42
3.3	Digitized map of slope polygons.	43
3.4	Processing steps from digitized design file to report retrieval.	46
4.1	Distribution of polygon area (bar) and frequency (line) by area class intervals	55
4.2	Composite map of resultant polygons	59
4.3	Distribution of surface feature classes across slope classes	60
4.4	Distribution of resultant polygon area (bar) and frequency (line), by area class interval	62
4.5	Average epsilon band area for surface feature polygons, by area class interval.	71
4.6	Average epsilon band area for slope polygons, by area class interval.	71
4.7	Average epsilon band area for resultant polygons, by area class interval	72
4.8	Comparison of average band areas of surface feature, slope, and resultant polygons, at epsilon distance of 0.5 meters	74

Figure Page

4.9 Increase in band area for constant polygon area and epsilon distance 77

1. Polygon Overlay and Geographic Data Analysis

1.1 Introduction

The use of maps is essential for the analysis and description of geographic spatial relationships that exist in the environment. Very often the maps used as data sources by investigators in natural resources, management, urban and regional planning, and other disciplines are thematic in nature. Analyses performed with these maps have a common goal to reduce the mapped information to a level which can be communicated to others. Visual analysis can supply descriptive measures, often subjective in nature. Quantitative analysis involves the conversion of mapped data to numerical data, providing a more effective measure of communication (Muehrcke 1978).

Frequently, the analysis of thematic maps necessitates combining two or more themes through the process of superimposition, or overlay. Automation of the overlay process for use in computer-assisted geographic data analysis has been a subject of interest that has seen continual development in recent years. Today, a variety of computer programs, or software, capable of performing overlay manipulations with different geographic data structures, are available. The terms "polygon overlay" and "polygon processing" are associated with these software systems, since normally thematic classes are represented as closed boundaries, or polygons. These software systems have considerable potential for many applications in analyzing and manipulating spatial data useful to the geographer and cartographer.

This chapter gives an overview of computer based geographic data analysis as a prelude to the development of polygon overlay. That area will then be discussed in terms of its history, development of automated methods, and the significance of polygon overlay as a technique for geographic data analysis. In this context, the objectives of the research are outlined.

1.2 Geographic Data Processing

Peucker (1980) notes that "modern" or quantitative geography developed almost parallel to the development of the computer. The demand for mapping programs initiated the first conversion of spatial data to computer compatible form in the late 1950's. Rapid progress was made in the years following in all aspects of computer cartography, which saw the involvement of many disciplines. Geographic data processing is a systematic sequence of operations performed on data, the distinction here lying in the locational nature of geographic data. The fundamental operations of geographic data processing are compilation, storage, transformation and display of mapped data (Nagy and Wagle, 1979).

Geographic information systems (GIS) evolved as an approach to geographic data processing, incorporating data base management and analysis techniques with data processing. The simultaneous increase in the numbers of processes, sophistication, and data volume, together with growing concerns for storage and processing efficiency, resulted in the development of "integrated comprehensive software packages which perform all phases of data management" (Peuquet 1978). Peuquet's definition of a GIS takes that notion and distinguishes it from a data base management system (DBMS) by adding that "information is derived from the analysis of data." Thus, a GIS would be defined as a DBMS for spatial data with the addition of analytical capabilities (Peuquet 1978).

As the term "system" denotes, a GIS is made up of integrated components, each of which may be viewed as a subsystem. The various approaches to GIS components each deal differently with the functional as well as spatial, non-spatial, and temporal aspects of data (for example, see Dangermond 1983; Nagy and Wagle 1979; Mitchell 1978). Tomlinson et al (1976) identifies the functional components as 1) management, 2) data aquisition, 3) data input and storage 4) data retrieval and analysis, 5) information output, and 6) information use. Within the data analysis component, several major categories are identified. These are measurement of distance and area, reclassification, statistical analysis, characterization of of cartographic neighborhoods, and overlay (comparison of multiple data sets) (Tomlin and Berry 1979; Tomlinson 1977). Of these four, overlay analysis techniques will be examined.

1.3 Map Overlay

In simple terms, map overlay is the placement of two maps, one over the other such that one may view the contents of both simultaneously. The maps must be of the same scale and registered such that any given point location is identical on both maps. Overlay can be defined as "the spatial synthetic process of grouping back together information which has been previously analytically isolated and classified from experience" (White 1978). This is often the case when using a computer, where cartographic data are frequently separated by thematic class.

The purpose of this procedure, whether the medium involved is paper, transparencies, or an electronic display, is to compare the mapped phenomena to each other, to determine their spatial association. Thus, the person performing the analysis can determine correspondence or non-correspondence between any number of phenomena, such as the relationship between soil types and vegetation types in a particular area.

Manual overlay techniques have been used for many years, but specific references are few. In 1950 Jaqueline Tyrwhitt provided what is considered the first explicit description of "the overlay technique" (Steinitz et al 1976). Other cases where the use of overlays is implied but not specifically stated date as early as 1912. From that time through the 1930's only a few scattered examples are found, by planners and landscape architects in New York, London, Dusseldorf, and other areas. (Steinitz et al 1976).

Better documentation exists for studies since 1950, especially in the work of Ian McHarg. His book, Design With Nature, illustrates the use of overlaying mapped ecological, physiographic, land use, and other "factors" or classes in the planning of medium scale residential developments and other projects. Transparencies shaded with grey tones or combinatorial color schemes, both representing attribute values such as water table level, soil drainage characteristics, and forest cover, were overlaid to provide indicators of, for example, suitability for residential and industrial development. McHarg (1969) states that the methodology was familiar but the "evidence" was not, indicating that he was initially unsure that the technique would produce the desired results, which in fact it did. McHarg's work was instrumental in demonstrating that the overlay methodology provided a means for "organizing, displaying, and synthesizing multi-dimensional spatial data" (Schlesinger et al 1979).

1.4 Automated Polygon Overlay in a GIS Environment

Automated polygon overlay performs two major functions in a GIS: extraction of data from a user-defined query window within a larger data set (Dangermond 1983), and production of a composite data set from multiple sets of thematic data (MacDonald 1982). This composite data set or map defines areas, or polygons, whose attributes are based on the attributes of the polygons contained in each separate map prior to composition (Dangermond 1983).

The importance of this capability is that it enables the user to determine areas that meet certain pre-established criteria. These criteria are categorical and commonly binary in nature. When using binary criteria, the sets of polygons may be compared using the operations of Boolean set theory of intersection, union, and exclusion. Figure 1.1 illustrates these logical functions through the use of modified Venn diagrams. The two top rows contain two sets of categorical maps, binary in nature. The left pair of maps contains two classes each, "A" and "not A", "B" and "not B", respectively. The polygons in the right pair of maps correspond to these classes. Below are shown the resultant polygons for the basic Boolean set operations of intersection, union, and exclusion between classes "A" and "B".

The representation of the mapped data and their combinations by polygons gives the user a "real world" situation for visual analysis, which is more effective for making decisions (Allen 1976). Perhaps the greatest advantage of automated overlay is its greater speed compared to its manual counterpart.

These capabilities and characteristics indicate that polygon overlay is important as a spatial analysis technique in a GIS environment. Miller (1980) emphasizes this by stating:

> "The function of map overlay or compositing is essential to spatial analysis. The ability to determine the union or intersection of various mapped data sets is probably the single most important manipulative function of a spatial data handling system."

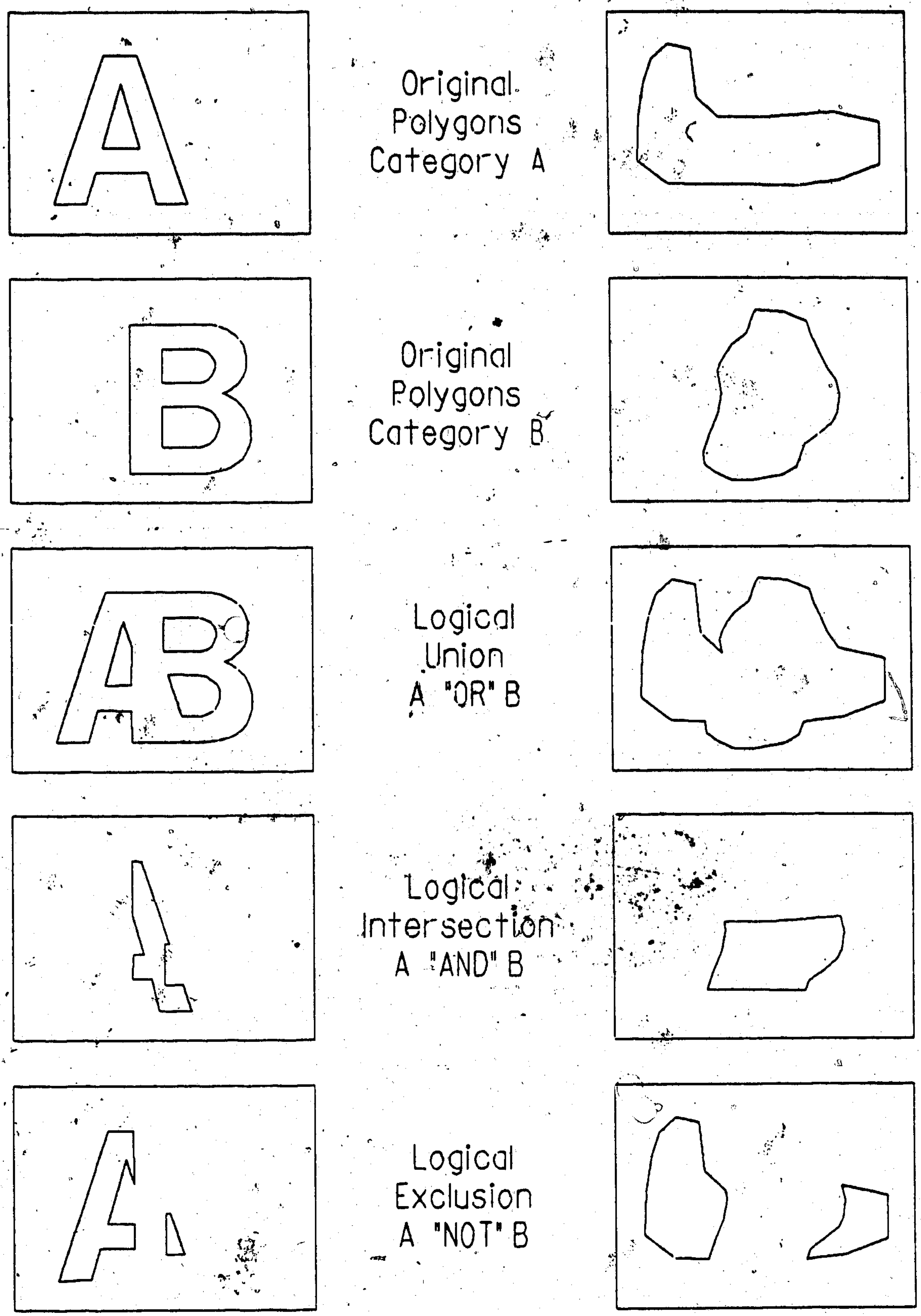


Figure 1.1 Examples of logical operations on polygons

1.5 Polygon Overlay in the Presence of Cartographic Error

Some degree of imprecision is present in all phases of the cartographic process. The mechanical methods of documenting spatial features, whether controlled by hand or by machine, introduce positional error in defining points or lines. Similarly, error in interpretation of features can lead to positional or classification inaccuracies. Another source of error comes from the indeterminacy of features or "fuzzy boundaries" (Dutton 1984) between two attributes which can only be approximated. The accumulation of these types of errors is not readily reflected in the high precision of cartographic data in digital form. A program designed to calculate the intersection of polygons can return an "exact" solution according to the data it is given, but the error which is attached to the data cannot be accounted for.

Recognition of cartographic error is essential to the assessment of data acquired through spatial analyses such as polygon overlay. In order for such products to be used effectively, the limits of their accuracy and/or reliability should be determined.

1.6 Introduction to the Research

1.6.1 Objectives

It is the objective of this thesis to examine the effects of cartographic error on the accuracy of polygon overlay results obtained with categorical maps of geomorphic features. Drawing on other research, a model of cartographic error suitable to the data will be used to calculate the amount of error present in the original maps and in their overlaid form. The error model is based on the variability in categorical boundaries, and expresses this variability as measurable area. The model can be used to determine error area for individual polygons, classes, and the whole map. The error measurements can provide information regarding usefulness and limitations of cartographic data obtained through overlay techniques.

1.6.2 The Data Set and Approach to Research

A small drainage basin in the Steveville badlands in southern Alberta has been the subject of long-term studies of erosion rates and processes. From these studies, erosion rates have been determined for lithological classes, and the effects of differing surface slopes on erosion have been calculated. As an integral part of this previous research, it is desirable to determine the proportion that each combination of slope and lithology represents within the study area. These data can then be used with meteorological data to estimate the amount of material eroded from the surface of the area over time.

Detailed categorical maps of surface features (combined vegetation cover and surficial lithology) and slopes have been compiled for the drainage basin. With the maps for slope and lithology entered into a computer file as polygons, polygon overlay can be used to determine the intersections between classes of each category. The areas of intersection can then be calculated and retrieved.

The mapped data used for this research are perhaps unique in their content and coverage. However, the nature of the complex natural boundaries they describe is not unlike the kind of map data which will be required for overlay analysis in many situations. Any spatial analysis dealing with natural features such as forest cover, hydrography, or soils will involve the use of boundaries of questionable accuracy, due to error accumulated in the cartographic process. The data chosen for this research, then, is suitably representative of what one would expect to encounter elsewhere. At the same time, the use of interactive computer graphics and polygon overlay in this application represents an innovative approach to the synthesis and analysis of geomorphological data.

The research is carried out in the following phases:

1. Application of overlay:

   The categorical maps are overlaid to determine relationships between classes of two main categories, surface slope and surface features, within the study area. The purpose of this application, from the point of view of the geomorphologist, is to produce quantitative data about these relationships that can be used in the study of geomorphic processes.

2. Error measurement:

   The epsilon distance model of cartographic error, described in chapter 2, is used to determine amounts of error associated with variability in polygon boundaries. Determination of epsilon distance is based on known error effects present in distinct stages of the cartographic process, from the field survey to the maps in digital form.

The results of the application of overlay are presented and the relationships thus determined are examined. Results of the error measurements are then given for the original and the overlaid, composite map. Error estimates are examined for each class and summed to a single measure of error for the whole map. An attempt will be made to assess the effects of the characteristics of the mapped data on the amounts of error found. These characteristics are the complexity of the natural boundaries found in the maps and the transitional nature of the attributes present. These findings will be discussed in light of arguments offered by other research.

2. Literature Review

2.1 Introduction

Of primary consideration in using spatial data handling systems is the manner in which the mapped data are organized and encoded in the machine environment. This organization of data structure has a direct bearing on the algorithm which must be developed to access the data and compute the intersection union, or other logical functions. Map overlay can be divided into two basic approaches according to data structure, either grid or vector. In this chapter, the grid and vector data structures and their corresponding overlay methods are reviewed, with an emphasis on vector-based approaches. Problem areas related to implementation of algorithms are described.

Problem areas are not confined to hardware and software capability. If the program used to perform polygon overlay is returning correct results, then the main influences on the quality of the overlay results are the maps themselves. Two aspects considered are map content and quality, in the form of data type and cartographic error, respectively. Recognizing these, an approach to measuring error introduced in the process of transferring ground truth to a digital file is described.

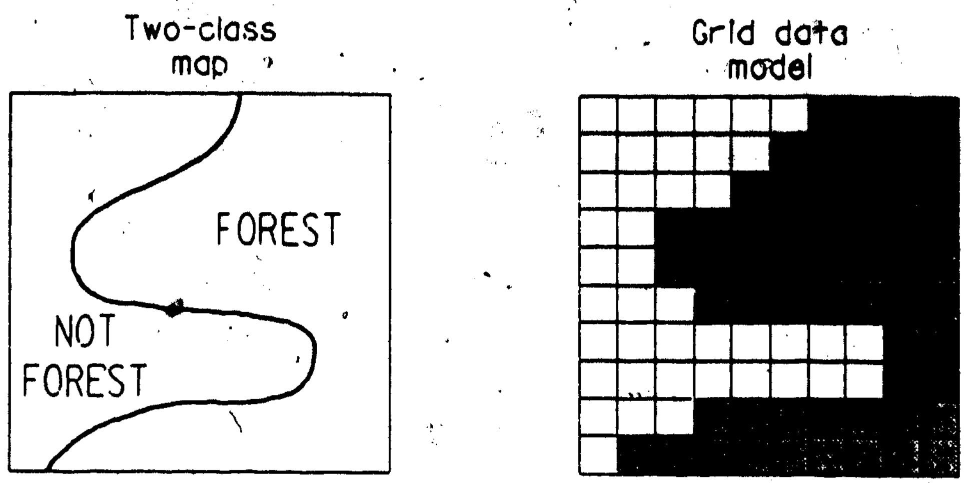
2.2 Data Structures

A spatial data structure can be defined as a data organization whose elements are items of spatial data that possess a relative location within a two-dimensional system, and an associated set of attributes (Guevara 1983). The manner in which these data are stored preserves the relationships and logical associations between them, and should provide access from one data element to another (Williams 1971). A representational model, usually in the form of diagrams, lists, and arrays, exists between the data model (map) and the computer file to reflect the recording of data in computer code (Peuquet 1984). Spatial data structures are divided largely into two categories, vector and grid, by their basic logical elements.

2.2.1 Grid data structures

A grid data model of a map can be perceived as a systematic sampling (Morehouse and Dutton 1979) in x and y at regular intervals. The inherent form of the mapped thematic data is thus inverted. In an analog map, thematic content is controlled and location of polygon boundaries is varied according to the description of the attribute. With grid data structures, location is controlled through the imposition of a rectilinear grid, and the thematic content is varied for each location or grid cell (Sinton 1978). This basic logical unit may be based on other types of division of the area (i.e., triangular or hexagonal). The square grid is the most widely used since it is compatible with array data structuring in FORTRAN programming, as well as with hardware devices used for spatial data capture and output (Peuquet 1984). In its most direct form, one storage location is required for each grid cell in an m by n matrix, and the locations are structured sequentially as a list or array (see Figure 2.1). Neighborhood relations in a grid data structure are implicitly defined by the position of the grid cell in the array. This characteristic and the high compatibility to the computer system make sorting and searching procedures easy to implement. Alternative methods to direct grid cell encoding are employed to reduce both the number of "empty" grid cells and the amount of storage required to represent "occupied" cells, such as run length encoding. In a single direction (usually along the row), sequences of grid cells of the same value are represented by a single value according to various notation schemes (see Semwal, 1984).

Because of the nature of gridded data, overlay operations are inherently simple. The corresponding areas of grid cells are treated as two-dimensional arrays where the grid cell P(i,j) of map A is compared to the same grid cell in map B. Logical operations are used for comparison of grid cell values to produce areas of intersection, union, exclusion, etc. (see Figure 2.2). Basic arithmetic operations may be used to compute new attribute values based on values of corresponding grid cells of the input maps (Tomlin and Berry 1979). For an m by n matrix, this requires (m x n) iterations of the logical or arithmetic operation. A faster algorithm has been developed by Miller (1980), utilizing a run length encoding scheme.



Row ID	Cell value									
1	0	0	0	0	0	0	1	1	1	1
2	0	0	0	0	0	1	1	1	1	1
3	0	0	0	0	1	1	1	1	1	1
4	0	0	1	1	1	1	1	1	1	1
5	0	0	1	1	1	1	1	1	1	1
6	0	0	0	1	1	1	1	1	1	1
7	0	0	0	0	0	0	0	0	1	1
8	0	0	0	0	0	0	0	0	1	1
9	0	0	0	1	1	1	1	1	1	1
10	0	1	1	1	1	1	1	1	1	1

Direct encoding of binary grid cell values

Figure 2.1 Grid data model and data structure

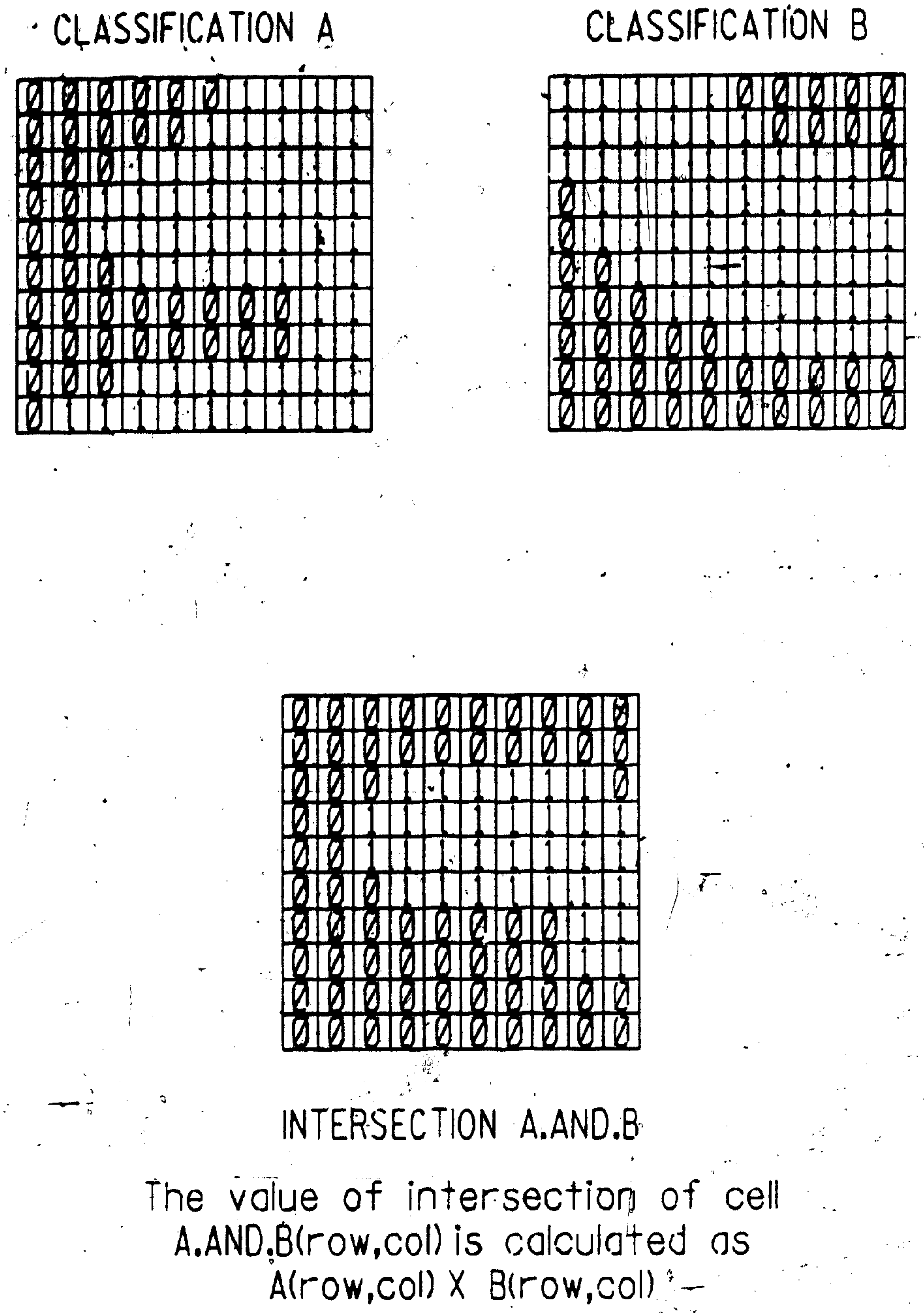


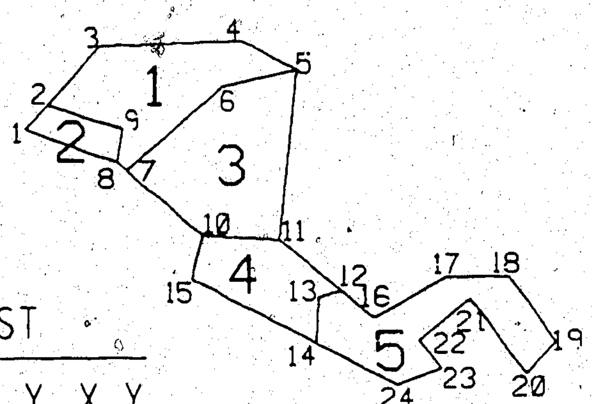
Figure 2.2 Intersection of grid cell data

The use of grid data structures involves consideration of two opposing constraints: storage requirements and grid resolution. Representation of attribute values for every grid cell results in wasted space where there are null attributes. Run length encoding alleviates this problem somewhat, depending on the homogeneity of the attributes of the map area. Directly related to the problem of storage requirement is the size of the grid cell used. When source data of vector or analog type are converted to grid format, error is introduced where original boundaries do not match grid boundaries, but instead pass through the cells. A decrease in cell resolution results in a linearly corresponding increase in error (Muller 1977). It is desirable to use a finer resolution to capture detail in the original maps, resulting in increased storage requirements. For a given map area, a reduction of grid cell size (increase in resolution) by half quadruples the number of cells. Efforts to minimize storage often result in a failure to replicate the source data adequately due to a less than optimum grid cell size (Miller 1980). These generalization effects present in vector-to-raster conversions are felt to lead to serious inaccuracies in identification and measurement of areas.

2.2.2 Vector Data Structures

Vector data structures commonly define points, lines, and polygons by using (x,y) Cartesian coordinates based on principles of Euclidean geometry (Dangermond 1983). Thus a point or sequence of points located by (x,y) coordinates may describe a geographic entity in terms of a single, dimensionless point location, and single or multiple straight-line segments of one dimension. A sequence of points which closes upon itself, beginning and ending at the same (x,y) location, defines a two-dimensional area or polygon. This fundamental map model can be represented by several types of data structure. This discussion will be limited to those structures more widely implemented and used with polygon overlay algorithms.

The most elementary type of vector data structure for encoding polygons is a sequential list file. A polygon identifier marks the beginning of a new polygon and is followed by an alternating sequence of x and y values. (see Figure 2.3) Thus each entity (polygon) on



POLYGON ID	COORDINATE LIST
1	$X_2 Y_2, X_3 Y_3, \ldots X_9 Y_9, X_2 Y_2$
2	$X_1 Y_1, X_2 Y_2, X_9 Y_9, X_8 Y_8, X_1 Y_1$
.	
.	
.	

Figure 2.3 Sequential list polygon encoding (after Peuquet, 1984)

the map equals one logical record in the digital file, as defined by the string of (x,y) coordinates, and, since polygons are closed, coordinates must be recorded twice for shared polygon boundaries (Peuquet 1984). Failure to match coincident points on boundaries results in overlaps and gaps between neighboring polygons.

The indexed list or point dictionary (Peucker and Chrisman, 1975) file structure eliminates the redundancy of twice-recorded polygon boundaries, reducing storage, as well as solving the gap and overlap problem. A dictionary contains the points needed to describe every boundary on the map, while a separate polygon list contains the labels associated with the points required to reconstruct that polygon.

These simple data structures are suitable for storing and retrieving cartographic lines for display and reproduction purposes, but the polygon records do not include any information about its position relative to other polygons. By adding the topological neighborhood function of each element to a data structure, large improvements in flexibility and applications can be realized (Peucker and Chrisman, 1975).

Topological data structures commonly use a hierarchy of *nodes*, significant points where two or more line segments meet, *chains*, the line segments between nodes, and *polygons*, formed from sets of chains. Each logical record is based on a chain, consisting of, at least, a chain identifiers for the left and right polygons, and the beginning and ending nodes. Data structures such as those presented by Lam (1977) and Baxter (1980) include the coordinates for the nodes and intermediate points in each chain record. In the POLYVRT structure (Peucker and Chrisman, 1975), separate tables are created for both the node coordinates and intermediate (detail) coordinates, indexed by labels in the chain record. The chain record is then indexed by a polygon list containing the name of the polygon and chain identifiers (see Figure 2.4). Thus the node, chain, and polygon hierarchy is maintained in the file structure, and the data can be searched at any of these levels, eliminating unnecessary processing overhead.

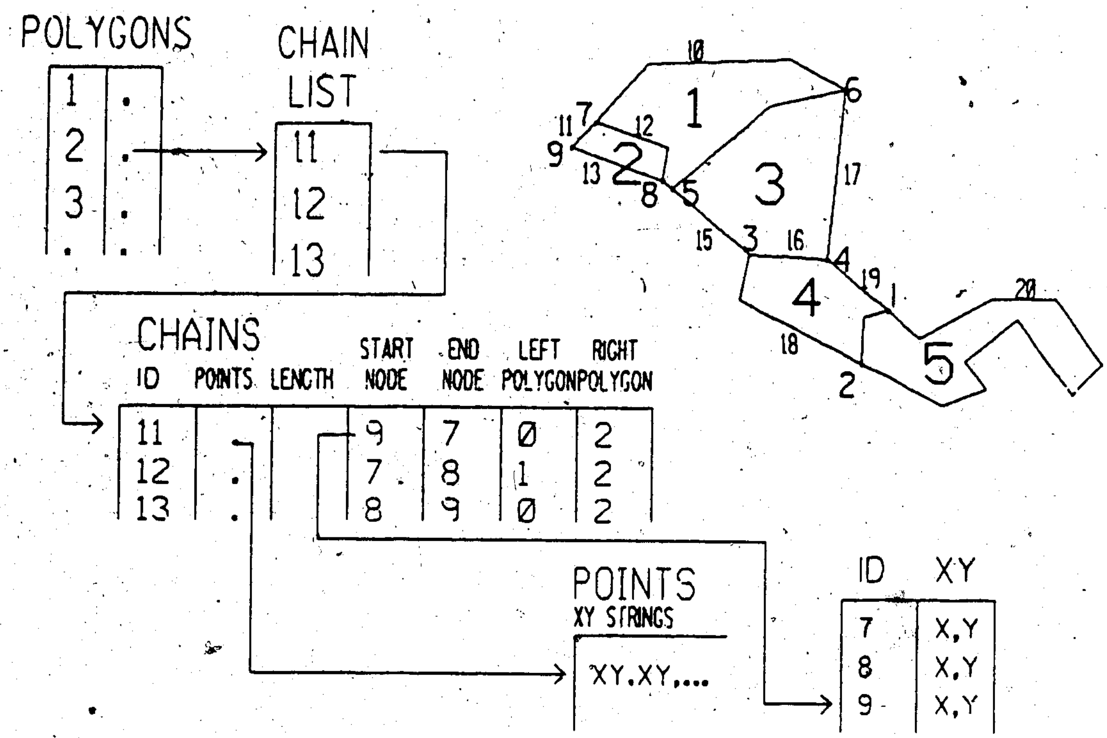


Figure 2.4 The POLYVRT data model and data structure (after Peuquet, 1984)

In the application of polygon overlay performed for this thesis, overlay operations with categorical maps use a vector data structure. This type of data structure is used because the original map data existed in analog form and was captured on a digital file in vector format. Conversion of the maps from vector to grid cell format would lend an additional error component to the definition of the polygons and would affect the area measurements of the original polygons as well as the overlay results. This is undesirable in the context of measuring and assessing cartographic error, which is fundamental to the analysis.

2.3 Polygon Overlay with Vector Data Structures

Polygon overlay can be divided into three sub-tasks: pre-processing, intersection, and reconstruction. Detailed descriptions of working programs are found in White (1978), Lam (1977), Guevara (1983) and Weiler (1980). For the purposes of this thesis, these sub-tasks are generalized, and specific programs are cited where necessary. The overlay program used for the application study is described in Chapter 3.

2.3.1 Pre-processing

Pre-processing is here considered to be any manipulation of the polygon data before line segment intersection is performed. The purpose of most pre-processing manipulations is to increase program efficiency by eliminating non-intersecting polygons, or by restructuring the polygon data for more efficient use. Pre-processing includes sorting polygon sets on the X or Y axis and creating new, ordered files to be overlaid (Tang 1985, personal communication). This reduces the amount of searching required to retrieve and compare polygons that are near each other.

A widely used pre-processing routine for vector polygon overlay is the bounding rectangle check. The minimum and maximum x and y coordinates, representing a rectangle containing the whole polygon, are compared for two polygons to be overlaid. If the rectangles do not overlap, the polygons cannot intersect. Bounding rectangle checks can also be performed on individual line segments (Burton 1977), so that intersection computation need only be done for those segments whose bounding rectangles overlap.

2.3.2 Intersection

Intersection is the fundamental task of overlay. There are two commonly used methods for calculating the intersection point of two straight lines in a plane. The slope-determinant form is used in the polygon overlay program OVER. Given the points A, B, C, and D which define line segments AB and CD, the respective slopes of the line segments are:

$S(AB) = (YA - YB)/(XA - XB)$ and $S(CD) = (YC - YD)/(XC - XD)$. The intersection of the two lines (XI, YI) is given by

$XI = (YC - SAB * XC - YA + SAB * XA) / (SAB - SCD)$ and

$YI = SAB * (XI - XA) + YA$ (from Lam 1977).

Vertical lines (infinite slope) and horizontal lines (zero slope) create difficulties in calculation and must be treated as special cases. The above form defines the line segments as lines of infinite length, extending beyond the four endpoints. Using this form, an intersection point may be calculated which does not lie within the segments, so this condition must be tested for. Another method of intersection applies the general linear equation $AX + BY + C = 0$:

> "By use of the relationship $u = ay - bx$, every point (x,y) on a single line (a, b, c) can be assigned a unique value u. ... the point of intersection of line (a, b, c) with line (p, q, r) is given by $x = [(br - cq)/(aq - bp)]$ and $y = [(cp - ar)/(aq - bp)]$. ... this line representation has the advantage that it leads to the elmination of the special case of lines with an infinite slope. The immediate disadvantage is that it is computationally more expensive since it involves the solution of a system of three simultaneous linear equations"(Guevara 1983, p. 58).

Regardless of the method used to compute the intersection point, a basic consideration of the algorithm is the number of times this operation must be performed. If no type of pre-processing is done before the intersection step, then each line segment in one set of polygons must be compared to every other line segment in the other set. For two sets of polygons, containing a total of n segments, the comparison must be performed n(n - 1), or, $(n^2 - n)$ times. In the notation of space/time complexity analysis of algorithms, the linear term of this function (n) is dropped, and the quadratic term (n^2) is used to express the order of magnitude of time required to perform the task. Pre-processing such as the bounding rectangle check eliminates line segments to be compared, but for those n segments that must subsequently be compared this step will be performed on the order of n^2 times. Line intersection is often referred to as an "n^2 problem" (Shamos 1975; Tilove 1981; Teng, personal communications, 1985) for this reason, and consequently can be very "expensive" or time consuming for large data sets involving thousands of segments. Hence the need for pre-processing manipulations such as bounding rectangle checks and line passing.

2.3.3 Reconstruction

Once intersection points are determined, the program must break the line segments at that point and determine which of these new line segments to use to generate the resultant polygon. This phase of polygon overlay has been observed to be the most difficult (Guevara 1983, p. 113). Unfortunately it is also the least discussed. The following descriptions are based on specific examples found in the literature.

Methods of reconstruction are highly dependent on data structure. In topological data structures, the induced direction of a chain given by its starting and ending node play an essential part in reconstructing polygons from broken line segments. For example, the program WHIRLPOOL (White 1978) "cycles" around broken chains and determines, based on parentage, to which resultant polygon the portion of the chain belongs. A similar operation is performed by OVER (Lam 1977). These two programs create a composite network of polygons with attribute values assigned according to whether the polygon is composed of either one of the original categories, or both.

An example of reconstruction in a non-topological data structure is in the overlay module A.NOT.B (Guevara 1983). It examines a decision table to track the segments of the resultant polygon. To construct the intersection between polygons A and B, for example, it starts at any known intersection point of the two polygons. The decision table determines which of the four line segments (originally two, now broken at the intersection point) radiating from that point define part of the resultant polygon. From there, the resultant polygon boundary (which might be part of A, for example) is followed until another intersection point is reached. It must then switch polygons to B, and the decision table determines the correct direction in which to continue following along B to define the rest of the resultant polygon. A similar approach is described in the overlay processor for PIOS (Tomlinson *et al* 1976), but the construction of the resultant polygons is performed simultaneously with intersection, and the use of decision tables is not mentioned.

2.4 Problem Areas in Polygon Overlay

A group of problems associated with polygon overlay has been identified by Guevara (1983, p. 11). Most closely related to the aims of this research are problems originating in 1) operational problems in the algorithm; and 2) accuracy and reliability problems of polygon overlay caused by data type and error in the mapping process.

2.4.1 Operational Problems

1. Digital roundoff. In early attempts to write routines for calculating the intersection point of two lines, Douglas (1972) identified cases where the intersection of two line segments lying very close to each other were incorrectly determined, due to rounding of the significant digit values for the vertices of the line segments as well as the calculated intersection point. His solution was to define a tolerance distance, equal to the amount of roundoff, such that a point is defined as being on the line if it is within the tolerance, thus making the point a valid intersection.
2. Subjoint or nested polygons. This condition occurs when one polygon is contained totally within another. Nested polygons may take on multiple levels of complexity. First, "holes"

can exist in classifications where a continuous polygon of one class completely surrounds another (e.g., forest surrounding a lake). Second, there can be "islands" within the holes (e.g., an island in a lake of the same forest class as that surrounding the lake). Identification of these conditions is necessary to correctly solve the logical intersection between two categories. Otherwise, significant errors can result. Commonly a method of identifying nested polygons is included at the data description level, such as a special identifier bit for nested/unnested condition in the polygon description (Tomlimson *et al* 1976). Another method used, for cases of islands within holes, is to connect the island to the exterior boundary with a line segment (Sinton *et al* 1972b).

3. Spurious polygons. Both Guevara (1983) and Lam (1977) classify spurious polygons as an operational problem, in that it results in the generation of many small, presumably erroneous polygons, requiring additional processing to perform the intersection, tracking, and generation of resultant polygons. This "unnecessary" processing is considered a hindrance to operational efficiency. However, it will be seen below that spurious polygons originate in map error.

2.4.2 The Spurious Polygon Problem

False or spurious polygons are formed when, as a result of overlay, two versions of the same geographic line fail to coincide, but instead cross over each other more than once over portions of their length. The net effect is the generation of many small, sliver-like polygons (see Figure 2.5). These are especially evident when two versions of the same, geographic line, digitized independently, are compared (Goodchild 1980a). For such cases it must be considered that [illegible] polygons from overlaying identical geographic lines is strictly the result of digitizing error, since ideally the lines should coincide. When dealing with independently derived coverages, which is more commonly the case in polygon overlay applications, the generation of spurious polygons is a result of the proximity of feature boundaries between the coverages, as well as an error component. It is possible that

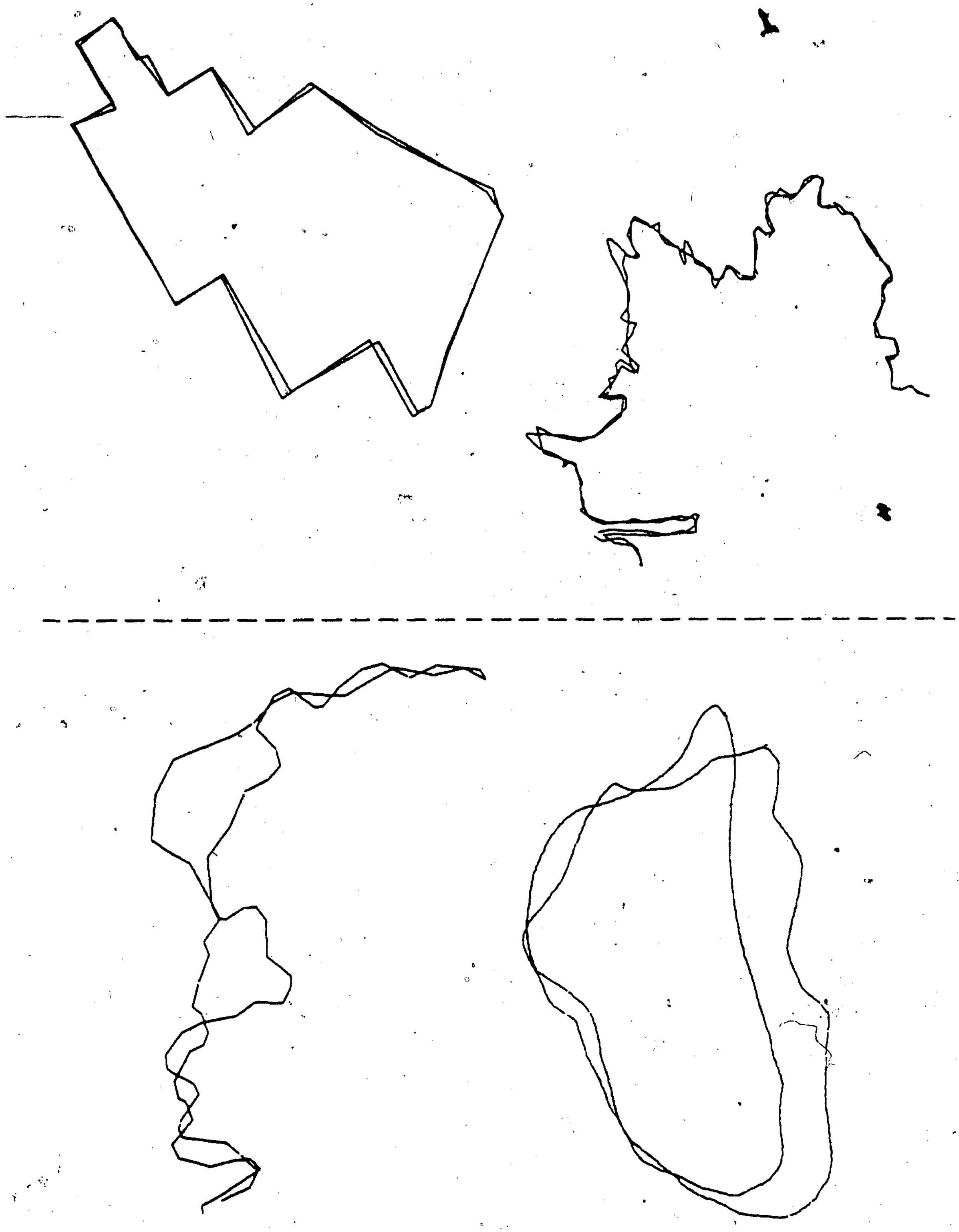


Figure 2.5 Spurious polygons generated by different versions of the same feature(top) and near-coincidence of different features (bottom)

sliver polygons generated from similar geographic lines are real, because the features do not coincide, but instead are just similar. Conversely, similar lines may coincide where they should not, due to error. Such conditions are difficult to identify with independent coverages. In the case of different coverages where the same geographic line is incorporated as part of a boundary on both coverages, spurious polygons can occur where that feature is delineated differently on each coverage. Difference in delineation of the same feature might also be considered to be mapping error, but it is difficult to avoid, especially when using coverages where boundaries incorporate man-made features such as census tracts and administrative boundaries that include roadways and survey grid lines in their definition.

2.4.3 Effects of Data Type on Spurious Polygons

The number of polygons generated from an intersection of two polygons of n_1 and n_2 vertices, respectively, can range from three to $(n_1 n_2 + 2)$ (Goodchild 1978). This exponential growth function means that, as the polygons become more complex, the potential number of spurious polygons increases tremendously. Goodchild illustrates that the number of spurious polygons created is greatest when the digitized lines have high density of points (greater level of detail) and have high statistical similarity, or tendency to follow each other. Conversely, if the lines overlaid are less detailed and are derived from independently mapped distributions, spurious polygons are reduced.

The number of polygons generated during overlay thus depends on the complexity of the boundaries that are compared. A line of homogeneous characteristics along its length is statistically stationary; a change in the threshold level of the digitizing criterion measure will produce a proportional change in point density. It will also have low serial correlation of direction, which is the degree to which the line's direction can be predicted given the direction at a previous point on the line separated bv a constant lag distance (Goodchild 1978). An example of a line with low serial correlation would be a highly sinuous, meandering river. Non-stationary lines have high serial correlation, for example, part of a provincial boundary composed of long, straight stretches interrupted by sharp corners. Given two versions of the same line, those with the lower serial correlation will intersect each other more often, resulting in greater numbers of spurious polygons, than will two lines of high serial correlation.

The effects of varying stationarity and serial correlation indicate a significant difference in overlay error using natural versus anthropogenic polygon boundaries. Digitized lines representing legal and administrative boundaries will incorporate survey grid lines, street centerlines, and cadastral boundaries. These types of lines should have relatively high accuracy and low point density compared to natural boundaries that are irregular, subjectively defined, and influenced by generalization processes. The lack of reliability in natural boundaries is reflected in cautions advanced specifically regarding the use of such boundaries for overlay analysis (Bie 1980; Goodchild 1981), but no similar warnings are found for data types such as administrative or legal boundaries. A very important point made regarding reliability is that the accuracy of source data cannot usually be improved, and must be accepted (Goodchild 1981). Considering that the majority of applications of polygon overlay will include natural boundaries, the unavoidable error associated with them should be taken into account.

2.5 Cartographic Error

> "The great difficulty that is encountered when attempting to solve the polygon overlay problem is not at all obvious. This is because it is really the separate problem of data error. The polygon boundary locations are not exact. Errors in surveys, the source maps, the digitizing process all combine to make the boundaries only approximate locations." (Dougenik 1979)

Error can be defined as any form of divergence between the map and the earth's surface (Chrisman 1982b). This includes generalization, where simplification, aggregation, or elimination of features (Rhind 1973) reduces the information content of the map. Generalization is a form of divergence, introducing both intentional and unintentional error through the failure to capture all the details on the map. Different generalization processes are associated with distinct sources of error introduced in the mapping process. The most recurrent sources of error, as defined by Chrisman (1982b), are:

1. Locating ground position. This is initial surveying error, and applies to points measured

and all information interpolated between the known points. This type of error is easier to work with and account for than other types of error, due to refinement of point error models used in surveying and geodesy (Chrisman 1982b).

2. Interpretation. Human error is present in both direct field observations and examination of other source data such as aerial photographs or maps. Error in discriminating between surface attributes results in inaccurate placement of boundaries. Interpretation error is related to descriptive generalization (Cook 1983), as human judgement affects the definition of homogeneous areas. Gross misclassification of surface attributes is included in this type of error.
3. Scale. The level of spatial generalization is determined by scale, again influencing the interpretation of boundaries and aggregation of inhomogeneous areas..
4. Conversion to map space. Manual drafting errors stem from the three components involved: the pen, the draftsman, and the paper or other medium used. Redrafting of boundaries from one map to another can involve errors between the two versions of the line which are not detected because they are within the combined thickness of the two lines (Chrisman 1982a). Generalization in the delineation of regions in an original or a redrafted version is the result of decision-making processes on the part of the draftsman. Other effects stem from misregistration of maps and the stability of the medium used.
5. Digital handling. One source of error in transferring mapped data to a digital file is numerical roundoff during digitizing. The amount of roundoff is dependent on the resolution of the digitizing surface, although this is usually greater than the precision of the person handling the cursor (0.025 mm is a common resolution).

The actual process of digitizing introduces the largest potential error in digital handling (Chrisman 1982b). Line-following errors, similar to manual drafting, of psychological and physiological origin result in errors that far exceed the resolution of digitizing equipment. Studies have shown that, despite efforts to maintain a standard of minimal error of about 0.1 mm, operator-related error can exceed that standard by a factor of five and sometimes by a factor of ten (Jenks 1981). Realistic estimates of manual digitizing error cited by Chrisman (1982a, pp. 62-64) range from 0.145 mm to 0.5 mm. Because most digitizing methods create a string of straight-line segments to represent curved lines, generalization is inherent. Generalization from simplification of lines results from point elimination and smoothing, especially on lines defining natural features (Jenks 1981).

The combined effects of errors in (1) through (5) above result in line segments in the digital file which do not agree with reality. Two versions of the same geographic line, digitized independently during separate mapping processes, will often disagree in location. This is where map error is reflected in polygon overlay, in the form of spurious polygons.

2.6 The Epsilon Distance Model: A Method of Measuring Cartographic Error

2.6.1 Background and Definition

Methods for measuring variability in the location of cartographic features can be applied to point, line, and areal data. Point error models, as applied to surveying and geodesy, usually take (x,y) coordinates as unbiased estimates of the position of a point. Random error in location forms a probability distribution, identical in all directions about the point, and converting the point error into areal error, as shown in Figure 2.6 (Chrisman 1982a). Under a normal distribution, the circle of radius σ represents one standard deviation from the point in all directions, and the dashed circle of diameter 4σ describes the area containing 95 percent of the distribution, at plus or minus two standard deviations.

Peucker's (1976) theory of the cartographic line represented by recursive bands lends itself to error measurement for linear features. A line string between any two points defines a rectangle or band of variability by construction of a single line segment between two endpoints, projected parallel on either side to the same distance as the maximum deviations from the central trend line (see Figure 2.7).

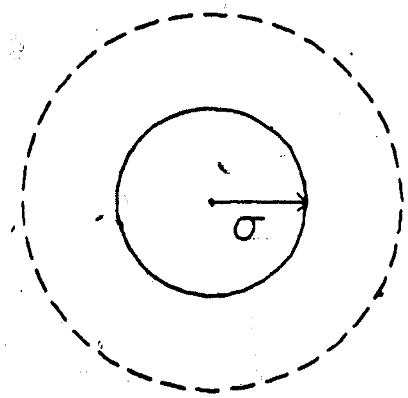


Figure 2.6 Normal distribution of error about a point, with standard deviation σ

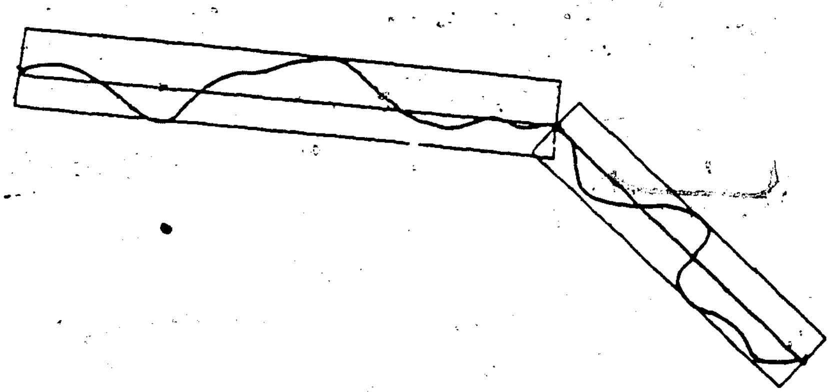


Figure 2.7 Construction of recursive bands for a line string

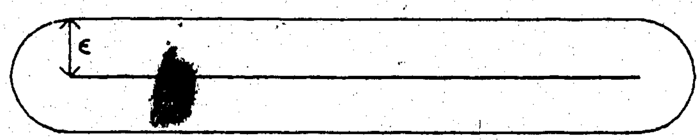
These notions of bands of deviation and probability of location are taken one step further in the epsilon distance model. This model is based on concepts put forth by H. Steinhaus and J. Perkal in the 1950's, and is defined by Chrisman (1982a) as follows:

> "Given a cartographic line as a straight line approximation, it might be assumed that the true line lies within a constant tolerance - epsilon - of the measured line. The parameter epsilon might be a technical feature of the measurement method or applied as an index of confidence in the map. In either case, the distance epsilon defines a locus of points around the line which should contain the true line. For a straight line segment this locus is simple, consisting of the union of a rectangle parallel to the segment, twice epsilon wide, and circles of radius epsilon centered at each end point." (Chrisman 1982a)

Determination of epsilon distance is based on the amounts of error introduced during the mapping process. A basic assumption of the epsilon distance model is that these errors are independent, unbiased, and, when combined, they are normally distributed with the distribution curve centered on the line. For each point, a probability distribution function for its location is assumed, where the point location represents the mean value and the distance epsilon is equal to the variance. When this distribution function is extended between points defining a polygon, it describes a band of total variability with a measurable area, on either side of the polygon (see Figure 2.8). The epsilon distance model for lines can then be viewed as a limit of the point model with all points along the line given equal weight (Chrisman 1982a).

The areas within the inside and outside bands around the polygon represent the upper and lower limits of the polygon area, and can be expressed in absolute terms or as a percentage of the polygon itself. The combined area of the inner and outer bands, expressed as a percentage of the measured polygon area, can be interpreted as a measure of "reliability" for the polygon. If the total area contained in the epsilon bands is 20 percent of the measured area, then one would say that the potential error associated with the polygon is 20 percent, or that the polygon area measured is 80 percent reliable.

The conditions of contiguity of polygons within the coverage affect the reliability for description of a single polygon. Every part of an epsilon band, inside a given polygon, is part



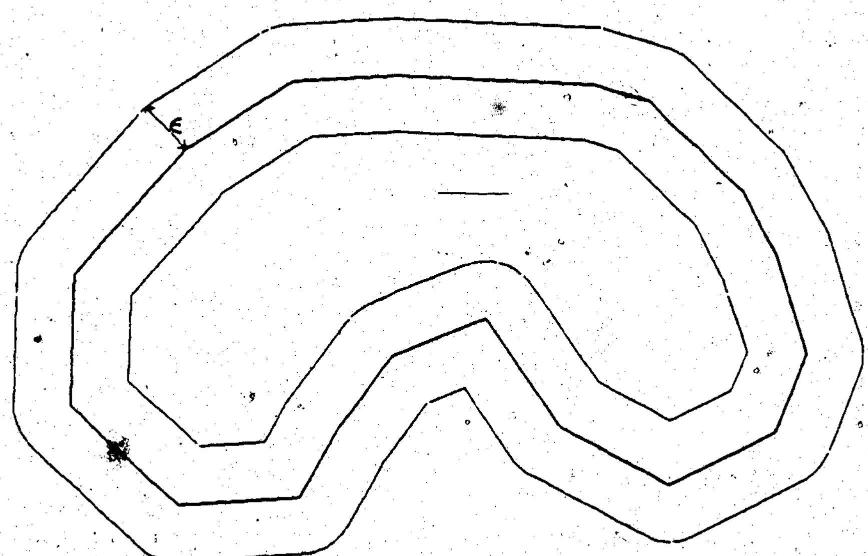


Figure 2.8 Epsilon locus for a line (top) and a polygon (bottom)

of the band for the outside of a neighboring polygon, and vice versa. Thus the area that is inside the polygon boundary and also within the inner epsilon-band is potentially part of any of the adjacent polygons. The fact that the bands of error are shared by contiguous polygons within the coverage will have an effect on calculating the total area of the bands.

2.6.2 Construction of Epsilon Error Bands

To construct the error band which defines all points within distance epsilon of the polygon boundary, two aspects of the boundary come into play: the boundary itself, which is the perimeter of the polygon, and the angles formed where two segments of the perimeter join. The band for a single segment, as described before, is composed of the union of a rectangle twice epsilon wide, and circles of radius epsilon centered at each endpoint. The band for a polygon, containing multiple segments, would then be the union of the individual bands for all the segments of the polygon. The polygon boundary essentially splits the band into two bands, for the inside and outside the polygon.

The rectangles defined by each line segment and a line parallel to the segment make up the greatest proportion of the band. Without taking the effects of the endpoints of each segment into account, perimeter times epsilon distance would result in rectangular bands associated with the inside and outside of the polygon boundary, whose total areas are equal. Where the segments meet, the rectangular bands also meet at that common endpoint. Unless the angle between the segments is 180 degrees, the rectangular bands will overlap on the concave side and fail to meet on the convex side. The resulting effect is to overcount the area on the concave side and undercount the area on the convex side (see Figure 2.9). If epsilon is very small, the effects of intersegment angles is slight, but as epsilon increases the proportion of overcount and undercount in relation to the basic perimeter bands becomes significant (Chrisman 1982a). Adjustment of the effects of intersegment angles results in asymmetric outer and inner bands, and likewise asymmeteric upper and lower bounds for the polygon area.

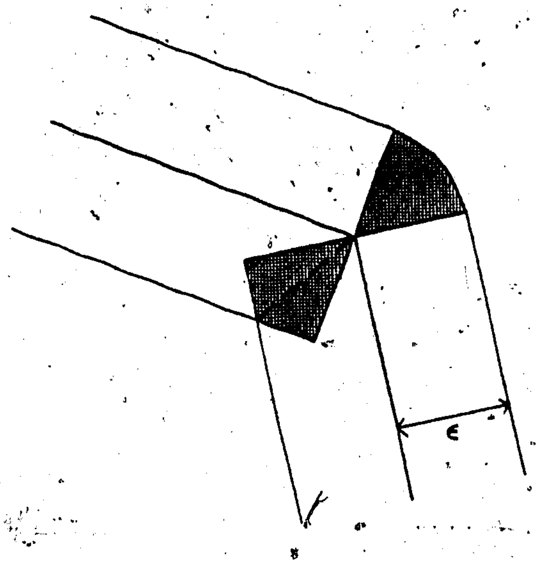


Figure 2.9 Effects of intersegment angles on area of epsilon bands. Shaded areas show overcount on concave side of angle, undercount on convex side (after Chrisman, 1982a)

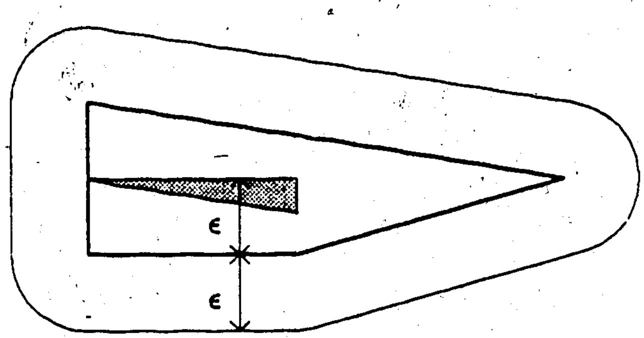


Figure 2.10 Complete overlap of inside epsilon bands, shown in shaded area

Further complications arise in construction of epsilon bands when the shape of the polygon causes nonadjacent bands to overlap. This condition is most likely to occur for inside bands, where the distance between segments that are not adjacent to each other is less than twice epsilon for part or all of their length. The area of overlap, if not subtracted, represents an overestimation of the lower bound of the polygon area, that is, the estimated lower bound area will be smaller than the "true" lower bound. In an extreme case, where all inside bounds overlap, the lower bound is negative, though realistically the "true" lower bound is zero area (see Figure 2.10).

The occurrence of zero or negative lower bounds for a polygon has significance in that it indicates that the variability of the polygon area, due to its size and/or shape, is so great that at one extreme the polygon would not exist; instead, the area within the polygon would actually belong to different, neighboring classes. The questionable existence of the polygon is a result of error associated with the polygon boundary, error accumulated in the cartographic process. Spurious polygons occur for the same reason, that is, cartographic error creating a polygon that should not exist. The relationship between this characteristic of the epsilon bands and the presence of spurious polygons will be explored later.

2.7 Implications for the Application and Analysis

The following sections focus on a specific application of polygon overlay with categorical maps of geomorphic data. The success of the application, from the point of view of geomorphic research, depends largely on the accuracy of the results. The accuracy of overlay results with polygon data is largely a function of the accuracy of the digitized maps. It is therefore essential, in assessing polygon overlay for use with geomorphic data, to quantitatively assess the error involved in the mapping process and its transmission to the overlay results. The epsilon distance model provides a means for doing so. Using this model, the effects of cartographic error can be expressed as an amount of reliability associated with the original maps as well as the resulting composite map.

3. Methodology of Polygon Overlay and Epsilon Band Measurement

3.1 Introduction

When working with applications in an automated data processing environment, the hardware and software configuration employed define part of the methodological framework. The first section of this chapter briefly describes the computer graphics system used in the application study. The actual steps taken in the overlay processing, from acquisition of the original mapped data, to generation of the overlay results and associated data, are then described. Finally, the steps taken to calculate the epsilon error bands are explained.

3.2 The Intergraph System

The system used is a turnkey interactive graphics system, designed by Intergraph Corporation, running on a VAX-11/730 minicomputer. The Interactive Graphic Design Software (IGDS) is a group of graphics programs used for input of design elements and digitized map features directly to a disk-resident file. A free-floating cursor is used to define graphic elements on the design surface, which may be either the command menu table or the digitizing table. The IGDS includes subsystems for file management and output of graphics to plotting devices. The IGDS is designed to interface with other Intergraph software systems, such that these systems may be accessed while the user is in graphics mode. The result is a flexible and user-friendly environment for building cartographic data files to use with database management and analytical software.

Design files are open-ended and must occupy contiguous space on the disk to be accessed by IGDS. Graphic coordinates are stored in 32-bit precision, resulting in a design plane of 4,294,967,296 units of resolution in both X and Y. Graphic elements may be assigned to 63 display planes in the file, called levels, which may be viewed separately or in any combination.

3.3 Graphic Polygon Processing Utilities (GPPU)

Intergraph's GPPU is described as "a single, fully integrated software package used to digitize, create, edit, analyze, and display thematic map data."[1] GPPU can generate polygon boundaries from a line string network or will accept boundaries already digitized as closed polygons. In either case, each polygon must have its boundary defined by a line string whose starting and ending points are identical. Thus boundaries between contiguous polygons exist twice in the design file, as with the simple vertex list data structure described in Chapter 2. GPPU includes processors capable of checking polygons for conditions such as unconnected line string ends, loop errors in digitizing, and polygons residing on the same level whose boundaries overlap. All of these conditions are invalid for polygons. Information in a database can be associated with a polygon by an element linkage. A linkage is a tag or pointer at the end of an element which refers to a particular occurrence record in the database (Intergraph Corp. 1985a).

The analytical processors of GPPU use element list files as input. This file contains the disk sector/byte pointer of each element, the design file specification, and the associated database specification (Intergraph Corp. 1985c). Six analytical processors, including the list file generator, are available.

Polygon overlay is done using the Polygon Overlay Processor (POP) within GPPU. Using polygon element list files as input, POP performs six functions. Three basic overlay functions are intersection (AND), union (OR), and exclusion (NOT). Two extended logical functions are also available; exclusive union (XOR), and inclusive union (OR2). Finally, a function to merge contiguous or overlapping polygons is available. When comparing two sets of polygons, GPPU designates the first input set as the *subject* set, and the second input as the *criteria* set. The subject set is tested to see if it satisfies the relationship to the criteria set as stated by the logical operation used. Comparison of subject and criteria sets for intersection could be stated as the question, "Does polygon A (subject) satisfy the criteria

[1]Intergraph Corp., *GPPU Users Guide (8.8)*, 1985

that polygon A intersects with polygon B (criteria)?"

For any of the logical functions, the basic operations are line segment intersection between two polygons, followed by construction of new polygons representing the correct solution to the logical function. This can require on the order of n^2 comparisons for two sets of polygons represented by a total of n vertices. Reduction in this maximum number of intersection calculations is achieved by performing a bounding rectangle check on pairs of polygons as they are processed. In some cases, further efficiency can be achieved by ordering the input of element list files such that the set with the larger number of polygons is used as the subject set, and the smaller number as the criteria set (Teng 1985, personal communication).

Like other programs operating in a non-topological environment, GPPU creates temporary files and tables during the comparison of polygon sets to "track" the resultant polygons during their creation. Detailed information on exactly how this is done is not available from Intergraph Corp. A general description of the processes involved is given by Teng (1985, personal communication)

> "A pre-processor strokes the polygon vertices and other relevant header information for all the polygons identified in the input list files into work files. The islands (if any) are properly identified in these work files. In performing the overlay analysis, two sets of tables are created to store, at any given time, the relevant information describing one polygon (and any islands) from the subject set and one polygon (and any islands) from the criteria set. Based on the type of Boolean operation, a range rectangle check [is] performed to see whether this pair of polygons overlay. If they overlay, further processing is performed to determine the intersection points. Again, based on the Boolean operation specified, resultant polygon(s) are created from the pair of 'parent' polygons. When all the analysis is done, a separate sub-processor generates an output design file containing all the resultant polygons."

Of primary importance to this application is the retrieval of areas and perimeters of the original and resultant polygons. This task is handled by another GPPU processor named AREA. The AREA processor calculates the area and perimeter of individual polygons using an element list file as input. These values are stored in database occurrences linked to the polygons.

3.4 The Database

As a requirement of the AREA processor and thus a necessary step in the application study, a database structure was defined using the Data Management and Retrieval System (DMRS). DMRS uses a network structure in the definition of database files. The network structure consists of elements which can be linked to any other elements and may or may not be arranged in a hierarchy of levels. The elements in a DMRS database are known as entities. Entities are related to each other with owner/member relationships, which are analogous to parent/child relationships in a hierarchical structure. [2] Records within entity files are called *occurrences*, and contain data items known as *attributes*.

The primary DMRS file is the schema file, which is created by the Data Definition Language (DDL) compiler. Subfiles in the schema file define the structure of the entity and attribute data records within the entity file. [3]

For the application study, each category of polygons within the study area (e.g., slope and surface feature) is associated with an entity level in the database structure. Information about each entity level is stored in an entity file in the form of attribute values. Individual graphic elements, or polygons, are linked to single occurrences in the corresponding entity file via an attribute linkage. The attribute linkage definition contains linkage type, entity number, and occurrence number, pointing to a unique occurrence of entity/attribute information in the database. [4] The name of the database is specified in the design file header. When the attribute linkage is created, a bit is set in the graphic element header information to indicate that the element has a linkage, and the linkage information is appended to the element.

The relationships between entity levels forms a simple two-level hierarchy of entity files shown in Figure 3.1. The entities shown [5] are numbered 1 through 5, to the left of the decimal point. Attribute value names are listed below for each entity, denoted by numbers to

[2]Intergraph Corp., *DMRS Data Definition Language Users Guide (8.8)*, 1985.
[3]Intergraph Corp., *DMRS Control Block Users Guide (8.8)*, 1985.
[4]Intergraph Corp., *Application Software Interface Document, Appendix C*, 1984.
[5]A total of 12 entities exist in the database, to accommodate data for other entity levels not used in this application.

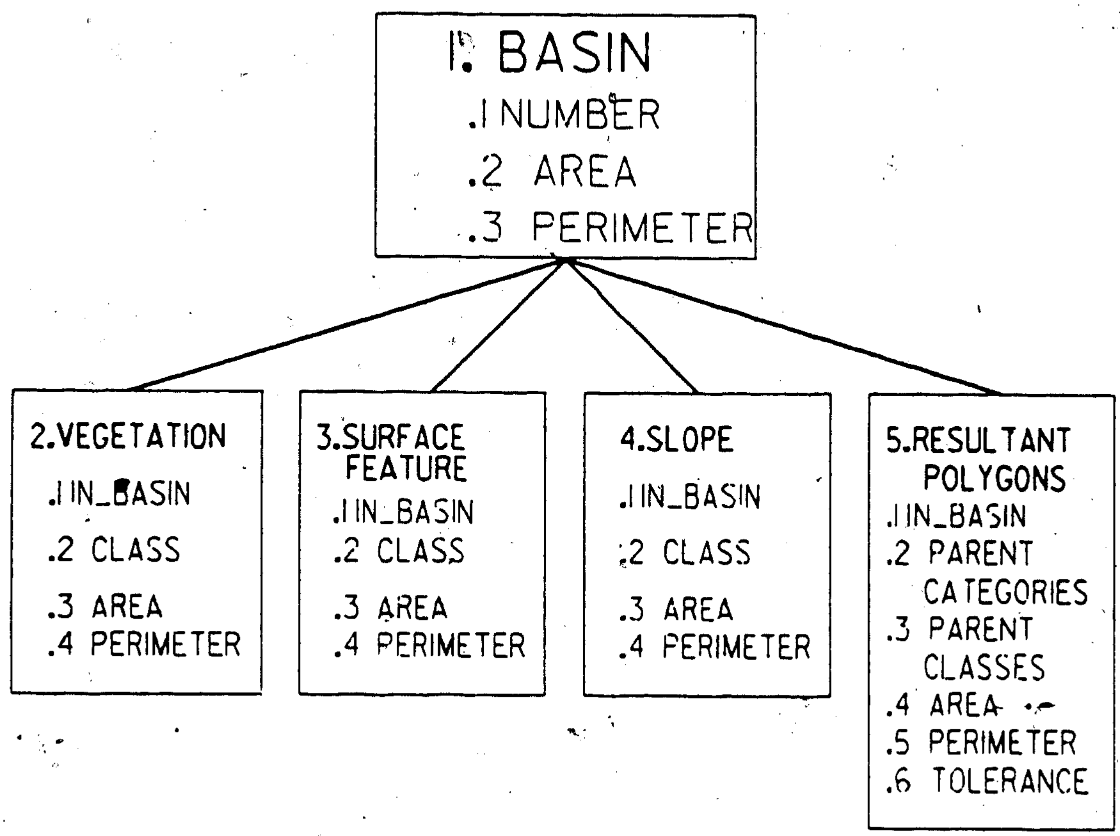


Figure 3.1 Database hierarchy of entity files.

the right of the decimal point. The boundary of the drainage basin defines the extent of the polygon categories, and resides at the "highest" entity level. Entity files for the polygon categories and their overlay resultant classes reside at the "lower" level. Polygons are associated with the basin entity level by a pointer attribute, named IN_BASIN, which contains the basin occurrence number. This attribute was created so that polygon data from neighboring drainage basins might be added in the future. Attribute values for each polygon occurrence in an entity file includes its class name, area, and perimeter. Attribute values for polygons resulting from POP processing include names of parent polygon classes and categories.

3.5 Preparation of the Data

3.5.1 Organization of the Source Maps

Within the drainage basin described in Chapter 1, a detailed field survey was undertaken to map the categories of three classes of interest: vegetation, surficial lithology, and surface slope. Boundaries between classes within each category were compiled directly on to enlarged aerial photographs of the study area, at a scale of 1:1 600. For each class, the field observations of the boundaries were traced in ink directly onto stable film, along with the boundary of the drainage basin.

Prior to digitizing, some of the classes on the original maps were aggregated (see Table 3.1). The aggregated categories represent similar features which do not need to be differentiated for the purposes of the overlay analysis. Two of the three original classes of the vegetation map were combined, resulting in a binary vegetation classification of "vegetated" and "unvegetated." Aggregation of lithology classes by similar characteristics resulted in seven digitized categories from twelve original categories. None of the four original slope classes were aggregated.

Table 3.1 Mapped classes of the study area, before and after aggregation.

Original Classes	Aggregated class
Shale Disturbed (mass movement)	Shale
Sandstone	Sandstone
Ironstone (bedrock and fragments)	Ironstone
Alluvial	Alluvial
Pediment/Fan Sandstone (pediment/fan)	Pediment/Fan
Unknown (grass covered)	Other
Interbedded shale and sandstone Interbedded shale and ironstone Interbedded sandstone and ironstone Interbedded shale, sandstone, and ironstone	Interbedded
0 to 4 degrees	0 - 4
4 to 10 degrees	4 - 10
10 to 30 degrees	10 - 30
Greater than 30 degrees	> 30

It is important to note here, that although the maps for all three main categories were digitized, the actual overlays were done for only two of the three original mapped categories; those of surficial lithology and slope. This is because the lithology class originally mapped as "unknown" represents areas where the surficial lithology class was not determined due to the presence of vegetation. As a result, most areas of vegetation and "unknown" lithology are identical and thus the vegetation classes are represented by a single lithology class, which was renamed as "other." The combined categories of lithology and vegetation are thus referred to as "surface feature" throughout this thesis.

3.5.2 Digitizing

The digitizing of the three source maps was done prior to the installation of the GPPU software, which was not available at that time. This was not considered to be a problem, as GPPU accepts lines and polygons for input that are created using the IGDS digitizing procedure. Point mode digitizing was used, as it was felt to be adequately accurate for the number and size of the polygons involved. Another consideration for using point mode digitizing was the need to fit contiguous polygon boundaries together, which is not possible with stream mode digitizing.

To facilitate input to GPPU, each class of the slope and lithology categories was digitized as a set of mutually exclusive polygons on a separate level in the design file. Polygons which required less than 100 points for their definition were digitized as *simple shapes*. A simple shape is an IGDS graphic element type which is essentially a single line string with identical beginning and ending vertices. Polygons requiring over 100 points were digitized as *complex shapes*. A complex shape is an IGDS graphic element type composed of multiple line strings connected at their end points. The stream network within the basin boundary, as well as the basin boundary itself, were also digitized. Original digitizing did not include the extensive "background" classes present on the slope and lithology maps. These are the "4-10" slope class and "other" surface feature class. These classes were created manually after the polygons in the other classes were edited, by defining polygons enclosing the remainder of the map area not previously digitized. Figures 3.2 and 3.3 show the final digitized polygons for surface feature and slope categories, respectively.

3.5.3 Editing and Data Organization

The digitized polygons were checked level by level for invalid conditions, such as crossing line segments or non-closing shapes, using the checking processors named LINCK and POLCK. This processing uncovered errors in digitizing attributable to the use of the IGDS snap lock function during digitizing. The operator attempted to match precisely the beginning and ending points of the polygons, and the boundaries of contiguous polygons, by searching for and locating the exact position of the point to be matched, and placing the new point there. The snap lock function, initiated with the hand-held digitizing cursor, performs the search for the point to be matched. However, this function does not match points exactly, resulting in open polygons, and gaps and overlaps between contiguous polygons. The open polygon conditions were corrected manually. Inexact matching of polygon boundaries could not be rectified with available GPPU processors.

The conditions of contiguity and mutual exclusivity are necessary for two important parts of the application: for obtaining correct results from overlay processing, and for maintaining exhaustive coverage within the map area. To correct the inexact matching of neighboring polygons within the slope and lithology polygons, a FORTRAN program named MERGER was written by the author. The program requires as input a list, sorted on x, of vertices with their corresponding level, polygon number, and vertex sequence number information. Points in neighboring polygons that are located within a tolerance distance of each other are merged, i.e., their mean location is computed and the points in question are moved to that location. The new, merged list of vertices is then sorted by level, and again by polygon information, to restore the order needed to construct the polygons in a new design file. Existing applications programs were modified to generate both the input lists and the

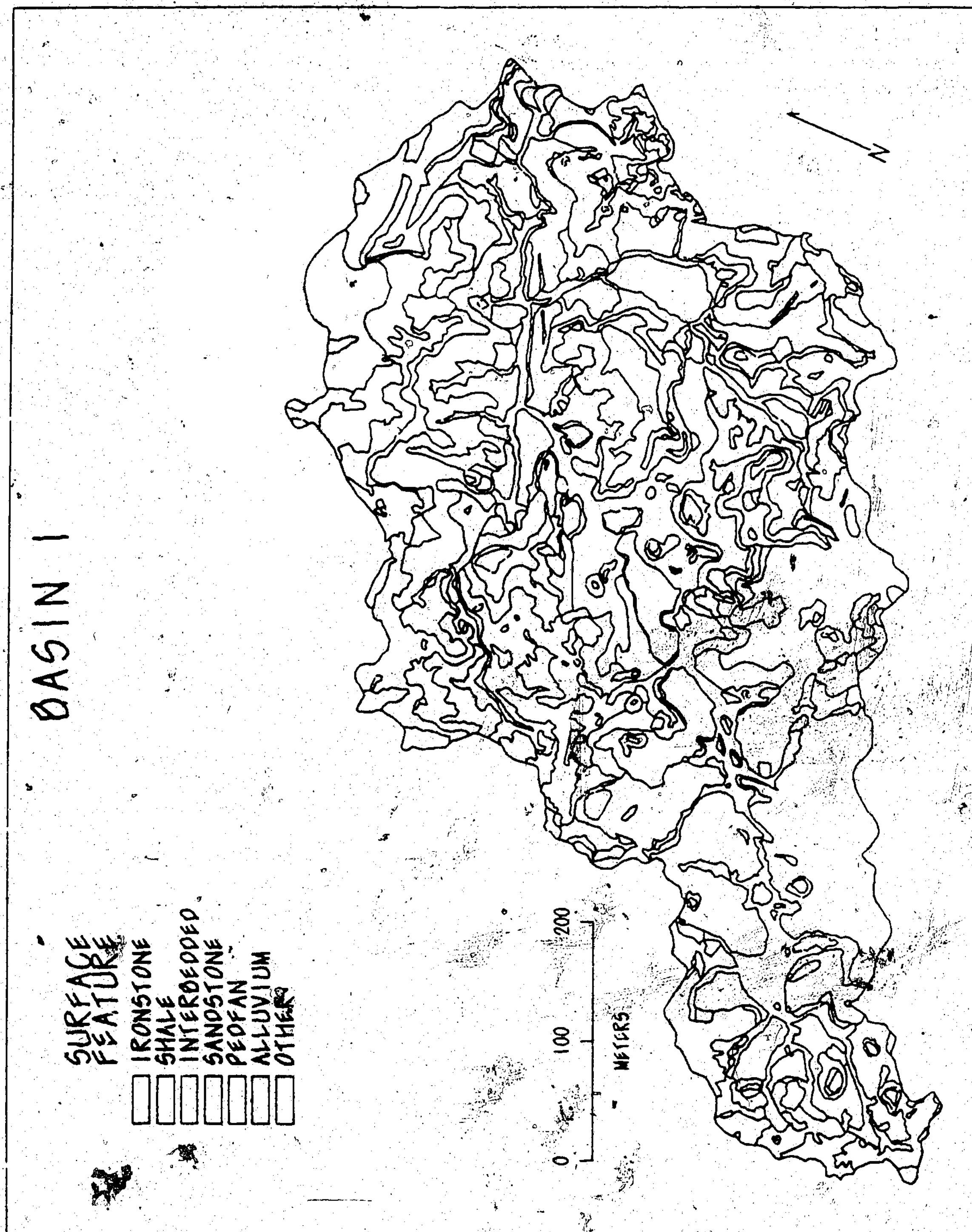


Figure 3.2 Digitized map of surface feature polygons.

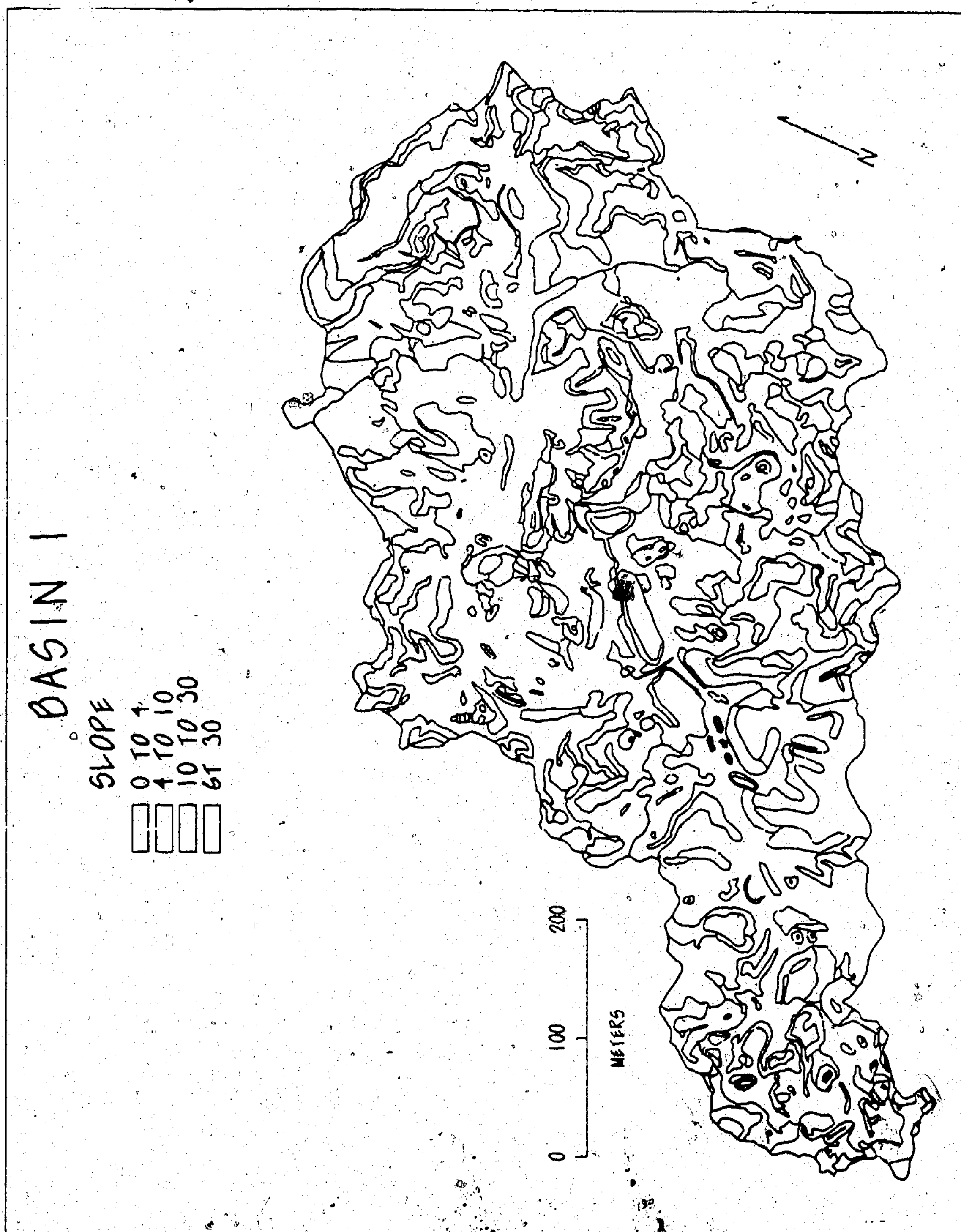


Figure 3.3 Digitized map of slope polygons.

resulting graphics files. Documentation and source code for the program MERGER is provided in Appendix 1.

One characteristic of the maps used in this application is that polygons belonging to one class often completely surround polygons belonging to other classes of the same category. This is especially true for the lithology classes. This condition of nested polygons required additional structuring of the polygon data in the design file. To compute the true area for a class, the area of the interior polygons or "holes" must be subtracted from the total area. Also, the perimeter of the holes must be added to the total perimeter to correctly give the linear contact between classes. Polygons which are interior to polygons of another class represent two quantities. First, they represent positive area and positive perimeter of their own class, and second, they represent negative area and positive perimeter of the surrounding class. The interior polygons for a single class may be from multiple classes.

During calculation of area, the subtraction of interior polygons is handled by the AREA processor. Two inputs are necessary: the list file for the polygons making up the desired class, and a list file identifying the holes. In order to generate this second input file, all polygons interior to one polygon class were duplicated in the design file at a single level separate to all other polygon classes. This was done for all slope classes and all lithology classes except "ironstone", which did not have any interior polygons. It was not necessary to include these duplicated polygons as attribute occurrences in the database.

Once the polygons were edited and organized, corresponding database occurrences were created and attached. In graphics mode, the database was linked to the design file using the IGDS Attribute Services task, which provides an interface between IGDS and DMRS. With this interface, an attribute occurrence was created automatically each time a polygon was selected for attachment, as well as the element linkage connecting the polygon to the occurrence. This was done in a single operation for each classes of the two categories by selecting all the polygons of a class as a group instead of individually.

3.5.4 Overlay Processing

Figure 3.4 illustrates the processing steps necessary for overlay, beginning with the edited, structured map design file to the final product, which is a tabular report and summary of the polygon areas and perimeters. The initial task for both overlay and area calculation is to generate element list files for each class of the two categories to be overlaid. For each list file, a corresponding list file containing interior polygons, when they exist, must also be generated. After generating the list files for input, the processing is carried out in two phases for each set of files.

First, total areas and perimeters of the two original classes are obtained. The files are input to the AREA processor, and area and perimeter for each polygon are output as attribute values in the corresponding occurrence in the database, via the element linkage. At any time after storing the area and perimeter values, they may be retrieved in a tabular report, using the DMRS command language to access and retrieve attribute values. The values are output according to a format specified by the user in an extended command procedure.

The second processing phase begins with the actual overlay. The same list files used in the first phase are used as input to the polygon overlay processor. As described earlier in this chapter, a new design file containing the resultant set of polygons is generated. An element list file for the resultant set is then created by the user, along with with a corresponding list file for the interior polygons in the resultant set. From here on, the processing steps are the same as in the first phase: area and perimeter calculation, storage in database occurrences, and report generation.

In the application for this thesis there are seven classes for surface features and four for slope. Thus the processing steps just described for the overlay of the two categories were performed 28 times, creating 28 separate new design files to contain the resultant polygons. These files were merged into a single design file. Polygons on each level were linked to the database entity created for resultant polygons, and attribute values according to parent classes stored in each occurrence. Areas and perimeters for all resultant polygons were then calculated

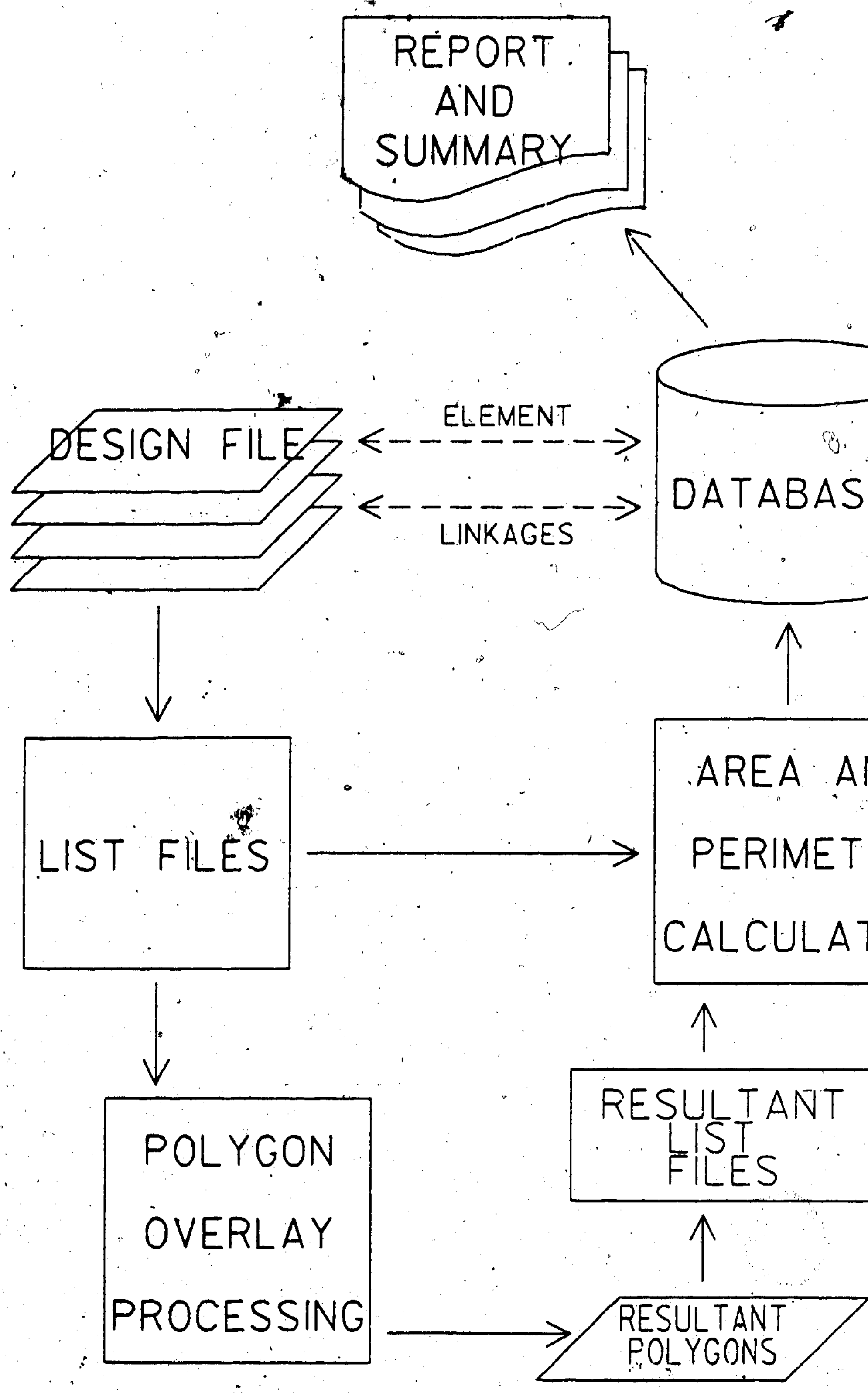


Figure 3.4 Processing steps from digitized design file to report retrieval.

and stored in the database.

3.6 Measurement of Epsilon Bands

GPPU software includes a processor named ZONE which is used to generate minimum distance zones around point, line, or polygon graphic elements. The minimum distance zones are identical to epsilon bands. The ZONE processor creates graphic elements defining the minimum distance zones which are written to an output design file. ZONE uses element list files as input.

From the onset of the application study, it was planned that the ZONE processor would be used to generate inner and outer epsilon bands. The advantage of using ZONE is that the graphic elements created to define the bands around the polygon do not overlap. Areas where bands of nonadjacent segments overlap are eliminated. This provides a precise measure of the epsilon band. For polygons inside which all the bands overlap, no graphic element can be generated.

At the time when the overlays were completed and it became necessary to generate the epsilon bands, the ZONE processor was found to return incorrect results. In an initial attempt, it was found that for some polygons no zone elements were created to describe the inside zone, while the outside zone was correctly created. The opposite case applied to other polygons. In other cases the graphic element generated for the zone switched from outside to inside, or vice versa, crossing the polygon boundary along a radius generated around a vertex. These problems were not solved through communication with Intergraph software support by the time of this writing.

A FORTRAN program named EBAND was written by the author to calculate the areas of the epsilon bands. Like the program MERGER described earlier in this chapter, it uses as input a list of (x,y) coordinates and associated design file level, polygon number, and vertex sequence number information. Area of the epsilon bands is calculated for the inside and outside of each polygon by traversing around the polygon perimeter, one segment at a time. For each segment, the basic epsilon area, width times perimeter, is calculated first, then adjusted for areas of overcount and undercount at the angle between the current segment and the next segment. In order to correctly assign the adjustments to the inside or outside of the polygon, the concave angle relative to the direction of the traversal is determined. The overcount is then subtracted from the concave side, and the undercount added to the convex side. This procedure is repeated until the end of the polygon is encountered. The subroutine used to calculate the intersegment angle and to determine the concave side was adapted from one used in the program MEASURE, written by N. R. Chrisman (see Chrisman 1982a, pp. 205-211). The program EBAND provides a better measure of epsilon band area than perimeter times epsilon width, but is limited. Adjustments for overlap of nonadjacent segments are not handled, as this is a very difficult programming task.

Output of EBAND includes the area of the inner and outer epsilon bands, the measured area and perimeter of the polygon, the upper and lower bounds of the polygon area, and the percent difference that the inner and outer bands represent. These figures are output for individual polygons and totaled for each class, and for all classes in the original and resultant polygon maps. Documentation and source code for the program EBAND are provided in Appendix 2.

3.7 Estimation of Epsilon Distance

Determination of the amount of error, epsilon, associated with the polygon boundaries is based on the known error effects accumulated in each step of the mapping process. These effects were described in section 2.5. It was assumed in section 2.6 that the combination of errors involved results in a normal distribution of error about the line, with a total variance equal to twice the value of epsilon. This is possible because even though some errors effects may not be normally distributed (such as digitizer resolution), when the distributions of different error functions are added, they approximate a normal distribution (Chrisman 1982a).

Three estimates of epsilon distance will be calculated for the maps used in this thesis, spanning a range of error likely to be associated with these maps. The value of each error effect is considered to be the standard deviation of measurement of any point on the line associated with its respective process. The combined effect of all the individual error effects is determined by the calculus of probability functions, as the square root of the sum of the individual errors squared (Blumenstock 1953; Chrisman 1982a). The total, combined effect is halved to arrive at the epsilon distance on either side of the line. Estimates of error effects are given in working scale, then converted to map scale once the total epsilon value is calculated.

The first, smallest estimate of epsilon is based on a minimum amount of error in the maps and in the digitizing process. Here it is assumed that there is negligible error in the transfer of boundaries from the manuscript map and in previous stages of compilation. It is also assumed that the error in digitizing is minimized, such that the cursor "spot" or crosshair is always within the width of the line, or at the very most, the edges of the cursor spot and the line are coincidental. Digitizer resolution is also included as an error effect in this measure.

Assuming no error prior to the working maps should not be done under most circumstances, as this ignores the map history as well as inherent error associated with map content. However, this is often what happens when maps with no history or a vague history are encountered. According to Cook (1983, p. 64), information on data quality and accuracy is frequently lost or neglected, and subsequently the uncertainty of the map is ignored. The Canadian Council on Surveying and Mapping (1984, p. 54) suggests that, where an estimate of accuracy cannot be made, the accuracy of the map should be classified as "unknown." The assumption of no error prior to the final maps is made here intentionally to create a range of epsilon distances, this first measure representing the minimum probable error.

The second epsilon distance incorporates human error in digitizing and error in transfer from the manuscript map. The line width of the manuscript map is assumed to be 0.5 mm. The digitizing error is set at .125 mm, which represents the distance away from the edge of the map line that the cursor can stray before a gap is readily detectable by the operator. This total measure of epsilon could be taken as one where steps in the mapping process are well-controlled; the manuscript is of good quality and digitizing error, though reasonable, is low.

The third measure of epsilon incorporates the error effects used above, with the addition of registration and medium effects. These errors are difficult to estimate, and here it will be assumed that misregistration and the instability of the film base might cause identical boundaries on two different maps to be displaced by the width of the final map line, or 0.3 mm. In addition, a less conservative estimate of digitizing error is used to account for the possibility of larger deviations from the line. All epsilon values calculated are rounded to the nearest 0.1 meter, at map scale.

The three epsilon distances used were calculated as follows (map scale = 1:1 600):

1. Error effects:

Line width of final map	0.3 mm
Cursor width	0.2 mm
Digitizer resolution	0.0254 mm
Combined effects:	0.3614 / 2 = 0.1807 mm
Epsilon distance at map scale:	290 mm or 0.3 meters

2. Error effects:

Line width of manuscript	0.5 mm
Line width of final map	0.3 mm
Cursor width	0.2 mm
Digitizer resolution	0.0254 mm
Digitizing error	0.125 mm
Combined effects:	0.6295 / 2 = 0.315 mm
Epsilon distance at map scale:	504 mm or 0.5 meters

3. Error effects:

Transfer from manuscript	0.5 mm
Line width of final map	0.3 mm
Cursor width	0.2 mm
Digitizer resolution	0.0254 mm
Digitizing error	0.5 mm
Registration and medium effects	0.3 mm
Combined effects:	0.8489 / 2 = 0.4245 mm
Epsilon distance at map scale:	679 mm or 0.7 meters

4. Presentation and Interpretation of Overlay and Error Measurement Results

4.1 Introduction

This chapter presents the data gathered from the polygon overlay of the two maps and the epsilon band measurements calculated for all polygons of classes in the original and resultant maps. Interpretation of the results is focused on the error measurements. The relationships within the cross-classification created by the overlay are described, but no in-depth analysis is attempted for two reasons. First, the examination is intended to be from a cartographic point of view, since the analysis of cartographic error is the stated objective of this research. Second, although the raw overlay results may lend themselves to further scrutiny, the available analytical methods for categorical maps are limited. The assumptions of standard statistical methods for categorical data are not applicable to exhaustive, two-dimensional coverages (Chrisman 1982a).

In the actual application of polygon overlay, it is assumed that the resultant polygons generated are correct (with inspection to make sure). Errors in the resultant or composite map could not be considered as part of the evaluation as they may be related only to the particular software used.

4.2 Original Categories

Table 4.1 summarizes the dimensions and frequencies of the polygons describing the classes of the two thematic maps which were presented in Figs. 3.2 and 3.3. The map of surface features contains a total of 369 polygons for the seven classes. The four slope classes are defined by 476 polygons. It is evident from the linework in Figs. 3.2 and 3.3 that the boundaries are often highly convoluted, reflecting the complex nature of the badlands topography. The number of vertices used to define these boundaries, ~~accounting~~ for shared boundaries and neglecting the basin perimeter, is over 7000 for the surface feature polygons, and over 6500 for the slope polygons, or an average of 4.4 meters, at map scale, between

Table 4.1 Summary of frequency and dimensions of classes of the original categories of surface feature and slope.

	Number of Polygons	Area	Perimeter	Percent Map Area
Surface Feature				
Ironstone	55	6126	2784	2.2
Shale	115	70327	18436	24.9
Interbedded	30	34509	7193	12.2
Sandstone	[illegible]	16148	5453	5.7
Pediment/Fan	[illegible]	36614	9844	12.9
Alluvial	7	33818	10327	11.9
Other	81	85337	12283	30.2
Total	369	282879	65942	100.0
Slope				
0-4	96	28190	8378	10.0
4-10	108	124422	23311	44.1
10-30	185	55587	18835	19.7
> 30	87	73993	15972	26.2
Total	476	282192	66271	100.0

every point along the polygon boundaries within the basin, for both maps. This is equal to 2.75 mm at the working scale.

The total areas and perimeters given for each class are adjusted for nested polygons. The discrepancy between the total areas for each map is attributable to the digitizing process, in which polygon boundaries were digitized independently. Although every effort was made at that time to match every point along shared boundaries, this was not possible for every case. The program MERGER was successful in correcting points that were very near each other but not coincident, but points which had no corresponding point on a neighboring polygon were not moved. Overlaps between polygons were detected and corrected using GPPU, but manual editing is the only way to correct for gaps between polygons, and like any manual procedure is susceptible to error. The resulting gaps in the slope and surface feature maps amount to a 0.5 and 0.3 percent loss in total area, respectively, which is certainly a reasonable margin of error.

Polygons within each map show a great range in size. Figure 4.1 shows the distribution of polygon frequencies and areas for both original maps. The height of each bar represents the percent of the total map area that polygons in each area class interval occupy. The line graph shows the distribution of polygon frequency within each area class interval as a percent of the total frequency. The range in area values is so great that a logarithmic transformation was performed for all area values in order to make the distribution more readable. Without this transformation, the distributions would show extreme positive skew.

The majority of the polygons occupy a small amount of the map area, for both maps. The class intervals containing the most polygons occur below the intervals in which the greatest proportion of the area is found. The asymmetry of the distributions is reflected by the fact that the majority of the polygons are smaller than the mean polygon area. For surface features, 81 percent of all polygons are smaller than the mean of 766 square meters, yet occupy less than 16 percent of the map area. Similarly, for slope classes 86 percent are below the mean of 596 square meters, while containing 22 percent of the map area. At the upper end of the area scale, there are few polygons. In the slope map, a single polygon from

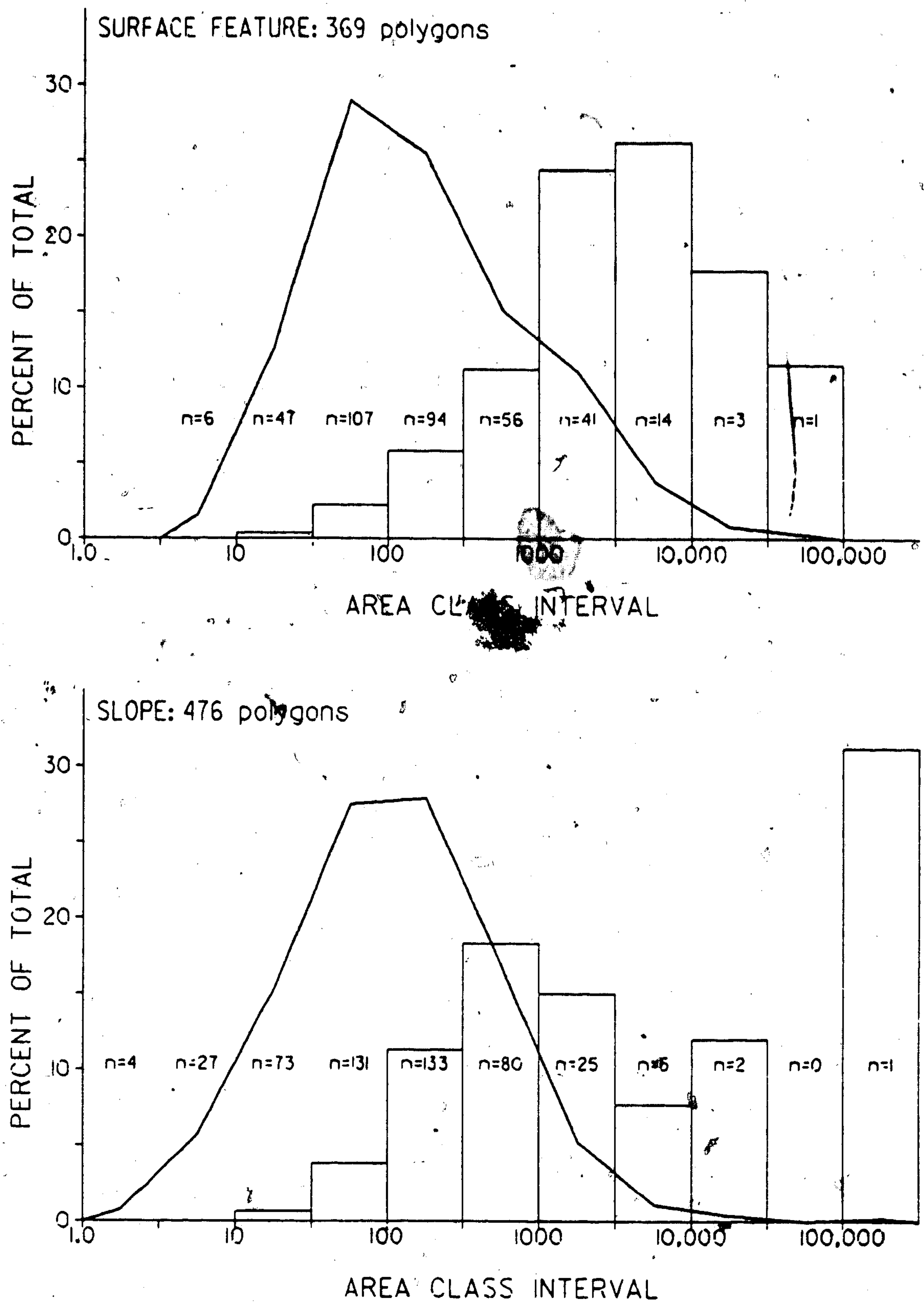


Figure 4.1 Distribution of polygon area (bar) and frequency (line) by area class intervals

the 4-10 degree slope class of over 100,000 square meters occupies over 30 percent of the map area. This particular polygon is very extensive and contains 110 hole polygons belonging to other classes.

4.3 Overlay Results

Table 4.2 shows polygon frequency, area, and perimeter for the 28 classes resulting from the overlay. Each cell in the table contains, from top to bottom, the number of polygons, total area, and total perimeter for the class of the intersection between the original classes given by the row and column. A total of 2723 polygons were generated by the overlay of the two maps, shown in the composite map in Figure 4.2. Table 4.3 shows the percentage that each total within the 28 classes represents of its two "parent" classes, and the percentage of the total map area occupied by each resultant class.

For the intersection of the seven and four original classes, there are 28 resultant classes, meaning that in the composite map all possible combinations of surface features and slopes are present. Likewise, there are no classes from one category which lie wholly within a single class of the other category. The amount of intersection of one class with another, as a percentage of total map area, ranges from 0.8 percent in the case of Ironstone with 0-4, to 17.9 percent for the intersection of Other with 4-10.

In looking for significant characteristics among the resultant classes, it is easiest and perhaps more natural to examine the distribution of surface feature classes across the slope classes, as illustrated in Figure 4.3. Here there are distinct tendencies for the intersection of surface features with certain slopes. Ironstone, Shale, and Sandstone all have a higher degree of intersection with >30 than any other slope class, at over 40 percent of their respective areas. Also, for these three surface features, roughly half of their area is found in 4-10 and 10-30, with very little intersecting with 0-4. Interbedded is similar except that the greatest amount of intersection is the same between 4-10 and >30, at nearly 40 percent each. The other three surface features show a distinct concentration at the opposite end of the slope

Table 4.2 Polygon frequency, area, and perimeter of resultant classes. For each cell, top entry is polygon frequency; center, total area; bottom, total perimeter. Areas given in square meters

Surface Feature	SLOPE 0-4	4-10	11-30	> 30	Row Total Polygons
Ironstone	15 216 264	51 2039 1501	38 1247 1036	47 2597 1616	151
Shale	110 2630 2693	274 14752 9912	229 19849 9703	208 32970 13016	821
Interbedded	50 1915 1167	99 13250 5660	99 6558 3823	56 12957 4536	304
Sandstone	25 702 584	83 4716 2602	64 3613 2100	72 6929 3572	244
Pediment/Fan	63 12646 5000	133 17287 7907	111 3692 2908	91 2941 2619	398
Alluvial	51 4215 2428	79 23601 9026	108 4028 3327	70 3311 2646	308
Other	84 5879 2757	174 49973 13237	130 16433 7226	109 12295 5191	497
Column Total Polygons	398	889	779	653	

Grand Total: 2723

Table 4.3 Areas of resultant polygons as a percent of each constituent class and the total map area. For each cell, top entry is percent of slope class; center, percent of surface feature class; bottom, percent of total map

Surface Feature	Slope 0-4	4-10	11-30	> 30
Ironstone	0.8 3.5 0.08	1.6 33.3 0.72	2.2 20.4 0.44	3.5 42.4 0.92
Shale	9.3 3.8 0.93	11.7 21.0 5.21	35.8 28.3 7.01	44.6 47.0 11.6
Interbedded	6.8 5.5 0.68	10.5 38.2 4.68	11.8 18.9 2.31	17.5 37.4 4.57
Sandstone	2.5 4.4 0.25	3.8 30.0 1.66	6.5 22.6 1.28	9.4 43.4 2.45
Pediment/Fan	44.8 34.6 4.46	13.8 47.3 6.10	6.7 10.1 1.30	4.0 8.0 1.04
Alluvial	15.0 12.2 1.49	18.3 66.6 8.14	7.3 11.6 1.42	4.5 9.6 1.17
Other	20.8 6.9 2.08	40.2 59.4 17.9	29.6 19.3 5.80	16.6 14.4 4.34

Figure 4.2 Composite map of resultant polygons

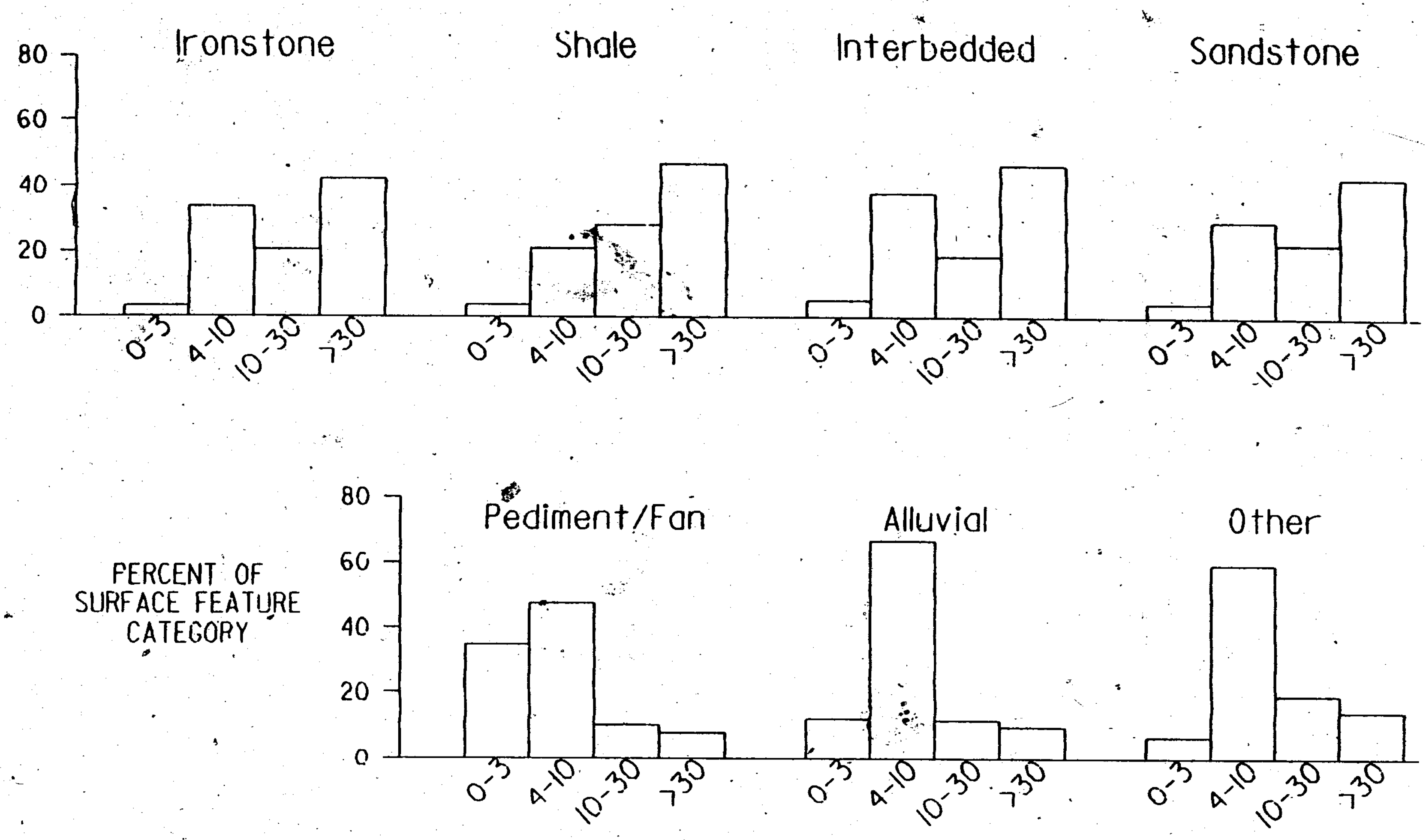


Figure 4.3 Distribution of surface feature classes across slope classes

range. For Pediment/Fan, Alluvial, and Other, the greatest proportion of intersection is with 4-10, as high as 67 percent for Alluvial, with little found in the steeper slope classes and, except for Pediment/Fan, low intersection with 0-4.

Overall, there are two groups of intersection characteristics for surface features. In one group, four of the seven classes show highest concentration in the steepest slope, generally decreasing to the least amount of intersection with the lowest slope. In the other group, the three remaining surface features are predominantly found in the 4-10 degree slope class, with the least amount of intersection with steeper slopes.

Comparison in the opposite direction shows that the distribution of slopes across surface features contains a greater range of proportions, since each slope class is divided among seven surface features. Dominant surface features for some slopes stand out; Pediment/Fan with 0-4, Other with 4-10, and Shale with >30.

A strict geomorphic interpretation of these relationships will not be attempted as it is not within the scope of this thesis. Intuitively, the type of material and its role as an erosional or depositional surface are the most influential factors. The fact that all possible combinations of surface feature and slope are present does seem to have substantiation. It has been observed that, within this study area, slopes of homogeneous lithology are rare, and that slope breaks are found within areas of similar lithology as well as at contact planes between lithological units (Harty 1984). This would indicate that, though there are noticeable tendencies for certain combinations of slope and surface feature, any given combination can be found within the area. However, the effect that cartographic error might have in creating nonexistent combinations or affecting areas of combinations has not been considered.

Figure 4.4 shows the distribution of all resultant polygons. Like the distributions for the two original maps, the majority of the polygons are relatively small in area. For the entire distribution, 81 percent of all polygons are smaller than the mean polygon area of 104 square [illegible]nd occupy 17.6 percent of the map area. In the original maps, the smallest polygons [illegible]s and surface features were around three and one square meters, respectively. In the

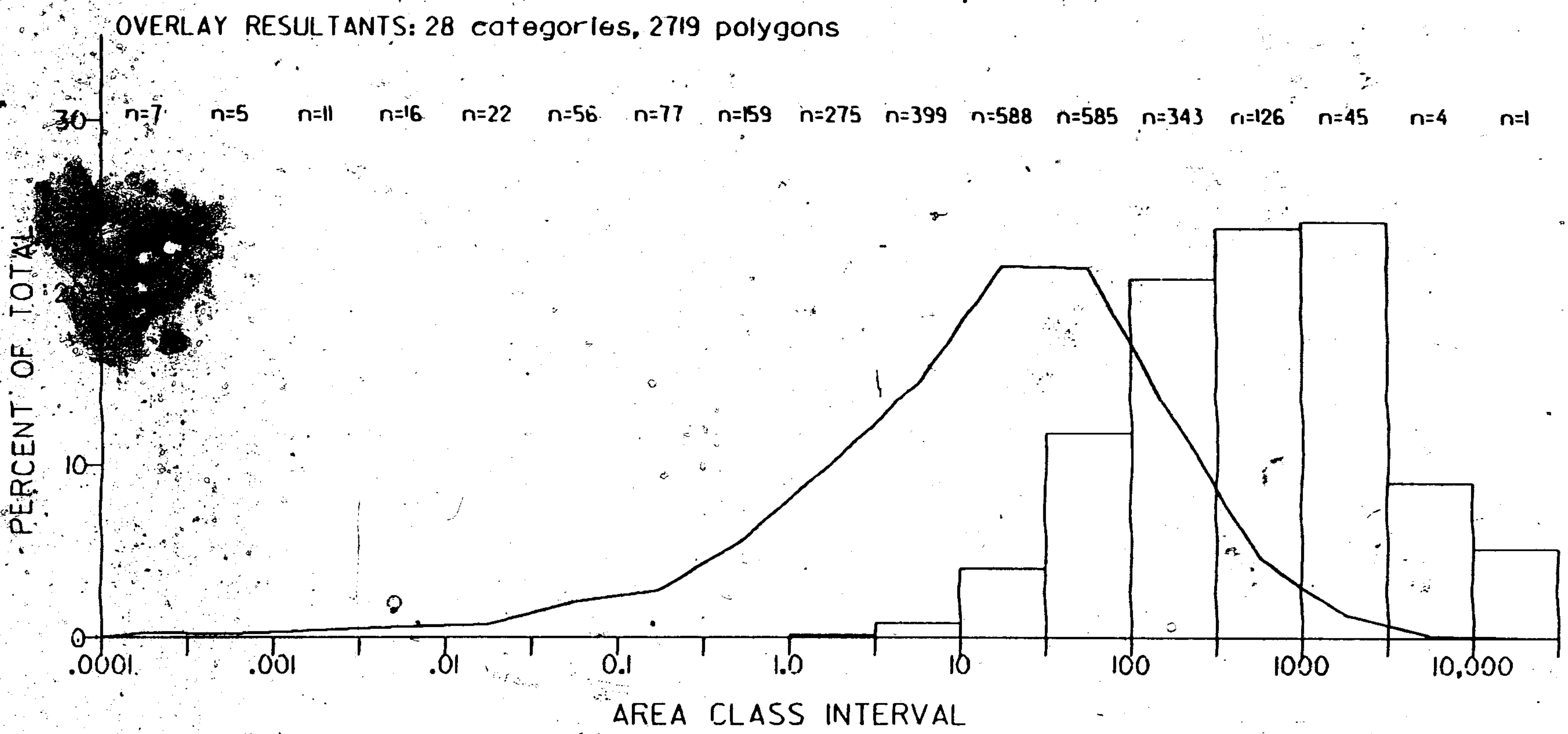


Figure 4.4 Distribution of resultant polygon area (bar) and frequency (line), by area class interval

resultant map, there are 352 polygons smaller than one square meter, or 13 percent of the total, and the smallest polygons created from the overlay are less than .0001 square meter.

High frequencies of very small polygons is characteristic of the effect of overlay (McAlpine and Cook 1971; Cook 1983). The map area has been fragmented by combining the two polygon distributions. Early work on the polygon overlay problem observed that overlay disaggregates the polygons of the original maps to "below the resolution proper to the study" (Chrisman 1982a). These polygons which are less than one square meter are smaller than the smallest mapped polygon, and are below the level of resolution attainable from the field survey. It is impractical to attempt to delineate a polygon smaller than one square meter, even at this large a scale. It is probable that the majority of such small resultant polyons are erroneous and exist only as a result of cartographic error. Additionally, their total area is so insignificant to the whole area (less than 0.1 percent) that they could be disregarded without affecting the results.

4.4 Upper and Lower Bounds on Polygon Area

The following Tables 4.4 through 4.9 give the upper and lower bounds for original polygon classes and for the resultant classes each of the three epsilon distances of 0.3, 0.5, and 0.7 meters. Band areas are calculated for individual polygons and summed for each class. The upper and lower bounds are the addition of outer band area and subtraction of inner band area, respectively.

The percent area that the bands occupy, and thus the percent difference between measured area and the upper and lower bounds, is given in the outermost columns of the table. The inequality in the percentages on either side of the measured polygon areas shows the asymmetry of the band areas caused by intersegment angles.

The upper and lower bounds express the total variability in each class area caused by error in the definition of the polygon boundaries. Of interest is the range of variability within each map for a given epsilon distance. At 0.3 meters, variability of polygon area by class

Table 4.4 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.3 meters. Areas given in square meters

	% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	13.3	815	5311	6126	6969	843	13.9
Shale	7.8	5485	6482	70327	75872	5545	7.9
Interbedded	6.2	2146	32363	34509	36670	2161	6.3
Sandstone	10.0	1618	14530	16148	17789	1641	10.2
Pediment/Fan	8.0	2920	33694	36614	39545	2931	8.0
Alluvial	8.9	3011	30807	33818	36822	3004	8.9
Other	4.3	3644	81693	85337	89027	3690	4.3

Total band area: 20360, or 7.27% of map area.

0-4	8.8	2480	25710	28190	30726	2536	9.0
4-10	5.6	6924	117498	124422	131453	7031	5.7
10-30	10.0	5581	50006	55587	61270	5683	10.2
> 30	6.4	4699	69294	73993	78719	4726	6.4

Total band area: 20732, or 7.31% of map area.

Table 4.5 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.3 meters. Areas given in square meters

Class		% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	+ 0-4	33.5	72	144	216	300	83	38.6
Shale	+ 0-4	28.1	740	1892	2631	3468	837	31.8
Interbedded	+ 0-4	17.0	326	1589	1915	2279	364	19.0
Sandstone	+ 0-4	22.4	157	545	702	884	182	25.9
Pediment/Fan	+ 0-4	11.1	1426	11455	12881	14360	1479	11.5
Alluvial	+ 0-4	15.6	674	3647	4321	5042	721	16.7
Other	+ 0-4	13.2	775	5104	5879	6730	851	14.5
Ironstone	+4-10	20.8	424	1615	2039	2503	464	22.8
Shale	+4-10	18.9	2801	12005	14806	17836	3031	20.5
Interbedded	+4-10	12.0	1603	11801	13403	15094	1690	12.6
Sandstone	+4-10	15.5	731	3985	4716	5519	803	17.0
Pediment/Fan	+4-10	12.8	2243	15323	17566	19927	2361	13.4
Alluvial	+4-10	10.2	2470	21697	24167	26717	2550	10.6
Other	+4-10	7.3	3784	47803	49973	55509	3922	7.6
Ironstone	+10-30	23.6	294	953	1247	1568	321	25.8
Shale	+10-30	13.7	2747	17234	19981	22924	2943	14.7
Interbedded	+10-30	16.5	1086	5492	6578	7745	1167	17.7
Sandstone	+10-30	16.4	594	3025	3619	4623	644	17.8
Pediment/Fan	+10-30	21.9	809	2884	3692	4595	903	24.5
Alluvial	+10-30	23.2	938	3097	4035	5058	1023	25.4
Other	+10-30	11.9	2043	15176	17219	19372	2153	12.5
Ironstone	+ >30	17.7	459	2138	2597	3093	496	19.1
Shale	+ >30	11.2	3716	29850	33206	37106	3901	11.7
Interbedded	+ >30	9.9	1299	11766	13066	14411	1345	10.3
Sandstone	+ >30	14.9	1034	5895	6929	8020	1091	15.7
Pediment/Fan	+ >30	24.9	732	2209	2941	3752	811	27.6
Alluvial	+ >30	22.1	740	2612	3352	4157	806	24.0
Other	+ >30	12.1	1487	10808	12295	13882	1587	12.9

Total band area: 38889, or 13.71% of map area.

Table 4.6 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.5 meters. Areas given in square meters

	% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	21.8	1337	4788	6126	7546	1421	23.2
Shale	12.9	9093	61234	70327	79587	9620	13.2
Interbedded	10.3	3562	30947	34509	38115	3606	10.4
Sandstone	16.6	2679	13469	16148	18891	2743	17.0
Pediment/Fan	13.2	4849	31765	36614	41500	4886	13.3
Alluvial	14.8	5015	28803	33818	38812	4994	14.8
Other	7.1	6042	79295	85337	91505	6168	7.2

Total band area: 34511, or 12.16% of map area.

0-4	14.5	4095	24905	28190	32441	4251	15.1
4-10	9.3	11574	112848	124422	136076	11654	9.4
10-30	16.6	9229	46358	55587	65099	9512	17.1
> 30	10.5	7804	66189	73993	81874	7881	10.7

Total band area: 34504, or 12.16% of map area.

Table 4.7 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.5 meters. Areas given in square meters

Class		% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	+ 0-4	52.4	113	103	216	360	144	66.5
Shale	+ 0-4	44.4	1169	1462	2631	4060	1428	54.3
Interbedded	+ 0-4	27.2	520	1395	1915	2537	622	32.5
Sandstone	+ 0-4	35.0	246	457	702	1013	311	44.3
Pediment/Fan	+ 0-4	18.2	2343	10538	12881	15363	2482	19.3
Alluvial	+ 0-4	25.3	1092	3229	4321	5539	1218	28.2
Other	+ 0-4	21.2	1247	4632	5879	7323	1444	24.6
Ironstone	+4-10	33.3	679	1359	2039	2828	789	38.7
Shale	+4-10	30.5	4518	10288	14806	19939	5133	34.7
Interbedded	+4-10	19.5	2612	10792	13403	16249	2845	21.2
Sandstone	+4-10	24.9	1174	3542	4716	6080	1364	28.9
Pediment/Fan	+4-10	20.8	3659	13907	17566	21539	3973	22.6
Alluvial	+4-10	16.8	4064	20103	24167	28438	4272	17.7
Other	+4-10	12.0	6215	45372	49973	58713	6586	12.8
Ironstone	+10-30	37.8	472	775	1247	1794	547	43.9
Shale	+10-30	22.3	4450	15532	19981	24955	4973	24.9
Interbedded	+10-30	26.7	1758	4821	6578	8553	1975	30.0
Sandstone	+10-30	26.5	958	2661	3619	4712	1093	30.2
Pediment/Fan	+10-30	34.9	1288	2404	3692	5231	1539	41.7
Alluvial	+10-30	37.4	1508	2527	4035	5774	1739	43.1
Other	+10-30	19.3	3331	13888	17219	20847	3628	21.1
Ironstone	+ >30	28.6	742	1855	2597	3438	840	32.4
Shale	+ >30	18.3	6078	27128	33206	39766	6560	19.8
Interbedded	+ >30	16.3	2133	10932	13066	15324	2258	17.3
Sandstone	+ >30	24.3	1686	524[illegible]	6929	8768	1839	26.5
Pediment/Fan	+ >30	39.9	1173	1768	2941	4321	1380	46.9
Alluvial	+ >30	35.6	1192	2159	3352	4715	1364	40.7
Other	+ >30	19.6	2416	9879	12295	14794	2678	21.8

Total band area: 64466, or 22.72% of map area.

Table 4.8 Bounds on area measurements and total epsilon band area for original classes, at epsilon distance of 0.7 meters. Areas given in square meters

	% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	30.1	1842	4283	6126	8136	2010	32.8
Shale	18.0	12666	57661	70327	83318	12991	18.5
Interbedded	14.4	4967	29542	34509	39563	5054	14.6
Sandstone	23.1	3727	12421	16148	20001	3853	23.9
Pediment/Fan	18.5	6765	29849	16614	43453	6839	18.7
Alluvial	20.7	7017	26801	33818	40793	6975	20.6
Other	9.9	8417	76920	85337	93997	8660	10.1

Total band area: 47996, or 16.92% of map area.

	% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
0-4	20.2	5681	22509	28190	34175	21.2	21.2
4-10	13.0	16161	108261	124422	140738	16316	13.1
10-30	23.1	12820	42767	55587	68967	13380	24.1
> 30	14.7	10890	63103	73993	85034	11041	14.9

Total band area: 48241, or 17.00% of map area.

Table 4.9 Bounds on area measurements and total epsilon band area for resultant classes, at epsilon distance of 0.7 meters. Areas given in square meters

Class		% Diff.	Inner Band Area	Lower Bound	Measured Area	Upper Bound	Outer Band Area	% Diff.
Ironstone	+ 0-4	69.1	149	67	216	424	208	96.1
Shale	+ 0-4	59.4	1564	1067	2631	4679	2047	77.8
Interbedded	+ 0-4	36.5	699	1216	1915	2807	892	46.6
Sandstone	+ 0-4	46.1	324	378	702	1149	446	63.6
Pediment/Fan	+ 0-4	25.1	3235	9646	12881	16381	3500	27.2
Alluvial	+ 0-4	34.6	1493	2828	4321	6048	1727	40.0
Other	+ 0-4	28.7	1690	4190	5879	7937	2058	35.0
Ironstone	+4-10	44.9	916	1123	2039	3166	1127	55.3
Shale	+4-10	41.4	6133	8673	14806	22107	7301	49.3
Interbedded	+4-10	26.7	3582	9822	13403	17426	4023	30.0
Sandstone	+4-10	33.7	1588	3128	4716	6660	1944	41.2
Pediment/Fan	+4-10	28.6	5023	12542	17566	23182	5616	32.0
Alluvial	+4-10	23.3	5626	18541	24167	30176	6010	24.9
Other	+4-10	16.6	8586	43001	49973	60877	9290	18.1
Ironstone	+10-30	51.2	639	608	1247	2029	782	62.7
Shale	+10-30	30.4	6072	13909	19981	27041	7059	35.3
Interbedded	+10-30	36.4	2393	4185	6578	9385	2807	42.7
Sandstone	+10-30	35.9	1300	2319	3619	5176	1557	43.0
Pediment/Fan	+10-30	47.0	1734	1958	3692	5894	2204	59.6
Alluvial	+10-30	50.5	2039	1995	4035	6516	2481	61.5
Other	+10-30	26.5	4570	12649	17219	22353	5134	29.8
Ironstone	+ >30	38.9	1009	1588	2597	3794	1196	46.1
Shale	+ >30	25.2	8370	24835	33206	42472	9266	27.9
Interbedded	+ >30	22.5	2945	10121	13066	16249	3184	24.4
Sandstone	+ >30	33.3	2311	4619	6929	9534	2604	37.6
Pediment/Fan	+ >30	53.8	1583	1358	2941	4913	1972	67.0
Alluvial	+ >30	48.3	1620	1731	3352	5291	1939	57.9
Other	+ >30	26.9	3308	8987	12295	16091	3796	30.9

Total band area: 89878, or 31.68% of map area.

ranges from 8.6 percent for Other to 27.2 for Ironstone. At epsilon of 0.5 and 0.7, variability increases consistently with the increase in epsilon. At 0.7 meters, total variability of Ironstone is over 60 percent. For resultant polygons the total variability is noticeably higher. The overlay of the two polygon sets has resulted in fragmentation of the map area into many smaller polygons, and the error bands associated with the now increased perimeter are occupying more area relative to polygon size as well as having increased the total band area within the map. For Ironstone + 0-3, variability exceeds 160 percent.

Generally the total variability is related to total class area; the smallest classes have the highest variability, and the largest have the lowest. Because epsilon times perimeter makes up the greatest proportion of the band area, variability is influenced by total perimeter. For two classes of roughly the same total area, Interbedded and Alluvial, the total band area for Interbedded is 30 percent less than Alluvial for all epsilon distances, because of Interbedded's lower perimeter, or greater compactness. Polygon compactness can be measured using a common index of

$$P/(2\sqrt{\pi A})$$

where P and A are polygon perimeter and area, respectively. Using this index, a circle would have an value of 1, with the value increasing as compactness decreases. This index was computed for individual polygons and averaged for each class. For the classes mentioned above, the average compactness index of Interbedded polygons is 1.92, versus 3.17 for Alluvial.

4.5 Relationships Between Percent Error and Polygon Area

In Figures 4.5 through 4.7 the average epsilon band area, as a percentage of polygon area, was plotted against polygon area for area class intervals. A logarithmic scale was used for both axes. The three lines on each graph show the relationship of band area to polygon area for the three epsilon measures used. This illustrates the proportional increase in band area as epsilon increases. The graphs are parallel and displaced proportionally by the increase

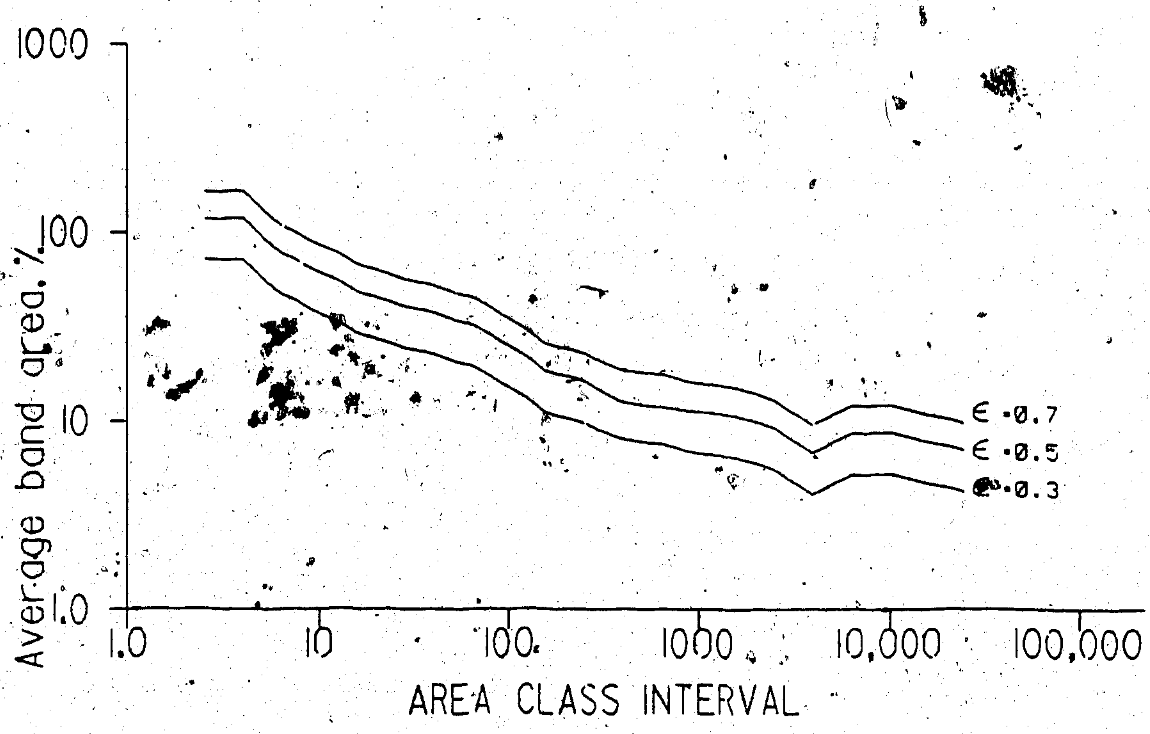


Figure 4.5 Average epsilon band area for surface feature polygons, by area class interval.

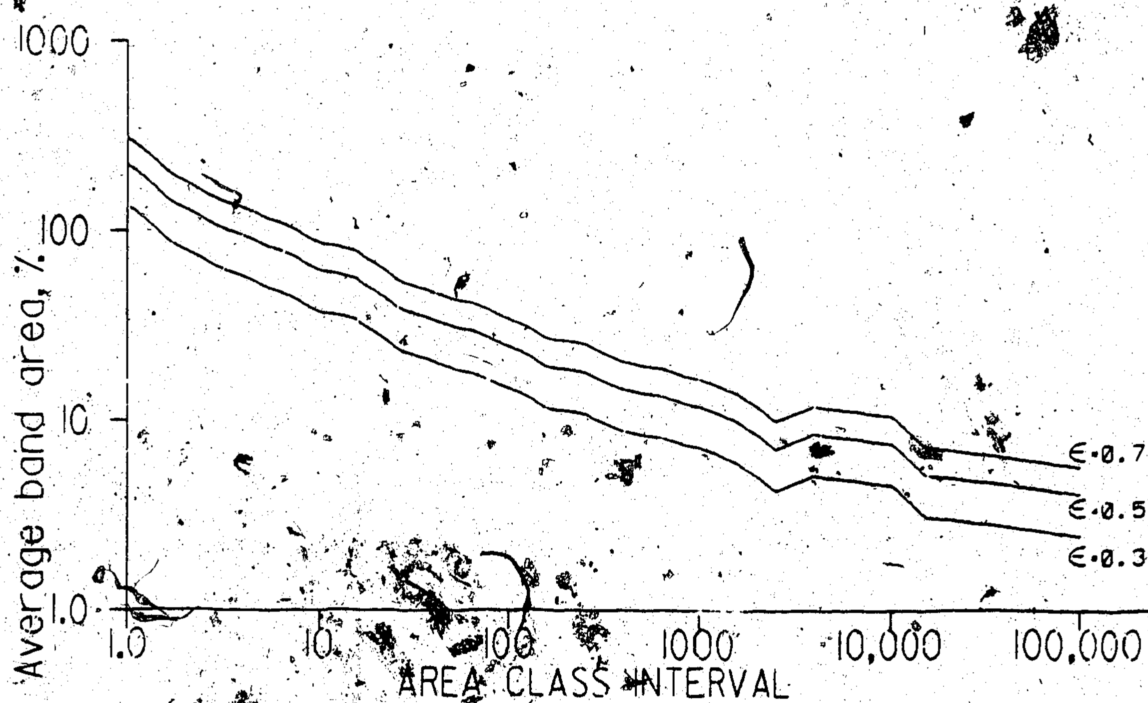


Figure 4.6 Average epsilon band area for slope polygons, by area class interval.

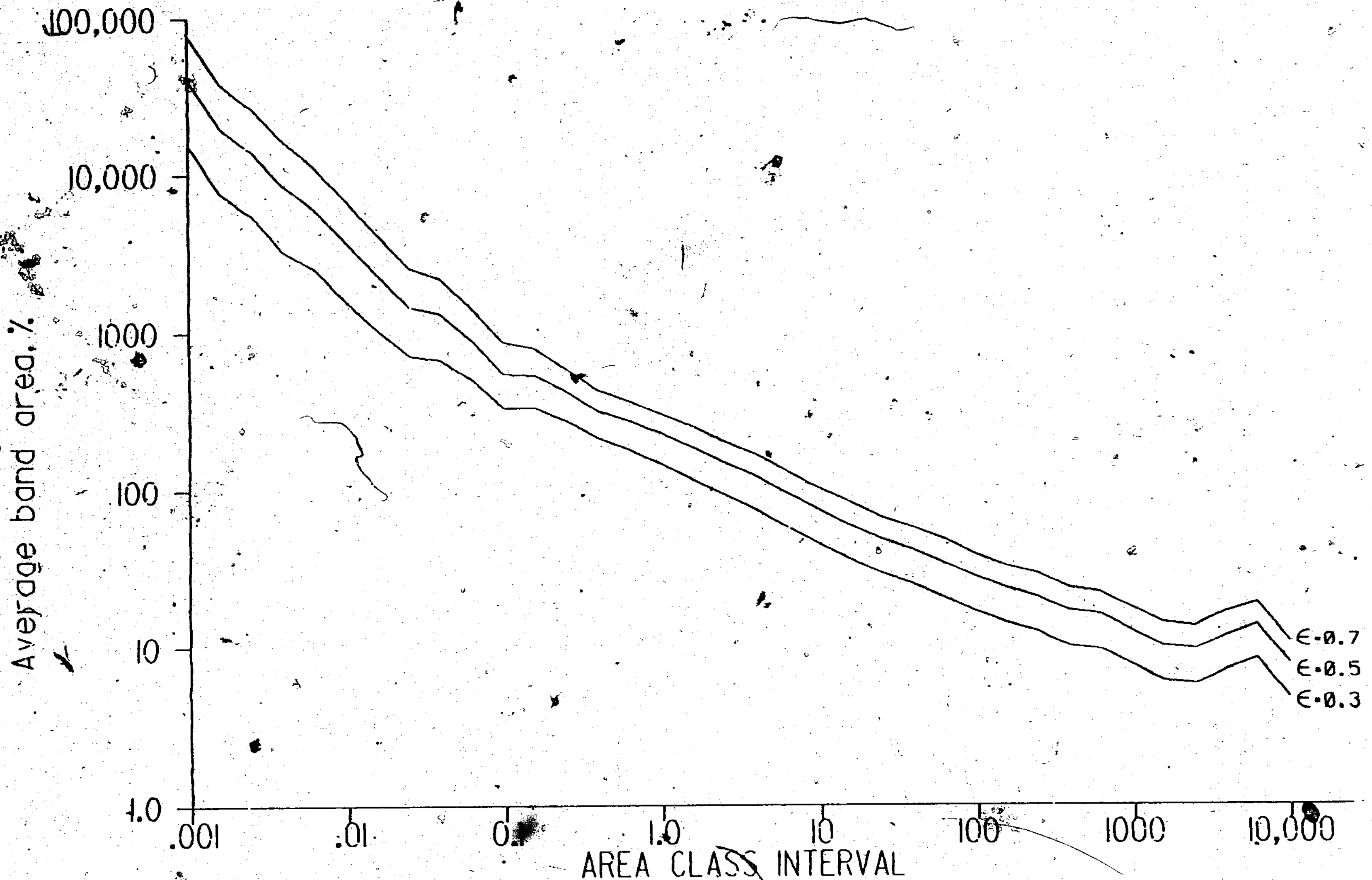


Figure 4.7 Average epsilon band area for resultant polygons, by area class interval

in epsilon distance. If it had been possible to subtract the overlap in bands at nonadjacent segments, a decrease in percentage of area for some class intervals might be seen. Irregularities at very large and very small polygon areas are caused by low frequencies of polygons, but overall the graph seems to show a strong linear trend. Of note are the extremely high values for percent error in the resultant polygons. Because of very small areas, the bands calculated around them (including inner bands which are frequently greater than polygon area), the band area as a percentage of polygon area is huge; in some cases, the bands are over 10,000 percent or one hundred times the polygon area.

Figure 4.8 shows this same relationship for the two original maps and the composite map simultaneously. Since the relationships are identical at all three epsilon distances, comparison at a single value of epsilon is sufficient to examine the relationship between band areas for the three maps. The middle value of 0.5 meters is used.

If band areas at the same value of epsilon are calculated for a circle, the most compact shape possible, the graph of the band area as a percentage of the circle area conveniently plots as a straight line. This relationship is shown by the heavy straight line in Figure 4.8. This line could be interpreted as the line of minimum possible error, or the lower bound on polygon error area for a given polygon area. With this lower bound in place, it is now seen that the line of average error for the three maps is not consistently linear, but instead there is a strong linear trend in the central portion, parallel to the minimum error line, with divergence away from this lower bound at both extremes in polygon area. This shows that as polygon area increases, the decrease in percent error differs from what would be expected if polygon compactness were constant, as it is for a circle. In other words, the divergence from a trend paralleling the minimum expected error for large polygons is caused by decreased compactness. The greater perimeter relative to area for these very large polygons results in greater error than if average compactness were greater, as for polygons in the 1 to 100 square meters range. This is due to the diversified nature of the mapped data. Both surface features and slopes change over short distances, and any extensive area of a single

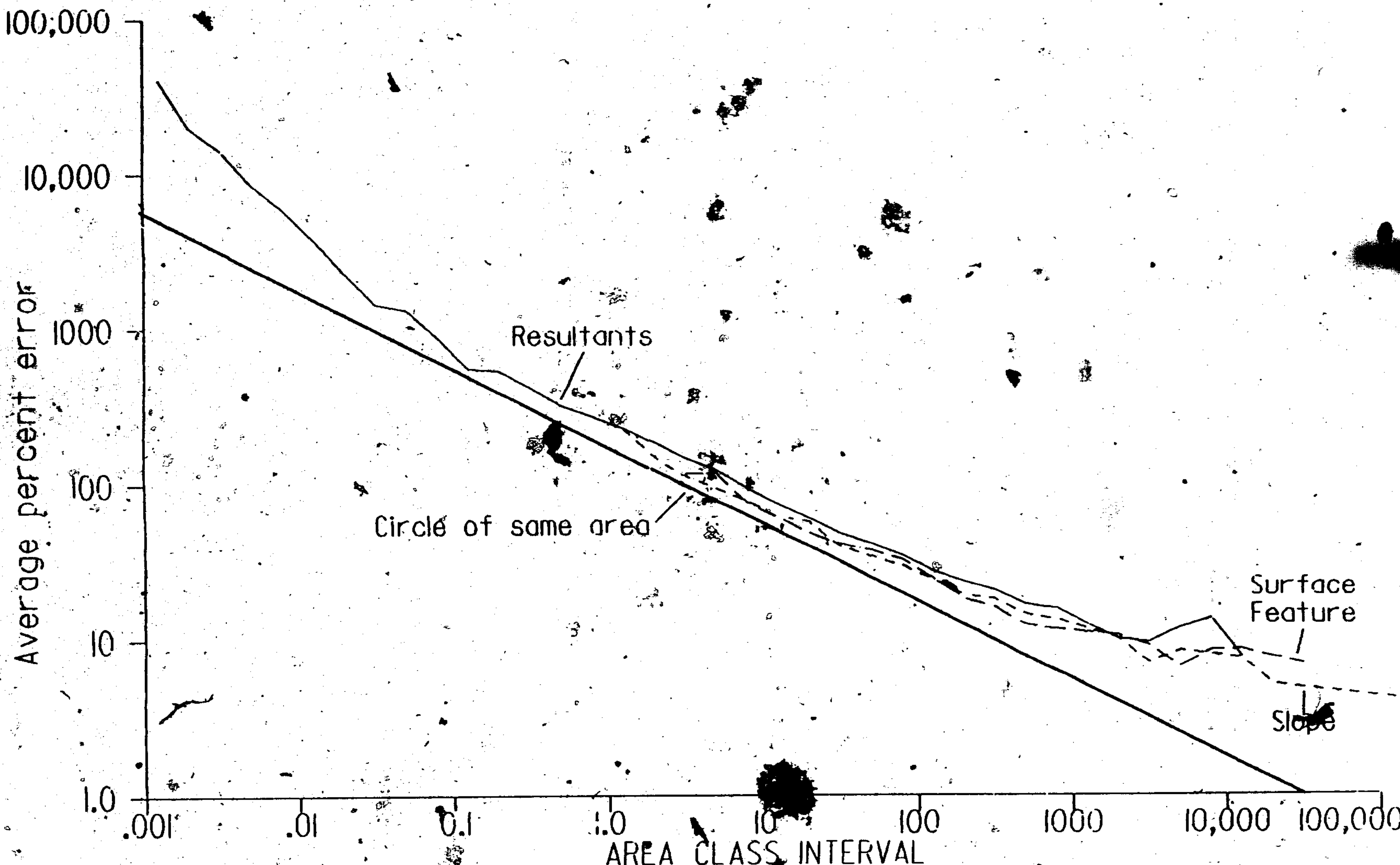


Figure 4.8 Comparison of average band areas of surface feature, slope, and resultant polygons, at epsilon distance of 0.5 meters

class, defined by a single polygon, tends to be elongated and convoluted, and tends to contain holes, both conditions contributing to decreased compactness.

While percent area for all surface features and slopes are approximately equal throughout their area range, the percentages for resultant polygons are consistently greater from one square meter to about 2000 square meters. Just as divergence from minimum error at large areas is caused by decreased compactness, this upward shift in band area in the resultant polygons is caused by decreased compactness. Mean index of compactness calculated [illegible] original polygons is 1.6972, while that of the resultant polyons for the same area range ([illegible] ding those below one square meter) is 1.7463. The process of overlay has caused polygons in the composite or resultant map to have increased perimeter relative to polygons in the original maps of the same area. The increase in band area for a constant polygon area and epsilon distance is illustrated for five example polygons in Figure 4.9.

4.6 Calculation of Total Epsilon Band Area

The band areas can be summed into a single figure representing the amount of variability or potential error within a single classification. To do so, the area of the bands shared by polygons must be accounted for. The total area of the bands is half of the total area measured for all individual polygons, except at the perimeter where the bands are not shared. These totals are summarized in Table 4.10.

For all maps, the increase in band area is directly proportional to the increase in epsilon distance, for the range of epsilon used. It was expected that the effects of angularity would diminish the increase in total band area as epsilon increased, but the effects are slight.

For each value of epsilon, the total band area in the map of resultant polygons is very close to the sum of the areas of the surface feature and slope maps. This is reasonable to expect, as the perimeters of the two original maps are combined into one map, and band areas combined as well. The reduction in total band area in the composite map is caused by reduction in total perimeter from two effects.

Table 4.10 Summary of total epsilon band areas. Areas given in square meters

Category	Epsilon Distance 0.3 Area	0.3 %	0.5 Area	0.5 %	0.7 Area	0.7 %
Surface Feature	20360	7.27	34511	12.16	47966	16.92
Slope	20372	7.31	34504	12.16	48241	17.00
Resultants	38889	13.71	64466	22.72	89878	31.68

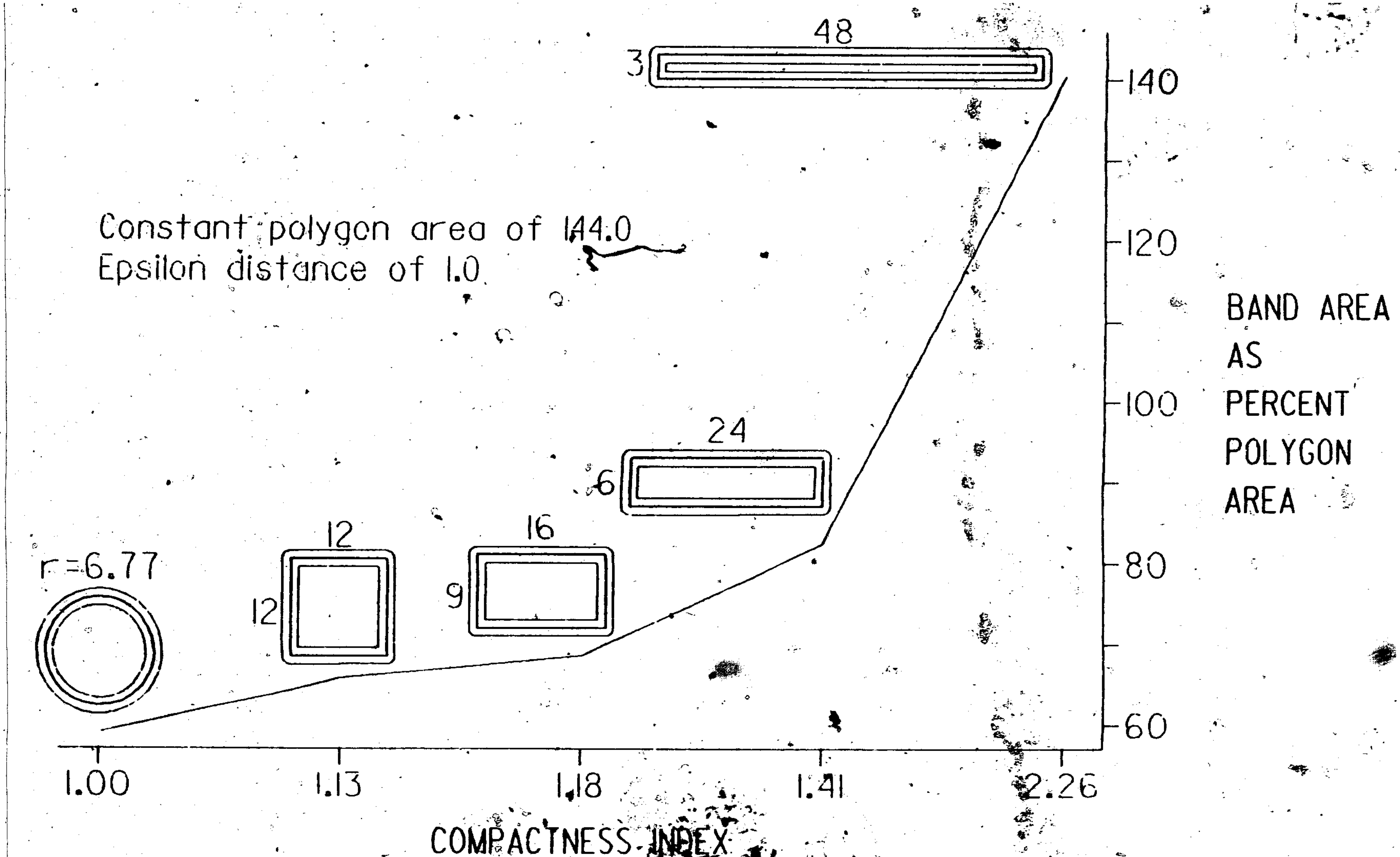


Figure 4.9 Increase in band area for constant polygon area and epsilon distance

Total perimeter in the resultant map is less than the sum of perimeters in the original maps, by 4156 meters. Of this amount, 3014 meters is accounted for by the drainage basin boundary, which is shared by both categories that were overlaid. The further reduction of 1142 meters is caused primarily by reduction in the number of holes in the composite map, which increase perimeter of some of the polygons of the original maps. The influence of polygon perimeter as the major contributor to band area is seen when comparing surface features and slope. Their perimeters are almost equal with surface features slightly higher, and this relationship for band areas is maintained at all epsilon distances.

What is of most concern is total map error as a result of the overlay, as measured by the bands. For the resultant map, total band area reaches as high as 31 percent for epsilon of 0.7 meters. This means that, with an overall variance of 0.7 meters on either side of the digitized version of the true boundary, almost a third of the map lies within the error bands and could be considered erroneous. In this example of using epsilon to measure map error, a practical range of values was chosen primarily to consider the variation in band asymmetry and total band area under different assumptions of map accuracy. The validity of each or any of these measures of epsilon warrants further discussion.

4.7 Simple Perimeter Estimates of Error

MacDougall (1975) defines two kinds of accuracy standards for categorical maps, those of horizontal accuracy and degree of uniformity, or purity, of regions. The lower limit of accuracy, or the upper limit of error, is a function of the sum of the positional errors and the product of the purities of the maps used. Horizontal accuracy for a single map is estimated by

$$H = (h \cdot l)/T$$

where H is the total horizontal error, h is a measure of horizontal error, l is the total length of boundary lines, and T is the total area of the map. This calculation gives an "estimate of the proportion of the map area which is considered to be uncertain or unreliable"

(MacDougall 1975), and is analogous to a simple perimeter estimate of epsilon band area, which is epsilon times perimeter. Substituting the values previously calculated for epsilon and determining perimeter for our maps gives the error values shown in Table 4.11. Values obtained from the epsilon band measurements in Chapter 4 are shown in parentheses for comparison. The error values for the simple perimeter estimates are consistently greater, although as a proportion of total map area they are within one percent for all maps. The epsilon band measurements obtained by the program EBAND include adjustments for overcount and undercount at intersegment angles. The small differences between the modified estimates of EBAND and the simple perimeter reflect the tendency for angles in polygon boundaries to cancel each other out, or convexities and concavities along the polygon boundaries are nearly equal. The purpose of applying the modified estimate was to get a better estimate of error than is possible using simple perimeter. This comparison shows that, for the range of epsilon used, the simple perimeter estimate is adequate.

Table 4.11 Epsilon band areas calculated by simple perimeter estimates. Values in parentheses are measurements obtained with program EBAND. Areas given in square meters

Category	Epsilon Distance 0.3 Area	0.3 %	0.5 Area	0.5 %	0.7 Area	0.7 %
Surface Feature	21591 (20360)	7.61 (7.27)	35985 (34511)	12.68 (12.16)	50379 (47966)	17.76 (16.92)
[illegible]	21690 (20372)	7.64 (7.31)	36150 (34504)	12.74 (12.16)	50610 (48241)	17.84 (17.00)
Resultant	39222 (38889)	13.86 (13.71)	65536 (64466)	23.10 (22.72)	91750 (89878)	32.34 (31.68)

l = total length of boundary lines

l(surface feature) = 35985 meters
l(slope) = 36150 meters
l(resultant) = 65536 meters

4.8 Summary of Overlay Results and Error Measurements

It is found that the overlay of the two categorical maps has broken the original classes down into new polygons which are smaller and less compact, and has generated a significant number of polygons which are smaller than the smallest area resolvable at the stated scale from direct field observation. The overlay of 369 surface feature polygons with 476 slope polygons resulted in a composite map containing 2723 polygons, 352 or 13 percent of which are smaller than the smallest original polygon.

The distribution of polygons by area is such that the majority of polygons account for a minority of the area. This characteristic is more pronounced in the resultant polygons because of high frequencies of very small polygons. In the composite map, mean polygon size is 101 square meters, less than one sixth of the mean area for all original polygons, of 672 square meters. Total perimeter within the map area, aside from that shared at the drainage basin boundary, decreased by 1142 meters.

Application of epsilon band measurements to calculate error area shows:

1. In general, error area as a proportion of a polygon class decreases as the class area increases.
2. The average error area as a proportion of the area of a single polygon decreases as polygon area increases, but for larger polygons the degree of compactness decreases, resulting in higher average error than would be expected.
3. For individual polygons and for whole classes in the composite map, error area as a proportion of polygon or class area has increased over that determined for polygons and classes of equal area on the original maps.
4. Average compactness of resultant polygons is less than that of original polygons, though magnitude of the difference is di[illegible] relative measure.
5. Total map area calculated to be error-prone due to variability in boundary placement is 13.7, 22.7, and 31.7 percent at epsilon distances of 0.3, 0.5, and 0.7 meters, respectively.

Error in the composite, overlaid map has increased significantly in the combination of the two polygon coverages, but is less th[illegible]e sum of the error of the two separate maps, primarily because of a reduction in tot[illegible]r for the composite map.

5. Discussion

5.1 Introduction

The main result of the overlay statistics and error measurements of the previous chapter is the demonstration of the use of the epsilon model to produce a statement of reliability in the overlay product. The following discussion first summarizes the pertinent relationships between the error measurements and the geometric properties of the digital map. Next the controlling factors of error are related to the processes involved in defining a categorical coverage of the study area. Finally, the epsilon distance model is assessed in the contexts of information quality, its relationship to spurious polygons, and overall strengths and limitations.

5.2 Relationships Found through Application of the Epsilon Model

5.2.1 Predictability of Results

In interpreting the results of the error measurements, relationships were sought that were explainable through the map, as well as ways to interpret the results through the error model. The relationship between such parameters as band area, polygon area, and polygon perimeter are predictable. For a given epsilon, the area of a band of error generated around a polygon is dependent on its perimeter. Likewise, for a given area, an increase in perimeter expressed as decreased compactness must result in a larger band and hence greater error proportional to the polygon area. Conversely, less perimeter for a given area results in less error, but there is a definite lower limit on error for a given polygon area/epsilon distance combination, as shown in Figure 4.7.

5.2.2 The Consequences of a Geometric Form of Error

The measurements of total epsilon band area show that, for the maps overlaid in this research, the error in the composite map approaches, but is slightly less than the sum of the error of the individual maps prior to overlay. The reduction in perimeter at the basin boundary and within the composite map accounts for the reduction in band area in the composite map, and likewise in the amount of total error.

Epsilon distance is applied to each point defining the individual line segments, and so error band area is implicitly tied to perimeter in the geometric construction of the band around each segment. If more line segments are introduced to the map area, increasing the total perimeter, it is an unavoidable consequence that the band area will increase.

5.2.3 Polygon Size and Reliability

Other research dealing with reliability and error measurement in overlay implies the effect of perimeter in subdividing the map area, before and after overlay, but does not deal with it specifically.

> "Common sense would indicate that broadly the reliability of description of derived map [polygons] from non-homogeneously described initial map [polygons] would decrease as the degree of heterogeneity in the initial description increased and as the number of subdivisions of the initial [polygons] caused by overlay increased (and hence the size of the derived [polygons] decreased)."(McAlpine and Cook 1971)

This statement implies that accuracy (or reliability) in the overlay product is affected by the number of subdivisions, or heterogeneity, of the original maps. Greater heterogeneity requires further subdivision, or more polygons, and an attendant increase in perimeter within the map. The intersection of two maps through overlay further increases heterogeneity by requiring unique combinations of the two classification systems to be described, and their descriptions are a function of the combined perimeters. What McAlpine and Cook are saying is that reliability of description will decrease as polygon size decreases. Tomlinson (1977) and Lam (1977) also refer to this as a consequence of overlay, the latter adding that the nature of inhomogeneities and the manner in which the map is subdivided should be examined. This general assumption that reliability of description decreases with polygon size is demonstrated geometrically by generating a band of error around its perimeter.

5.3 Influences on Perimeter and Epsilon

5.3.1 Classification and Scale

The extent of perimeter within the map area is a consequence of subdividing the map into contiguous, well-defined (on the map, at least), categorical polygons. The use of a classification scheme implies that the attributes to be mapped are distinct, and the partitioning of the area implies homogeneity within each polygon. The distinction between two classes must be perceived before a boundary can be drawn. Within the map area, an attempt is made to separate these two classes, to define their local limits. The ability to define a "sharp" boundary between two "distinct" classes is limited by the resolution of the means of recording this perceived boundary. In our case this is the ability of the field surveyor to distinguish one patch of ground from another on the manuscript map. The scale of the initial recording and the means by which it is done (manually drawing the line, in our case), will determine the smallest area that can be defined. Working upwards from this lower limit of resolution, transition zones between classes must be dealt with. On top of this, the diversity of the classes, in terms of both number and spatial distribution, will affect how and where polygon boundaries are placed.

5.3.2 Influence of Landscape on Mapping Accuracy

To relate the spatial characteristics of the study area to their mapped representation and the potential error therein, we draw on data provided by the error measurements, and on examination of the maps and observations made at the study area.[1] The main considerations are heterogeneity of classes of surface features and slopes, the classification system used in

[1]Field observations by the author, April 1986.

mapping them, and the nature of the perceived boundaries between classes. Heterogeneity of classes contributes to the amount of subdivision within the map area, as described earlier. The accuracy of definition of boundaries is a key factor in determining the proper epsilon distance for the mapped data.

The badlands features of the study area are characterized by their diversity. Changes in lithological units are frequent and abrupt, vertically as well as horizontally (Campbell and Honsaker 1982). The sharp break from one type of lithology to another results in distinct, relatively well-defined boundaries. However, these boundaries are also very complex and sinuous because of the high degree of lateral interspersion of classes. The complex boundaries introduce a high degree of subdivision of the map area not only from heterogeneity of classes but also from the need to distinguish between two intermingling classes. The result of such sinuous, convoluted boundaries is the difficulty in defining boundaries at the field mapping level, resulting in initial positional error in the manuscript map.

Slope boundaries are also complex, and furthermore they are not as clearly definable as those for surface features. The breakdown of slopes from 0 degrees to 90 degrees into four classes is essentially the division of a continuous variable into four class intervals. During field mapping, the surveyor is attempting to define areas of varying slope angles of a range that is within a preset class interval. In other words, he is aggregating slopes prior to plotting the boundaries on the manuscript. This process would seem to lend itself to generalization effects, as the surveyor is often using his own judgement to determine where one trend in a continuously varying slope ends and another begins. Judgemental error would be greatest in the cases where gentler slopes of the 0-4 degree and 4-10 degree classes meet.

Boundaries between vegetation and other surface features are usually not distinct at all, but only a fuzzy zone of transition. The boundaries are sharp in a few places where the break in vegetation occurs together with a sharp break in slope, as with flat-topped, vegetation-capped features with near-vertical sides which stand out as "islands" above surrounding areas.

5.3.3 A Realistic Measure of Epsilon

In summary, the complexity of boundaries between classes of both surface features and slopes, the difficulty in distinguishing between slope angles, and the persistent uncertainty of vegetation boundaries all serve to reduce the precision with which these boundaries can be mapped. The precision of field plotting was not included as an error effect in the calculation of epsilon distances, as it is difficult or even impossible to ascertain this without documentation for the field survey measurements.

If this effect could be measured and integrated into the epsilon distance calculations, undoubtedly they would be larger. The liberal estimation of digitizing error used in the third and largest estimate of epsilon distance would absorb some of the imprecision of field plotting, but not all of it. On the basis of the above observations and arguements, it is proposed that the epsilon distance of 0.7 meters is a conservative but reasonable estimate of error in boundary variability for the surface feature and slope maps.

5.4 Assessment of the Epsilon Model

5.4.1 The Epsilon Model as a Component of Digital Cartographic Data Quality

With an epsilon distance of 0.7 meters, the horizontal error measured in the original maps is 17 percent each, and in the the composite, overlaid map stands at 31.7 percent, nearly a third of the total area of the drainage basin. This is a clear illustration of the potential for large errors in products of polygon overlay, due to errors accumulated in the digital source data. The epsilon model provides a means to advise the users of such products in the form of a statement of reliability.

Widespread concern for accuracy and reliability in digital cartographic data is a relatively recent phenomenon. Gordon (1979) observed that "there is a noticable lack of concern for data accuracy and data documentation." Based on a survey of private and public organizations in the Pacific Northwest U. S. with functioning spatial data handling systems, it was found that "less than 15 percent of the respondents were aware of the precision of their data, and most do not mention descriptions of the basic characteristics necessary to assess its utility."

Accounting for the accuracy and reliability of cartographic data is encompassed in recent efforts to develop standards for digital cartographic databases. The National Committee for Digital Cartographic Data Standards (NCDCDS) was formed in 1982, and has recently proposed interim standards which include information on "quality" of cartographic data. This information is in the form of a quality report which is broken down into five components: lineage, positional accuracy, attribute accuracy, logical consistency, and completeness. Methods of measuring positional accuracy include, among others, deductive estimates "based on knowledge of errors in each production step"(NCDCDS 1985). In this research we have included an example of defining these steps and the related errors. Deductive estimates of error are considered part of testing the quality of cartographic information in which a quality statement for a particular product is made by propogating the separate error effects (Chrisman 1985, *in* NCDCDS 1985, p. 114), as was done in this research to arrive at a measure of epsilon. The epsilon model thus plays a role in fulfilling needs for information quality and has, for this research, provided a realistic means to assess at least one aspect of the quality of the mapped data that was used.

5.4.2 Epsilon and Spurious Polygons

It was noted in Chapter 2 that when epsilon bands completely overlap inside the polygon, the lower bound on area is zero. For such polygon, all points defining the boundary must be within twice epsilon of at least one other point. The range of error measured in this case is so great that, at the lower limit, it should not occur. This condition is analogous to the presence of spurious polygons, i.e., questionable polygons as a result of cartographic error.

Spurious polygons are regarded as "noise" in a digital cartographic database (Chan 1982) and attempts have been made to build procedures for their removal. However, lacking is a rigid set of criteria to identify spurious polygons. Their main characteristics are small area and/or sliver-like shape. These and other properies are advocated (Goodchild 1978, 1980, 1981) as criteria for identification. More recently, Goodchild (1985, personal communication) has experimented with identification on the basis of statistical distributions of polygon areas. Still, no strict rules have been developed. Chan (1982) has demonstrated the use of an algorithm to remove spurious polygons, but in the dataset on which this was performed the spurious polygons were identified visually before implementing the procedure.

Epsilon error distance has been used as a single criterion for the elimination of spurious polygons. The polygon overlay program WHIRLPOOL (Dougenik 1979) incorporates an error distance (epsilon) which is used to merge points if they are within the error distance of each other, eliminating sliver-shaped areas in the resultant polygons that are narrower than the error distance.

The difference in the operation of WHIRLPOOL and the analogy drawn between spurious polygons and polygons with zero lower bound on area is that in WHIRLPOOL the polygon must have all points within one epsilon of at least another point instead of two times epsilon. This makes WHIRLPOOL's definition of a spurious polygon is a more stringent case than the case of zero lower bound, as was suggested here.

5.4.3 Strengths and Limitations

The strength of the epsilon model is in its sound basis. It takes the geometric form of the final cartographic product and assigns a value of uncertainty to this well-defined boundary which is applicable to all boundaries, and therefore to the whole map. This is important because, even though epsilon can vary from map to map or even within a single map, it facilitates measurement of variability or error for all its subdivisions. Other methods of assessing map accuracy apply probability theory to a random sample of points within the map. Depending on the coverage involved, and the number of points sampled and subsequently compared with ground truth, these methods may suffer from over- or underestimation (Chrisman 1982a). These procedures are basically aimed at determining percent classified correct, or the reliability of attribute classification. This does not take individual classes into consideration, and we have illustrated that variability differs from class to class due to effects of polygon size, distribution of total class area, and polygon compactness. The epsilon model gives real estimates of error at all levels, by polygon, by class, or for an entire category, allowing examination of the structure of error within the map. The variability of area measurements are presented as upper and lower bounds on area. These bounded area estimates parallel basic statistical methods, such as standard error estimates of means or confidence limits on proportions. The concept of a geometric definition of variability is recurrent in other works as well. It has been suggested that error bands of cartographic lines could be graphically incorporated into the map as an indicator of their accuracy (MacDougall 1975, Yoeli 1984).

All of the limitations to the use of the epsilon model for measuring cartographic error in this application are due to the manner in which it was implemented. The epsilon model should ultimately construct a proximity network around lines, i.e., the neighborhood of all points within epsilon of the line and closer to that line than any other line (Chrisman 1982a). The accurate construction of this network requires the splitting of bands that overlap on nonadjacent segments, which was not achieved in the program EBAND. As a result, band areas are overestimated by an unknown amount. The GPPU software system is capable of graphically creating the full version of the proximity network, but even if successful, this processor was not intended to calculate areas within those bands. Instead, a rather complicated and time-consuming method would be required to generate the graphics, handle them as polygons, then successively subtract areas to arrive at band areas, not to mention the organizational gymnastics necessary to accomodate nested polygons.

6. Conclusions

6.1 Summary and Conclusions

The goal of this thesis was to examine the effects of cartographic error with an automated spatial analysis procedure, polygon overlay. An applied approach was taken to the [illegible]ing a set of real-world, albeit atypical, mapped data. The epsilon distance values used to estimate error in the maps were based on deductive measures of known positional error accumulated from the field survey to the digitized version of the maps. A range of values were generated for analytical purposes, and the value of 0.7 meters has been argued as a reasonable measure for the dataset.

Examination of error in the maps of surface features and slopes before and after overlay illuminates a fundamental relationship between the error model that was used and the vector polygon data organization. Planimetric error, as measured by variability in vector-encoded boundaries, is a function of the degree of subdivision, or perimeter, within the map area, due to the intrinsic geometric properties of the error model. This relationship is as simple and straightforward as the equation for "horizontal accuracy," or positional error, given earlier by MacDougall in Chapter 4. An increase in perimeter results in a corresponding increase in error. It follows, then, that by combining two sets of polygons through overlay, the perimeters are also combined, and the error in the resultant, composite map increases by the amount of increase in resultant perimeter. The epsilon model reduces the error estimate by adjusting inequalities in band area created at angles in the line segments. However, our results show that, in the overlaid map, these adjustments reflect a decrease in actual error of less than one percent, which could not be considered significant.

Other observations made in the error measurement results are a direct consequence of the role of perimeter in error measurement. Polygon compactness decreases in the resultant map over the original map because the total area is fixed, while perimeter increases. Proportions of error for a given polygon area are defined by its perimeter, such that for a fixed polygon area, error increases as the polygon becomes less compact, or increases its perimeter.

What this research best illustrates is a means of assisting users of digital cartographic data in determining the limitations of usefulness of such data for analytical purposes. Positional accuracy is an important component of information quality, with which cartographers are becoming increasingly concerned.

6.2 Future Perspectives

Two by-products of the application of overlay and error measurement have potential for other uses: the data produced from the polygon overlay and the program created to measure epsilon bands. In spite of the large amount of error found in the resultant polygon set, the raw overlay results could be useful for geomorphological purposes. The cross-classification of surface features and slopes showed distinct trends in intersection characteristics which may or may not be useful for integration with studies of erosion rates and processes. In this we have an example of the use of information regarding data reliability. The geomorphologist knows the reliability of the composite map and can now judge whether any or all of the data are suitable for his purposes.

The implementation of a program to calculate epsilon bands points to many research possibilities. We have already seen that epsilon error distances can be used in a polygon overlay program to avoid generation of spurious polygons. This is essentially a spatial filtering process; lines which are within a set tolerance of each other are considered identical and are generalized so that they are coincident. Aside from this function within polygon overlay, the epsilon tolerance function can be used specifically for line generalization (Chrisman 1982a, 1983), removing detail and reducing numbers of points based on a clustering analysis of points within epsilon of each other. The basic concept of moveable points within a probability region is the foundation upon which many applications can be built.

References

Allen, D.N. (1976). "Polygonal Processinig of Geographic Resource Information". in *Proceedings, Resource Data Management Symposium*, pp. 105-112. American Society of Foresters.

Baxter, R. (1980). "A Computer File and Data Structure for Manipulating Digital Outlines of Polygons." *Geo-Processing* 1(3): 243-255.

Bie, S.W. (1980). "Computer-assisted Soil Mapping. "In *Progress in Contemporary Cartography*, ed. D.R.F. Taylor. Vol. I: *The Computer in Contemporary Cartography*, pp. 123-149. New York: John Wiley & Sons.

Blumenstock, D.I. (1953). "The Reliability Factor in the Drawing of Isarithms." *Annals of the Association of American Geographers*, 43: 289-304.

Burton, W. (1977) "Representation of Many-Sided Polygons and Polygonal Lines for Rapid Processing." *Communications of the ACM*, 20(3): 166-177.

Campbell, I.A., and J. L. Honsaker (1981) "Variability in Badlands Erosion; Problems of Scale and Threshold Identification." In *Space and Time in Geomorphology, Binghamton Symposia in Geomorphology, International Series 12*, pp. 59-79.

Canadian Council on Surveying and Mapping (1984). *National Standards for the Exchange of Digital Cartographic Data.* Vol I: *Data Classification, Quality Evaluation and EDP File Format.* Second Draft Report, July 1984. Ottawa: Topographical Survey Division, Surveys and Mapping Branch, Energy, Mines, and Resources Canada.

Chan, K. (1982). "Digital Stereo Photomapping and Surface Modelling." Unpublished M. A. Thesis, University of Western Ontario. Appendix D: "A Procedure for the Identification and Removal of Spurious Polygons," pp. 163-178.

Chrisman, N.R. (1982a). "Methods of Spatial Analysis Based on Error in Categorical Maps." Unpublished Ph.D. thesis, University of Bristol.

Chrisman, N.R. (1982b). "A Theory of Cartographic Error and its Measurement in Digital Data Bases." In *Auto-Carto 5, Fifth International Symposium on Computer-Assisted Cartography: Environmental Assessment and Resource Management*, ed. J. Foreman, pp. 159-168. Falls Church, Va.: ASP and ACSM.

Chrisman, N. R. (1983). "Epsilon Filtering: A Technique for Automated Scale Changing." In *Technical Papers, 43rd Annual Meeting, ACSM*, pp. 322-331. Falls Church, Va.: ACSM.

Cook, B.G. (1983). "Geographic Overlay and Data Reliability." In *Proceedings, United States/Australia Workshop on Design and Implementation of Computer-Based Geographic Information Systems*, eds. D. Peuquet and J. O'Callaghan, pp. 64-69. Amherst, N.Y.: IGU Commission on Geographical Data Sensing and Processing.

Dangermond, J. (1983) "A Classification of Software Components Commonly Used in Geographic Information Systems." In *Proceedings, United States/Australia Workshop on Design and Implementation of Computer-Based Geographic Information Systems*, eds. D.

Peuquet and J. O'Callaghan, pp. 70-91. Amherst, N.Y.: IGU Commission on Geographical Data Sensing and Processing.

Dougenik, J. (1979) "WHIRLPOOL: A Geometric Processor for Polygon Coverage Data." In *Auto-Carto IV, Proceedings of the International Symposium on Cartography and Computing*. Vol II: *Applications in Health and Environment*, pp. 304-311. Falls Church, Va.: ACSM and ASP.

Douglas, D. (1972) "It Makes Me so CROSS." Reprinted from unpublished manuscript, in *Basic Readings in Geographic Information Systems*, eds. D. Marble *et al*, pp. 4-3 to 4-7. SPAD Systems Ltd., 1984.

Dutton, G. (1984). "Truth and its Consequences in Digital Cartography." In *Technical Papers, 44th Annual Meeting, ACSM*. Falls Church, Va: ACSM.

Goodchild, M.F. (1978). "Statistical Aspects of the Polygon Overlay Problem." In *Harvard Papers on Geographic Information Systems*, ed. G.H. Dutton. Vol. VI: *Spatial Algorithms: Efficiency in Theory and Practice*. Cambridge, Mass: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Goodchild, M.F. (1980a). "The Effects of Generalization in Geographical Data Encoding." In *Map Data Processing*, eds. H. Freeman and G.G. Pieroni, pp. 191-205. New York: Academic Press, Inc.

Goodchild, M.F. (1981). "Accuracy and Spatial Resolution: Critical Dimensions for Geoprocessing." In *Computer Assisted Cartography and Geographic Information Processing: Hope and Realism, Proc. Conference, Calgary, 1981*, eds. D.H. Douglas and A.R. Boyle, pp. 87-90. Ottawa: Canadian Cartographic Association.

Gordon, K. E. (1979). "Computer-Assisted Environmental Data Handling: Issues and Implications." In *Auto-Carto IV, Proceedings of the International Symposium on Cartography and Computing*. Vol II: *Applications in Health and Environment*, pp. 259-266. Falls Church, Va.: ACSM and ASP.

Guevara, J.A. (1983). "A Framework for the Analysis of Geographic Information System Procedures: The Polygon Overlay Problem, Computational Complexity and Polyline Intersection." Unpublished Ph.D. dissertation, State University of New York at Buffalo.

Harty, K. (1984). "The Geomorphic Role of Snow in a Badland Watershed." Unpublished M.Sc. thesis, University of Alberta.

Intergraph Corporation (1985a). *DMRS Control Block (CB) Users Guide (8.8)*, no. DIXD3360.

Intergraph Corporation (1985b). *DMRS Data Definition Language (DDL) Users Guide (8.8)*, no. DIXD3310.

Intergraph Corporation (1985c). *Graphic Polygon Processing Utilities (GPPU) Users Guide (8.8)*, no. DIXD4300.

Intergraph Corporation (1985d). *IGDS (8.7) Application Software Interface Document*, no. DIXD2740. Appendix C: *IGDS Element Formats*.

Jenks, G.F. (1981). "Lines, Computers, and Human Frailties." *Annals of the Assoc. of*

American Geographers, 71(1): 1-10.

Lam, N. (1977). "Polygon Overlay: An Examination of an Algorithm and Related Problems." Unpublished M.Sc. thesis, University of Western Ontario.

MacDougall, E. B. (1975). "The Accuracy of Map Overlays." *Landscape Planning*, 2(1): 23-30.

McAlpine, J.R., and B.G. Cook (1971). "Data Reliability from Map Overlay." In *Proceedings, Australian and New Zealand Association for the Advancement of Science, 43rd Congress, Brisbane.* ANZAAS.

McHarg, I.L. (1969). *Design With Nature*. Garden City, N.Y.: The Natural History Press. 197 pp.

Miller, S.W. (1980). "A Compact Raster Format for Handling Spatial Data." In *Proceedings of the Fall Technical Meeting of the ACSM and ASP*, pp. CD4A1 to CD4A18. Falls Church, Va.: ACSM/ASP.

Mitchell, W. B. (1978) "Geographic Information Systems in a Federal Government Environment." In *Harvard Papers on Geographic Information Systems*, ed. G. H. Dutton. Vol II: *Geographic Information Systems: Surveys, Overviews, and Criteria*. Cambridge, Mass.: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Morehouse, S., and G. Dutton (1979). "Extraction of Polygonal Information from Gridded Data." In *Proceedings, Auto-Carto IV*, pp. 320-327. Falls Church, Va.: ACSM/ASP.

Muehrcke, P.C. (1978). *Map Use: Reading, Analysis, and Interpretation*. Madison, Wisc.: JP Publications, Inc.

Müller, J.-C. (1977). "Map Gridding and Cartographic Errors/A Recurrent Argument." *The Canadian Cartographer*, 14(2):152-167.

Nagy, G., and S. Wagle (1979). "Geographic Data Processing." *Computing Surveys*, 11(2): 139-181.

National Committee for Digital Cartographic Data Standards (1985). *Issues in Digital Cartographic Data Standards*. Report #6: *Digital Cartographic Data Standards: An Interim Proposed Standard*, ed. H. Moellering. Columbus, Oh.: ACSM/NCDCDS. 163 pp.

Peucker, T.K. (1976). "A Theory of the Cartographic Line." *International Yearbook of Cartography*, 16: 134-143.

Peucker, T.K. (1980). "Literature for Geographic Information Systems." In *Harvard Library of Computer Graphics, 1980 Mapping Collection*, ed. P.A. Moore. Vol II: *Urban, Regional, and State Government Applications of Computer Mapping*. Cambridge, Mass.: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Peucker, T.K., and N. Chrisman (1975). "Cartographic Data Structures." *The American Cartographer*, 2(1): 55-69.

Peuquet, D.J. (1978). "Raster Data Handling in Geographic Information systems." In

Harvard Papers on Geographic Information Systems, ed. G.H. Dutton. Vol. II: *Geographic Information Systems: Surveys, Overviews, and Criteria.* Cambridge, Mass: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Peuquet, D.J. (1984). "A Conceptual Framework and Comparison of Spatial Data Models." *Cartographica*, 21 (4): 66-113.

Rhind, D.W. (1973). "Generalisation and Realism within Automated Cartographic Systems." *The Canadian Cartogrpher*,. 10(1): 51-62.

Schlesinger, J., B. Ripple, and T.R. Loveland (1979). "Land Capability Studies of the South Dakota Automated Geographic Information System (AGIS)." In *Harvard Library of Computer Graphics, 1979 Mapping Collection.* Vol. IV: *Computer Mapping in Natural Resources and the Environment.* Cambridge, Mass: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Semwal, S.K. (1984). "Data Structures for Spatial Information." Unpublished M.Sc. thesis, University of Alberta.

Shamos, M.I. (1975). "Geometric Complexity." In *Proceedings of Seventh Annual ACM Symposium on Theory of Computing*, pp. 224-233. ACM.

Sinton, D.G. (1978). "The inherent Structure of Information as a Constraint to Analysis: Mapped Thematic Data as a Case Study." In *Harvard Papers on Geographic Information Systems*, ed. G.H. Dutton. Vol. VII: *Spatial Semantics: Understanding and Interacting with Mapped Data.* Cambridge, Mass.: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Sinton, D., *et al* (1972). "Spatial Data Manipulation Techniques." In *Geographical Data Handling*, ed. R.F. Tomlinson. Sympsoium Edition, Vol. II, Ch. 11. Ottawa: IGU Commission on Geogrpahical Data Sensing and Processing.

Steinitz, C., P. Parker, and L. Jordan (1976). "Hand-Drawn Overlays: Their History and Prospective Uses." *Landscape Architecture*, 66(5): 444-455.

Tilove, R. B. (1981). "Line/Polygon Classification: A Study of the Complexity of Geometric Computation." *IEEE Computer Graphics and Applications*, 1(2):75-86.

Tomlin, C.D., and J.K. Berry (1979). "A Mathematical Structure for Cartographic Modeling in Environmental Analysis." *Proceedings of the ACSM 39th Annual Meeting, Washington, D.C.* Falls Church, Va.: ACSM.

Tomlinson, R.F. (1977). "Geo-Information Systems and the Use of Computers in Handling Land Use Information." In *Background Materials on Geographic Information Systems and Computer Cartography.* Amherst, N.Y.: State University of New York at Buffalo.

Tomlinson, R.F., *et al* (1976). *Computer Handling of Geographical Data.* Paris: The UNESCO Press.

Weiler, K. (1980). "Polygon Comparison Using a Graph Representation." *Computer Graphics*, 14(3): 10-18.

White, D. (1978). "A Design for Polygon Overlay." In *Harvard Papers on Geographic Information Systems*, ed. G.H. Dutton. Vol. VI: *Spatial Algorithms: Efficiency in Theory* *and Practice.* Cambridge, Mass.: Harvard University, Laboratory for Computer Graphics and Spatial Analysis.

Williams, R. (1971). "A Survey of Data Structures for Computer Graphics Systems." *Computing Surveys*, 3(1): 1-21.

Yoeli, P. (1984). "Error-Bands of Topographical Contours with Computer and Plotter (Program KOPPE)." *Geo-Processing*, 2: 287-297.

Personal Communications

M. F. Goodchild, Professor
Department of Geography
University of Western Ontario
London, Ontario, Canada

A. Teng
Senior Technical Manager
Land Use Management Applications
Intergraph Corporation
One Madison Industrial Park
Huntsville, Alabama, U. S. A.

Appendix 1: Documentation and Source Code for Program MERGER

Program MERGER

NOTE: This documentation describes the restrictions, input requirements, and output characteristics of the program MERGER. No instructions on how to run the program are given, since this will vary from system to system. If this program is to be used on the VAX/Intergraph system in the Department of Geography, U. of Alberta, please see the system manager for specific instructions on how to run this program.

1. **Language:** FORTRAN

2. **Purpose:** Reads X and Y values of polygon vertices from a sorted list. X and Y values for all vertices that are within a specified radial tolerance distance of each other are averaged and the new values are output as the new location of the vertices.

3. **Restrictions:**
   - Maximum number of vertices is 15000.
   - Tolerance distance is set to 42949. To change this, edit the value of variable TOL in line 94 of the main program.

4. **Input requirements:**
   - X and Y values must be type INTEGER.
   - The vertices defining a single polygon must be listed only once in the input, i.e., the first vertex of the polygon must not be repeated as the last.
   - Input records must be sorted on X, with the smallest X at the beginning of the input file.
   - Input list format is (I2,2I5,2I11). Each field in an input file record contains the following:
     - I2 Level number (optional)
     - I5 Polygon occurrence number
     - I5 Polygon vertex sequence number
     - I11 X value (right-justified in data field)
     - I11 Y value (right-justified in data field)
   - The input filename must be MERGER.DAT.

5. **Output:** The output filename is MERGER.OUT. The output records are in the same format as the input records, with new X and Y values for vertices that have been merged.

```
C     ************************************************************
C
C     MAIN PROGRAM UNIT
C
C     ************************************************************
C
C     VARIABLES           DESCRIPTION
C
C     _________           ___________
C
C     N                   PARAMETER FOR ALLOTTING ARRAY SPACE
C     XX                  ARRAY OF X VALUES READ FROM PROGRAM LSTDGN
C     YY                  ARRAY OF Y VALUES READ FROM PROGRAM LSTDGN
C     INF                 GRAPHIC ELEMENT INFORMATION PASSED WITH X & Y
C     USED                LOGICAL FLAG TO MARK USE OF X & Y VALUES
C     TOL                 TOLERANCE DISTANCE
C     COUNT               DETERMINES NUMBER OF LOCATIONS IN XX AND YY
C     I                   COUNTER FOR MAIN LOOP
C     STARTX              FIRST UNUSED X IN XX
C     STARTY              FIRST UNUSED Y IN YY
C     BNDX                ARRAY OF X WITHIN TOL OF STARTX
C     BNDY                ARRAY OF Y FOR BNDX
C     BNDINF              INFO PASSED FOR POINT IN BAND
C     BNDUSE              LOCATION OF BAND POINTS IN INPUT LIST
C     NUMBND              COUNTER FOR BAND ARRAY
C     SQRX                ARRAY OF X FOR SQRY
C     SQRY                ARRAY OF Y WITHIN TOL OF STARTY
C     SQRINF              INFO PASSED FOR POINT IN SQUARE
C     SQRUSE              LOCATION OF SQUARE POINTS IN INPUT LIST
C     NUMSQR              COUNTER FOR SQUARE ARRAY
C     RADX                ARRAY OF X WITHIN TOL RADIUS OF STARTX,Y
C     RADY                ARRAY OF Y WITHIN TOL RADIUS OF STARTX,Y
C     RADINF              INFO PASSED FOR POINT IN RADIUS
C     RADUSE              LOCATION OF RADIUS POINTS IN INPUT LIST
C     NUMRAD              COUNTER FOR RADIUS ARRAY
C
C     ............................................................
C
C     PROGRAM NAME       : MERGER
C
C     LANGUAGE           : FORTRAN
C     SYSTEM             : VAX-11/730 VMS 4.1
C     AUTHOR             : JAMES W.F. SMITH
C     UIC                : [262,10]
C     LAST UPDATED       : NOVEMBER 1985
C
C     PURPOSE            : READS X AND Y VALUES FROM A SORTED LIST
C                          AND MERGES THOSE THAT ARE WITHIN A
C                          SPECIFIED TOLERANCE OF EACH OTHER.
C
C     ............................................................
C
C
      PROGRAM MERGER
C
      PARAMETER N=15000
C
      IMPLICIT INTEGER (A-Z)
```

```
C
      LOGICAL*1 USED(N)
      CHARACTER*12 INF(N),BNDINF(N),SQRINF(50),RADINF(50)
C
C     ------------------------------------------------------------
C
      COMMON /AREA1/ XX(N),YY(N),BNDX(N),BNDY(N),BNDUSE(50),
     +SQRX(50),SQRY(50),SQRUSE(50),RADX(50),RADY(50),RADUSE(50),
     +NUM,I,STARTX,STARTY,TOL,NUMBND,NUMSQR,NUMRAD
C
      COMMON /FLAGS/ USED
      COMMON /INFO/ INF,BNDINF,SQRINF,RADINF
C
C
C     FILL LOGICAL ARRAY WITH FALSE VALUES
C     ------------------------------------
C
      DATA USED/N*.FALSE./
C
C     OPEN INPUT AND OUTPUT FILES
C     ---------------------------
C
      OPEN(UNIT=4,FILE='MERGER.DAT',STATUS='OLD')
      OPEN(UNIT=6,FILE='MERGER.OUT',STATUS='NEW')
C
C
      COUNT=0
C
      DO 10 I=1,N
         READ(4,100,END=10)INF(I),XX(I),YY(I)
         COUNT=COUNT+1
   10 CONTINUE
C
C     SET TOLERANCE IN DESIGN FILE UNITS
C     ----------------------------------
C
      TOL=42949
C
      NUM=COUNT
C
C
C     MAIN LOOP; SET STARTX,STARTY, BEGIN BAND
C     ----------------------------------------
C
      DO 20 I=1,NUM
         IF (USED(I) .EQ. .FALSE.) THEN
            STARTX=XX(I)
            STARTY=YY(I)
            NUMRAD=1
            RADINF(NUMRAD)=INF(I)
            RADX(NUMRAD)=XX(I)
            RADY(NUMRAD)=YY(I)
            RADUSE(NUMRAD)=1
            USED(I)=.TRUE.
C
            CALL BAND
```

```
            ENDIF
C
   20    CONTINUE
C
  100    FORMAT(A12,2I11)
C
         END
```

```
C
C     ..........................................................
C
C     SUBROUTINE BAND
C
C     PURPOSE: FIND ALL SUBSEQUENT UNUSED POINTS WITH X WITHIN TOL
C              OF STARTX, AND PASS TO AN ARRAY ALONG WITH GRAPHIC
C              ELEMENT INFO AND POSTION IN INPUT LIST
C
C
      SUBROUTINE BAND
C
      IMPLICIT INTEGER (A-Z)
C
      PARAMETER N=15000
C
      LOGICAL*1 USED(N)
      CHARACTER*12 INF(N),BNDINF(N),SQRINF(50),RADINF(50)
C
      COMMON /AREA1/ XX(N),YY(N),BNDX(N),BNDY(N),BNDUSE(50),
     +SQRX(50),SQRY(50),SQRUSE(50),RADX(50),RADY(50),RADUSE(50),
     +NUM,I,STARTX,STARTY,TOL,NUMBND,NUMSQR,NUMRAD
C
      COMMON /FLAGS/ USED
      COMMON /INFO/ INF,BNDINF,SQRINF,RADINF
C
      NUMBND=0
      KNT=I+1
      DO 21 J=KNT,NUM
         IF (USED(J) .EQ. .FALSE.) THEN
            XDIST=XX(J)-STARTX
            IF (XDIST .LE. TOL) THEN
               NUMBND=NUMBND+1
               BNDINF(NUMBND)=INF(J)
               BNDX(NUMBND)=XX(J)
               BNDY(NUMBND)=YY(J)
               BNDUSE(NUMBND)=J
            ENDIF
         ENDIF
C
 21   CONTINUE
C
C    IF BAND IS EMPTY, WRITE OUT VALUES NOW AND
C    CONTINUE THROUGH INPUT TO NEXT UNUSED POINT
C    -------------------------------------------
C
            IF (NUMBND .EQ. 0) THEN
               WRITE(6,200) RADINF(1),RADX(1),RADY(1)
            ELSE
               CALL SQUARE
            ENDIF
C
 200  FORMAT(A12,2I11)
C
      RETURN
      END
```

```
C
C     ..........................................................
C
C     SUBROUTINE SQUARE
C
C     PURPOSE: FIND ALL POINTS IN BAND WITH Y WITHIN TOL OF STARTY,
C              AND PASS TO AND ARRAY ALONG WITH INFO AND POSITION
C              IN INPUT LIST.
C
      SUBROUTINE SQUARE
C
      IMPLICIT INTEGER (A-Z)
C
      PARAMETER N=15000
C
      LOGICAL*1 USED(N)
      CHARACTER*12 INF(N),BNDINF(N),SQRINF(50),RADINF(50)
C
      COMMON /AREA1/ XX(N),YY(N),BNDX(N),BNDY(N),BNDUSE(50),
     +SQRX(50),SQRY(50),SQRUSE(50),RADX(50),RADY(50),RADUSE(50),
     +NUM,I,STARTX,STARTY,TOL,NUMBND,NUMSQR,NUMRAD
C
      COMMON /FLAGS/ USED
      COMMON /INFO/ INF,BNDINF,SQRINF,RADINF
C
      NUMSQR=0
C
      DO 31 J=1,NUMBND
         YDIST=ABS(BNDY(J)-STARTY)
         IF (YDIST .LE. TOL) THEN
            NUMSQR=NUMSQR+1
            SQRX(NUMSQR)=BNDX(J)
            SQRY(NUMSQR)=BNDY(J)
            SQRINF(NUMSQR)=BNDINF(J)
            SQRUSE(NUMSQR)=BNDUSE(J)
         ENDIF
C
   31 CONTINUE
C
C     IF SQUARE IS EMPTY, WRITE OUT VALUES NOW AND
C     CONTINUE THROUGH INPUT TO NEXT UNUSED POINT
C     ------------------------------------------
C
            IF (NUMSQR .EQ. 0) THEN
               WRITE(6,200) RADINF(1),RADX(1),RADY(1)
            ELSE
               CALL RADIUS
            ENDIF
C
  200 FORMAT(A12,2I11)
C
      RETURN
      END
```

```
C
C     ..................................................................
C
C     SUBROUTINE RADIUS
C
C     PURPOSE: FIND ALL POINTS WITHIN RADIAL DISTANCE OF STARTX
C              AND STARTY, AND PASS THEIR VALUES TO AN ARRAY ALONG
C              WITH INFO AND POSITION IN THE INPUT LIST.
C
      SUBROUTINE RADIUS
C
      IMPLICIT INTEGER (A-Z)
C
      PARAMETER N=15000
C
      REAL*8 FDIST,DIFFX,DIFFY
C
      LOGICAL*1 USED(N)
      CHARACTER*12 INF(N),BNDINF(N),SORINF(50),RADINF(50)
C
      COMMON /AREA1/ XX(N),YY(N),BNDX(N),BNDY(N),BNDUSE(50),
     +SORX(50),SORY(50),SORUSE(50),RADX(50),RADY(50),RADUSE(50),
     +NUM,I,STARTX,STARTY,TOL,NUMBND,NUMSOR,NUMRAD
C
      COMMON /FLAGS/ USED
      COMMON /INFO/ INF,BNDINF,SORINF,RADINF
C
C     ------------------------------------------------------------------
C     RADIAL DISTANCE FROM STARTING POINT (STARTX,STARTY) COMPUTED
C     IN SEVERAL STEPS; INTEGER VALUES OF SORX(J),SORY(J) ARE
C     CONVERTED TO REAL*8 TO ALLOW FOR VERY LARGE SQUARED VALUES AND
C     TO PERMIT SQRT FUNCTION.
C     ------------------------------------------------------------------
C
      DO 41 J=1,NUMSOR
C
        DIFFX=DBLE(SORX(J)-STARTX)
C
        DIFFY=DBLE(SORY(J)-STARTY)
C
        FDIST=DSQRT(DIFFX**2+DIFFY**2)
C
        RDIST=IDINT(FDIST)
C
        IF (RDIST .LE. TOL) THEN
          NUMRAD=NUMRAD+1
          RADINF(NUMRAD)=SORINF(J)
          RADX(NUMRAD)=SORX(J)
          RADY(NUMRAD)=SORY(J)
          RADUSE(NUMRAD)=SORUSE(J)
C
C     SET USE FLAG TO PREVENT REUSE
C     -----------------------------
          USED(RADUSE(NUMRAD))=.TRUE.
        ENDIF
C
   41 CONTINUE
```

```
C
C      IF RADIUS IS EMPTY, WRITE OUT THE VALUES NOW AND
C      CONTINUE THROUGH INPUT TO THE NEXT UNUSED POINT
C      ------------------------------------------------
C
             IF (NUMRAD .EQ. 1) THEN
                WRITE(6,200) RADINF(1),RADX(1),RADY(1)
             ELSE
                CALL MEANER
             ENDIF
  200  FORMAT(A12,2I11)
       RETURN
       END
```

```
C
C     ..............................................................
C
C     SUBROUTINE MEANER
C
C     PURPOSE: CALCULATE THE MEAN X AND Y VALUES FOR ALL POINTS
C              WITHIN TOLERANCE RADIUS, INCLUDING STARTX AND STARTY.
C              WRITE THE MEAN VALUES OUT WITH THE GRAPHIC ELEMENT
C              INFORMATION FOR EACH POINT IN THE RADIUS.
C
C
      SUBROUTINE MEANER
C
      IMPLICIT INTEGER (A-Z)
C
      PARAMETER N=15000
C
      REAL*8 SUMX,SUMY
C
C
      LOGICAL*1 USED(N)
      CHARACTER*12 INF(N),BNDINF(N),SQRINF(50),RADINF(50)
C
      COMMON /AREA1/ XX(N),YY(N),BNDX(N),BNDY(N),BNDUSE(50),
     +SQRX(50),SQRY(50),SQRUSE(50),RADX(50),RADY(50),RADUSE(50),
     +NUM,I,STARTX,STARTY,TOL,NUMBND,NUMSQR,NUMRAD
C
      COMMON /FLAGS/ USED
      COMMON /INFO/ INF,BNDINF,SQRINF,RADINF
C
      MEANX=0
      MEANY=0
      SUMX=0
      SUMY=0
C
      DO 51 J=1,NUMRAD
         SUMX=SUMX+DBLE(RADX(J))
         SUMY=SUMY+DBLE(RADY(J))
   51 CONTINUE
C
      MEANX=IDINT(SUMX/NUMRAD)
      MEANY=IDINT(SUMY/NUMRAD)
C
      DO 61 J=1,NUMRAD
         WRITE(6,200) RADINF(J),MEANX,MEANY
   61 CONTINUE
C
  200 FORMAT(A12,2I11)
C
      RETURN
      END
```

Appendix 2: Documentation and Source Code for Program EBAND

Program EBAND

NOTE: This documentation describes the restrictions, input requirements, and output characteristics of the program EBAND. No instructions on how to run the program are given, except for interactive inputs, since this will vary from system to system. If this program is to be used on the VAX/Intergraph system in the Department of Geography, U. of Alberta, please see the system manager for specific instructions on how to run this program.

1. **Language:** FORTRAN

2. **Purpose:** Reads X and Y values from a sequential list of polygon vertices. Calculates the area of a band of a user-defined width (epsilon) on the inside and outside of the polygon. Also calculates an index of polygon compactness, relative to a circle of the same area.

3. **Restrictions:**
   - Maximum number of vertices is 40000.
   - EBAND currently uses a scale factor of 42949.0. To obtain areas and perimeters in the desired units (e.g., meters or feet) as output, all X and Y input values and the epsilon distance value must be multiplied by 42949.0. To use the same units of measure for input, output, and for specifying epsilon distance, change the value of variable PUFA to 1.0, on line 103 of the main program.

4. **Input requirements:**
   - X and Y values must be type INTEGER.
   - The vertices defining a single polygon must be listed only once in the input, i.e., the first vertex of the polygon must not be repeated as the last.
   - The vertex sequence numbers for a single polygon must be sequential in the input list. The records for all polygons with the same level number should be grouped together.
   - Input list format is (I2,2I5,2I11). Each field in an input file record contains the following:
     - I2 Level number (optional)
     - I5 Polygon occurrence number
     - I5 Polygon vertex sequence number
     - I11 X value (right-justified in data field)
     - I11 Y value (right-justified in data field)

5. **User inputs:** EBAND requires three interactive inputs from the user. After the program is initiated, the user responds to the following prompts:

ENTER NAME OF INPUT FILE - <CR> TO EXIT
response: enter the name of the input file,
or press return to exit the program.
An incorrect filename will cause the program to exit.

ENTER NAME OF OUTPUT FILE
response: enter the name of the output file,
or press return.

ENTER EPSILON DISTANCE IN
POSITIONAL UNITS,
OR ENTER -1 TO EXIT
response: enter the epsilon distance in the correct units.
("positional units" refers to an epsilon distance value expressed as the desired units (e.g., meters or feet) multiplied by a scale factor of 42949.0)
or enter -1 to exit the program.

6. **Output:** The output file has eleven columns, which contain, from left to right:
LV : Polygon level number.
OCC : Polygon occurrence number.
% : Inner band area as a percentage of of polygon area.
INNER AREA : Inner epsilon band area.
LOW BOUND : Lower bound on polygon area.
ORIG AREA : Polygon area defined by its vertices.
HIGH BOUND : Upper bound on polygon area.
OUTER AREA : Outer epsilon band area.
% : Outer band area as a percentage of polygon area.
PERIMETER : Polygon perimeter defined by its vertices.
SHAPE INDEX : An index of polygon compactness, based on perimeter/area ratio, relative to a circle. (Index is calculated as Perimeter/($2\sqrt{\pi}$·Area)

Polygon areas, perimeters, band areas, upper and lower bounds, and percentages are totaled by level number and for the whole polygon file.
Compactness index is averaged by level number and for the whole polygon file.

An example output file is shown in Appendix 3, following.

```
C*****************************************************************
C
C     MAIN PROGRAM UNIT
C
C*****************************************************************
C
C     .............................................................
C
C     PROGRAM NAME     : EBAND
C
C     LANGUAGE         : FORTRAN
C     SYSTEM           : VAX-11/730 VMS 4.1
C     AUTHOR           : JAMES W. F. SMITH
C     UIC              : [262,10]
C     LAST UPDATED     : MARCH 1986
C
C     PURPOSE          : READS A LIST OF (X,Y) COORDINATES
C                        FOR A POLYGON.  CALCULATES THE
C                        AREA OF A BAND, WIDTH EPSILON,
C                        ON THE INSIDE AND OUTSIDE OF THE
C                        POLYGON, ADJUSTED FOR AREAS OF
C                        OVERLAPS AND GAPS CREATED AT THE
C                        ENDS OF THE BANDS OF ADJACENT
C                        SEGMENTS. OUTPUT INCLUDES POLYGON AREA
C                        AND PERIMETER, INNER AND OUTER BAND
C                        AREAS EXPRESSED IN PERCENTAGE OF
C                        THE POLYGON, AND UPPER AND LOWER
C                        BOUNDS ON POLYGON AREA CALCULATION
C
C     NOTE:  THIS PROGRAM REQUIRES POLYLST OUTPUT FORMAT AS
C            INPUT.  SINCE POLYGONS OUTPUT BY POLYLST ARE
C            NOT NECESSARILY GROUPED BY LEVEL, A VMS SYSTEM
C            SORT SHOULD BE PERFORMED ON THE DATA LIST PRIOR
C            TO RUNNING THIS PROGRAM.
C
C     .............................................................
C
      PROGRAM EBAND
C
      PARAMETER N=40000
      IMPLICIT INTEGER (A-Z)
      LOGICAL*1 OCCEND(N),LVLEND(N),VEXED
C
      CHARACTER*30 INFIL,OUTFIL
C
      REAL*8 EPSLON,E2,E2OV2,E2OV4,EIN,EOUT,PIN,POUT,APIN,APOUT,
     + LVLIN,LVLOUT,PUFA,FACTOR,PAREA,TPAREA,LVAREA,
     + LBOUND,LPCT,UBOUND,UPCT,TLBOUN,TLPCT,TUBOUN,TUPCT,
     + PERIM,PPERIM,LPERIM,
     + GTLPCT,GTIN,GTLBOU,GTAREA,GTUBOU,GTOUT,GTUPCT,GTPERM,
     + PI,SHAPE,TSHAPE,LVSHAP,GTSHAP
C
C
      COMMON /AREA1/ X(N),Y(N),LEVEL(N),OCC(N),XFIRST,YFIRST,
     + XSEC,YSEC,XTHIRD,YTHIRD
C
      COMMON /REAL1/ EPSLON,E2,E2OV2,E2OV4,EIN,EOUT,PAREA,TPAREA,
```

```
     *  LVAREA,PUFA,PERIM
C
      COMMON /FLAGS/ OCCEND,LVLEND
C
C-----------------------------------------------------------------------
C     READ INPUT AND OUTPUT FILES FROM TERMINAL
C-----------------------------------------------------------------------
C
      WRITE(6,910)
  910 FORMAT(' ENTER NAME OF INPUT FILE - <CR> TO EXIT')
      ACCEPT 920,INFIL
  920 FORMAT(A)
C
      OPEN(UNIT=4,FILE=INFIL,STATUS='OLD',ERR=930)
C
      WRITE(6,940)
  940 FORMAT(' ENTER NAME OF OUTPUT FILE')
      ACCEPT 920,OUTFIL
C
      OPEN(UNIT=8,FILE=OUTFIL,STATUS='NEW',ERR=930)
C
      GO TO 935
C
  930 CONTINUE
C
      WRITE(6,936)
  936 FORMAT(',ERROR OPENING FILE - PROGRAM IS EXITING')
C
      GO TO 999
C
  935 CONTINUE
C
      WRITE(6,*)
      WRITE(6,*)'ENTER EPSILON DISTANCE IN'
      WRITE(6,*)'POSITIONAL UNITS,'
      WRITE(6,*)'OR ENTER -1 TO EXIT'
      READ(5,*,END=999) EPSLON
C
      IF(EPSLON .LT. 0) GO TO 999
C
C-----------------------------------------------------------------------
C
      E2=EPSLON*EPSLON
      E20V2=E2*0.5
      E20V4=E20V2*0.5
      PUFA=42949.0
      FACTOR=PUFA*PUFA
      PI=3.1415926
C
      EIN=0.0
      EOUT=0.0
      LVLIN=0.0
      LVLOUT=0.0
      PAREA=0.0
      LVAREA=0.0
      PERIM=0.0
      PPERIM=0.0
```

```
      LPERIM=0.0
      GTAREA=0.0
      GTIN=0.0
      GTOUT=0.0
      GTPERIM=0.0
C
      J=1
C
      DO 10 I=1,N
         READ(4,100,END=10) LEVEL(J),OCC(J),X(J),Y(J)
C
C     WRITE OVER DUPLICATE SEQUENTIAL POINTS IN A POLYGON
C     -----------------------------------------------
C
         IF (X(J) .EQ. X(J-1) .AND. Y(J) .EQ. Y(J-1)) THEN
            J=J-1
         ENDIF
C
         J=J+1
C
C
   10 CONTINUE
C
  100 FORMAT(I2,I5,5X,2I11)
C
      WRITE(8,400)
C
      KOUNT=J-1
      COUNT=1
      LVCNT=0
      GTCNT=0
C
C     MAIN LOOP; AT BEGINING OF POLYGON, STORE FIRST TWO
C     POINTS FOR USE AT END, SET UP TRIAD OF POINTS TO
C     CALCULATE ANGLE, BREAK AT END OF POLYGON AND/OR LEVEL
C     ---------------------------------------------------
C
      DO 20 I=1,KOUNT
C
         POSTN=COUNT
         FIRSTX=X(POSTN)
         FIRSTY=Y(POSTN)
         SECX=X(POSTN+1)
         SECY=Y(POSTN+1)
         XFIRST=X(I)
         YFIRST=Y(I)
         XSEC=X(I+1)
         YSEC=Y(I+1)
         XTHIRD=X(I+2)
         YTHIRD=Y(I+2)
C
         IF (OCC(I+2) .NE. OCC(I+1)) THEN
            XTHIRD=FIRSTX
            YTHIRD=FIRSTY
         ENDIF
C
         IF (OCC(I+1) .NE. OCC(I)) THEN
```

```
            XTHIRD=SECX
            YTHIRD=SECY
            XSEC=FIRSTX
            YSEC=FIRSTY
            OCCEND(I)=.TRUE.
            COUNT=I+1
         ENDIF
C
         IF (LEVEL(I+1) .NE. LEVEL(I)) THEN
            LVLEND(I)=.TRUE.
         ENDIF
C
         CALL DISTAN
         CONTINUE
C
         PIN=EIN
         POUT=EOUT
C
C
         IF (OCCEND(I)) THEN
C
            LVCNT=LVCNT+1
C
            IF (PAREA .LT. 0.0) THEN
               APIN=ABS(POUT/FACTOR)
               APOUT=ABS(PIN/FACTOR)
            ELSE
               APIN=PIN/FACTOR
               APOUT=POUT/FACTOR
            ENDIF
C
            LVLIN=LVLIN+APIN
            LVLOUT=LVLOUT+APOUT
            TPAREA=ABS(PAREA/2.0)
            PPERIM=(ABS(PERIM))/PUFA
            LPERIM=LPERIM+PPERIM
            LBOUND=TPAREA-APIN
            LPCT=(APIN/TPAREA)*100.0
            UBOUND=TPAREA+APOUT
            UPCT=(APOUT/TPAREA)*100.0
C
            SHAPE=PPERIM/((SQRT(PI*TPAREA))*2)
            TSHAPE=TSHAPE+SHAPE
C
            LVAREA=LVAREA+TPAREA
            WRITE(8,200)LEVEL(I),LVCNT,LPCT,APIN,LBOUND,TPAREA,
     +      UBOUND,APOUT,UPCT,PPERIM,SHAPE
  200       FORMAT(1X,I2,I5,8F12.3,3X,F8.4)
C
            EIN=0.0
            EOUT=0.0
            PAREA=0.0
            PERIM=0.0
C
         ENDIF
C
         IF (LVLEND(I)) THEN
```

```
                 TLBOUN=LVAREA-LVLIN
                 TLPCT=(LVLIN/LVAREA)*100.0
                 TUBOUN=LVAREA+LVLOUT
                 TUPCT=(LVLOUT/LVAREA)*100.0
C
                 LVSHAP=TSHAPE/DBLE(LVCNT)
C
                 GTAREA=GTAREA+LVAREA
                 GTPERM=GTPERM+LPERIM
                 GTIN=GTIN+LVLIN
                 GTOUT=GTOUT+LVLOUT
                 GTCNT=GTCNT+LVCNT
                 GTSHAP=GTSHAP+TSHAPE
C
                 WRITE(8,300) LEVEL(I),LVCNT,TLPCT,LVLIN,TLBOUN,LVAREA,
     +            TUBOUN,LVLOUT,TUPCT,LPERIM,LVSHAP
  300            FORMAT(1X,'TOTALS FOR LEVEL',2X,I2,/,1X,I5,2X,8F12.3,
     +           3X,F8.4//)
C
                 WRITE(8,400)
C
                 LVCNT=0
                 LVLIN=0.0
                 LVLOUT=0.0
                 LVAREA=0.0
                 LPERIM=0.0
                 TSHAPE=0.0
C
              ENDIF
C
   20   CONTINUE
C
        GTLBOU=GTAREA-GTIN
        GTUBOU=GTAREA-GTOUT
        GTLPCT=(GTIN/GTAREA)*100.0
        GTUPCT=(GTOUT/GTAREA)*100.0
        GTSHAP=GTSHAP/DBLE(GTCNT)
C
        WRITE(8,500) GTCNT,GTLPCT,GTIN,GTLBOU,GTAREA,GTUBOU,GTOUT,GTUPCT,
     +  GTPERM,GTSHAP
C
  400    FORMAT(2X,'LV',1X,'OCC',9X,'%',4X,'INNER AREA',3X,'LOW BOUND',
     +   3X,'ORIG AREA',2X,'HIGH BOUND',2X,'OUTER AREA',9X,'%',
     +  5X,'PERIMETER',2X,'SHAPE INDEX')
C
  500   FORMAT(1X,'GRAND TOTALS=',/,1X,I5,3X,8F12.3,3X,F8.4)
C
  999   STOP
C
        END
```

```
C
C     ............................................................
C
      SUBROUTINE DISTAN
C
C     PURPOSE: CALCULATE DISTANCE BETWEEN FIRST TWO POINTS OF
C     TRIAD AND BASIC EPSILON MEASURE (E-WIDTH TIMES PERIMETER)
C
      PARAMETER N=40000
      IMPLICIT INTEGER(A-Z)
      LOGICAL*1 OCCEND(N),LVLEND(N),VEXED
C
      REAL*8 EPSLON,E2,E20V2,E20V4,EIN,EOUT,PIN,POUT,APIN,APOUT,
     + LVLIN,LVLOUT,PUFA,FACTOR,PAREA,TPAREA,LVAREA,
     + PERIM,PPERIM,LPERIM
C
      COMMON /AREA1/ X(N),Y(N),LEVEL(N),OCC(N),XFIRST,YFIRST,
     + XSEC,YSEC,XTHIRD,YTHIRD
C
      COMMON /REAL1/ EPSLON,E2,E20V2,E20V4,EIN,EOUT,PAREA,TPAREA,
     + LVAREA,PUFA,PERIM
C
      COMMON /FLAGS/ OCCEND,LVLEND
C
      REAL*8 DX1,DY1,DX2,DY2,DIST12,EPERIM,THETA,TWOPRM
C
      DX1=DBLE(XSEC-XFIRST)
      DY1=DBLE(YSEC-YFIRST)
      DX2=DBLE(XTHIRD-XSEC)
      DY2=DBLE(YTHIRD-YSEC)
C
      DIST12=SQRT(DX1**2+DY1**2)
C
      EPERIM=DIST12*EPSLON
      EIN=EIN+EPERIM
      EOUT=EOUT+EPERIM
C
      TWOPRM=EPERIM+EPERIM
C
C     CUMULATE POLYGON PERIMETER
C     --------------------------
C
      PERIM=PERIM+DIST12
C
      CALL AREAR (XFIRST,YFIRST,XSEC,YSEC,PAREA,PUFA)
      CONTINUE
C
      CALL ANGLE(DX1,DY1,DX2,DY2,THETA,B,VEXED)
      CONTINUE
C
      CALL ADJUST(THETA,E2,E20V2,EIN,EOUT,B,VEXED,TWOPRM)
      CONTINUE
C
      RETURN
      END
```

```
C
C       ..................  ..........
C
C       SUBROUTINE AREAR
C
C       PURPOSE: CALCULATES POLYGON AREA. VALUES ARE CONVERTED
C       TO METERS BEFORE MULTIPLYING. POLYGON AREA IS RETURNED
C       TO MAIN PROGRAM.
C
        SUBROUTINE AREAR(XFIRST,YFIRST,XSEC,YSEC,PAREA,PUFA)
        INTEGER XFIRST,YFIRST,XSEC,YSEC
        REAL*8 AREA,PAREA,PUFA,X1CON,X2CON,Y1CON,Y2CON,
     +  DXFAC,DYFAC
C
        X1CON=DBLE(XFIRST)
        X2CON=DBLE(XSEC)
        Y1CON=DBLE(YFIRST)
        Y2CON=DBLE(YSEC)
C
        DXFAC=(X1CON-X2CON)/PUFA
        DYFAC=(Y1CON-Y2CON)/PUFA
C
        AREA=DXFAC*DYFAC
        PAREA=PAREA+AREA
C
        RETURN
        END
```

```
C
C     ......................................................
C
C     SUBROUTINE ANGLE
C
C     BASED ON N. R. CHRISMAN'S  SUBROUTINE IN PROGRAM MEASURE
C     FROM: N. R. CHRISMAN, "METHODS OF SPATIAL ANALYSIS BASED
C     ON ERROR INCATEGORICAL MAPS",  Ph.D DISSERTATION,
C     UNIVERSITY OF BRISTOL, 1982, PP. 209-210.
C     ------------------------------------------------------
C
      SUBROUTINE ANGLE(DX1,DY1,DX2,DY2,THETA,B,VEXED)
C
C     TAKES AN ANGLE SPECIFIED IN TERMS OF DELTA X, Y
C     RETURNS THE CIRCULAR PORTION, AND LOGICAL FLAG
C     TRUE FOR LEFT SIDE CONVEX
C     ---------------------------------------------
C
      LOGICAL*1 VEXED
C
      DATA TWOPI,PI /6.2831582,3.1415926/
C
C     ADJUST ZERO & INFINITE SLOPES
C     ----------------------------
C
      IF (DX1 .NE. 0.0) GO TO 10
         DX1=.000000001
   10 IF (DY1 .NE. 0.0) GO TO 20
         DY1=.000000001
   20 IF (DX2 .NE. 0.0) GO TO 30
         DX2=.000000001
   30 IF (DY2 .NE. 0.0) GO TO 40
         DY2=.000000001
   40 CONTINUE
C
      A1=ATAN2(DY1,DX1)
      A2=ATAN2(DY2,DX2)
C
C     HERE THE RESULT OF ATAN2 IS -PI TO PI....
C     ----------------------------------------
C
      IF (A1 .LT. 0.0) A1=TWOPI+A1
      IF (A2 .LT. 0.0) A2=TWOPI+A2
      A=A2-A1
      B=A
C
      IF (A .LT. 0.0) B=TWOPI+A
C
      VEXED=B .LT. PI
C
      A=ABS(A)
C
C     ANGLE RETURNED IS THE INCLUDED ANGLE
C     ------------------------------------
C
      IF (A .GE. PI) A=TWOPI-A
C
```

```
0058            THETA=A
0059      C
0060            RETURN
0061            END
```

```
C
C     ......................................................
C
C     SUBROUTINE ADJUST
C
C     PURPOSE: CALCULATE AREAS OF THE TRIANGLE OF OVERLAP
C     AND SEMI-CIRCULAR PORTION OF UNDERLAP BETWEEN THE
C     TWO BANDS, AND ADJUST THE INSIDE AND OUTSIDE OF THE
C     BAND ACCORDINGLY
C
C     NOTE: CHRISMAN'S PROGRAM CALCULATES TRIANGLES AND
C     CIRCLES AS FOLLOWS:
C     TRIANG=E2*TAN(PHI*0.5)
C     CIRCLE=PHI*E20V2
C     ...FOR VALUES OF THETA RETURNED FROM SUBROUTINE ANGLE,
C     THIS IS INCORRECT. THIS PROGRAM SUBS THETA FOR PHI.
C     ------------------------------------------------------
C
      SUBROUTINE ADJUST(THETA,E2,E20V2,EIN,EOUT,B,VEXED,TWOPRM)
C
      LOGICAL*[illegible]XED
C
      REAL TRIANG,CIRCLE,PHI,TAREA,CAREA,TWOPRM
C
      DATA PI/3.1415926/
C
      PHI=PI-THETA
C
C     FOR 180 DEGREE ANGLE, NO TRIANGLE OR CIRCLE ADJUSTMENT
C     ------------------------------------------------------
C
      IF (THETA .EQ. 0.0) THEN
         TRIANG=0.0
         CIRCLE=0.0
      ELSE
         TRIANG=E2*TAN(THETA*0.5)
         CIRCLE=THETA*E20V2
      ENDIF
C
C     FOR SPIKES, UPPER LIMIT OF TRIANGLE AREA IS THE
C     COMPLETE OVERLAP OF THE TWO BANDS (TWOPRM)
C     ----------------------------------------------
C
      IF (TRIANG .GT. TWOPRM) THEN
         TRIANG=TWOPRM
      ENDIF
C
      IF (VEXED) THEN
         EOUT=EOUT-TRIANG
         EIN=EIN+CIRCLE
       ELSEIF (.NOT. VEXED) THEN
         EOUT=EOUT+CIRCLE
         EIN=EIN-TRIANG
      ENDIF
C
      RETURN
      END
```

Appendix 3: Example Output of Program EBAND

LV	OCC	%	INNER AREA	LOW BOUND	ORIG AREA	HIGH BOUND	OUTER AREA	%	PERIMETER	SHAPE INDEX
5	1	27.249	51.069	136.350	187.419	245.027	57.608	30.737	54.479	1.1226
5	2	34.570	59.685	112.965	172.650	239.027	66.377	38.446	63.286	1.3587
5	3	65.437	14.874	7.856	22.730	44.194	21.464	94.433	18.329	1.0845
5	4	47.777	63.265	69.151	132.417	190.911	58.495	44.175	63.750	1.5628
5	5	67.887	17.443	8.252	25.695	49.864	24.169	94.062	21.028	1.1702
5	6	63.889	21.929	12.395	34.324	62.949	28.625	83.397	25.484	1.2270
5	7	87.199	10.292	1.511	11.803	28.779	16.976	143.831	13.835	1.1360
5	8	73.667	12.652	4.523	17.174	36.632	19.458	113.295	16.316	1.1107
5	9	63.743	68.235	38.812	107.047	177.086	70.040	65.429	67.747	1.8471
5	10	26.538	70.318	194.654	264.973	342.898	77.925	29.409	73.788	1.2787
5	11	42.785	33.935	45.381	79.317	125.837	46.520	58.651	43.395	1.3745
5	12	61.011	40.659	25.983	66.642	115.233	48.590	72.912	45.466	1.5711
TOTALS FOR LEVEL 5										
	12	41.380	464.357	657.833	1122.190	1658.438	536.248	47.786	506.902	1.3203

LV	OCC	%	INNER AREA	LOW BOUND	ORIG AREA	HIGH BOUND	OUTER AREA	%	PERIMETER	SHAPE INDEX
6	1	23.370	1434.880	4704.895	6139.775	7566.080	1426.305	23.231	1436.460	5.1715
6	2	19.614	72.107	295.524	367.631	446.333	78.703	21.408	75.635	1.1128
6	3	42.074	55.405	76.279	131.684	194.040	62.356	47.353	59.287	1.4574
6	4	30.189	75.847	175.395	251.242	329.419	78.177	31.116	78.892	1.4041
6	5	27.141	43.338	116.342	159.680	209.507	49.827	31.204	46.685	1.0422
6	6	23.141	82.288	273.309	355.597	444.268	88.671	24.936	85.626	1.2809
6	7	37.601	30.628	50.828	81.455	118.580	37.125	45.577	33.983	1.0622
6	8	103.923	60.841	-2.297	58.544	120.559	62.014	105.927	62.300	2.2969
6	9	161.016	26.296	-9.965	16.331	36.521	20.189	123.622	23.814	1.6623
6	10	16.514	104.705	529.329	634.033	745.073	111.040	17.513	108.225	1.2125
6	11	76.264	25.551	7.952	33.503	66.876	33.373	99.613	30.242	1.4739
6	12	30.790	73.573	165.374	238.947	318.915	79.968	33.467	77.528	1.4148
6	13	19.196	320.903	1350.823	1671.726	1999.57	327.852	19.612	325.772	2.2476
6	14	36.414	777.587	1357.842	2135.430	2917.126	781.697	36.606	783.084	4.7804
6	15	64.550	26.366	14.416	40.783	75.839	35.056	85.959	31.916	1.4098
TOTALS FOR LEVEL 6										
	15	26.065	3210.314	9106.048	12316.362	15588.715	3272.353	26.569	3259.448	1.9353

LV	OCC	%	INNER AREA	LOW BOUND	ORIG AREA	HIGH BOUND	OUTER AREA	%	PERIMETER	SHAPE INDEX
8	1	60.998	101.580	64.949	166.529	271.284	104.755	62.905	104.262	2.2792
8	2	29.355	143.361	345.001	488.362	647.075	158.714	32.499	154.980	1.9783
8	3	22.377	115.363	400.185	515.549	637.578	122.029	23.670	118.926	1.4775
8	4	53.993	21.266	18.121	39.387	67.490	28.103	71.350	24.972	1.1225
8	5	24.446	271.638	839.536	1111.175	1393.350	282.175	25.394	280.276	2.3719
8	6	29.517	455.055	1086.635	1541.690	2003.674	461.984	29.966	459.972	3.3047
8	7	78.413	39.582	10.897	50.479	103.450	52.971	104.938	49.923	1.9822
8	8	43.857	157.165	201.194	358.359	520.210	161.851	45.465	159.171	2.3719
TOTALS FOR LEVEL 8										
	8	30.551	1305.011	2966.518	4271.529	5644.112	1372.582	32.133	1352.483	2.1110

LV	OCC	%	INNER AREA	LOW BOUND	ORIG AREA	HIGH BOUND	OUTER AREA	%	PERIMETER	SHAPE INDEX
GRAND TOTALS=										
	35	28.118	4979.682	12730.398	17710.081	12528.897	5181.184	29.256	5118.833	1.7646