centro de educación continua

división de estudios superiores

facultad de ingeniería, unam

A LOS ASISTENTES A LOS CURSOS DEL CENTRO DE EDUCACION CONTINUA

Las autoridades de la Facultad de Ingeniería, por conducto del Jefe del Centro de Educación Continua, Dr. Pedro Martínez Pereda, otorgan una constancia de asistencia a quienes cumplan con los requisitos establecidos para cada curso. Las personas que deseen que aparezca su título profesional precediendo a su nombre en la constancia, deberán entregar copia del mismo o de su cédula a más tardar el SEGUNDO DIA de clases, en las oficinas del Centro con la señorita Barraza, encargada de inscripciones.

El control de asistencia se llevará a cabo a través de la persona encargada de entregar las notas del curso. Las inasistencias serán computadas por las autoridades del Centro, con el fin de entregarle constancia solamente a los alumnos que tengan un mínimo del 80% de asistencia.

Se recomienda a los asistentes participar activamente con sus ideas y experiencias, pues los cursos que ofrece el Centro están planeados para que los profesores expongan una tesis, pero sobre todo, para que coordinen las opiniones de todos los interesados constituyendo verdaderos seminarios.

Es muy importante que todos los asistentes llenen y entreguen su hoja de inscripción al inicio del curso. Las personas comisionadas por alguna institución deberán pasar a inscribirse en las oficinas del Centro en la misma forma que los demás asistentes entregando el oficio respectivo.

Con objeto de mejorar los servicios que el Centro de Educación Continua ofrece, al final del curso se hará una evaluación a través de un cuestionario diseñado para emitir juicios anónimos por parte de los asistentes.

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO
FACULTAD DE INGENIERIA
DIVISION DE ESTUDIOS SUPERIORES
CENTRO DE EDUCACION CONTINUA
DIRECTORIO GENERAL

REGISTRO DE ASISTENTES Y PROFESORES.

NOMBRE DEL CURSO: ______________________ FOLIO [| | | |] CLAVE ASOC [|]
1 5 6 7

[| | | | |] [...]
8 13 14 NOMBRE(S) APELLIDO PATERNO APELLIDO MATERNO 41

REG. FED. CAUS. [...] CED. PROF. [...]
42 51 52 58

TEL. PARTICULAR [...] TEL. OFICINA [...] EXTENSION [...]
59 65 66 72 73 76

MARQUE CON UNA CRUZ

ASISTENTE [ ] PROFESOR [ ] [ ] [1]
77 80

[...]
8 DOMICILIO PARTICULAR (CALLE. NUMERO Y No. INTERIOR) 41

[...] Z.P. [|]
42 COLONIA 71 72 73

______________________ [|]
ESTADO 74 75

______________________ [|] ______________________ [|] [2]
TITULO PROFESIONAL 76 77 ESPECIALIDAD 78 79 80

[...]
8 DOMICILIO DE OFICINA (CALLE, NUMERO Y No. INTERIOR) 41

[...] Z.P. [|]
42 COLONIA 71 72 73

______________________ [|] [3]
ESTADO 74 75 80

:OCIACIONES A LAS QUE PERTENECE.

______________________ [|] ______________________ [|]

______________________ [|] ______________________ [|]

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION
(del 24 de julio al 4 de agosto, 1978)

FECHA	DURACION	TEMA	PROFESOR
24 de julio	18:00 h	Recepción y entrega de material	
	18:15 h	Inauguración e Introducción al curso.	
	19 a 21 h	Bancos de Datos Geográficos.	Dr. Adolfo Guzmán Arenas
25 de julio	18 a 19:50	Producción de mapas por computadora.	Arq. Alejandro Villanueva Egan
	20 a 21 h	1a. práctica. Producción de mapas con el programa SYMAP.	Arq. Alejandro Villanueva Egan Mario Rodríguez Green Odette Barrón Roberto Carrasco Martín Rivera
26 de julio	18 a 21 h	Sistemas de Computación Gráfica Sesión Teórica-práctica en American Trade Center Liperpool No. 31	Ing. Douglas Thorson Ing. Carlos Triay
27 de julio	18 a 19 h	Geo-estadística	Arq. Alejandro Villanueva Egan
	19 a 21 h	3a. práctica Construcción de Superficies Estadísticas con el programa SYMAP.	Laboratoristas de la Sección de Planeación

28 de julio	18 a 19:50 h	Geo-informática y Planeación. Desarrollo para los Censos de 1980.	Dr. Enrique Calderón Alzati
	20 a 21 h,	4a. práctica Sistema GEo-municipal de información. Secretaría de Programación y Presupuesto	
31 de julio	18 a 19:30 h	Recnocimiento de formas por computadora	Dr. Adolfo Guzmán Arenas
	19:40 a 21 h	Modelos digitales de terrenos	Mat. Dora Gómez
1º de agosto	18 a 21 h	Sistema IMGRID Análisis de uso del suelo y su impacto ambiental 5a. práctica Arq. Ale	Arq. Alejandro Villanueva Egan y Laboratoristas de la Sección de Planeación, D.E.S.F.I.
2 de agosto	18 a 21 h	Sistemas DETENAL San Antonio Abad No. 124	Ing. Manuel González Ing. Ramiro Bermudez Arq. Rafael Moranchell
3 de agosto	18 a 21 h.	Sistemas de Percepción Remota 6a. práctica Sistema PR, IIMAS, UNAM	M. en C. Armando Jinich Mat. Rosa M. Seco
4 de agosto	18 a 19:50	Geo-Procesamiento en los sistemas de información urbanos y regionales	Ing. Alberto Torfer Martell
	20 a 20:40	Conclusiones y revisión de prácticas	
	20:40	Clausura y entrega de cer	

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION.

PRACTICA No. 1 PAQUETE SYMAP

OBJETIVO.-

- Preparar la entrada necesaria para realizar un mapa isopleta con el programa SYMAP, utilizando los paquetes: A- DELINEAMIENTO, B-PUNTOS DATO, C- LETREROS, E- VALORES, F- MAPA. (opciones 1-10).

MATERIAL.-

- Mapa fuente del contorno de la República Mexicana definido por los vértices señalados, y con la localización territorial aproximada de las 15 ciudades marcadas por un asterisco y un número adjunto.
- Lista con los 15 números de referencia, los nombres de las ciudades y la población estimada para 1975 por la Dirección General de Estadística en base a proyecciones del X Censo General de Población 1970.

PROCEDIMIENTO.-

1.- Geocodificar los vértices del contorno de la República y preparar un paquete A- DELINEAMIENTO.

2.- Geocodificar los puntos que representan a las ciudades, y con esto preparar un paquete B- PUNTOS DATO.

3.- Digitalizar los letreros que aparecen en el mapa (1 letrero de punto vertical; 1 letrero de punto horizontal; 1 letrero de línea; y un letrero de área, y con estos preparar un sub-paquete C- LETREROS.

4.- Con los valores proporcionados en la lista de la población considerada para cada una de las 15 localidades mencionadas desarrollar un paquete E-VALORES. Recuerde que el orden en el que fueron proporcionadas las coordenadas de los puntos debe ser el mismo orden en el que se proporcionen los valores asignados a cada punto.

5.- Construír el paquete F-MAPA perforando en las tres primeras tarjetas (en formato alfanumérico y hasta la columna 72 lo siguiente:

Tarjeta 1: Centro de Educación Continua DESFI-UNAM.

Tarjeta 2: SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION.

Tarjeta 3: Nombre del usuario, número de práctica, fecha.

Tarjeta 4: Opción 1 para el manejo del tamaño del mapa de salida.

Tarjeta 5: En adelante opción 10, diseñe para esta opción forma de arreglar la lista proporcionada en el material de la práctica, con los números de referencia, (orden en el que fueron proporcionadas las ciudades) sus nombres y los valores de población asociados a cada una.

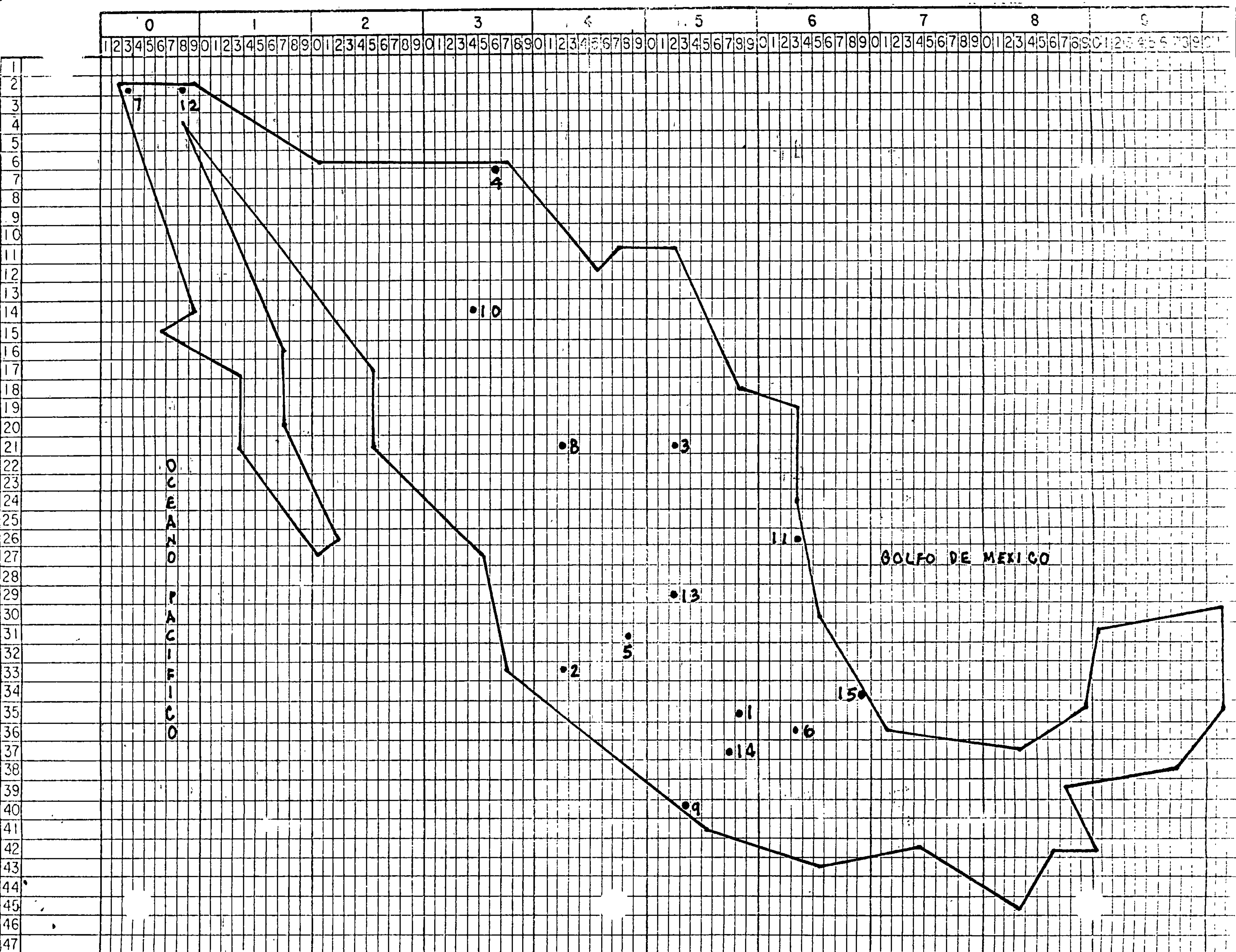
6.- Conjunte los paquetes realizados y proceda a efectuar última revisión de la codificación, tenga presente que un buen chequeo en el escritorio ayuda a evitar corridas infructuosas.

Perfore las tarjetas de control y entregue su paquete para ser leído y ejecutado por la computadora.

Obtenga su listado, investigue las causas de errores, corríjalos y vuelva a procesar sus tarjetas hasta obtener un mapa satisfactorio.

MATERIAL PARA LA PRACTICA

No.	NOMBRE DE LA CIUDAD	ENDIDAD FEDERATIVA	POBLACION	ESTIMADA PARA 1975
01	MEXICO	DISTRITO FEDERAL	11 339 774	HABITANTES
02	GUADALAJARA	JALISCO	1 963 277	"
03	MONTERREY	NUEVO LEON	1 631 681	"
04	CD. JUAREZ	CHIHUAHUA	520 539	"
05	LEON	GUANAJUATO	496 598	"
06	PUEBLA	PUEBLA	482 155	"
07	TIJUANA	B.C.N.	386 852	"
08	TORREON	COAHUILA	364 469	"
09	ACAPULCO	GUERRERO	352 673	"
10	CHIHUAHUA	CHIHUAHUA	346 003	"
11	TAMPICO	TAMAULIPAS	343 384	"
12	MEXICALI	B.C.N.	331 059	"
13	SN. LUIS POTOSI	S.L.P.	281 534	"
14	CUERNAVACA	MORELOS	273 986	"
15	VERACRUZ	VERACRUZ	266 255	"



0 1 2 3 4 5 6 7 8 9
OCEANO PACIFICO
GOLFO DE MEXICO
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

MAPAS POR COMPUTADORA

APENDICE BIBLIOGRAFICO

JULIO, 1978.

Palacio de Minería Calle de Tacuba 5, primer piso. México 1, D. F.

PREFACE

CROPP, or Conformal Representation of the Prairie Provinces, was developed as one of the tools to meet the specific needs of two projects: the review of branch line abandonment applications by the Canadian Transport Commission, and the Whitemud River Watershed Resource Study undertaken in part by the Lands Directorate, Department of the Environment. These projects related agricultural statistics to individual parcels of farmland in the Prairie Provinces in order that they could be mapped on overlays for conformal maps and for calculation of distances by computer.

The acronym CROPP was chosen to stress one of the applications of the file rather than to describe the file. CROPP contains latitude and longitude coordinates of the centroids of quarter sections of land described by the Dominion Land Survey. The use of latitude and longitude coordinates enables each parcel of land to be expressed on any projection system.

CROPP represents the cooperative efforts of many persons from various government organizations. We wish to take this opportunity to express to all of them our sincere thanks. We are particularly grateful to Mr. Maurice Head, General Director of Management and Information Services, Canadian Wheat Board; Messrs. G. Moppett and P. Hibert, Computer Science Centre; Mr. C.E. Hoganson and Miss M. Brennan, Geodetic Survey Branch, Department of Energy, Mines and Resources; Messrs. B. Gill and W. Bel., Economics Branch of the Canadian Transport Commission; and

LIST OF FIGURES

INTRODUCTION

The purpose of CROPP was to increase the usefulness of data banks that are based on land descriptions using the Dominion Land Survey (DLS). CROPP, a file of coordinate points of the centroids of sections and quarter sections in the DLS, was developed to spatially relate parcels of land in the Prairie Provinces and to permit, among other things, computer mapping. In designing the system, maximum flexibility was accomplished by assigning unique, universally recognized coordinates, that is, latitude and longitude correct to 5 decimal places, to the appropriate centroids. Because these coordinates can be readily transformed into any coordinate system, CROPP can be used with any computer mapping technique as well as merged into any data bank based on the DLS system.

An understanding of the DLS system is essential in order to comprehend CROPP. In brief, the DLS system is based on a township 6 miles square, which is divided into 36 sections (Figure 1) of 1 square mile or 640 acres each. The section is legally subdivided into four quarter sections of 160 acres each. Each parcel of land has its unique legal description. The sections are numbered 1 to 36 within a township according to the arrangement in Figure 1. Even if the entire section is not present, the numbering system remains constant for those sections that are present. Townships are numbered consecutively, northward from the 49th parallel of latitude and the ranges are numbered sequentially east and west of the principal meridian, 97°27'28" west longitude, and west from the other five meridians. Meridians are separated by approximately 4^{o} of longitude. Thus, parcels of land are officially designated by quarter section compass location, that is, NE, NW, SE, SW, section, township, range and meridian numbers.

The townships were laid out along a base line, two townships on each side. The base lines were used as controls with correction lines inserted every fourth township to allow for the convergence of the lines of longitude. Thus, in a north-south direction, a jog occurs at every fourth township and some ranges completely disappear as they merge toward meridian lines. For example, the townships within Range 30 become progessively smaller and at Township 14, Range 30 completely disappears (Figure 2).

There are five different surveys, three of which incorporate the Prairie Provinces. Where the surveys meet, irregularly shaped townships can occur. For example, Township 18A, which contains less than 36 sections, occurs where the second and third surveys meet. Townships lying adjacent to a meridian may also contain less than 36 sections, for example, townships in Range 30 south of Township 14.

However, in spite of the variation in the number of sections within a township, the numbering system was consistent with that in Figure 1. This consistency was critical to the development of CROPP because it enabled the creation of hypothetical, regular-sized townships for those that contained less than 36 sections and it assigned coordinates to the existing sections without modifying the program.

METHODOLOGY

Development of CROPP

In the spring of 1971, a search was made to determine if a computer program existed that would assign coordinates to the legal land descriptions found in various data banks. As far as was determined, such a system had not been devised. During the summer, CROPP was developed in three steps: coding township corners, editing the coded data, and calculating the coordinates for the quarter section centroids.

The Geodetic Survey Branch, Department of Energy, Mines and Resources, provided a list of corrected latitude and longitude coordinates expressed in degrees, minutes, seconds and hundredths of seconds (a hundredth of a second is approximately 1 foot on the ground) for one corner, usually the northeast, of each township. This information was coded onto punched cards by first locating the approximately 12,000 corners on 1:250,000-scale national topographic series (NTS) maps. Assuming the coordinates of one township represented a corner of the adjacent township, the latitude and longitude coordinates were coded first and then the corner that had been recorded by the Geodetic Survey Branch and the adjacent corners on the topographic maps. For example, the coordinates for the NE corner of Township 3, Range 10 are also the coordinates for the SE corner, Township 4, Range 10; SW corner, Township 4, Range 9; and NW corner, Township 3, Range 9. Normally, the coordinates represented four corners, but in the case of jogs at correction lines they represented only two (Figure 4). The township corner identification for each set of coordinates were coded in a clockwise direction, that is NE, SE, SW, NW.

By this process, all known corners for each township were identified. In order to generate coordinates for the quarter sections, at least two corners for each township had to be known. In cases where only one township corner could be located a second hypothetical corner was created.

To check this coded data, an edit program was written to locate keypunch errors, incorrect record length, incorrect corner identification, and incorrect township indentification. The latitude and longitude coordinates were checked by generating theoretical coordinates correct to 5 decimal places and comparing these with the originally coded ones. If there was a discrepancy of more than 20 seconds, approximately 1/3 of a mile, an error was recorded and the record was checked manually.

When the coordinate data cards had been checked and corrected, their data were put onto tape and sorted by meridian, range, township, and corner. From this information, a program was written to assign latitude and longitude coordinates to the centroid of each section and quarter section by using three basic data manipulations: the identification of the sections and part sections by number and part, that is NE, NW, SE, SW; the calculation of the centroid of each section and quarter section from the township corners; and the creation of CROPP by combining the calculated coordinates with the appropriate section or quarter section identification.

The calculations for the centroid of each section were made by dividing the sides of each township into 12 equal parts and numbering them sequentially. In the case of the quarter sections, the township sides were divided into 24 equal parts (see Figure 5). The intersection of the odd-numbered lines formed the desired centroids.

The calculations of the centroid coordinates were done within a single township. Therefore, errors that may arise were contained within the township and are not cumulative. The coordinate file, CROPP, contains centroids of quarter sections, but by a simple modification of the program the centroid of any desired size of unit within a township can be calculated.

In summary, CROPP contains approximately 1,900,000 records sorted in order of meridian, range, township, section, and part-section. Each of these records contains the latitude and longitude coordinates correct to 5 decimal places.

Development of Data Banks

The purpose of CROPP was to increase the usefulness of existing data banks based on the DLS land descriptions. To date, CROPP has been successfully incorporated into copies of the Canadian Wheat Board records, the Manitoba Municipal Assessment Rolls, and the Manitoba Crop Insurance Corporation files. These records were sorted into the same order as CROPP and put through a match program that assigned latitude and longitude coordinates to the legal land descriptions in the data bank. When the two records did not match, zero latitude and longitude coordinates were assigned to the description to allow for later updating and the record was written in a no-match file for checking and correction. However, all of the records adequately described by the DLS system matched.

Not all of the parcels of land are described by quarter sections or sections, which were the only units for which centroid coordinates were calculated. Two alternative measures were used to overcome this problem; any description for a parcel greater than a quarter section was assigned to coordinates of the section, or all units greater than a quarter section were broken into quarter sections. For example, if the northern half of a section was described, it was divided into the NE and NW quarter sections. The particular method used depended upon the type of output that was desired.

Once data banks with coordinates exist various manipulations, such as calculating distances and mapping, can be carried out.

APPLICATIONS

Non-mapping Applications

The CROPP coordinates enable easy and accurate calculation of distances. For example, from the Canadian Wheat Board file containing the legal land description of individual parcels of land, the distance from a given parcel of land to a delivery point can be calculated. To compute the road distance, it was assumed that this distance approximates the distance east-west and north-south to the delivery point. That is, the distance traveled by a farmer on prairie roads that are laid out on a grid would be the distance along two sides of a right angle rather than the straight-line distance. The formula used to calculate this road distance in nautical miles is:

$$\text{distance} = \left| lat_1 - lat_2 \right| + \left| long_1 - long_2 \right| \left(\cos \left[\frac{lat_1 + lat_2}{2}\right]\right)$$

where latitutde and longitude are in minutes and the argument of the cosine is in degrees.

Road distances were also calculated by using the Lambert Conic Conformal projection. The latitude and longitude coordinates were converted into X, Y coordinates measured in feet for this projection. Using the same assumption, that road distance is the right angle distance, the distance was calculated as:

$$\text{road distance} = \left| X_2 - X_1 \right| + \left| Y_2 - Y_1 \right|$$

Mapping Applications

In addition to the calculation of distances, the created data banks enabled the production of accurate maps on an off-line plotter. The process used to produce the maps is described schematically in Figure 7.

Maps were produced on both flatbed and drum plotters by various plotter software packages. When using the flatbed plotter, actual parcels of land were plotted on an UTM (Universal Transverse Mercator) projection so that they could be overlain on NTS maps.

As previously noted, two methods were used to overcome the problem of having only the centroids of sections and quarter sections on the coordinate file. For the grain hinterland maps (Figure 8), each quarter section was plotted, whereas on the other maps this accuracy was not required.

To produce the grain hinterland maps, each parcel of land described on the legal land description file of the Canadian Wheat Board was broken into quarter sections and coordinates assigned to each quarter section. Each parcel of land was then assigned a symbol corresponding to the delivery point to which grain from that parcel was delivered. These records were then sorted by coordinate to make the plotting more efficient.

The main mapping program, written in FORTRAN, handled the titles, border, and grid systems needed to produce the final plot. It read the data, converted the coordinates to UTM, checked for coordinates outside the map boundary, scaled the coordinates to plotter inches, allocated the correct symbols to coordinates. The program included a subroutine that created plotter instructions to handle all titles, character strings, and punctuation. As many as 26 alphabetics and 14 special characters in strings of up to 50 characters can be drawn to any size.

The output of this program was a set of instructions for the flatbed plotter, which drew the maps on sheets of cronaflex. These, in turn, were overlain in NTS maps. These overlays confirmed the accuracy of CROPP. Maps of publication quality were then produced either photographically or cartographically.

Less detailed maps were produced to show selected farm characteristics. For example, maps were produced to show farm sizes in six categories: 0 to 160 acres, 161 to 320 acres, 321 to 480 acres, 481 to 640 acres, 641 to 800 acres, and over 801 acres, (Figure 9). For these maps, the farm size was read on the legal land description file and the first parcel of land described for an individual farm assigned to the appropriate category. If the description was for a unit larger than a quarter section, the coordinate for the appropriate section was assigned to it. Thus, each farm was assigned the coordinate location of the first parcel of land listed on the legal land description file.

This new record was then put through the same mapping program as the grain hinterland records. Instead of plotting the map on a sheet of clear cronaflex, the plot was made on a cronaflex base map. This was accomplished by lining up the coordinate points of the map border with the corresponding points on the base map.

In addition to the plotting programs for the flatbed plotter, contour maps were produced on the drum plotter by GPCP (General Purpose Contouring Program) developed by CalComp. These maps also used the UTM grid, enabling overlays on NTS maps at a scale of 1:250,000. Information obtained from all of the developed data banks provided input for this program.

GPCP required X, Y and Z coordinates as inputs. The X and Y coordinates were the latitude and longitude locations, which had been transformed into UTM coordinates. The Z value was the value to be mapped according to calculations based on the data banks. For example, cropping indexes were derived from the Canadian Wheat Board records and mapped by this program. The cropping index was used as a measure of the intensity of land use for crops and is defined by the following formula:

$$\frac{\text{acreage in crops/farm}}{\text{total farm acreage}} \times 1000$$

The processing of this map occurred in a number of distinct stages (Figure 7). First, the prime number on the legal land position file was matched with the prime number on the detailed acreage summary file to produce a tape containing coordinates and the acreage summary. Second, the latitude and longitude coordinates were converted to northing and easting values and put on tape in a format acceptable to GPCP. Finally, the Z value was computed and put on the same tape in an acceptable format. This output tape could then be used as input to GPCP.

The GPCP program calculates the contour lines by generating values for the mesh points of a regular grid. These values are estimated from a tangent plane approximated at each data point from the weighted values of a number of neighboring data points.

The overlays produced by GPCP were combined photographically with a base map by lining up the border points on the GPCP output with the corresponding locations on the base map (Figure 10).

The Manitoba Municipal Assessment Rolls were also put through this procedure. The only calculation needed on these data was to combine the parcels of land that were smaller than quarter sections into quarter sections. The assessed value listed for each quarter section was used as the Z value for GPCP.

The assessment rolls contained approximately 12,000 points, which could not be handled by GPCP in one computer run. To overcome this problem, five overlapping overlays were made and joined manually (Figure 11).

CONCLUSION

In summary, it has been shown that CROPP, a coordinate file based on the DLS system, was successfully added to the Canadian Wheat Board records, the Manitoba Crop Insurance Corporation files, and the Manitoba Municipal Assessment Rolls. All of these files contain the legal land descriptions of individual parcels of land, and any data banks that contain these descriptions can be merged with CROPP.

There are several uses for a data bank containing CROPP. Distances can be calculated, individual parcels of land can be plotted, and isolines can be drawn. Latitude and longitude coordinates were used since these are universal and can be readily converted to any type of coordinates desired by the user. In the applications, both the Universal Transverse Mercator and the Lambert Conformal projections have been used.

The maps, which were done by several different programs, were produced both on a flatbed and a drum plotter. They were then reproduced photographically with good results. Thus, the user can obtain satisfactory results using whatever equipment or software that is available.

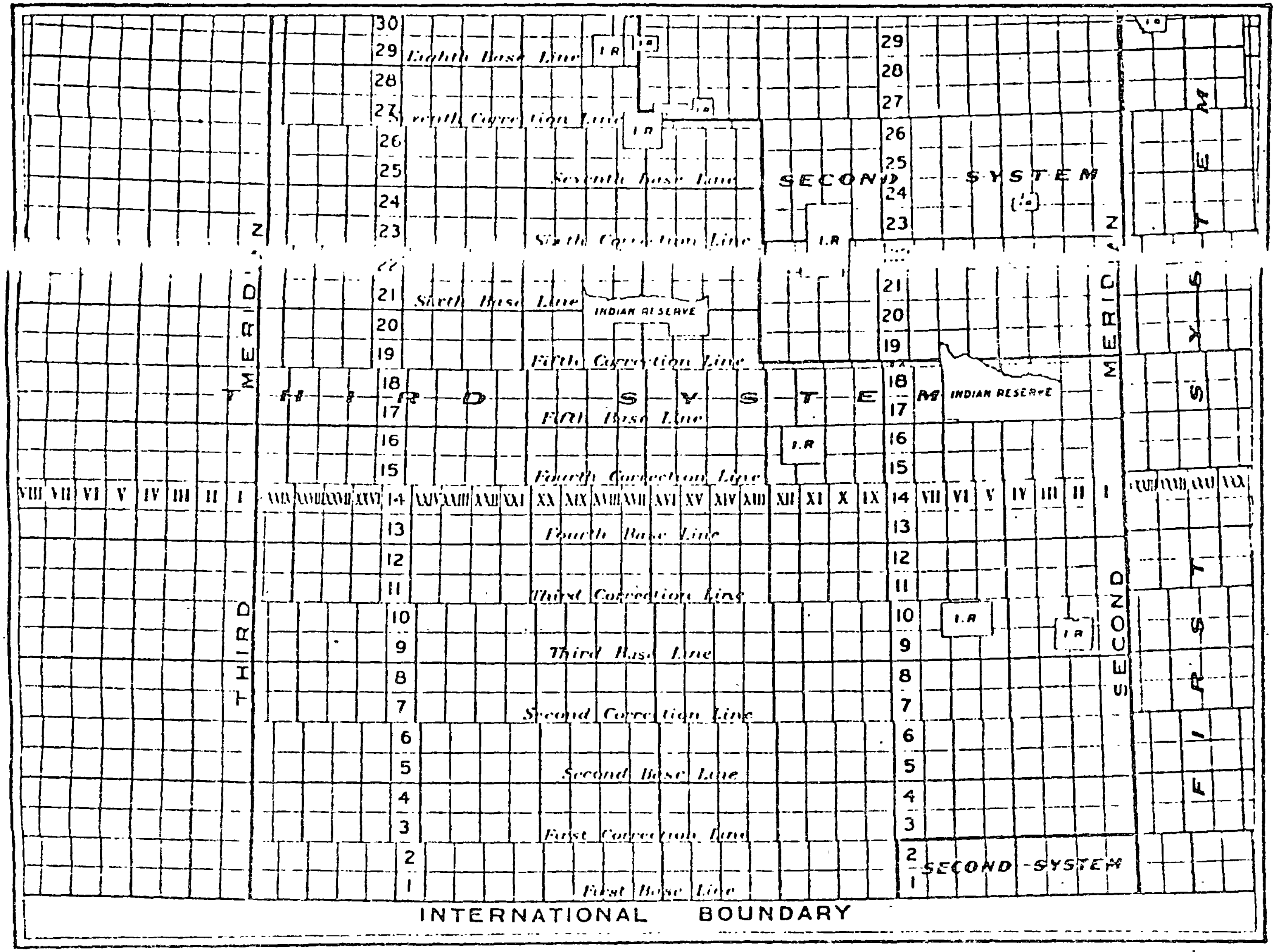
CROPP is not restricted to mapping applications, or to one mapping system. Although the applications listed in this report are social science oriented, the system is also a potential tool for the physical scientist.

SECTION PLAN OF A TOWNSHIP

31	32	33	34	35	36
30	29	28	27	26	25
19	20	21	22	23	24
18	17	16	15	14	13
7	8	9	10	11	12
6	5	4	3	2	1

Figure I

DOMINION LAND SURVEY SYSTEM



SOURCE.

BOND, COURTNEY C. J. SURVEYORS OF CANADA 1867 TO 1967

OTTAWA, CANADIAN INSTITUTE OF SURVEYORS, 1966. P. 21.

Figure 2

STEPS IN THE DEVELOPMENT OF CROPP

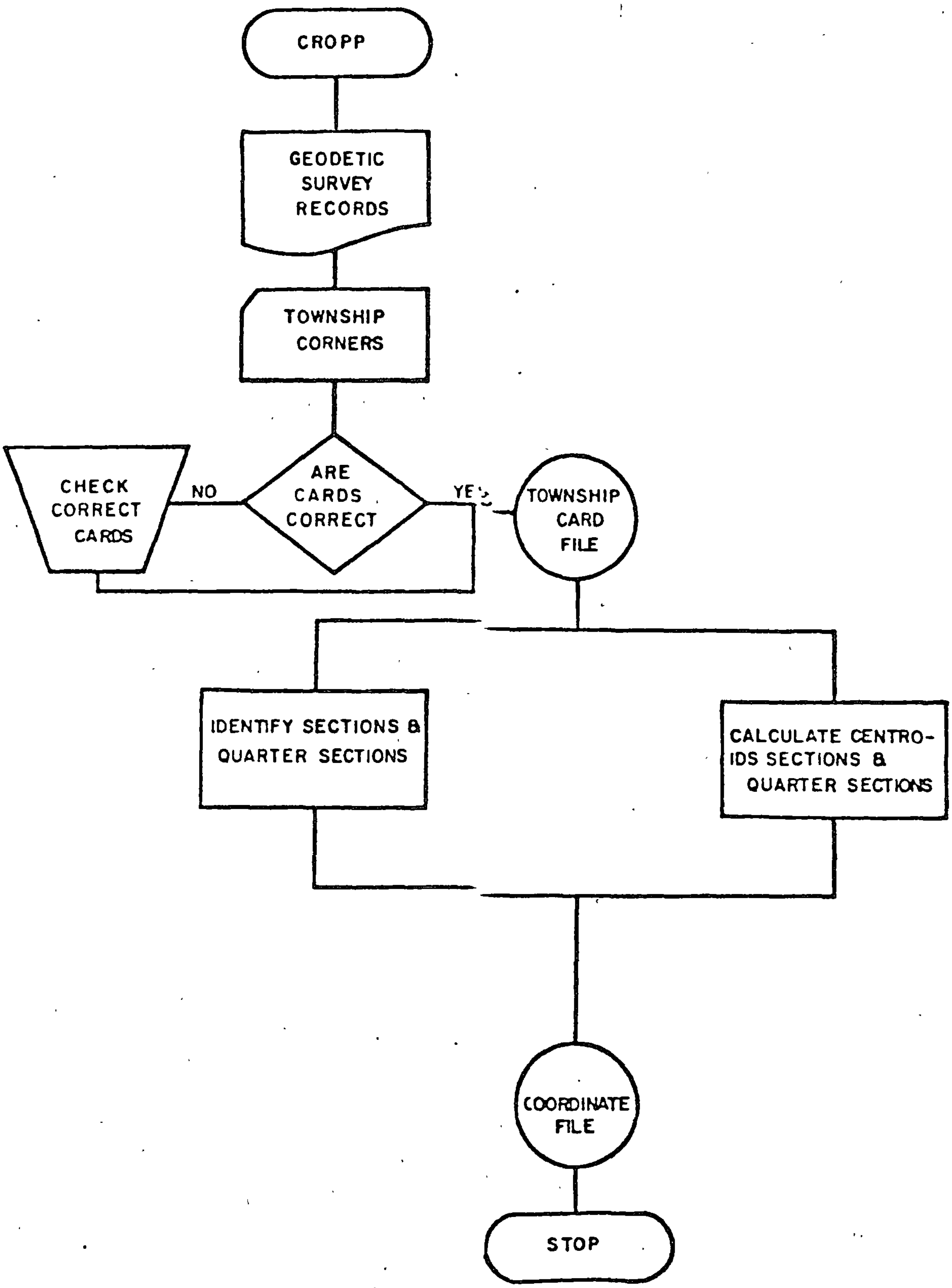


Figure 3.

TOWNSHIP LAYOUT

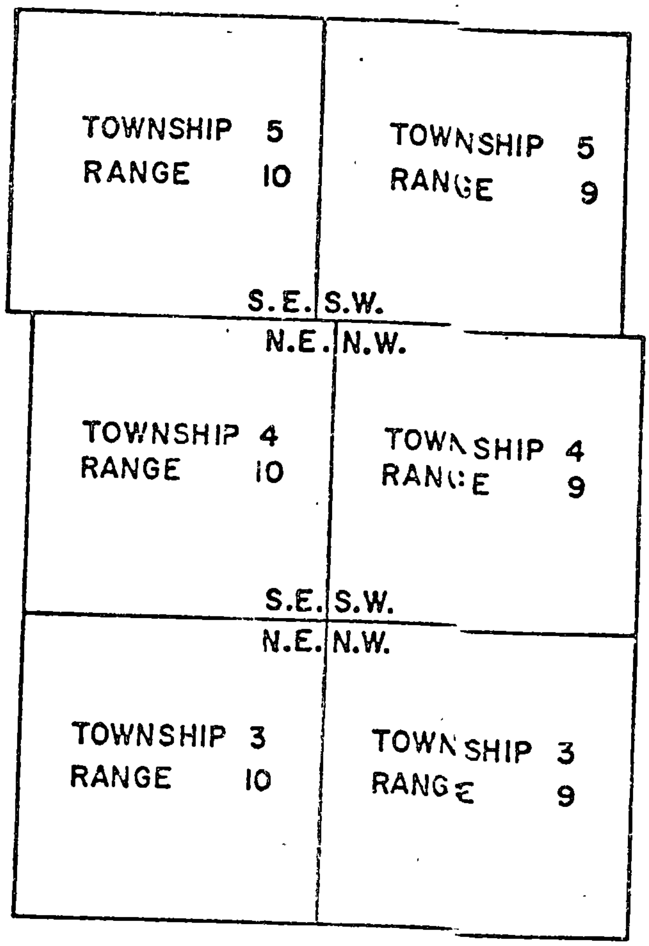


Figure 4.

LOCATION OF CENTROIDS OF A TOWNSHIP

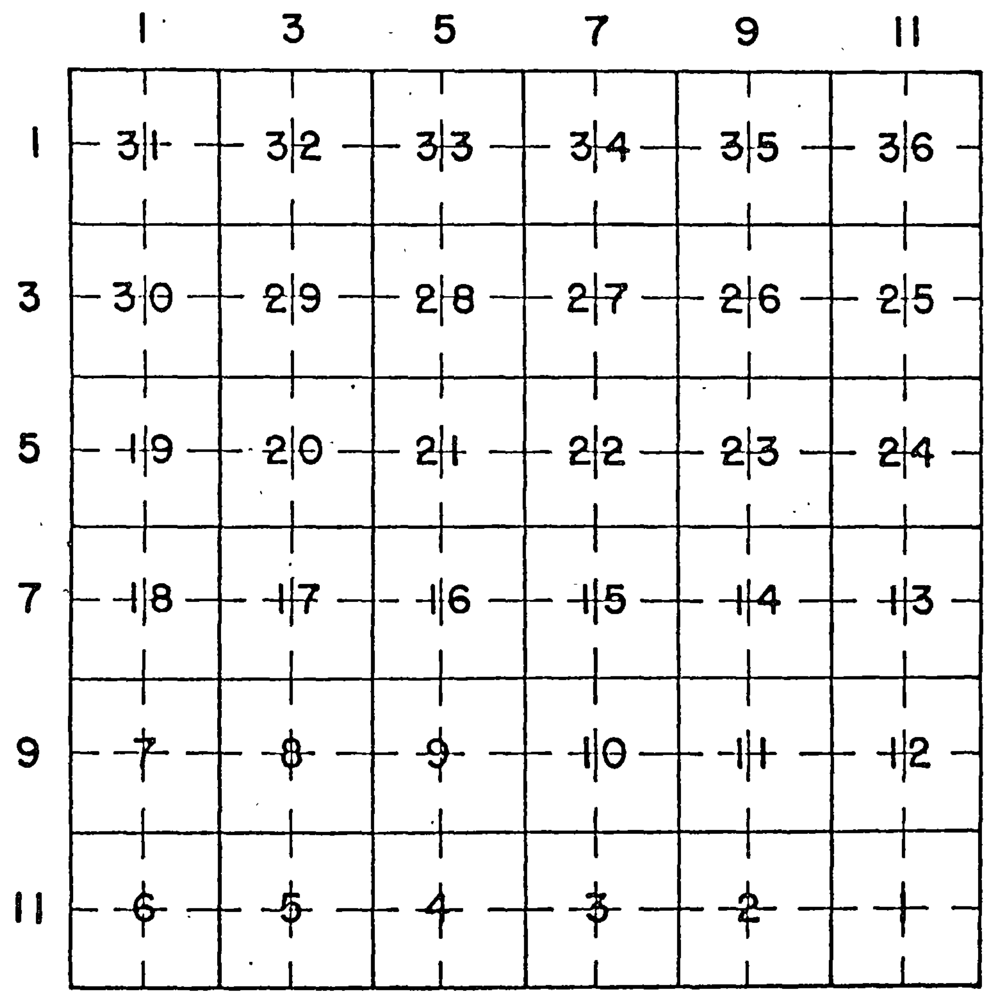


Figure 5

DLS-CROPP MATCH PROGRAM
FLOW CHART

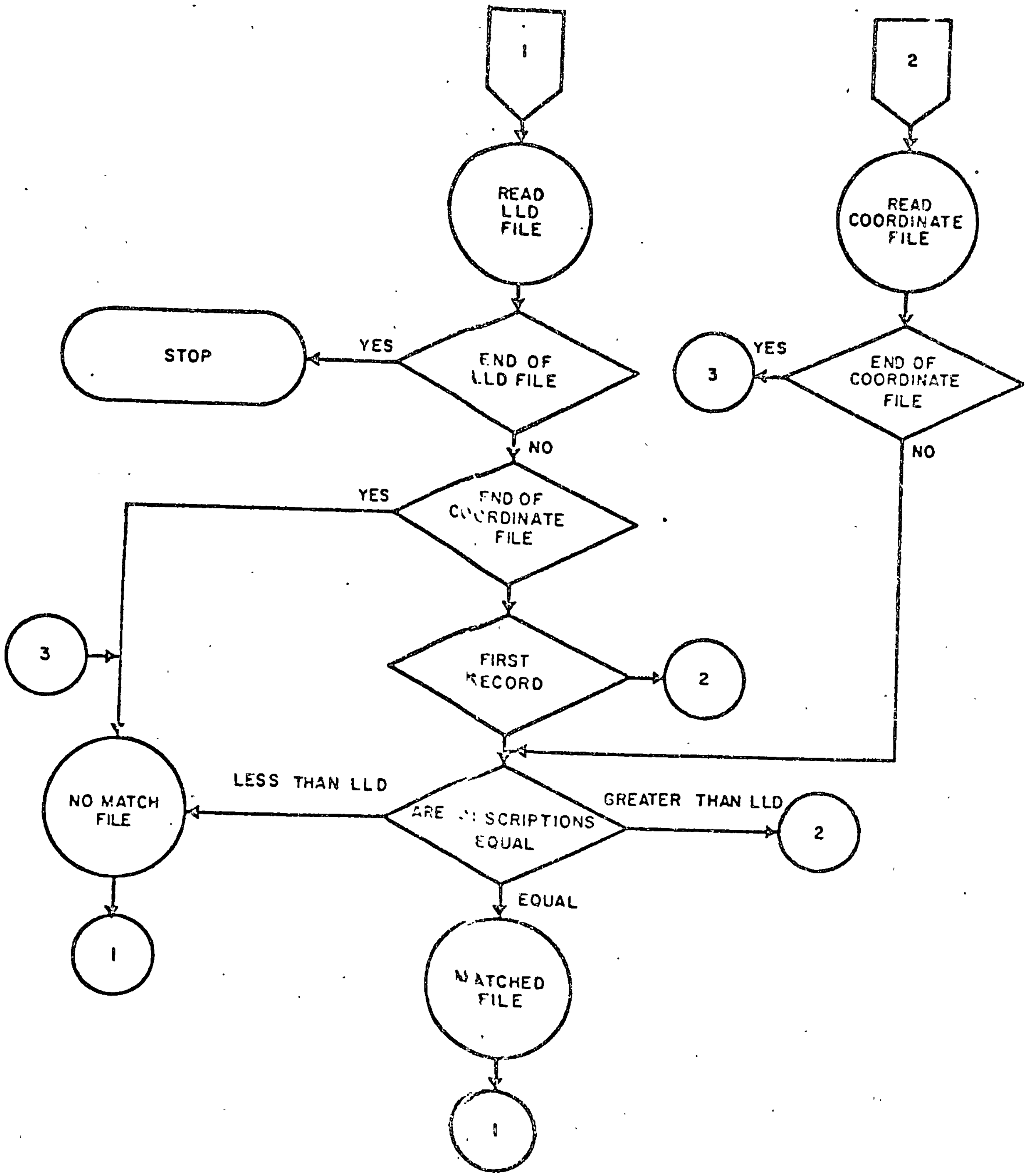
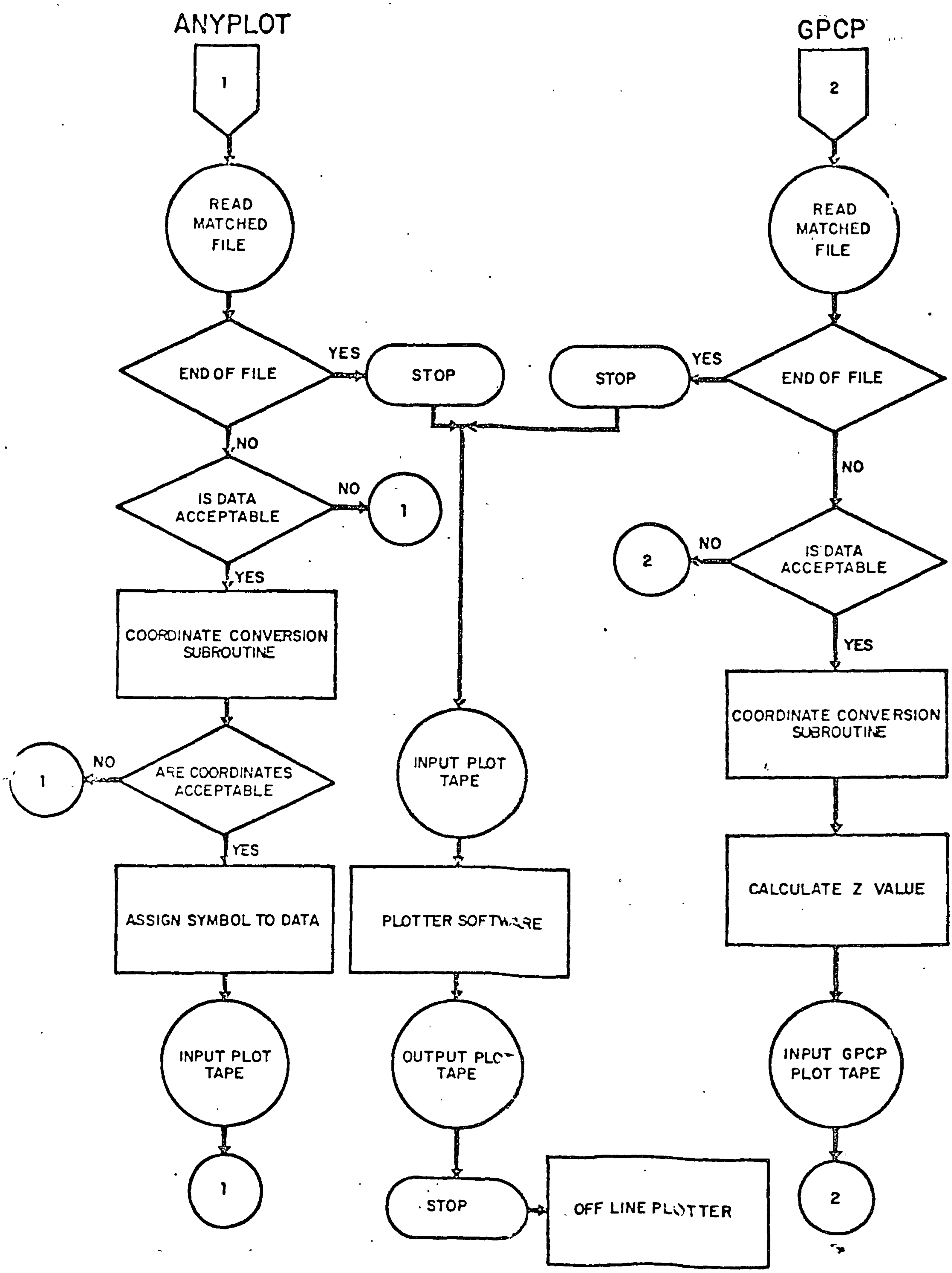


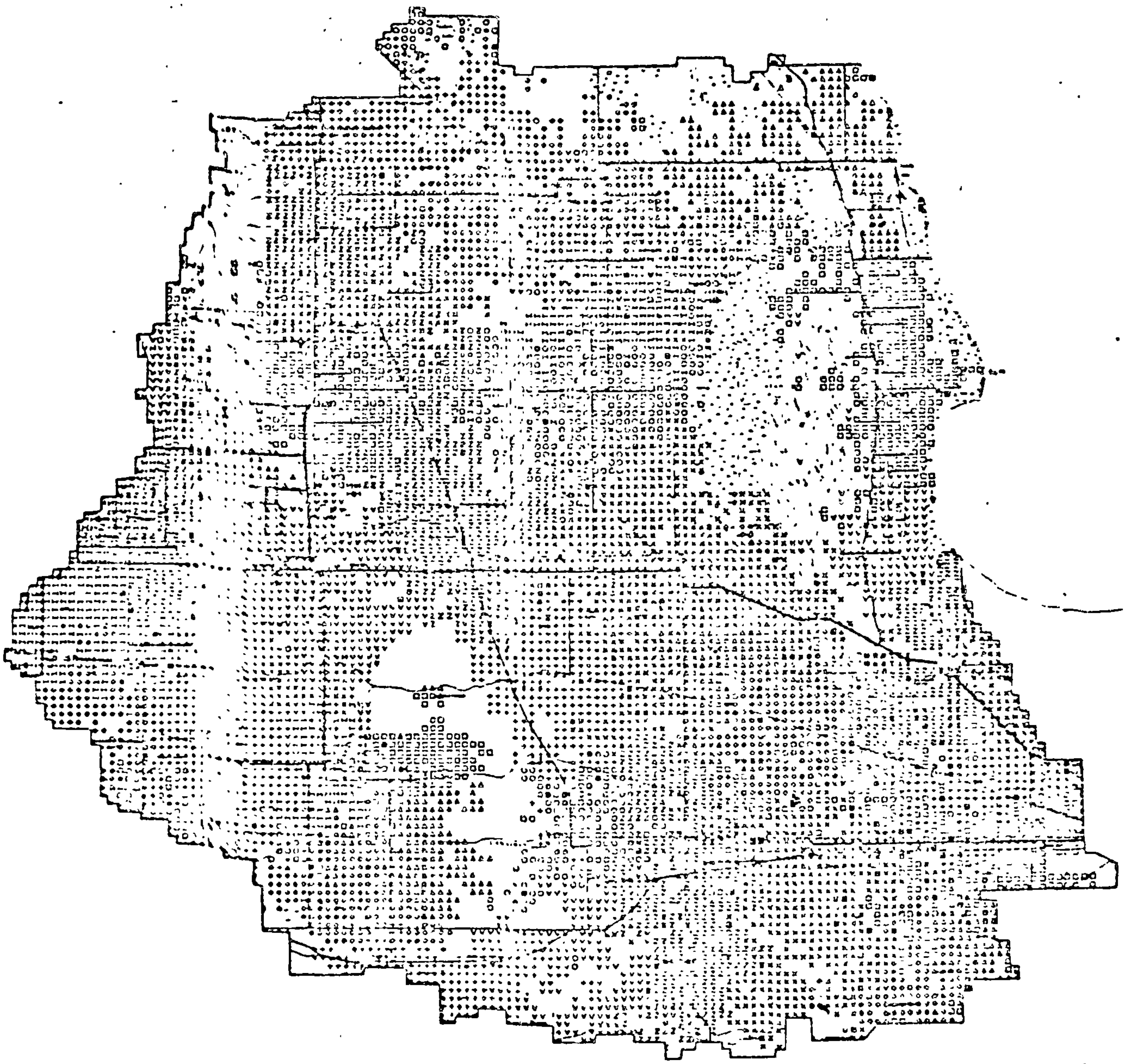
Figure 6



ANYPLOT
1
READ MATCHED FILE
END OF FILE
YES
STOP
NO
IS DATA ACCEPTABLE
NO
1
YES
COORDINATE CONVERSION SUBROUTINE
NO
1
ARE COORDINATES ACCEPTABLE
YES
ASSIGN SYMBOL TO DATA
INPUT PLOT TAPE
1
INPUT PLOT TAPE
PLOTTER SOFTWARE
OUTPUT PLOT TAPE
STOP
OFF LINE PLOTTER
GPCP
2
READ MATCHED FILE
END OF FILE
YES
STOP
NO
IS DATA ACCEPTABLE
NO
2
YES
COORDINATE CONVERSION SUBROUTINE
CALCULATE Z VALUE
INPUT GPCP PLOT TAPE
2

Figure 7

WHITEMUD RIVER WATERSHED

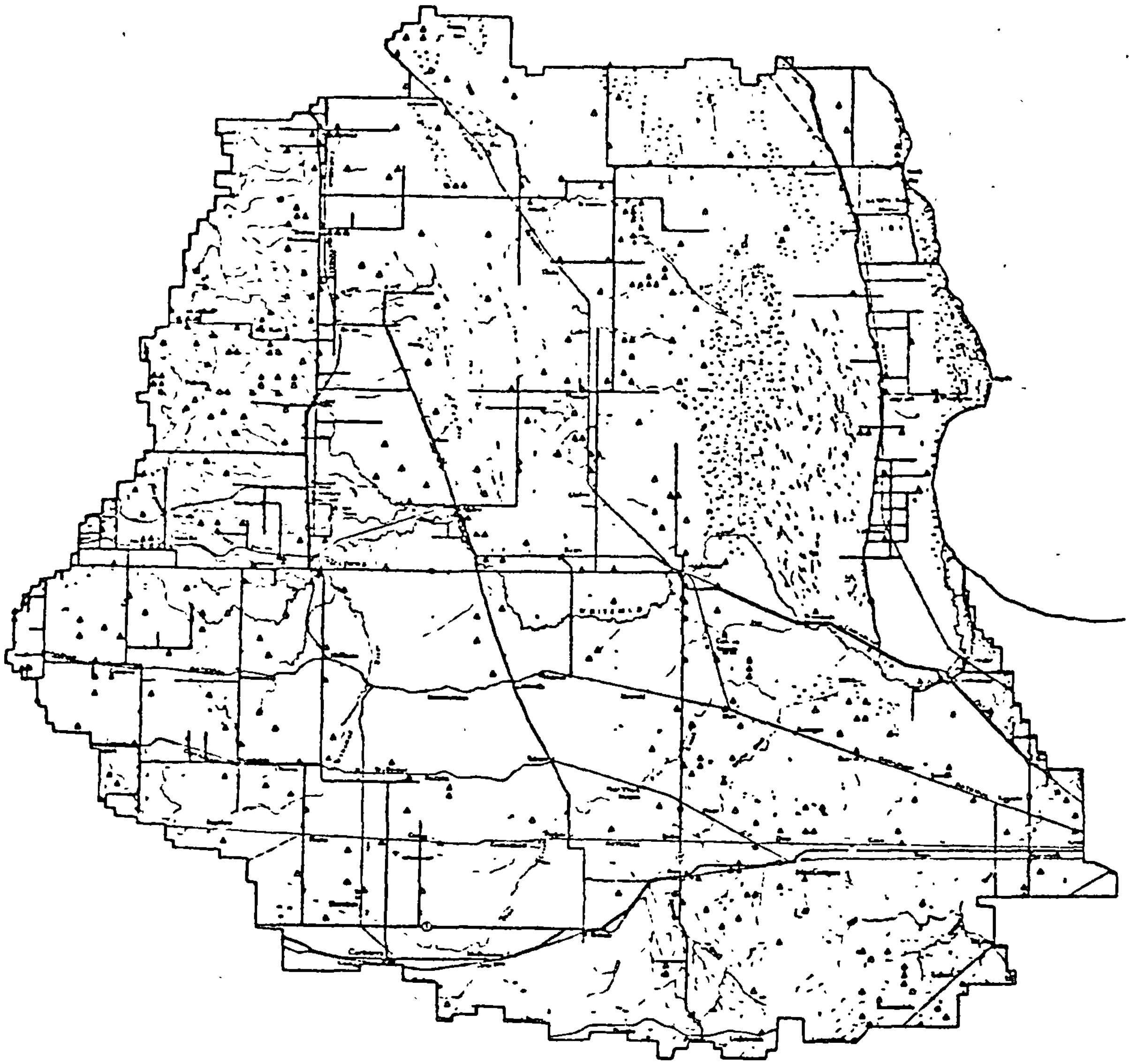


MANITOBA

GRAIN HINTERLANDS

Scale 1:250,000

WHITEMUD RIVER WATERSHED



MANITOBA

FARMS LESS THAN 160 ACRES IN SIZE

Each Symbol Represents A Farm Of Less Than 160 Acres

Scale 1:725,000

**Produced By Lands Directorate, Lands, Forests And Wildlife Service
Department Of The Environment
For Whitemud River Watershed Resource Study,
Manitoba Department Of Mines, Resources And Environmental Management**

October 1971

WHITEMUD RIVER WATERSHED

MANITOBA

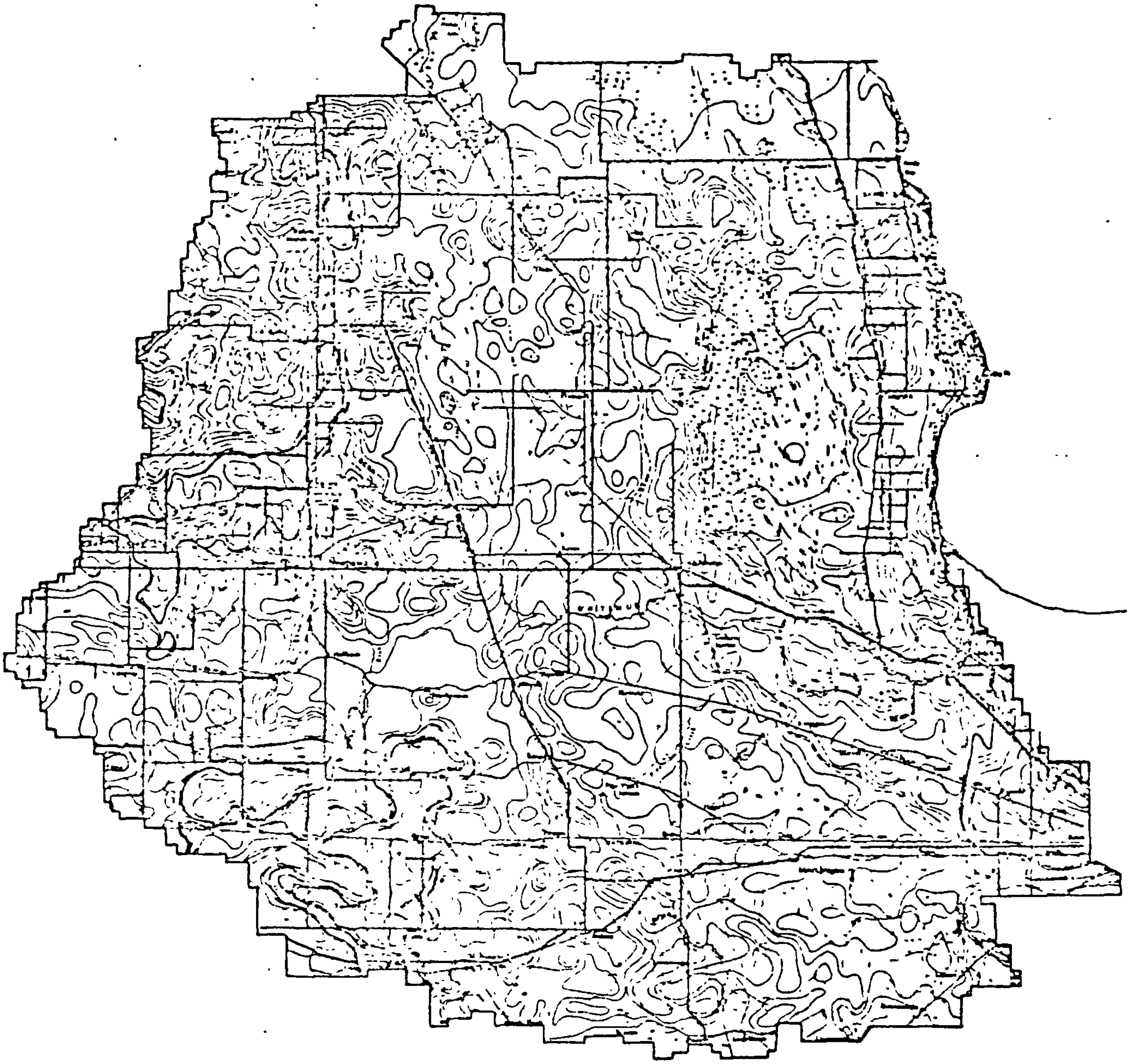
CROPPING INDEX 1970

Scale: 1:250,000

Legend:

Produced By

WHITEMUD RIVER WATERSHED



MANITOBA

ASSESSED VALUE PER QUARTER SECTION

Scale 1:250,000

Laboratory for Computer Graphics and Spatial Analysis
Graduate School of Design Harvard University

Acting Direc...
Eric Teicholz
Assoc. Direc...

THE ODYSSEY MAPPING SYSTEM - AN INTRODUCTION

Submitted to:
Tektronix, Inc.

By: Eric Teicholz
Denis White
Nicholas Chrisman

Date: January 1976

520 Gund Hall 48 Quincy Street Cambridge, Mass 02138 Tel: (617) 495-...

Abstract

The following paper describes the proposed ODYSSEY mapping system to be implemented on the Tektronix 4081 Interactive Graphics System. ODYSSEY is designed to generate thematic (social and environmental) maps from a variety of input data for CRT's and pen-plotter output devices.

It is anticipated that the ODYSSEY mapping system will solve, in a general fashion, many of the existing creation, maintenance, processing and display requirements of a large number of federal, urban, state and regional information planning agencies as well as satisfy land use and thematic mapping needs.

The relationship between thematic and other types of maps in terms of mapping characteristics is illustrated by the chart below. The larger the number, the greater the attribute in question.

ATTRIBUTE \ MAP TYPE	TOPO	THEMATIC		MANAGEMENT INFO SERVICE	UTILITY & ENGINEERING DRAWINGS	SURVEY & PLAT MAPS
		SOCIAL	ENVIRONMENTAL (LAND USE)			
RESOLUTION	9	3	6	5	8	9
DATA BASE						
-COMPLEXITY	5	6	6	7	3 (Except 3-D)	2
-BULK	10	4	7	9	8 (Segmented)	8
CURRENT MARKET 1272-113M (total US Gov't)	USGS, DMA, NOAA, OIL 10 (35M)	BUSINESS UNIV. 7	MINES, EPA, DOT 8	2	8	10 (40
GROWTH POTENTIAL	2	6	7	10	8	7
USERS	10	7	5	1	4	3
PRODUCERS	1	9	6	1	3	7
STATE OF AUTOMATION	3	9	7	1	8	2
DIGITIZING COMPLEXITY	2	6	5	8	7	4
AM'T OF DATA MANIPULATION (application level)	3	8 SUMMARIES OVERLAYS MULTI-VARIATE COMPUTATIONS	6	8	4	3
AM'T OF GRAPHIC MANIPULATION	7	5	4	3	7	7
4081 POTENTIAL	0	10	8	9	6	4
POSSIBLE ENHANCEMENTS		• COLOR • CHART SOFTWARE				

TABLE OF CONTENTS

I. Introduction

The concept of automated cartography has been present for only a dozen years. Within this time span several different approaches to automated mapping have been developed by the federal government, state agencies, universities, utilities and industry. The mapping market has grown within this time frame. In 1972, for example, the U.S. government spent over 113 million dollars on map making - most of it by conventional means.

Some automated mapping (primarily display) systems have received wide distribution and acceptance by the mapping community. Programs distributed by the Census Bureau to support their data bases would be included in this category. At the Laboratory for Computer Graphics and Spatial Analysis at Harvard University, four such mapping display programs and a geographic "preprocessor" called POLYVRT are supported and currently have a world-wide following of over 700 federal and state agencies, universities and commercial companies using this software. Newsletters, technical literature and program support material are distributed to over 5,000 individuals annually.

It has been demonstrated that the concept of automated cartography is now a feasible and cost-beneficial application of computer graphics for a variety of applications. The 'display' part of map production has been well worked out. The incorporation of display technology and software into larger general purpose spatial capture and analysis systems is clearly a more complex problem and one that the Laboratory has been addressing for several years. Several considerations must be evaluated when one attempts to develop a single comprehensive interactive graphic mapping system that is to be used by a variety of groups that have different needs, different management structures and different policies.

The Laboratory has been designing such a mapping system using Tektronix 4014 displays interfaced to a remote host computer. Because of recent discussions with Tektronix, design has been taking place around the new Tektronix 4081 Interactive Graphic System. The new system is to be called ODYSSEY.

II. The ODYSSEY Mapping System

The objective of ODYSSEY is to provide managers and urban researchers with the capability of manipulating and analyzing large volumes of map data to assist them in their planning and management functions. As far as we know, the ODYSSEY system is unique in its ability to catalogue, inventory, correlate, edit and analyze a variety of data sources in a consistent and uniform manner.

The ODYSSEY polygon mapping system is a modular system of computer software that is designed to create, convert, correct and finally chart geographic base files (GBF's). The entire mapping system is controlled by a command language which uses a syntax common to all of the modules but a specific vocabulary designed around each module and appropriate to the task performed within that module. Appropriate data is communicated from module to module in files managed in a directory system.

III. Data Structure

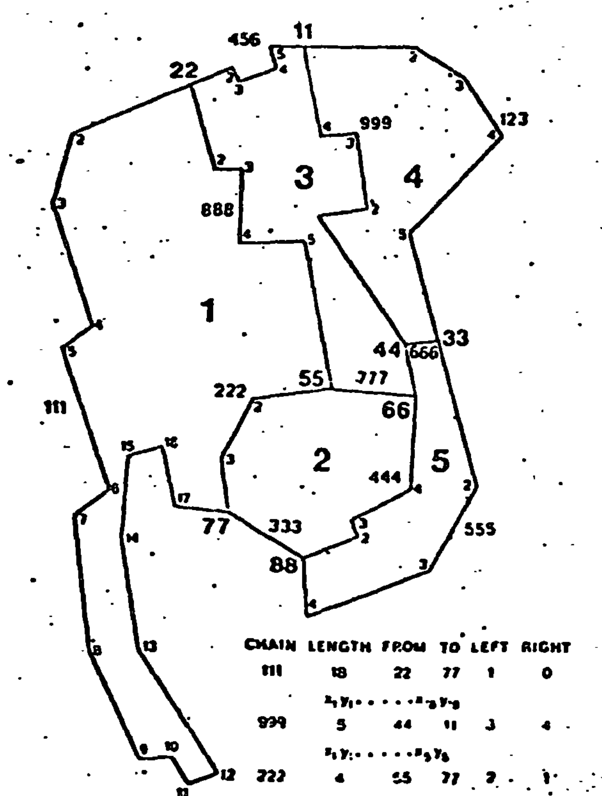
There are presently a variety of data structures being used for cartographic applications. They basically can be divided into cell and polygon structures.

The cell or grid structure involves a uniformly structured spatial sampling filter to be superimposed on the source graphics and the "tagging" of each cell with the particular attribute in question that falls within the grid location - sometimes called a pixel. This is a simple but laborious approach. It has both advantages and disadvantages - the advantages being that it is easy to create, it is easy to do overlay operations and digitizing errors (i.e. incorrect attributes assigned to a particular cell) usually have little impact on the overall logical record. Finally, since it uses a grid or raster scan format, it is usually directly compatible with remote processing data coming from aerial photography or satellite data (LANDSAT for example) which has been put into a raster format using a variety of electronic/optical scanning devices.

Polygon structures (the data structure used by ODYSSEY) have many advantages over grid or cell structures and have some disadvantages but, in general, are vastly superior for both accurate data representation and ease of data base creation. The data structure, upon which the ODYSSEY modules operate and from which the external files are created, embodies a topological relationship of the geographical space represented. That is, in addition to containing information about the names and numerical properties (attributes) of the polygons in the system and coordinate definitions of the boundaries, explicit information is created and maintained about the neighborhood relationships. This allows for verification of the primary adjacency information about a space. Since data creation and edit operations on the source data is such a significant problem at present, the ODYSSEY data structure offers significant advantages over previous systems.

The graphic detail is directly recorded thereby preserving boundary detail as required. More importantly, the data structure accommodates all of the desired processing options while insuring minimum digitizing edit corrections.

Specifically, ODYSSEY uses what we call a "chain" form of polygon structure which consists of an unlimited number of Y-Y coordinates recorded from one intersection (called a node) to another which describes, in turn, a boundary between two polygons and their associated attributes. All graphic detail is captured (and can later be generalized or filtered as resolution require-ments require). Lines are digitized only once and in any desired sequence. The chain structure thus captures maximum detail and allows for a simplified software approach to data capture and subsequent analysis capabilities. This structure is illustrated below:



CHAIN	LENGTH	FROM	TO	LEFT	RIGHT
111	18	22	77	1	0
	$x_1y_1 \cdots x_{18}y_{18}$				
999	5	44	11	3	4
	$x_1y_1 \cdots x_5y_5$				
222	4	55	77	2	1
	$x_1y_1 \cdots x_4y_4$				

Fig. 4. The external representation of the POLYVRT chain-file. Every chain has a "name," the number of inner points (length), the two limiting nodes, the two boundary polygons, and a series of coordinates for the points.

As the diagram above illustrates, chains of x,y coordinates allow the structuring of data as independent line segments. For a low computer overhead, the segments are combined to form polygons which determine bounded regions and allows the complex processing requirements to be met with a simplified directory software approach.

For purposes of reference, several other types of geographic base files are listed below. Some file types are the result of digitizing processes and become input files to the ODYSSEY system. Other file types are specially suited to certain kinds of display processes.

Files having coordinate information only:

POINT - a series of point coordinate pairs, with an identifier (and attribute) for each point

LINE - groups of point coordinates representing continuous lines, with an identifier (and attribute) for each line

REGION - groups of point coordinates representing the complete outlines of regions in the network, with an identifier (and attribute) for each region

Files having adjacency (neighborhood) information only:

NAP - listing of the identifiers of the points around each region (Nodes Around Polygon)

CAN - listing of the identifiers of the Chains Around Nodes

CAP - listing of the identifiers of the Chains Around Polygons

Files having both coordinates and adjacency information:

CHAIN - a set of lines with the line identifier (and attribute), terminal nodes, and adjacent polygons for each line

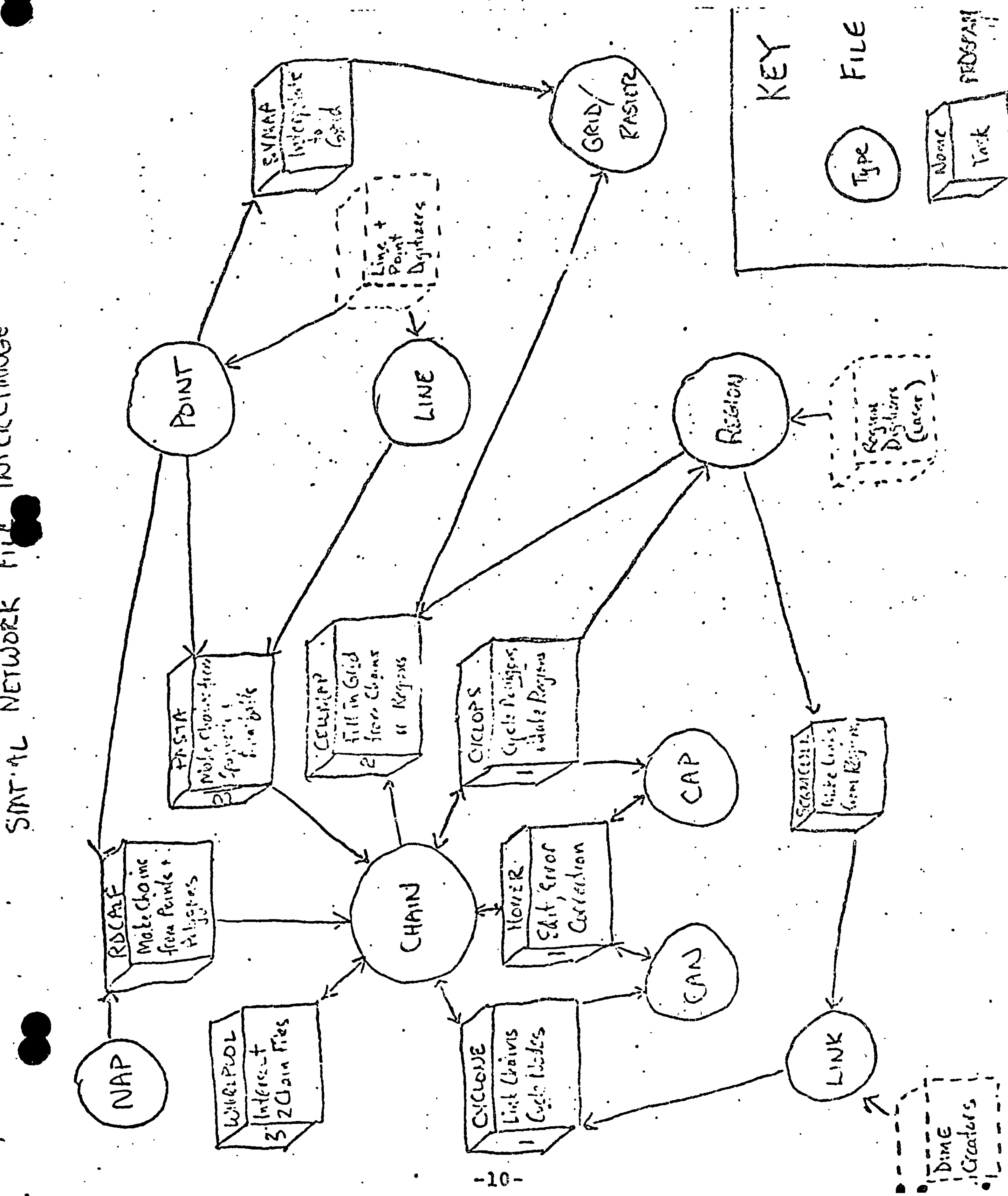
LINK - special case of a CHAIN file where all chains have only two points (these are also known as DIME files)

Special files:

GRID/RASTER - special case of a point file where the coordinates are implicit in the order of the points and the data consists of the value (or attribute) for each grid point or cell.

IV. Components

The ODYSSEY system can be schematically represented by the diagram below:



The modules of ODYSSEY are described as follows:

Zeus	- Supervises the use of the other modules and the file system.
Cyclone	- Aggregates chains from smaller chains, including two point chains (links). Verifies that each node in the network is consistent with all its neighbors by checking the polygons and chains which cycle around it.
Cyclops	- Assembles all chains around a polygon. Verifies that each polygon is consistent with its neighbors by checking the chains which cycle around it. Deletes irrelevant gaps, "sliver errors", in the network.
Homer	- Performs editing of the coordinate description of the file. Inserts, replaces, deletes nodes, chains, polygons.
Polyps	- Displays a geographic file after being processed by Cyclone, Cyclops, and Homer if necessary. Uses selection, shading, and windowing features.
Pasta	- Rectifies line digitized input into chains, corrects errors in the topology.
Scull	- Converts region descriptions into links, deleting duplicate boundaries.
Whirlpool	- Integrates two chain file descriptions by intersecting the networks and producing the minimum chain description for the resulting file.
Cellmap	- Spreads polygon values onto a grid or raster format data base using chain or region files.

The heart of the ODYSSEY system consists of two programs, CYCLONE and CYCLOPS, which construct and verify the topological consistency of the geographic files.

CYCLONE assures that the lines or chains incident at a "node" reference a consistent series of polygons. The CYCLONE processor first verifies the first order integrity of a topologically structured file. That is to say that it assures that the lines or chains incident at a node reference a consistent series of polygons. In a sense, each node is "cycled" (hence the name CYCLONE).

If the program finds that two chains with the same polygon reference meet at the same node, that node is unnecessary and the two chains are automatically joined into one. This process allows files, such as the Census Bureau's DIME file which is based on a single line segment, to be converted to a similar chain structure. This vastly reduces storage and processing requirements. Joining also occurs when a set of larger polygons are created as aggregates of the original file units such as would be the case in overlay operations.

CYCLONE is a local processor which passes chains, as soon as they are completed, out of its storage as soon as it cycles the two nodes at either end. If the input is topologically correct, then there will be no data left after the last input record is processed. If there are, an error in data creation has taken place and the relevant information is passed to the user for correction. Thus CYCLONE is constructed to take the "residue" and to perform error correction processes in which "left over" nodes are made equivalent if they lie within a tolerance distance of one another. Other left over chains, as was noted, that are not within the given tolerance, indicate the likelihood of their being incorrect.

CYCLOPS then assembles each polygon from its constituent chains. As a chain is received by CYCLOPS from the CYCLONE file (or from any other input source), it is placed into the linked lists representing the cycle of chains for each of the two polygons it divides. When the links have been completed for a polygon, a cycled list of chains and the list of bounding coordinates are written out and references to the polygons are immediately deleted from internal storage. When the polygons on each side of a chain have been cycled, then the chain is removed from internal storage.

After all chains have been received, any polygons remaining with gaps in their boundaries are duly noted. Statistics are also computed measuring the lengths of chains and areas of polygons (in the coordinate system used).

Both CYCLONE and CYCLOPS have error correction features which allow them to attempt to correct many of the topological inconsistencies in the file. For correcting errors in the naming of polygons, nodes, or chains or correcting coordinate positions of any of these, there is an interactive editing processor called HOMER. The HOMER editor makes corrections and also handles replacements, insertions and deletions providing annotated displays.

For drawing shaded maps of the corrected, completed polygon system, the display processor POLYPS has an efficient, rapid algorithm relying upon the cycled polygon information from CYCLOPS. Using a variety of shading types and a file of the numeric attributes of polygons, POLYPS produces choropleth maps with headings, keys and labels for display on CRT's or hardcopy produced on pen plotters.

V. The Edit Module

The file correction and editing module of ODYSSEY is called HOMER. While the CYCLONE and CYCLOPS modules generate their own file preparation modules, HOMER acts as an interface between the user and his own files. The edit operations relate to two categories involving the data structure: topology and coordinates. The topologic data consists of the linkages between nodes, chains and polygons. Correction of this data requires a knowledge of the identifier schemes used for specific files. The process is facilitated using interactive techniques such as pointing. Coordinate input and correction is also accomplished with graphic editing techniques.

HOMER, like other critical elements of ODYSSEY, utilizes the local edit capabilities of the 4081 such as sequential file handling to minimize storage space in core memory. At a given point in an edit session, for example, HOMER will have an input file partially read, an output file partially written and a number of chains in active storage. Active storage consists of two sets of chains - the active buffer and the reserved chains. Reserved chains are removed from and reinserted in the active buffer by explicit commands from the user. It is not necessary to have a large number of chains in core just as it is not necessary to have a large amount of text in a text editor because both editors can rely on the sequential nature of a file to structure the edit cycle.

The command dialect of HOMER is defined in a general manner using the techniques of formal language representation and modular table-driven implementation. General features of these dialects include recognition of short forms of commands, ability to handle expressions with arithmetic and logical operations, and consistent formats and definitions from program to program.

VI. Display Module

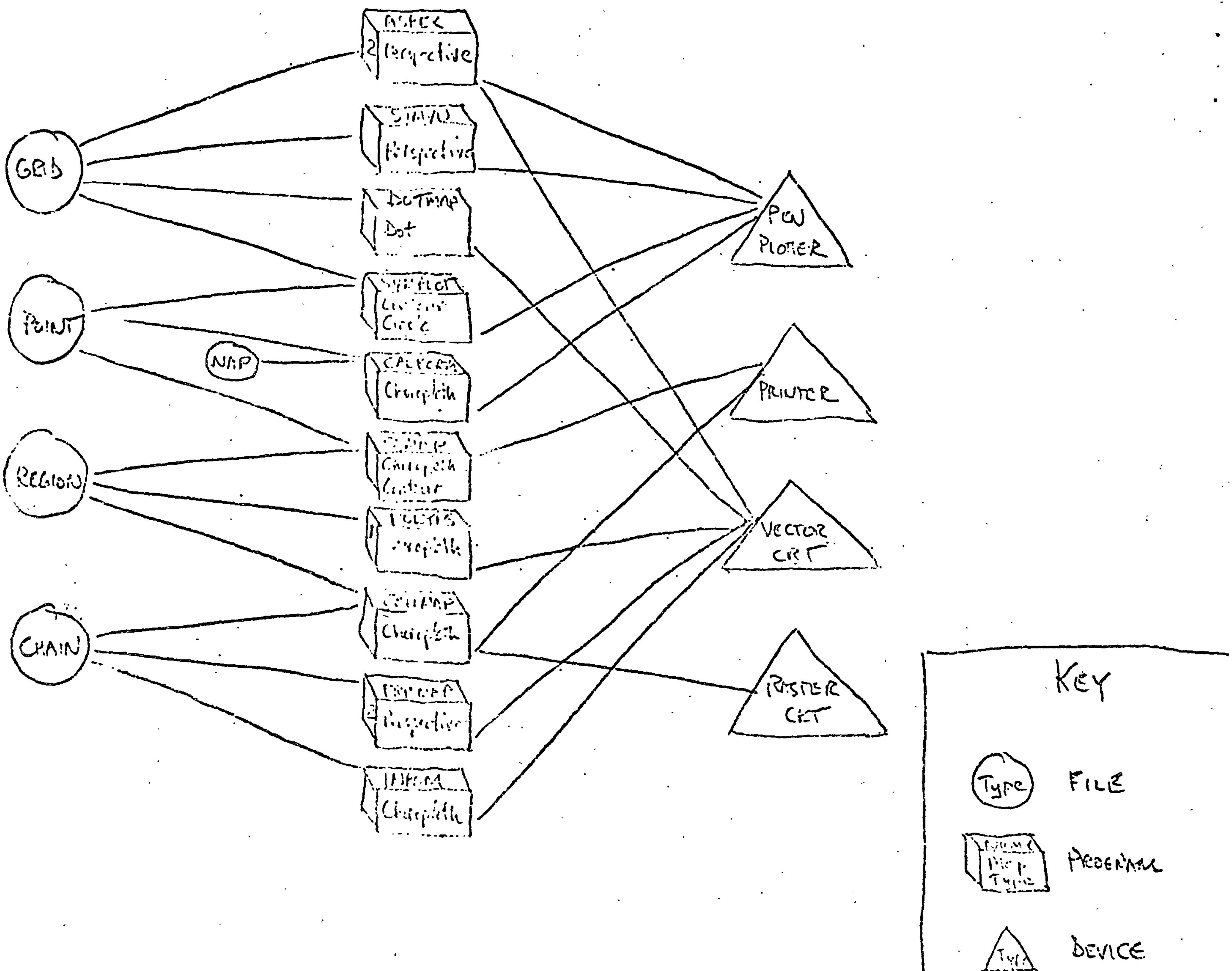
The final output display generated from POLYPS will result in shaded choropleth (conformant) maps. The command language is extremely well designed from a human engineering point of view and allows a user to:

1. Input create, edit or query files
2. Select, window, scale and position polygons
3. Define symbolism, keys, shading densities and types and other 'cosmetic' operations
4. Determine CRT or plotter display parameter instructions.

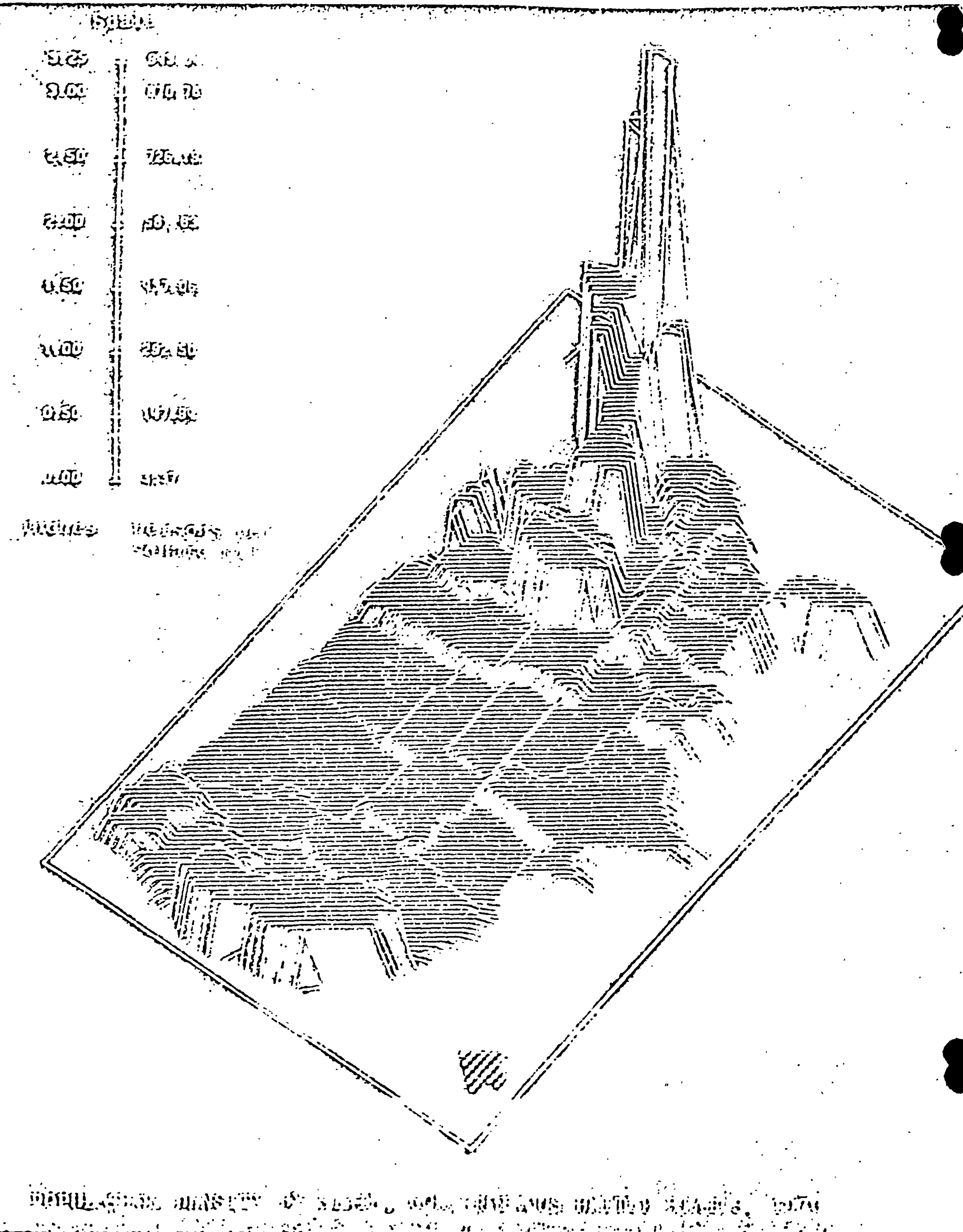
Other types of displays will eventually be added to the ODYSSEY mapping system. These include line printer maps of contours and conformant types, perspective maps, dot maps and three-dimensional surface maps. All the display modules will be based on existing Laboratory display packages.

The ODYSSEY file display network is illustrated below:

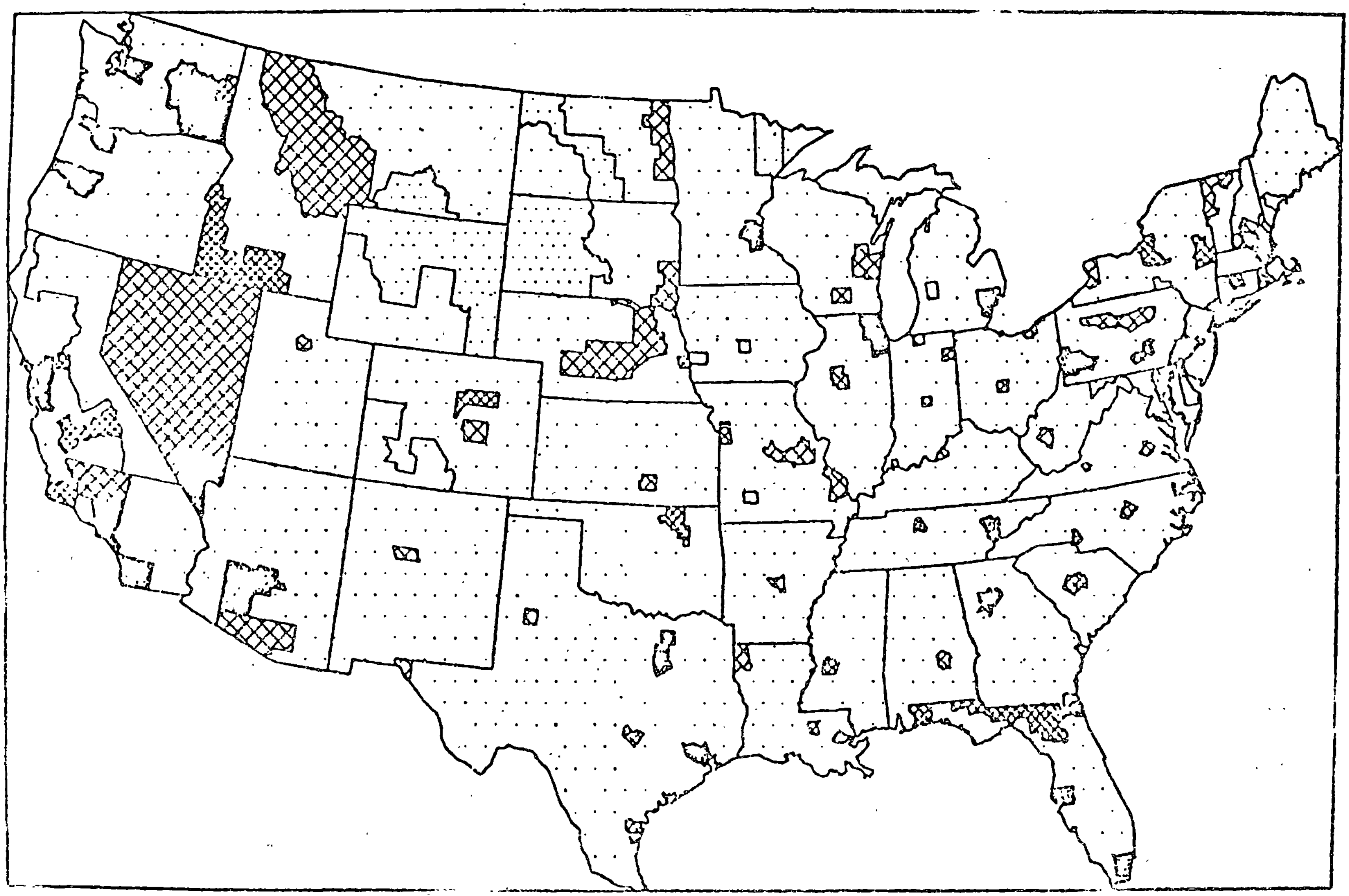
SPATIAL NETWORK FILE DISPLAY

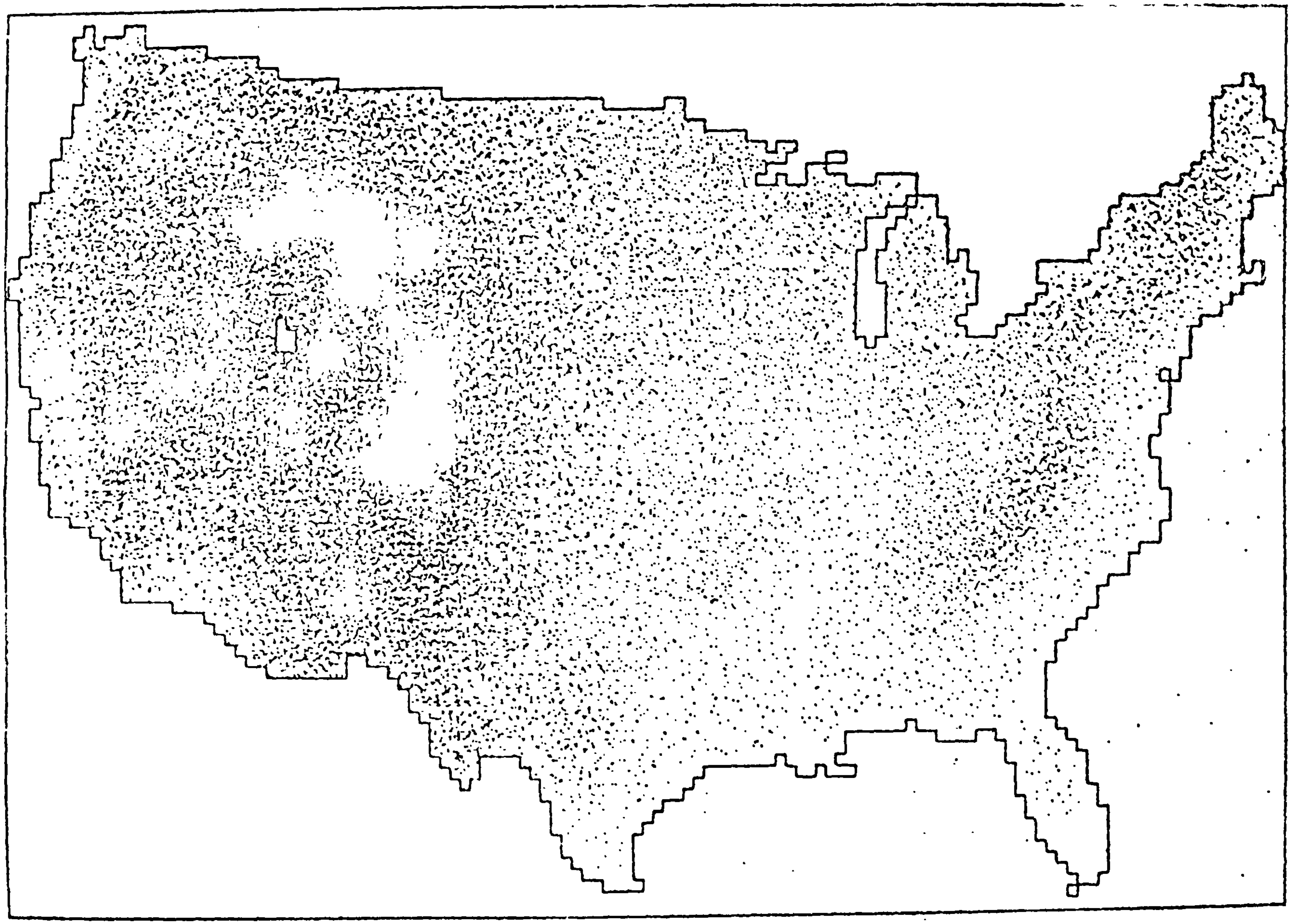


Examples of existing Laboratory display modules appear below:



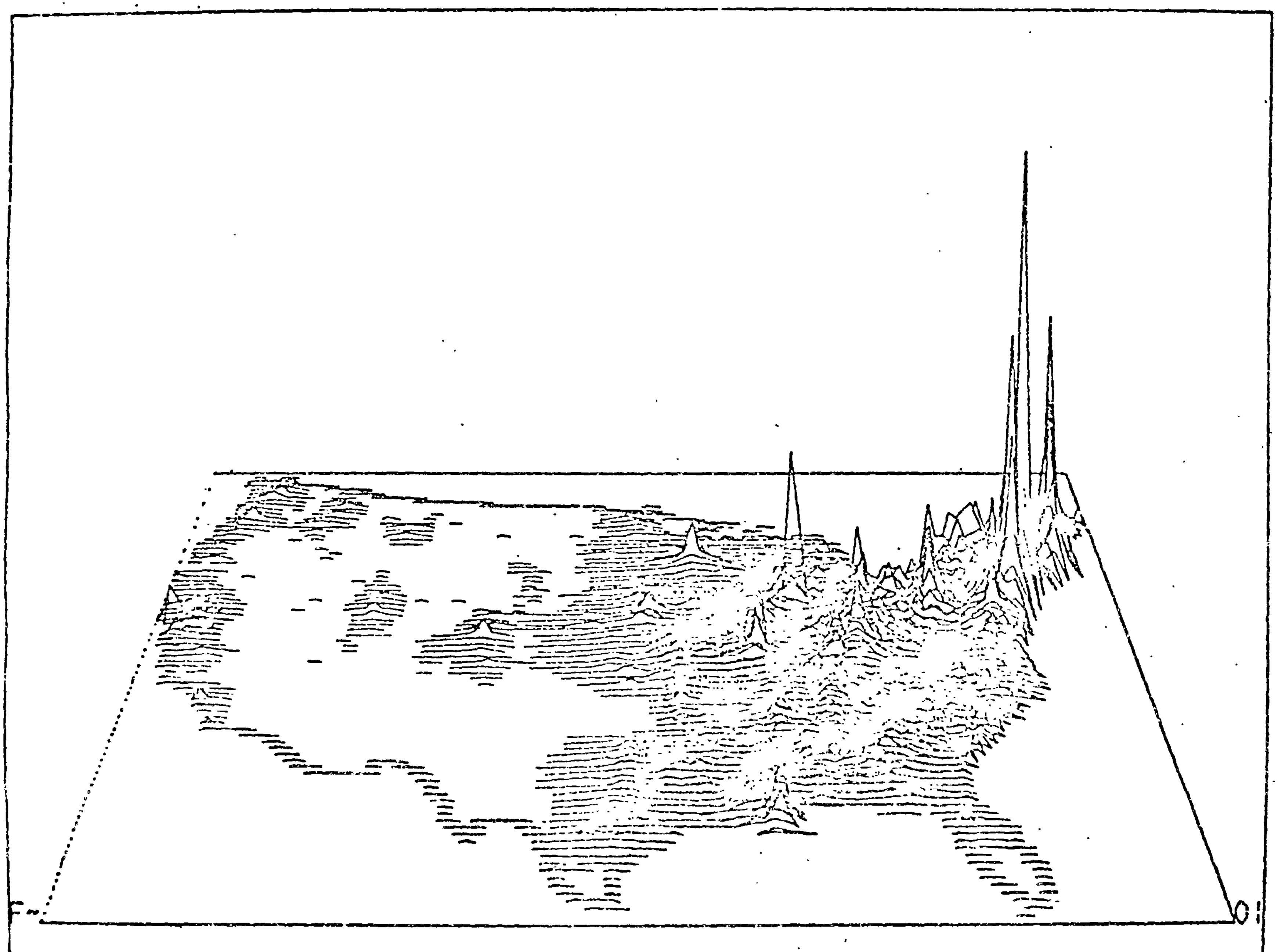
HEATING FUELS- ELECTRIC

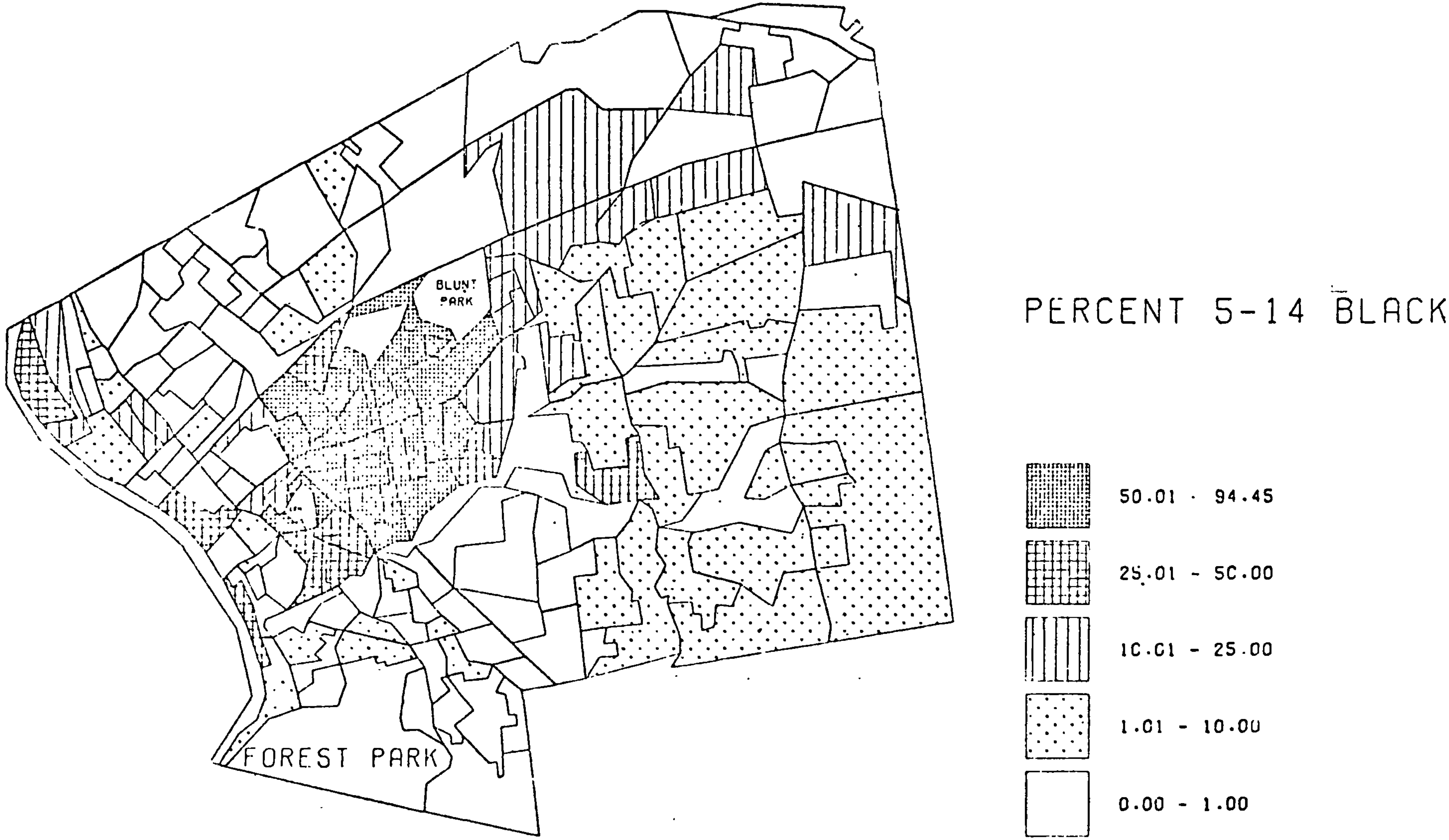




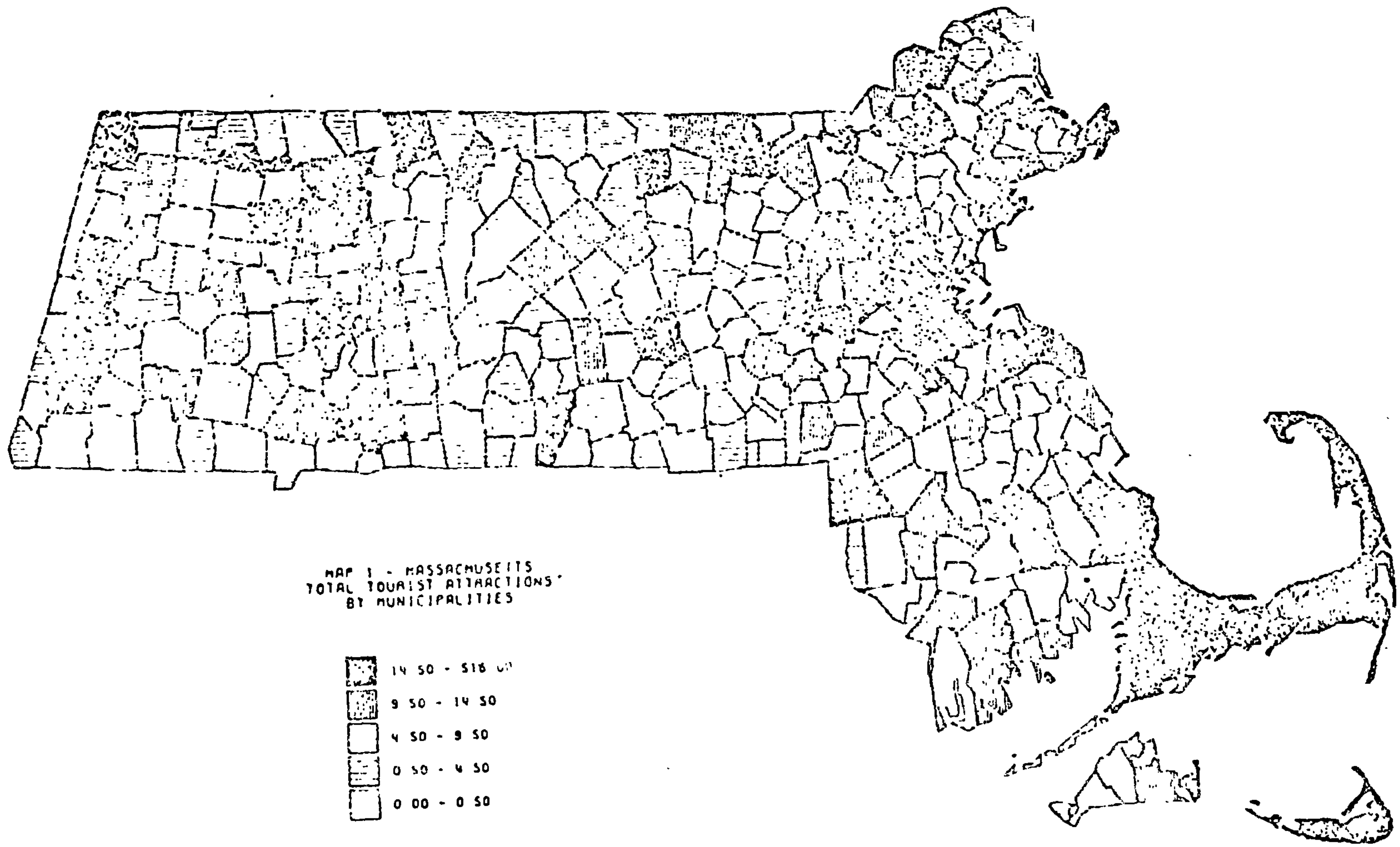
UNITED STATES LANDFORMS: 1 DOT = 450 FEET

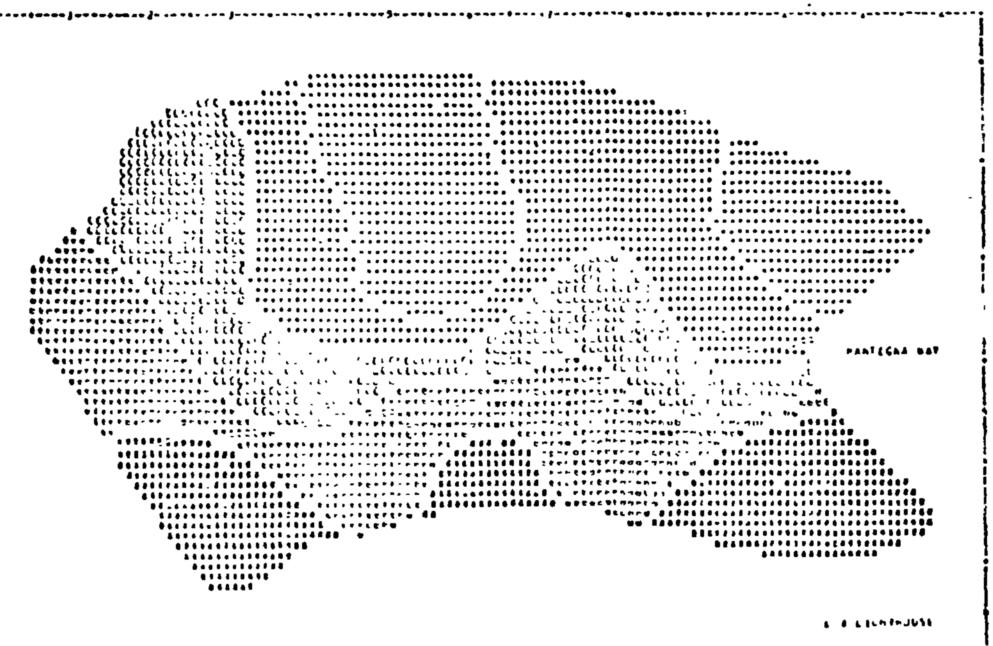
UNITED STATES POPULATION, 1890
UNITED STATES POPULATION, 1850

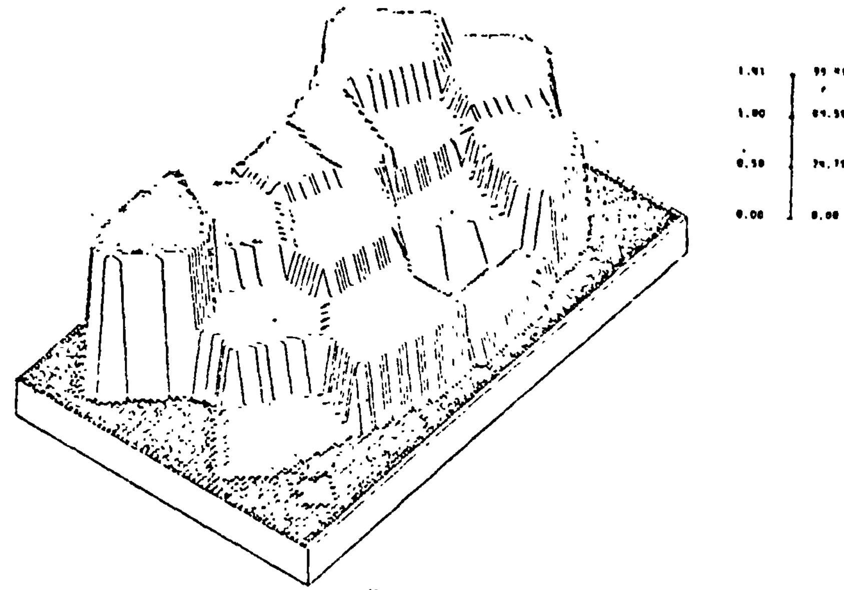




SPRINGFIELD BY CENSUS BLOCK GROUP, 1970


MAP 1 - MASSACHUSETTS
TOTAL TOURIST ATTRACTIONS
BY MUNICIPALITIES
14 50 - 518
9 50 - 14 50
4 50 - 9 50
0 50 - 4 50
0 00 - 0 50



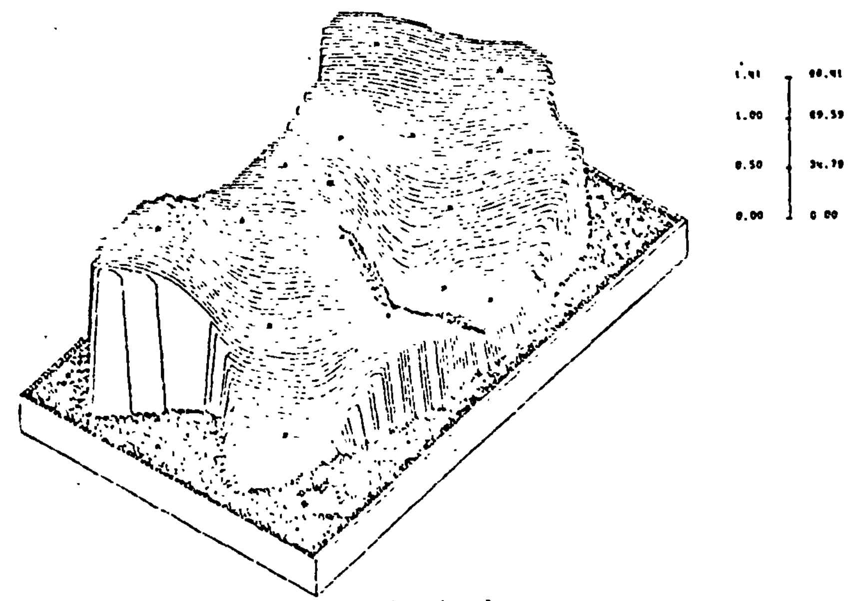


Example 6-2

FIGURE A: MANTEGNA BAY - CONFORMANT

AZIMUTH = 231 ALTITUDE = 45
*WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 09/25/70



Example 6-3

FIGURE B: MANTEGNA BAY - CONTOUR

AZIMUTH = 231 ALTITUDE = 45
*WIDTH = 6.00 *HEIGHT = 2.00

* BEFORE FORESHORTENING 09/25/70

File Input, Creation and Querying

- INC (INput Chains)
- INP (INput Polygons)
- VAL (Input VALues)
- FN (File Name)
- RV (Read Values)
- WV (Write Values)
- LIN (define LINe legends)
- CVRT (input Chains from the POLYVRT program)
- REN (REName chains and/or polygons)
- INFO (list INFOrmation on chains, polygons, lines)

Value Level and Symbolism Definition

- NL (Number of value Levels)
- LVL (define value LeVeLs)
- SHD (define SHaDing for levels)
- FAS (FActor Shading density)

Termination

- EXIT (EXIT from INPOM to monitor level)

Windowing, Scaling and Polygon Selection

- WORG (Window ORiGin location)
- WSIZ (Window SIZe specification)
- MW (Move Window across map)
- PMM (Polygon selection by Minima and Maximum extents)
- DMM (Data coordinate Min-Max selection of polygons)
- FDW (Fill Data Window with partial chains)
- FSW (Fill entire Screen Window with polygons and chains)
- FA (FActor for expanding or shrinking map)
- XFM (Transform point coordinates with respect to a location)

Graphic Manipulation Instructions

- OL (to specify OutLine or shaded maps as output)
- DET (highest DETail level to be drawn)
- DL (Draw Line legends)
- DWO (Draw Window Outline)
- PLT (PLoTing mode for Tektronix 4014 display only)
- MAP (draw a MAP)

FIGURE 2
Command File for INPOM

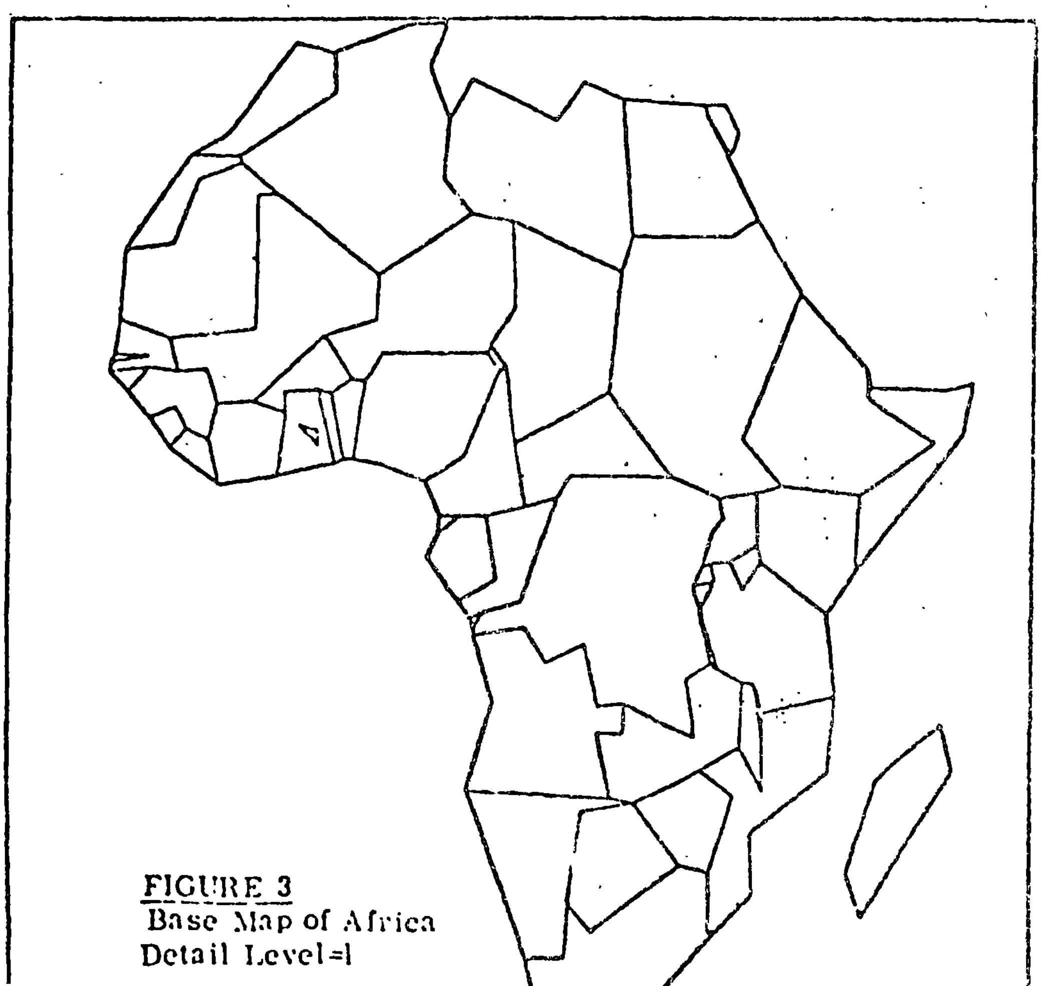


FIGURE 3
Base Map of Africa
Detail Level=1

VII. Hardware and System Considerations

For any geographically based mapping or information system, some input device (a digitizer), storage device (disk, tape, etc.), processing capability (a mini-computer with a communications interface), and output device (hardcopy from a CRT and/or a plotter) must be present.

Each of the components can be great or small depending upon the capabilities of the host processor. We currently believe that most of the ODYSSEY operations can be performed locally using a minimal configuration consisting of a 4081 processor, a large data tablet, a plotter, and minimal floppy disk storage. ODYSSEY would make maximum use of the current EDITOR and BROWSER capabilities as well as the linked list data structure. A 64K mini would be required to perform the local ODYSSEY operations.

Both CYCLONE and CYCLOPS are, as was mentioned, local processors in the sense that they operate on only a portion of the geographic space at any one time, the portion including a particular node or polygon and its neighbors. This allows extremely large files to be processed in very modest amounts of storage, since the input is spatially contiguous. Because of the use of linked list storage structures, there is a low degree of computational complexity. Processing time is merely a linear function of the bulk of the data.

Finally, the following chart represents the current status of the major modules that make up ODYSSEY mapping system. Modules that already exist (CYCLONE, PASTA, SCULL and CELLMAP) would have to be converted to ODYSSEY. Others would be directly programmed on the 4081.

MODULE	DESIGN	PROGRAM
CYCLONE	100%	90%
CYCLOPS	90%	10%
HOMER	40%	0%
POLYPS	20%	20%
PASTA	80%	10%
SCULL	90%	80%
WHIRLPOOL	10%	0%
CELLMAP	100%	80%
ZEUS (File Manager & Command Language)	20%	0%

GRDSR (The geographically referenced data storage and retrieval system)

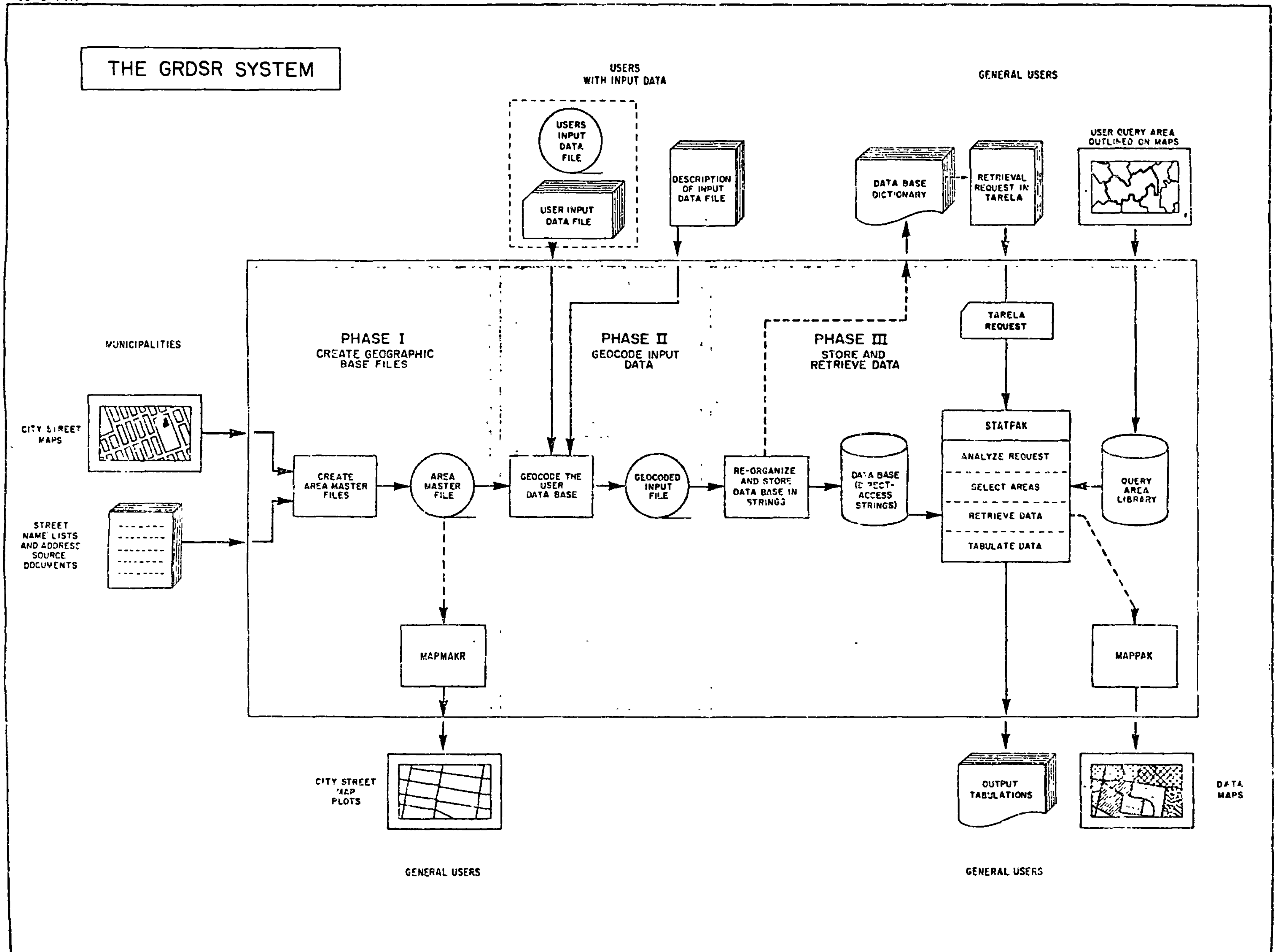
A new method of assembling statistical information by user-specified areas

AN INTRODUCTION

Published by Authority
of the Minister of Industry, Trade and Commerce

June 1972

FIGURE-VIII



THE GRDSR SYSTEM
USERS WITH INPUT DATA
GENERAL USERS
USERS INPUT DATA FILE
USER INPUT DATA FILE
DESCRIPTION OF INPUT DATA FILE
DATA BASE DICTIONARY
RETRIEVAL REQUEST IN TARELA
USER QUERY AREA OUTLINED ON MAPS
MUNICIPALITIES
PHASE I
CREATE GEOGRAPHIC BASE FILES
PHASE II
GEOCODE INPUT DATA
PHASE III
STORE AND RETRIEVE DATA
TARELA REQUEST
CITY STREET MAPS
STREET NAME LISTS AND ADDRESS SOURCE DOCUMENTS
CREATE AREA MASTER FILES
AREA MASTER FILE
GEOCODE THE USER DATA BASE
GEOCODED INPUT FILE
RE-ORGANIZE AND STORE DATA BASE IN STRINGS
DATA BASE (DIRECT-ACCESS STRINGS)
STATPAK
ANALYZE REQUEST
SELECT AREAS
RETRIEVE DATA
TABULATE DATA
QUERY AREA LIBRARY
MAPMAKR
MAPPAK
CITY STREET MAP PLOTS
OUTPUT TABULATIONS
DATA MAPS
GENERAL USERS
GENERAL USERS

CONTENTS

INTRODUCTION

What is GRDSR?

A unique and flexible system now makes it possible, for the first time, to provide information by user-specified areas in Canada's larger urban centres. Fully computerized, the GRDSR (Geographically Referenced Data Storage and Retrieval) system is the outcome of five years' research by Statistics Canada into solving the many problems associated with the storage and retrieval of statistics about small areas.

Through GRDSR, statistical information can now be quickly and inexpensively obtained about retrieval areas that range in size from a few city blocks to an entire urban centre.

Retrieval is made possible through a technique called geocoding, whereby urban areas are divided into many small building blocks or micro-areas. The blocks must be small enough that they can be assembled to approximate most retrieval areas required by users. Each building block is assigned a unique identifying coordinate number which, in turn, allows files of households, persons, or events to be coded to appropriate building blocks in the city area. The appropriate building block is usually the place of residence or location where the event occurred. At this point, the files are said to be geocoded. When an interested user needs information from a geocoded file, he outlines his area of interest (or "query area") on a map and makes a request. GRDSR then identifies all the building blocks contained within his query area and, using the corresponding coordinates, automatically retrieves all data belonging to the blocks. The statistics are then tabulated in a convenient report.

Background to GRDSR development

The need for concise, timely statistics is constantly pressing in many sectors of the economy. Diverse planning and decision-making efforts are often frustrated by a lack of relevant and timely data; the socio-economic benefits of more fully informed decisions may, as a result, be diluted or lost. Today, it is obvious that the pressure for diverse and specialized statistical information cannot help but increase. The comprehensive and thorough use of data already collected is now, therefore, more relevant than ever.

At Statistics Canada, this pressure is evident not only in the mounting volume of special data requests but by their changing nature. The trend consistently points to the need for flexible information systems fully capable of retrieving data on a specialized, often one-time basis. The basic requirement is, essentially, for an integrated information service — not just a data collection facility.

The development of GRDSR thus focuses upon an important trend, and the nature of this trend is clear: user requirements will increase, in terms of the amount of data required, types of aggregation and manipulation available, ease of retrieval, the format of the final statistical product and, of course, the response time. The evolution of the GRDSR system is now at the point where each of these requirements is substantially met.

Nature of the Problem

The gathering of small-area data has presented a difficult problem for some time. Urban planners, municipal agencies, school administrations and governments each impose different zoning patterns or jurisdictions over settled areas of land. Many agencies maintain records and use their own jurisdictions to collect and identify statistical information. At some point these records may attract general interest. But problems arise when outside groups try to use this information, because their requirement is for facts related to different area breakdowns.

Today, special-interest areas such as marketing zones, census tracts, school districts and land-use areas are in everyday use in major cities. However, these areas usually overlap and have little in common but the land area they reference. Thus, it is difficult to relate information from one source to outside areas of interest (see Figure I, page 2.)

In the past, when the sole means of disseminating statistics from the census was through published volumes, the statistics had to be summarized in terms of enumeration areas, census tracts or other standard areas. The standard census areas did not, however, coincide with many query areas for which data were required. Consequently, the requirements of many census data users could either not be met, or met only with great difficulty, at considerable cost and with considerable delay.

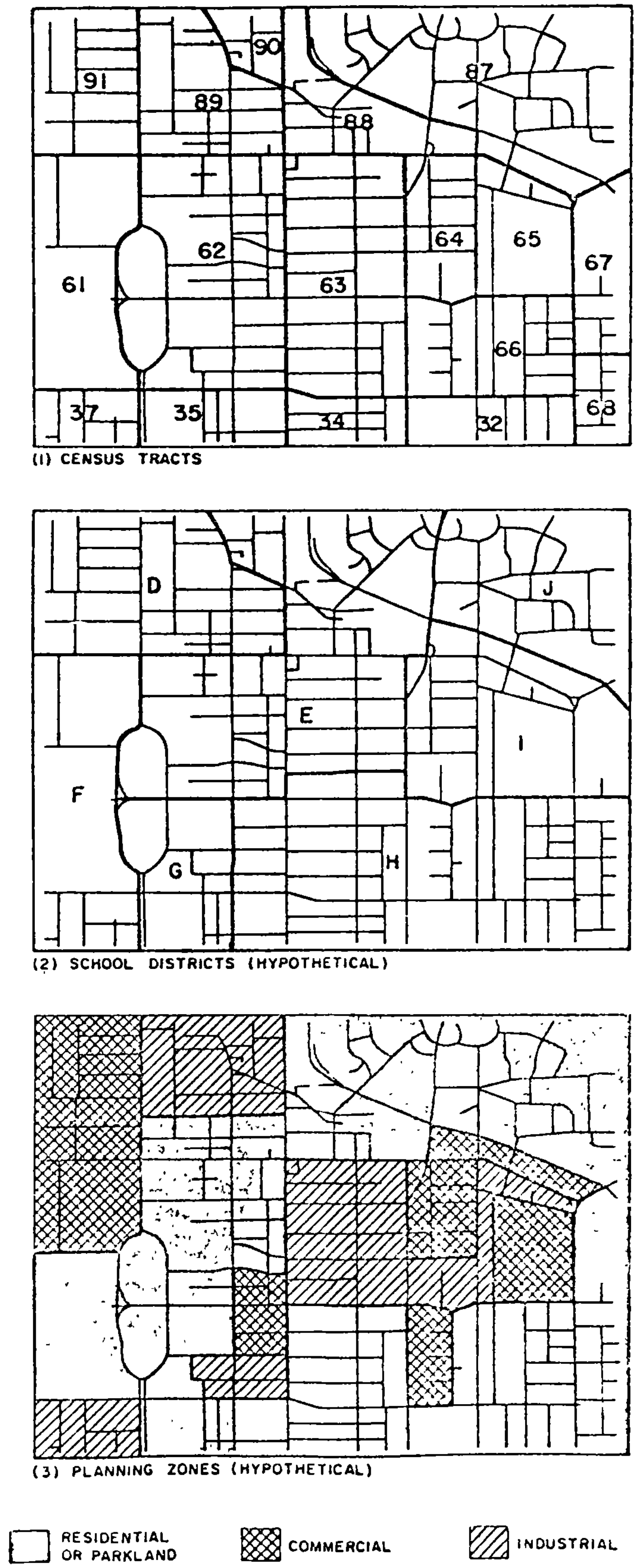
Where can GRDSR be Applied?

Given that the fundamental purpose of GRDSR is to allow users more flexibility in obtaining information about special-purpose areas, it is significant that the first major application of GRDSR has been the 1971 Census of Canada.

Originally conceived in anticipation of special census requests, the system has since been developed for general-purpose applications. Municipal assessment files, fire and accident reports, marketing surveys and hospital records are among several applications discussed in the next section.

FIGURE - I

HOW USERS IMPOSE DIFFERENT ZONING PATTERNS OVER A CITY AREA



(1) CENSUS TRACTS

(2) SCHOOL DISTRICTS (HYPOTHETICAL)

(3) PLANNING ZONES (HYPOTHETICAL)

APPLICATIONS AND POTENTIAL OF GRDSR

While Statistics Canada expects to serve many requests for statistical information from geocoded 1971 Census data, GRDSR is designed to handle the majority of address-bearing files and survey data which originate in larger urban centres in Canada (see Figure II, page 5.) Extensive geocoding applications are now possible in both the public and private sectors. Potential users include municipalities, planning and research groups, industrial and commercial firms, public utilities, social agencies, universities and governments — in short, any group using geographically-based information for research, planning or decision-making.

Noteworthy features of the Census application are outlined in part (i). Next, in part (ii), a number of other specific GRDSR applications are discussed. Finally, some aspects of a possible geocoding application, health services planning, are described in part (iii).

The Geocoded Census

GRDSR will provide a new dimension in census retrieval services: the facility to provide statistical data for user-specified areas anywhere in Canada.

Confidentiality

While Statistics Canada attaches great importance to meeting the need for custom-made, user-oriented data on a uniform, national basis, it can only do so within the confidentiality constraints imposed by the Statistics Act (1971). As a result, no information can be disseminated in such a way as to identify an individual respondent. Automatic routines within the system ensure that no such disclosure of information is possible.

Query Areas

In 14 larger urban centres (see Figure II, page 5), users may request data for areas as small as a few city blocks. Users should not, however, expect to receive representative data for smaller areas, such as one side of a block. There are two important reasons:

First, the results would be subject to high response and sampling errors, due to the small number of cases on which the statistics would be based. The usual process of compensating errors for larger samples could only take place to a limited extent.

Second, a carefully-controlled amount of statistical error is purposely introduced to all retrieved data so that no census respondent can be identified from the final tabulations. This random error, while of little or no significance to normal tabulations, would further obscure any information obtained about very small areas.

Outside the major urban centres, statistical information will be available at a coarser level of geographical detail. Here, query areas will be assembled using traditional census enumeration areas (EA's), which contain approximately 150-200 households each. As a result, extensive census data will be available for more than 27,000 EA's — either individually, or in any aggregation of interest to the user.

In either case, the desired areas are simply outlined on a suitable map, named clearly and submitted to Statistics Canada along with the tabulation request.

Request Language

Users may request census tabulations using an English-like language called TARELA (Tabulation Request Language), which can be learned in a few hours without previous programming knowledge. TARELA allows subject-matter specialists to write requests in terms familiar to their work. With this language users can create cross-tabulations of any combination of 1971 Census variables (which number more than 120) and generate tables having up to 10 dimensions. Users who are not familiar with TARELA can, of course, submit their request in precise narrative form or in the form of "dummy" tables. The required TARELA coding will then be generated by Statistics Canada.

Data Mapping

In addition to supplying census data in tabular form, GRDSR also includes a facility for data mapping. MAPPAK, which incorporates the Harvard mapping package SYMAP, is a remarkable feature in that it can accurately depict the distribution of data values over any area in graphical form. This type of map is particularly useful in locating areas where extreme values of some factor occur, and can be used to reveal problem areas at a glance.

Both TARELA and MAPPAK are general-purpose features of GRDSR, by no means limited to census applications. TARELA and MAPPAK are further described in Features and Components, page 10.

General Applications

Geocoding applications can be served by many data bases in addition to the Census. The system is designed to geocode many types of address-identified files, provided they originate in one of the larger Canadian urban centres (See Figure II, page 5.) The GRDSR programs will be available to municipal users wishing to geocode local data bases. Examples of suitable data bases include assessment rolls, traffic surveys, hospital and welfare records, marketing surveys, school census data and certain accident, fire and police records.

Municipal Administration and Government

Public Services: Research studies are being conducted, using geocoding, to determine the frequency of accident, fire and police reports originating from various sections of large cities. Such statistics would clearly be a significant aid in planning or re-allocating municipal resources and services; the use of GRDSR is possible whenever records of such incidents are address-identified.

Education: A new method for planning the location of new schools and school districts is now possible through geocoding Facts related to this application may include the concentrations and age distribution and projected growth rates of school-aged children within the community.

The routing of school buses is another application where geocoding offers considerable promise. GRDSR is ideally suited to provide statistics such as the geographic distribution of school-aged children.

Other applications include analysis of districts by such socio-economic factors as country of origin, language, religion, occupation and income as an input to planning of day-school curriculums and adult-education programs.

Urban Planning

Interests in the urban planning area include study and analysis of planning zones, optimizing the location of city services and facilities, planning of mass transit and analysis of potential urban renewal areas, land values and housing data.

In planning the route of a new city transit system, for instance, the starting points and destinations of potential users form a definite network or pattern. Subject to further analysis, such as transportation modelling, this network can have decisive impact on the final route chosen.

Further possibilities include planning of municipal services according to socio-economic factors such as population density, language, and income within selected urban zones. New approaches to planning the nature and location of welfare services may also become possible.

Medical Services

Typical problems include planning the location of hospitals, out-patient clinics and medical centres, and the establishment of a geographically-referenced inventory of nurses.

Industry, Commerce and Utilities

Geocoding has played a part in the allocation of facilities and services such as telephone exchanges and banks. Other applications include population and demographic studies of city areas, the planning of marketing zones and radio and television coverages, the optimization of retail store location in terms of customer proximity and resource allocation problems faced by oil, hydro and gas utilities.

A number of simulation and modelling techniques exist for solving network problems in the commercial transportation/distribution area. Typically, data related to some grid pattern constitute an essential requirement for this approach. GRDSR is an ideal research tool to help meet this need.

Universities

Interests include economic, political and social studies of neighbourhoods, electoral districts and socio-economic research into city areas defined by such factors as country of origin, language or income.

Health Services Planning: A Potential GRDSR Application

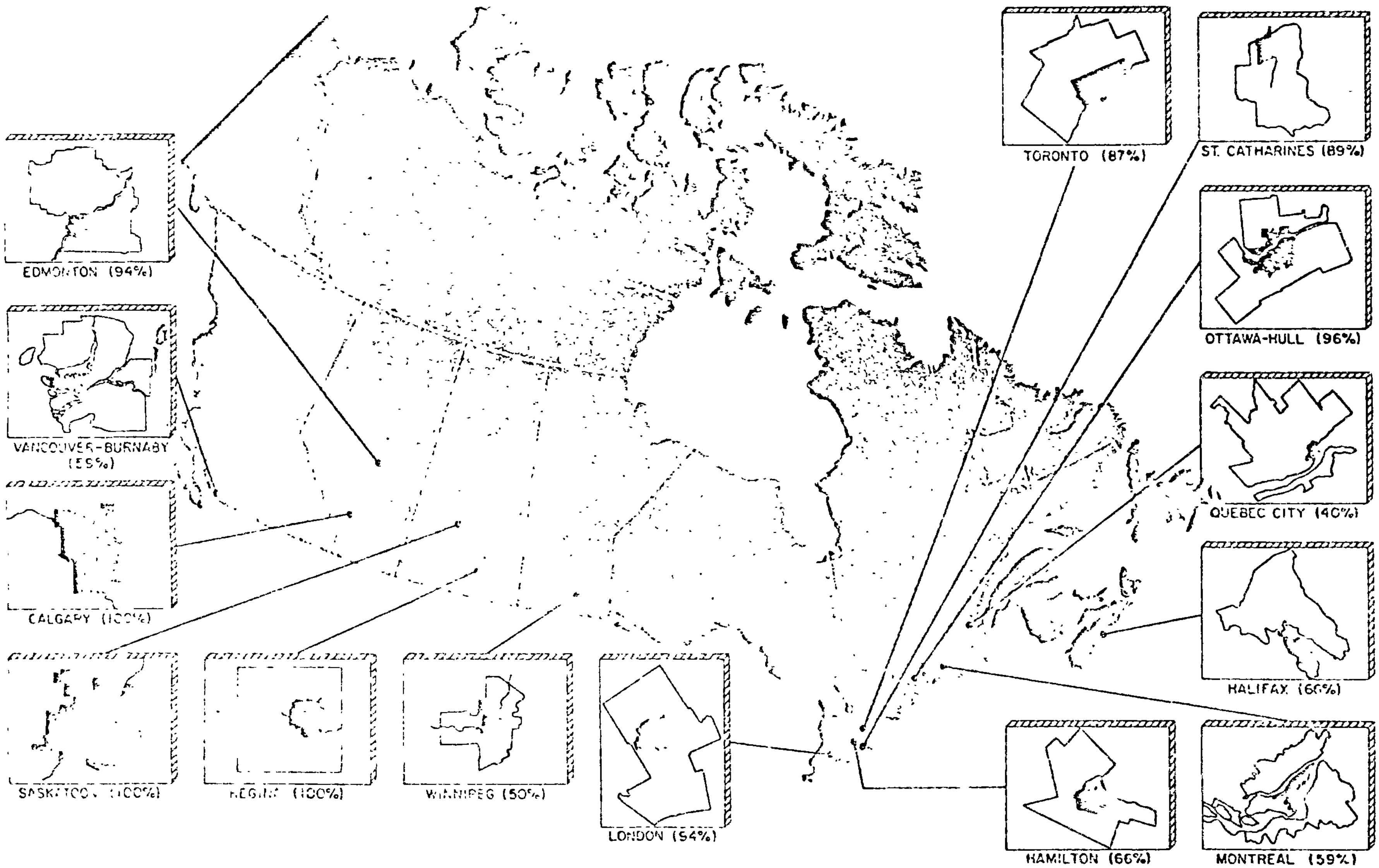
A number of factors influence the choice of location for a new hospital or health services clinic in a major city, such as accessibility through major traffic arteries, availability of professional staff, areas most in need of services.

GRDSR can be particularly helpful in deciding which city areas are most in need of proximal medical facilities. One approach is to find out where past patients have lived and what medical services and equipment they required, using city hospital records.

Hospital visitation records bear, in addition to medical content, an address identifier for every patient. Therefore geocoding operations can, in most cases, be carried out on the visitation records. Through GRDSR, considerable statistical information can then be generated (for instance, the incidence of hospital visits originating from each and every portion of the city). The retrieved information can be cross-tabulated by the type of medical services required or by any other item of information contained in the original records. For example, the incidence of various diseases, illnesses, or special health problems in certain city areas can be ascertained. Such statistics can prove to be an invaluable aid in determining which city areas would best be served by neighbourhood medical services or a new hospital.

FIGURE - II

GEOCODING COVERAGE AT THE BLOCK-FACE LEVEL (JUNE, 1971)



IN EACH DIAGRAM, THE OUTLINE DEPICTS THE BOUNDARY OF THE CENSUS METROPOLITAN AREA WHILE THE SHADED PORTION REPRESENTS THE AREA OF BLOCK-FACE COVERAGE. THE PERCENTAGE OF POPULATION RESIDING IN THE SHADED AREA IS ALSO SHOWN. DIAGRAMS ARE ALIGNED IN THE NORTH-SOUTH DIRECTION.

APPROXIMATELY 7 MILLION, OR 34% OF THE POPULATION OF CANADA ARE NOW COVERED AT THE BLOCK-FACE LEVEL.

CONCEPTS AND METHODS

Review of small-area problems

An urban planner, faced with comparing the expropriation costs of several expressway routes, might attempt to use municipal assessment files and find that records were identified by address, city wards or in some other way. To obtain statistics about land values and dwelling types, the file must be inspected one record at a time to determine which data to include in estimates for the proposed expropriation area. The expense of this approach has been prohibitive but, until recently, few alternatives were available.

Another type of requirement, now directed to the census, might be phrased as follows:

> *"A tabulation of the number of people resident in the Toronto area bordered by Summerhill Avenue, Yonge Street, Mount Pleasant Cemetery, and the boundary for East York is required. Break this tabulation down by age, sex, income, country of origin and occupation."*

Alternatively, another request might read:

> *"Provide the same statistics for the area named Ward Five, as outlined on the attached city map."*

Such requests have been difficult to service, since census data have been summarized by census tracts and enumeration areas, which may not coincide with the required boundaries.

To solve small-area data requests economically, Statistics Canada required an efficient system to repeatedly assemble and tabulate information according to arbitrary special-interest areas. Before describing the conceptual aspects in detail, let us expand upon the operational steps in GRDSR.

Before a data base can be geocoded, each record must be assigned some reference code which identifies the record to its proper geographical source. In GRDSR, the source of each record or data observation is precisely located using a comprehensive geographical coordinate system. Each record is assigned a coordinate value, or "key", which actually becomes part of the record during the geocoding operation. The geocoded file is then stored for later use. Ultimately, at retrieval time, GRDSR automatically identifies each query area with a list of coordinate values and, using the coordinate values as keys, retrieves the precise set of data records required. The retrieved information is then summarized according to the tabulation request; the user receives statistics in the form of a convenient report.

The UTM System

The coordinate system chosen for GRDSR is known as the UTM (Universal Transverse Mercator) System. UTM is an established international convention for specifying point-locations on the globe, and is shown on the popular National Topographical Map Series produced by the Department of Energy, Mines and Resources. This system divides the globe into 60 vertical zones. Altogether, 16 zones cover the land mass of Canada. Each zone has a width of 6 degrees longitude and a central meridian which becomes the vertical axis for the zone. The horizonal axis is formed by the earth's equator.

In UTM, point-locations within a zone are based on two distances in metres (one easting, one northing) from the zone axes. The central meridian is assigned an artificial value of 500,000 metres easting; the equator is assigned the value 0. Distances are measured on a plane rectangular grid onto which the zone's surface features have been projected. The two values are combined with a zone number to arrive at a unique coordinate value for every point on the land mass of Canada.

For example, the UTM coordinates of the Peace Tower, Ottawa, are:

Zone	X	Y
18	445177	5030250

In this way, the UTM coordinates seem to define a point-location to the nearest metre, although the projection of the earth's surface onto a plane grid introduces minor distortions.

Basic definitions

Points at which streets intersect or curve sharply in the city pattern are referred to as *nodes*. Every street is represented by a series of nodes connected by straight-line *segments*.

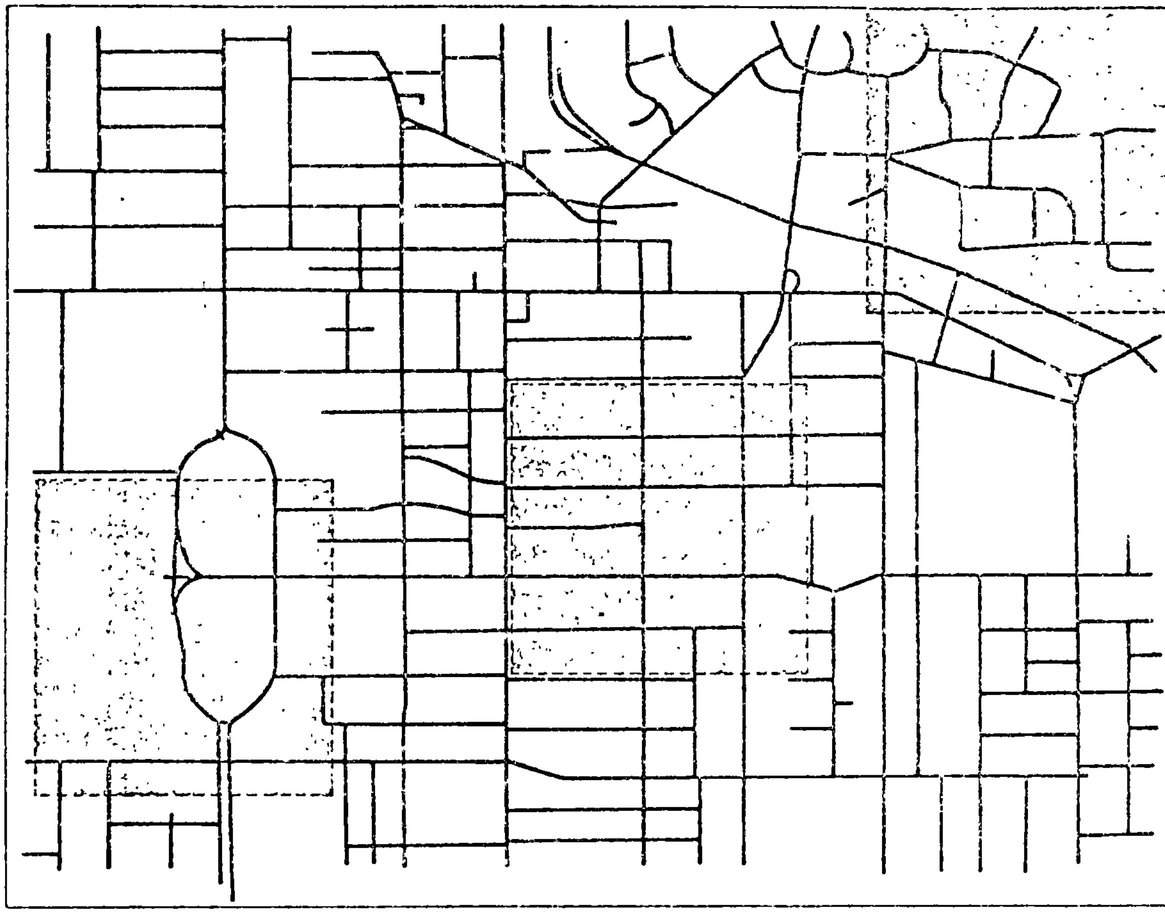
A *block-face* is defined as one side of a city street between consecutive intersections with other streets. Thus, up to two block-faces can be formed by a pair of adjacent nodes, each located at a four-way street intersection. However, a block-face can also encompass several nodes. For example, a block-face may contain one intermediate node marking a change in direction and another node representing an intersection on the opposite side of the street only (see Figure III, page 7).

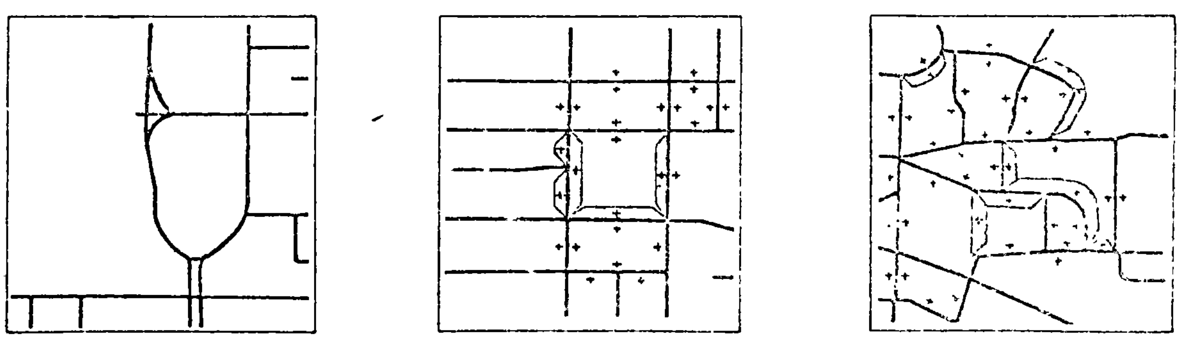
Whenever a block-face is to be formed by a pair of nodes, these nodes must constitute the beginning and end of a valid civic address range.

In this way, block-faces become the basic building blocks used in the GRDSR System.

FIGURE III

HOW BLOCK-FACES AND CENTROIDS (+) ARE CHOSEN





(1) STREETS THROUGH RECREATIONAL OR PARKLAND — — NO CENTROIDS

(2) STREETS IN REGULAR (GRID) PATTERN

(3) STREETS IN IRREGULAR PATTERN

 THE SHAPES OF SEVERAL BLOCK-FACES ARE SHOWN BY SHADED AREAS

Why addresses are necessary

The GRDSR System is partly based on the premise that most agency records and survey responses are identified, geographically, by the addresses of respondents. An address is the starting point in coordinate assignment, because every street address in an urban centre can be identified as belonging to some block-face.

How addresses are converted into coordinates

In GRDSR, all street addresses along a block-face are assigned, and share, the coordinates of the block-face centroid, which is simply a reference point offset from the street midway between the two nodes forming the block-face. During the conversion operation, the address of each record or data observation is matched to a block-face (using a list of valid street names and address ranges). From there, the correct centroid is known and its coordinates can be added to the record.

The Area Master File

The actual geocoding operation (or assignment of coordinates to data) is carried out using GRDSR components known as Area Master Files (AMF), which will be described in detail in Features and Components, page 10.

Area Master Files contain a logical representation of all city streets, plus some other features, in computer-readable form. An AMF references every street, address range, block-face and centroid coordinate in the covered area. Also itemized are other features (such as railroad tracks, rivers, and municipal boundaries), which help users to choose query areas. During the geocoding operation, centroids are obtained by matching addresses against street names and address ranges within the Area Master File. (In this way, address ranges can be thought of as representing the actual building blocks, rather than block-faces.)

Area Master Files have been created for major portions of 14 Canadian urban centres, which include a total of 16 cities (see Figure II, page 5). These files reference more than 225,000 block-faces, corresponding to a population figure of approximately seven million.

Rural Geocoding Coverage (1971 Census)

The 14 Area Master Files have already served to geocode certain urban portions in the 1971 Census. For the remainder of Canada not covered at the block-face level (urban and rural), census geocoding, as already noted, has been carried out using standard enumeration areas, with one centroid assigned to the approximate population centre of each. Enumeration Areas outside Area Master File coverage number more than 27,000. Tabulation requests for query zones in rural areas or in the urban shadow of developing urban areas are easily (and automatically) handled using centroids at the EA level, the block-face level, or both. During retrieval, inaccuracy in data selection at the EA level is minimized by the choice of centroids near the population centre and by a process of compensation, whereby errors from including or missing centroids in a query area are self-cancelling.

ADVANTAGES, LIMITATIONS OF CONCEPTS

Choice of block-faces

Block-faces become the finest level of resolution possible when each cluster of data observations or survey records belonging to one block-face share the same centroid coordinate. This is a logical outcome of the building block principle adopted by GRDSR.

As one alternative, geocoding to the land parcel or household level achieves higher resolution which may be desirable for some purposes. This approach requires extensive local research. Since block-face resolution is expected to satisfy the vast majority of geocoding requests, land-parcel geocoding could not be justified for a Canada-wide system such as GRDSR.

A second alternative was to identify data by city block, a poorer resolution. However, this approach would not have allowed users enough flexibility, since the integrity of city blocks would have to be respected in specifying query areas. For instance, it would not be possible to obtain tabulations for one side (or both sides) of a city street.

The choice of block-faces as basic geocoding building blocks has several implications. All observations originating from one block-face bear the coordinate of its declared centroid. As a result, the integrity of block-faces should preferably be respected in specifying query areas for retrieval. They should not be split: observations referenced to a split block-face will appear in the results only if the query area includes the block-face centroid. If not, the observations are missed entirely. Another implication is that geocoding to the household or land parcel level (each individual property bears a centroid) is not possible using this system. This may pose definite restrictions on municipal services, engineering and land-banking applications where higher resolution is required.

Identification by street address

Statistics Canada recognizes that a majority of statistical surveys and agency records are address-identified and provides for this with a System component known as the Postal Address Analysis System (PAAS).

In geocoding a file, addresses are analyzed and converted to centroid co-ordinates. Because the conversion is done by computer, complete addresses must be decomposed into separate, clearly identified components (such as street name, type, house number and municipality name). Because PAAS achieves a high efficiency and success rate, address specifications of relatively poor quality can still be geocoded. This feature clearly extends the scope of GRDSR applications. More information about PAAS is provided in Features and Components, page 10.

However, GRDSR cannot perform the geocoding operation on records which, by their nature, are not identified to street addresses. Certain city facilities, such as sewers, gas and hydro lines, traffic signals and overhead structures may be of interest from a geocoding standpoint. In this case, the user must geocode the file before submitting it to GRDSR.

Choice of coordinate system

While UTM is ideally suited to geocoding at the block-face level it has some limitations in land survey and civil engineering operations where the 3 Transverse Mercator System is more accurate, and thus a frequent choice. However, programs are available to convert files geocoded with the UTM system to 3 TM and vice-versa.

FEATURES AND COMPONENTS

The Area Master File

How an AMF is created

Geocoding starts with an accurate street map. A large-scale, current map showing block-face address ranges is required, together with an up-to-date street index. After the map is divided into sections a node is assigned to each street intersection. Nodes are also assigned to points where streets begin, end or curve sharply. A non-distorting overlay is prepared for each map section and the position of each node is marked on the overlay.

Once serial numbers have been assigned to the nodes, descriptive codes for every street segment are transcribed onto a specially-prepared form. The codes include feature names, types, directions, node numbers, and addresses at the intersections. Then the overlay is placed on a digitizing table. The digitizing equipment measures node positions relative to control points on the overlay, and generates one horizontal and vertical "table" coordinate for each node. Since the UTM coordinates of the control points are known in advance, the UTM coordinates for the nodes can then be calculated from the table coordinates. During subsequent computer processing, centroid coordinates are calculated for each block-face using the coordinates of the two nodes bordering the block-face. Finally all items are merged to create an Area Master File for the city (see Figure IVa, page 11).

How the AMF is used

Three operations, each related to address conversion, require files of information contained in the Area Master File. To eliminate the maintenance and updating of three separate files, each is derived from a clean, up-to-date AMF as required.

(i) Street name lists are used by PAAS to verify input addresses prior to the assignment stage.

(ii) The Address Conversion File (ACF) is used to obtain centroid coordinates for input addresses once the PAAS stage is complete. Addresses are matched against block-face address ranges and the corresponding centroid coordinates are selected from the ACF. Geocoding is complete once centroid coordinates replace addresses in the original file.

(iii) The Block-Face File was created specifically to geocode the 1971 Census. This file makes it possible to link parcels of census data, which are not otherwise address-identified, to block-faces and centroids.

As a geographic base file, the Area Master File design is unique. The central concept is to provide a geographical framework that is as practical as possible for a variety of potential users, but efficient from a file creation/update standpoint.

A series of error-handling and correction procedures comes into play whenever Area Master Files are being built or updated. Extensive computer checking is done to ensure that each node is linked to the correct street segments, and vice-versa. This process locates the majority of clerical errors. When each section file is complete, it is plotted at the same scale as the original map. The two maps are then compared to verify node locations. Usually, further plotting followed by two to three update cycles, will produce a clean Area Master File.

Local area breakdowns, such as census tracts, electoral wards, city wards, and other extra codes were purposely excluded from the AMF. Its design is such that these jurisdictions are easily constructed independently of the AMF, but using the identical building-block technique. Because areal boundaries are constantly changing, their inclusion would have seriously prolonged the operation needed to build and maintain an accurate, up-to-date base file.

Urban Street Maps

Computer-plotted street maps are an important by-product of building an Area Master File. Because the AMF is a logical representation of city features, its contents can be used, in reverse, to create facsimile maps at any scale. Plotting is accomplished using the GRDSR component, MAPMAKR (see Figure V, page 15).

These maps have several purposes:

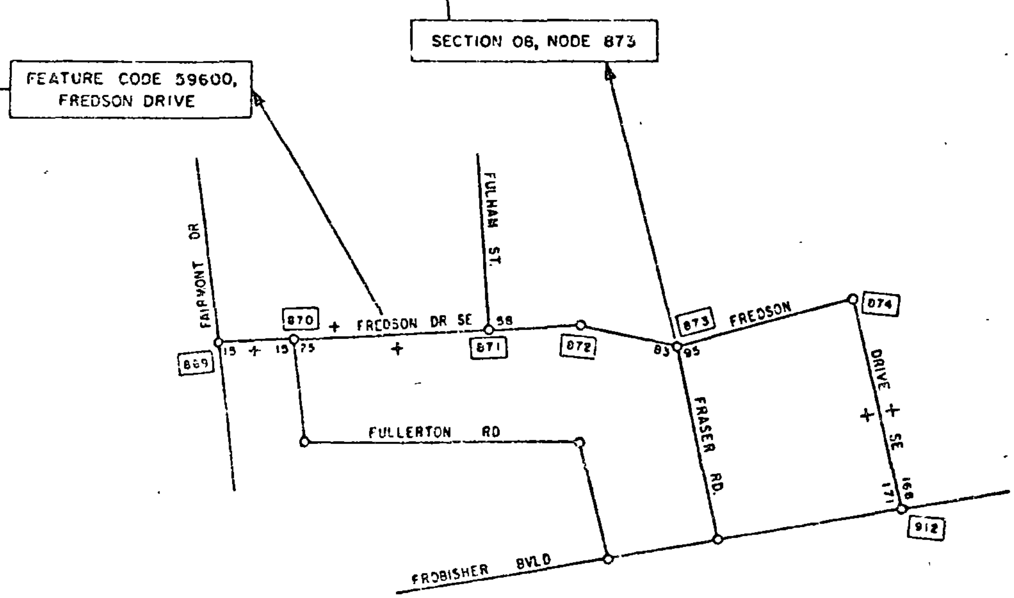
- The best way to edit or validate an Area Master File is to recreate the original map, using the plotter. Errors and inconsistencies are clearly highlighted.
- They provide a return service to municipalities who in turn are aware of what updates are required.
- The maps are supplied to users for outlining query areas and depict city features as seen by the AMF.

Using MAPMAKR, maps can be produced to suit a variety of purposes. It is possible to pre-specify the area to be plotted and the scale required. Parameters are used to determine whether various options, such as nodes, feature names, centroids, address ranges and control points, will appear on the final plot.

FIGURE - IV a

THE AREA MASTER FILE (SPECIAL FORMAT)

MUNIC CODE	FEATURE CODE	SEQ No	STREET NAME	TYPE	DIR	NODE NUMBER	NODE COORD X	NODE COORD Y	ADDRESS BEFORE L	ADDRESS BEFORE R	ADDRESS AFTER L	ADDRESS AFTER R	CENTROID LEFT X	CENTROID LEFT Y	CENTROID RIGHT X	CENTROID RIGHT Y	INTERSECTING FEATURES
4835	59600		FREDSON	DR	SE												
		040E				08912	706954	5651540	168	171	—	—	706868	5651755	706861	5651673	FROBISHER BV
		035				08874	706844	5651755	—	—	—	—	—	—	—	—	
		030				08873	706771	5651722	—	83	—	95	—	—	706635	5651700	FRASER RD
		025				08872	706733	5651729	—	—	—	—	—	—	—	—	
		020				08871	706631	5651722	—	—	58	—	706516	5651734	—	—	FULHAM ST
		015				08870	706497	5651711	—	15	—	25	—	—	706452	5651687	FULLERTON RD
		010B				08869	706406	5651707	—	—	—	15	—	—	—	—	FAIRMONT DR



(+) CENTROIDS FOR FREDSON DR SE

Postal Address Analysis System

Addressing conventions vary according to locality, language and post office regulations, but few comprehensive systems are available to digest and organize a file of street addresses. The PAAS system is a flexible and inexpensive device for accomplishing this job. For geocoding applications, addresses can originate from any city having an Area Master File at Statistics Canada (see Figure II, page 5). Otherwise, PAAS can re-structure and organize virtually any address file in use today.

While the number of addressing conventions across Canada is considerable and many conventions often appear in one file, PAAS consistently demonstrates a high success rate at exceptionally low cost. In its current version, it accepts street addresses (including municipality names) in completely free format and decomposes each address into several elements such as street name, street type and direction. The addresses are then matched against a subset of the Area Master File (the Address Conversion File) and, if the match is successful, a centroid coordinate is assigned to each record in the original file (see Figure IVb, page 13).

The flexibility of PAAS is enhanced through parameters which are passed to the program when geocoding starts. These parameters improve PAAS efficiency by indicating the nature and characteristics of the incoming addresses.

Significantly, the entire conversion process is accomplished at an average cost of less than one cent per address.

The Query Area Library

Many users are expected to submit special-purpose areas for data retrieval and refer to them repeatedly in making requests. Statistics Canada also expects continuing requests for census statistics arranged by the traditional standard areas – provinces, counties, census tracts and enumeration areas. (Altogether, there are 13 distinct sets of standard census areas, each set covering most of the settled area of Canada. The 13 sets comprise more than 53,000 separate areal units.)

Before information about any query area can be retrieved, GRDSR must define the query area in terms of the geocoded data base. Definition is accomplished by associating the area name with "pointers", which indicate precisely where the desired elements can be found. Pointer sets for each standard census area and for special-purpose areas are kept in a system component called the Query Area Library. A QAL is opened specifically for each new data file stored in GRDSR.

In normal practice, users outline the boundary of a special query area on a map. Vertices along the boundary are located using a digitizer so that their positions can be converted to UTM coordinates. A computer-programmed algorithm is used to test whether each successive centroid coordinate in the data base belongs in the query area. Finally, the coordinates selected are converted to pointers, which serve to locate the corresponding data elements required.

To avoid repeating this process, frequently-used area names are stored in the Query Area Library. Each area name is associated with a set of pointers. Areas that will be requested often and by different users are stored in a portion of the QAL reserved for permanent areas. Other area definitions will be stored for a limited time in the temporary QAL. Several other methods for designating query areas are described in Operations, page 18.

STATPAK

STATPAK was developed for GRDSR as a generalized program to retrieve statistics efficiently by arbitrary areas. Users communicate with STATPAK through the problem-oriented language TARELA and receive statistics in the form of convenient, easy-to-read tables.

How a file is stored

STATPAK's efficiency is made possible by changing the structure of an incoming file after it has been geocoded. Instead of keeping all data characteristics for a respondent together in one record, each data characteristic is handled separately. The entire set of responses for one data characteristic are assembled and stored as a continuous string. Because more than one record is usually attached to each centroid coordinate, an index is built to locate precisely where the responses for each distinct coordinate value are found. The index is then used to provide pointers for new query areas before they enter the QAL.

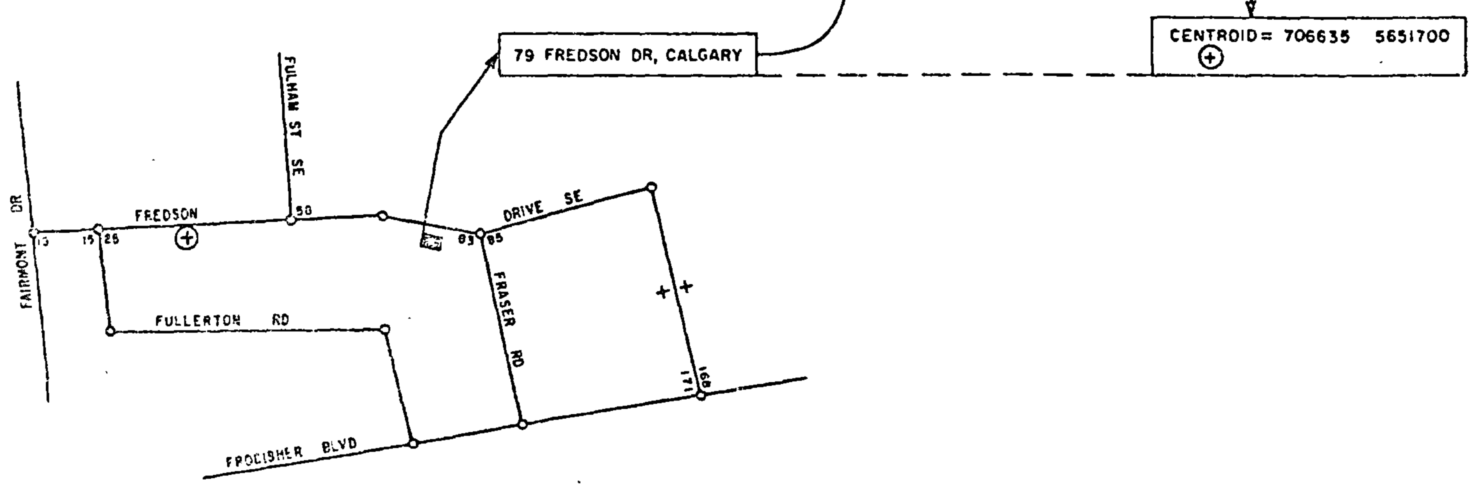
Complete strings of data characteristics are finally stored on direct-access devices and a Query Area Library is established for the file.

To visualize the final geocoded data base, imagine a huge matrix. Using the 1971 Census as an example, 21.6 million people counted in the census are arranged vertically in order of their centroid coordinates along the left side. Approximately 120 data characteristics form vertical parallel strings suspended from the top of the matrix. Instead of storing all characteristics for a person in a self-contained record, one string is created for each characteristic (such as age, sex, marital status, income or occupation). This method allows each string to be compressed to occupy the least possible space for

FIGURE- IV b

THE ADDRESS CONVERSION FILE (SPECIAL FORMAT)

STREET NAME	TYPE	DIR	FEATURE No	ODD/ EVEN	ADDRESS RANGE		CENTROID		
					LOW	HIGH	ZONE	X	Y
FRASER									
FREDSON	DR	SE	59600	ODD	15	15	18	706452	5651687
"	"	"	"	ODD	25	83	18	706635	5651700
"	"	"	"	ODD	95	171	18	706861	5651673
FREDSON	DR	SE	59600	EVEN	58	168	18	706868	5651755
FROBISHER									



CENTROIDS (+)

the information contained. As a result, the use of costly direct-access storage space is minimized.

During the storage operation a name for each data characteristic is retained along with code names for the values the characteristic can assume. The names appear in a document called the Data Dictionary which is used, in turn, to code TARELA requests. The problem-oriented nature of TARELA rests on these names, because they are chosen by subject-matter specialists when files are submitted to GRDSR for geocoding.

How information is retrieved

After STATPAK accepts and analyzes a TARELA request, it generates a tailor-made program to retrieve the data. The program is then executed.

The operating advantage rests on large files where only a small portion is accessed at one time, that is, whenever tabulations are requested for small areas or relatively few data characteristics. Because data are retrieved in direct-access mode, the actual execution cost is strictly dependent on the extent of the query area and on the nature of the tabulations required, not on the size of the whole file. In a file of 1.5 million records, the costs of a tabulation vary from $30 to over $100, depending on the number of records to be retrieved.

Any file which has fixed or variable-length records with geographic identification (ideally UTM coordinates) may be reorganized into a form acceptable to STATPAK. Written in PL/I, STATPAK is a set of modules assembled into a tailor-made source program for each new TARELA request. The tailor-made program is exceptionally efficient because it immediately locates the required data string and accesses only those portions belonging to the user-defined query area. It is erased when the final tabulation is complete.

STATPAK is implemented at Statistics Canada on the System/360-65, under OS/MVT and HASP, and occupies roughly 150 K bytes of core storage.

TARELA

Tabulation requests are coded in a highly user-oriented language called TARELA, requests in which are submitted directly to the system and will normally be returned within one or two days, depending on the computer workload and the size of tabulations requested.

As a data retrieval language TARELA offers significant advantages It spares non-programming users the trouble of writing retrieval requests for subsequent analysis and programming, and it frees programmers and analysts for more complex work, such as refining the GRDSR System. Programming, debugging and testing delays are bypassed. Finally, potential communications problems inherent in dealing with different professional groups are avoided because the ultimate user can himself communicate directly with the data base.

To write a TARELA request, users must have access to the appropriate Data Dictionary created when their data base was geocoded. A standard data dictionary for the census files will be available to interested users. Using the dictionary, each response characteristic (i.e. age, sex, occupation) is selected by name and code words representing numerical values appearing in the data base. The user can also specify appropriate functions (such as a COUNT of persons satisfying some criteria, or SUM and AVERAGE of a set of retrieved data values). As the request is formulated, coded information is simply written after each TARELA keyword as shown in Figure VI, page 17.

Data Mapping by Computer

MAPPAK is a facility to display spatial distributions of a statistic in the form of a map. MAPPAK operates as an interface between STATPAK and SYMAP, a mapping program developed at the Laboratory for Computer Graphics, Harvard University.

Reading statistical data from a map often has compelling advantages over having the same information tabulated in report form. Inspection of the map can instantly show where extreme values of some function occur. A map can highlight problem areas at a glance.

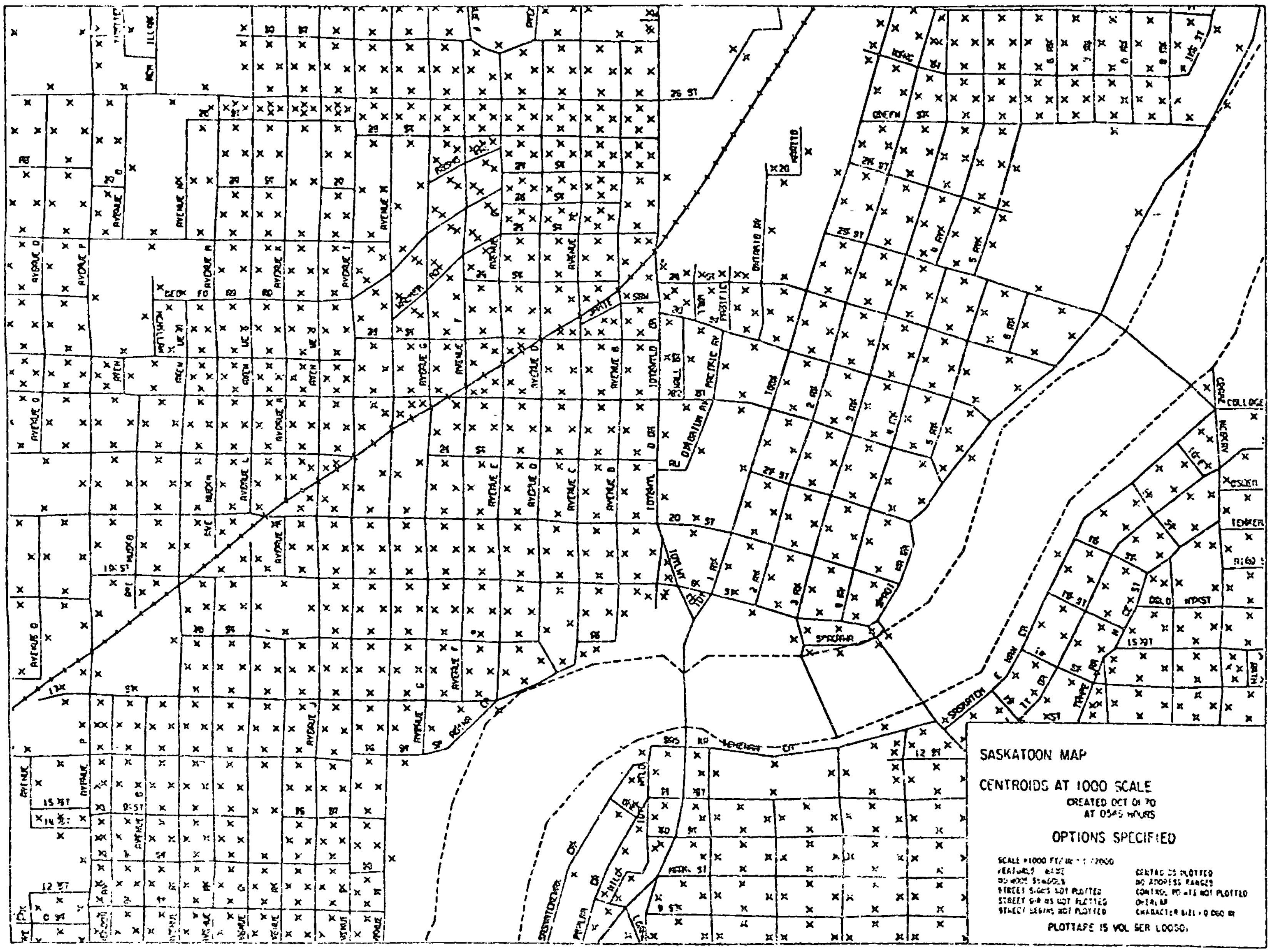
MAPPAK can be used to stratify data values into several classes or to filter a data characteristic. The results are shown as numbers or as shaded areas on the paper surface.

For instance, if a MAPPAK user is interested in census data, the distribution of average income can be depicted in many levels of shading over a city area. Or, a user can specify that an area be subdivided into 400 by 800 foot rectangular cells, with the average number of children per family shown as a number within each cell. To illustrate the filtering characteristic, MAPPAK can be requested to shade city areas where half the population is of foreign origin, or where a majority of families rent rather than own homes.

The uniform data areas generated by MAPPAK can take many forms Users can request data relative to any grid cell pattern, by rectangles of any size, or in terms of concentric circles At one extreme, a data value can be mapped for every centroid point in the city area (subject to confidentiality constraints). At the other extreme, a single data value for some arbitrary area sketched on a street map can be obtained.

FIGURE-V

COMPUTER-PLOTTED STREET MAP OF SASKATOON (PORTION)



MAPPAK incorporates all SYMAP facilities including contour mapping of surface data, classification of data values within arbitrary, pre-defined areas and summing the distribution of a set of data values. It has the flexibility to display detail down to the finest level on the data base (the block-face) and can generate maps to any desired scale (see Figure VII, page 19).

Again, it must be pointed out that the routines for confidentiality checking will be applied when MAPPAK is used to retrieve data from the geocoded census files. The routines will operate in the same manner as for regular statistical tabulations.

FIGURE-VI

PREPARING A TARELA REQUEST

```
RESEARCH PROJECT PLANNING ZONES

AREA SET ID--
TS01674000

AREA ID---     DESCRIPTION ---

TA01674001     PLANNING ZONE 1
TA0167.002     PLANNING ZONE 2
TA01674003     PLANNING ZONE 3
TA01674004     PLANNING ZONE 4
TA01674005     PLANNING ZONE 5
```

```
TARELA DATA DICTIONARY FOR FILE
****   GEOCODED 71 CENSUS FILE SHORT FORM
FILENAME:GE71CFSF:

PART3 : VARIABLE AND CODE DESCRIPTION
VARIABLES ON LEVEL POPULATION
```

VARIABLE INTERPRETATION	VARIABLE NAME	CODE OR CODE RANGE	CODE NAME	CODE INTERPRETATION	STUB TEXT
RELATIONSHIP TO HEAD OF HOUSEHOLD	RELTOHD	1	HEAD_OF_HOUSEHLD	HEAD OF HOUSE HOLD	HEAD OF HO
		2	WIFE	WIFE OF HEAD OF HOUSEHOLD	WIFE OF HE
		3	SON	SON OF HEAD OF HOUSEHOLD	SON OF HE
		4	DAUGHT_IN_LAW	DAUGHTER-IN-LAW OF HEAD OF HOUSEHOLD	DAUGH.TER-
		5	SON_IN_LAW	SON-IN-LAW OF HEAD OF HOUSEHOLD	SON-.IN-.
		6	DAUGHTER	DAUGHTER OF HEAD OF HOUSEHOLD	DAUGH.TER
		7	GRANDSON	GRANDSON OF HEAD OF HOUSEHOLD	GRAND.SON
		8	GRANDDAUGHTER	GRANDDAUGHTER OF HEAD OF HOUSEHOLD	GRAND.DAUG
TOTAL INCOME OF PERSON	TOTINCOM	0-99999		TOTAL INCOME IN $	

```
FILE: GE71CFSF;
AREA: TA01674001;
DEFINE: HEAD-INC FOR HOUSEHOLD AS TOTINCOM IF RELTOHD=1;
HEADING: NUMBER OF HOUSEHOLDS
      AVERAGE INCOME OF HEAD OF HOUSEHOLD
      EACH BY NUMBER OF ROOMS IN DWELLING AND NUMBER OF PERSONS
      IN HOUSEHOLD;
CHARACTERISTICS:
   PERSINHH 1,2,3,4,5 OR MORE;
   ROOMS 1,2,3,4-5,6 OR MORE;
TABULATE:
   COUNT,
   AVERAGE(HEAD-INC) /#AVERAGE HEAD INCOME#/;
```

NUMBER OF HOUSEHOLDS
AVERAGE INCOME OF HEAD OF HOUSEHOLD
EACH BY NUMBER OF ROOMS IN DWELLING AND NUMBER OF PERSONS
IN HOUSEHOLD
2 FEBRUARY 1972 PAGE 1

AREA: PLANNING ZONE 1

	ROOMS 1	ROOMS 2	ROOMS 3	ROOMS 4-5	ROOMS 6 OR MORE
PERSINHH 1					
COUNT...............	15	15	40	40	30
AVERAGE HEAD INCOME.	3,555	3,035	1,810	1,515	850
PERSINHH 2					
COUNT...............	5	15	40	180	115
AVERAGE HEAD INCOME.	3,010	3,020	2,725	3,250	3,500
PERSINHH 3					
COUNT...............	-	5	15	140	75
AVERAGE HEAD INCOME.	-	3,050	3,615	4,380	3,990
PERSINHH 4					
COUNT...............	5	-	5	135	90
AVERAGE HEAD INCOME.	2,550	-	3,305	4,835	4,910
PERSINHH 5 OR MORE					
COUNT...............	-	-	10	200	285
AVERAGE HEAD INCOME.	-	-	3,190	5,350	5,150

IN THE REQUEST, THE AREA ID IS PROVIDED BY THE AREA LIST (TOP LEFT). THE FILE ID, VARIABLE NAMES AND CODE NAMES ARE PROVIDED BY THE DATA DICTIONARY SHOWN (IN PART) AT THE TOP RIGHT. IN THE OUTPUT TABULATION (ABOVE) THE AREA DESCRIPTION, "PLANNING ZONE 1" HAS BEEN RETRIEVED FROM THE QUERY AREA LIBRARY

OPERATIONS

Handling User Surveys

Many geocoding applications are of interest to municipal administrations. GRDSR can be used to access information of significant importance to urban planning and administrative processes. Several possible applications were described on page 4.

Geocoding and Data Storage

The geocoding operation can now be carried out in 14 larger Canadian urban centres having Area Master Files at Statistics Canada (see Figure II, page 5). Since GRDSR is fully computerized, input files can be in machine-readable form (such as punched cards or magnetic tape). Any unusual address structures in the input file may require definition prior to submission. Once the user has provided a description of the file (including record length, variable names and values, address location, etc.) geocoding can begin. This can happen in one of two ways. Users in the public sector (municipal, provincial, and federal governments) who have suitable computing facilities may obtain the GRDSR System for their own use. In other cases this operation as well as subsequent data retrievals may be carried out by Statistics Canada under contract.

In either case the actual processing phases are as follows:

- To geocode the input file, address identifiers are removed, analyzed and used to assign a centroid coordinate to each record.
- Data characteristics are gathered together and arranged in strings. The re-organized file, together with control information describing the strings, is stored on direct-access devices, ready for data retrieval.
- A Query Area Library is opened for the file.
- Finally, a Data Dictionary for preparing TARELA requests is created.

Data Retrieval

Any number of retrievals can be carried out once the storage operation is complete. Definition of query areas for the retrieval phase can also be done in several ways. Initially, it will be necessary to outline the desired query areas on a city map, name them clearly, and submit these specifications along with the tabulation request. For instance, query areas for a municipal retrieval might be defined as "Planning Zones I, II, III, and IV". Mapped query area boundaries are then digitized and converted to UTM. At this point, the area definitions are stored in the Query Area Library opened specifically for this file. Subsequent tabulation requests for these areas can then be referred to the QAL for definition rather than repeating the UTM conversion operation.

From this point onwards, it is possible to obtain data tabulations through the GRDSR System. TARELA is used as a vehicle for the tabulation request, which is coded using the Data Dictionary. For instance, if the input data base was an assessment file, a tabulation of assessed value for various dwelling types could be specified using convenient characteristic names (such as "VALUE", "DWELLTYPE"), area names (such as "PLANNINGZONE4"), file names (such as "ASSMFIL6") and parameters indicating the format of tabulations desired.

Requests for further tabulations can be handled in a similar manner. Tabulation requests can be processed with exceptionally quick turnaround once files have been geocoded.

How users can specify areas

An important feature of GRDSR is that it accepts area descriptions in several convenient ways.

Outlines on maps

Users will probably find it most convenient to outline query areas on a map. In most cases any convenient map can be chosen. Statistics Canada is producing copies of computer-plotted city maps, which are particularly appropriate for graphically displaying block-faces in each city area. Outside the urban areas of block-face geocoding coverage, users will be advised to use the National Topographic Series (NTS) maps produced by the Department of Energy, Mines and Resources. The important thing is that users choose an appropriate map scale, then mark out and name query areas as clearly and as accurately as the problem demands.

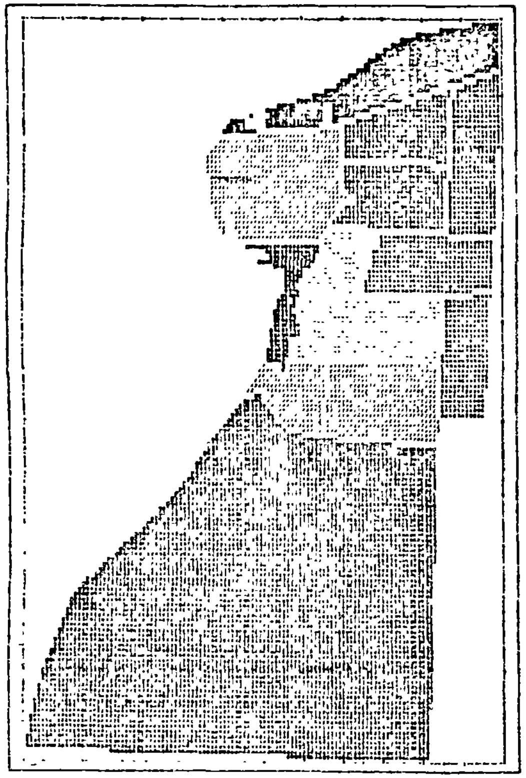
Defined by features

It will be possible to specify a query area in terms of known features (rivers, streets, railroad tracks). For instance, an Ottawa user could describe in writing that Research District No. 5 consists of an area bounded on the north by St. Patrick Street, on the east by Chapel Street, on the south by Templeton Street and on the west by Nelson Street. It is possible to request data for street-oriented query areas in the same manner. For example, a user could request statistics for one side of Rideau Street, in the municipality of Ottawa; from Sussex Drive to King Edward Avenue (odd-numbered side, even-numbered side, or both).

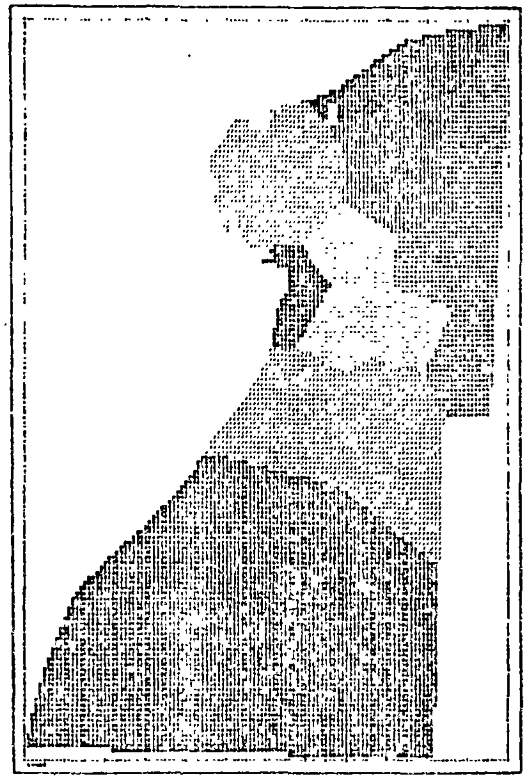
Rather than submit a list of feature names, users can define areas by a set of node numbers copied from the computer-plotted city maps. Nodes are chosen at

FIGURE VII

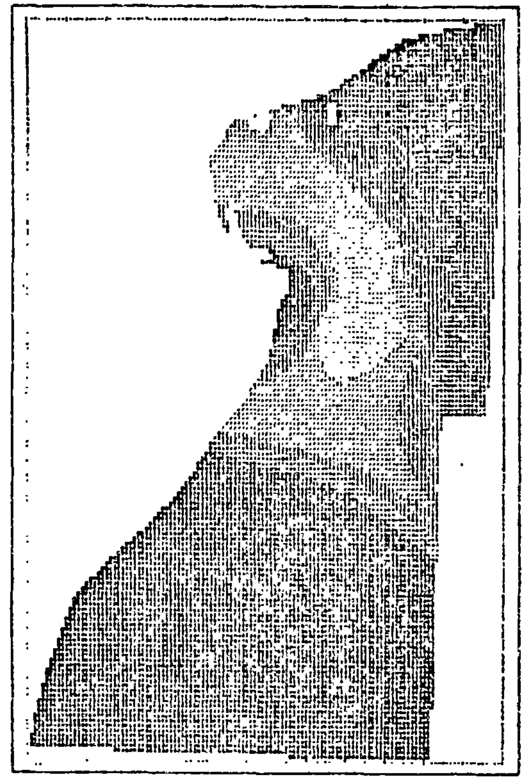
MALE/FEMALE RATIO FOR THE CITY OF SARNIA



(1) CONFORMAL



(2) PROXIMAL



(3) CONTOUR

MAPS ARE ALIGNED IN THE NORTH-SOUTH DIRECTION. BOUNDARIES FOR THE CONFORMAL MAP ARE FORMED BY CENSUS TRACTS. FOR THE OTHER TWO MAPS, BOUNDARIES ARE DETERMINED FROM THE DATA. RATIO VALUES ARE DIVIDED INTO FIVE CLASS INTERVALS BETWEEN EXTREME VALUES OF 0.89 AND 1.13. EACH HIGHER CLASS INTERVAL IS REPRESENTED BY A PROGRESSIVELY DARKER SHADE.

points where the boundary features intersect. Thus, the area perimeter is defined by the nodes, which are matched to the Area Master File before storing the area in the QAL

Using grid coordinates

UTM coordinates can be used to specify query areas in two ways. Data can be retrieved according to a list of individual centroids chosen from the Area Master File. Or, a set of coordinates along a boundary can be used by the system to calculate an enclosed area.

Using area names

Once an area has been submitted using one of the above methods, its name and description are entered and stored, temporarily, in the Query Area Library. For subsequent references the QAL description will be referenced directly by area name, bypassing the map conversion operation.

Of course, all requests for census data by traditional standard areas will also be serviced through the Query Area Library. The QAL contains a pointer set for each province, county, municipality, census tract, enumeration area, and all other standard geostatistical areas used in the 1971 census.

Using other areas

The system permits addition and subtraction of query areas to form a new query area. For instance, a user can outline and request statistics for six areas on a map, naming these areas Area 1, Area 2, ... Area 6. He can then request further data for a new zone, defined as follows:

QZONE1 = Area 1 + Area 2 + Area 3

If Area 5 is contained within Area 6, the following specification would result in a doughnut-shaped query zone

QZONE2 = Area 6 - Area 5

FURTHER INFORMATION

Users who are primarily interested in census statistics using GRDSR may obtain further information by contacting:

User Inquiry Service
Census Division
Statistics Canada
Ottawa, K1A 0T6

Statistics Canada is prepared to provide assistance and further information to users who wish to geocode their own data files. Detailed system documentation will be available in response to technical requests. This information will be provided by a manual entitled *A Technical Description of the GRDSR System*, followed by User Manuals for certain components. For further information of a specialized or technical nature, please contact:

General Survey Systems
Methodology and Systems Branch
Statistics Canada
Ottawa, K1A 0T6

SCOP
Stuttgart Contour Program

K. Kraus
A general digital terrain model — theory and applications

W. Stanger
The Stuttgart Contour Program — description and results

WILD HEERBRUGG

Extract from
F. Ackermann: Numerische Photogrammetrie, Sammlung Wichmann,
Neue Folge Buchreihe Band 5, Karlsruhe 1973

These two papers have been translated and reprinted by Wild Heerbrugg Ltd.,
with the permission of Wichmann-Verlag, Karlsruhe.

A GENERAL DIGITAL TERRAIN MODEL

– Theory and Applications

by K. Kraus, Stuttgart

1. Introduction

The term "Digital Terrain Model (DTM)", previously used principally in connection with specialised problems in road construction, is applied in this paper as a generic term for all problems involved in digital height measurement. In accordance with K. Linkwitz and F. Silar [1], the DTM comprises not only a large number of terrain points whose X, Y and Z coordinates are stored on a digital data carrier and can be read by a computer, but also the computation for processing these recorded data. The theory of this computation is the main subject in the first part of this paper, while the second part deals with the application of this theoretical knowledge and presents an overall concept of how the conventional contour plot can be replaced by a DTM.

2. Theory

2.1 Requirements

The theory must ensure complete freedom in the data acquisition, i.e. provision must be made for the following data-acquisition instruments:

- photogrammetric plotters
- self-recording electronic tacheometers

As regards point distribution, it must be possible to record data

- along profiles, with regular and irregular intervals between points
- along topographically significant structure-lines
- at randomly distributed, topographically significant points
- along contours

In addition to these various point distributions, which may also occur in combinations, provision must be made for a wide range of point densities.

Before and/or during the processing, the measuring errors contained in the data acquired must be eliminated. The main such errors are

- the scanning error, in the case of data acquisition in profiles
- irregular measuring errors at the individual terrain points
- the terrain roughness

The demands made on the theory by the terrain form can be summarised as follows:

- continuous terrain with varying "wave" structure
- terrain break-lines
- displacements at man-made objects such as bridges etc.

Great demands are made on the theory by the different forms of data output. The main forms of output - independent of the data input - are as follows:

- contours from the original measurements
- generalised contours at various scales
- profiles along random planimetric lines
- marking of cut and fill surfaces as the basis for volume computations
- specialised maps (slope maps etc.)

2.2 Short discussion of various theoretical formulations

The central problem of the DTM is one of interpolation. For some points (reference points), both position and height (reference values) are known, while for other points of either known planimetry (interpolation or prediction points) or height, the height and planimetric coordinates respectively must be found.

In the first methods published (list in [1], [2] and [3]), the conventional graphical plotting procedures were transferred more or less unchanged to the computers. The reference points nearest to the prediction point were found, and the contour points then interpolated linearly along triangle or other lines. This procedure was particularly well adapted to computerisation in [4], where the breaking-down of the triangles is subject to the two conditions that the sum of the sides becomes a minimum, and that there are no intersections of the sides of the triangles. Although satisfactory for specific purposes, these methods do not meet the requirements for an overall system, as laid down in Section 2.1. In particular, no measuring errors can be corrected, continuous terrain cannot be adequately approximated, and there is no provision for the various forms of data output.

Other procedures are based on the weighted mean of the heights of the adjacent reference points. The weights can be inversely proportional to the distances or to other parameters. The progress here, compared with linear interpolation, was that the problem was now regarded as one of areas, and not of lines. Apart from that, however, these methods still do not meet the requirements.

This consideration of the problem by area is also the basis of the so-called polynomial approximations. A least-squares adjustment is used to determine the coefficients of a polynomial in such a way that the square sum of the residual height differences h, relative to the polynomial surface, becomes a minimum. A case of this type is illustrated in Fig. 1. Let us take a 2nd-degree polynomial surface:

$$Z = a_0 + a_1x + a_2y + a_3x^2 + a_4y^2 + a_5xy \tag{1}$$

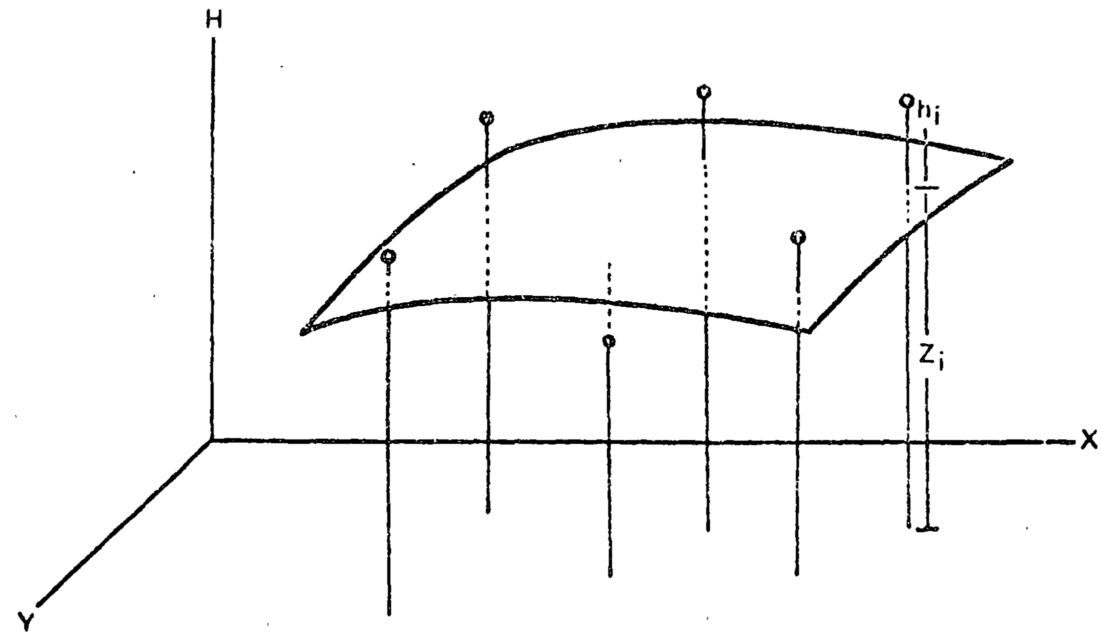


Fig.1

Approximation by polynomial surface

If there are n reference points, there are n "error equations" in the least-squares adjustment:

$$\begin{bmatrix} H_1 \\ \vdots \\ H_n \end{bmatrix} = \begin{bmatrix} Z_1 \\ \vdots \\ Z_n \end{bmatrix} + \begin{bmatrix} h_1 \\ \vdots \\ h_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & y_1^2 & x_1y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_n^2 & y_n^2 & x_ny_n \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} + \begin{bmatrix} h_1 \\ \vdots \\ h_n \end{bmatrix} \tag{2}$$

or, in matrix notation:

$$H = Z + h = Bx + h \tag{3}$$

If we introduce the relationships $H = l$, $B = A$ and $h = -V$, the classic problem of least-squares adjustment becomes particularly evident. In a least-squares adjustment, the "corrections" h must be assumed to be irregular measuring errors. In the polynomial approximation illustrated above, the size of the height difference h depends mainly on the degree of the polynomial - and therein lies the difficulty with this method. If the degree is too low, the polynomial surface is not flexible enough for approximation to the surface of the terrain; if, on the other hand, the degree is too high, the polynomial surface more or less gets out of control in areas with few reference points.

In 1971, at the Photogrammetric Weeks in Karlsruhe, an interesting development of the polynomial approximation was presented by representatives of the Rijkswaterstaat, Delft [5]. Instead of one polynomial of higher degree, the idea is to introduce several partial polynomials of lower degree, which merge continuously with each other at the limits. This gives better approximation to the terrain and avoids the dangerous oscillations which are a particular feature of high-degree polynomials. Another essential improvement is the proposed extension of the minimum condition: in addition to the square sum of the height differences h, some other parameters (e.g. the tilts) of the polynomial surfaces are also minimised. This means that, in areas with few reference points, the polynomial surface tends towards the horizontal plane. By choosing suitable weights, the two components of the minimum condition can be given varying significance, i.e. it might be possible, in this way, to control the weights of the reference-point heights relative to the size of the measuring errors in such a way that it is mainly the measuring errors which are left over. It, is, however, precisely this determination of the weighting which constitutes the problem with this method.

Since the choice of correct polynomial involves certain difficulties, new ways of solving the problem had to be found. In [6], the writer discovered a statistical method which was proposed by Moritz in [7] for the interpolation of gravity values. In the meantime, Koch and Lauer [8], of the Institute for Theoretical Geodesy at the University of Bonn, and other institutes in the United States have proposed that the same theory should be used in other program concepts - some of them very different - for digital contour plotting.

2.3 Derivation of the basic equation for least-squares interpolation

Previously, this basic equation was derived from the minimum variance of the estimation error (see [9] inter alia). Because of this condition, the method was known as "least-squares interpolation". In [10], however, Moritz recently proposed a new derivation based on the familiar minimum principle of the conventional adjustment computation, i.e. $V'Pv \rightarrow$ minimum. In this way, he developed the least-squares adjustment in a "general least-squares model". This gives a generic model which connects the least-squares interpolation, which had previously been somewhat isolated, with the conventional adjustment computation, while at the same time clearly showing the mutual relationships. From now on, the Moritz derivation [10] will be taken as a starting point, and the theory will be applied to DTM requirements.

The reference points - i.e. the terrain heights - are divided into three components (Fig. 2):

a) The trend or determinant component. In geomorphological terms, this is the characteristic form of the terrain. The trend is determined functionally, e.g. in the form of a strongly over-determined polynomial (equation (1)).

b) The correlated component. In geomorphological terms, this is the terrain detail. Because of their variety, these details cannot be determined functionally. In this sector, statistics are employed, based chiefly on variances and covariances. The variance is a measure of the size of the differences relative to the trend surface, while the covariance is an indication of how closely adjacent points are correlated. The closer together the points, the greater the covariance.

c) The irregular component or noise. In terms of our problem, these are the measuring errors and terrain roughness, or else insignificant details which must be eliminated in generalisations. In contrast to the correlated component, there is generally no correlation between adjacent points.

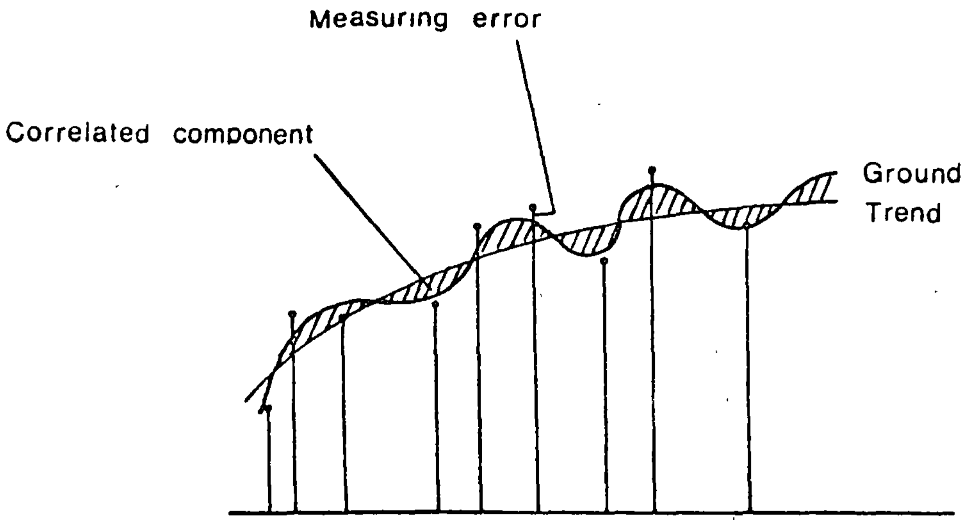


Fig. 2

How can these three components be related and determined on the basis of the measured terrain heights? Conventional adjustment computation can only relate the trend to the height differences h - also known as the centered reference values - (equation (3)), so that this formulation has to be developed twice (Fig. 3):

a) At each of the n reference points, the centered reference value h_i must be divided into the correlated component s_i and the irregular component r_i:

$$h_i = s_i + r_i$$

b) To define the surface of the terrain, the correlated component s_i or s_k must be found for each of a large number N of points. If the trend is known, it is thus possible to find the terrain surface at N points and the irregular components r_i at the n reference points. The N correlated components are combined into the vector

$$S = \begin{bmatrix} s_1 \\ \vdots \\ s_i \\ \vdots \\ s_k \\ \vdots \\ s_N \end{bmatrix}$$

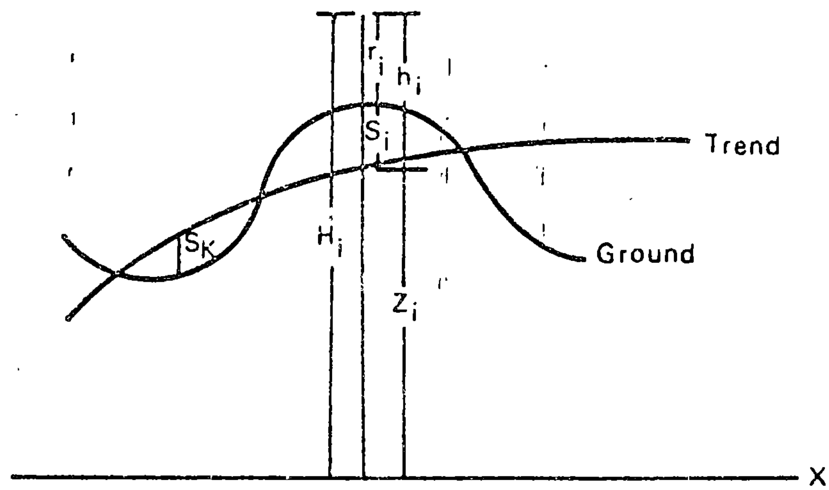


Fig.3

By employing the trick of introducing a null-matrix $O_{n,N}$, the equation (3) of the conventional adjustment computation can first of all be extended - purely formally - to include the correlated components S:

$$H = B \cdot x + h + O \cdot s \tag{4}$$

With the help of the vector

$$v = \begin{bmatrix} s \\ h \end{bmatrix} = \begin{bmatrix} s_1 \\ \vdots \\ s_N \\ h_1 \\ \vdots \\ h_n \end{bmatrix}$$

and the matrix

$$A_{n,N+n} = [O_{n,N}, I_{n,n}]$$

where I is the unit matrix, equation (4) is changed to

$$H = Bx + Av \tag{5}$$

If we now change the term $H = W$, we can recognise the standard problem in adjustment computation, "condition equations with unknowns", which can be solved with the minimum condition $V'PV \rightarrow$ minimum. To do this, P must be replaced by the following variance/covariance matrix in accordance with the two "correction groups" s and h:

$$P = \begin{bmatrix} C_{ss} & C_{sh} \\ C_{sh} & C_{hh} \end{bmatrix}^{-1} \tag{6}$$

If we apply the formal equations pertaining to "condition equations with unknowns", we obtain the result:

$$x = [B'C_{hh}^{-1} B]^{-1} \; B'C_{hh}^{-1} H \tag{7}$$

$$s = C_{sh} C_{hh}^{-1} [H - Bx] = C_{sh} \; C_{hh}^{-1} h \tag{8}$$

This means that we have a two-stage solution. Equation (7), the first stage, gives the coefficients X of the polynomial or trend function. It is important to note here that only the variance/covariance matrix C_{hh} and the original reference values H are introduced, whereas the correlated components S and their related variance/covariance matrices need not be known at this stage. In the second step - equation (8) - we need the centered reference values h, which can be computed as deviations from the polynomial values, and their variance/covariance matrix C_{hh}. The correlation matrix C_{sh} of the correlated components S is also introduced together with the centered reference values h. Since the correlated components S must be determined at a large number of points ($N \to \infty$) in order to assess the surface of the terrain, it is of particular advantage that equation (8) can be evaluated separately for each point to be predicted. If the point to be predicted is included with the reference points, the irregular components (measuring errors etc.) become evident as differences between the centered reference values h_i and the correlated components s_i. This particular procedure is known as least-squares filtering.

2.4 Additions to the theory of least-squares interpolation

Up till now, there have been no stipulations as to how the necessary variance/covariance matrices were obtained when applying least-squares interpolation and filtering to compute a general DTM. In the first stage, equation (7), the variances and covariances have the same significance as in least-squares adjustment. They play a subordinate role in comparison to the second stage, equation (8), which would break down without a knowledge of the correlations.

It is interesting to note that even correlations of the result S with the centred reference values h must be known.

The computation of the variance/covariance matrix C_{hh} and the covariance vector $C_{s_i h}$ for equation (8) from the centred reference values h has already been described in [6] and [9]: Empirical covariances are computed as a function of the distance, and these values are approximated as a covariance function by a bell-curve. The variance V generally appears as a point above the coveriance function (Fig. 4 in the following paper by Stanger). In some cases which will not be discussed further here, a straight line is suitable as a covariance function.

This still does not answer the question of how to obtain, in the first stage (equation (7)), the variance/covariance matrix C_{hh} of the centered reference values h, since these latter cannot be computed until this equation has been solved. In this case, we can make use of one of the findings in conventional adjustment computation - that errors in the correlations and weightings have only a slight effect on the accuracy of the result. In equation (7), we can thus replace C_{hh} with the unit matrix:

$$X = [B'B]^{-1} B'H \qquad (9)$$

The resultant slight errors in the centered reference values h are subsequently corrected in the empirical determination of the covariance function and the evaluation of equation (8). An iterative approximation could also be applied: evaluation of equation (9) followed by empirical determination of the variance and covariance function of the provisionally centered reference values h, and then computation of a provisional variance/covariance matrix C_{hh} which can be processed in a second step in equation (7).

In least-squares interpolation, the choice of degree of polynomial is again of interest. Compared with polynomial approximation, however, the matter is considerably simpler.

Extensive experience has shown that, once the map sheet has been sub-divided into a number of overlapping computing units, a first or second-degree polynomial is sufficient. The decisions can be mace as a function of the recording density and the course of the covariance function. In [11], Grafarend shows this subject, which will only be touched upon here, in an interesting light. For a correlated component s_i, equation (8) can also be expressed as the sum of the centered reference values h_i and the coefficients a_i, the latter being obtained by multiplying $C'_{s_i h}$ with C^{-1}_{hh}:

$$s_i = a_1 h_1 + \dots + a_n h_n = a'h \tag{10}$$

The correlated component s_i is related linearly (linear prediction) to the centered reference values h_i. The particular feature of linear prediction is that correlations between two points are introduced. These are known as two-point correlations. The change-over from linear to non-linear prediction. e.g. in the form

$$s = a'h + h'R\,h \tag{11}$$

leads, among other things, to three-point correlations, i.e. reciprocal correlations between three points. It is immediately evident that non-linear prediction makes lower demands on the quality of the trend function than linear prediction. In many cases, it might even be possible to dispense with equation (7) completely. Since non-linear prediction involves considerably more computation than linear prediction - and this latter is already very computation-intensive in comparison to other methods - linear prediction will probably be retained for a long time to come. The criterion of the three-point correlations can, however, be used to achieve optimum separation of the trend. This is done by increasing the degree of polynomial until there are no more three-point correlations between the centered reference values.

2.5 Different interpretations of the basic equation for least-squares interpolation

In view of the demands laid down at the beginning of this paper for computation equations in a DTM, the basic equations for least-squares interpolation must be interpreted under several aspects. It is sufficient to deal only with equation (8), as the coverage of large-scale geomorphological features with equation (9) does not pose any problems.

2.5.1 Error-free terrain heights and continuous terrain

In the case of error-free reference values or terrain heights, the variance V coincides with the peak value C(0) of the covariance function. This means that the basic equation (8) contains these peak values C(0) along the principal diagonal of the matrix C_{hh}. In this way, the terrain surface to be interpolated is forced to pass through all the reference points. The numerical problems which may crop up in the process will not be dealt with here.

Least-squares interpolation adapts itself automatically to the shape of the terrain, which at this stage should not contain any break-lines or discontinuities. If the centered heights still contain large correlated components, the empirical determination of the covariance function from the centered heights gives a very flat covariance function. If the details of the terrain are more varied, the covariance function is steeper. This automatic adaptation to the shape of the terrain is possible only if the map sheet is sub-divided into relatively small computing units and the covariance function is determined for each unit.

2.5.2 Terrain heights containing errors

If the reference values contain irregular measuring errors, or if the irregularities in the terrain are of importance - e.g. for mapping at very large scales - provision must be made for the variance to exceed the peak value of the covariance function, and the variance V must be introduced in equation (8) along the principal diagonal of matrix C_{hh}. In this way, the terrain surface to be interpolated no longer passes exactly through the measured terrain heights. The measuring errors are filtered out at the reference points. The more the variance V deviates from the peak value C(o) of the covariance function, the greater the value filtered out.

In addition to these irregular measuring errors, the so-called "scanning error" plays an important part when the data are acquired in profiles. Adjacent profiles scanned in opposite directions tend to be slightly displaced relative to each other, particularly at high scanning speeds and in the presence of sharp changes of slope. This component of the scanning error can also be eliminated through least-squares interpolation, by combining all the profiles running in one direction into one group of reference points, and the profiles running in the other direction into a second group of points. Statistically, the two groups differ from each other in that the (auto-) correlation within each group is greater than the (cross-) correlation between the two groups of reference points. These auto- and cross-correlations can be arranged in equation (8) in such a way that the terrain surface to be interpolated is free of that component of the scanning error [12].

2.5.3 Generalisation

Filtering can also be used for generalisation of the surface of the terrain, the degree of generalisation being controlled by the difference between the variance and C(o). The following forms of generalisation can be applied:

- Uniform generalisation independent of the shape of the terrain. With σ_k as the mean generalisation value, the distance between the peak value C(o) and the variance V is obtained from

$$C(o) = V - \sigma_k^2$$

- Relative generalisation as a function of the shape of the terrain. The variances of the centered terrain heights - assuming that the polynomial degree of the trend function is the same in each case - are an indication of the size of the details in the individual computing units, i.e. the variance in irregular terrain which has to be strongly generalised is greater than in simpler terrain. A fixed relationship between the variance and the peak value of the covariance function thus causes the degree of generalisation to increase together with the degree of difficulty of the terrain.

- Accentuation of geomorphologically significant points and structure-lines. The first two forms of generalisation simplify the terrain within each computing unit according to the same law of formation. In terms of equation (8), this is expressed by a uniform variance V along the principal diagonal of matrix C_{hh}. At such points, and along those structure-lines which are contained in the generalised picture and which must be accentuated to a greater or lesser degree in comparison to their surroundings, the distance between the variance and the peak value of the covariance function must be reduced. This is an imitation of the cartographer's manual generalisation.

In practice, all three forms of generalisation will be applied simultaneously. They are intended primarily for generalisation of the terrain surface and only secondarily for all the values derived from this. This surface generalisation was recently described by Gottschalk [13] as possessing many advantages. It should be pointed out, in this connection, that we have been carrying out surface generalisations of this type for about six months now.

2.5.4 Terrain break-lines

The statistical definition of a terrain break-line is that there are no- or only low - correlations across these lines. If this criterion is applied to the basic equation (8), the covariance between those points joined by lines intersecting a break-line must be set to zero or suitably reduced. Reference points marking the break-line are correlated with points on both sides. In this way, it is possible to obtain the characteristic sharp bends in the terrain at erosion furrows and man-made objects.

2.5.5 Discontinuities

Discontinuities in the terrain can be taken into account in the same way as break-lines. In this case, however, reference points along the line of discontinuity may be correlated with points on either one <u>or</u> the other side of the line, depending on whether they are on the upper or lower surface. Discontinuities of this type occur at retaining walls and bridges, as well as along lines of dislocation in the case of geological deposits.

3. Applications

3.1 Data acquisition

The theory presented here places no restrictions on the method of data acquisition or on the point distribution. The only requirement is that the terrain information must be contained in the reference points. The data acquisition instrument can be either a self-recording tacheometer or a photogrammetric plotter - or else both of these combined.

Self-recording tacheometers are preferred for small areas in particular, but they also play a certain role in the mapping of forests at large scales - e.g. for the 1:5000 German Standard Map. A description of how these instruments can best be employed for data acquisition is given in [14].

Photogrammetric plotters have more universal applications, one of the main fields being that of data acquisition during orthophoto production. In this case, the data for the DTM are obtained more or less as a by-product. To improve the quality, the terrain points in the profiles are supplemented by measurements along geomorphological structure-lines and by individual terrain heights at topographically significant points.

Quite apart from the field of orthophoto production, however, measurement of the stereomodel in profiles, instead of plotting the contours, is gaining increasing importance. A model can be scanned in profiles in about half the time it takes to plot the contours, and this ratio in favour of profile-scanning can be considerably improved if the scanning speed is increased and one deliberately accepts a larger scanning error - which is subsequently eliminated by means of the filtering technique mentioned above. Even in the long term, profile-scanning has the advantage that it is much easier to automate completely than direct scanning of the contours.

In addition, least-squares interpolation is also suitable for using digitised contours as initial data for computation of grid heights, profiles etc. This results in a general system in which, depending on the circumstances, either contour or profile points can be stored, and the one then derived from the other as required.

In flat terrain, point-by-point scanning of the terrain in a more or less regular grid has been the preferred method in photogrammetry for some time now. Even in wooded areas, this point-by-point registration can often be applied when direct plotting of the contours fails. In such difficult conditions, it is easier to measure individual points stereoscopically than to follow contours in the stereomodel. The conditions when recording heights in densely populated areas are often similar.

This general data acquisition concept also takes account of the photogrammetric plotters of the future. Of particular interest in this respect is the Epipolar Scanner [15] developed by Bendix. This instrument produces, inter alia, an irregular collection of points with X, Y and Z coordinates.

3.2 A DTM as a substitute for a conventional contour map

The benefits of electronic computation can be used to particular advantage if, instead of just processing the data for the one special purpose in mind when the data were being acquired, the original data or the final results - or even intermediate results containing the entire information - are stored on a peripheral data carrier where they are later available in digital form as initial data for other problems. This is the idea behind the so-called "data banks".

The many and varied purposes of a topographical contour map, for instance, can, in principle, be transferred to a data bank of this type. Some of the most important purposes were given at the end of Section 2.1, although this list is not claimed to be complete. It is not an economical propositon to develop different programs for each form of data input and output. It is better to set up a high-performance central program system which responds to all forms of data acquisition and in which the first step edits the data in such a way that it is a simple matter to obtain the various forms of output.

Because of its independence of the point distribution and facilities for filtering, least-squares interpolation can be adopted as the basic theory for this central program system. The contours favoured as basic information in our topographical maps, however, are not suitable as an output from the central program system. The main disadvantages of contours are the sharp variation in information content as a function of the terrain slope and the over-accentuation of the information along the contours.

The heights of a very close-meshed square grid are much better suited for automation than contours, and are much more suitable for representing the shape of the terrain. Independent of the form of data input and output, the central program system has thus been designed in such a way that the heights of a very close-meshed square grid are first of all interpolated by least-squares interpolation and stored digitally on a peripheral data carrier. To assume the functions of the standard topographical map, a grid mesh of about 10 m must be chosen, this being equivalent to two millimetres at the map scale of 1:5000. A magnetic tape with the recorded grid heights is extremely easy to handle as a digital terrain model for a wide variety of purposes. Because of the high information density, it is sufficient to operate with linear interpolation. If, in the rest of this paper and in Stanger's paper, the abbreviation DTM is used, it often refers to the grid points and the relatively simple computation. The computation time involved in interpolating this grid - including filtering the initial data - accounts for approximately 4/5 of the computation time for the total process. From this point of view as well, it is thus advisable to compute the square grid once only, to store it, and then to use it in solving the various problems.

If the grid heights, as a DTM, are to take over the functions of the conventional contour map, suitable data organisation systems and archives must be available. As Stanger describes in his paper, this grid is obtained in the form of oriented map sheets in the ground coordinate system. The relationship to the previous line and orthophoto maps would be particularly well ensured if a magnetic tape were prepared and filed for each map sheet. In the case of Baden-Wurttemberg this would amount to 9000 magnetic tapes. Although the storage of the 40 000 grid heights in a Standard Map Sheet of 2 km x 2 km takes up only a fraction of the capacity of a magnetic tape, it is better to be over-generous in this respect. These grid heights must always be supplemented by the geomorphological structure-lines and the spot heights at topographically significant points. Those contours which are likely to be recalled most frequently from the DTM can also be stored. In addition, space must be reserved for the planimetric information. Similar concepts have recently been developed by Brindöpke [16].

Finally, an illustration of how this square grid can be utilised as a DTM for various purposes:

3.3 Contours for the standard topographical map

The problem of how to measure and sort the contour points in this square grid in such a way that the contour lines can be plotted automatically on an automatic plotting table and provided with the familiar cartographic additions is dealt with in Stanger's paper, which also contains an example. It is merely pointed out here that, depending on the purpose involved, the contour plot can be produced at different scales, with different contour intervals and in different map segments - even in segments made up of several "map sheets". In this, our program system is supplemented by the software of the automatic plotting table used.

3.4 Generalised contours

Generalisation of the contour plot is necessary for reproduction at smaller scales. As already explained, a DTM of the generalised terrain surface is first of all produced, and the contour lines contained in this fixed. Appendix 1 shows the generalisation of a contour plot for the scale 1:5000. The segment marked is reproduced at a scale of 1:2500 in Stanger's paper. An even stronger generalisation for the scale 1:10 000 is shown in Appendix 2. Since the programs for the 3rd generalisation principle given in Section 2.5.3 have not yet been developed, the spot heights at the various scales deviate from each other by the "generalisation amounts".

3.5 Profiling

As will be explained in the following paper, it is relatively easy, in this DTM, to compute the required profile heights for random planimetric lines and to plot them on the automatic plotting table. The importance of such profiles in civil engineering need not be emphasised here, but it is interesting to point out some special applications.

For the investigation of radio-telephone links, for instance, all the line-of-sight profiles can be plotted automatically.

Of particular importance for orthophoto production is the fact that the contours can be digitised, and that these data can then be introduced into the program and used to compute the profiles for a digitally-controlled orthoprojector [17] .

For revision of orthophoto maps, there is the possibility of computing the DTM from the registrations for the older profile-by-profile orthophoto plot, and of then storing the DTM. This can subsequently be used to compute the new profiles in the orientation of the new photo combination. This means that revision of the orthophoto maps is fully automated. The flight planning for the repeat flight is independent of the first photo-flight. Here, too, a digitally-controlled orthoprojector is essential.

As the last application in conjunction with a digitally-controlled orthoprojector, it should be mentioned that the profile-scanning can be speeded up considerably if the profiles are first of all recorded, and the scanning error is then eliminated with the help of this program [12] . The improved profiles can then be used to control the orthoprojector.

3.6 Cut-and-fill traces and volumes - DTM revision

After changes in the surface of the terrain - particularly those resulting from engineering projects - a new grid can be used to produce a DTM in the same way. The grid interval should be the same, so that the height of both the old and the new DTM is obtained for each grid point. Before replacing the old DTM by the new one on the magnetic tape, one first of all plots the intersection lines of the two terrain surfaces - traces of cut and fill - and computes the changes in volume. Similar problems occur in studies of geological deposits and in open-cast mining.

This method also solves the problem of revising the DTM and the forms of representation derived from it. After revising the DTM, for instance, the contour sheet for the whole map sheet can be re-scribed, so that the contour manuscript for a new edition of the topographic map is obtained fully automatically.

3.7 Slope map

Stanger's paper describes how the DTM can be modified so that it produces, at the grid points, the slopes of the terrain surface instead of the terrain heights. From these values, it is possible to derive the lines of equal slope, and these can then be plotted graphically by the automatic plotting table. This "slope map" is of great importance in land redistribution. In addition, its value in other fields must also be emphasised. In civil engineering, for instance, it can be used as an additional output to the contour map to avoid the time-consuming task of manual derivation of slopes from contours. Slope maps may also be of use to geographers and geologists for terrain studies.

3.8 Further applications

The list of possible applications for such a DTM can be extended almost at will. In conclusion, it should be pointed out that many program developments in civil engineering (e.g. [18]) are based on a square grid of this type, and thus fit in closely with this overall concept. Some of the main applications should be mentioned:

- Perspective and axonometric representations of the terrain.
- Output of two contour plots with artificial parallaxes for stereoscopic viewing. If this pair of contour plots is superimposed on stereo-orthophotos, stereo-orthophoto maps are obtained.

4. Summary and Prospects

With the demands made on a general digital terrain model as a basis, a theory has been developed which imposes no restrictions on the mode of data acquisition, and of which the principal intermediate results are the heights of a square grid. In the subsequent processes, this special DTM takes the place of the earlier standard topographic map. It is the information carrier for fully automatic derivation of contours at various scales, for the derivation of profiles and earth-movement volumes, as well as for various other forms of representation. In view of the high capabilities of computers and automatic plotting tables, future contour maps of an area will be accompanied by various specialised maps. Revision of these maps poses no problems - the old DTM is replaced by a new one, and the various maps in each sheet are then derived from the new DTM by computation.

The combination of digital contour plotting with digital plotting of the planimetry will give a decisive impulse to the further development of the program system. The planimetry also contains information which is of interest for height representation. The breaking-off of contours at buildings or bridges, for instance, can then be taken into account automatically in the contour plot. In addition, it will be possible to improve the automatic geomorphological editing at the edges of roads and waterways.

Bibliography:

[1] Linkwitz, K.: Digitale Geländemodelle, BuL 38, 76-84, 1970

[2] Schrader, B.: Die automatische Herstellung von Höhenlinienplänen; Sammlung Wichmann, Neue Folge, Heft 15, 96-103, 1971

[3] Brunken, G.: Automatische Zeichnung von Höhenlinienplänen, DGK, Reihe C, Heft 173, 1972

[4] Düppe R. und Gottschalk, H.: Automatische Interpolation von Isolinien bei willkürlich verteilten Stützpunkten, AVN 77, 423-426 1970

[5] Bosmann, E., Eckhart, D. und Kubik, K.: Delft - A Program System for Surface Approximation, BuL 40, 13-20, 1972

[6] Kraus, K.: Automatische Berechnung digitaler Höhenlinien, ZfV 96, 233-239, 1971

[7] Moritz, H.: Statistische Methoden in der gravimetrischen Geodäsie, ZfV 88, 407-416, 1963

[8] Koch, K. und Lauer, S.: Automation der Isoliniendarstellung mit Hilfe des Wiener- und des Kalman-Filters, Mitteilungen a.d.Inst.f.Theor.Geodäsie der Univ. Bonn, Heft 2, 1971

[9] Kraus, K.: Interpolation nach kleinsten Quadraten in der Photogrammetrie, BuL 40, 3-8, 1972

[10] Moritz, H.: A Generalized Least-Squares Model, Studio geoph. et geod. 14, 353-362, 1970

[11] Grafarend, E.: Nichtlineare Prädiktion, ZfV 97, 245-255, 1972

[12] Kraus, K.: Prädiktion und Filterung mit zwei verschiedenen Stützpunktgruppen, ZfV 98, Heft 4, 1973

[13] Gottschalk, H.: Die Generalisierung von Isolinien als Ergebnis der Generalisierung von Flächen, ZfV 97, 489-494, 1972

[14] Kraus, K.: Rationalisierung der tachymetrischen Geländeaufnahme und Automatisierung der Weiterverarbeitung zur grossmassstäbigen Karte, AVN 80, Heft 1, 1973

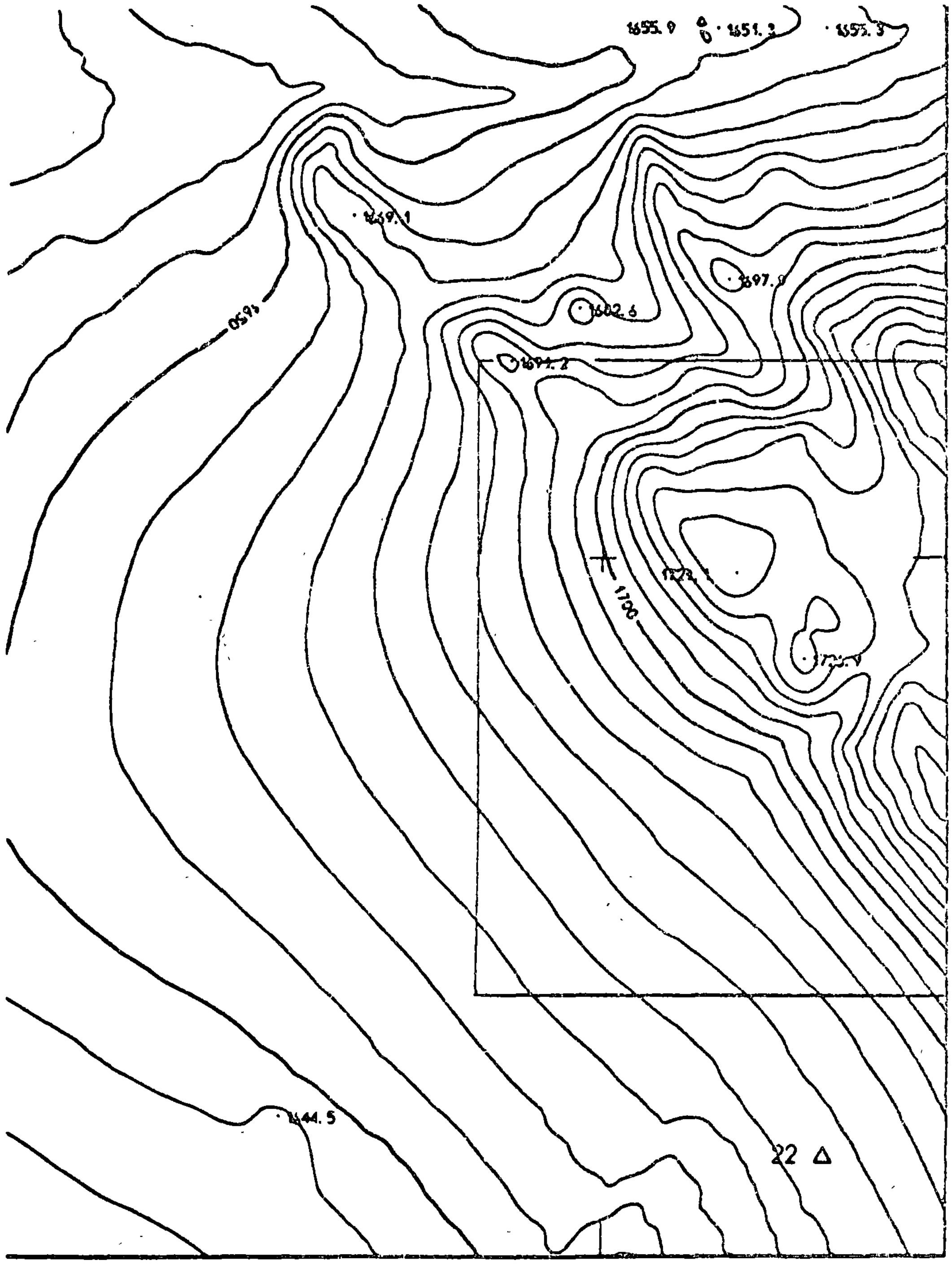
[15] Helava, U. und Chapelle, W.: Epipolar-Scan Correlation, Bendix Technical Journal, Vol. 5, No. 1, 19-23, 1972

[16] Brindöpke, W.: Systemplanung in der topographischen Photogrammetrie, AVN 79, 149-259, 1972

[17] Finsterwalder, Rü.: Zur Gewinnung von Profilen aus Schichtlinien zum Zwecke der Differentialentzerrung, ZfV 97, 193-196, 1972

[18] Benner, E. und Schult, R.: Report on a Digital Terrain Model, Presented paper, XII. International Congress of Photogrammetry, Comm. IV, Ottawa, 1972

[19] Blachut, T.: Mapping and Photointerpretation System Based on Stereo-Orthophotos, Mitteilungen aus dem Inst. für Geodäsie und Photogrammetrie der ETH Zürich, Heft 14, 1971

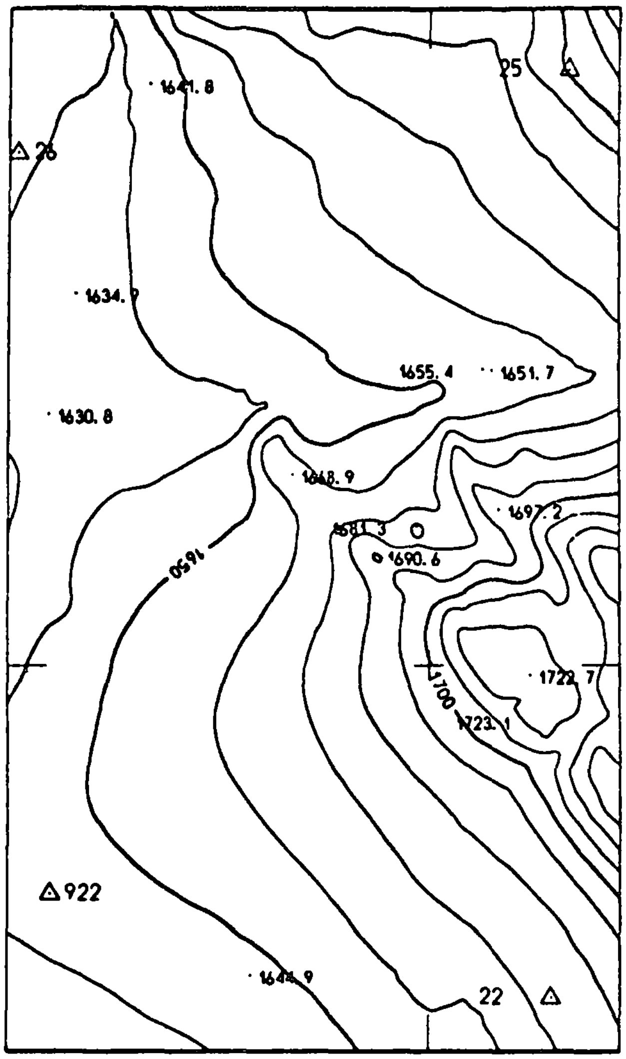


Appendix 1

Automatic generalisation

1:5000

The area enclosed in the box is shown in the original plot scale of 1:2500 in Appendix 1 of the paper by Stanger



Appendix 2

Automatic generalisation

1:10 000

THE STUTTGART CONTOUR PROGRAM - DESCRIPTION AND RESULTS

by W. Stanger, Stuttgart

1. Introduction

In his paper, "A general digital terrain model", Kraus illustrates various ways of processing digital height information. Of the greatest importance in actual practice is the computation of contours from randomly distributed terrain points.

In practical photogrammetry, the data can be obtained as a by-product during the profile scanning of a model to produce an orthophoto. This scanning in profiles is approximately twice as fast as direct scanning of the contours in the model [1]. Apart from this, profiles in flat country or in forests have a higher accuracy than contours obtained directly, and to some extent they can nowadays be scanned completely automatically with the help of correlators. In principle, it is already possible to find a flexible software solution to the problem of computing contours from randomly distributed terrain points, now that very fast and large computers are available as well as high-performance data recorders and automatic plotting tables.

This was the background to the cooperation with Wild Heerbrugg, in Heerbrugg, and Contraves in Zürich. A combined system was developed, in which the data are collected by the Wild EK8 and processed by the Stuttgart Contour Program, while the contour map is plotted on the Coragraph DC2. This ensures that the systems are adapted to one another at the interfaces.

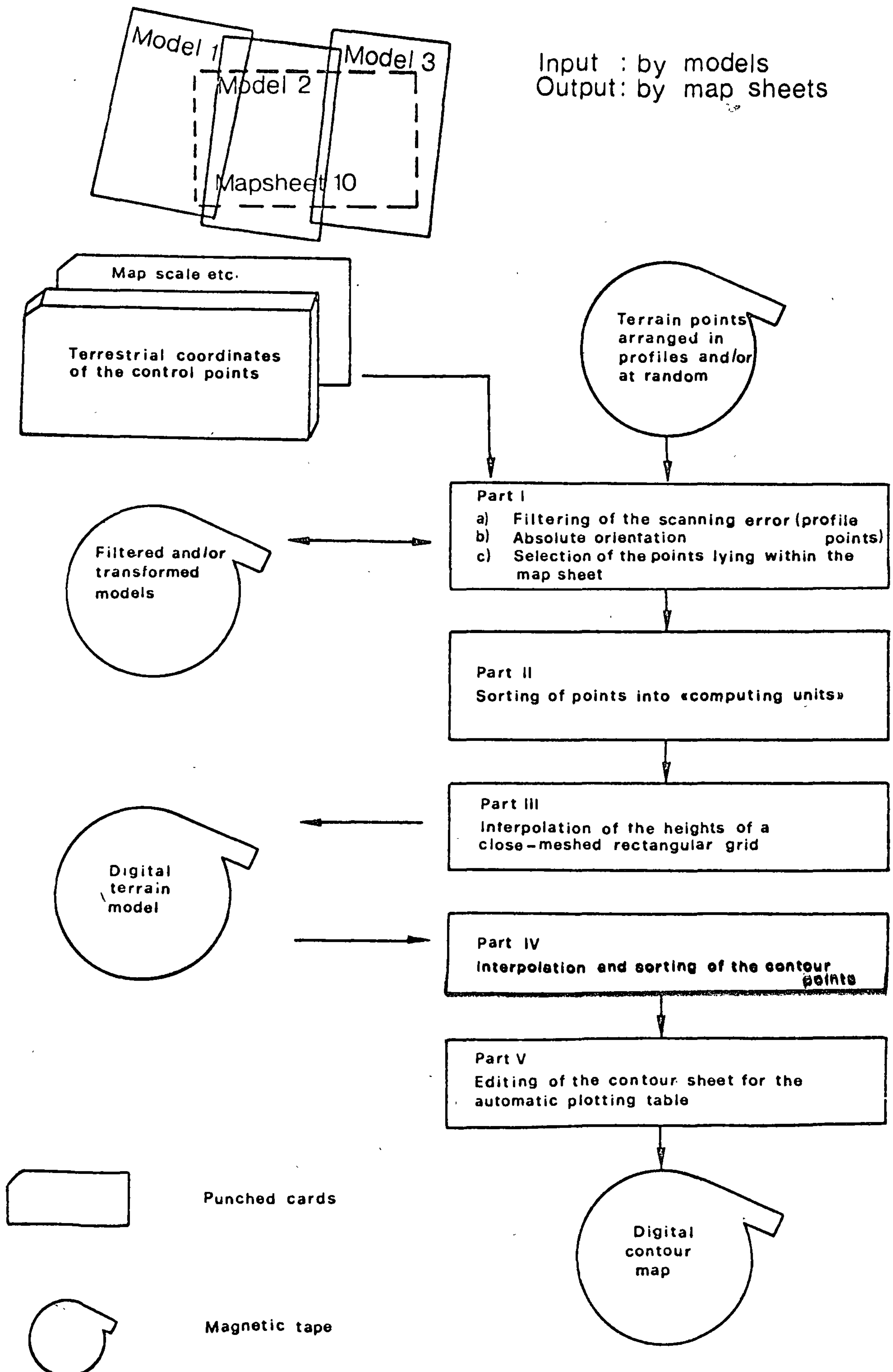
In its method and concept, however, the Stuttgart Contour Program has been kept general, so that it is not restricted to one particular configuration.

2. Program Description

Purpose of program system:

Computation of contours from randomly distributed terrain points.

2.1 Flowchart of the program



2.2 INPUT of Program System

The input is by models. In the overlap areas of the models, terrain points of all the models involved are included in the computation to prevent jumps at the edges of the models. In the same way, some terrain points lying outside the edge of the map sheet are included in the computation to avoid irregularities at the edge of the sheet. The input is divided into model data and map-sheet data. The model data - basically the terrain points with X, Y and Z coordinates - are recorded on magnetic tape (e.g. with the Wild EK8). There are no restrictions on point density or choice of points. The program will accept: (Fig. 1)

- Regularly and irregularly spaced profile points
- points selected by the operator along geomorphological structure-lines
- spot heights of particular topographical importance.

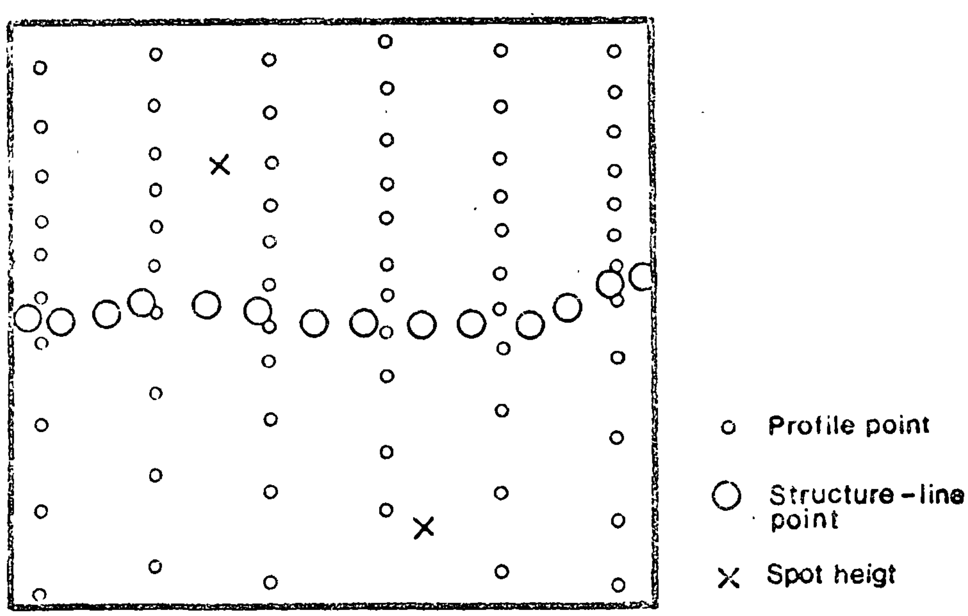


Fig. 1

Input of model data

The map-sheet data, which are fed in on punched cards, comprise:

- map-sheet number and coordinates of border lines
- map scale
- contour interval and interval between index contours
- control-point coordinates in ground system.

2.3 Part 1

Purpose: Filtering of the systematic scanning error in the case of profile-data compilation. Absolute orientation of the models. Selection of the terrain points lying within the map sheet.

Input: Terrain points recorded in profiles, with additional structure-line points, spot heights and map-sheet data.

or: Individual points which need not be filtered (e.g. individual, photogrammetrically measured points in flat terrain).

One by one, those models are located which cover the map sheet and which may be on different magnetic tapes. Systematic height errors caused by reversals of scanning direction in profiling can be eliminated in this part of the program.

The profile data are sorted into "computing units" (CU) (see Section 2.4), separated into forward and reverse profiles. A statistical analysis of both groups of profile points provides information on whether or not such a scanning error is present and, if present, its magnitude. If a scanning error is detected all points in the gross CU (see Section 2.4) are used in the method of filtering by least squares [2] to clear the error from both groups of profile points within the nett CU (see Section 2.4).

There follows the transformation of the model coordinates into the ground coordinate system. A two-dimensional, planimetric transformation is performed, followed by a scaling and mean shift of all height control points. Finally, the model is tilted in both coordinate directions so as to make the sum of the squares of the residuals on the height control points a minimum. It is assumed that an absolute orientation has been performed (spot heights in topographically important points can only be placed properly in a levelled model) so that this final correction is limited to small "linear" rotations.

If the models contain data to be filtered and/or transformed, the filtered and/or transformed data are written out in Part 1 onto magnetic tape. This tape can then be used as an input to Part 1 if, for example, some of the data are to be used in neighbouring map sheets.

Part 1 of the program selects from all models those points lying within the specified map sheet. In order to ensure that there are no jumps at the edges of the sheet points lying in zones outside the edges of the sheet are also selected.

If the terrain points have been observed individually rather than in profiles, (e.g. in forest areas), the filtering is bypassed. The coordinates are merely transformed and stored by map sheets.

Output (disc): - Filtered profile points, or unfiltered individual points, structure lines and spot heights, in the ground coordinate system.

(tape) - If filtering and/or a transformation has been performed, the filtered and/or transformed coordinates on magnetic tape.

2.4 Part 2

Purpose: Sorting of terrain points into computing units

Input: Output from Part 1.

Since there may be several thousand reference points (i.e. recorded points), in one map sheet, and only reference points in the immediate vicinity can usefully be employed to predict a desired height, the map sheet is sub-divided into smaller parts - so-called computing units (CU). Each of these contains an average of 70 reference points, and it is only when the number is reduced to this extent that the subsequent computation becomes feasible. The CU's are mutually overlapping, so as to ensure the connection to neighbouring CU's (Fig. 2).

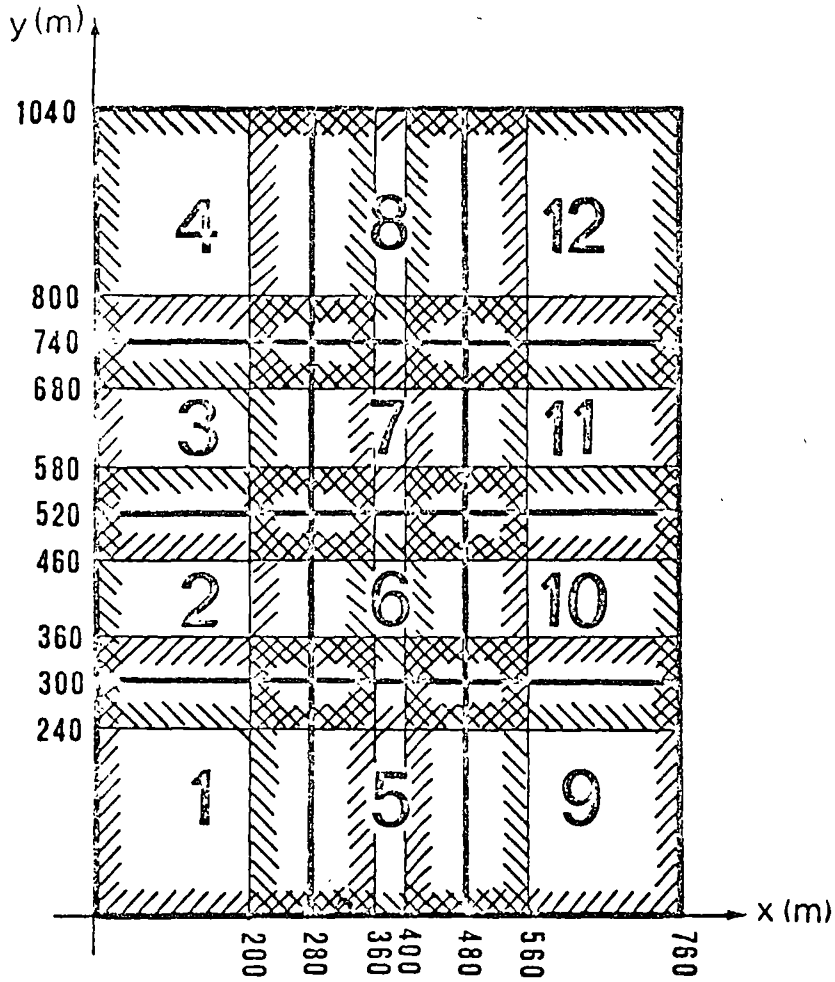


Fig 2

Subdivision of map sheet into overlapping computing units

The CU's will henceforth be referred to as "nett" (excluding overlap), and "gross" (including overlap).

The size of a CU is a function of the profile width and the mean point density in the profiles. It remains the same for the whole map sheet. The size of the arrays for these CU's is such that a wide range of recording densities and a large number of structure-lines can be processed. If, in spite of this, the point density should ever rise too sharply, the program automatically effects a data reduction by interpolating means. This ensures that, when the profiles are being scanned, the operator can vary the scanning speed to suit the terrain as well as possible.

The measured terrain points are stored in the arrays of the individual computing units as reference points, while strips of the CU's are combined into blocks and processed jointly. This means that the map sheet can contain any number of CU's. The individual CU-blocks are then filed on disc.

Output (disc): Reference points combined into blocks of overlapping CU's.

2.5 Part 3

Purpose: Interpolation, over the entire map sheet, of the heights of a regular, close-meshed grid.

Input: Output from Part 2

The grid heights are computed by CU's, i.e. only the reference points for one CU must be in the core storage of the computer. First of all a uniform trend (centering) is determined for each CU, without checking the quality of the centering by a three-point correlation. The centering is effected with a polynomial surface of degree 0, 1 or 2. The desired degree of polynomial is controlled through input parameters; if the covariance function computed with these does not satisfy certain criteria, the program reduces the degree of the polynomial and repeats this section. The polynomial is fitted to all reference points by a least-squares adjustment. Reduction of all reference-point heights in the computing unit by this polynomial gives the centered reference-point heights h (Fig. 3).

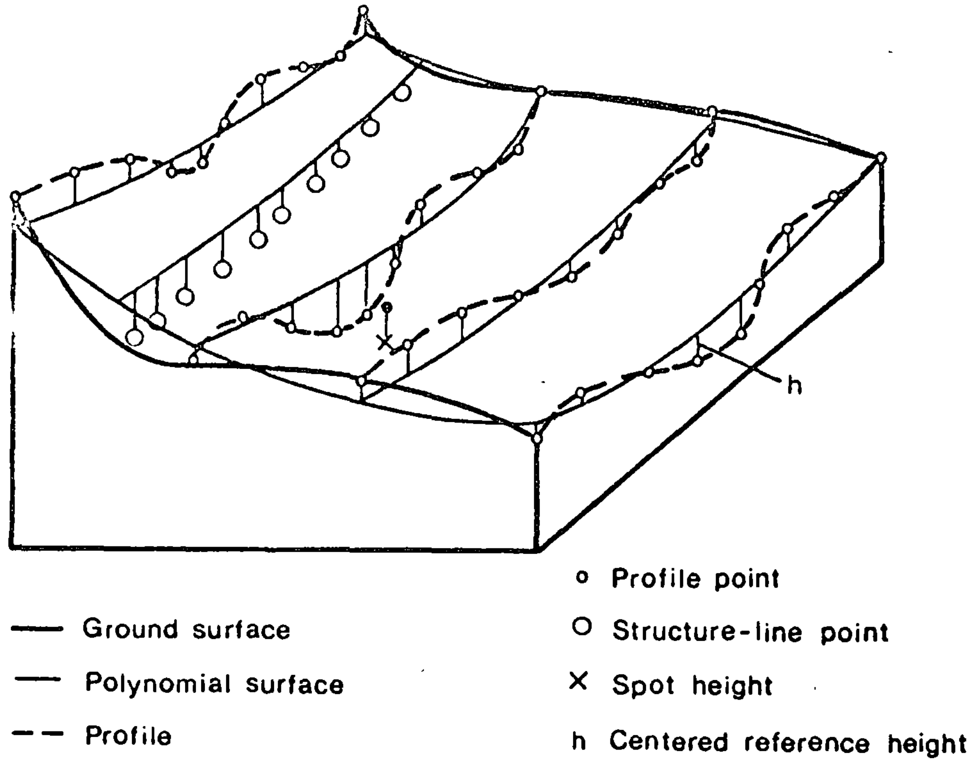


Fig.3

Determining a polynomial surface through the reference points in the CU

The next step is the empirical determination of a covariance function as a bell-shaped curve for each CU from the centered reference-point heights. This is done by computing the variance and the covariances for a few distance intervals (Fig. 4).

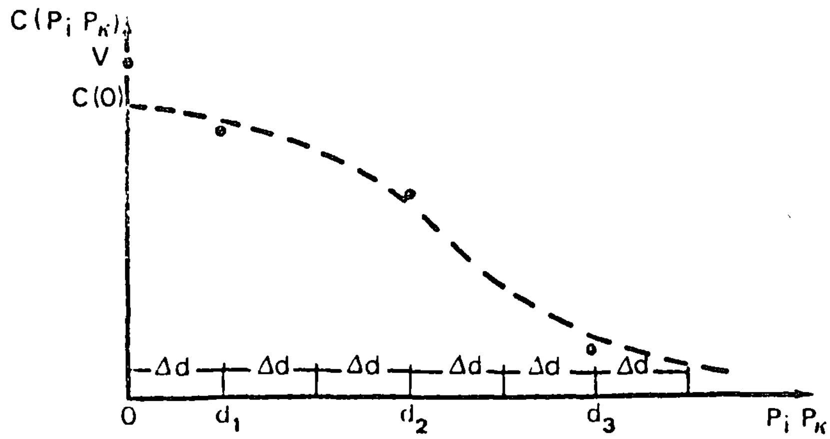


Fig. 4

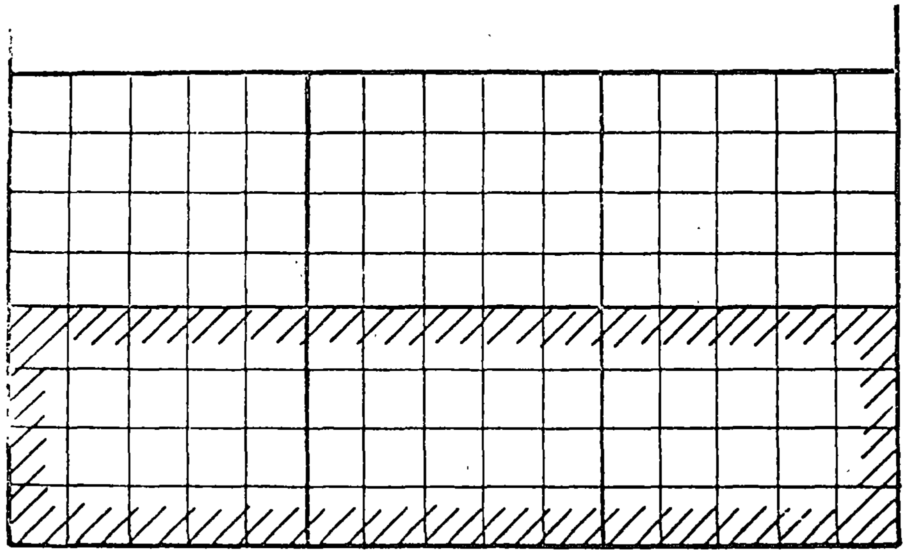
Empirical determination of the covariance function

In the case of a CU with low reference-point density, the covariance function is taken to be a straight line. This step is followed by the part involving the largest amount of computation - least-squares interpolation, in the CU, of the heights of a regular grid. The grid runs parallel to the edges of the map sheet. It there are n randomly distributed reference points P_i with reference values h_i, we obtain an interpolated value of h' for a random point P from equation (1) (equation (8) in the preceding paper, but with s_i designated h'):

$$h' = \left[C(\overline{PP_1})\ldots C(\overline{PP_n})\right] \begin{bmatrix} V & C(\overline{P_1P_2})\ldots & C(\overline{P_1P_n}) \\ C(\overline{P_1P_2}) & V & C(\overline{P_2P_n}) \\ \vdots & & \vdots \\ C(\overline{P_1P_n}) & C(\overline{P_2P_n})\ldots & V \end{bmatrix}^{-1} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{bmatrix} = c \cdot C^{-1} h \quad (1)$$

The row vector C' contains the covariances $C(\overline{PP_i})$ between the interpolation point P and the n reference points P_i, while the C-matrix contains all covariances $C(\overline{P_iP_k})$ between the reference points and, along the principal diagonal, the variance V. The covariances in C and C are computed with the help of the distances $\overline{P_iP_k}$, which are entered into the covariance function. The column vector h contains the centered reference-point heights.

Once the centered height has been computed for each grid point by solving this basic equation, and the polynomial value has been added to it, the heights of the CU edge lines (Easting constant) are meaned with the heights of the identical edge line in the following CU (Fig. 5).



—— Edge of map sheet

—— Edge of computing unit

—— Grid lines (interpolated grid)

////// CU strip

Fig.5

Meaning the grid heights at the edge lines of the CU

Once a complete CU strip has been computed, the grid heights are stored in profiles on discs as a digital terrain model. The upper edge profile of a CU strip is not stored, however, until it has been meaned with the lower edge profile of the following CU strip.

Finally, and in spite of the fact that their higher accuracy has been taken into account in the program, the spot heights are filtered, so that their height agrees with the terrain model - and hence with the contours.

Output (magnetic tape): Grid heights in profiles (digital terrain model)
Filtered spot heights

2.6 Part 4

Purpose: Interpolation of contour points in the regular grid and sorting of these points to form plausible contour elements in the CU.

Input: Output from Part 3.

or: Terrain points recorded directly in a sufficiently close-meshed grid.

All the profiles needed for a CU strip are read-in, and from these profiles are derived the grid heights forming the CU to be processed. Because of the high density of the grid heights, the interpolation of all contour points in the CU-grid can be linear (Fig. 6).

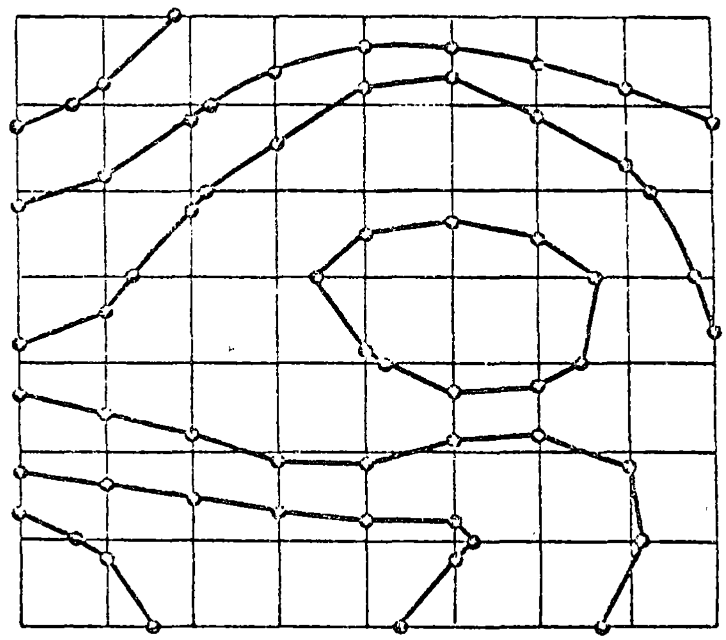


Fig.6

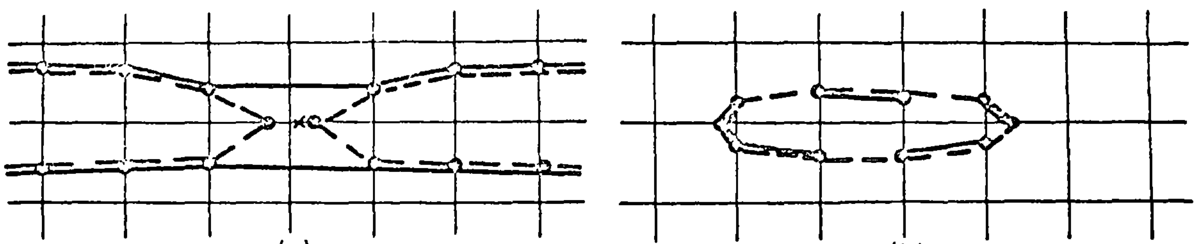
Interpolation of discrete contour points in the grid

In principle, the interpolation in the grid takes place in the direction of greatest height difference - which is more or less equivalent to interpolation along the fall-line. If, however, the height differences to the east and north agree to within a certain fixed percentage, the interpolation is in both grid directions, so as to increase the information density along the contour line.

The following criteria could be applied to join these contour points into contour elements:

- starting from one point, the point the shortest distance away
- starting from two points, the point with the smallest change in direction and a short distance away.

In certain cases, neither criterion on its own gives a plausible solution (Fig. 7).



(a) Direction criterion

(b) Distance criterion

– – – – Real contour

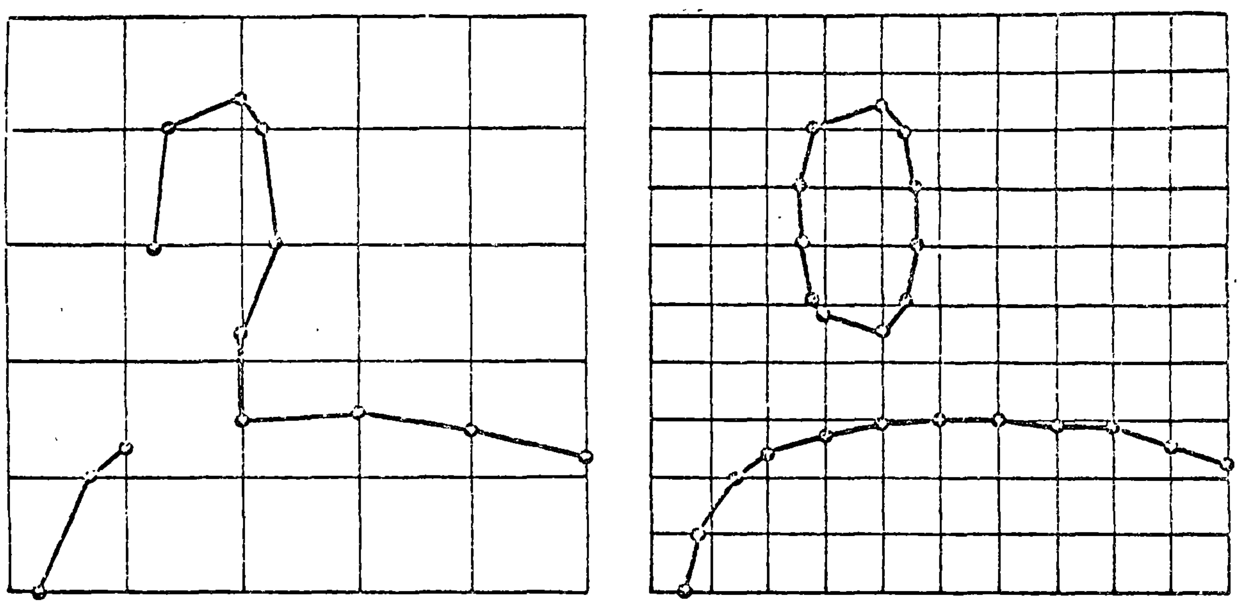
———— Contour according to criterion

X Spot height

Fig.7

Joining the contour points to form plausible contour elements

To keep the computing time as short as possible, the program is based essentially on the distance criterion, but with an upper limiting distance between neighbouring points which is related to the mesh of the grid and which may not be exceeded. In addition to this, the program caters for important special cases. If the discrete points cannot be joined to form plausible contour elements in the CU, the information density in the CU must be increased by halving the grid mesh. After renewed linear interpolation in the halved grid, there are correspondingly more discrete contour points (Fig. 8).



a) No plausible contour in original grid

b) Plausible contour after halving the grid interval

Fig. 8

Increase in information density by halving the grid interval

This makes it possible to adapt the amount of computation to the difficulty of the terrain. If necessary, the grid can be halved yet again, so that the amount of height information in the grid is increased 16-fold.

It is also necessary to consider two types of contour within a CU - open and closed contours. An open contour begins with an intersection with a CU edge-line and also ends with such an intersection. When all the open contours have been assembled, the remaining points must all belong to closed contours. These begin at some unspecified point within the CU and return to this starting point.

If an open contour does not end at a CU edge-line, or if, in a closed contour, the distance between the first and the last points is greater than the limit mentioned above, no plausible contour segments can be found, and the grid mesh is halved as shown in Fig. 8 and the program section repeated.

When the contour elements are being stored on disc, they are sorted according to the height of the element, so that elements of the same height are stored together.

Output (disc): Contour elements sorted according to height.

2.7 Part 5

Purpose: Connection of the contour elements in the CU's to form continuous contours in the map sheet.
Editing of the contour plot for the automatic plotting table.

Input: Output from Parts 3 and 4.

The contour elements pre-sorted in Part 4 are assembled in ascending order of height, beginning with the lowest contour in the map sheet. All initial and terminal points of the elements are situated on CU edge-lines, with the exception of the closed contours in a CU. The division of the contours into open and closed, referred now to the map sheet, corresponds to the division in the CU. This means that the criteria for successful connection are similar.

The heights of the index contours can be printed at the intervals programmed or fed-in on punched cards. The curvature of the contour at the intended point is first checked by the program. If it exceeds a pre-determined limit, a new point is chosen close by and a new check is carried out. Four such displacements are incorporated in the program. To ensure the topographically correct direction of printing, there is a cast-back to the digital terrain model, with its grid of heights. The heights themselves are printed in circular arcs corresponding to the curvature of the curves. Contours and index-contour heights are given control signals for the automatic plotting table and recorded on magnetic tape. The automatic plotting table joins the contour points according to the control signals, e.g. by a 3rd order curve through adjacent points.

By modifying the input parameters, it is possible to have the index contours plotted thicker than the others.

Once the highest contour in the map sheet has been processed, the spot heights are edited. They are checked for their position relative to one another and to the edge of the map sheet, so as to avoid superimposition. If there are two spot heights close to each other, one height is printed to the left of the point, and the other height to the right.

If the contour plot is intended to supplement an orthophoto, the control points are edited and recorded on the magnetic tape. The final step is the processing and output of the map frame, containing the following elements:

- computation of the coordinate grid in the sheet
- grid marks at edge of sheet, with printed coordinates, and grid crosses.
- sheet border
- title, scale and sheet number.

2.8 OUTPUT of Program System

The end-product of the computations is a magnetic tape edited for the automatic plotting table and containing the entire plot for the map sheet. At present, this output is adapted for the Coragraph DC2 automatic plotting table made by Contraves AG, Zürich.

3. Exemples

Map-sheet No.	121/122[1]	2107/2106	6631667		4547	
Diapositive scale	1:7500	1:32 000	1:30 000		1:80 000	
Model scale	1:3750	1:13 333	1:15 000		1:40 000	
Plot scale	1:2500	1:10 000	1:10 000		1:24 000	
Contour interval	2 m	10 m	5 m		200 ft	
Registration:						
Scan direction	X	Y	Y		Y	
Scan width	15 m	50 m	50 m		120 m	
Registration interval	Constant time 1 sec (≙ 15 m)	Constant time 1 sec (≙ 53 m)	Constant time 1 sec (≙ 75 m)		Constant time (1 sec ≙ 160 m	0.8 sec ≙ 96 m)
No. of reference points	6 600	6 400	3800		6500	10 000
No. of models in map sheet	1	1	1		2	
Grid interval	5 m	20 m	20 m		120 ft = 36.6 m	
No. of interpolated grid points	44 500	39 040	43 300		142 300	
Size of nett CU	50 x 50 m^2	200 x 200 m^2	280 x 300 m^2	500 x 500 m^2	1680 x 2400 ft^2	
Overlap of gross CU	30 m	75 m	60/105 m	105 m	552 ft	
No. of CU in map sheet } Part 3	16x27 = 432	14x27 = 378	10x20 = 200	7x15 = 105	25 x 20 = 500	
Average no. of reference points per gross CU	ca. 70	ca. 50	ca. 56	ca. 101	ca. 70	
Computing time (CDC 6600)						
Part 1	--	--	94	88	224	
Part 2	69 SS (System seconds)	72 SS	22	15	99	
Part 3	401	230	141	229	552	
Part 4	103	71	111	118	144	
Part 5	67	48	51	57	72	
Total:	640 SS	421	419	507	1091	

[1] Appendix 1 shows an excerpt from this model

In model 121/122, the overlap of the gross CU's - two profile widths - is very large. This leads to a high average number of reference points in the gross CU and a corresponding sharp rise in the computing time for Part 3. If this overlap is reduced and the time needed for the extensive print-out required in this case is subtracted, the computing time would be about 400 system seconds and would correspond approximately to the time for one sheet of the German Standard Map (1:5000), with about 6000 reference points.

4. Program Features

4.1 Freedom in data acquisition

According to the quality required in the end product, the operator can select the profile width freely and choose either constant intervals of time or constant intervals of distance for the registrations. In addition, he can measure as many geomorphological structure-lines and spot heights as he feels may be necessary to ensure the desired quality.

The position of the model is independent of the map sheet, but the map sheet itself must be completely covered by models.

4.2 Adaptation of the computing and plotting effort to the extent and choice of data

Since not only the quality, but also the amount of computation, rises with the increase in reference-point density, the economic factor must be weighed against the required quality of the end product in each individual case. The following features may improve the economy:

- registration of the model coordinates in the national system
- direct registration of the terrain points in a rectangular grid
- increasing the grid interval

In addition to reducing the computing time - particularly the case with the second suggestion - the third suggestion also leads to a reduction in plotting time.

1) ... 1 shows an excerpt from this mode

4.3 Further processing

As a by-product, the program system provides the heights of a regular, close-meshed grid, which can be transferred to magnetic tape and used, among other things, for volume computations and highway profiles.

The contours are also available in digital form, and can easily be combined - in both computation and plotting - with a digitised plot of ground details.

4.4 Continuous adaptation to terrain

Of great influence on the quality of the contour plot are the various ways of adapting the program system to the type of terrain involved:

- the coding of the type of terrain by the operator fixes the grid interval in the map sheet and hence determines the quality of the approximation of the grid to the surface of the terrain
- the covariance function is determined separately for each CU
- the grid interval is automatically halved in any CU in which no plausible contours can be found

It should be emphasised that, in this way, it is only at geomorphologically difficult points that the amount of computing is increased to ensure uniform quality over the entire map sheet.

5. Computer

5.1 Program language

The program is written in standard FORTRAN IV. It is, however, possible to incorporate Assembler sub-routines, which are generally available, so as to save as much computing time as possible.

5.2 Configuration

Core storage about 56 K words of 60 bits or
about 256 K bytes of 8 bits (in addition to operating system)

2 magnetic-tape units

1 disc or drum memory

The time for floating-point multiplication should be less than 10 μsec.

Examples: IBM 360/65 upwards

IBM 370/155 "

CDC 6400 "

6. Accuracy of Contours

6.1 Geometric accuracy

Only initial results will be given, as the studies have not yet been completed.

6.1.1 Model 121/122

Conventional plots of this model, produced by two operators, were available.

For a topographic map at scale 1:2500, Imhof gives the mean square error of the contours as

$$m_h = \pm\ (0.25 + 1.5 \cdot \tan\alpha)\ [m] \qquad (2)$$

and Finsterwalder-Hofmann quotes the error in a photogrammetric contour plot as

$$m_h = \pm\ (0.2 + 0.4 \cdot \tan\alpha)\ [m] \qquad (3)$$

where α is the slope.

To determine the relative accuracy of a conventional plot, the planimetric displacements of identical points were found by superimposing the two conventional plots, and the mean square error of the contours was found to be

$$m_h = \pm\ (0.18 + 0.7 \cdot \tan\alpha)\ [m] \qquad (4)$$

which is of the same order as the accuracy given in (3).

The automatic plot was superimposed on one of the conventional plots, and the mean square difference between conventional and automatic plots was found to be

$$m_h = \pm (0.23 + 1.4 \cdot \tan\alpha) \quad [m] \qquad (5)$$

On the basis of (4) and (5), the mean square error of the contours in an automatic plot is found to be

$$m_h = \pm (0.14 + 1.2 \cdot \tan\alpha) \quad [m] \qquad (6)$$

Comparison of (4) and (6) shows that the automatically plotted contours are slightly more accurate in flat terrain, and slightly less accurate in steep terrain, than the contours plotted in the conventional way.

The error limit for the mean square contour error in the German Standard Map 1:5000 is

$$m_h = \pm (0.4 + 5 \cdot \tan\alpha) \quad [m] \qquad (7)$$

6.1.2 Model 2107/2106

For the scale of 1:10 000 involved in this case, Imhof quotes a mean square contour error of

$$m_n = \pm (0.6 + 1 \cdot \tan\alpha) \quad [m] \qquad (8)$$

while Finsterwalder-Hofmann gives

$$m_h = \pm (0.4 + 2.0 \cdot \tan\alpha) \quad [m] \qquad (9)$$

Since a conventional plot was also available in this case, it was possible to make a direct comparison. From the differences between the conventional and automatic plots, the mean square difference was found to be

$$m_h = \pm (1.1 + 1.6 \cdot \tan\alpha) \quad [m] \qquad (10)$$

If the mean square error given in (8) or (9) for a conventional plot is subtracted from the error given in (10), the mean square error for the automatically plotted contours is found to be

$$m_h = \pm (1.0 + 0.6 \cdot \tan\alpha) \quad [m] \qquad (11)$$

6.2 Assessment of cartographic quality

Of decisive importance for the quality of the reproduced terrain forms is the density of the reference points.

In extremely flat terrain, conventionally plotted contours sometimes oscillate about the automatically plotted contours. This is due to the fact that the planimetric uncertainty in conventional plotting increases as the slope of the terrain decreases. Under such conditions, the automatic plot is more accurate, since the profiles can be measured fairly accurately in flat terrain.

In steeper terrain, there are some planimetric displacements, between the conventional and automatic plots, of characteristic terrain forms such as mountain ridges, gullies and cuttings. In such cases, a good operator will always supplement his conventional plot with a characteristic interpretation, e.g. by marking an erosion trough with hairpin contours, by making a sharp turn at embankments or by crossing roads more or less perpendicularly. The automatic plot, on the other hand, depends particularly on structure lines with additional points being recorded in sufficiently close succession not only in the case of break-lines (e.g. the edges of embankments), but also in the case of all other characteristic terrain forms such as mountain ridges. This additional information requires interpretation on the part of the operator during the data acquisition.

The cartographic benefits of automatic plotting are particularly evident on steep mountainsides, where the excellent plastic representation gives a much better impression of height than the conventional plot with its differential uncertainty in the course of the lines.

Whereas a conventional plot is usually subjected to cartographic editing to smooth out these uncertainties, the scribed automatic plot is generally the end-product.

To sum up, it can be said that the recording density used in these examples -- 3 to 4 mm profile interval in the model and an average recording interval of 3 to 4 mm in the profile - is sufficient to attain the accuracy of a conventional plot. Apart from this, the contour plot can be used as the end-product for most purposes, e.g. for engineering or planning, without any further cartographic revision

In a few cases which require still higher cartographic quality, this can be achieved by higher point density, a smaller grid interval or by introducing small corrections to the data on the magnetic tape for the automatic plotting table - e.g. recording modifications to the contours on the magnetic tape by means of a digitising table or a light pen on a computer-fed storage tube.

7. Program Extensions

7.1 Program version for plotting tacheometric surveys

In this case, the initial data may come, for instance, from a recording tacheometer. The reference points are generally distributed irregularly over the terrain, so that the point density can vary greatly. Because of this variation in point density, a straight line has mostly been used as a covariance function.

An important difference with respect to the normal version is that the map sheet borders are irregular. Within the map sheet, there may also be areas without reference points.

Break-lines are of supreme importance in tacheometric survey. For this reason, the program caters for densification of the reference points along the marked break-lines. In addition, covariances between reference points on different sides of a break-line are set at zero in the interpolation equation (1).

7.2 Program version for slope maps

For land valuation and allocation of new plots of land in areas with broken terrain, the Land Redistribution Office needs maps in which areas of different ground slope are delimited. In Baden-Württemberg, for instance, land with a slope of 0 to 6 per cent is regarded as level, so that no reduction is applied to the valuation, whereas land of more than 6 per cent slope may be given a lower valuation.

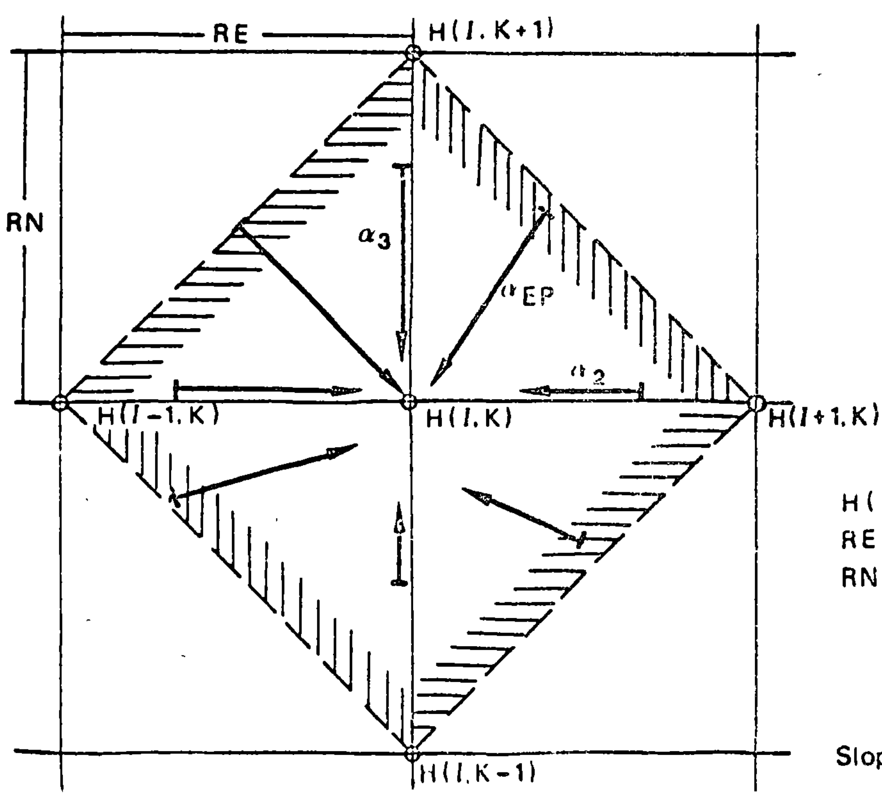
If, for instance, the 0 to 6 per cent slope is to be delimited from the following 6 to 10 per cent slope, the problem is to locate a large number of closely-spaced adjacent points with a slope of 6 per cent. Here too, it is lines which are required, so that the problem corresponds to the contour interpolation described in Section 2, except that the digital terrain model has to be replaced by a grid in which the terrain slope is stored instead of the terrain heights. In the same way as lines of constant height, lines of constant slope are then interpolated in this grid of slope values.

The slope at a grid point is computed from the height information in the digital terrain model. To do this, the angle of inclination α_E of a plane must first be determined. In our particular case, involving two grid directions perpendicular to each other and with inclinations α_2 and α_3, the equations (23) and (22) in [3] (page 163) give us the following:

$$\tan\alpha_E = \sqrt{\tan^2\alpha_2 + \tan^2\alpha_3} \qquad (12)$$

With the symbols in Fig. 9, and with slopes expressed as percentages, we have:

$$\alpha_{EP} = \sqrt{\left(\frac{(H(I+1,K)-H(I,K)) \cdot 100}{RE}\right)^2 + \left(\frac{(H(I,K+1)-H(I,K)) \cdot 100}{RN}\right)^2} \qquad (13)$$



H(I, K) Grid height at point (I, K)
RE Grid interval, easting
RN Grid interval, northing

Fig. 9

Slope of a plane

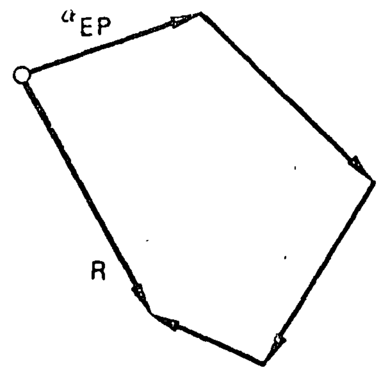


Fig. 10

Vectorial addition of
the slopes of 4 planes

The slope at grid point (I,K) is computed as the mean of the slopes of the 4 planes shaded in Fig. 9, the known percentage grades of these 4 planes being added vectorially (Fig. 10). The value of the resultant vector R is divided by 4 to obtain the slope G (I,K). This is the slope of the tangential plane to the digital terrain model at grid point (I,K).

$$G(I,K) = \sqrt{\frac{((H(I+1,K)-H(I-1,K))(100/RE))^2+((H(I,K+1)-H(I,K-1))(100/RN))^2}{4}} \qquad (14)$$

The desired limits of slope are read-in on punched cards.

In the case of model 121/122[2)] used as an example, the computation of the slopes from the digital terrain model on the CDC 6600 required 78 SS extra time in Part 4, and 56 SS in Part 5 - a total additional computing time of 134 SS, although here again extensive print-outs are included.

No accuracy studies have yet been made, but the slope map and the contour plan were superimposed and compared. Particularly striking is the variety of detail in the slope map. These details are contained in the digital terrain model but no longer evident from the contours. Apart from ascertaining that the slopes appear plausible, it can already be said that the slopes derived directly from the digital terrain model will be generally superior, in both accuracy and completeness, to those derived from the contours.

---

2) Appendix 2 shows an excerpt from the slope map corresponding to the contour plan in Appendix 1. The digital terrain model used in obtaining the slope map is slightly more strongly filtered than that used for the contours.

8. Further developments

Now that work on the basic version of the contour program, as described in Section 2, has been completed, a start will be made on the following versions:

8.1 Generalisation program

Selective filtering of the height data in the reference points will make it possible to compute a generalised terrain model in which generalised contours can subsequently be interpolated. First, encouraging results are shown in the paper "A general digital terrain model" by Kraus.

8.2 Program version for geomorphological editing

In this case, the influence of terrain break-lines on the digital terrain model, as described in Section 7.1, is taken into account to ensure optimum reproduction of geomorphological details in the contour plot.

8.3 Program version for slopes

This version must be tested still further and accuracy studies made.

8.4 Program version for profiles from the digital terrain model

First results have been obtained. Within certain limits, profiles can be computed from digitised contour lines by the contour program described in Section 2. The variation of density of the reference points within the map sheet must not exceed the ratio 1:5. Further work is necessary to raise this limit.

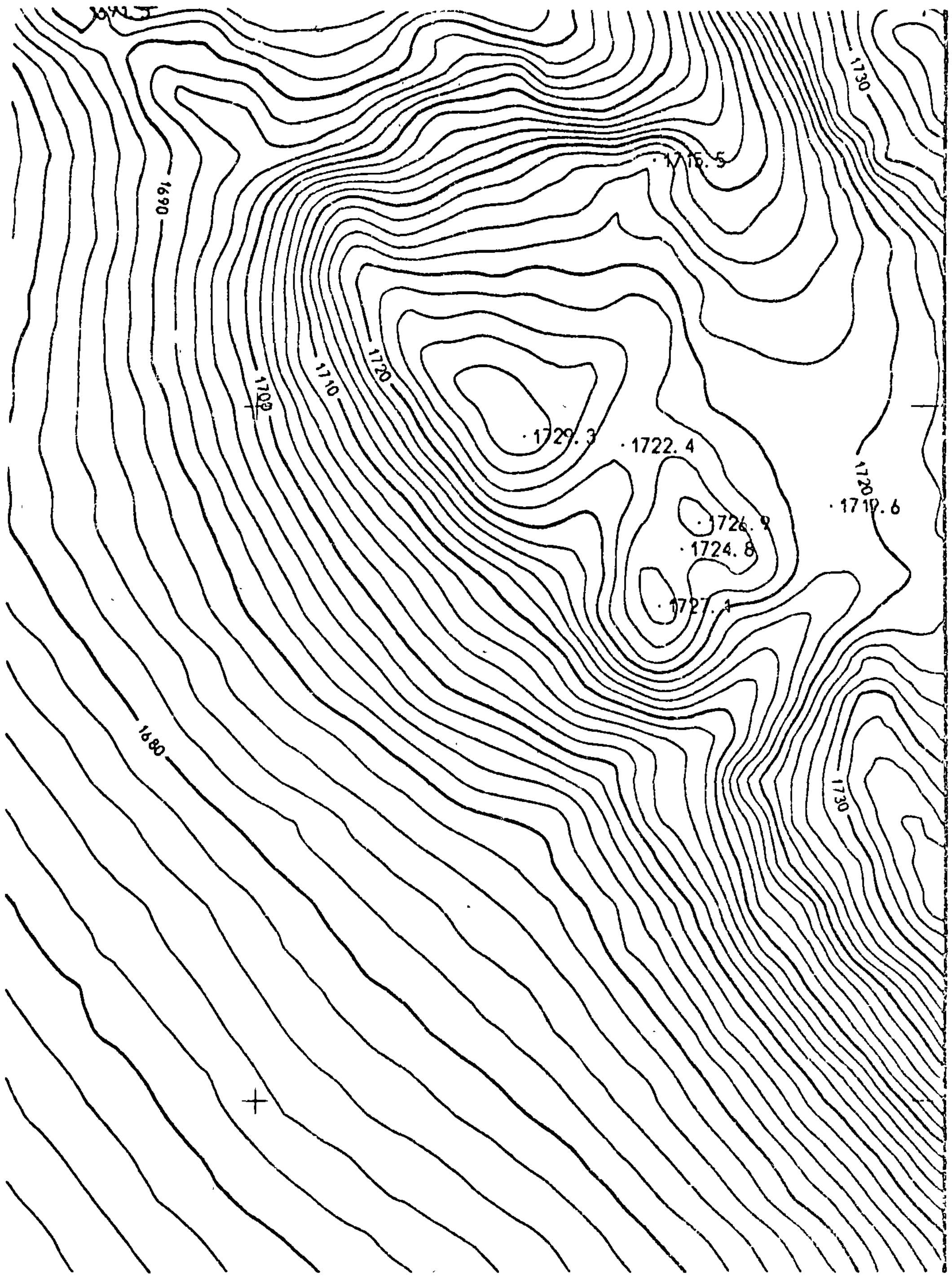
Possible further developments are cut-and-fill maps for earth movement and volume computation, and derivation of fall-lines from the digital terrain model.

Bibliography

1 Finsterwalder, R.: Zur Gewinnung von Profilen aus Schichtlinien zum Zwecke der Differentialentzerrung, ZfV 5, p. 193, 1972

2 Kraus, K.: Prädiktion und Filterung mit zwei verschiedenen Stützpunkt-Gruppen, ZfV 4, p. 146, 1973.

3 Kraus, K.: Zur Auswertung nicht orientierter Luftbilder, BuL 4, p. 163, 1971.



Appendix 1

Automatic contour plot
Scale 1:2500

Slope intervals 0 6 10 14 18 22 per cent

Appendix 2

Automatically - produced slope map

Scientific Applications | C.L. Lawson Editor

Representation of Contours and Regions for Efficient Computer Search

R.D. Merrill
Lockheed Palo Alto Research Laboratory

A novel computer-searchable representation for the three basic pictorial features, contour maps, region coverage, and line structures, is described. The representation, which has practical storage requirements, provides a rapid means of searching large files for data associated with geometric position as well as with attribute value. An application of this representation to handling terrain information illustrates its utility. The algebraic properties of the data structure make it computationally easy to determine whether a point lies within a closed boundary; compute the area contained by a closed boundary; generate the closed boundary representing the union or intersection of two closed boundaries; and determine the neighboring boundaries to a point and the minimum distances between them and the point.

Key Words and Phrases: contour map representation, region boundary representation, computer-searchable structure, graphic data retrieval, graphic language, two-dimensional patterns, computer graphics, graphic display

CR Categories: 3.21, 3.23, 3.79, 6.35

This work was sponsored by the Lockheed Independent Research Program. Author's address: Lockheed Palo Alto Research Laboratory, Palo Alto, CA 94304.

1. Introduction

There are numerous scientific applications for which it would be desirable to represent certain two-dimensional data in computer-searchable form. An important class of such data can be characterized either as piecewise-continuous single-valued functions, or step functions of two independent continuous variables. These are commonly treated either in the form of contour maps, or sets of labeled region boundaries.

Applications using these different kinds of data are numerous. For example, in modeling terrain for automatic interpretation of remote sensor signals [1, 2], or radar simulation [3, 4], it is desirable to portray ground cover, topological features, elevations, and slopes. The data necessary to represent this information can be characterized in the form of sets of contour lines and region boundaries. In implementing the terrain model, this data must be structured in storage so that the computer can search it efficiently for surface properties at any point in the two-dimensional geographic space being modeled. The representations of interest must depict these properties with resolutions on the order of tens of feet over space covering hundreds, possibly thousands, of square miles. Thus it is important to have an efficient means of handling this spatial data.

Numerous techniques for handling this and other types of multivariate data have been reported [5, 6]. Notable results by Freeman and Morse [7, 8] have provided a means of representing contour maps in computer-searchable form. Their scheme uses the grid-intersect method of chain-encoding line drawings [9]. A graph form is used to represent the relationships between line segments making up a contour map.

Pfaltz's geographic map analysis system [10] represents surface properties by binary arrays or "PAX planes" [11] rather than locus structures. Region and area data are represented in a similar manner in the Automatic Cartographic System, which is a large information handling system for the manipulation, retrieval, and display of topographic data [12].

The data representation presented in this paper was first disclosed by the author in a paper on the correlation of remote sensor signals with ground truth information [13]. Since then it has also been applied to the representation of regional features in an interactive system for analysis of remote sensor data [14] and the manipulation of digitized biomedical pictures [15].

A description of the data structure and its properties, and the techniques of using the structure to represent region boundaries and contour maps in computer-searchable form is given. An operational system to handle terrain information, which incorporates these techniques, is described along with its application to the problems of representing surface coverage and elevation contours which are derived from aerial photographs topographic maps for the purpose of predicting microwave radiation from extended regions.

2. Data Representation

This section presents a novel technique for structuring closed lines and shows how they can be organized to unambiguously represent region boundaries and contour maps. The algebraic properties of these representations are discussed, and methods are given for computing area, point-boundary distances, and boundary intersections and unions, as well as strategies for searching the region boundary map and contour map representations.

2.1. Basic Data Structure

The basic data structure developed to represent a closed line is illustrated with a simple example. Consider the boundary of a region shown in Figure 1 defined by a point locus in a rectangular or quadruled grid.[1] This locus can be structured so that it is very simple to determine if a point (x,y) is in the region. The technique uses a discrete data form of the well-known topological property of a closed boundary; that is, if a test line is drawn from a point outside the region boundary to (x,y), then (x,y) is in the region if and only if the boundary is crossed an odd number of times by the line.[2] This property holds for finite point loci representations of boundaries defining arbitrary regions provided three restrictions are met.

One restriction is that the boundary must be continuous and closed, in the sense that the absolute distance between successive points in the locus cannot be greater than the grid element diagonal. If in scanning the boundary locus it is found that this restriction does not hold, the "missing" coordinate pairs are linearly interpolated and inserted in the locus. This was done between the E and S points of the example.

The second restriction is that provisions must be made for cases where the test line intersects the boundary tangentially. This is done by constraining the test line passing through (x,y) to be horizontal and augmenting every local maximum and minimum in the boundary (e.g. ① and ② in Figure 1) so that a tangential horizontal line always passes through an even number of boundary points at each extremum. This is accomplished during the boundary scanning process by repeating one of the boundary extremum points if an even number of points does not exist. Similarly, where this line passes through an inflection on the boundary (③ and ④ in Figure 1) and does not pass through an odd number of points, it is necessary to repeat one of the inflection points. This computational procedure is illustrated in Table I for the first few points of the Figure 1 boundary where added points are underlined.

Table I. Lists of Point Coordinates for Figure 1

Original Boundary

(4, 26), (5, 26), (6, 25), (5, 24), (4, 23), (4, 22), (4, 21), (4, 20), (5, 21), (6, 21), (7, 21), (8, 21), (9, 21), (10, 22), (11, 23), (12, 23), (13, 24), (14, 24), (15, 24), (16, 23), (17, 24), (18, 23), ...

Augmented Boundary

(4, 26), (5, 26), (6, 25), (5, 24), (4, 23), (4, 22), (4, 21), (4, 20), (*4, 20*), (5, 21), (6, 21), (7, 21), (8, 21), (9, 21), (10, 22), (11, 23), (12, 23), (*12, 23*), (13, 24), (14, 24), (15, 24), (*15, 24*), (16, 23), (17, 24), (*17, 24*), (18, 23), ...

A third restriction based on topological properties is that the closed boundary cannot loop back on itself. There are certain conditions in which an overlaid loop leads to an ambiguous representation.

A boundary which has been closed so that contiguous points in the locus are connected in the grid and which has been augmented at extrema and inflections, in the manner described above, is called a *tightly closed boundary* (TCB). This data structure can define an arbitrarily-shaped simple or compound region with the resolution determined by the resolution of the quadruled grid used.

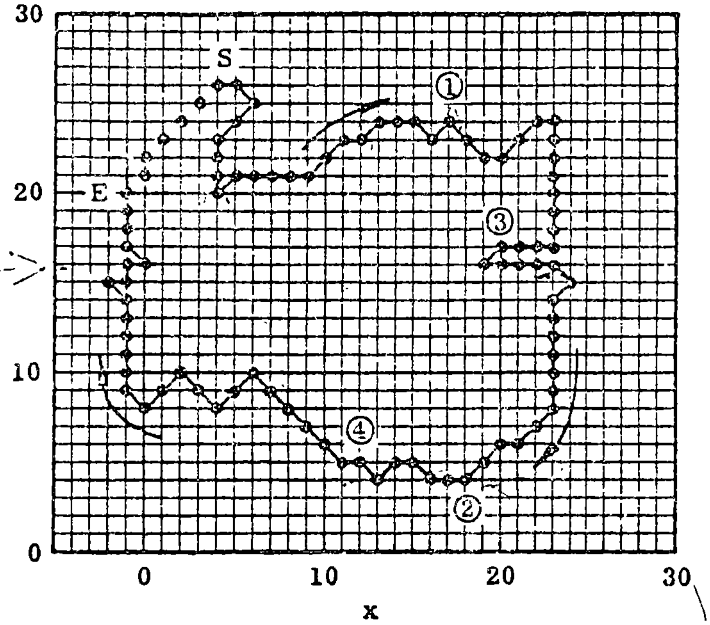
To facilitate the search of a TCB given (x,y), its coordinates are sorted and partitioned into sets so that each set contains only points which have the same y-coordinate. The associated x-coordinates of each set are ordered monotonically increasing. Each of these

[1] This discussion uses a rectangular grid for display purposes: however, the theory holds equally well for the hexagonal grid representation.

[2] This is a polygonal version of the Jordan-Brouwer separation theorem [16].

[3] Proofs for this and following theorems are contained in [19].

Fig. 1. A boundary example



ordered sets is called a *y-partition* of the TCB. Symbolically, the data structure can be expressed in the following way. The region R_k has a TCB, P_k, defined as follows:

$$P_k = \{y^k_{min}, y^k_{max}, Y^k_1, Y^k_2, \ldots, Y^k_{n_k}\} \qquad (1)$$

where y^k_{min} is the smallest y-coordinate in the boundary, y^k_{max} is the largest, and the number of y-partitions $n_k = y^k_{max} - y^k_{min} + 1$. The y-partition of P_k for y_i is:

$$Y_i^k = (r_{ik}, x_1^{ik}, x_2^{ik}, \ldots, x_{r_{ik}}^{ik}) \qquad (2)$$

where r_{ik} is the number of y_i-coordinates in the TCB of R_k and is always an even number. The x_j^{ik} are the points on the TCB having y_i-coordinates. TCB processing usually involves computations made on the pairs of these xs with "odd-even" subscripts. Using the eq. (2) form, the y-partitions for $y_1 = 4$ and $y_{13} = 16$ in the Figure 1 boundary are:

$$Y_1 = (6, 13, 13, 16, 16, 17, 18),$$
$$Y_{13} = (8, -1, 0, 0, 19, 20, 21, 22, 23).$$

Also to facilitate the searching process, the extreme coordinate values x_{min}, x_{max} of the TCB are determined during the y-partitioning. These, together with y_{min}, y_{max} define a rectangle bounding the TCB called the *geometric index*.

There is an alternate structure which preserves the integrity of the TCB properties and has certain advantages over the eq. (2) form. A list $\{d_{ik}\}$, $1 \leq i \leq n_k + 1$ is placed at the beginning of the array containing the TCB. The displacement of the first x-coordinate in the ith partition is d_{ik}, which can be used to quickly find the ith partition without having to scan through the entire structure, as must be done in the eq. (2) structure. Moreover, it eliminates the need for the r_{ik} since the number of x-coordinates in the ith partition can be quickly calculated as $r_{ik} = d_{i+1,k} - d_{ik}$ because there are $n_k + 1$ displacements.

Another approach to representing region coverage approximates a boundary with a polygon [17]. Each polygon side is expressed in terms of its end points and straight-line equation. To determine if a point is in the region, this structure is searched by counting odd-even side crossings of a line connecting the point to a position known to be outside the boundary. Possible side crossings are determined by a set of interval tests much like those used in searching the TCB y-partition structure. However, the search can be considerably longer than that for TCBs since each possible side crossing requires three subtractions, a multiplication, and a division. Another disadvantage is that representing an irregular-shaped boundary can require a polygon with a very large number of sides. An earlier approach [18] used a similar technique to represent boundaries; however, inclusion was determined by partitioning the enclosure into rectangular regions.

2.2. Properties of the Data Structure

Given that a region A is defined by a TCB structured into y-partitions, it is possible to very simply and rapidly answer questions about the two-dimensional properties of A. Questions of particular interest are:

- Is point $p = (x,y)$ within or on the boundary of A; $p \in A$?
- What is the area of A; $a(A)$?
- What is the minimum distance between p and the boundary of A; $\min\{\delta(p, A)\}$?
- What is the boundary of the area common to areas A and B; $A \cap B$?
- What is the boundary of the area which includes either area A or B or both areas A and B; $A \cup B$?

Determining Containment. Because of the structural simplicity of the TCB, it is possible to determine very rapidly whether a point on the grid is in the region defined by the TCB. The result of this search is always unambiguous, as shown in the following theorem.

THEOREM 1. *A point $(\bar{x},\bar{y})$ is within or on the boundary of region R_k defined by P_k if and only if there exists a Y_i^k, $i = 1, 2, \ldots, n_k$, with an odd-even pair x_j^{ik}, x_{j+1}^{ik} such that $y_i = \bar{y}$ and $x_j^{ik} \leq \bar{x} \leq x_{j+1}^{ik}$.*[3]

The search strategy to determine whether a point (x,y) is on or within a closed boundary is as follows.

Step 1.
Test the geometric index; if $(\bar{x},\bar{y})$ is outside this bounding rectangle, it is also outside the TCB.

Step 2.
Retrieve the $\bar{y}$-partition for the TCB if $(\bar{x},\bar{y})$ is on or within the bounding rectangle.

Step 3.

Test the odd-even pairs of xs in the $\bar{y}$-partition; if there exists a pair (x_0, x_e) such the $x_0 \leq \bar{x} \leq x_e$, then $(\bar{x},\bar{y})$ is on or within the TCB.

The program for this algorithm makes at most four comparisons in step 1; one mass-storage access in step 2 if the $\bar{y}$-partition is not already in main memory; and at most r_{ik} comparisons in step 3. *No arithmetic operations* are needed, aside from address-indexing functions. Thus it is simple and can be executed very rapidly.

The y-partitioned TCB representation can also apply to $\bar{A}_k$, the set of all points in the digitizing mesh outside the boundary of A_k. The following corollary to Theorem 1 formally states this concept of an *area inverse.*

COROLLARY 1. *A point $(\bar{x},\bar{y})$ is in $\bar{A}_k$ if and only if there exists no Y_i^k, $i = 1, 2, \ldots n_k$, of A_k with odd-even pairs x_j^{ik}, x_{j+1}^{ik}, $j = 1, 3, \ldots, r_{ik} - 1$ such that $\bar{y} = y_i$ and $x_j^{ik} \leq \bar{x} \leq x_{j+1}^{ik}$.*

Computing Area. The area defined by a sampled boundary is interpreted on the basis of the number of grid points on and within the TCB representation. This assumes each point represents a unit area which is a square centered on the point.[4] The area of A can be computed directly from the y-partitions as follows:

THEOREM 2.

$$a(A) = \sum_{i=1}^{n} (L_i + r_i/2)$$

where

$$L_i = \sum_{j=1}^{r_i/2} (x_{2j}^i - x_{2j-1}^{i'})$$

$$x_{2j-1}^{i'} = x_{2j-1}^i + 1 \quad \textit{if } x_{2j-1}^i = x_{2j-2}^i$$
$$= x_{2j-1}^i \quad \textit{otherwise}$$

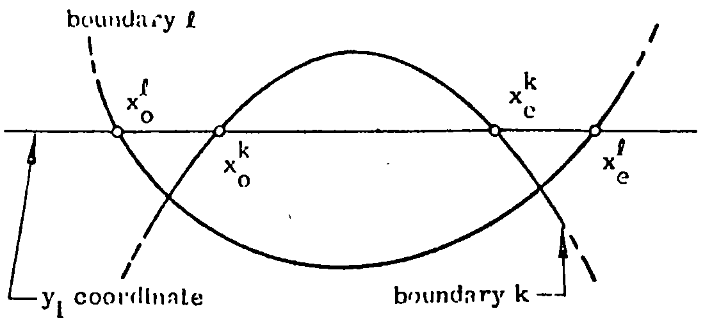
n is the number of y-coordinates in the TCB for A, and r_i is the number of x-coordinates associated with the y_i-coordinate.

The L_i computes the sum of odd-even x interval lengths in the y-partition, and the $r_i/2$ term accounts for the end effects of each odd-even interval. The $x_{2j-1}^{i'}$ adjustment is required because the TCB form for an inflection which originally had an even number of x-coordinates with the same y_i-coordinate contains a repeated value for either the first or last x-coordinate. In instances where these repeated values occur as an even-odd sequence in the y_i-partition, it is necessary to augment the odd member so that the repeated coordinate is not counted twice. It will be noticed that where they occur as an odd-even sequence, the point is automatically counted only once, and therefore the sequence test is required to distinguish the two cases.

Minimum Distance from Boundary to a Point. The minimum distance between a point and boundary can be computed several ways. One method which is rapidly executed and is valid for all boundary configurations first finds the y-coordinate of the area boundary closest to the point, then finds the x value in the associated y-partition closest to the point. Finding this x boundary value within a y-partition is accomplished with a binary search. All subsequent searches of y-partitions are limited to a y-value neighborhood about the point which is less than the last computed minimum Euclidian distance between the y-partitions and the point. The search is complete where the distance is less than or equal to the neighborhood radius in which all y-partitions have been searched.

Region Intersections. The area common to two boundaries A_k and A_l is enclosed by the boundary $A_k \cap A_l$. Thus $p = (x,y)$ is in both A_k and A_l iff p is in $A_k \cap A_l$. By Theorem 1, A_k contains p if there exists

Fig. 2. Intersecting boundaries and their associated "odd-even" coordinates.



a y-partition of P_k with an odd-even pair of xs which defines a range containing $\bar{x}$. Consequently, p is in both A_k and A_l if there exists such a y-partition range in both P_k and P_l. It follows that P_m, the TCB defining $A_k \cap A_l$, is the set of y-partitions determined by forming the intersection of respective y-partitions from P_k and P_l. Theorem 3 states this result concisely.

THEOREM 3. *P_m, the TCB defining $A_k \cap A_l$, is the set of y-partitions*

$$Y_i^m = Y_i^k \cap Y_i^l, \; y_{min}^m \leq y_i \leq y_{max}^m$$

where

$$y_{min}^m = \max\{y_{min}^k, y_{min}^l\}$$

and

$$y_{max}^m = \min\{y_{max}^k, y_{max}^l\}.$$

In the method for creating the intersection of two y-partitions, each odd-even pair of x-coordinates in one of the partitions are successfully examined. The interval defined by a given pair in this kth partition will

[4] In practice, a boundary will be sampled either manually with an x-y digitizer, or automatically with a film reader. In either case, the accuracy with which the boundary is represented in the grid coordinate system will nominally approach the linear dimensions of the grid point spacing.

Table II. Odd-Even Intersection Assignments

Cases	Odd-Even x Coordinate Relationships		Intersection Assignments	
	x_0^k	x_e^k	x_0^m	x_e^m
1	$\geq x_0^l$	$< x_e^l$	x_0^l	x_e^k
2	$< x_0^l$	$\geq x_e^l$	x_0^l	x_e^l
3	$< x_0^l$	$< x_0^l$	x_0^l	x_e^k
4	$\geq x_0^l$	$\geq x_e^l$	x_0^k	x_e^l
5	$= x_e^l$	don't care	x_0^k	x_0^k
6	don't care	$= x_0^l$	x_e^k	x_e^k
7	$> x_0^l$	don't care	no action	
8	don't care	$< x_0^l$	no action	

Fig. 3. Terrain map of regional features.

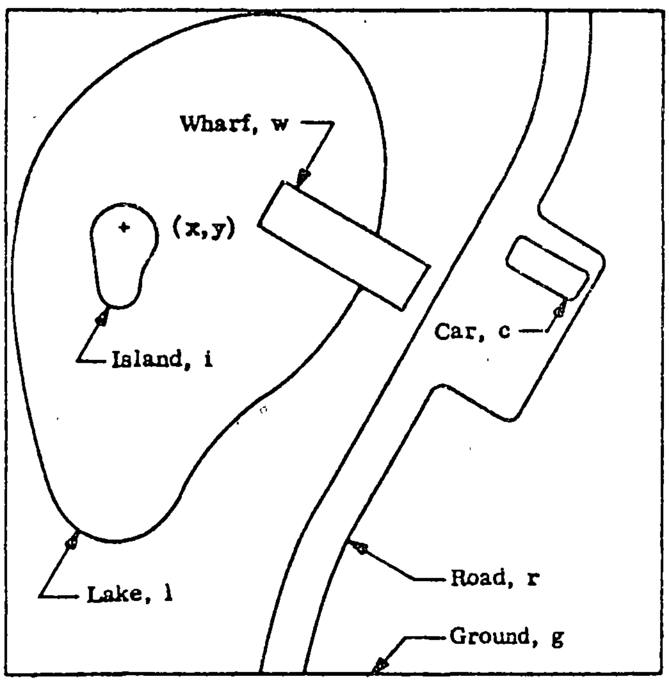


exhibit one of eight possible states with each pair in the other or lth partition. An enumeration of these relationships and the intersection assignments in each case is tabulated in Table II. The example in Figure 2 illustrates the Case 1 relationship.

Another important problem is $A_k \cap \tilde{A}_l$ given that $\tilde{A}_l$, the inverse of A_l, is the boundary containing all points in the digitizing mesh not in A_l as in Corollary 1.

COROLLARY 2. *P_m, the TCB defining $A_k \cap \tilde{A}_l$, is the set of y partitions*

$$Y_i^m = Y_i^k \cap \tilde{Y}_i^l, \quad y_{min}^k \leq y_i \leq y_{max}^k,$$

where $\tilde{Y}_i^l$ defines the points on the line $y = y_i$ which are either between even-odd x pairs of Y_i^l, less than x_1^{il}, or greater than $x_{r,l}^{il}$. Taking these three exceptions into account for the Theorem 3 combining method, completely enumerate the possible conditions that can arise.

Region Unions. The union of A_k and A_l, $A_k \cup A_l$, is the boundary containing all points in either A_k or A_l.

THEOREM 4. *P_m, the TCB defining $A_k \cup A_l$, is the set of y-partitions*

$$Y_i^m = Y_i^k \cup Y_i^l, \; y_{min}^m \leq y_i \leq y_{max}^m$$

where

$$y_{min}^m = min \{y_{min}^k, y_{min}^l\}$$

and

$$y_{max}^m = max \{y_{max}^k, y_{max}^l\}.$$

The computational procedure for creating the union of the two y-partitions treats the eight states enumerated in Table II in a manner analogous to that shown for intersections. However, in the result of forming the union of two partitions, certain redundancies can occur in the intervals covered by two adjacent odd-even pairs: one pair may be contained by the other, or both pairs may include a common interval. In the first instance, the contained pair is eliminated; in the second instance, both are integrated to form a single odd-even pair covering the combined interval.

2.3. Representing Regional Information

Many terrain properties can be represented as sets of regions, each defining the surface coverage of a particular property value. To select the region boundaries, a straightforward procedure can be used which facilitates the manually controlled process of converting the boundaries into computer form, and also provides an unambiguous presentation of the scene. In the Figure 3 example of scene material coverage, six region boundaries are selected: one each for the ground, lake, island, wharf, road, and car. Notice that the "ground" boundary (the map border) includes all the other features; similarly, the lake boundary includes the island and part of the wharf, and the road boundary includes the car. These obvious overlaps are taken into account by the order used in encoding the boundaries.

The person entering the data with an x-y digitizer traces the boundaries for this example using the following sequence; $C = \{g, l, i, w, r, c\}$. The boundaries can be represented for computer search either in TCB form, or polygon form [17]. It is easily verified that in either case the resulting set of closed boundaries unambiguously defines the scene if the boundaries are searched in an order reverse to the encoded sequence. The search strategy is simple in this reverse order: test each TCB to determine if the specified point (x,y) is contained on or within its boundary. The value of the first boundary containing (x,y) is assigned to the point. In Figure 3, the TCBs for the car, road, and wharf were searched and found not to contain (x,y). The island TCB is the first found to contain this point; therefore, it can be concluded that the point is on the island even though subsequently searched TCBs might also be found to contain (x,y).

Note that in the encoding procedure the operator began by selecting and encoding a boundary for the ' on which extended over the largest portion of the scene. This was done even though the boundary included areas of other features. He then selected and encoded the boundary of the next most expansive feature. In this way he was not required to retrace a common boundary dividing dissimilar materials. In repeating this process for successively smaller, less extensive features, the operator had only to concentrate on selecting the boundary which most expeditiously defined the feature region being considered. The operator has much latitude in selecting these boundaries and the order in which they are encoded, although the two operations are not mutually exclusive.

The filing system for sets of TCBs representing a scene has been designed to minimize the number of direct access storage (DAS) device read operations required during the search operation. Two file entries are created for each set of TCBs representing a surface property: a *search list* and a *master file* record. The first c ains one element for each TCB of the set representing an *attribute* or property. It is placed in the scene directory. The master file record includes the TCB structured in the eq. (1) form. An element of the search list contains six fields:

1. Boundary attribute value or the surface property code.
2. Geometric index $(x_{min}, x_{max}, y_{min}, y_{max})$.
3. An address pointing to the TCB unit record in the master file.
4. TCB record length.
5. Address of the search list element for the boundary to be searched next if the point being examined is not contained by the boundary just searched (called the *no-transition*).
6. Address of the search list element for the boundary which is to be searched next if the point is contained by the boundary (called the *yes-transition*). For region boundary maps, this is the address of the search list element itself.

Fig. 4 Search graph for the Figure 3 example.

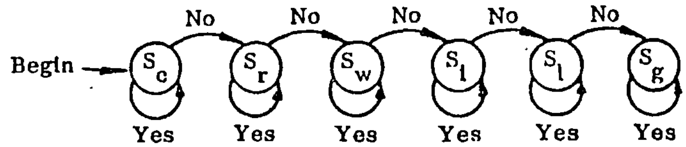


Fig. 5. Bridging in contour maps.

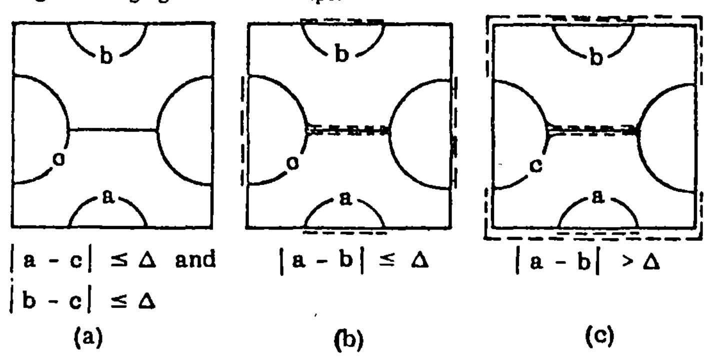
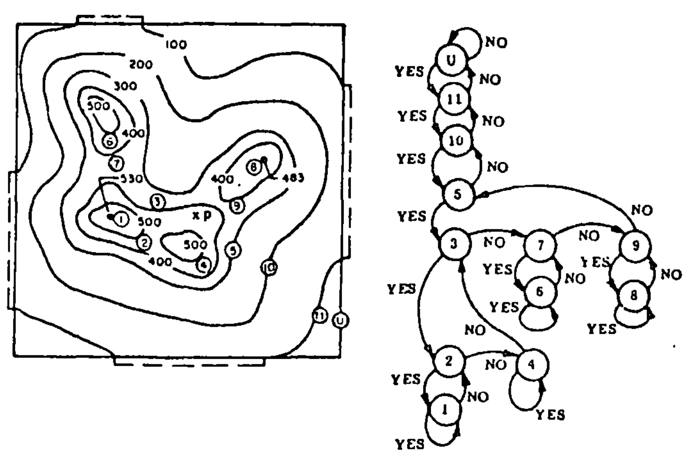


Fig. 6. Contour map with search graph. (a) Contour Map. (b) Search Graph.



The search list for Figure 3 is illustrated in Figure 4 which shows the no- and yes-transitions in directed graph form.

The algorithm which implements the search strategy is a simple one. It starts at the beginning of this list and performs a search of each successive boundary as directed by the no-transitions. The first boundary found to contain the point being considered represents the property characterizing the point.

Each search list is identified with its associated attribute, a symbolic code designating the type of surface property represented. Thus, for example, a particular terrain scene may be represented in its scene directory by four search lists: one for the TCB set representing surface material compositions; one for surface roughnesses; one for load-bearing strengths; and one for cultural features such as houses, roads, towers, towns, bridges, and power/telephone lines. Usually each search list is associated with one set of master file records; however, TCB records may be referenced by elements from more than one search list.

2.4. Representing Contour Map Information

Terrain elevations are customarily represented by contour maps as are other similar piecewise continuous single-valued functions of two variables. This subsection presents a computer-based technique for handling contour map information so that the value of the function at an arbitrary point can be efficiently derived from the contours. Others have also considered this problem [7, 8]. Their approaches were to treat the contours as line segments and define the relationships between these segments in graph form. In this development, each contour line of a map is treated as a closed curve, and the relationships between adjacent contours are explicitly defined by the search list. The closed contours are structured as TCBs for ease of search. Just as in the region boundary case, these could also be represented in polygon form [17]. There are two contour line configurations which require special consideration during the map encoding.

1. A contour line that intersects the map border must be closed. This is done by splicing its ends together with a line segment lying outside the border. Sometimes there will be more than one such contour line with the same elevation value. These often represent one contour which has been bisected by the map border and can be joined using two border line segments to produce closure. Other similar situations can arise which will require a "splice-and-try" procedure to arrive at the best configuration for encoding. In all cases, the splicing should be done outside the border where the induced elevation distortions are negligible in the vicinity of the border.

2. Two contours that are connected by a "bridge" line as shown in Figure 5(a) are separated and encoded individually. The separation procedure depends on the elevation values of the contours adjacent to the bridge. If, as in Figure 5(b), $|a - b| \leq \Delta$, the contour map elevation increment, then the bridge is treated as a continuum of one of the contours as shown dashed. The left contour was bisected and the bridge spliced on and encoded as a loop enclosing zero area. Where $|a - b| > \Delta$, the two bridged contours are bisected at the bridge junctions and the bridge is used twice as shown in Figure 5(c) to produce a legitimate encoding configuration.

The procedure for encoding a map and representing the contour line relationships is illustrated with Figure 6. First the contours are checked for closure. Line ⑪ is not closed, for example, so four artificial segments were created (shown as dashed lines) and spliced on to ⑪. Next, the encoder looked for extrema: regions on the map where the contour values are maximum or minimum. The concentric contours encircling these extrema were organized into sets: {①, ②, ③}, {④, ③, ⑤}, {⑥, ⑦, ⑤}, {⑧, ⑨, ⑤, ⑩, ⑪}. These contours were then encoded by entering their elevation values and point loci in the following order

$$C = (\varnothing, 1, 2, 3, \varnothing, 4, \underline{3}, 5, \varnothing, 6, 7, \underline{5}, \varnothing, 8, 9, \underline{5}, 10, 11, \mathrm{U}) \quad (3)$$

The symbols ∅ and U are used as delimiters: ∅, called the *null* contour, designates that the concentric set begins at an elevation extremum; and U, called the *universal* contour, designates that the last contour of a concentric set is not contained by any other contour. The underlined elements in the sequence designate that the associated contour has been previously entered, hence it is only necessary to reference that entry.

The formal procedure for deriving the encoding sequence is as follows:

Step 1.
Select an arbitrary extremum.

Step 2.
Enter ∅ and select the first contour of the extremum.

Step 3.
Trace the contour line.

Step 4.
Check within the contour that was last traced.

Step 5.
If the last traced contour contains an extremum which has not been entered, perform step 2; otherwise, perform step 6.

Step 6.
If a concentric contour does exist, perform step 3; otherwise, perform step 7.

Step 7.
Enter U and look for an extremum which has not been entered.

Step 8.
If an extremum exists which has not been entered, perform step 2; otherwise, terminate the procedure.

A contour map configured for encoding has the following property: each contour is closed, and no two contours cross or are connected (except under the pathological conditions which can arise outside the border where elevation is not defined—note that overhanging cliff conditions are not treated here). Therefore, only one of three conditions can exist between each pair of adjacent contours A and B: A contains B; B contains A; or neither contains the other and both are contained by a third contour C. Note that the encoding procedure takes care of each of these conditions as it arises so that all adjacencies are correctly represented at the completion of the encoding procedure.

The contour map search list is derived from the encoded sequence so that, given an arbitrary sample point in the plane, it will be possible to determine the contours which border the region containing the point. The list represents the contour adjacencies and containment relationships which are required by the searching process to determine the bordering conditions. For example, in Figure 6(a) contour 5 surrounds contours , 7, and 9. The search list, shown in Figure 6(b) as a directed graph, is set up so that contour 3 is searched first if the sample point is contained by contour 5; next, contour 7 is searched if 3 does not contain the point; and finally, contour 9 is searched if 7 does not contain it. (Note that the order 7, 9, 3 or 9, 7, 3 or any of the other possible combinations could also be used.) The contour map search list has elements with the same fields as the search list for region boundary maps. The yes- and no-transitions are derived for each contour in the encoding sequence $C = \{C_1, C_2, \ldots, C_m\}$ as follows.[5] A search list element is created for each contour in C which is not a reference to a previously encoded contour, or not the null $\varnothing$ contour. The search list $S = (S_1, S_2, \cdots, S_n)$ where $S_{\sigma(i)}$ is the unique search list element for C_i. If C_i is a reference to a previously entered contour, then it is said to be equivalent to that entry, and it is represented by the same list element.

The search list construction procedure is as follows:

Step 1.

Examine the first (next) pair of contours C_i, C_{i+1} to determine their search list element transitions:

(a) If $C_i = \varnothing$, then assign the yes-transition for $S_{\sigma(i+1)}$ as itself. This is written $S_{\sigma(i+1)} \rightarrow^{Yes} S_{\sigma(i+1)}$. Flag C_{i+1} as having been processed and go to step 2.

(b) If $C_{i+1} = \varnothing$, then go to step 2.

(c) If C_i, C_{i+1} have both been flagged, go to step 2; otherwise, step 1(d).

(d) Scan $C_{i+1}, C_{i+2}, \ldots, C_m$ for the set of entries which is equivalent to C_{i+1}. Let this set be $\{C_{l_1}, C_{l_2}, \ldots, C_{l_p}\}$. Form the sequence $r_i = \sigma(l_i - 1)$, $1 \leq i \leq p$. Then assign the transitions as follows:

$$S_{\sigma(i+1)} \rightarrow^{Yes} S_{r_p} \rightarrow^{No} S_{r_{p-1}} \rightarrow^{No} \cdots \rightarrow^{No} S_{r_1} \rightarrow^{No} S_{\sigma(i)} \rightarrow^{No} S_{\sigma(i+1)}.$$

Flag $C_{i+1}, C_{l_1}, C_{l_2}, \ldots, C_{l_p}$ as having been processed, and go to step 2.

Step 2.

If there is another member of C which has not been examined, go to step 1, otherwise assign the no-transition for the U search list element as itself. The search list construction procedure is complete.

To illustrate the construction procedure, consider Figure 6. In its encoded sequence C, eq. (3), note that the $\underline{3}$ and $\underline{5}$ elements are equivalent to the 3 and 5 contour labels appearing earlier in the sequence. The search list elements (whose indices are also their associated contour labels) were generated as follows:

C_i, C_{i+1}	:	Transition assignment	*Rule applied*
$\varnothing$, 1	:	$S_1 \rightarrow^{Yes} S_1$	Step (1) (a)
1, 2	:	$S_2 \rightarrow^{Yes} S_1 \rightarrow^{No} S_2$	(1) (d)
2, 3	:	$S_3 \rightarrow^{Yes} S_1 \rightarrow^{No} S_2 \rightarrow^{No} S_3$	(1) (d)
3, $\varnothing$	:	Skipped because $C_{i+1} = \varnothing$	(1) (b)
$\varnothing$, 4	:	$S_4 \rightarrow^{Yes} S_4$	(1) (a)
4, 3	:	Skipped because both were previously flagged	(1) (c)
$\underline{3}$, 5	:	$S_5 \rightarrow^{Yes} S_9 \rightarrow^{No} S_7 \rightarrow^{No} S_3 \rightarrow^{No} S_5$	(1) (d)
5, $\varnothing$	:	Skipped because $C_{i+1} = \varnothing$	(1) (b)
$\varnothing$, 6	:	$S_6 \rightarrow^{Yes} S_6$	(1) (a)
6, 7	:	$S_7 \rightarrow^{Yes} S_6 \rightarrow^{No} S_7$	(1) (d)
7, $\underline{5}$	:	Skipped because both were previously flagged	(1) (c)
$\underline{5}$, $\varnothing$	:	Skipped because $C_{i+1} = \varnothing$	(1) (b)
$\varnothing$, 8	:	$S_8 \rightarrow^{Yes} S_8$	(1) (a)
8, 9	:	$S_9 \rightarrow^{Yes} S_8 \rightarrow^{No} S_9$	(1) (d)
9, $\underline{5}$	:	Skipped because both were previously flagged	(1) (c)
$\underline{5}$, 10	:	$S_{10} \rightarrow^{Yes} S_5 \rightarrow^{No} S_{10}$	(1) (d)
10, 11	:	$S_{11} \rightarrow^{Yes} S_{10} \rightarrow^{No} S_{11}$	(1) (d)
11, U	:	$S_U \rightarrow^{Yes} S_{11} \rightarrow^{No} S_U$	(1) (d)
U	:	$S_U \rightarrow^{No} S_U$	(2)

The result of this procedure is given in Figure 6(b).

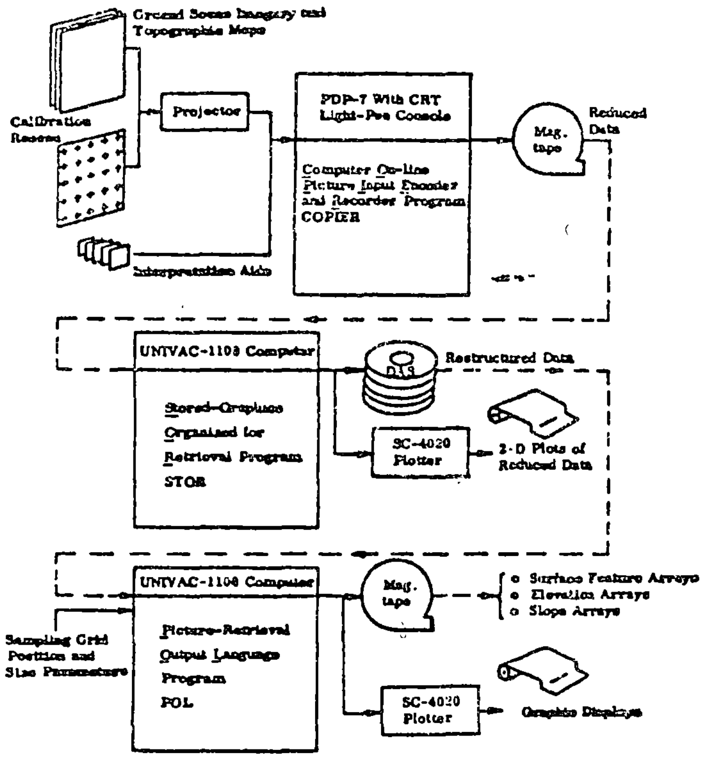
It can be shown that the union of all transitions forms a complete transition assignment for S. This list defines the relationships between contours so that the elevation value at any point in the map can be determined in a simple efficient manner. This process involves two operations.

Operation 1. Retrieve the set of contours bordering the region containing the point.

Operation 2. Measure the distance between the point and these contours. These distances, together with the

[5] In the algorithm to determine which contours border the region containing a sample point, the address of the next contour to be searched is given by the yes- or no-transition of the contour just searched, depending upon whether the point was, or was not, contained by the contour just searched.

Fig. 7. Terrain information handling system.



contour elevations, can be used in an appropriate interpolation formula to estimate elevation at the point.

The search strategy to find the bordering contours is:

Step 1.
Begin at S_U. Select the yes-transition if the point is within the map and perform step 2.

Step 2.
Search the new contour. If the point p is in the contour, select the yes-transition and repeat step 2. Also save the identify of the last contour searched which contains p. If p is not in the contour, select the no-transition and perform step 3.

Step 3.
If the new contour is not the last contour searched which contains p, perform step 2. If it is the same, then the search is complete. The set of contours bordering the region containing p is defined by the no-transitions forming the loop which threads through the contours just searched.

Every region surrounded by contours is defined by a loop in the search graph. Since this region is contained by one of these contours, there will always be a yes-transition in the loop. Thus, it can be shown that this strategy always results in finding the set of contours bordering the region containing the point. For example, p in Figure 6 is in the region bordered by contours 2, 3, and 4. Note that this region is defined by a loop in the search graph. Thus, the search strategy seeks out the correct loop by selecting the first one found with only one node having a contour containing the given point.

A simple first-order linear interpolation formula is used here to estimate the elevation value of a given point. Other more sophisticated methods could be implemented, however. For n contours bordering the region containing the point with elevation values α_1, $\alpha_2, \ldots, \alpha_n$ and minimum point-to-contour distances $\delta_1, \delta_2, \ldots, \delta_n$, then

$$\alpha_p = \left(\frac{\alpha_1}{\delta_1} + \frac{\alpha_2}{\delta_2} + \cdots + \frac{\alpha_n}{\delta_n}\right) \Big/ \left(\frac{1}{\delta_1} + \frac{1}{\delta_2} + \cdots + \frac{n}{\delta_n}\right) \quad (4)$$

2.5. Representing Line Structures

Line structures are handled as a special case of region boundary data which are not closed. The "tightly closed line" x- and y-partitions which form an orthogonal set can be utilized with the line structure search list to facilitate the integration of sets of line segments making up a coherent structure. There is an element for each line segment which contains the segment's property description, master file TCL address and length, and geometric index; and associated with each end-point coordinate pair, there is a set of search list element addresses, one for each TCL sharing that end point. With this representation, it can be shown that it is relatively simple to extract features from distorted line segments: e.g. "to the right of," "to the left of," above, below, concave and convex forms, as well as their sizes, shapes, and opening directions.

3. A Terrain Information Handling Application

An experimental system employing the TCB data representation techniques has been programed and applied to the problems of handling surface coverage and elevation data. It integrates the three key functions: converting the spatial structures into computer form; filing the searchable representations for these structures; and retrieving and analyzing data from the file storage.

3.1. The Information Handling System

The system comprises three software packages: COPIER (*C*omputer *O*n-line *P*icture *I*nput *E*ncoder and *R*ecorder), STOR (*ST*ored-graphics *O*rganized for *R*etrieval), and POL (*P*icture-retrieval *O*utput *L*anguage) as depicted in Figure 7. COPIER runs on a PDP-7 with graphics terminal. STOR and POL are programed in FORTRAN IV and versions run on both the UNIVAC 1108 and IBM 360/40 with mass direct access storage (DAS).

The COPIER data conversion facility is used to extract source data depicting features from pictures such as topographic maps, contour maps, aerial photographs, and radar images. Picture data structures found in terrain imagery are generally quite detailed, and the salient characteristics of these structures are difficult to identify and define precisely. Because automatic encoding is not within the present state of the art, it was necessary to use manual selection and encoding. The conversion process using an off-line digitizer was found to be tedious, and the resulting encodings usually had numerous errors and omissions. In an attempt to overcome these difficulties, an interactive graphics facility was adapted as a digitizer and found to provide considerable improvement. This facility is a computer-driven graphics terminal with 15 in. CRT and lightpen, which is controlled by the COPIER software. A simple projector with overhead mirror puts an image of the picture to be encoded onto the screen, which is the CRT phosphor underface.[6] Points and lines encoded with the lightpen and labels entered on the keyboard or control console by the operator are displayed on the CRT by the computer. The points and lines appear in registration with the projected image as they are encoded. This is the most important feature of the data conversion facility since the operator can immediately judge the accuracy with which he has traced the image structures and make corrections if necessary, releasing the results to bulk storage only after he is satisfied.

A calibration procedure is used so that the optical and electronic distortions can be rectified. This is accomplished by encoding the intersections of an equally-spaced precision 5 × 5 reseau projected through the optics onto the CRT at the beginning of each digitizing schedule after setting up the optical equipment, positioning and scaling the CRT digitizing grid, and allowing the electronics to stabilize.

The STOR software processes and organizes the encoded data in DAS retrieval files. First, the reseau data is used to rectify the spatial distortions in the encoded point coordinates. Tests of representative COPIER setup conditions have verified that the rectification procedure provides 1000:1 spatial resolution for this equipment. Each boundary is then converted to a TCB, sorted into y-partitions, organized by search lists to define the intended pictorial structure, and stored in DAS files as described earlier in the paper.

The POL retrieval software answers questions about the properties and relationships of features depicted in the STOR files. Procedures have been programed to compute area, minimum distances between points and boundaries, boundary intersections, and boundary unions, and to retrieve the attribute values of points from region boundary and contour map representations.

[6] There are at least three other ways of achieving a functionally equivalent interactive digitizer capability: using a CRT with a transparent port suitable for rear projection; using half-silvered mirrors to mix the projected and CRT images on a rear projection screen or microscope eye piece; and using high resolution TV cameras or film scanners which can produce a video version of the projected picture on the CRT.

Table III. Radiometer Material Code Array for Sampling Grid in Figure 8(b)

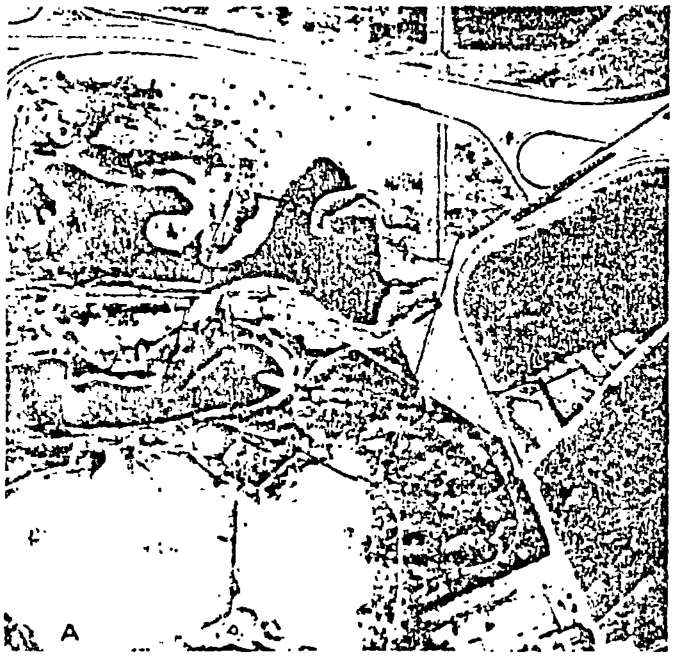
```
5555522222552222225555555555
5555522222552222222555555555
5555552222222222222255555555
5555552222222222222255555555
5555522222222222222222255555
5555522222222222222222555555
5555522222222222222222555555
5555222222222222222222555555
5552222222222222222222555555
5522222555222222222222555555
2222255555555222222225555555
2222555555555555555555777555
5555555555555555555555777755
5555555555555555555577777755
5555557777755755577777777777
5555577777777777777777777777
5555557777777777777777777777
5555552222557777777777777777
5552282222225777777777777777
5522882222222577777777777777
5228822222555577667777777777
2288222222222577777777777777
2288222222222577777777777777
2282222222225777777777777777
2282222222257777777777777777
2822222257777777777777777777
2222257777777777777777577777
2222777777677777777775555777
```

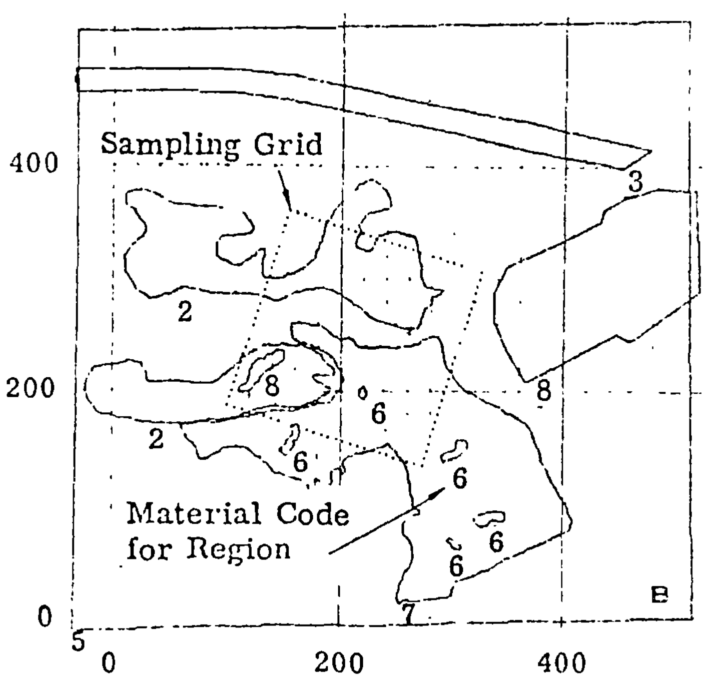
3.2. Constructing Synthetic Microwave Images

In aircraft guidance [20] and Earth resources remote sensing [21], there is an interest in the analysis of imagery obtained with scanning microwave radiometers. In support of these investigations, it has been necessary to devise a practical method for predicting the radiometric imagery which would be obtained from an actual sensor over extended terrain regions. The information system of Figure 7 was applied to this problem.

A microwave radiometer measures the emitted and reflected energy from various materials in a ground scene. These measurements are essentially independent of terrain contours. The instantaneous output signal is a measure of the energy collected by the antenna. The example in Figure 8 illustrates how the information system was used to predict the radiometer measurements. The aerial photograph in Figure 8(a) is the source from which the region boundary map of material coverage shown in Figure 8 (b) was produced using COPIER and STOR. Not all radiometric material regions were encoded for this example, so that it would be more easily understood by the reader. Figure 8(b) is composed of a general background of soil ⑤, two lakes ②, one lake island ⑧, golf course fairways and greens ⑦, sand traps ⑥, a vegetable patch ⑧, and an asphalt highway ③, encoded in that order.

Fig. 8. Aerial photograph and corresponding computer representation. (a) Aerial photo of scene. (b) Region boundary map and sampling grid.





The synthetic microwave radiometer image is constructed with POL by sampling the terrain material coverage representation at regular intervals in two dimensions to retrieve the required array of material codes. In another step not covered in this paper, the materials array is translated into a digital radiometer image, or array of microwave signals, by first converting the material codes into signal amplitudes according to the appropriate ground truth signatures, then smoothing the results with a 2-D filter representing the radiometer antenna response pattern [13].[7] The materials array of Table III is the sampled result for the 28 × 28 grid shown in Figure 8(b).

The terrain information handling system has been used to encode over 60 scenes and construct approximately 500 microwave images with sampling grids ranging in size from 12 × 12 to 120 × 120. It has been found that with COPIER, an operator can encode a typical scene possessing 60 boundaries exhibiting a normal TCB boundary length distribution having a 400-point mean and 150-point standard deviation in approximately one hour. Or equivalently, the operator can digitize approximately 110 points per minute. In tightly closing the boundaries, about four points are added for every one digitized. This ratio is primarily due to the operator's liberal use of straight segments which are defined only by end points.

On the UNIVAC 1108, the STOR software rectifies, structures, sorts, and files these data at the rate of 100 to 500 points per second for scenes having distributions such as the above scene. This rate depends on boundary length since one of the STOR processing operations sorts the coordinates. The number of storage cells required for the y-partitions of a TCB range from 100 percent to 150 percent of the number of points in the TCB. The number of bits required in each cell depends on the digitizing grid size. This application needed 10 bits.

For typical scenes, material codes were retrieved at random from TCB files by POL at the rate of 400 to 1000 sample points per second. Since the search algorithm uses the geometric index to isolate the candidate boundaries for thorough search, the retrieval rate tended to be higher for boundary shapes which approach that of the boundary rectangles. This search rate was consistently high even for sampling grids that extend over large spans of the scene because POL uses a last-in first-out paging strategy to minimize the number of times a TCB must be transferred from DAS to main memory for search. For this application, the main memory buffer was sized to hold 20 average-length TCBs.

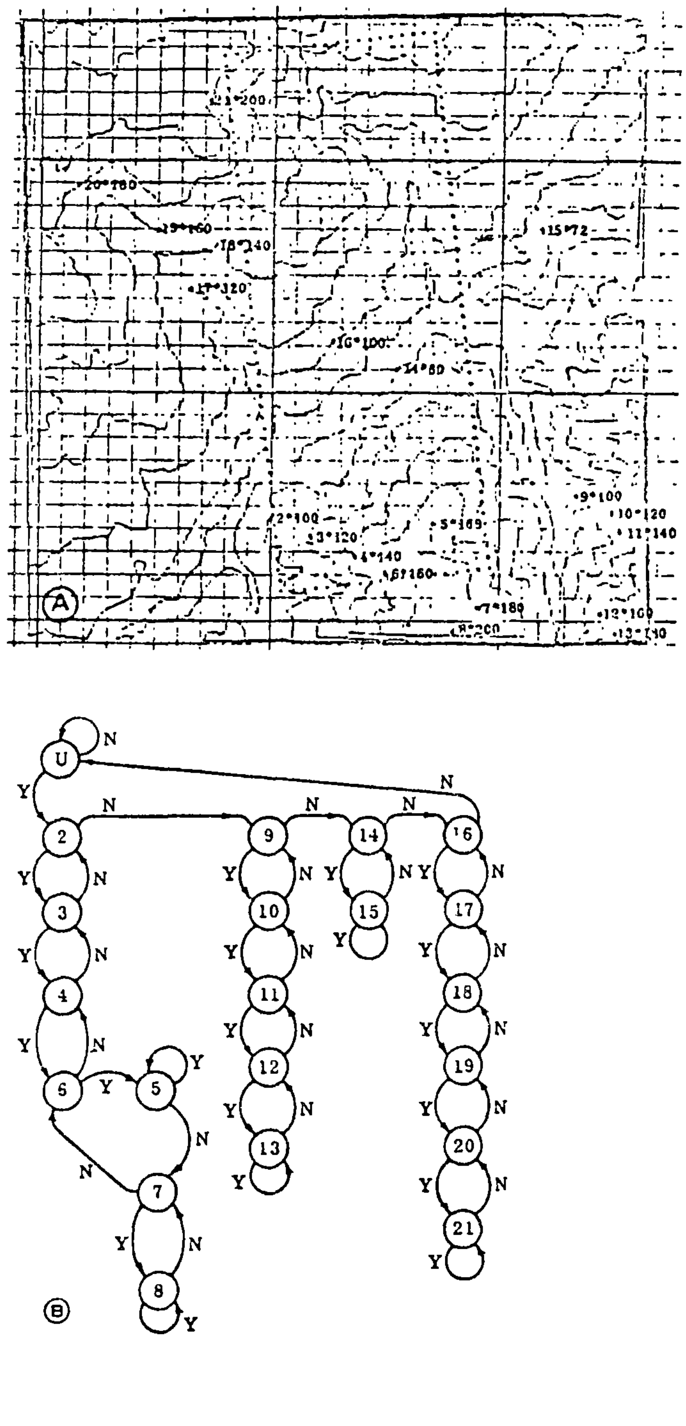
[7] A POL procedure for concatenating a sequence of overlapping aerial photos to construct a map is described in [13]. This method includes an automatic technique for handling oblique exposures, unequal exposure time intervals, aircraft drift, and aircraft velocity/height ratio changes.

POL can also be used to relate the actual radiometer signals to ground truth information visible in the ground scene photographs. Two methods have been developed for doing this: one for handling systematic linear radiometer scan patterns [13]; and one for handling random radiometer scan patterns [19].

Table IV. Array of Elevation Values for Sampling Grid in Figure 9.

200	200	200	195	186	181	170	164	169	166	162	166	166	165	165	165	164	163	163	162
200	200	194	190	188	183	172	159	158	158	158	162	162	162	163	163	162	162	163	161
196	194	187	180	184	182	173	163	154	152	155	158	158	158	159	160	159	161	162	162
190	185	179	174	177	178	171	160	149	147	150	152	151	152	154	154	155	159	161	161
187	179	174	167	164	165	165	161	140	143	146	146	145	146	147	148	149	152	158	158
189	183	178	168	158	156	156	154	142	139	142	141	139	141	141	142	143	145	146	148
193	188	181	170	160	155	144	146	141	135	135	134	134	135	134	135	135	136	138	144
196	188	183	173	165	154	139	138	136	128	125	125	128	127	127	129	128	130	137	143
192	182	177	169	159	151	143	137	131	126	119	120	122	120	120	123	121	126	133	140
188	180	174	167	162	159	150	141	136	135	128	120	118	117	117	118	117	120	128	135
180	178	174	169	165	161	154	144	143	143	137	125	118	115	114	114	116	120	125	130
176	173	171	168	164	162	157	149	147	145	140	127	119	115	113	110	113	117	123	127
168	169	169	166	162	158	155	151	149	144	140	129	120	118	113	108	110	117	123	125
162	165	165	164	161	157	154	152	148	144	140	135	129	121	114	104	107	114	120	122
[illegible]	161	162	162	159	156	154	150	148	145	142	141	133	124	113	102	100	108	114	117
[illegible]	156	158	160	157	154	151	148	146	143	142	139	131	123	112	102	100	105	110	112
149	152	154	155	154	151	148	146	143	141	137	131	126	118	110	102	98	101	106	109
147	150	150	150	150	148	145	143	141	135	130	124	118	113	106	100	98	102	106	108
146	147	146	146	147	146	142	138	134	129	124	119	114	110	104	97	97	102	104	105
143	145	143	141	143	143	140	134	128	121	120	116	111	107	102	96	94	98	101	103
140	142	142	139	139	140	136	130	123	119	115	111	107	103	101	94	89	92	99	100
136	138	138	137	135	134	131	126	119	116	113	108	103	100	95	89	80	86	93	95
130	132	134	133	130	129	128	122	117	113	109	105	101	97	93	88	80	85	90	90
125	128	131	130	127	125	122	120	116	111	108	104	101	96	92	87	80	83	88	87
122	126	128	128	126	122	117	115	112	108	104	102	99	95	90	87	81	81	84	81
123	125	125	124	123	121	115	110	108	104	99	97	94	91	86	82	80	79	79	77
120	122	122	121	119	116	111	105	103	99	95	91	90	84	80	79	78	78	78	77
111	117	118	115	114	110	105	100	96	92	88	84	83	80	79	79	80	79	79	78
103	109	110	108	107	104	99	96	92	87	80	80	79	79	79	80	85	85	80	79
98	102	102	101	101	99	95	91	88	87	80	79	79	79	80	86	90	92	88	89
95	95	95	95	95	94	90	85	81	81	80	79	80	84	88	93	95	98	95	100
94	89	89	91	91	89	84	80	80	79	79	80	80	91	99	102	101	105	102	105
97	82	85	87	87	84	80	80	79	79	80	82	89	98	105	110	109	111	111	111
94	80	80	80	80	80	80	79	80	80	83	95	101	105	112	117	116	118	121	120
91	80	80	80	80	79	79	80	80	89	91	103	109	115	121	123	122	121	125	123
86	80	80	80	80	80	80	80	89	100	101	103	118	121	125	127	127	127	129	123
80	80	80	80	80	84	86	90	95	104	110	113	123	127	130	132	134	135	132	125
80	80	80	80	87	89	91	94	101	107	117	120	123	130	135	136	139	141	138	131
80	84	89	89	91	93	94	97	104	114	122	125	128	135	141	141	143	145	143	137
[illegible]	93	98	97	95	97	97	100	109	117	124	131	135	142	145	146	148	148	144	134
[illegible]	101	103	102	98	102	101	104	113	123	130	137	138	144	149	153	154	152	145	138
100	104	107	107	104	102	107	109	115	122	132	141	145	148	154	158	160	157	149	143
100	105	110	111	110	111	112	116	118	122	134	144	149	154	159	164	164	161	152	142
103	107	112	116	118	121	118	123	127	130	136	146	153	159	164	166	166	162	151	140
104	107	113	119	123	127	126	130	137	142	141	149	158	164	168	168	166	161	147	136
99	105	113	121	124	131	136	137	143	151	153	156	164	168	169	168	165	156	146	138
99	106	109	117	122	126	137	143	149	159	163	164	168	172	172	168	159	157	154	152
97	99	100	109	115	121	134	142	151	160	166	169	174	179	180	174	168	166	164	163
97	101	104	110	121	127	136	141	146	156	165	170	178	181	182	181	178	176	172	167
96	101	103	108	115	118	123	132	139	150	163	170	176	182	185	184	183	181	173	166

Fig. 9. Contour map and search graph (Lexington Reservoir, Santa Clara County, California). (a) Contour map showing a 50 × 20 sampling grid. Note: the digital representation of each contour is labeled as ⊙ 8*120, for example, where the first symbol encircles a point on the contour, the second is the sequence number of the contour; and the third symbol is the contour's elevation. (b) Search graph showing the contour sequence numbers as nodes, and Y (yes)- and N (no)-transitions as links



3.3. Constructing Synthetic Radar Images

Unlike signals measured by the microwave scanning radiometer, signals measured by radar are dependent on elevation and slope as well as the materials on the terrain surface. Hence, to construct a synthetic digital radar image, it is necessary to have a computer searchable representation for both the terrain elevation contours and the surface materials, roughness, and specular properties. Because such surface objects as the opposite banks of rivers and communication or power line towers are prominent reflectors, it is also necessary to represent certain line structures to obtain a more complete terrain model.

The capability of the terrain information system to handle terrain elevation contour data is illustrated by the example in Figure 9. Elevation contours were encoded with COPIER such that the systems of concentric contours (21, 20, 19, 18, 17, 16), (15, 14), (13, 12, 11, 10, 9), (8, 7, 6), (5, 6, 4, 3, 2) were represented.

The Figure 9(b) search graph was produced from this contour encoding sequence by STOR. Because of the simplicity of this representation, it is very easy to find the set of contours passing near a particular point on the terrain. Also it is easy to determine the minimum distances between the point and these contours so that an accurate estimate of the points elevation can be made. To illustrate this, POL was used to derive the 1000 elevations for the 50 × 20 sample grid shown in Figure 9(a). The elevation at each point was computed using eq. (4). The results are tabulated in Table IV. POL performance was analyzed and it was found that elevations can be derived on the UNIVAC 1108 at the effective rate of 50 to 100 sample points per second for scenes comparable to this example. It can be shown that this rate does not change substantially even for significantly larger numbers of contours.

4. Summary

The paper describes a novel computer-searchable representation for each of the basic pictorial features: region or area coverage, contour maps, and line structures. It is shown that these representations provide a very rapid means of searching large files, which have practical storage requirements, on geometric coordinates as well as on attribute values. An application of this representation to handling terrain information is described. The computer-based implementation used comprises a prototype system which integrates the three key pictorial data handling functions: converting pictorial structures to digital form; filing the searchable representation for these structures; and retrieving and analyzing selected data from the files.

The computer-searchable data structure is shown to possess very useful algebraic and geometric properties which lead to efficient means of representing the relationships and attribute values of pictorial features.

Procedures have been developed and programmed which use these properties: to determine whether a point lies within a closed boundary; to compute the area contained by a closed boundary; to generate the closed boundary representing the union or intersection of two closed boundaries; and to determine the neighboring boundaries to a point, and the minimum distances between them and the point.

The computerized data handling system was applied to the construction of synthetic microwave radiometer and radar images of terrain scenes from aerial photographs and elevation contour maps of these scenes. The conversion procedure for radiometer data has been evaluated by two Air Force photointerpreters; the images constructed from their conversion results were compared to the actual images and found to fall well within acceptable limits. The system has been used successfully to construct over 500 digital radiometer images ranging in size from 12 × 12 to 120 × 120 samples. It has been proven that the system is far more economical than the usual manual methods, provides for quicker turn-around time, and produces accurate images more reliably.

Acknowledgments. The author wishes to thank the many people who were involved in this investigation. Dr. W.G. Eppler, J.K. Matsunaga, and G.H. Clancy deserve particular mention because of their direct involvement in defining the radiometer image prediction problem and aiding in its solution. M. Tannenbaum's assistance in solving contour map representational problems and programing the results was most important. The author acknowledges the contributions of O. Firschein in reviewing the original manuscript, and M.N. Collins in editing and supervising the preparation of this paper.

The support and encouragement of Dr. M.A. Fischler and Dr. C.E. Duncan are greatly appreciated.

Received November 1970, revised December 1971

References

1. Proc. 7th Internat Symp. on Remote Sensing of Environment, The U of Michigan, Ann Arbor, Mich , May 1971.
2. Computers in geoscience research. 1971 IEEE Internat Geosci Electronics Symp , Washington, D.C., Aug. 1971.
3. Boyell, R L., and Ruston, H. Hybrid techniques for real-time radar simulation. Proc. AFIPS 1963 FJCC, Vol. 24, Spartan Books, New York, pp 445-458.
4. Development of a hybrid radar land mass simulator. Pennsylvania Res Assoc. Engineering Rep , USN Training Device Center Tech. Rept No. NAVTRADEVCEN 68-C-0155-3, Jan. 1969
5. Boehm, B.W. Tabular representations of multivariate functions with applications to topographic modeling Proc. ACM 22nd Nat. Conf , 1967, pp 403-415.
6. Gray, J.C. Compound data structure for computer aided designs; a survey. Proc ACM 22nd Nat. Conf., 1967, 355-365.
7. Freeman, H , and Morse, S.P On searching a contour map for a given terrain elevation profile *J. Franklin Inst. 284*, 1 (July 1967), 1-25.
8. Morse, S.P. Concepts of use in contour map processing. *Comm. ACM 12*, 3 (Mar. 1969), 147-152.
9. Freeman, H. On the encoding of arbitrary geometric configurations. *IRE Trans. EC-10*, 2 (June 1961), 260-268.
10. Pfaltz, J.P. MANS, a map analysis system. TR-67-42, U. of Maryland Comput. Sci. Center, College Park, Md , Feb. 1967.
11. Butt, E.B., and Snively, J.W. Jr. The PAX II picture processing system. TR-68-67, U. of Maryland Comput. Sci. Center, College Park, Md , May 1968.
12. Apfelbaum, H.C. Automatic Cartographic System (ACS) (Mod II) Vols. I and II. RADC-TR-69-339, Pennsylvania Res. Assoc., Inc., Mar. 1970.
13. Eppler, W.G., and Merrill, R D. Relating remote sensor signals with groundtruth information. *IEEE Proc. 57*, 4 (Apr. 1969), 665-675.
14. Eppler, W.G., et al. Interactive systems for remote sensor data analysis Proc. 7th Internat. Symp. on Remote Sensing of Environment, The U. of Michigan, Ann Arbor, Mich., May 1971.
15. Macon, N., and Kiefer, M. Computer Manipulation of Digitized Pictures. Proc. AFIPS 1971 SJCC, Vol. 39, AFIPS Press, Montvale, N J., pp. 407-413.
16. Spanier, E.H. *Algebraic Topology*. Sect. 4.17.5, McGraw-Hill, New York, 1967.
17. Jacobsen, J.D. Geometric relationships for retrieval of geographic information. *IBM Syst. J. 7*, 3 & 4 (1968), 331-341.
18. Loomis, R.G. Boundary networks. *Comm. ACM 8*, 1 (Jan. 1965), 44-48.
19. Merrill, R.D. *Encoding, filing, and retrieving contour and area map information.* 6-83-71, Lockheed Palo Alto Res. Lab. Nov. 1970.
20. Eppler, W.G., and Simas, G.A. Theory of map matching. LMSC-D051104, Lockheed Palo Alto Res. Lab., Mar. 1968.
21. Kunzi, K., Wuthrich, M., and Schanda, E. A mm-wave scanning radiometer for terrain mapping. Proc. 7th Internat. Symp. on Remote Sensing of Environment. The U. of Michigan, Ann Arbor, Mich., May 1971.

Automatic CONTOURING

T. K. Peucker
Department of Geography
Simon Fraser University
Burnaby, B.C., Canada

Since man first attempted to represent parts of the Earth's surface in reduced form, the problem of representing the third dimension, height, on a two-dimensional plane has occupied him. In the early times of cartography, mountains and hills were portrayed merely as landmarks and little care was taken to represent their true absolute or relative heights. With the increasing availability of accurate surveys and a growing demand for precise surface information, measurable representations of three-dimensional surfaces in two-dimensional planes became more and more frequent.

Different methods exist to accomplish this (Peucker, 1972~~a~~). We will restrict ourselves to the group of planimetrically correct representations. Since relief-shading and hachuring do not allow for precise measurements on the map, the production of "isograms" or horizontal contours is the most frequently used alternative. However, contour maps demand considerable time by the user for analysis since the intersection of the surface with horizontal planes does not reveal the surface's macro-features. The increasing availability of programs for perspective views (Douglas, 1972) and different types of relief shadings (Peucker, 1972~~b~~) suggests the addition of these representations for a quicker understanding of the surface.

The types of surfaces which have been mapped by isograms are numerous. Thrower (1972) summarizes and catalogs many types of horizontal contour lines for which "isogram" is the general term. They are all lines of constant value following the surface, i.e., traces of the intersection of horizontal planes with the surface. The planes may be equispaced but this is not a requirement. Indeed, many studies (Warntz, 1966) have shown that unevenly spaced, surface specific contours give better understanding.

The conceptual framework of automatic contouring is poorly defined, with the effect that the user may be unclear about underlying ideas and does not know the specific assumptions implicit in the type of contouring he is using. Often added to this air of uncertainty is the fog of implicit interpolations which in some cases are so badly documented that they can only be called the "secret" of the distributors of the program. The result is a pagan confidence in the reliability of interpolation and contour programs which should worry the conscientious programmer and induce him to document his procedure better than is usually done.

It is a general belief that contour line programs can only handle single-valued surfaces; that is, surfaces without vertical or overhanging cliffs. This is true only because the usual data structures for surfaces do not allow for more than one z-value for any point in x-y space. Programs exist, however, which allow for discontinuities in the first derivative. For example, the barrier package in SYMAP interrupts the interpolation process along specified lines and thus builds cliffs. As in traditional cartography, contour lines stop at one end of cliffs and continue at the other end.

Two types of contouring are usually distinguished; point contouring and line contouring. In the first, a regular raster of points fills out the map area and the entirety of a point is either part of a contour or between two contours. The points have no spatial extension in the second case and areas between three or more points are considered subsurfaces along which lines of constant z-value are constructed.

Although the two types of contouring appear different, their solutions show many similarities and they can be treated together. A much more important distincution is in the different methods for data storage during processing. Since the loading of information from auxiliary storage such as tape or disc is time-consuming compared to ~~in-core~~ memory processing time, loading operations should be kept to a minimum. This means that a single data-record should never be loaded twice in one run.

This results in two processing types for contouring a surface. The first group of procedures loads and processes the entire data set at once. The second loads sections of the data and processes them individually and sequentially, since the whole data set may be too large to be kept in core at one time.

THE IRREGULAR GRID

Most contour procedures are based on a regular array of heights or a regular raster of points. However, contouring of an irregular mesh of data points without first interpolating to a regular grid can be advantageous, especially in small computers where the smaller number of data points can make contouring feasible, and in output systems which have to economize on the number of lines drawn.

The first step in contouring an irregular mesh of points is to divide the map into a set of polygons. Although triangles are most frequently used (ARCON, 1968; IBM, 1965), quadrilaterals also have been used (Kaplan and Papetti, 1971).

ARCON (1968) gives a detailed description for "triangulation" of points. The procedure first creates a starting boundary on the left margin of the map, using the left-most points. This boundary is used as the initial triangulation front which is then advanced to the right by incorporating successive points. As soon as a point is processed, it can be discarded since it is not used in any further computations. The program can therefore process a relatively large number of points even on small computers since only a small proportion of the data must be kept in core at any one time.

Since the contouring is done at the same time as the triangulation, transfer of data from slow to fast storage devices is kept at the minimum possible level. The program also allows for smoothing of contour lines.

Since a polynomial surface of the first order (a plane) has three terms (ax + by + c), three data points are necessary and sufficient to define the plane. If a contour crosses such a plane, it must intersect two of the sides of the triangle. It is therefore necessary only to find these two intersecting points and link them by a line. If the map is filled exhaustively by the set of triangles, the lines will join automatically since one side will always be shared by two triangles (except at the rim of the map).

If the polygon has four or more sides the assumption that the corner points lie in a plane is somewhat risky. Kaplan and Papetti (1971) therefore propose a method which allows construction of smooth surfaces with convex quadrilaterals. They interpolate along a family of line segments connecting opposite sides of the quadrilateral, where the two end points for each line segment of the family lie the same fractional distance along the two opposite edges. These line segments then can be used as intermediary intersects for contours.

The last procedure shows that the question of interpolation must often be considered in contouring problems. It is therefore understandable that many people believe they are discussing contouring when they describe interpolation algorithms. The concepts of contouring and interpolation will be kept strictly apart in this section. It will be assumed that the surface is either sampled on a regular grid or has been interpolated to a regular grid in a separate process.

THE REGULAR GRID

It has been said that only the triangle defines a plane unequivocally. Several attempts have been made to create regular triangular grids for contouring (Bengtsson and Nordbeck, 1964). This approach has been largely abandoned because it is believed that an array in triangular form is difficult to store in systems which are implicitly based on a rectangular storage array. This is incorrect, because any map can be rotated 60°, interpolated in normal (rectangular) fashion and rotated back for plotting of contours. This approach will not be discussed further as it has not been applied, and we will consider only rectangular grids.

The specific task is to find points along a line and to decide whether a point is part of a contour or not (that is, whether the point should become ~~a~~ black ~~contour~~ or remain white). The procedure can be considered as a series of switching decisions which can be expressed in one bit per point. The resulting contour map is therefore often represented in the form of a bit-plane.

Since the construction of a contour line can be based on a variety of repeated point or cell operations, these operations will be discussed with a single unit before "strings" of units are considered. For the raster-type method two different approaches are known. The first is to switch a point-bit to positive if the point's height is within a margin above or below the contour value. This is done by adding the margin to the height, dividing the result by the contour interval and testing whether the remainder is less than or equal to the doubled margin, or dividing the height by the doubled margin (which must be an integer portion of the contour interval) and rounding the result up or down to the closest integer. Those points whose values are multiples of the quotient of the contour interval over the doubled margin are contour points.

This approach, which is called quantization in the field of picture processing (Rosenfeld, 1969), yields contours of varying width, depending on the slope of the surface. Although this might not be desirable for a cartographic presentation, it corresponds to the reliability margins of surface sampling and corresponds to the type of isarithms proposed by Tobler (1970). Picture processing provides algorithms to thin these lines.

The second method of finding contour points in a point raster is to test each point to determine if one of its neighbors is "on the other side" of a contour level. That is, to determine whether a contour passes between the point and any one of its neighbors. The fastest algorithm is to divide every value by the contour interval and delete the part to the right of the decimal point. Contour points can then be found by testing whether any of the neighbors has a value different from the value of that point. This procedure, usually referred to as edge detection in picture processing (Rosenfeld, 1969), will yield contour lines two points thick which can again be reduced by line-thinning procedures.

When contouring a rectangular grid of heights the task is to test one cell bounded by four data points and determine if a contour touches or crosses (i.e., whether it intersects at least two lines either between the two end points or at one of them) and then to find the crossing and touching points. The number of crossing points can be one or three if the contour passes through one or more of the corner points, two in the normal case of intersection of the contour with a cell on a slope, or four if a saddle-point lies within the cell.

The third case has caused geographers considerable consternation because several solutions exist (Mackay, 1953; Morrison, 1968). Mathematicians willingly accept that a contour line can cross at a saddle-point, but this provokes horror and mistrust in one's own faith for geographers. Most contour programs have special procedures to handle the saddle-point problem and avoid crossing.

Several contouring algorithms are well described and widely implemented. Murray (1968) subtracts values of the corner points from the contour level and first works only with the signs of the differences. With a series of conditional branches (a tree of approximately two dozen arithmetic IF-statements) the algorithm decides whether the contour lies outside the cell, on one or more of the corner points or crosses any of the sides. The result is one of 41 possible situations. For 12 of these, one crossing of a side must be computed, for 10 two crossings must be computed, and for one, four crossings must be found. In the last case, Murray always connects opposite sides so the contour lines cross (Murray is a mathematician) and leaves it to the user to "correct" the lines manually if he wishes.

Thirty-two of Murray's 41 situations have at least one corner point on the contour, although such cases occur infrequently in practice. It has therefore been suggested (Coulthardt, 1969; Walton, 1969) that such values be changed slightly by switching the least significant bit.

Murray (1968) ignores the fact that a plane is overdefined with four points and simply constructs straight lines between intersection points. This is usually satisfactory if the mesh width of the grid is dense enough. Although the results are cruder, the computation is faster by an order of magnitude as compared to the other extreme of a contouring algorithm, CALCOMP's General Purpose Contouring Program (CALCOMP, 1968). The GPCP adds the eight next neighbors to the four corner points and fits a second-order least-square polynomial to the 12 points. The resultant contours are smooth though not necessarily accurate.

Hebin (1969) dissects the cell into four triangles by computing the midpoint as the mean of the four corner points. If higher resolution is desired, an option dissects the cell into four rectangles and creates four triangles in every rectangle.

SEARCHING THE ARRAY

Dayhoff (1968) uses two triangles per cell but the algorithm is flexible with respect to which of the possible triangles it uses. Coulthard (1969), in a development based on Dayhoff (1968), uses four triangles in the cell. It will be described here in more detail because it searches through a set of cells.

Once a first contour intersect has been found along a side of a cell, the higher corner point on the side is called the reference point (RP) and the lower point the sub-point (SB). The contour line is then traced by rotating clockwise around RP from SP where the midpoint of the cell is one of the points which can be reached during the rotation. If the next point reached is lower than the contour, the point becomes a new SP, the contour intersect is calculated and the rotation is continued. If the new point is higher than the contour, the point becomes the new RP and the rotation starts from the last line with an intersection. Every RP is flagged so it cannot be used again, and the contouring stops when the rim of the map or a flagged RP is reached.

This immediately presents the problem of how to efficiently handle an array of points or cells. The choice is not simple; not only must the available main memory size be considered, but also the ease and necessity of smoothing, labeling, and providing similar facilities. In general, however, the higher the requirements for these additional facilities, the higher the proportion of the array which must be kept in main storage at one time.

There are two ways to treat point rasters. The first is by processing row-by-row or column-by-column. This procedure is efficient as it uses the inherent storage structure of the computer and can be used on small systems because three rows or columns kept in main memory at one time are usually sufficient.

The alternative procedure is "parallel processing," often used in its simulated form in picture processing (Johnston, 1970) and an area of active hardware development. Parallel processing is a very elegant and efficient way to perform local operations such as quantization and edge detection. The approach divides a procedure into a number of very simple steps and performs them simultaneously on a series of points. Parallel processing has a methodological appeal to geographic studies because it allows a conceptual treatment of surfaces which is fundamentally different from the sequential approach.

The contouring of an array of cells can also be performed in two ways. The first is sequential, either row-by-row or column-by-column. For every cell all possible contour lines are computed and drawn at a time so that the cell never has to be loaded again for computation. Thus, only two rows or columns must be kept in core at any one time. This approach, however, gives little flexibility for smoothing or labeling of contours. Smoothing can be done by making the cells very small and the steps very short but this is computationally extravagent. The approach is also very wasteful in plotting time in those systems which do not allow for the pen to skip to the new location when the pen is up.

An alternative method of contouring a regular array of cells is by searching through the array moving from one cell to the next as the intersections guide the procedure. The algorithm first searches along the borders for a (usually down-hill) intersection of a cell side with the contour level. As soon as a point on a contour line is reached, the contour is followed until it reaches the map border again. When the search along the border is completed, the rest of the array has to be scanned for possible inside contour loops.

It is clear that with this approach the whole array, or at least large rectangular sections, must be kept in core. The approach is not fast in comparison to the first since the array has to be scanned for every contour level. The great advantage of this method is that it does not create individual contour line segments but strings of segments which can be plotted as they are produced, with concurrent smoothing and labeling of the lines.

CONTOUR SMOOTHING

It is a requirement for all but the most superficial uses of contour maps that the contours be smooth. We know that surfaces are smooth and round and therefore will not accept contour lines with sharp corners; in fact, we often even believe that the accuracy of the map increases with its smoothness. This is sometimes true in the manual production of contour maps when an experienced cartographer, surveyor, or meteorologist adds his knowledge of the surface behavior which is not implicit in the x, y, and z coordinates of the available points. The same cannot be said about automatic contouring where all that is known is contained in the original sampling points. The more interpolation and smoothing, the greater the possible deviations from the original surface. The only accurate representation of a surface is the display of the sample points themselves, save for measurement errors, of course.

One could argue that the demand for accuracy is an inheritance from the surveyor, which makes sense at a scale of 1:1000 or more but might be questioned at small scales for non-topographic maps. In short, smoothing is recommended because it pleases the eye, but its results must be interpreted with caution.

Smoothed curves may or may not pass through the original data points. This can be regarded as a basic distinction. Since the requirements are the possibility of overriding certain values by chosen values, blanking whole areas, and specifying contour spacing. Another group of desirable features is the handling of grids with varying grid spacing, the automatic blanking of areas where data are too sparse, and the automatic indication of areas where data are unusually dense and/or varied.

The last group of features are useful for the evaluation of the contoured surface and give some safety checks for doubtful data. The second helps in understanding the map, and suggests the possibility of some interactive procedures which are largely ignored in the literature. Blanking effects the "artistic" quality of the contour map.

It is the intent of many contour programs to produce maps which can be presented or published as they come off the plotter. This is not always wise since computation costs increase disproportionately with quality. There is a point where a corrective touch from a draftsman can accomplish improvements which can only be achieved by very large and expensive programs. We should keep in mind that the cartographer will always be ahead in esthetics. Let the computer do the number-crunching and rough work and let the cartographer do the thinking and supply the fine touch.

Please supply the missing details!

ARCON, 1968, CONTOUR, an automatic contouring program: Advanced Research Consultants, Wakefield, Mass., ______ p.

Beck, J., 1967, Subroutine ZWDIMP: Technische Hochschule, Munich, ___ p. personal communication

Bengtsson, B.E., and S. Nordbeck, 1964, Construction of isarithms and isarithmic maps by computer: BIT, v. 4, p. 87-105.

CALCOMP, 1968, GPCP, general purpose contouring program: California Computing Products, Anaheim, Calif., 130 p.

Carnahan, B., H.A. Luther, and J.O. Wilkes, 1969, Applied numerical methods: ______ , New York, ____ p.

Coulthard, W.J., 1969, Contouring a grid. Program description (UBC CNTØUR): Computing Centre, Univ. British Columbia, Vancouver, B.C., 15 p.

Dayhoff, M.O., 1963, A contour-map program for x-ray crystallography: Communications, Assoc. Computing Machinery, v. 6, no. 11, p. 620-622.

Douglas, D.H., 1972, VIEWBLOK: a computer program for constructing perspective view block diagrams: Revue de Geographie de Montreal, v. 26, p. 102-104.

Hebin, O., 1969, Computer drawn isarithmic maps: Geografisk Tidsschrift, v. 68, no. 2, p. 50-63.

Holroyd, M.T., and B.K. Bhattacharyya, 1970, Automatic contouring of geophysical data using bicubic spline interpolation: Geological Survey of Canada, Ottawa, Ontario, Paper 70-55, 16 p.

IBM, 1965, Numerical surface techniques and contour map plotting: IBM Data Processing Application, White Plains, New York, 36 p.

Johnston, E.G., 1970, The PAX II picture processing system, in B.S. Lipkin and A. Rosenfeld, eds., Picture processing and psychopictorics: Academic Press, New York, p. 427-512.

Kaplan and Papetti, 1971, to be supplied by author

Mackay, J.R., 1953, The alternate choice in isopleth interpolation: Professional Geographer, v. 5, no. 4, p. 2-4.

McCormick, B.H., 1963, The Illinois pattern recognition computer - Illiac III: IEEE Trans. Electronic Computers, v. EC-12, p. 791-813.

Morrison, J.L., 1968, The effects of sampling and interpolation in isarithmic mapping: Ph.D. thesis, Dept. Geography, Univ. Wisconsin, Madison, ___ p.

Murray, F.W., 1968, A method of objective contour construction: The Rand Corp., Santa Monica, Calif., 32 p. Available from NTIS as AD666830.

Peucker, T.K., 1972a, Computer cartography: Assoc. American Geographers, Commission on College Geography, Resource Paper No. 17, 75 p.

Peucker, T.K., ~~1972b,~~ The computer versions of three methods of relief representation: ~~presented at Inter. Cartographic Assoc. Congress,~~ Ottawa, ~~Ontario, Canada,~~ ~~p.~~

Postma, R.J., 1970, A programming technique for curve fitting non-singular data strings: Environmental Systems Research Inst., Redlands, Calif., 7 p.

Rosenfeld, A., 1969, Picture processing by computer: Academic Press, New York, 196 p.

Thrower, N.J.W., 1972, Maps and man, an examination of cartography in relation to culture and civilization: Prentice-Hall, Inc., Englewood Cliffs, N.J., ___ p.

Tobler, W.R., 1970, Data display and presentation index paper, in R.F. Tomlinson, ed., Environment information systems: ________ , Ottawa, p. 128-137.

Walters, R.F., 1969, Contouring by machine: a user's guide: Bull. American Assoc. Petroleum Geologists, v. 53, no. 11, p. 2324-2340.

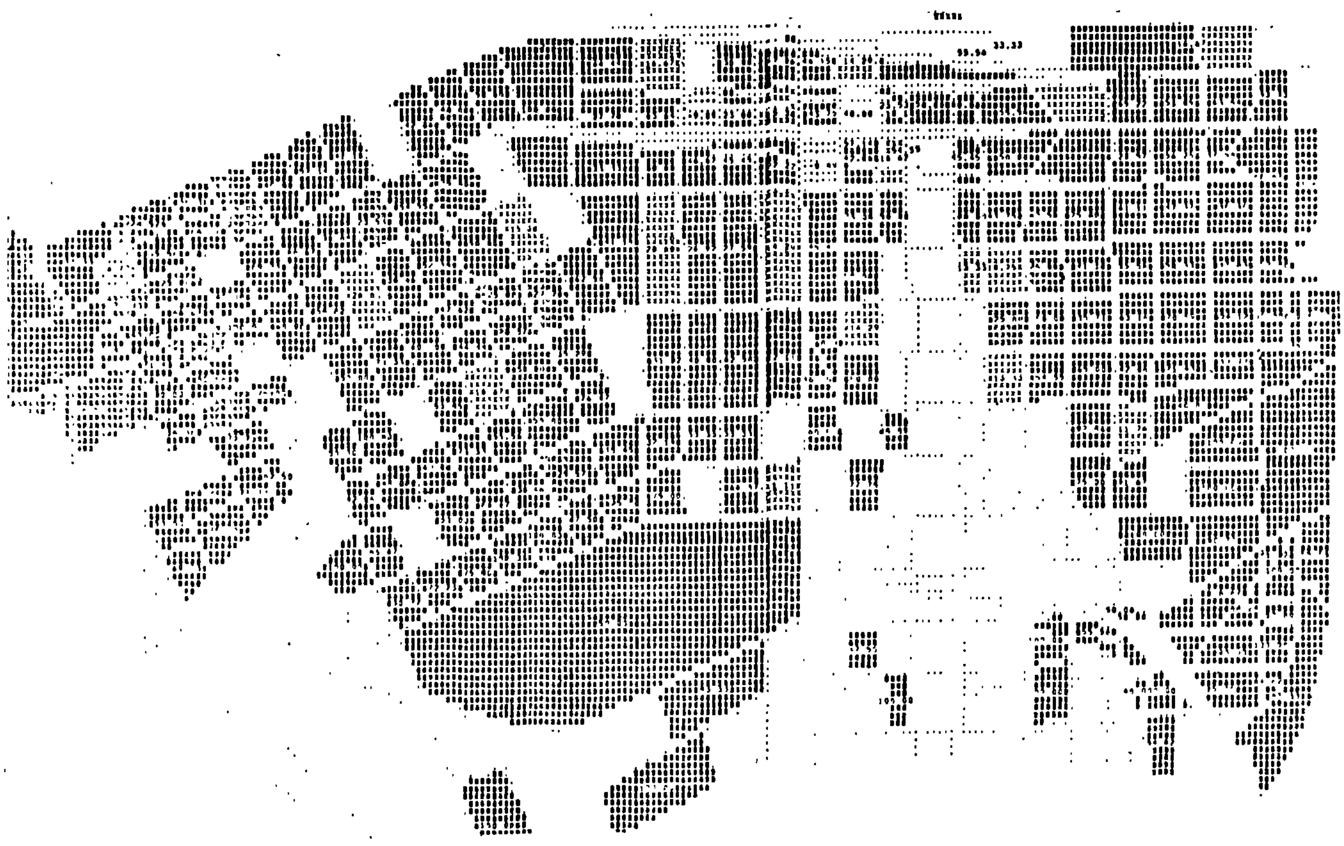
Walton, D.F., 1969, Program CTRCTR, program description, program library: Computer Center, Univ. Utah, Salt Lake City, 10 p.

Warntz, W., 1966, The topology of a socioeconomic terrain and spatial flows: Regional Science Assoc., v. 17, p. 47-61.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

PROGRAMA SYMAP versión DESFI/B-6700
Manual del usuario



LABORATORIO DE PLANEACION
DESFI/UNAM

I.1 INTRODUCCION

Actualmente las universidades y algunas oficinas de gobierno - han encontrado que los mapas desarrollados por medio de computadoras tienen grandes ventajas sobre aquellos que se producen con papel, lápiz y escuadras. Los mapas resultan valiosos para planificadores, geógrafos y cartógrafos en el análisis, registro y comunicación de información eficientemente. En forma paralela a este desarrollo, aumenta la disponibilidad de diferentes tipos de identificadores geográficos y la demanda de mapas automatizados que muestren densidad de población, distribución de grupos sociales en las ciudades, mapas de uso del suelo, posibilidades de empleo, de la lluvia y aspectos similares. También está teniendo lugar una mayor utilización de las computadoras y la cartografía automatizada en la mayoría de las áreas de la vida humana. El desarrollo de la cartografía automatizada inicialmente se vió obstaculizada por la falta de una instrumentación gráfica satisfactoria y por el hecho de que las comunidades académicas y profesionales creían, incorrectamente que resultaría muy difícil manejar aspectos no numéricos mediante computación. Los equipos de computación gráfica pueden manejar los tres tipos de características de los mapas: puntos, líneas y superficies. Los tres son símbolos especiales de diferentes dimensiones. Un mapa contiene diversos tipos de símbolos como signos o caracteres sin dimensiones específicas, pero generalmente con localizaciones precisas. - Los dispositivos para la graficación de mapas más frecuentemente utilizados, son las impresoras de línea, los graficadores digitales y los tubos de rayos catódicos. Las impresoras de línea, son realmente los dispositivos mas comunmente disponibles; estas pueden producir dibujos como una máquina de escribir, sacando ventaja de los diferentes tonos de gris de los caractéres, de manera que éstas, pueden producir mapas a la velocidad de 2000 líneas por minuto por lo cual constituyen instrumentos sumamente rápidos. Los inconvenientes de la impresora de línea son su relativa baja resolución de 10 caracteres por pulgada horizontalmente y ocho caracteres verticalmente y la relativa rápidez con que cambian de intensidad las impresiones, lo cual da por resultado variaciones notables de una hoja a otra. Los graficadores digitales trazan los mapas definiendo y conectando una secuencia de coordenadas; mientras un tambor gira sobre el eje de las yes, el brazo que lleva la pluma se desliza sobre el eje de las equis. Los graficadores planos operan una pluma que se mueve tanto en dirección del eje de las equis como de las yes sobre una mesa. Puede disponerse de graficadores digitales con una gran gama de resolución, desde 100 hasta 10,000 caracteres por pulgada, una rápidez baja ó alta, de una a cuarenta pulgadas por segundo y una plumilla intercambiable ó una cinta con una cabeza de mayor precisión, dependiendo del precio que se pueda pagar. El tubo de rayos catódicos es familiar para todo mundo, ya que se utiliza en las-pantallas de los aparatos de T.V.; son sumamente rápidos y con frecuencia interactivos, pero todavía demasiado caros para la-producción de mapas finales. Sin embargo como un dispositivo-inicial, los tubos de rayos catódicos pueden ser hoy de enorme utilidad porque permiten trabajar con el mapa antes de ser elaborado con un graficador digital.

Desde que se desarrollaron los diversos programas para la producción de mapas, pocos de ellos son los que pueden utilizar - otras personas aparte de quienes los han desarrollado; SYMAP - es uno de estos programas. En la actualidad existen mas de -- 500 usuarios de este programa en todo el mundo. El programa - se desarrolló originalmente y actualmente es financiado por el Laboratorio de Computación gráfica y Análisis Espacial, en la-Escuela Superior de Diseño de la Universidad de Harvard. La - posibilidad de utilización de SYMAP se basa en el hecho de que todos los centros de computación tienen los dos ingredientes - necesarios para elaborar mapas con este programa: una computadora y una impresora de línea estándar. La impresora de línea, del mismo modo que la máquina de escribir que mencionamos, produce dibujos sacando ventaja de los diferentes tonos de gris - de las letras y de la sobreimpresión de diferentes caracteres-alfanuméricos, pero a la velocidad de entre 600 a 1200 líneas-por minuto y son generalmente 132 símbolos por línea. La an-- chura del mapa de salida (132 símbolos) no es una restricción-ya que se pueden imprimir mapas en tiras y después ensamblar-- las. Otras importantes características son su flexibilidad de entrada y la gran variedad de salida. La forma mas empleada - de entrada es con tarjetas perforadas, pero toda o parte de la información puede ser leída a partir de cintas o discos magnéticos. Este proceso normalmente sería el resultado de alguna-computación previa, tal como la utilización de datos estandarizados, análisis de regresión, análisis factorial y otros similares. Hay un formato estándar para las tarjetas de SYMAP que le proporciona a la computadora lo que ella necesita sobre --- ciertos datos, campos o grupos de columnas pero todo esto puede cambiarse bajo el control del programa. SYMAP está escrito en FORTRAN IV nivel G y consta aproximadamente de 6000 instrucciones, contenidas en un programa principal y 49 subrutinas. - Para producir un mapa deben de conjuntarse diversos paquetes y opciones que se describen en los capítulos siguientes.

I.2 MAPAS QUE SE PRODUCEN EN SYMAP

1.- MAPAS COROPLETAS.

Representan información cuantitativa ó cualitativa referida a zonas ó regiones con fronteras definidas en las que se divide el área de estudio (países en el mundo, estados ó regiones en un país, municipios en un estado, etc.). El valor de cada variable es uniforme en cada una de las zonas ó regiones. Se presentan discontinuidades en los límites ó fronteras.

Paquetes necesarios:

A-CONTORNOS, E-VALORES y F-MAPA.

Paquetes opcionales:

C-LETREROS y E-INDICE.

2.- MAPAS ISOPLETAS.

Representan líneas que conectan todos los puntos con la misma altura ó valor numérico, esto es, se mantiene un valor uniforme en toda su longitud. Estas líneas aparecen cuando la superficie creada a partir del algoritmo de interpolación es intersectada por planos horizontales a diferentes niveles que son determinados por la escala del mapa y el rango de los datos. Sirven para mapear información espacialmente contínua. Deben definirse los límites del área de estudio y los puntos donde se realizaran las observaciones. El programa utiliza los valores dados en los puntos de observación para interpolar basándose en los valores y las distancias a los puntos.

Pueden utilizarse diferentes opciones en los paquetes D-BARRERAS y F-MAPA para modificar el algoritmo de interpolación creando modelos espaciales más precisos.

Paquetes necesarios:

A-DELINEAMIENTO, B-PUNTOS DATO, E-VALORES y F-MAPA.

Paquetes opcionales:

C-LETREROS, D-BARRERAS y E-INDICE.

3.- MAPAS DE PROXIMIDAD

A cada localización de impresión en el mapa se le asigna el valor correspondiente al punto dato mas cercano. Es útil para representar valores cualitativos en los que una superficie continua (isopleta) sería inadecuada. Ejemplos áreas de mercado, tipos de suelo, etc.

Paquetes necesarios:

A-DELINEAMIENTO, B-PUNTO DATO, E-VALORES y F-MAPA.

Opciones 31, 36, 37.

Paquetes opcionales:

C-LETREROS, D-BARRERAS y E-INDICE.

4.- MAPAS BASE

Se representan zonas, contornos, puntos-dato y leyendas sin que se grafique ningún valor. Se realizan antes de la producción de mapas temáticos con el objeto de corregir errores en la geo-codificación. Son menos costosos, y significan ahorro. Pueden ser de cualquera de los 3 tipos mencionados.

Paquetes necesarios:

A-DELINEAMIENTO ó A-CONTORNOS y F-MAPA.

Paquetes opcionales:

C-LETREROS

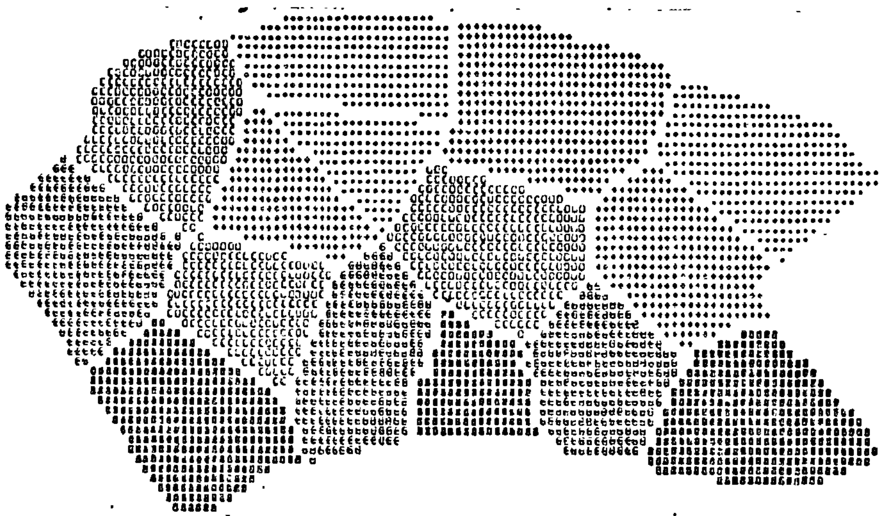


Fig. 1 MAPA COROPLETA

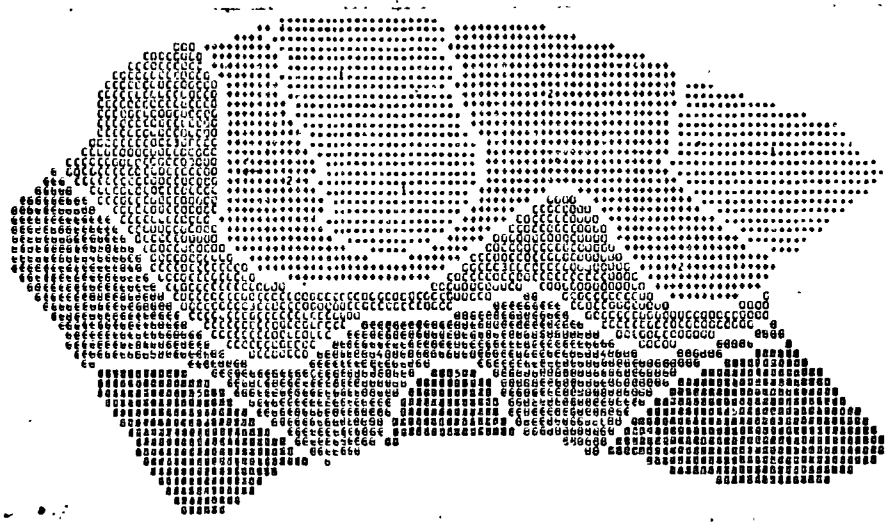


Fig. 2 MAPA ISOPLETA

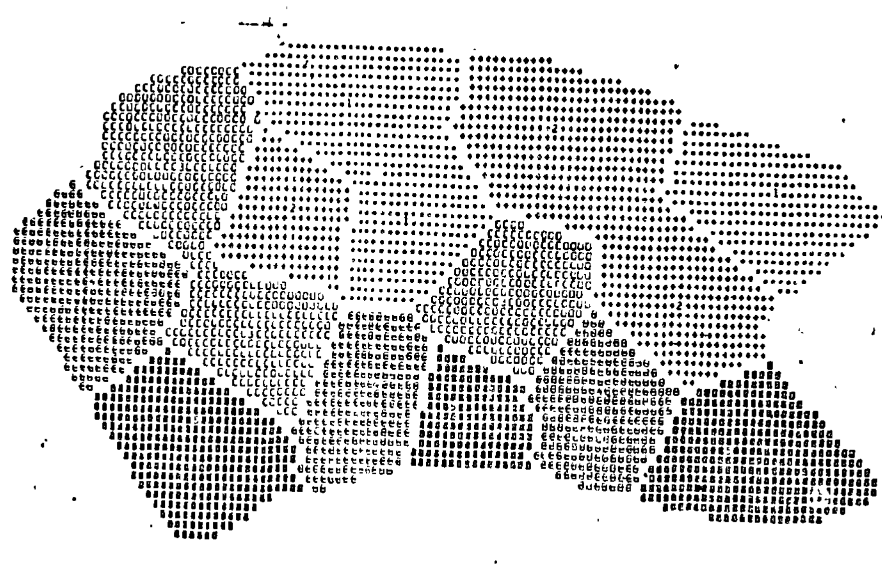


Fig. 3 MAPA DE PROXIMIDAD

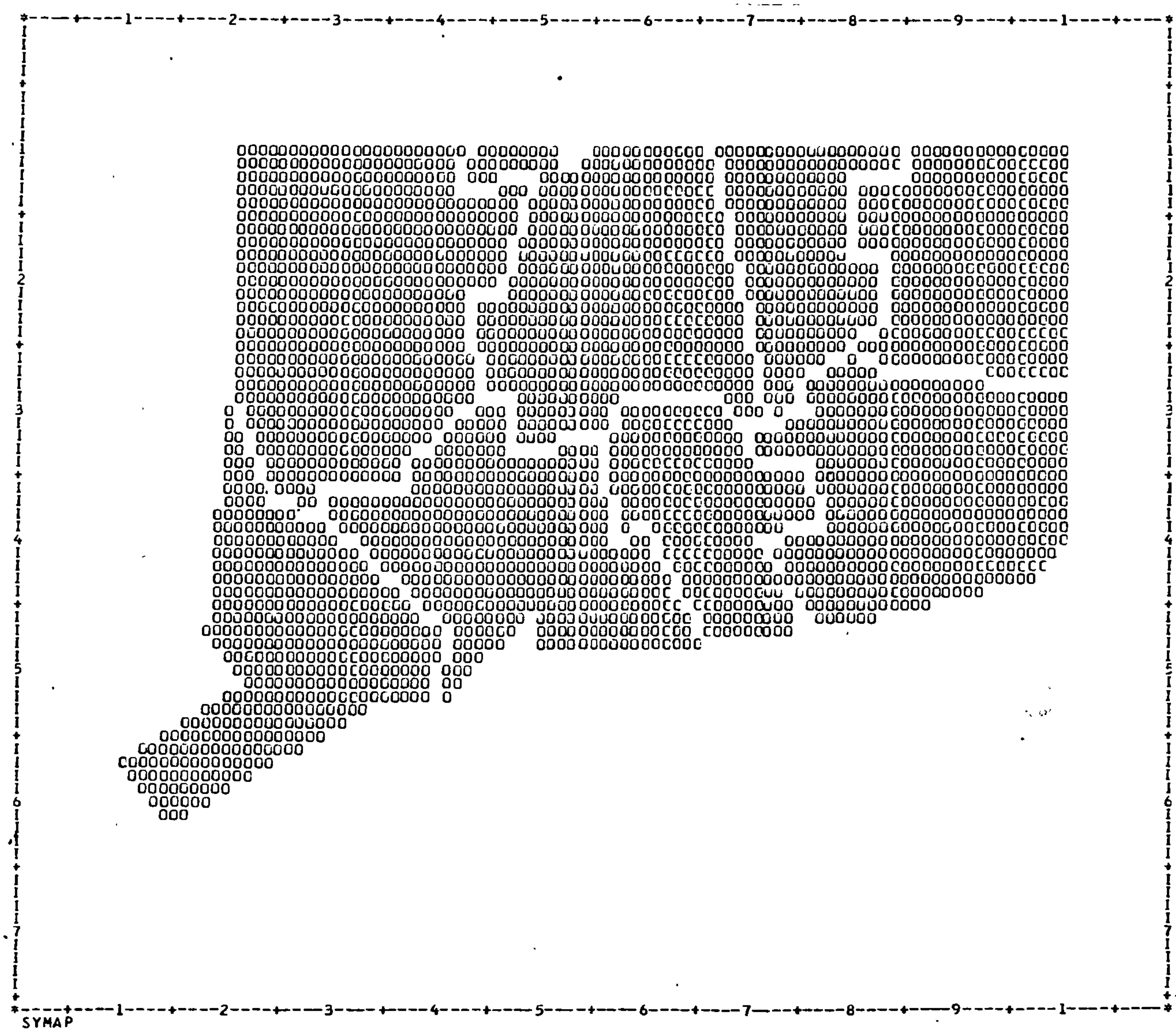


Fig. 4 MAPA BASE

Fig. 5 Mapa Coropleta.

Area metropolitana de la Ciudad de México
División por delegaciones y municipios 1970

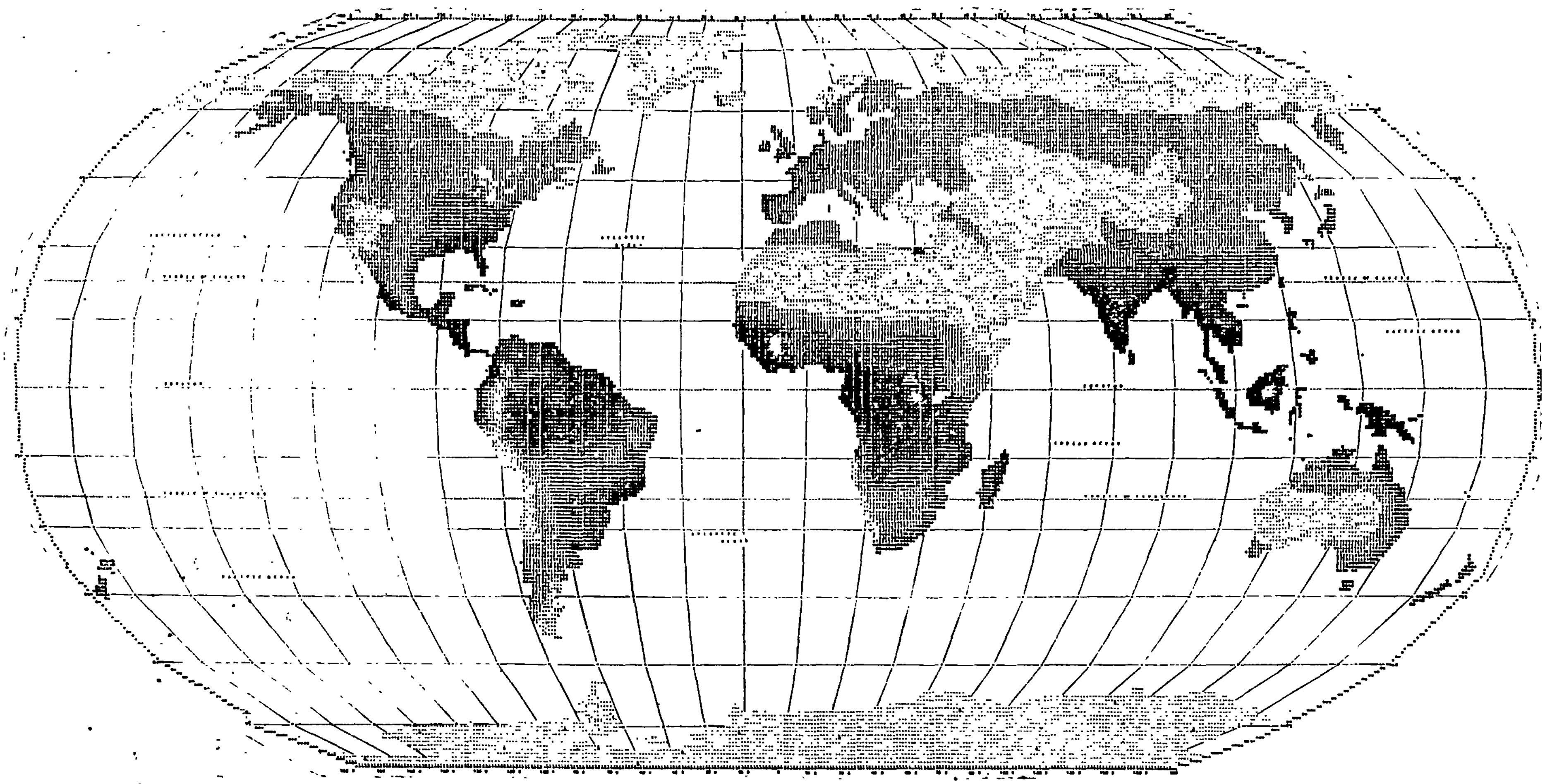


Fig. 6 Indice de productividad agrícola potencial basado en la precipitación y temperatura promedios, tomadas en las estaciones metereológicas existentes. Mapa isopleta.

I.2 PAQUETES DE SYMAP

Cada uno de los siguientes paquetes permite al usuario -- dar al programa un conjunto modular de datos e instruccio nes.

A-CONTORNOS: Especifica los límites de cada una de las - zonas (áreas, puntos ó líneas) en las que está dividida - el área de estudio. Se utiliza en mapas coropletas.

A-DELINEAMIENTO: Especifica el controno exterior del á-- rea de estudio, se utiliza en mapas isopletas y de proxi- midad.

B-PUNTOS DATO: Especifica las localizaciones de los pun-- tos en donde se realizaron las observaciones. Se utiliza en mapas isopletas y de proximidad.

C-LETREROS: Especifica el simbolismo y la localización de cualquier información complementaria que se desea aparez- ca en el mapa, (lagos, ríos, carreteras, orientaciones, - nombres, etc.). La localización relativa de los letreros es ajustada automáticamente al variar la escala y/ó el ta maño del mapa.

D-BARRERAS: Especifica la localización y resistencia de - barreras a la interpolación en vértices determinados. Las barreras pueden ser impermeables y no permitir la interpo lación ó barreras con diferentes grados de permeabilidad- y que permiten la interpolación dependiendo de la resis-- tencia especificada.

E-VALORES: Define los valores que se van a mapear. Estos deben estar en el mismo orden que sus puntos-dato ó zonas correspondientes. Es necesario para todos los mapas ex-- cepto los mapas base.

E-INDICE: Da un nuevo orden a los valores cuando estos no están dados en la misma secuencia que sus puntos-dato ó - zonas correspondientes.

F-MAPA: Se requiere en todos los mapas. Su inclusión ha- ce que se produzca un mapa con la información contenida - en paquetes precedentes. Especifica el título y define - prácticamente todos los aspectos que debe tener el mapa.- También se utiliza para ejercer cierto control sobre el - proceso de interpolación.

II. PREPARACION DE PAQUETES.

II.1 A-CONTORNOS

Este paquete sirve para especificar el contorno de cada una de las zonas que pertenecen al área de estudio en los mapas coropletas. Los contornos deben ordenarse secuencialmente de acuerdo a su número correspondiente en el mapa fuente. En caso necesario, una zona puede estar compuesta de varios contornos (islas). Se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-11	A-CONTORNOS
23	X Si no desea la impresión tabular de las coordenadas.
Col. 29-30	PR Si se desea que los centroides de las zonas se impriman en el mapa.
Col. 31-32	8. Si las coordenadas verticales fueron medidas en octavos de pulgada.
Col. 41-43	10. Si las coordenadas horizontales fueron medidas en décimos de pulgada.
Col. 63	X Si los datos están en cinta.

Ultima tarjeta:

Col. 1-5 99999 para terminar el paquete.

Tarjetas intermedias:

Perforar las coordenadas de los vértices de los contornos zonales. Para cada contorno deben perforarse las siguientes tarjetas:

1a. Tarjeta:

Col. 1-5	Número de la zona, perforado como número entero y justificado a la derecha.
Col. 10	A, L ó P para indicar que el contorno de la zona vá a representarse como un área, una línea ó un punto.
Col. 11-20	Coordenada vertical del primer vértice, perforada como número decimal.
Col. 21-30	Coordenada horizontal del primer vértice, perforada como número decimal.

Las coordenadas de los vértices restantes deben ser perforadas en los campos 11-20 y 21-30 de manera similar a la tarjeta anterior, utilizando una tarjeta por cada vértice. En cada uno de los contornos zonales debe repetirse el primer vértice para indicar que ha terminado un contorno y poder pasar a la siguiente ó cerrar el paquete.

II.2 A-DELINEAMIENTO

Este paquete sirve para especificar el contorno del área de estudio en los mapas isopletas y de proximidad. Se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-15	A-DELINEAMIENTO.
Col. 23	X Si se desea suprimir la impresión de las coordenadas.
Col. 31-32	8. Si las coordenadas en sentido vertical fueron medidas en octavos de pulgada.
Col. 41-43	10. Si las coordenadas en sentido horizontal fueron medidas en décimos de pulgada.
Col. 63	X Si los datos estan en cinta.

Ultima tarjeta:

Col. 1-5	99999 para terminar el paquete.

Tarjetas intermedias:

Las tarjetas que se colocan entre la primera y la última registran las coordenadas de los vértices del contorno, utilizando una tarjeta por vértice. El primer vértice que se prefora es el que está por arriba de todos y más a la izquierda. Se procede en seguida en sentido de las manecillas del reloj hasta incluir de nuevo el vértice inicial. Muchas veces el área de estudio no está definida en un solo contorno, por ejemplo el mapa de Italia requerirá tres contornos, uno para la Península principal, otro para Sicilia y otro para Cerdeña. En tal caso deben usarse uno o más contornos, presentados en cualquier secuencia. Ningún delineamiento debe tener menos de tres vértices ni más de 100, si requiriese más debe ser subdividido.

Las tarjetas con las coordenadas de cada vértice deben perforarse de la siguiente manera:

Col. 11-20	Coordenada vertical, perforada como número decimal.
Col. 21-30	Coordenada horizontal, perforada como número decimal.
Col. 73-80	Para uso del usuario. Se sugiere que los vértices sean numerados ó tengan alguna referencia.

II.3 B-PUNTOS DATO

Este paquete sirve para especificar las coordenadas de los puntos donde se realizaron las observaciones en los mapas isopletas y de proximidad.

Se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-13	B-PUNTOS DATO
Col. 23	X Si se desea suprimir la impresión de las coordenadas.
Col. 31-32	8. Si las coordenadas en sentido vertical fueran medidas en octavos de pulgada
Col. 41-43	10. Si las coordenadas en sentido horizontal fueron medidas en décimos de pulgada.
Col. 63	X Si los datos están en cinta

Ultima tarjeta:

Col. 1-5	99999 para indicar que termina el paquete

Tarjetas intermedias:

Llevan las coordenadas de los puntos, cada uno en una tarjeta por separado. No deben utilizarse más de 1000 puntos en un mapa. Cada tarjeta debe perforarse con el formato siguiente:

Col. 11-20	Coordenada en sentido vertical, perforada como número decimal.
Col. 21-30	Coordenada en sentido horizontal, perforada como número decimal.

Las tarjetas deben ser ordenadas en orden creciente de acuerdo al número de referencia de cada punto dato en el mapa fuente.

II.4 C-LETREROS

Este paquete sirve para especificar la posición y contenido de letreros, numeraciones y simbolismos dentro del rectángulo en el que está dibujado el mapa, así como la información que deben contener por igual todos los mapas de una serie, (titulo general, nombre del área de estudio, orientación, escala, zonas de importancia, ríos, vías de comunicación, etc.). El programa permite cuatro tipos de letreros:

Area:	Se determina un área o región la cual es llenada con un símbolo que especifica el usuario. El contorno es determinado con el método del paquete A-DELINEAMIENTO.
Línea:	Imprime un símbolo especificado por el usuario a lo largo de una línea determinada.
Punto:	Imprime un símbolo especificado por el usuario en una localización determinada.
Hilera de caracteres:	Imprime una hilera de caracteres horizontal ó verticalmente con relación a un punto determinado.

Los cuatro tipos de letreros vienen ilustrados en la figura. El paquete se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-10	C-LETREROS
Col. 23	X Si se desea suprimir la impresión de las coordenadas.
Col. 31-32	8. Si las coordenadas verticales fueron medidas en octavos de pulgada.
Col. 41-43	10. Si las coordenadas horizontales fueron medidas en décimos de pulgada.

Tarjetas siguientes:	Letreros deseados.

Ultima tarjeta:

Col. 1-5	99999 para terminar el paquete

Letreros de área

1a. tarjeta:

Col. 6-9	Caracteres que se van a sobre-imprimir para formar el símbolo deseado.
Col. 10	A para indicar que es letrero de área.
Col. 11-20	Coordenada vertical del primer vértice del contorno, perforada como número decimal.
Col. 21-30	Coordenada horizontal del primer vértice del contorno, perforada como número decimal.

Tarjetas siguientes:	Coordenadas de los vértices del contorno. En la última tarjeta se repiten las coordenadas del primer punto para indicar que ha terminado el contorno. Debe tenerse una tarjeta por cada vértice. El órden en que se dan los vértices es de acuerdo a las manecillas del reloj.
Col. 11-20	Coordenada vertical perforada como número decimal
Col. 21-30	Coordenada horizontal, perforada como número decimal.

Letreros de línea:

1a. tarjeta:

Col. 6-9	Caracteres que se van a sobre-imprimir para formar el símbolo deseado. (Algunos pueden dejarse en blanco).
Col. 10	L para indicar que es letrero de línea.
Col. 11-20	Coordenada vertical del primer vértice de la línea, perforada como número decimal.
Col. 21-30	Coordenada horizontal del primer vértice de la línea, perforada como número decimal.

Tarjetas siguientes:	Coordenadas de los vértices de la línea, dados secuencialmente. Debe tenerse una tarjeta para cada vértice.
Col. 11-20	Coordenada vertical, perforada como número decimal.
Col. 21-30	Coordenada horizontal, perforada como número decimal.

Letreros de punto:
(1 tarjeta)

Col. 6-9		Caracteres que se van a sobre-imprimir para formar el símbolo deseado; algunos pueden ser blancos.
Col. 10	P	Para indicar que es letrero de punto.
Col. 11-20		Coordenada vertical del punto donde se desea localizar el símbolo, perforada como número decimal.
Col. 21-30		Coordenada horizontal del punto donde se desea localizar el símbolo, perforada como número decimal.
Col. 31-40		Si se desea desplazar verticalmente el símbolo con respecto al punto dado, deber perforarse el número de renglones que se desea recorrer, dado como número decimal. Si el desplazamiento es hacia arriba del punto, el número de renglones debe ir precedido por el signo menos.
Col. 41-50		Desplazamiento horizontal, expresado como el número de columnas recorrido, perforado como número decimal. Si el desplazamiento es a la izquierda, debe ir precedido por el signo menos.

Si las columnas 31-50 son dejadas en blanco el símbolo aparecerá exactamente en el punto especificado.

Letreros de hilera de caracteres:

1a. tarjeta:

Col. 1	En blanco si la hilera de caracteres es horizontal;- si la hilera de caracteres es vertical.
Col. 4-5	Número de caracteres de que se compone el letrero (incluyendo los blancos.). Este número no debe exceder de 50 y debe perforarse como número entero y justificado a la derecha.
Col. 10	P Para indicar que la hilera de caracteres está localizada con respecto a un punto de referencia.
Col. 11-20	Coordenada vertical del punto de referencia, perforada como número decimal.
Col. 21-30	Coordenada horizontal del punto de referencia, perforada como número decimal
Col. 31-40	Desplazamiento del letrero con respecto al punto de referencia. Número de renglones ó columnas que se desean recorrer, perforado como número decimal. Si el desplazamiento es hacia arriba ó hacia la izquierda, debe ir precedido por el signo menos. Si estas columnas son dejadas en blanco, el letrero, comenzará a imprimirse exactamente en el punto de referencia.

2a. tarjeta:

Col. 1-50	Perforar el letrero deseado. Debe ocuparse solamente el mismo número de columnas que fue especificado en la 1a. tarjeta en las columnas 4-5.

II.5 D-BARRERAS

Este paquete sirve para especificar la localización y las características de barreras (ríos, carreteras, límites políticos, etc.) que modifican la interpolación entre los puntos dato en los mapas isopletas y de proximidad.

Se tienen dos tipos de barreras:

- Impermeables: no permiten interpolación.
- Permeables: restringen la interpolación pero no la detienen, dependiendo de la resistencia que tengan a lo largo de su camino. Pueden ser una línea divisoria ó de relieves (cada vértice con una resistencia diferente).

1. Efecto de las barreras sobre la interpolación.

El paquete D-BARRERAS hace que en el proceso de interpolación se determine el número de barreras que atraviesa la línea (vector de búsqueda) que conecta un punto dato con una localización de impresión (carácter en el mapa). Si se atraviesa una barrera, se calcula el punto. Esta resistencia es sumada a la distancia real entre el punto dato y la localización de impresión donde se desea interpolar.

Si la distancia total está dentro del radio de búsqueda, el algoritmo de interpolación utiliza la distancia incrementada para calcular la proporción de los valores asociados a los puntos dato que recibe la localización de impresión para tomar su valor. Esta proporción es inversamente proporcional al cuadrado de la distancia entre los puntos dato y la localización de impresión.

En el caso de que la distancia total exceda el radio de búsqueda (como pasa siempre en las barreras impermeables, ya que la distancia añadida es muy grande) no se lleva a cabo la interpolación.

El efecto de una barrera es añadir una cierta distancia, igual a la resistencia local de la barrera a cada vector de búsqueda que la atraviese.

2. Especificación de las resistencias en las barreras.

En las barreras permeables, las resistencias deben especificarse en las mismas unidades en que se midieron las coordenadas del mapa base. La resistencia en los vértices terminales debe ser 0.0. La resistencia en cada vértice de una barrera de relieves es el doble de su distancia al primer vértice.

En otras configuraciones se pueden tener ceros en los extremos e ir aumentando la resistencia al acercarse al centro.

En las barreras impermeables, la resistencia en todos los vértices debe ser codificada como -1.0.

Se recomienda usar la opción 37 del paquete F-MAPA que instruye al programa para interpolar en cada una de las localizaciones de impresión y evitar así el procedimiento estándar de interpolar a cada dos localizaciones horizontalmente y a cada tres verticalmente, produciendo así una superficie con menos discontinuidades.

El paquete se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-10	D-BARRERAS
Col. 23	X Si se desea suprimir la impresión de las coordenadas de los vértices de las barreras.
Col. 31-32	8. Si las coordenadas verticales fueron medidas en octavos de pulgada.
Col. 41-43	10. si las coordenadas horizontales fueron medidas en décimos de pulgada.
Col. 73-80	Cualquier información que desee el usuario.

2a. tarjeta:

Col. 11-20	Coordenada vertical del primer vértice de la barrera, perforada como número decimal.
Col. 21-30	Coordenada horizontal del primer vértice de la barrera, perforada como número decimal.
Col. 31-40	Resistencia de la barrera en ese vértice. Para barreras impermeables siempre debe perforarse -1.0; para barreras permeables cualquier número mayor que cero, perforado como número decimal.
Col. 73-80	Cualquier información que desee el usuario.

Tarjetas siguientes: Todos los vértices sucesivos de la barrera deben ser perforados de la misma manera que la tarjeta 2.

Si una barrera tiene un extremo en el mapa y el otro en el borde, el primer vértice debe ser el que este en el mapa.

Para otro tipo de barreras, se debe comenzar con el vértice extremo que esté más arriba y si los dos están a la misma altura, el vértice inicial será el que este situado mas a la izquierda. Los vértices restantes deben ser perforados en órden de aparición. Si la barrera es cerrada debe procederse en el sentido de las manecillas del reloj.

Para indicar la terminación de una barrera se debe repetir el último vértice. Se pueden especificar hasta 20 barreras diferentes pero el número total de vértices no debe ser mayor que 50. Estos no incluyen la repetición de los últimos vértices, pero si aquellos que forman parte de dos o más barreras distintas.

Ultima tarjeta:

Col. 1-5 99999 para indicar que termina el paquete.

II.6 E-VALORES

Este paquete sirve para especificar los valores asociados a cada uno de los puntos dato contenidos en el paquete B-PUNTOS DATO ó a cada una de las zonas contenidas en el paquete A-CONTORNOS. Debe darse una lista de valores, uno para cada punto ó zona, perforando un valor por tarjeta. Los valores deben estar en el mismo orden en que están dados los puntos dato ó las zonas, de otra forma hay que usar el paquete B-INDICE para reordenarlos. Si hay mas valores que zonas ó puntos, los valores sobrantes son ignorados. Si hay mas zonas ó puntos que valores, se asigna un valor de 0.0 a las zonas ó puntos extra.

El paquete se prepara de la manera situiente:

1a. tarjeta:

Col. 1-9	E-VALORES
Col. 23	X Si se desea suprimir la impresión de la lista de valores
Col. 73-80	Para uso del usuario.

2a. tarjeta:

Col. 11-20	Valor asociado al primer punto ó zona. perforado como número decimal.
Col. 73-80	Cualquier información que desee el usuario.

Tarjetas siguientes:	Igual que la 2a. tarjeta. Deben tenerse tantas tarjetas como valores haya en la lista.

Ultima tarjeta:

Col. 1-5	99999 para indicar que termina el paquete.

II.7 E-INDICE

Este paquete sirve para cambiar el órden de referencia en el paquete E-VALORES en el caso de que estos no esten dados en la misma secuencia que los puntos dato en el paquete B-PUNTOS DATO ó las zonas en el paquete A-CONTORNOS.

El paquete se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-8	E-INDICE
Col. 23	X Si se desea suprimir la impresión de los valores
Col. 73-80	Cualquier información que desee el usuario

2a. tarjeta:

Col. 1-5	1 Indicando que se trata de la primera zona ó punto dato, perforado como entero justificado a la derecha.
Col. 6-10	Número de referencia del valor que va a ser asignado al primer punto dato o zona, perforado como número entero justificado a la derecha.

Tarjetas siguientes:	Para cada punto dato o zona a la que se desea asignar un valor con un número de referencia distinto, debe perforarse una tarjeta similar a la segunda. Si las columnas 6-10 son dejadas en blanco, se asigna el número siguiente al especificado en la tarjeta anterior.

Ultima tarjeta:

Col. 1-5	99999 para indicar que termina el paquete.

II.8 F-MAPA

Para cada mapa que se desee producir, es necesario tener un paquete F-MAPA después de los paquetes donde se tiene la información que se desea mapear; especifica la forma precisa del mapa mediante la utilización de opciones del programa que controlan aspectos tales como el tamaño, escala, simbolismo, análisis de frecuencias, interpolación, etc.. No es obligatorio usar estas opciones ya que se tienen valores estándar para cada una, los cuales son utilizados si no se especifica una opción.

El paquete se prepara de la manera siguiente:

1a. tarjeta:

Col. 1-6	F-MAPA
Col. 23	X Si se desea suprimir la impresión del contenido del paquete.

Tarjetas 2-4

Col. 1-72	Perforar la información que se desea aparezca al pie del mapa, pudiéndose dejar en blanco. No obstante deben ser incluidas las tres tarjetas.
Tarjetas siguientes:	Opciones deseadas. (Ver capítulo III).

Ultima tarjeta:

Col. 1-5	99999 para indicar que termina el paquete.

Tarjeta de terminación

Col. 1-6	999999 Debe ir después del último F-MAPA de todos los mapas que se vayan a correr.

Para tener mapas diferentes de la misma información pero variando las opciones, basta con colocar paquetes F-MAPA, en forma sucesiva.

III. OPCIONES CARTOGRAFICAS DEL PAQUETE F-MAPA

Opción 1.- (1 tarjeta)	Dimensiones del mapa
Col. 5	1 Para identificar la opción
Col. 11-20	Dimensión vertical del mapa, dada en pulgadas y como número decimal.
Col. 21-30	Dimensión horizontal del mapa dada en pulgadas y como número decimal.

Si no se incluye esta opción, el programa determina las dimensiones del mapa asignando 13 pulgadas a la dimensión mayor, siempre y cuando no se utilice la opción 13, sola ó en combinación con las opciones 12 y 14. Si la dimensión horizontal excede de 13 pulgadas (ancho del papel de la computadora), el mapa es impreso en secciones que deberán ensamblarse. El tamaño del mapa está limitado a 72 pulgadas en cualquier dirección. Para lograr mapas mayores, debe especificarse la opción 16.

Opción 2. (1 tarjeta)	Ventana del mapa
Col. 5	2 Para identificar la opción.
Col. 11-20 y 21-30	Coordenadas vertical y horizontal del punto superior izquierdo de la ventana del mapa
Col. 31-40 y 41-50	Coordenadas vertical y horizontal del punto inferior derecho de la ventana del mapa

La ventana del mapa es una región rectangular que el usuario desea observar dentro del área de estudio. Esta es especificada mediante los puntos extremos del rectángulo, el cuál deberá tener los lados paralelos a los márgenes del mapa. Si no se especifica esta opción, puede lograrse el mismo efecto mediante la opción 14.

Opción 3. (1 tarjeta)	Número de niveles ó intervalos de clase.
Col. 5	3 Para identificar la opción
Col. 11-20	Número deseado de niveles, desde 1 hasta 10, perforado como número decimal.

Esta opción sirve para especificar el número de niveles ó intervalos de clase en que se desea dividir el rango total de los valores contenidos en los datos, con el propósito de asignar un simbolismo particular a todos los datos contenidos en el mismo intervalo. Si no se especifica esta opción, el pro grama divide el rango total de valores en cinco intervalos iguales. Puede usarse en combinación con las opciones 4, 5 y 6.

Opción 4. (1 tarjeta) — Valor mínimo del rango

Col. 5 — 4 Para identificar la opción
Col. 11-20 — El valor mínimo deseado, perforado como un número decimal.

Todos los valores menores que este aparecerán con un simbolismo de 'L', que puede ser modificado mediante la opción 7. Si no se especifica esta opción, el programa toma como valor mínimo al menor de los valores contenidos en el paquete - - E-VALORES y que no esté inválido por las opciones 18, 19 y/ó 20.

Opción 5. (1 tarjeta) — Valor máximo del rango

Col. 5 — 5 Para identificar la opción
Col. 11-20 — El valor máximo deseado, perforado como un número decimal.

Todos los valores mayores que este aparecerán con un simbolismo especial de 'H', que puede modificarse mediante la opción 7. Si no se especifica esta opción, el programa toma como valor máximo al mayor de los valores contenidos en el paquete E-VALORES y que no está declarado inválido por las opciones 18, 19 y/ó 20.

Opción 6. (1 a 3 tarjetas) — Rangos de los niveles ó intervalos de clase.

Col. 5 — 6 Para identificar la opción.

Especificando solo este número, el programa distribuirá equitativamente los valores de manera que haya aproximadamente la misma frecuencia en cada nivel. Si se desea especificar diferentes rangos para cada uno de los niveles, el tamaño de estos deberá estar indicado en la misma tarjeta de la siguiente manera:

Col. 11-20 — Tamaño del primer intervalo, perforado como un número decimal.
Col. 21-30 — Tamaño del segundo intervalo, perforado como un número decimal.

Así se continúa en campos de 10 columnas hasta la columna 70. En caso de más de 6 niveles, se usa otra tarjeta. Esta opción puede usarse en combinación con las opciones 3, 4 y 5. El tamaño de los intervalos puede especificarse de diferentes maneras:

Ejemplo 1: Si existen 5 niveles y el tamaño de cada nivel es el doble del tamaño del nivel previo, se deben perforar los siguientes números:

Columna	5	11 - 20	21 - 30	31 - 40	41 - 50	51 - 60
	6.	1.	2.	4.	8.	16.

Ejemplo 2: Si se desea dividir los datos en cuatro grupos el menor 10%, el siguiente 25%, el siguiente 35% y 30% el restante, se deben perforar los siguientes números:

Columna	5	11 - 20	21 - 30	31 - 40	41 - 50
	6	10.	25.	35.	30.

Ejemplo 3: Para especificar los siguientes intervalos

0 150 200 271.5 500 750 889 1000

Se deben perforar los números siguientes:

Columna	5	11 - 20	21 - 30	31 - 40	41 - 50	51 - 60	61 - 70
Tarjeta 1	6	150.	50.	71.5	228.5	250.	139.
Tarjeta 2		111.					

Cualquier valor que cae en el límite entre dos intervalos, es asignado al nivel mayor, con la excepción de valores que caigan en el límite del último intervalo.

Opción 7. (5 tarjetas) Simbolismo

En esta opción se especifican los caracteres que se desean imprimir y sobre-imprimir para formar los símbolos representativos de cada uno de los niveles. Cada símbolo puede estar compuesto hasta de cuatro caracteres, perforados en la columna correspondiente al nivel, en estas tarjetas.

Tarjeta 1:

Col. 5	7 Para identificar la opción

Tarjeta 2-5

Col. 1-10	Símbolos para cada uno de los intervalos dados en orden ascendente. Se debe utilizar solo las columnas necesarias, para los niveles especificados.
Col. 11-20	Simbolismo para los puntos señal de los intervalos 1-10 en ese orden. Solo deben usarse las columnas necesarias para el número de intervalos especificado, dejando las demás en blanco.
Col. 21	Símbolo para aquellas áreas del mapa cuyo valor interpolado es menor que el especificado en la opción 4.
Col. 22	Símbolo para las posiciones de los puntos dato cuyos valores son menores que el mínimo establecido en la opción 4.
Col. 23	Símbolo para aquellas áreas del mapa cuyo valor interpolado es mayor que el especificado en la opción 5.
Col. 24	Símbolo para las posiciones de los puntos dato cuyos valores son mayores que el máximo establecido en la opción 5.
Col. 25	Símbolo para el fondo, el cual aparece entre el área de estudio y los bordes del mapa, así como en las zonas declaradas inválidas por las opciones 18, 19 y/ó 20.
Col. 26	Símbolo para las isolíneas y contornos de las zonas
Col. 27	Símbolo para aquellas áreas del mapa donde no hay interpolación, debido al uso de la opción 35 y/ó barreras impermeables.
Col. 28	Símbolo para indicar la presencia de dos o más puntos datos en la misma localización de la impresora.
Col. 29	Símbolo para indicar las posiciones de los puntos dato declarados inválidos en las opciones 18, 19 y/ó 20.

Cuando no se especifica esta opción el programa tiene un conjunto de simbolismos estándar que se muestran a continuación; para 10 niveles se tiene:

```
                        11111111112222222222
Columna       12345678901234567890123456789
Tarjeta 1         7
Tarjeta 2     .'-=+X0000123456789*LLHH  NSM
Tarjeta 3           -XX           . H
Tarjeta 4            · A            H
Tarjeta 5              V            /
```

En caso que se especifiquen menos de 10 niveles los simbolismos son los siguientes:

```
  Número de
   niveles
especificado        Simbolismo estándar
              1 2 3 4 5 6 7 8 9 10
    10        . ' - = + X 0 θ ⊗ ⊗
     9        . ' = + X 0 θ ⊗ ⊗
     8        . ' + X 0 θ ⊗ ⊗
     7        . ' + X θ ⊗ ⊗
     6        . + X 0 θ ⊗
     5        . + 0 θ ⊗
     4        . + 0 ⊗
     3        . 0 ⊗
     2        . ⊗
     1        .           (OXAV=⊗)
```

A continuación se muestran algunos ejemplos de simbolismos no-estándar:

Ejemplo 1: Se tienen 6 niveles especificados y se desea invertir el orden del simbolismo estándar:

```
                                  1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2
Columna     1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
Tarjeta 1           7
Tarjeta 2   0 0 0 X + .           1 2 3 4 5 6
Tarjeta 3   X X -
Tarjeta 4   A
Tarjeta 5   V
```

Ejemplo 2: Se desea suprimir todo el simbolismo, dejando las isolíneas ó los contornos de las zonas en negro y con un fondo de guiones:

Columna	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Tarjeta 1					7																					
Tarjeta 2											1	2	3	4	5										-	0
Tarjeta 3																										X
Tarjeta 4																										A
Tarjeta 5																										V

Ejemplo 3: Se desea usar los símbolos A, B, C, D y E para las clases de datos en un mapa de proximidad.

Columna	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Tarjeta 1					7																					
Tarjeta 2	A	B	C	D	E						A	B	C	D	E											
Tarjeta 3											/	/	/	/	/											
Tarjeta 4																										
Tarjeta 5																										

<u>Opción 8.</u> <u>Eliminación de las líneas de los contornos ó las isolíneas</u>
(1 tarjeta)

<u>Col. 5</u> 8 Para identificar la opción

Estas líneas generalmente aparecen como blancas, a menos que sea especificado otro simbolismo para ellas en la opción 7. Si no se especifica esta opción, estas líneas aparecerán en el mapa.

<u>Opción 9.</u> <u>Eliminación del histograma</u>
(1 tarjeta)

<u>Col. 5</u> 9 Para identificar la opción

Si no se especifica esta opción, aparecerá impreso debajo del mapa un histograma que muestra la distribución de frecuencias de los datos en los intervalos de clase.

<u>Opción 10.</u> <u>Texto explicativo</u>
(2 a 32 tarjetas)

<u>Tarjeta 1:</u>

<u>Col. 4 y 5</u> 10 Para identificar la opción

<u>Tarjetas 2-31</u>

Col. 1-72 El texto explicativo deseado, pueden incluirse hasta 30 tarjetas. El texto apare

cerá debajo del mapa.

Ultima tarjeta

Col. 1-4 — 9999 para terminar el texto.

Opción 11 (1 tarjeta) — Impresión del valor de los datos en sus localizaciones correspondientes

Col. 4-5 — 11 Para identificar la opción.

Si no se especifica esta opción aparecerá impreso el número del nivel ó intervalo en el que se encuentre el valor del punto dato.

Opción 12. (1 tarjeta) — Repetición múltiple de opciones.

Col. 4-5 — 12 Para identificar la opción.

Esta opción permite repetir todas las opciones usadas en el mapa previo.

Opción 13 (1 tarjeta) — Escala del mapa

Col. 4-5 — 13 Para identificar la opción
Col. 11-20 — La escala deseada, dada como número decimal

Si el mapa está medido en cualquier sistema de medida, la escala especifica el número de pulgadas que representarán a cada unidad de medida del mapa base. Por ejemplo, 2.0 producirá un mapa en donde cada unidad del mapa base está representada por 2 pulgadas.

Si el mapa está medido en octavos y décimos de pulgada, la escala especifica que tantas veces se desea aumentar o disminuir el mapa. Por ejemplo, 1.0 producirá un mapa a la misma escala del mapa base, 2.0 uno al doble, etc.. Si no se especifica esta opción, la escala quedará determinada por las opciones 1, 2 y/ó 4.

Opción 14 (1 tarjeta) — Márgenes del mapa

Col. 4-5 — 14 Para identificar la opción
Col. 11-20 — Número de pulgadas deseadas como márgen a partir del límite superior del mapa
Col. 21-30 — Número de pulgadas deseadas como márgen a partir del límite izquierdo del mapa

Col. 31-40 — Número de pulgadas deseadas como márgen a partir del límite inferior del mapa

Col. 41-50 — Número de pulgadas deseadas como márgen a partir del límite derecho del mapa

Todos los márgenes deben perforarse como número decimales

Se pueden especificar valores positivos y negativos para los márgenes. Los valores positivos añaden área a la ventana del mapa, mientras que los negativos le restan. Los márgenes pueden ser sombreados mediante la opción 7.

Opción 15. (1 tarjeta) — Número de caracteres por pulgada

Col. 4-5 — 15 Para identificar la opción

Col. 11-20 — Número de renglones por pulgada en los que la computadora ha sido ajustada para imprimir, dado como número decimal.

Col. 21-30 — Número de columnas por pulgada

En muchas computadoras la impresora de línea ha sido ajustada para imprimir 6 renglones y 10 columnas de caracteres por pulgada. SYMAP supone que los resultados estarán dados en 8 renglones y 10 columnas de caracteres por pulgada y así es como produce mejores resultados. Si la impresora no está ajustada de esta manera, se deberá usar esta opción para evitar deformaciones en el mapa.

Opción 16. (1 tarjeta) — Mapas grandes

Col. 4-5 — 16 Para identificar la opción.

Esta opción permite al usuario producir mapas mayores de 72 pulgadas (183 cm) en cualquier sentido. Si el mapa pasa de esta medida y esta opción no es usada, el programa reducirá la dimensión mayor a 13 pulgadas.

Opción 17. (1 tarjeta) — Eliminación de los resultados tabulares

Col. 4-5 — 17 Para identificar la opción.

Mediante esta opción se eliminan los resultados tabulares que aparecen antes del mapa y que contienen información sobre la localización, valor y nivel asignado a cada zona ó punto dato.

Opción 18. (1 tarjeta) — Invalidación de valores faltantes

Col. 4-5 18 Para identificar la opción

Con esta opción se hacen inválidos los valores de -0.0 y en blanco. El símbolo 'M' aparecerá en su localización. Si no se especifica, todos los valores serán considerados válidos.

Opción 19. Valor máximo para datos válidos
(1 tarjeta)

Col. 4-5 19 Para identificar la opción
Col. 11-20 El valor que se declara inválido, perforado como número decimal.

Este y todos los valores mayores serán declarados inválidos.

Puede usarse en combinación con la opción 20.

Opción 20 Valor mínimo para datos válidos
(1 tarjeta)

Col. 4-5 20 Para identificar la opción
Col. 11-20 El valor que se declara inválido, perforado como número decimal.

Este y todos los valores menores quedarán declarados inválidos.

Opción 21 Grabación del mapa en cinta
(1 tarjeta)

Col. 4-5 21 Para identificar la opción.
Col. 19-20 1.0 Para obtener un listado de estos valores

Al producirse un mapa, el programa calcula un valor para cada localización impresa en el mapa (en el caso de mapas isopletas y de proximidad lo hace mediante interpolación). Esta opción almacena estos valores en cinta para su utilización posterior por otros programas (SYMVU, por ejemplo).

Opción 22 Contornos contínuos
(1 tarjeta)

Col. 4-5 22 Para identificar la opción.

Esta opción asegura la aparición de líneas de contorno que podrían ser suprimidas para permitir la representación de simbolismo descriptivo cuando el espacio entre puntos dato ó zonas es inadecuado para representar simbolismo y líneas de contorno a la vez.

Opción 23. (1 tarjeta) — Eliminación del simbolismo de punto inválido.

Col. 4-5 — 23 Para identificar la opción.

Elimina el símbolo que aparece en cualquier punto dato cuyo valor asociado está declarado inválido por las opciones 18, 19 y/ó 20.

Opción 24. (1 tarjeta) — Eliminación de interpretación numérica

Col. 4-5 — 24 Para identificar la opción

Después del mapa y el texto explicativo, el programa imprime información sobre los valores extremos, valores inválidos, los rangos de los valores de los intervalos de clase y el porcentaje que representan en el rango total de valores. Si no se desea esta información, debe especificarse esta opción.

Opción 25. (1 tarjeta) — Eliminación de los símbolos de los puntos dato.

Col. 4-5 — 25 Para identificar la opción.

Elimina la aparición del simbolismo de los puntos dato, poniendo en su lugar el simbolismo del valor interpolado en esa localización.

Opción 26. (1 tarjeta) — Alineamiento de sobre-impresión

Col. 4-5 — 26 Para identificar la opción.

La sobre-impresión es realizada con sistemas diferentes en diversas computadoras, que no siempre coinciden con el de SYMAP. Esta opción sustituye este método en el caso de que el alineamiento sea incorrecto.

Opción 27. (1 tarjeta) — Tipo de mapa

Col. 4-5 — 27 Para identificar la opción.

Esta opción permite identificar al programa el mapa como isopleta, cuando se incluyen datos de isolíneas y de conformación en la misma corrida.

Opción 31. (1 tarjeta) — Extrapolación fraccional relativa (Necesaria para mapas de proximidad).

Col. 4-5 — 31 Para identificar la opción.
Col. 11-20 — Límite deseado para la extrapolación, expresado como una fracción del rango total de valores y perforado como número decimal.

El programa extrapola esta fracción del rango total de valores arriba ó abajo de un extremo local. Si no se especifica esta opción, el programa supone una extrapolación fraccional de 0.1. Para mapas de proximidad, dejar en blanco las columnas 11-20.

Opción 31. (1 tarjeta) — Mínimo absoluto en la extrapolación

Col. 4-5 — 32 Para identificar la opción.
Col. 11-20 — Valor deseado, perforado como número decimal

Esta opción fija un valor máximo arriba del cual la computadora no extrapolará. Se recomienda su uso cuando el usuario conoce un máximo lógico para sus datos (110.0 para datos porcentuales, por ejemplo). Si no se especifica, el programa tendrá un máximo de extrapolación igual al valor máximo de los datos válidos.

Opción 34. (1 tarjeta) — Radio inicial de búsqueda

Col. 4-5 — 34 Para identificar la opción
Col. 11-20 — Radio inicial de búsqueda deseado, perforado como un número decimal.

El radio de búsqueda es la distancia sobre la cual el programa busca puntos dato para usarlos como base en la interpolación. Esta opción preserva el radio inicial de búsqueda de un mapa previo después de alterar el número ó localizaciones de los puntos dato dentro del área de estudio. Si no se especifica, el radio de búsqueda está basado en el número y dispersión de los puntos dato. En promedio se utilizan 7 puntos.

Opción 35. (1 tarjeta) — Radio máximo de búsqueda

Col. 4-5 — 35 Para identificar la opción.
Col. 11-20 — Radio máximo de búsqueda perforado como un número decimal.

La computadora buscará, si es necesario, puntos dato con los cuáles interpolar a una distancia no mayor de este radio. Este no deberá ser menor que el inicial.

Opción 36. (1 tarjeta) — Número de puntos dato para la interpolación (necesaria para mapas de proximidad).

Col. 4-5 — 36 Para identificar la opción
Col. 11-20 — Número deseado de puntos (no más de 10), perforado como número decimal.

Si no se especifica esta opción, se tiene un mínimo de 4 y un máximo de 10, teniendo como promedio 7 puntos para la interpolación. Para mapas de proximidad dejar en blanco las columnas 11-20.

Opción 37. (1 tarjeta) — Independencia (necesaria para mapas de proximidad)

Col. 4-5 — 37 Para identificar la opción.

Esta opción previene el suavizamiento de las líneas creadas por las barreras ó mediante la utilización de las opciones 35 y 36. Si no se especifica esta opción, el programa calcula valores a cada 2 caracteres en sentido vertical y a cada 3 en sentido horizontal.

IV. EL ALGORITMO DE INTERPOLACION DE SYMAP.

1. Introducción.

Su objetivo es crear una superficie que cumpla ciertas propiedades para poder representar espacialmente un fenómeno a partir de la información proporcionada, en un cierto número de puntos dato. Puede aplicarse a diferentes campos tales como demografía, meteorología, planeación urbana, contaminación ambiental, etc.

El programa toma las coordenadas de los puntos dato y sus valores asociados para construir una superficie diferenciable en forma continua, que pasa por los puntos dato y representa las tendencias que estos muestran.

El método consiste en obtener para cada localización de impresión en el mapa, un promedio ponderado de las pendientes y los valores de los puntos dato cercanos, calculado mediante un modelo de tipo gravitacional.

El valor para cada localización se estima a partir de la fórmula:

$$Z_p = \frac{\sum_i w_i \, Z'_i}{\sum_i x_i} \qquad (1)$$

donde:

w_i = la ponderación del punto dato i

Z'_i = el valor en el punto dato, modificado por la pendiente en i y su desplazamiento con respecto al punto P.

2. Modelo básico

De acuerdo al modelo básico, el valor en el punto P debe ser el promedio ponderado de los valores en los puntos-dato 1, 2, ..., n, considerando la ponderación como el inverso de la distancia al cuadrado.

Sea: $\overline{P1}$ = distancia del punto P al punto dato 1

$\overline{P2}$ = distancia del punto P al punto dato 2

Z1 = valor en el punto dato 1

Z2 = valor en el punto dato 2

Zp = valor que se va a calcular para el punto P

$$Z_p = \frac{\frac{1}{(\overline{P1})^2} Z_1 + \frac{1}{(\overline{P2})^2} Z_2 + \frac{1}{(\overline{P3})^2} Z_3 + \ldots + \frac{1}{(\overline{Pn})^2} Z_n}{\frac{1}{(\overline{P1})^2} + \frac{1}{(\overline{P2})^2} + \frac{1}{(\overline{P3})^2} + \ldots + \frac{1}{(\overline{Pn})^2}}$$

O bien:

$$Z_p = \frac{\sum_{i=1}^{n} \frac{1}{(\overline{P_i})^2} Z_i}{\sum_{i=1}^{n} \frac{1}{(\overline{P_i})^2}}$$

Si el punto P está muy cerca del punto 1, por ejemplo, entonces $\overline{P1}$ es pequeña comparada con $\overline{P2}$ y $\overline{P3}$, etc., por lo tanto el peso $\frac{1}{(\overline{P1})^2}$ es grande comparado con $\frac{1}{(\overline{P2})^2}$ y $\frac{1}{(\overline{P3})^2}$ etc.

3. Modificaciones.

Se deben hacer las siguientes modificaciones al método para hacerlo más eficiente:

a. Radio de búsqueda

Por razones prácticas, tiene que limitarse el número de puntos-dato que se consideren para interpolar el valor en una localización dada. Tomando en cuenta el número de puntos dato y el área sobre la que se extienden, el algoritmo determina un radio inicial de búsqueda R, en tal forma que un círculo con ese radio generalmente tendrá el número promedio de puntos en los que se basará la interpolación.

Para cualquier localización P, se eligen aquellos puntos dato cuya distancia efectivo a P sea menor que R. Si dentro del círculo hay mas puntos que el número máximo especificado, se contrae el radio de búsqueda hasta que queden solamente el número de puntos dato permitidos. Si dentro del radio inicial quedan menos puntos que el número mínimo permitido, el radio es expandido hasta que se encuentren todos los puntos requeridos, se llegue al máximo radio de búsqueda especificado por el usuario ó hasta que sean usados todos los puntos que no estén bloqueados por una barrera impermeable.

Las ponderaciones consistentes en la inversa de la distancia al cuadrado, se usan para puntos dato cercanos a P. Cuando la distancia a un punto-dato se aproxima al radio final de búsqueda R', la ponderación en ese punto tiende a cero.

b. Dirección.

Para que las localizaciones relativas de los puntos dato entren en el cálculo, debe encontrarse el "aislamiento direccional" del punto dato i mediante la fórmula:

$$Q_i = \frac{1}{\overline{P1}}\left[1-\cos(i\ P1)\right] + \frac{1}{\overline{P2}}\left[1-\cos(i\ P2)\right] + \ldots + \left[1-\cos(i\ Pn)\right]$$

Si los otros puntos dato 1, 2, 3, ... están en la misma dirección que i con respecto a P, entonces los ángulos i P1, i P2, i P3, son pequeños; las cantidades 1-cos(i P1) son también pequeñas y la suma Q_i es cercana a cero. Dado que i no es el único punto en una dirección particular, se le da una ponderación reducida definida de la siguiente manera:

Si $\overline{p_j}$ es la distancia de P al punto dato j, la distancia ponderada es

$$S_j = \begin{cases} \left(\frac{1}{\overline{P_j}}\right) & \text{para } 0 < \overline{P_j} \leq R'/3 \\ f(\overline{P_j}) & \text{para } R'/3 < \overline{P_j} \leq R' \end{cases}$$

Sea $H = \sum_j S_j$ y sea

$$T_i = \sum_{j \neq i} S_j \times \left[1-\cos(i\ P_j)\right], \text{ donde}$$

los puntos j, son puntos datos vecinos dentro del radio de búsqueda R'. La ponderación total en el punto dato i para la localización P es:

$$W_i = (S_i)^2 \times (H+T_i)$$

Entre mayor sea el aislamiento direccional de un punto dato y menor su distancia al punto considerado mayor es su ponderación y por tanto su influencia en la determinación del valor interpolado.

c. Pendientes.

Para evitar que la superficie presente niveles en los puntos dato, se calcula un gradiente bi-dimensional (pendiente) $\frac{\partial Z}{\partial X}\hat{i}\ \frac{\partial Z}{\partial y}\hat{j}$ en cada punto dato, tomando un promedio ponderado de las pendientes de varios planos secantes. Cada plano contiene al punto P y a uno de sus puntos dato vecinos i; el plano es tan horizontal como sea posible; la línea que va de P al punto dato debe ser la línea de pendient e más pronunciada en ese plano. El valor de la superficie en el punto P se aproxima a:

$$Z_i = Z_i + \Delta Z_i$$

donde

$$\Delta Z_i = \left\{ \left.\frac{\partial Z}{\partial x}\right|_i \Delta_x + \left.\frac{\partial Z}{\partial y}\right|_y \Delta y \right\} x\ K_i$$

Δ_x y Δ_y son las distancias x y y tomadas a partir de i.

El factor $K_i = \dfrac{a}{a + \overline{P_i}}$ se introduce en tal forma, que el efecto de considerar la perndiente es pequeño a grandes distancias. El parámetro a se escoge en tal forma que aunque i fuera el punto dato con la pendiente mas pronunciada, ΔZ sea menor en magnitud que una fracción especificada del rango total de Z.

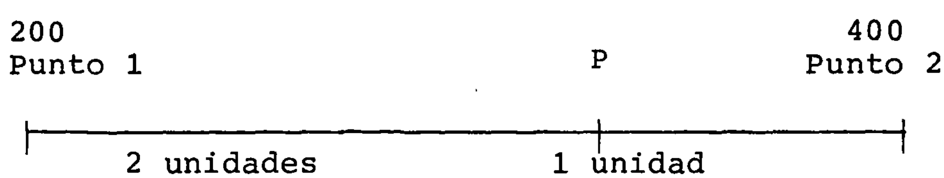
Si el valor en un punto dato i es mayor que los valores de los puntos dato circundantes, la pendiente en ese punto no sera cero a menos que las pendientes ponderadas de los planos secantes se cancelen a si mismas. Así, la superficie continuará creciendo en alguna dirección y por tanto, los máximos y mínimos relativos no ocurren con frecuencia en los puntos dato.

d. Barreras.

El algoritmo se modica si se incluyen barreras a la interpolación. Las distancias usadas en los cálculos son "distancias efectivas". El cuadrado de la distancia efectiva es igual al cuadrado de la distancia real mas la suma de los cuadrados de las resistencias de todas las barreras atravesadas un número non de veces. A las barreras impermeables se les asigna una resistencia tan grande que la distancia efectiva resultante hace que no sea considerado un punto como punto dato cercado, y por tanto se elimine del proceso de interpolación.

4. Ejemplo de interpolación usado el algoritmo de SYMAP:

Densidad de población
punto 1 = 200 hab/km^2
punto 2 = 400 hab/km^2



i	S_i	H	T_i	W_i
1	.5	1.5	2.0	.875
2	1.0	1.5	1.0	2.5

i	Z_i	$\frac{\partial Z}{\partial x}\Big\vert_i$	k-i	$\Delta \underline{\underline{Z}}_i$	$\underline{\underline{Z}}_i$
1	200	66.7	.261	17.4	217.4
2	400	66.7	-.231	-15.4	324.6

$$\Sigma w_i = 3.375$$

El valor en el punto P, es:

$$Z_p = \frac{Z_1 W_1 + Z_2 W_2}{\Sigma w_i}$$

sustituyendo:

$$Z_p = \frac{(217.4)\ (8.75) + (384.6)\ (2.5)}{3.375} = 341$$

Densidad de población en el punto P = 341 hab/km^2

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

MODELOS DIGITALES DE TERRENO

APENDICE BIBLIOGRAFICO

JULIO, 1978.

Palacio de Minería Calle de Tacuba 5, primer piso. México 1, D. F.

TOWARD CREATION OF A DIGITAL TERRAIN DATA BASE

Arthur A. Noma
DEFENSE MAPPING AGENCY
TOPOGRAPHIC CENTER
Washington, D. C. 20315

Presented to
The Annual Convention of the
AMERICAN CONGRESS ON SURVEYING AND MAPPING
and the
AMERICAN SOCIETY OF PHOTOGRAMMETRY

St. Louis, Missouri
10–15 March 1974

TOWARD CREATION OF A DIGITAL TERRAIN DATA BASE*

Arthur A. Noma
Defense Mapping Agency Topographic Center
Washington, D.C.

ABSTRACT

A number of digital terrain data collection systems have come into operation at the Defense Mapping Agency Topographic Center during the past decade; the oldest being the *D*igital *T*opographic *D*ata *C*ollection *S*ystem (DTDCS), most current being the *S*emi-*A*utomated *Cart*ographic *S*ystem (SACARTS). This paper describes the digital contents of terrain data collected by these systems and its availability for computer applications. It also discusses the use of the collected digital terrain data for establishing a digital data base for areas of Department of Defense interest. The paper concludes with some recent applications of the digital terrain data.

* Presented to the Annual Convention of the American Congress on Surveying and Mapping–American Society of Photogrammetry (ACSM ASP), St. Louis, Missouri, 10–15 March 1974.

TOWARD CREATION OF A DIGITAL TERRAIN DATA BASE

Arthur A. Noma

INTRODUCTION

The effort toward exploiting digitized terrain data for mapping and charting began in the late 1950's, only a few years after the advent of the first commercial electronic computers. At the Defense Mapping Agency Topographic Center (DMATC), then the Army Map Service (AMS), the effort in utilizing the digital technology for cartographic work began shortly after the installation of the first commercially available computer, the UNIVAC I,* in 1952. Three systems began in the era and their successors continue in operation today at DMATC. These systems are:

- The Integrated Mapping System.
- The Automatic Model Carving System.
- The Symbols and Names Placement System.

These systems in one form or another utilize digital media for collecting, processing, and controlling the graphic product output. The systems diagram, Figure 1, from the publication entitled "Let's Go Over the Hill – Potential Benefits of Profile Scanning the Stereo-Model" by Messrs. Spooner, Dossi, and Misulia in 1957, [1] shows the creation of a byproduct labeled "Topo Tape Library."

During the early and middle sixties, techniques and ideas pioneered by these systems were refined and incorporated into a number of cartographic systems. The activities of the late sixties showed the rapid swing in mapmaking toward exploitation of digital techniques. Today, perhaps more importantly, cartographic technology is on the threshold of making available map information in digital form for more comprehensive planning. The forecast by William C. Aumen in the midsixties of a "A New Map – The Numerical Map" [2] may have arrived. This paper will describe three systems now operational at DMATC for the collection, processing, and storage of digitized terrain and other topographic data. These systems are:

- The Digital Topographic Data Collection System (DTDCS).
- The Universal Automatic Map Compilation Equipment (UNAMACE) System.
- The Semi-Automated Cartographic System (SACARTS).

SACARTS encompasses a much larger scope than DTDCS and the UNAMACE System. In addition to collecting, processing, and storing digital data, SACARTS considers the total cartographic steps needed for the production of topographic maps. Hence, this

* Any mention herein of a commercial product does not constitute endorsement by the United States Government.

paper addresses only that portion of SACARTS currently operational for the creation of digital data. Subsequent papers are forthcoming on full exposition of SACARTS.

DIGITAL TOPOGRAPHIC DATA COLLECTION SYSTEM (DTDCS)

A decade ago, DMATC (then AMS) began collecting digital terrain data on a production basis. As originally conceived and elaborated in the publication entitled "Programming Topographic Maps for Automatic Terrain Model Construction" by Messrs. Noma and Misulia in 1959,[3] the system automated the carving of three-dimensional terrain or relief maps.

During 1961–62, the initial approaches were modified and a new digitizing technique and processing methods were implemented. The equipment constructed, and concept still in use, was the Digital Graphics Recorder (DGR), sometimes called the Floating Arm Graphics Recorder (FAGR), Figure 2. The April 1963 issue of the Communications of the ACM, Vol 6, No. 4, the journal of the Association for Computing Machinery, presented a description of a "prototype" system.[4] The system consisted of a digitizing table which collected the incremental motion of traced contour lines on a magnetic tape. This was followed by a sequence of computer programs to verify the data by producing plots, filling gaps between contours by interpolation, and outputting a tape for directing a tape-controlled milling or carving machine, Figure 3.

Before the fully automatic modeling capability was put into use, the system's ability to produce digital terrain data triggered production programs to provide data for a number of Department of Defense (DoD) agencies.

The requirement to provide elevation data temporarily shelved the completion of a milling device until 1967–68, and the system was successfully put into operation in 1969. The publication "Automated Terrain Modeling" by Messrs. Mays and Noma in 1970[5] described the system, Figure 4. Figure 5 illustrates the surface appearance depictable using the system. One of recent experiments carried out is the digital enlargement of the horizontal and vertical scale. That is, data collected and stored from 1:250,000 scale have been carved at 1:125,000 and even 1:62,500. The shrinkage of scale is also possible.

The increased production requirement by the DoD community for digital elevation data accelerated the acquisition of more rapid and efficient collection equipment, culminating in the Digital Topographic Data Collection System (DTDCS) developed in 1966–67, Figure 6. The DTDCS is designed to handle five DGR tables (maximum) via data flow traffic control by computer. The system collects "on-the-fly" traced data to a core buffer area and thence to assigned tracks on a 1.5 million word capability Disk

Pack. The system will, on the average, hold at least one shift's worth of traced data for each table. Before the data on the Disk Pack are transferred to tape, they are plotted for validation and necessary corrections are made on the disk.

The production rates are for the system to handle 200 to 300 digital terrain sheets per year, or from 4 to 6 sheets per week, or at least one a day. This level of production would require two full DTDCS systems, with 10 DGR's in operation.

The processing software has similarly evolved from original concepts and was modified as production rates increased. The procedure for processing was first published in the paper entitled "Digitizing Graphic Data at the Army Map Service" by Messrs. Mays, Noma, and Aumen. [6] Figure 7 illustrates the current process used.

Present digitizing practice calls for identification of two types of data:

- Contour lines and spot elevations.
- Stream and ridge lines.

On the initial sort/merge step, the digitized data are sorted into profiles and the two types of data are separated. The sorted contour and spot elevation file is used as a control file for the subsequent steps that follow.

The sorted stream and ridge line file is matched to the control file to obtain the elevation values. The data are reordered back into stream and ridge lines and all elevation values for these lines are obtained by ordinary linear interpolation from the intersection points. These data are again sorted back into profiles and merged with the control file.

Next the sheet neatline information is derived. This process uses coordinates provided as input and first generates X's and Y's defining the neatline. Using these points, a match is made with the control file to obtain height information for interpolation in defining all neatline elevations. Here, two types of interpolation have been used; a moving cubic spline fit and an ordinary point-to-point linear interpolation. Experience indicates that the spline produces more rational values if attention is paid to collecting sufficient control points at max-min and inflection points; however, this type of precision normally requires interactive processing which adds to time and effort needed to complete the job. Thus, the current practice has turned more and more toward the use of the simpler but sufficiently accurate linear method.

A repetitive interpolation scheme, called planar interpolation, is used to determine all undefined surface points within the sheet region. The derivation of the method has been described in a number of earlier works and a detailed exposition will not be given here. As shown in Figure 8, the approach is to obtain surface values point by point by passing planes through three known points. The algorithm operates within an ordered array and, hence, at completion of the calculation all points within the sheet are defined.

The process at DMATC is specifically tied to the recording density of the collection system which is 0.01 inch in X and Y for the DGR. Thus, the digital array produced reflects the unit of the digitizer and is independent of source material scale.

As stated earlier, the digitizing and collection of elevation data began over 10 years ago at DMATC. Within this timespan, the completion of digital elevation coverage of the continental United States, Figure 9, is a major accomplishment. This index shows the latest inventory check made of the agency holdings of available digital data. Gaps in the coverage are sheets being verified and inventory checks still taking place. The bulk of the data has been collected and processed from the standard topographic 1:250,000 scale series map and retained on magnetic tape reels by map sheet number. Copies of the data are available in DMATC standard compacted elevations on industry-standard magnetic tapes. The procedure for handling requisition and distribution are described in the current military map supply catalogs.

The current size of library holdings at DMATC is about 1,600 reels. The major coverage is of the United States and plans call for collection in other world areas. The primary source is the 1:250,000 scale topographic map; however, 1:50,000 scale topographic maps will be digitized and processed for areas of specific interest based upon current available coverage.

UNAMACE

Figure 10 illustrates the data flow and processing of the UNAMACE System. The use of digitized data from the UNAMACE System has been described in "Automatic Contouring at the Army Map Service" by Messrs. Viticllo, Biggin, and Middleton in 1968.[7] Additionally, a more recent publication by Biggin, entitled "Computer Generated Contours from Numerical Data," in 1971,[8] describes current work at DMATC. The system produces digitized elevations directly from rectified stereo imagery. As described in the cited papers, the primary effort is directed toward techniques for processing the collected digital data for the automatic plotting of contour lines.

Figure 11 shows the current UNAMACE data flow. First, the data from each model are collected by an update program and stacked onto a master. After all models have been collected for the mapping area, the data are processed through a contour program. The program is provided with orientation parameters and a smoothing criterion. Proper scaling and smoothing is performed on the data followed by a process to digitally "draw" lines for the contour elevations specified. The resulting output of the process are (1) contour plot tape for drawing labeled contour lines and (2) smoothed elevation data in profile to drive an orthophoto device, such as the Gigas-Zeiss, for producing orthophotography.

Two major operational problems in the processing of UNAMACE data to produce usable terrain data are the inability to detect areas of inconsistencies on the digital data and the inability to panel or mosaic digital data to construct consistent terrain data for an entire map region. Areas of inconsistencies occur because of equipment limitations where correlation is lost. Locating these areas and correcting the errors are necessary for effective use of the digital data. Techniques for digital mosaicking are now being developed and plans call for their incorporation as a subsidiary step prior to the contour program.

In addition to the two points discussed, the question comes up as to what smoothing does to the raw data. Is smoothing done to produce cartographically acceptable contour lines or to better represent the true ground configuration? Are the two the same? The point of immediate concern related to the UNAMACE data is whether the master or the smoothed be retained. Perhaps certain cartographic digital definition may be needed to make this determination.

There are approximately 400 UNAMACE master tapes at DMATC. However, their current utility is limited since the data are strictly discrete individual digital models. The effectiveness of using the UNAMACE terrain data will depend upon the successful implementation of a system to collect and combine data to cover entire map sheet areas.

SACARTS

The initial phases of the *Semi-Automated Cartographic System* (SACARTS) have just been put into production use at DMATC. The system is the combination of both hardware and computer software for aiding the cartographer in the production of repro quality materials, Figure 12. Inputs to the system are both hypsography, from processed UNAMACE type elevation data, and planimetric data, from vector or line digitizers such as the CALMA and BENDIX units. The processing of the data will be done on a large central computer such as the UNIVAC 1108. Pr[illegible] the table digitizer is used for servicing, correcting, and adding features; however, p[illegible]s call for interfacing the *Digital Input Output Display Equipment System* (DIODES) for rapid interactive editing. This subsystem consists of cathode ray tubes and digitizing table for displaying and correcting digitized data. For production of repro quality material, a high-precision digitally controlled drafting machine such as the Concord plotter is used.

For more rapid output of high-content data, a raster-type plotter will be used. The raster unit can also be used for rapid scanning of manuscript as input to the processing steps.

SACARTS consists of a number of different computer-controlled subsystems; however, central to the concept is data processing on a large-scale computer. The initial version of processing software has been developed and tested, called Graphic *I*mprovement *S*oftware *T*ransformation *S*ystem I (GISTS I). Details of GISTS data bases have been published in the paper "A System for Automated Cartographic Analysis and Map Production" by Messrs. Burdette, Dario, and Spencer, presented at the October 1973 Regional Cartographic Conference for Asia and the Far East in Tokyo, Japan.[9] Figure 13 shows the general schematic data flow and logical processing of GISTS. Additional details on the system are given in two other papers to be presented at this 1974 Spring ACSM-ASP Convention, "Improved Cartographic Copy from Digitized Map Compilation" by Henry R. Cook[10] and "Color Separation Symbolization in Semiautomated Map Production" by William Burdette.[11] The former paper provides additional details on systems design and data structure while the latter discusses the software techniques used to produce control tapes for outputting repro quality, color-separated, symbolized plots.

This system, as presently constituted, is designed for efficient production of graphics; however, elements for the construction of a planimetric digital data file are there. Data fields are expandable to handle a wide variety of coded line and point features. In anticipation of a raster scanner/plotter, software for determining the inside and outside of regions, which can be highly relevant to an effective digital data base, can be incorporated into the GISTS file structure.

The present digital holding from this system is small and for actual use available only for graphic manipulation; however, as the system is put into production, auxiliary processes will be designed to manage the planimetric data files.

Current production plans call for the system to be phased in, beginning with rates up to 50 map sheets per year to a capacity of 500 map sheets per year in a 3- to 4-year span. With acquisition of additional and new equipment as well as improved software, the system can be pushed upwards toward a 1000-maps-per-year rate. These production rates will obviously create an explosion of digital tape data. Hence, in a few years this system should be a significant generator of digital data, especially planimetric information.

FILE ATTRIBUTES

The three major digital data collection systems can be interfaced for the creation of a common digital terrain, or more generally called topographic, data base. The base should consist of possibly two distinct, however, interrelated files; namely, elevation and planimetric data.

The elevation data should be in the form of elevation arrays organized into uniform logical regions, for instance, 1° by 1° blocks. The planimetric data can be broken into two categories – line and point information. Line data such as drainage, roads, railroads, and other communications networks, should be stored using some form of chain encoding. Point information such as buildings, mine locations, wells, etc., on the other hand, should be stored in position lists by information category for a specified region.

A point for consideration is how area data should be retained. The classification of area data covers such things as lakes, swamps, forests, rice paddies, urban outlines, etc. – in general, cartographic "open windows." These types of data can be retained in two forms: first, as a direction-coded line to indicate the inside and outside of the region outlined by the line, or, second, like elevation data, carried in array form. The choice of the structure may depend on response requirements needed to meet user needs. Array form should be much more effective in responding to inquiry of whether a point is inside or outside of a region, since, if data are in line form, either a search for line must be made or the line data expanded to array to respond.

In order to establish a responsive system, efficient file maintenance and distribution will be necessary. However, before such a system is put into operational use, the present file, as well as files to be generated, must be structured for universal coverage. The current digitized data are being retained in some arbitrary local form; in most cases, the instrument system. These files should be organized into a universal, contiguous system; for example, into latitude bands.

Two major problems must be solved before a universal digital terrain data base or file can be effectively instituted:

- Selection of a universal coordinate system for organizing the digital data.
- Developing a matching process to combine digital data such that information is contiguous.

Aumen has suggested using geographic coordinates as the universal reference. The use of the latitude-longitude system ties effectively to the idea of maintaining data in latitude bands. A fairly uniform system can be developed if the longitude interval increases as latitude increases. However, the selection of a coordinate system is far from complete and further studies should be made in the light of user needs and requirements.

As regards the matching of data to make adjoining data contiguous, different techniques have been studied by a number of groups. One method which has been tried is smoothing by spreading the difference over an arbitrary band adjoining a junction line. Other techniques used have involved mathematical surface fitting. The issue, however, is that a technique should be selected to operate effectively on a universal data base.

Another major point of consideration on data base creation is data compaction. Some studies have been made in this area. However, the current and near-future requirements indicate that use of current computer technology should more than keep up with present files and those being created. Such items as efficient data compaction by chain encoding and use of high-density storage media can comfortably keep pace with the collection rates. As an example, over 6,000,000 elevation points can be stored on a reel of tape using the present DTDCS procedures. By installation of available high-density tape units, the capacity can be doubled. In fact, there are available systems for a four-fold increase if needed. The compaction problem should certainly be studied to be ready to handle the long-range needs, although it is not paramount to the present establishment of a universal digital terrain data base.

SUMMARY

Once the system is established, a positive investment of resources will be needed to maintain the file and handle its distribution. As new data are collected, information should be processed to be incorporated into the data base. This will require a substantial allocation of computer resources, and manpower for reviewing and editing to retain data fidelity.

The distribution of the data will also require the development of software and commitment of computer resources. The copying of digital data for distribution will be one of the simpler functions; but, for effective response, the distribution system should have the capability to selectively retrieve data from only the specified regions as well as extract the feature requested. It should also have the ability for some amount of format changing of the data to fit the user's need. Additionally, future needs may dictate the design of logical synthesizing software to extract for the user implied topographic information for more effective decision making.

The foundation for the creation of a universal digital terrain or topographic data base is available today.

LITERATURE CITED

1. SPOONER, C.S., Jr.; DOSSI, S.W.; and MISULIA, M.G. "Let's Go Over the Hill–Benefits of Profile Scanning the Stereo-Model." *Photogrammetric Engineering.* December 1957.

2. AUMEN, W.C. "A New Map–The Numerical Map." Presented to the 32d Annual Meeting of the American Society of Photogrammetry, Washington, D.C. March 1966.

3. NOMA, A.A. and MISULIA, M.G. "Programming Topographic Maps for Automatic Terrain Model Construction." *Surveying and Mapping.* September 1959.

4. "Terrain Model System." *Communications of ACM.* Association for Computing Machinery. April 1963: Vol 6, No. 4.

5. MAYS, R.R. and NOMA, A.A. "Automated Terrain Modeling." Presented to the 30th Annual Meeting of the American Congress on Surveying and Mapping, Washington, D.C. March 1970.

6. MAYS, R.R.; NOMA, A.A.; and AUMEN, W.C. "Digitizing Graphic Data at the Army Map Service." Presented to the American Society of Photogrammetry–American Congress on Surveying and Mapping Convention, Washington, D.C. March 1965.

7. VITIELLO, D.; BIGGIN, M.J.; and MIDDLETON, G. "Automatic Contouring at the Army Map Service." *Photogrammetric Engineering.* October 1968.

8. BIGGIN, M.J. "Computer Generated Contours from Numerical Data." Presented to the American Congress on Surveying and Mapping American Society of Photogrammetry Convention, Washington, D.C. March 1971.

9. BURDETTE, W.; DARIO, E.; and SPENCER, N. "A System for Automated Cartographic Analysis and Map Production." Presented to the Seventh United Nationals Regional Cartographic Conference for Asia and the Far East, Tokyo, Japan. October 1973.

10. COOK, H.R. "Improved Cartographic Copy from Digitized Map Compilation." Presented to the Annual Convention of the American Congress on Surveying and Mapping–American Society of Photogrammetry, St. Louis, Missouri. March 1974.

11. BURDETTE, W.H. "Color Separation Symbolization in Semiautomated Map Production." Presented to the Annual Convention of the American Congress on Surveying and Mapping–American Society of Photogrammetry, St. Louis, Missouri. March 1974.

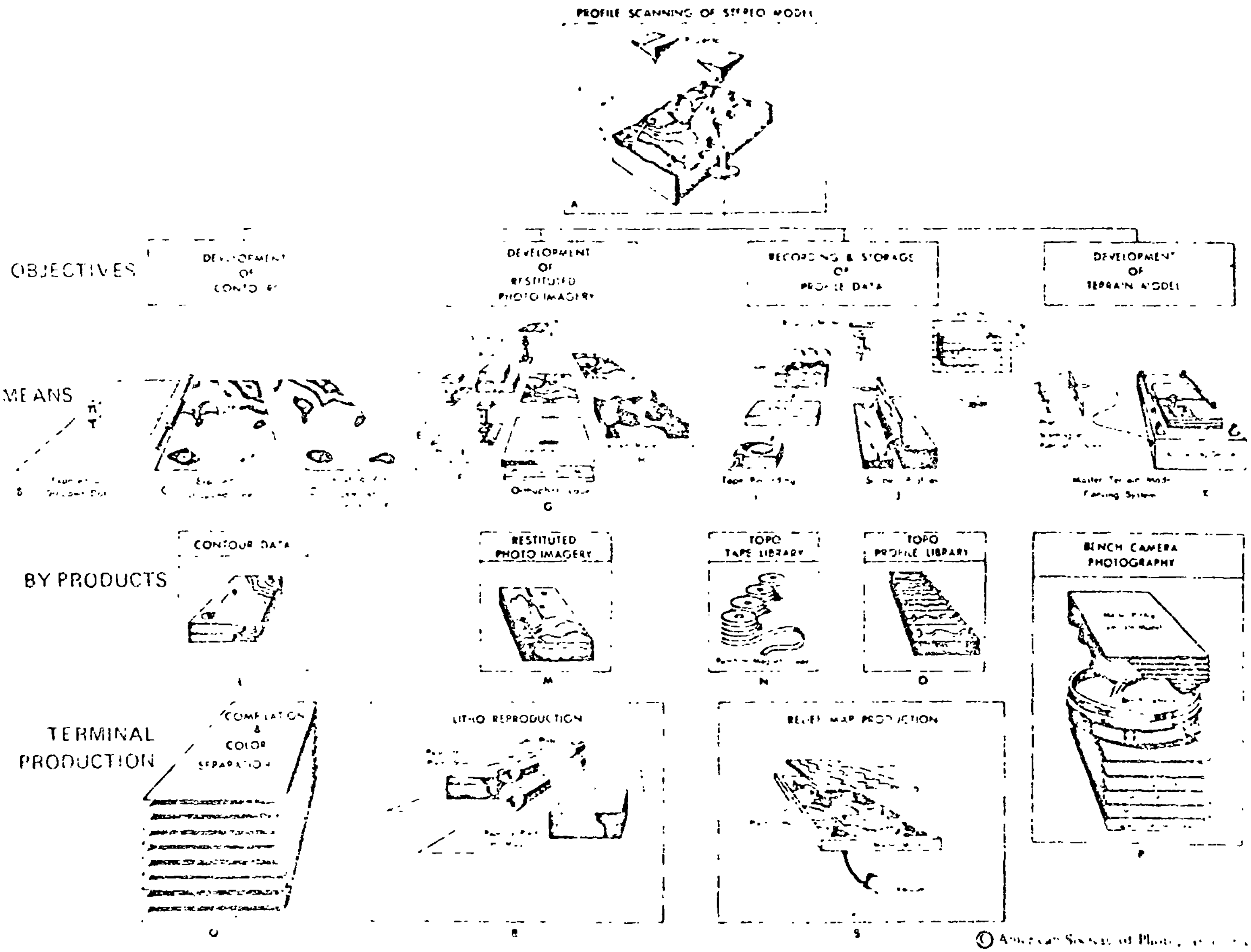


Figure 1. Integrated mapping system diagram.

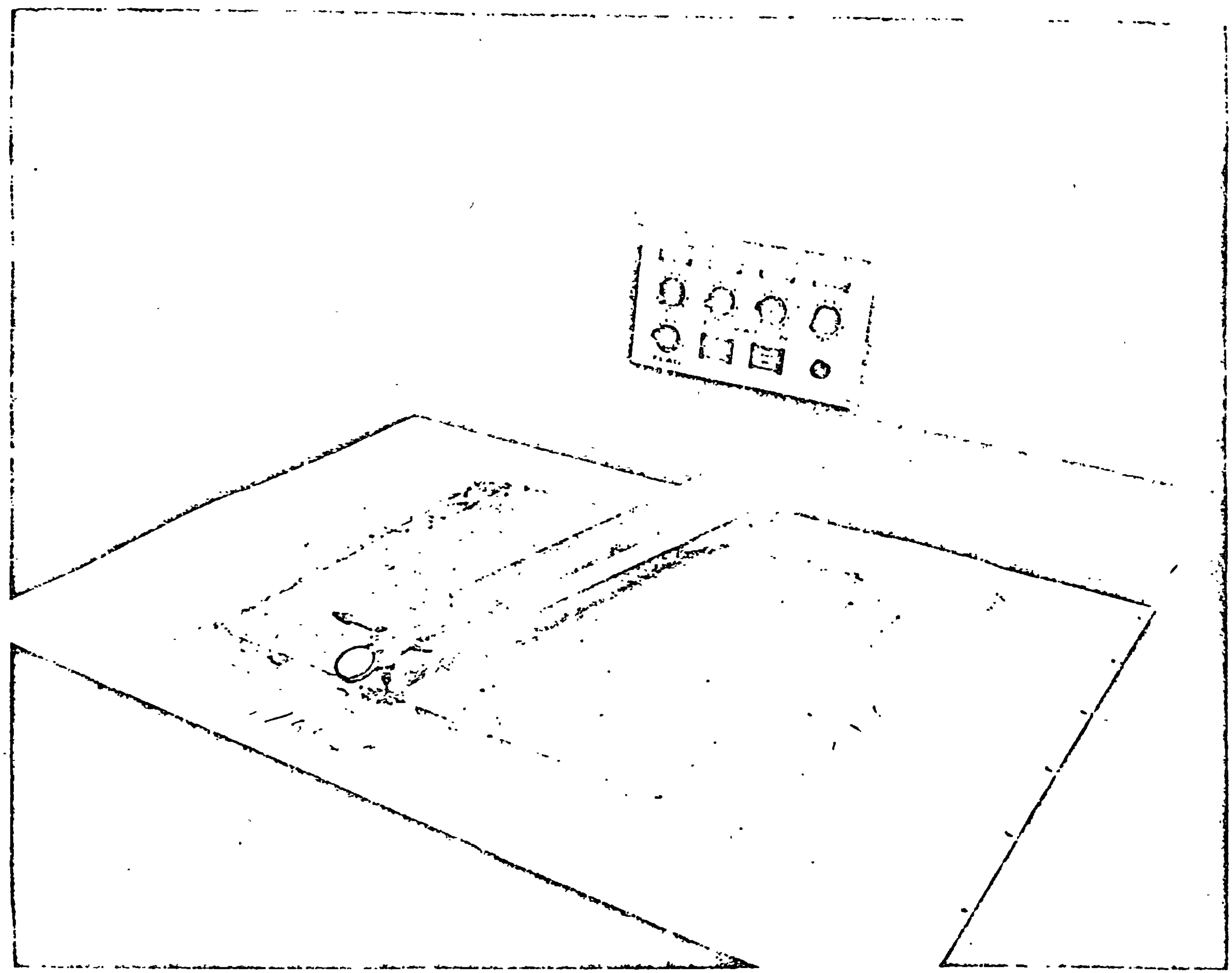


Figure 2. Floating arm graphics recorder.

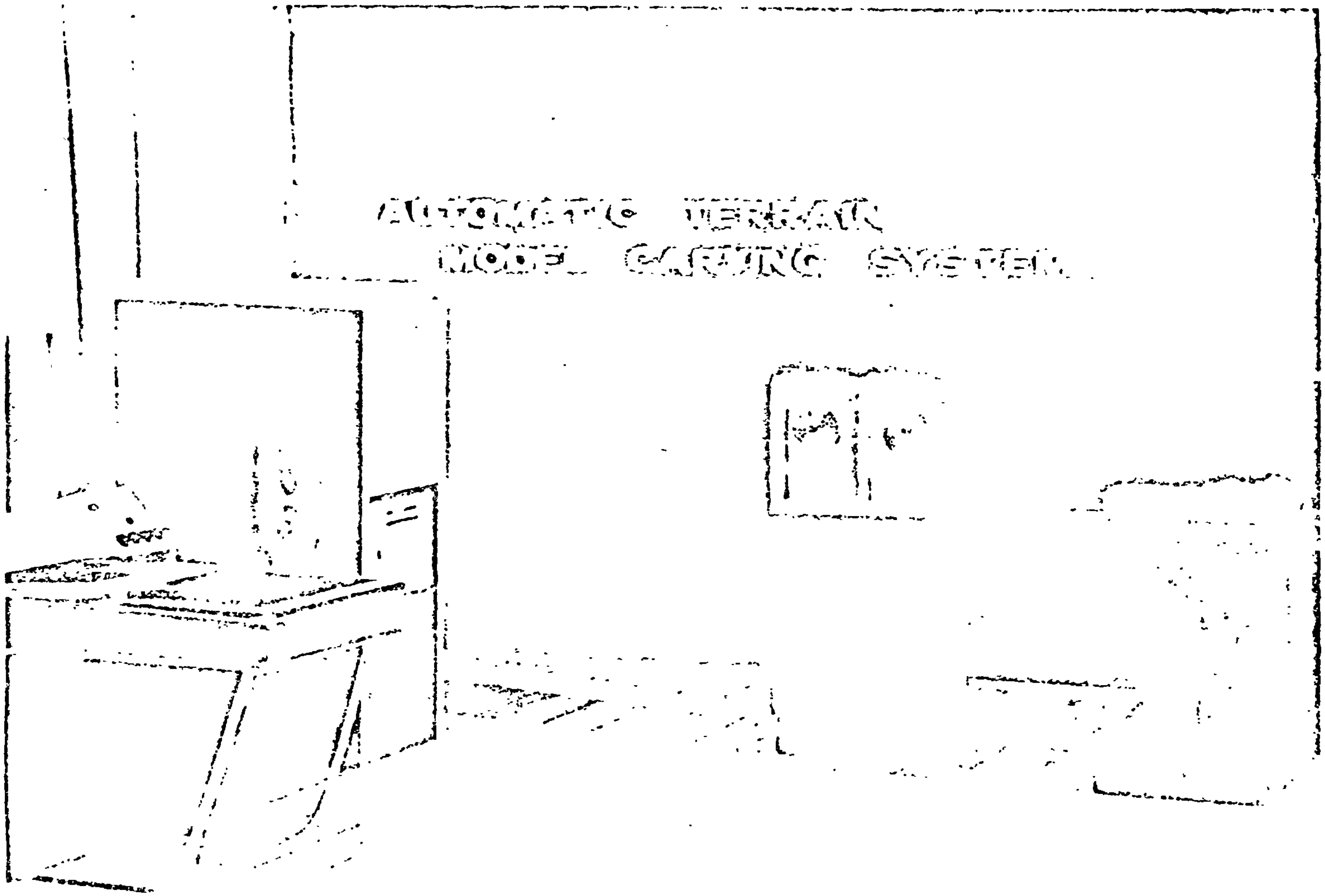


Figure 3. Automatic terrain model carving system—model of layout.

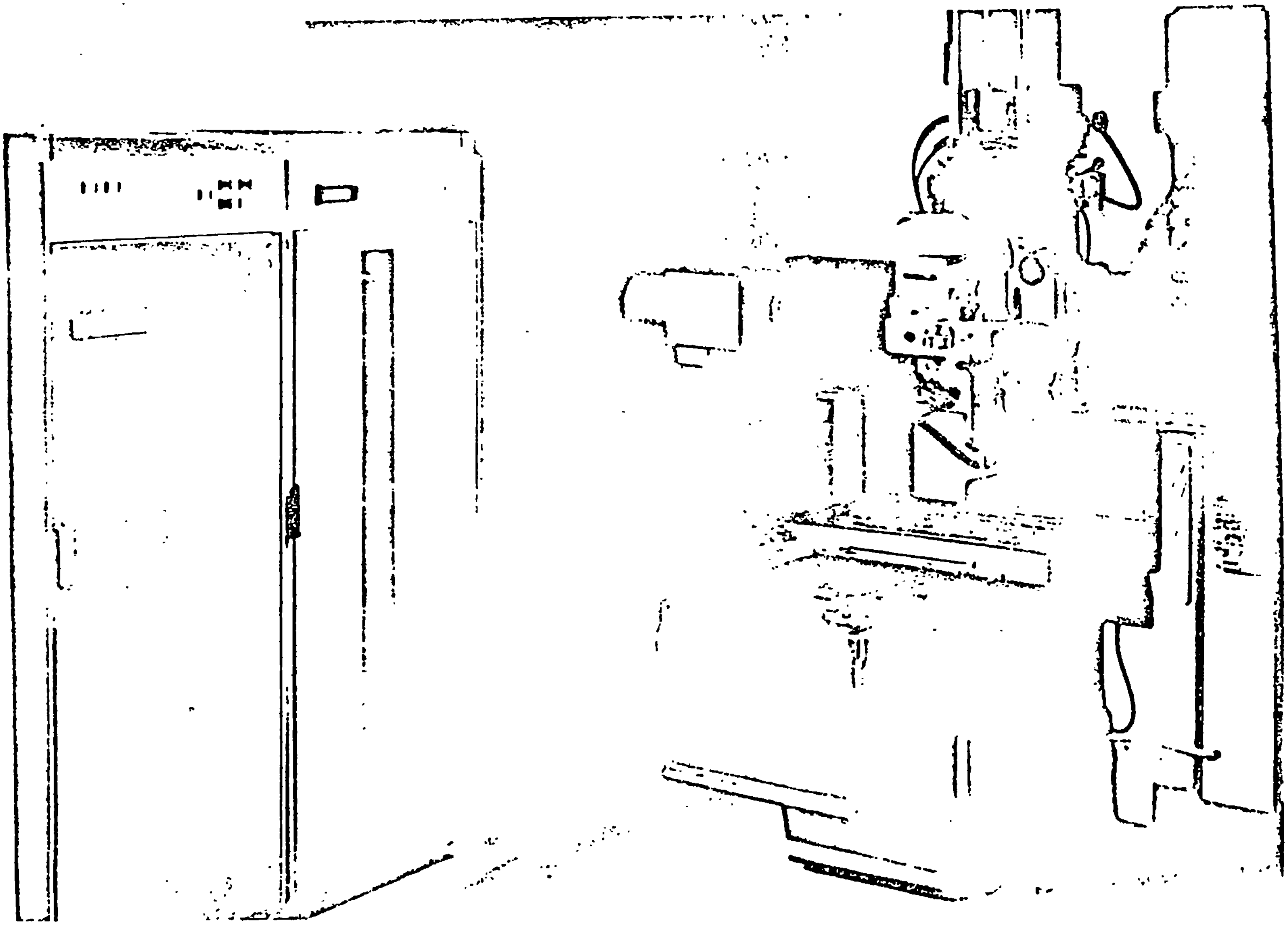


Figure 4. Automated terrain model carver.

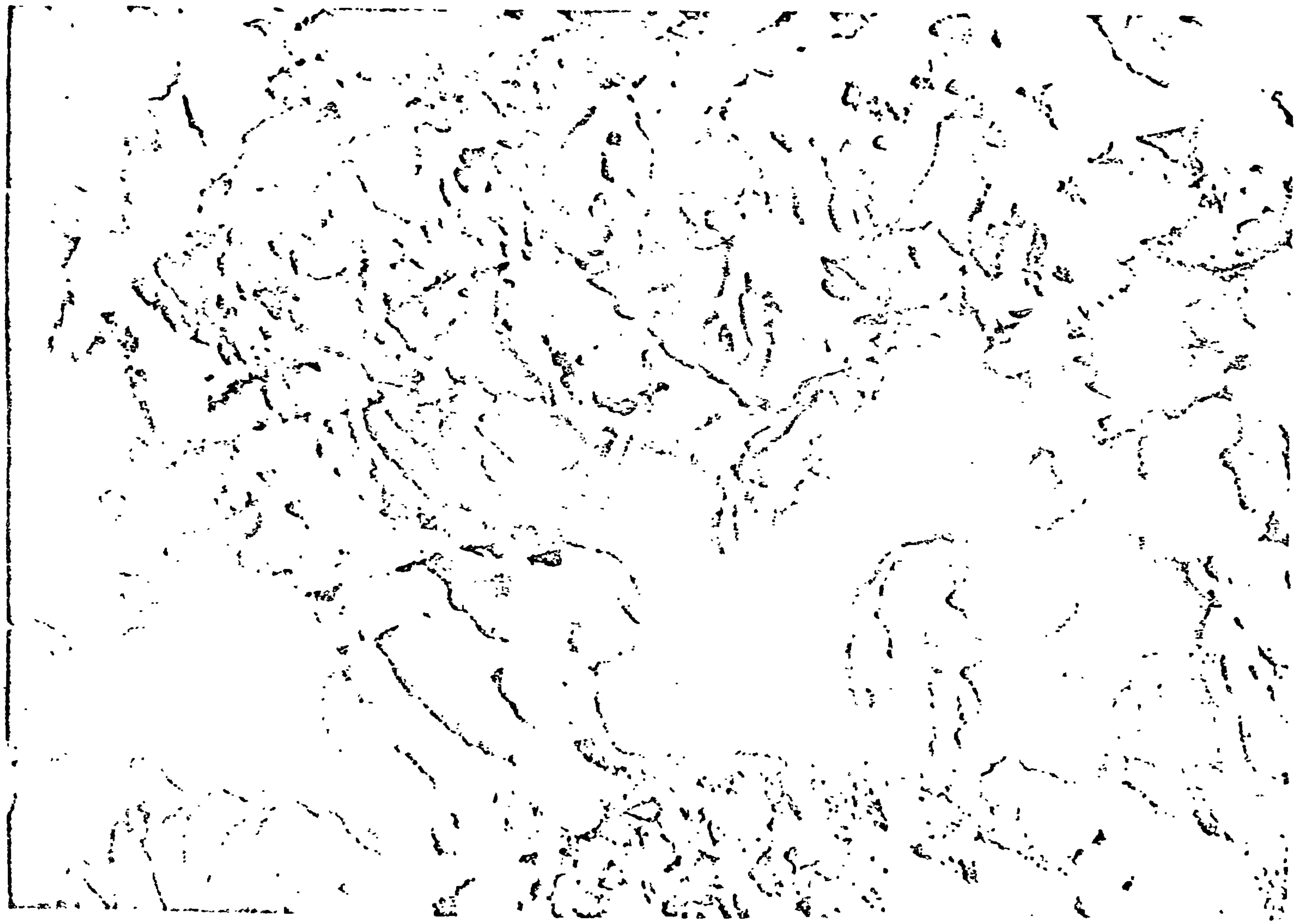


Figure 5. Terrain model carved by system.

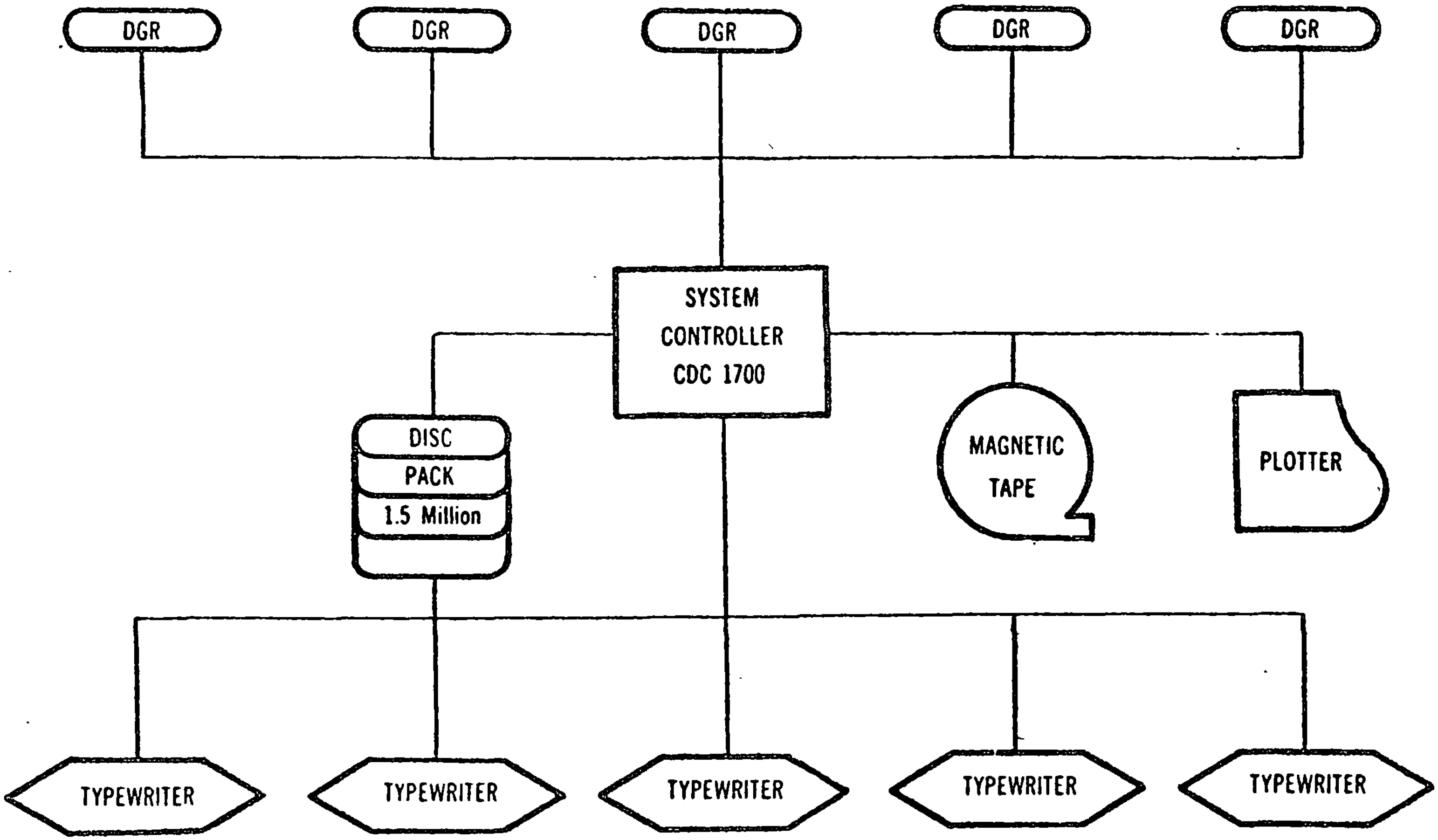


Figure 6. Digital topo data collection system

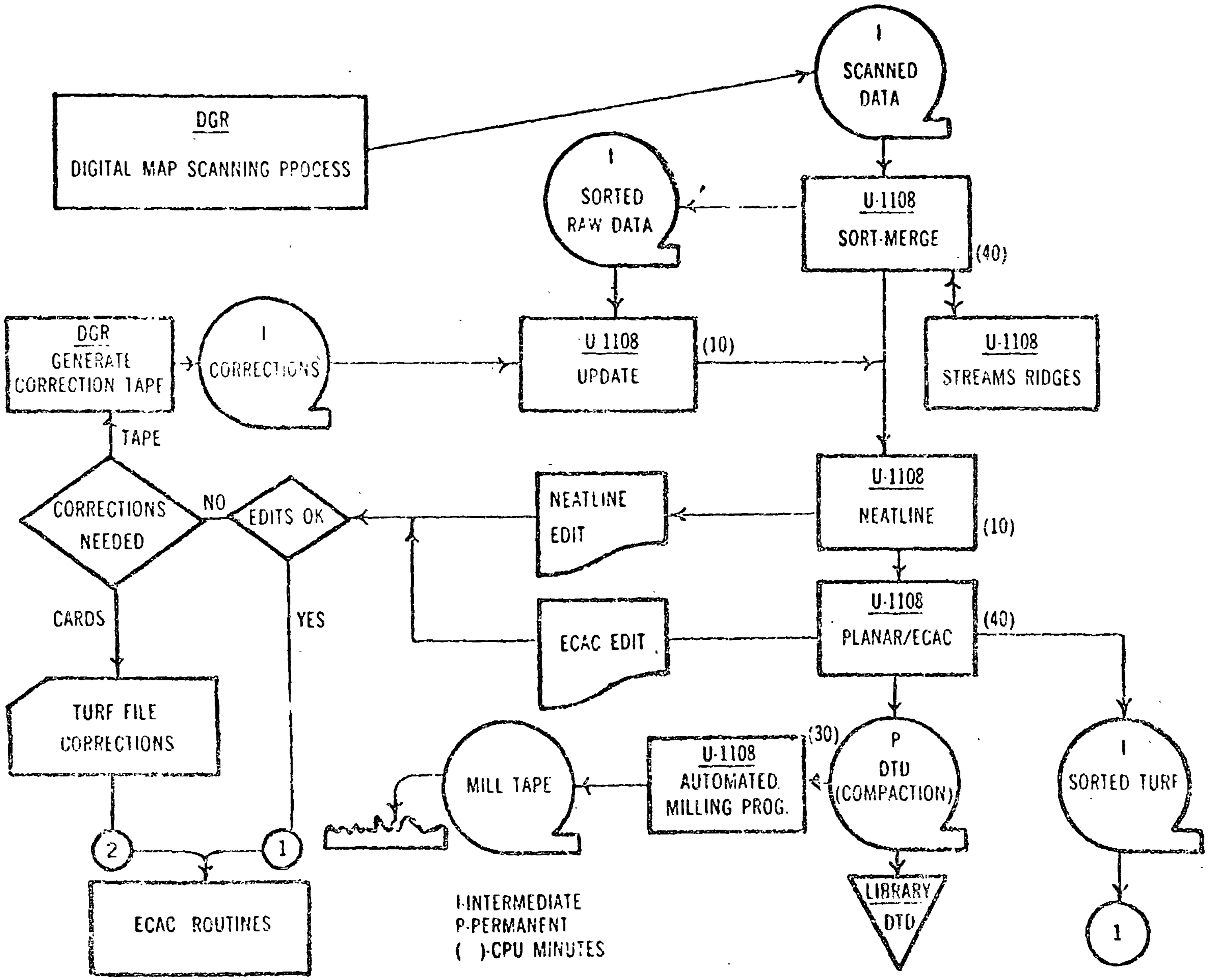
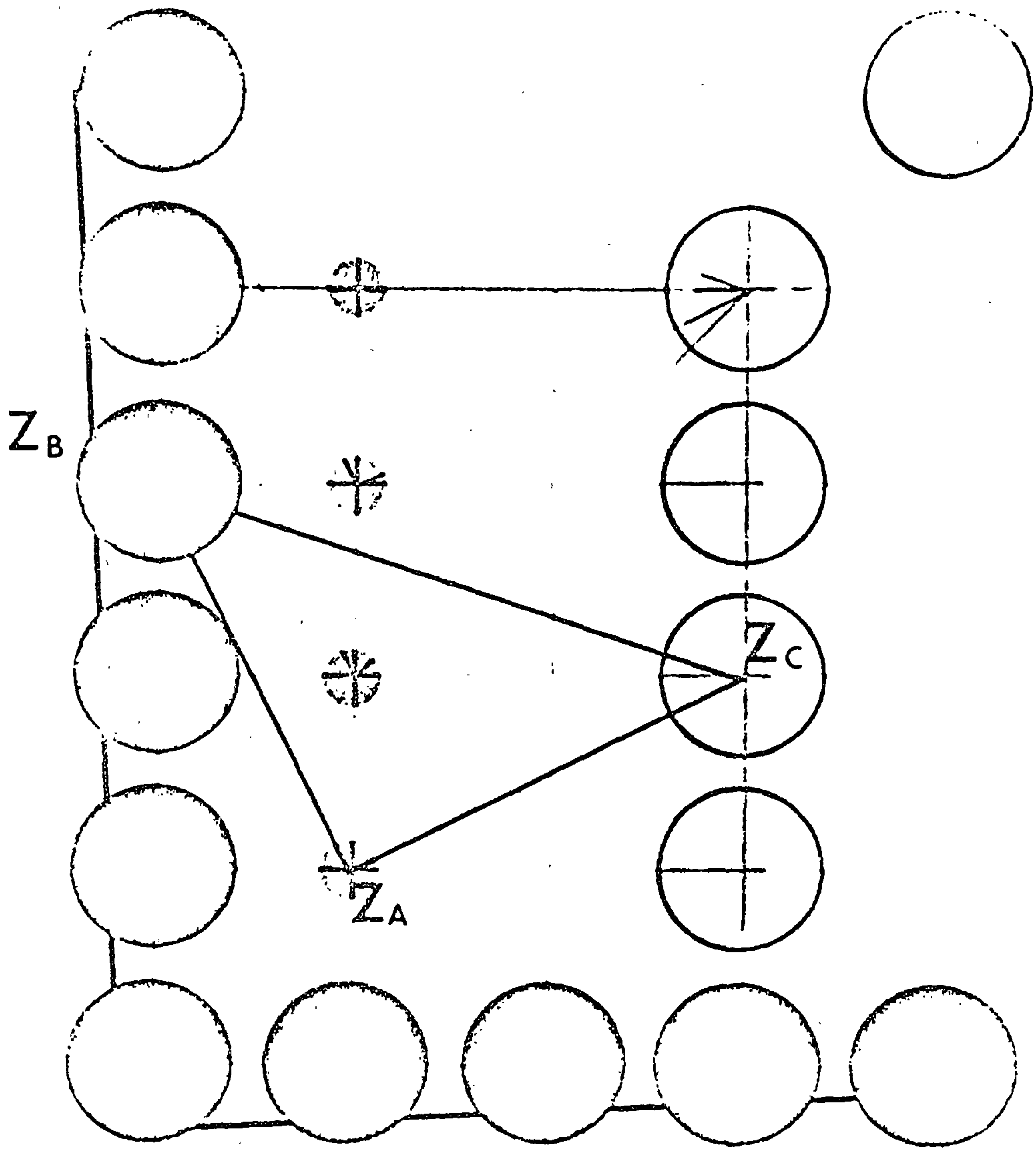


Figure 7. Digital graphic data flow.



$$Z = \frac{Z_A(y + x) + Z_B x + Z_C}{2x + y + 1}$$

Figure 8. Planar interpolation.

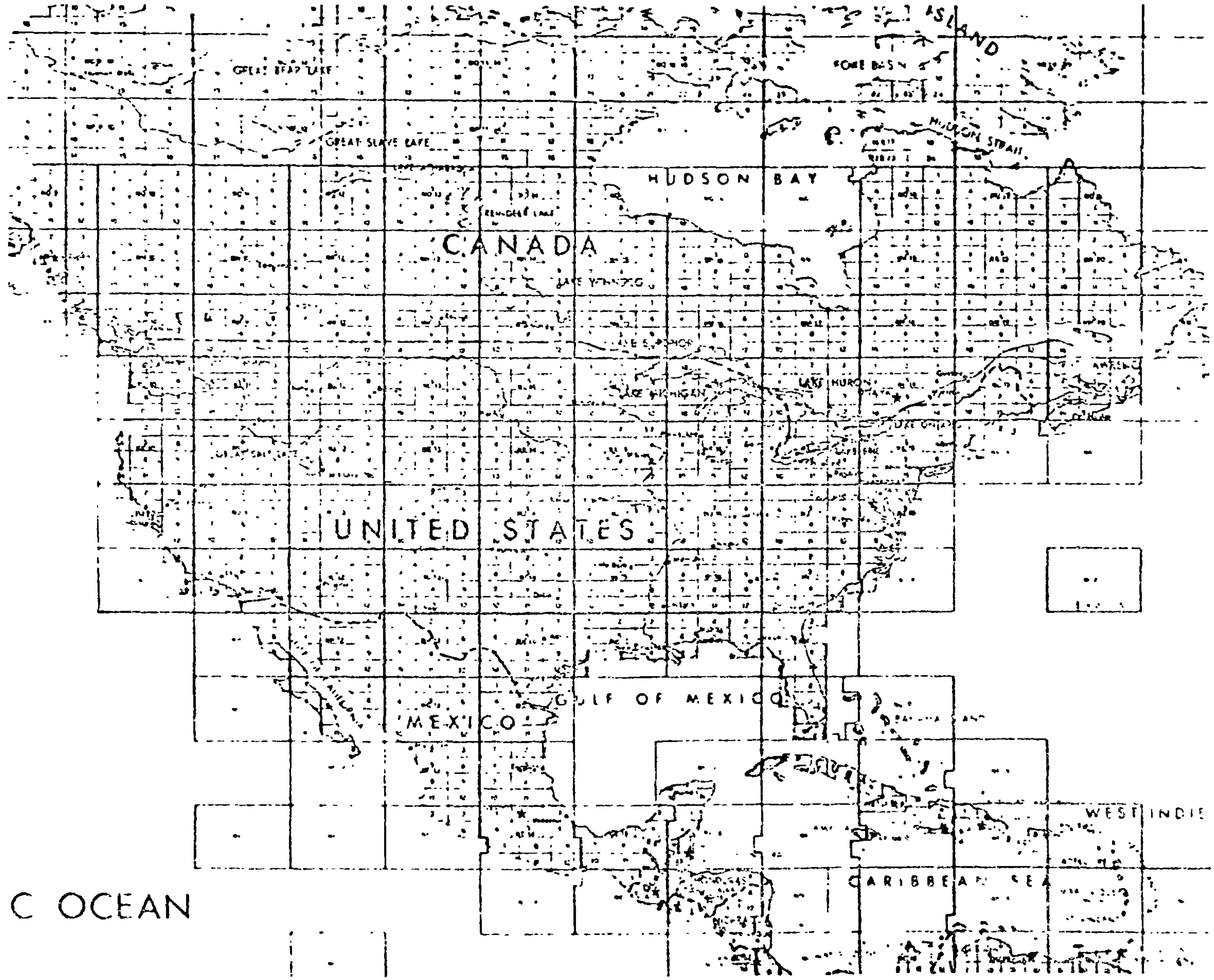


Figure 9. Index of digital elevation coverage of continental United States.

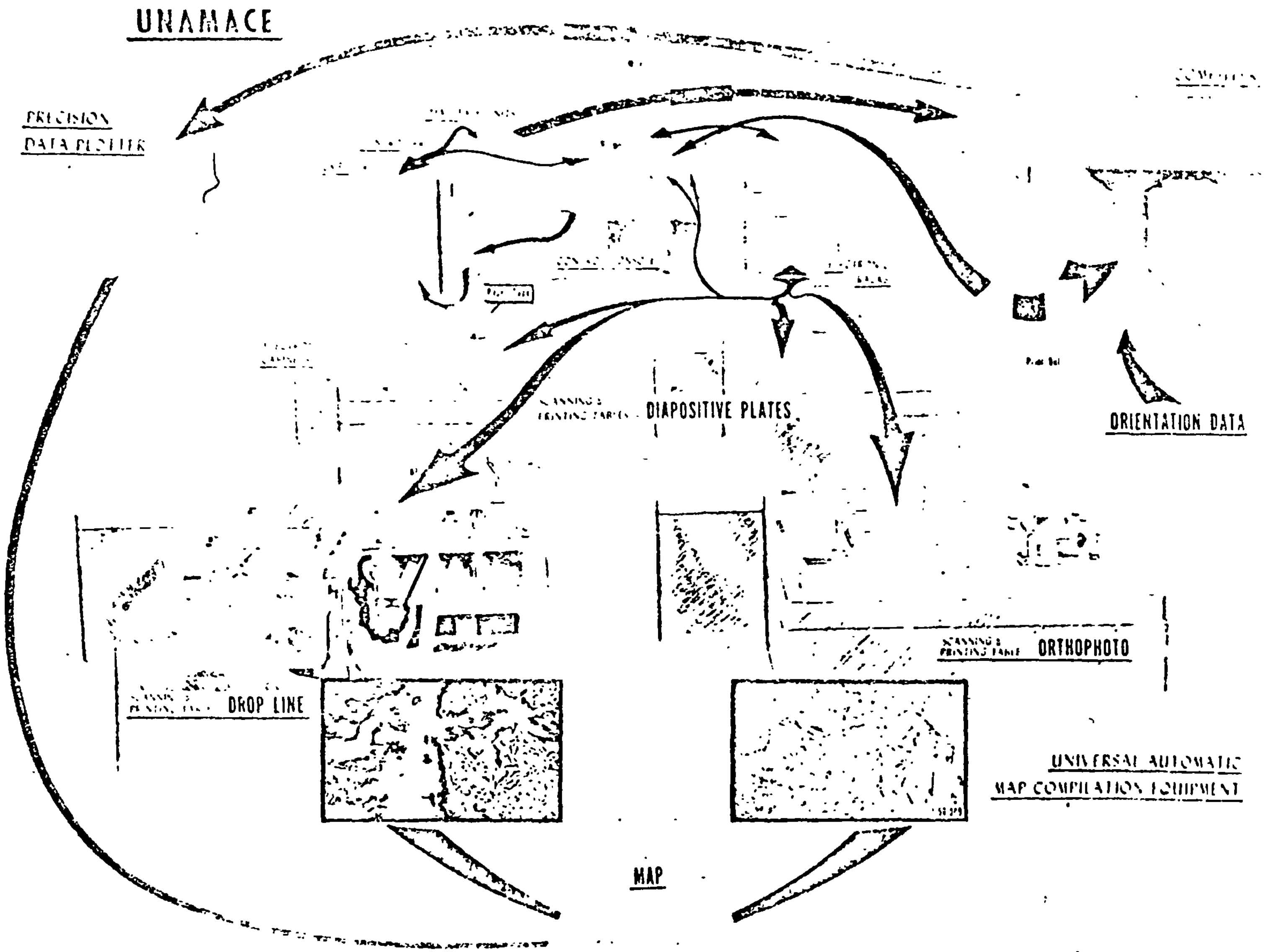


Figure 10. UNAMACE data flow model.

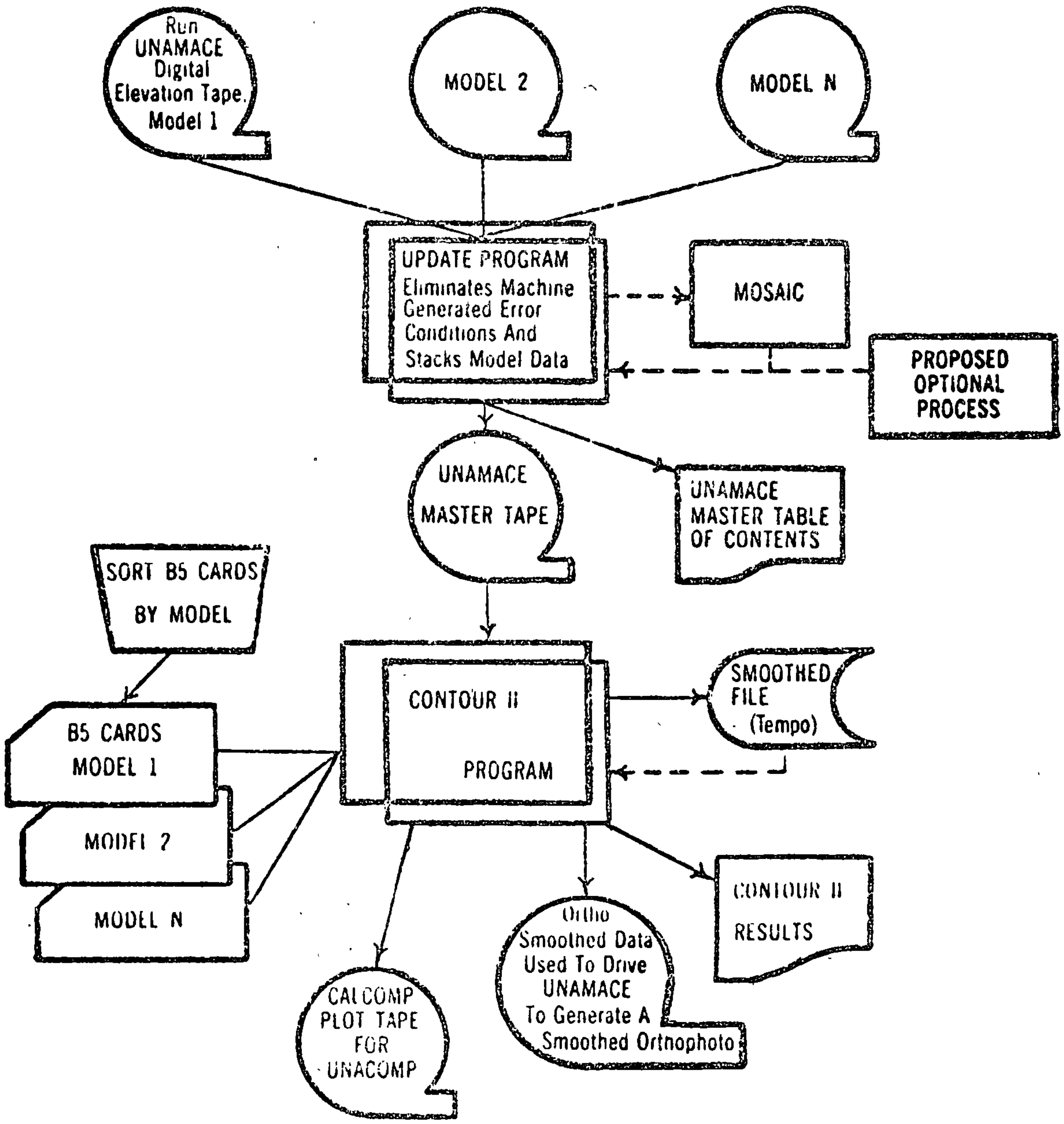


Figure 11. UNAMACE data flow chart.

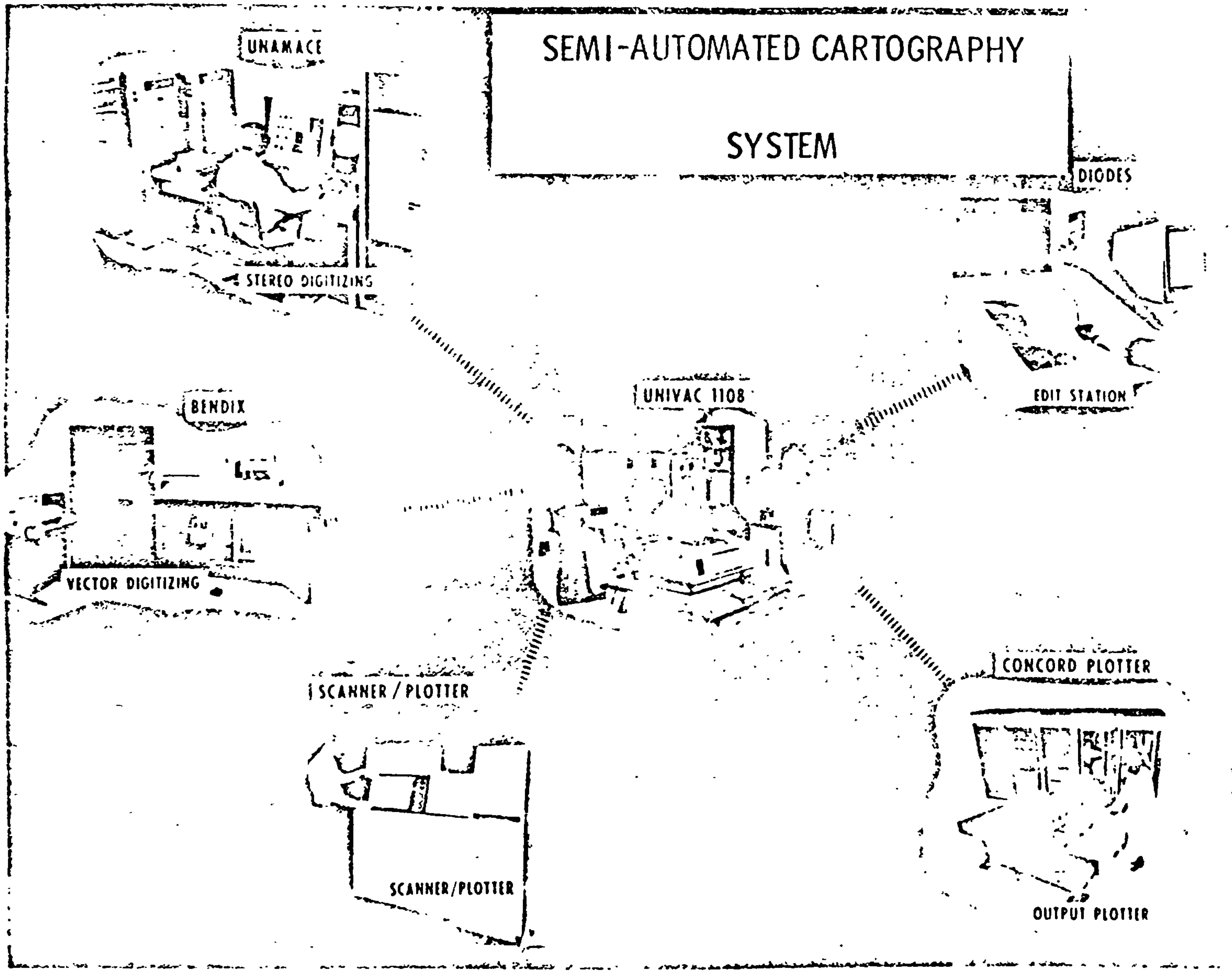


Figure 12. Semi-Automated Cartography System.

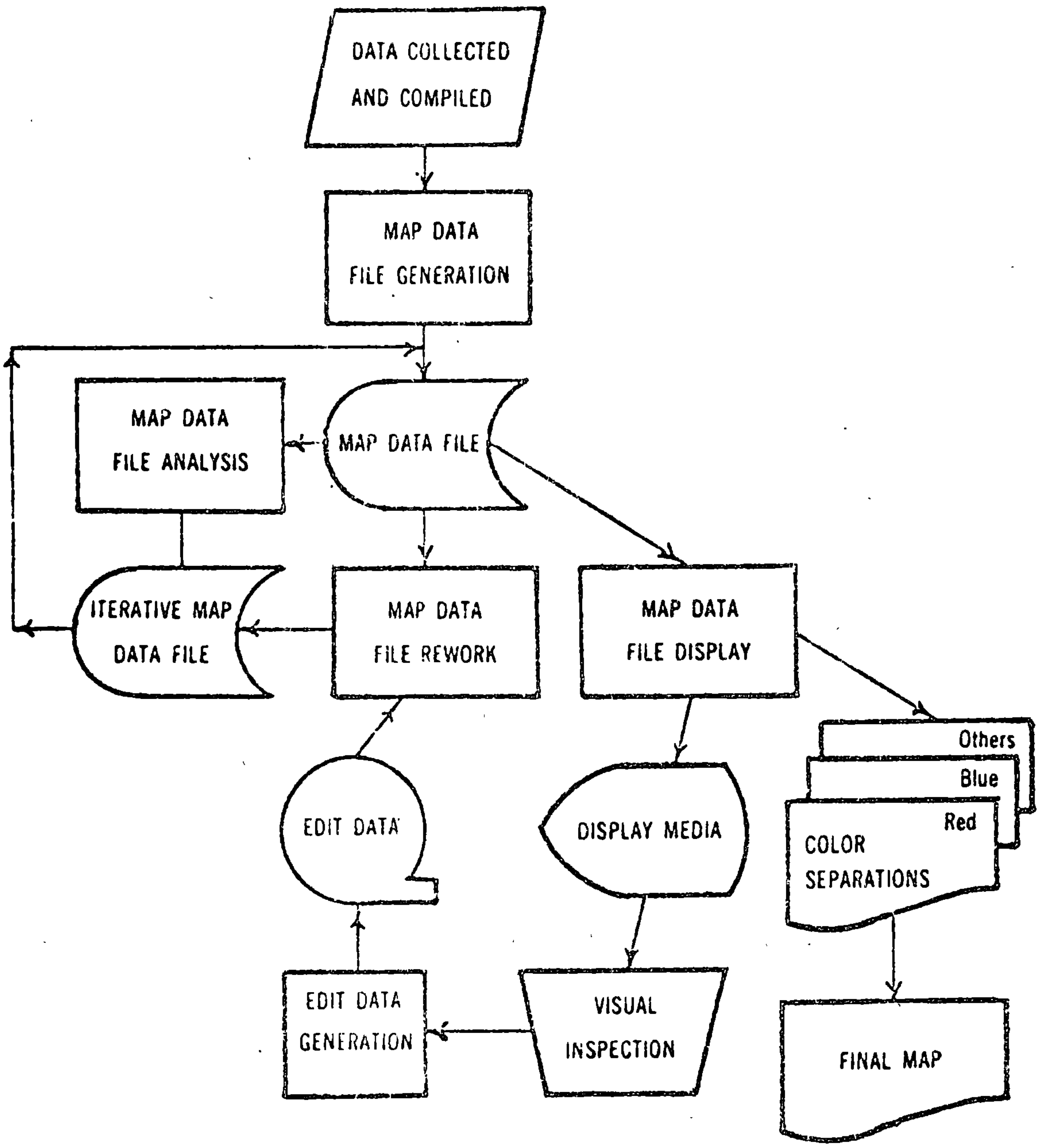


Figure 13. Graphic Improvement Software Transformation System data flow.

A Framework for Encoding Spatial Data*

Kenneth J. Ducker

K 152, Encoding, Polygons

Increasingly, geographers are encoding data for machine processing, yet replicating complex geographic patterns in machine-readable form for processing is poorly understood. This paper sets forth alternative methods for encoding geographic patterns.

Geographic information systems differ from other information systems in the explication of locational identifiers, and the importance of the locational identifiers in manipulating data. This paper accomplishes two tasks: 1) the development of a conceptual framework for encoding spatial data for incorporation into geographic information systems, and 2) the presentation of a notation for interpreting the framework. The choice among alternative ways of encoding spatial data is determined for a particular application by considering the purpose as constrained by the available hardware/software environment, quality control needs, and the magnitude of data to be encoded.

Locational Identification

Spatial data are conveniently stored in machine-readable form, not as the image itself, but in some abstract form, such as intensity values for small grid units or attributes of areal units.

The subsets of spatial data that are addressed here are those phenomena, such as land use, ownership patterns, and vegetation coverage, that can be exhaustively partitioned by type or class into regions described as polygons.

For example, assume that an aerial photograph has been exhaustively partitioned into a set of polygons that identify homogeneous land regions, i.e., residential, non-residential, agriculture, wooded areas, etc. Replication of these various sized regions in digital form usually is done as a sequence of coordinate values or equivalents describing the perimeter of each area. Problems and potentials of this mode of operation are described in more detail in the next section.

Encoding Data

Technical problems in the development of a geographic information system in the generic sense can best be approached by developing a conceptual framework for encoding geographic data, i.e., points, lines, and areas, and then evaluating alternative ways in which the encoded geographic data can be input, stored, retrieved, and output. Clearly, the major issue is the efficient manipulation of geographic data. Once that issue is determined, any number of data items related to a specific point, line or area can be included providing additional data storage is made available.

All elements of the information system including input and output must be viewed in the context of handling geographic data encoded as points, lines, or areas. For example, data might be abstracted from imagery in terms of coordinates for points, and areas encoded as a sequence of points. Data might then be stored as coordinate values for points making up areas, and the data might be displayed as line segments making up the system of areas. Clearly, the input, storage, and output encodings are not independent. Translation from one stage to another must be thought out in advance.

Figure 1 illustrates alternative ways of encoding geographic data. Depending upon ultimate needs, and the storage media and file structure environment, not all possible encoding schemes need be utilized. However, more than one is usually needed to provide a redundant coding for editing purposes to detect errors for quality control and completeness. Some encodings can also be generated from others, which provides the means for edit. Each of the procedures listed has a variety of advantages and disadvantages associated with it when viewed in light of the storage, comparison, retrieval and output elements of the geographic information system.

The choice of an encoding method from the alternatives listed in Figure 1 must be made considering: (1) usefulness in terms of purpose, (2) ease of data capturing or encoding, (3) ability to generate other encodings for error detection, (4) ambiguities, causes by non-unique identification, (5) ease of synchronizing graphic data to descriptive data at input, and (6) storage media and file structure available. When comparing these criteria with the possible encodings, some encodings are more appropriate than others for typical uses of image data in machine-readable form. For example areas encoded as polygons (encoded method #1) are highly useful as is, can be used to generate other encodings, and can be encoded with ease. Conversely, areas encoded as being contiguous to other areas (#2) is less useful and may not uniquely identify an area when two separate areas are wholly contained within the same larger area. Also, method #4 is difficult to encode directly, although the encoding can be generated from method #3. For error detection it is important to have two independent encodings one of which can generate the other for comparison.

The input problem can be considered a problem of digitizing and coding a sequence of points that make up an area, or a line segment. Alternatively the image could be scanned and encoded as small xy cells which are or are not a part of a line segment. The data storage problem is one of selecting the appropriate encoding method to meet retrieval needs and at the same time be compatible with the file structure and medium upon which the data are

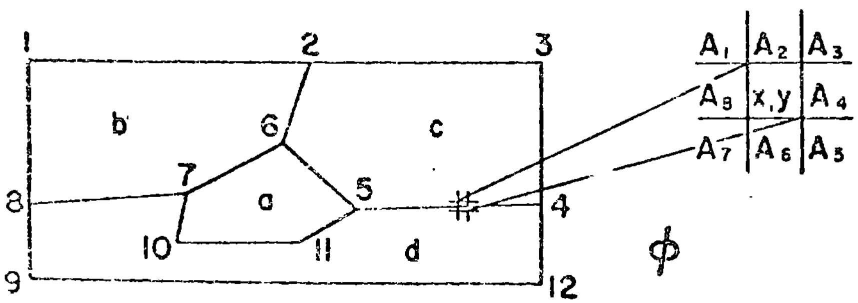
*This study was supported by the Geographic Applications Program, United States Geological Survey under Contract 14-D8-001-12505. The author gratefully acknowledges the comments and suggestions of Samuel Arms, Frank Horton, Robert Shaw, and Robert Wittick.

Kenneth J. Ducker is associate professor of urban and regional planning and geography at the University of Iowa.

...red. Pattern recognition type queries may require contiguity characteris-cs about small grid units be stored in a direct access mode so as to assemble omogeneous regions for comparison to a mask. Other queries might require mparison of two polygon sets, such as determining areas of vacant land se with areas of land suitable for industry. In this case the existing land se data set would be compared to the land capability data set. Thus data must be encoded to make area description and area characteristics acces-ble and comparable.

CONVERTING GEOGRAPHIC DATA TO MACHINE-READABLE FORM

Specifically, three different input methods for converting geographic data o machine-readable form are utilized. The assumption here is that geo-



IG. 1. Alternative Methods of Encoding Geographic Data.
Possible Encodings

1. Areas encoded as polygons made up as a sequence of points:
   E(a) : A(6, 5, 11, 10, 7, 6,) [Read as the entity is area *a* and attributes are a sequence of point numbers 6, 5, 11, 10, 7, and 6.]
   E(b) : A(1, 2, 6, 7, 8, 1)
2. Areas encoded as being contiguous to other areas (where φ is the area outside the area system):
   E(b) : A(c, a, d, φ)
   E(b) : A(c, d, b)
3. Line segments encoded in relationship to their end-points and their contiguity to areas
   E(1,2) : *A*(*b*,φ)
   E(1,8) : *A*(φ,*b*)
   E(2,3) : *A*(*c*,φ)
   E(2,6) : *A*(*b*,*c*) } reverse encoding for redundancy edit
   E(6,2) : *A*(*c*,*b*) }
4. Points encoded as being connected to other points
   E(1) : *A*(2,8)
   E(2) : *A*(1,3,6)
5. Points encoded as being related to areas
   E(1) : *A*(*b*)
   E(2) : *A*(,*c*,*b*)
   E(3) : *A*(*b*)

...phic patterns are manually identified as line segments that exhaustively partition the map or image into areas. The three ways of making the resultant polygons machine readable are:

1. point digitizing polygon vertices,
2. polygon trace with line follower digitizer, and
3. scanning for the presence and absence of lines making up polygon regions.

In all three cases the image must be partitioned into areas representing phenomena, such as land use, land capability or jurisdictional areas. The problem is to describe these areas in machine readable form for internal computation of intersections between phenomena or data sets and mapping of the original data or some derivative therefrom.

Point digitizing is the most direct way of making area data machine readable and requires the least hardware and software capability. Operating in this mode all polygonal vertices are digitized and in a separate operation all the polygon vertice numbers are coded, and sometimes for edit check the line segment, i.e., connectivity between vertices are coded. The resultant entities for this encoding process are:

E(vertice number) : *A*(*x*,*y*);
E(polygon) : *A*(an ordered sequence of vertice numbers bounding the polygon);

6. Small grid units encoded as whether part of line segments or not E(x,y) : A(0 or 1) absence or presence of being on a line segment.
7. Small grid units encoded as whether they are same as contiguous grid units (E(x,y) : (A_2, A_3, A_3, A_4, A_5, A_6, A_1, A_1) = (0, 0, 0, 1, 0, 0, 0, 1)
8. Point Connectivity Matrix

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	0	0	0	0	0	1	0	0	0	0
2	1	0	1	0	0	1	0	0	0	0	0	0
3	0	1	0	1	0	0	0	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0	0	0	1
5	0	0	0	1	0	1	0	0	0	0	1	0
6	0	1	0	0	1	0	1	0	0	0	0	0
7	0	0	0	0	0	1	0	1	0	1	0	0
8	1	0	0	0	0	0	1	0	1	0	0	0
9	0	0	0	0	0	0	0	1	0	0	0	1
10	0	0	0	0	0	0	1	0	0	0	1	0
11	0	0	0	0	1	0	0	0	0	1	0	0
12	0	0	0	1	0	0	0	0	1	0	0	0

9. Area Connectivity Matrix

	a	b	c	d
a	0	1	1	1
b	1	0	1	1
c	1	1	0	1
d	1	1	1	0

10. Coordinate definition of:
    a. points
    b. area bounds
    c. area centroids

line segment: vertice *i*, vertice *j*) : *A* (polygon code for left adjacent area, polygon code for right adjacent area).

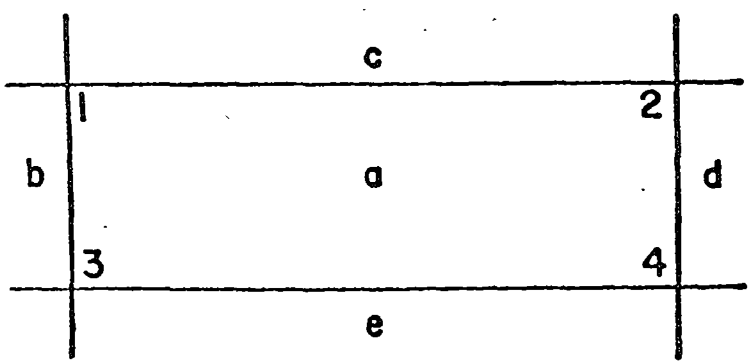
ie DIME Geocoding System (3) and MAP/MODEL (1) both use tions of this encoding scheme. Initially, only the line segments are en- d in the DIME mode. Polygons are in effect created as the DIME al- hm links line segments based on polygon (block) codes and redun- ly checks by matching-up the block numbers, as illustrated in Figure 2. ie Primal edit sorts boundary segments in order by matching node num- The edit traverses the boundary of the block by linking the segments ther in order by matching node numbers.

ie redundant edit also traverses the boundary, with the block left and right fields being switched if the nodes have to be reordered to see if block number appears entirely in the block right field.

ne MAP/MODEL System encodes the points of inflection and junc- s of polygons. The resultant entity is an ordered set of vertices for each gon:

E (polygon code) : *A* (*x,y* coordinates of points of polygon inflection and junction).

he edit procedure used in MAP/MODEL utilizes the characteristic of method that encodes each line segment twice (because it is part of two



2. DIME Edits.

IME Coding [*E*(*i* node, *j* node)]: *A* (left block; right block))

Node i	*Node j*	*Block Left*	*Block Right*
1	2	c	a
1	3	a	b
3	4	a	e
4	2	a	d

imal Edit (traverses boundary) — *Redundant Edit* (block encoding)

Node i	*Node j*	*Block Left*	*Block Right*
1	2	c	a
2	4	d	a
4	3	e	a
3	1	b	a

polygons). The separately encoded line segments are compared and if they fall within a given tolerance are accepted; if not the records are rejected and a correction must be made.

Input from imagery via a polygon trace with a line follower digitizer is sometimes used to digitize a coastline, contour lines or irregular polygons. This eases the input problem significantly with resultant ease of instructions to the digitizing personnel. The only instruction necessary is that the digitizer stylus trace around each polygon. The resultant entity:

E (polygon code) : *A* (*x,y* coordinates sequentially ordered and at regular intervals)

In some instances, a computer algorithm must then be developed to identify junctions and points of inflection to create a polygonal file changing the attribute structure of a sequence of *x,y* coordinates to a sequence of line segments, thus simplifying the polygonal description similar to one developed by the point digitizer and manual coding. The edit employed by this method utilizes the characteristic of the method that encodes each line segment twice. The separately encoded line segments are compared and if they fall within a given tolerance are accepted. The resultant record for storage and retrieval is the same as the resultant records from point digitizing.

A drum scanner as used by the Canadian Land Inventory (2) creates machine-readable records for each resolution cell of the image, consisting of whether a line is present or absent for that cell:[1]

E (*x,y*) : *A* (presence or absence of a line segment for *x,y*)

An algorthm is necessary to connect contiguous cells having lines present to form line segments in order to proceed as above. For identification of polygons, a separately encoded polygon number and coordinate value falling within the polygon region must be digitized separately:

E (point within polygon) : *A* (*x,y*).

Again the resultant data for storage and retrieval consists of a polygon description and/or line segment description with derived coordinates for the vertices of the polygon regions.

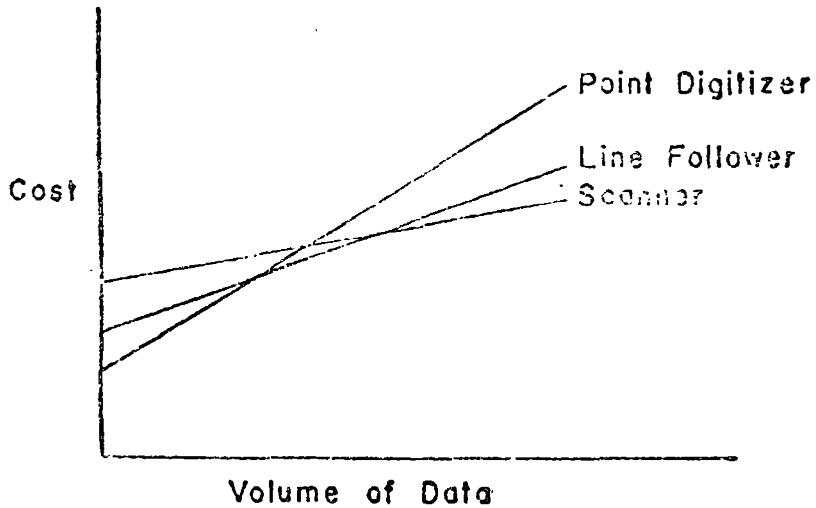
The choice of encoding methods for imagery data is dependent upon a variety of factors, that are largely related to the magnitude of the data being encoded which in turn is a function of the size of the area and the degree to which it is being partitioned into subareas, and the availability of hardware and software. For example, the Canadian Land Inventory procedure requires redrawing maps to be scanned to eliminate all detail but the boundaries of areas.

Operating in the point digitizing mode the costs for digitizing and polygon encoding are high whereas costs for computer processing are low because the encoded data are directed to storage without intervening processing. The

[1]As the "line's" thickness exceeds the cell width the edge must first be determined before the "line" is useful.

: follower mode has a higher computer processing cost reflecting proces- g to determine points of inflection and junction. Operating in the scanner de also involves extensive computer processing to determine end points line segments. There are high costs involved in suitable partitioning of a, topological comprehension, and ambiguities of imagery such as rela- e positional errors.

The scanning mode and the line follower mode have higher initial costs erms of the development of algorithms to convert the data for storage and rieval, but operate more efficiently at higher volumes of data from imag- . Figure 3 represents a conceptual cost comparison of three modes. Sup-



3 Hypothetical Cost Comparison for Alternative Methods of Input Processing of ographical Data.

ing cost comparison data is not readily available though it needs to be ed.

e volume of data is a function of the size of the area being digitized, of the image, the resolution being used, the density of phenomena be- igitized, and encoding method:

Volume = *f* (size, scale, resolution, density, encoding method).

hen the geographic data are densely distributed on imagery and there large number of images to be processed the scanning mode becomes omically feasible. When gross or less dense patterns occur on images, encoding of area definitions and point digitizing appears most feasible.

Closing Note

's paper attempts to explicitly treat aspects of making graphic data

machine-readable that are usually implicitly considered. It stems from the author's view that independent decisions regarding one element of an information system, say input or storage, force decisions pertaining to another element, say retrieval or output, that may be contrary to the purpose of the system.

Viewing ways in which spatial data can be encoded provides the connective tissue by which integrated geographic information systems can be designed. Similarly, the encoding concepts provide a means by which systems can be compared.

Literature Cited

1. Arms, S. "The MAP/MODEL System. System Description and Users Guide," Bureau of Governmental Research and Services, University of Oregon, May, 1970.
2. Tomlinson, R. F. "An Introduction to the Geo-Information System of the Canada Land Inventory," Canada Department of Forestry and Rural Development, Ottawa, Canada, 1967.
3. U. S. Bureau of the Census. *Census Use Study. The DIME Geocoding System*, Report No. 4, Washington, D. C., 1970.

The Trade Area of a Displaced Hexagonal Lattice Point

M. F. Goodchild

Consider the classic central place system of uniform demand density on an infinite plain. Let us suppose that a single entrepreneur is, for some undefined reason, unable to locate at his optimal hexagonal lattice point. He is forced instead to locate at a distance r away in a line bisecting the angle between two of his nearest neighbours (Figure 1). We adopt a coordinate geometry in which the distance separating adjacent lattice points is 1, the optimal point (0, 0,) and the six neighbours are located at $(\pm\sqrt{3}/2 \pm 1/2)$ and (0 ± 1). Our unfortunate friend has been forced to locate at $(r, 0)$.

The upper half of the modified trade area of point 0 is found by connecting the perpendicular bisectors of 0A, 0B and 0C to form $KLMN$. The lower half is symmetrical. By a little application of the principles of coordinate geometry, we find that the trade area is

$$\frac{9\sqrt{3} \quad (1-r^2)^2}{2(3-r^2) \quad (3-4r^2)}$$

M. F. Goodchild is an associate professor of geography the University of Western Ontario, London, Canada.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

PROGRAMAS DE GEO-INFORMATICA
DEPARTAMENTO DE GEOGRAFIA
UNIVERSIDAD DE MICHIGAN

JULIO, 1978.

Palacio de Minería Calle de Tacuba 5, primer piso. México 1, D. F.

S E L E C T E D

C O M P U T F R   P R O G R A M S

Second Printing
1973

A Michigan Geography Publication

Edited by

Waldo R. Tobler

Department of Geography

University of Michigan

Ann Arbor

1970

References:

D. Boettner, ed., Michigan Terminal System, Computing Center, University of Michigan, Ann Arbor, 1967.

C. Dinger, et al, Computer Programs in Oceanography, National Oceanographic Data Center, Washington D.C., 1967, 95 pp.

W.J. Dixon, BMD Biomedical Computer Programs, University of California Press, Los Angeles, 1968, 600 pp.

E. Fronczak, The University of Michigan Digital Plotting System, U.M. Computing Center, Ann Arbor, September 1966, Revisions + 140 pp.

W.L. Garrison, and W.C. Krumbein, Environmental Sciences Computer Study, Northwestern University, Evanston, 15 reports, 1962-1965.

D.F. Marble, Some Computer Programs for Geographic Research, Department of Geography, Northwestern University, Evanston, 1967.

D. Merriam, ed., Computer Contributions, Kansas State Geological Survey, Lawrence, 35 reports, 1966-1970.

NASA, Computer Program Abstracts, GPO, Washington D.C., quarterly.

G. Rushton, et al, Technical Reports, Computer Institute for Social Research, Michigan State University, East Lansing, various dates.

F. Sarno, Polygraphics Manual, Polytechnic Institute of Brooklyn, New York, 1968.

J. Sonquist, and J. Morgan, The Detection of Interaction Effects, Survey Research Center, ISR, University of Michigan, Ann Arbor, 1964.

J.R. Tarrant, Computers in Geography, University of East Anglia, Norwich, Norfolk, England, 1970.

Introduction

The program listings and documentation presented here have been assembled from among those in use at the Geography Department at the University of Michigan. They are obviously the result of the efforts of many individuals, foremost of whom is Frank Rens, for several years one of my graduate assistants. More recently H. Moellering and D. Rhynsburger undertook the documentation task and prepared the programs for publication. None of this work would have been possible without the splendid cooperation from many individuals connected with the University Computing Center. Although the programs have been tested extensively (some for eight years) it is necessary to make the usual disclaimer.

The programs are in FORTRAN (IV, G level) and normally are used under control of the Michigan Terminal System on the IBM 360/67, a very large system by current standards. Several programs make use of the 763 Calcomp plotter via the *PLOTSYS program as described in the plotting manual; some have also been used with the SC 4020 CRT display after relatively minor changes.

Not all of the programs available to the Geography Department are included here. Specifically, the multivariate analysis programs from the BMD system are available for call from teletype terminals; many of the programs listed in the Kansas Geological Survey series, the Northwestern University series, the Michigan State series, et cetera, have been converted for use on the MTS system. The Institute for Social Research and the Computing Center also maintain large files of programs. Persons interested in these programs should consider obtaining the appropriate references from those listed on the preceding page.

Waldo R. Tobler
Professor of Geography
June 1970

Geographical Publications of Related Interest

B. Berry and D. Marble, Spatial Analysis, Englewood Cliffs, Prentice Hall, 1966.

William Bunge, Theoretical Geography, Second Revised and Enlarged Edition, Lund, Gleerup, 1966.

R.J. Chorley, et al, Models in Geography, New York, Barnes and Noble, 1967.

R. Chorley and P. Haggett, Network Models in Geography, London, 1969.

J. Cole and C. King, Quantative Geography, New York, J. Wiley, 1968.

W. Garrison and D. Marble, eds., Quantative Geography, Vols. 1 & 2, Evanston, Northwestern University, 1967.

P. Haggett, Locational Analysis in Human Geography, New York, St. Martin's, 1966.

L. King, Statistical Analysis in Geography, Englewood Cliffs, Prentice Hall, 1969.

L. King, editor, Geographical Analysis, a quarterly international journal of theoretical geography published by the Ohio State University Press, Columbus, Ohio.

J. Nystuen, ed., Discussion Papers, 1-12, Michigan Interuniversity Community of Mathematical Geographers, 1967; available as OP-33067 from University Microfilms.

A. Wrobel, and B. Berry, eds., Economic Regionalization and Numerical Methods, Geographia Polonica Nr. 15, Warszawa, 1968.

LIST OF PROGRAMS

Program Availability

Card decks or magnetic tape copies of the programs listed in this document (and others)are now available from the Geography Program Exchange, Computer Institute for Social Research, Michigan State University, East Lansing, Michigan, 48823, U.S.A. : telephone, 517-353-2042, attn. Prof. R. Wittick.

Disclaimer

Although the programs contained herein have been tested extensively no warranty, expressed or implied, is made by the University of Michigan or individuals as to the accuracy and functioning of the program and the related program descriptions.

DETERMINATION OF GEOGRAPHICAL ORIGINS

Purpose: The program estimates sets of source coordinates from empirical geographical distributions.

Description: The program is a simple modification of an earlier program by Casetti and Semple. The modification consists of (a) generalization of the types of trends which can be estimated, (b) addition of a contouring subroutine for automatic plotting, and (c) minor improvements the program flow and output formats. A complete description is given in the paper by Casetti and Semple, which is available. In brief, the procedure may be considered to be an inverse diffusion operator in two dimensions. The contouring subroutine is general and can be used with other programs.

Deck Make up: 1) Title Card

2) Format Card

3) Control Card

4) First Contouring Controls (optional)

5) Second Contouring Controls (optional)

6) Third Contouring Controls (optional)

7) Data Cards

Cards (1) through (7) may be repeated as many times as desired. All are read from unit 5, printing is on unit 6, and punching of output on unit 7.

Card Format: 1) Title Card

Any title in columns 1 through 72.

2) Format Card

The format for one observation consisting of a 4 character name, x coordinate, y coordinate, and weight z, in that order. The name should be specified as an A4 field, the remaining variables as E or F fields. The format is punched in columns 1 through 72.

3) Control Card (6I3)

column	
1-3	N the number of observations.
4-6	IKOT a switch to define the independent variable. 1 1.0/distance 2 Log_e (distance) 3 exp (distance) 4 distance 5 distance squared.
7-9	ILOG 1 implies take natural log of dependent variable z. 0 default, no transformation of z.
10-12	KOPT printing switch 1 print all correlation matrices 2 print correlation map 3 print both map and matrix.
13-15	IPUN.GT.0 to punch results
16-18	NN The size of the correlation matrix. Default = 11

4) First Contour Control Card

Use only if KOPT.GT.1.

Blank card yields default cases.

column

column		
1-2	CON	number of contour intervals Default = 10
3	TOUR	Type of contour interval 0 = values calculated (Default case) 1 = values specified 2 = variable interval specified 3 = standard deviation units.
4	LINES	type of contour map 0 continuous bands 1 alternate bands 2 lines
5-9	INCHES	(F5.2) width of contour map, up to 12.7 inches.

5) Second Contour Control Card

Use only if KOPT.GT.1 and TOUR = 1

column

column	
1-10	Lowest contour value (F10.0)
11-20	Highest contour value (F10.0)

6) Third Contour Control Card

Use only if KOPT.GT.1 and TOUR = 2

Up to 15 variable contour levels punched as (15F4.0).

7) Data Cards

name, x, y, z punched as described on the Format Card.

References: E. Casetti, and R.K. Semple, A Method for the Stepwise Separation of Spatial Trends, MICMOG paper #11, 1968. Available from University Microfilms in OP - 33067.

```
C     PROGRAM "GEOFIT"
C
C     THE PROGRAM ESTIMATES SOURCE COORDINATES FOR SPATIAL TRENDS
C     BASED ON MICMOG PAPER # 11,1968, BY CASETTI AND SEMPLE
C     REVISION BY L. TREVILLIAN ACCORDING TO INSTRUCTIONS FROM
C     W. TOBLER, GEOGRAPHY DEPARTMENT, UNIVERSITY OF MICHIGAN
C     READS 5, WRITES 6, PUNCHES 7
        INTEGER CON,TOUR,RX,RY,GX,GY,HX,HY
        REAL INCHES,INCON
        REAL*8 ZVA,ZAV,XAV,XVA,YAV,YVA,DAV,DVA,ZB,RAV,RVA(10)
        DIMENSION INCON(15),ORANG(25),FCN(6),TITL(18),EXVT(10)
        DIMENSION X(500),Y(500),Z(500),NAME(500),SLOPE(10),A(500),D(500),
     1 ZZ(500),ZI(500),XORG(10),YORG(10),K(50,50),H(50,50)
     2,G(50,50),SAVIT(500),CEPT(10),TIT(18),FMT(18)
        DATA TITL/'MAP OF CORRELATION COEFFICIENTS',10*' '/
C     SUBROUTINES
C        CORR COMPUTES THE CORRELATION COEFFICIENTS AND FINDS
C        THE MAXIMUM COEFFICIENTS.
C        STATS FINDS MEAN, MAX, MIN, VARIANCE OF AN ARRAY
C        COMFCN FINDS AND TRANSFORMS DISTANCES
C        SCON PRINTS CONTOUR MAP OF CORRELATIONS
C
C     INPUT VARIABLES
C      N IS THE NUMBER OF DATA POINTS
C      IKOT DEFINES THE INDEPENDANT VARIABLE
C      IF IKOT=1, 1.0/DISTANCE
C      IF IKOT=2, LOG(DISTANCE)
C      IF IKOT=3, EXP(DISTANCE)
C      IF IKOT=4, DISTANCE
C      IF IKOT=5, DISTANCE SQUARED
C     NON-ZERO ILOG YIELDS LOG TRANSFORM OF DEPENDANT VARIABLE
C      I.E., ILOG=1, IKOT=4 YIELDS EXPONENTIAL DECAY
C      I.E., ILOG=1, IKOT=5 YIELDS GAUSSIAN CURVE
C      KOPT, A SWITCH
C        IF KOPT=1, PRINT ALL CORRELATION MATRICES
C        IF KOPT=2, A CONTOUR MAP WILL BE PRINTED
C        IF KOPT =3, BOTH THE MATRIX AND THE MAP WILL BE PRINTED
C      IPUN .GT. 0 TO PUNCH RESULTS
C      NN IS THE SIZE OF THE CORRELATION MATRIX.
C
1       READ(5,507,END=999) (TIT(I),I=1,18)
        READ(5,507) (FMT(I),I=1,18)
507     FORMAT(18A4)
        READ(5,1000) N,IKOT,ILOG,KOPT,IPUN,NN
1000    FORMAT(6I3)
        WRITE(6,1000) IKOT,ILOG,KOPT,IPUN,NN,N
C
C     IF A CONTOUR MAP IS DESIRED, READ PARAMETERS NECESSARY FOR
C     CALL ON SCON. BLANK CARD YIELDS VALID DEFAULT OPTIONS.
        IF(KOPT.LE.1) GO TO 30
        READ(5,1010) CON,TOUR,LINES,INCHES
        IF(TOUR.GT.3 .OR. TOUR .LE. 0)  GO TO 30
        GO TO (10,20,30),TOUR
10      READ(5,1020) VL,VU
        GO TO 30
```

```
20      READ(5,1030) INCON
        JJJ=CON+1
        READ(5,INCON) (CRANG(I),I=1,JJJ)
C
C      NAME,X,Y, AND Z ARE THE NAME, X AND Y COORDINATES, AND HEIGHT
C      OF THE POINTS TO BE FIT.
30      READ(5,FMT) (NAME(I),X(I),Y(I),Z(I),I=1,N)
        IF (ILOG.EQ.0) GO TO 35
        WRITE(6,1050) (NAME(I),X(I),Y(I),Z(I),I=1,N)
        DO 34 I=1,N
        RA=1.0
        IF(Z(I).LT.0.) RA=-1.0
        Z(I)=Z(I)*RA
        IF (Z(I).EQ.0.0) GO TO 999
        Z(I)=ALOG(Z(I))
34      CONTINUE
        WRITE(6,512)
512     FORMAT(' LOG TRANSFORM APPLIED TO THESE DATA')
35      ITER=0
        RA=N
        EXVC=0
        IZ=1
        IF(NN.LE.0) NN=11
C
C      COMPUTE MEAN AND VARIANCE OF X,Y,AND Z
C      FIND THE MAXIMUM RANGE OF X AND OF Y FOR USE IN DETERMINING
C      THE GRID SIZE
C
        CALL STATS(N,IZ,Z,ZAV,ZVA,ZMAX,ZMIN,500,1)
        DO 40 I=1,N
        ZI(I)=Z(I)
        SAVIT(I)=ZAV
40      Z(I)=Z(I)-ZAV
        ZB=ZVA
        CALL STATS(N,IZ,X,XAV,XVA,BIGX,SMAX,500,1)
        CALL STATS(N,IZ,Y,YAV,YVA,BIGY,SMAY,500,1)
        WRITE(6,1050) (NAME(I),X(I),Y(I),ZI(I),I=1,N)
        WRITE(6,1060) XAV,YAV,ZAV,XVA,YVA,ZVA
        ZVA=CSQRT(ZVA)
        ZAV = 0.
        DX=BIGX-SMAX
        DY=BIGY-SMAY
        IF(DX-DY) 60,60,70
60      VA=DY/(NN-1.)
        GO TO 80
70      VA=DX/(NN-1.)
80      ITER=ITER+1
C
C       FIRST ITERATION.  GRID SIZE VA
C
        CALL CORR(N,Z,ZAV,ZVA,NN,VA,R,RBIG,RX,RY,X,SMAX,Y,SMAY,IKOT)
        XXX=SMAX+(RX-2.)*VA
        YYY=SMAY+(NN-RY-1.0)*VA
        IF(RX.LE.1) XXX=SMAX
        IF(RX.GE.NN) XXX=SMAX+(NN-3.)*VA
        IF(RY.LE.1) YYY=BIGY-VA*2.0
```

```
      IF(RY.GE.NN) YYY=SMAY
      HA=(2.*VA)/(NN-1.)
C
C     SECOND ITERATION.  GRID SIZE HA
C
      CALL CORR(N,Z,ZAV,ZVA,NN,HA,H,HBIG,HX,HY,X,XXX,Y,YYY,IKOT)
      XXXX= XXX+(HX-2.)*HA
      YYYY=YYY+(NN-HY-1.0)*HA
      IF(HX.LE.1)XXXX=XXX
      IF(HX.GE.NN)XXXX=XXX+(NN-3.)*HA
      IF(HY.LE.1) YYYY=YYY+(NN-3.0)*HA
      IF(HY.GE.NN) YYYY=YYY
      GA=(2.*HA)/(NN-1.)
C
C     THIRD ITERATION.  GRID SIZE GA
C
      CALL CORR(N,Z,ZAV,ZVA,NN,GA,G,GBIG,GX,GY,X,XXXX,Y,YYYY,IKOT)
C
C    FIND THE MAP COORDINATES OF MAXIMUM CORRELATION AND RECOMPUTE THE
C    FUNCTION
C
      XORG(ITER)=XXXX+((GX-1.)*GA)
      YORG(ITER)=YYYY+(NN-RY)*GA
      CALL COMFCN(X,Y,N,C,XORG(ITER),YORG(ITER),IKOT)
      BABA=0.
      SLOPE(ITER)=0.
      CALL STATS(N,IZ,D,DAV,DVA,DMAX,DMIN,500,1)
      DO 90 I=1,N
      D(I)=D(I)-DAV
      BABA=BABA+D(I)*Z(I)
90    SLOPE(ITER)=SLOPE(ITER)+D(I)*D(I)
      SLOPE(ITER)=BABA/SLOPE(ITER)
      CEPT(ITER)=ZAV-SLOPE(ITER)*DAV
C
C    FIND THE EXPLAINED VARIANCE AND CALCULATE THE RESIDUALS
C
      DO 100 I=1,N
      D(I)=SLOPE(ITER)*D(I)
      SAVIT(I)=SAVIT(I)+D(I)
      ZZ(I)=Z(I)-D(I)
100   CONTINUE
      CALL STATS(N,IZ,ZZ,RAV,RVA(ITER),RMAX,RMIN,500,1)
      CALL STATS(N,IZ,D,DAV,DVA,DMAX,DMIN,500,1)
      EXVAR=DVA
      UNEXP=RVA(ITER)
      EXVT(ITER)=DVA/ZB*100.
      EXVC =EXVC+EXVT(ITER)
C
C    SECTION TO PRINT OUT DATA AND RESULTS
C
      ZVA=ZVA*ZVA
      WRITE(6,501) ITER
      WRITE(6,500) (NAME(I),X(I),Y(I),Z(I),D(I),ZZ(I),I=1,N)
      WRITE(6,620) XAV,YAV,ZAV,DAV,RAV,XVA,YVA,ZVA,DVA,RVA(ITER)
      WRITE(6,2030) ITER
      K=1
```

```
      RY=NN-RY+1
      WRITE(6,590) K,RX,RY,RBIG,VA
      IF (KOPT-2) 520,530,520
520   WRITE(6,601)
      DO 521 I=1,NN
521   WRITE(6,600) (R(I,J),J=1,NN)
530   K=2
      HY=NN-HY+1
      WRITE(6,590) K,HX,HY,HBIG,HA
      IF(KOPT-2) 540,550,540
540   WRITE(6,601)
      DO 541 I=1,NN
541   WRITE(6,600) (H(I,J),J=1,NN)
550   K=3
      GY=NN-GY+1
      WRITE(6,590) K,GX,GY,GBIG,GA
      IF(KOPT-2) 560,570,560
560   WRITE(6,601)
      DO 561 I=1,NN
561   WRITE(6,600) (G(I,J),J=1,NN)
570   WRITE(6,610) XORG(ITER),YORG(ITER),EXVAR,UNEXP
      WRITE(6,2020) EXVT(ITER),EXVC
      IF(KOPT.EQ.1) GO TO 250
C
C     CALL SCON WITH ARGUMENT R
C
      CALL SCON(R,NN,NN,TITL,CON,TOUR,LINES,INCHES,DRANG,VL,VU,
     & SMAX,SMAY,VA)
250   CONTINUE
      I=ITER-1
      IF(I.LE.0) GO TO 209
      IF(EXVT(ITER).GT.EXVT(I)) ITER=I
      IF (ITER.EQ.I) GO TO 210
209   CONTINUE
      IF((EXVT(ITER).LE.1.).OR.(ITER.GE.5)) GO TO 210
      CALL STATS(N,IZ,ZZ,ZAV,ZVA,ZMAX,ZMIN,500,1)
      DO 200 I=1,N
200   Z(I)=ZZ(I)-ZAV
      ZAV=0.
      ZVA=DSQRT(ZVA)
      GO TO 80
210   CONTINUE
      EVEC=0.0
      WRITE(6,2000)
      I=0
      BABA=100.*RVA(1)/(100.-EXVT(1))
      WRITE(6,1999) I,BABA,EVEC,EVEC
      DO 220 I=1,ITER
      EVEC=EVEC+EXVT(I)
220   WRITE(6,2010) I,XORG(I),YORG(I),RVA(I),EXVT(I),EVEC,CEPT(I),
     & SLOPE(I)
      WRITE(6,630)
      IF (ILOG.NE.0) WRITE(6,640)
      GO TO (631,632,633,634,635),IKOT
631   WRITE(6,641)
      GO TO 650
```

```
632    WRITE(6,642)
       GO TO 650
633    WRITE(6,643)
       GO TO 650
634    WRITE(6,644)
       GO TO 650
635    WRITE(6,645)
650    CONTINUE
       WRITE(6,508) (TIT(I),I=1,18)
       WRITE(6,505) ITER
505    FORMAT(' THE FINAL RESULTS OF THE ' I2,' ITERATIONS ARE:'/)
       DO 300 I=1,N
       IF (ILOG.EQ.0) GO TO 299
       ZI(I)=EXP(ZI(I))
       SAVIT(I)=EXP(SAVIT(I))
299    ZZ(I)=ZI(I)-SAVIT(I)
       IF (IPUN.GT.0) WRITE(7,506) NAME(I),X(I),Y(I),ZI(I),SAVIT(I),
     & ZZ(I)
300    CONTINUE
       WRITE(6,500)(NAME(I),X(I),Y(I),ZI(I),SAVIT(I),ZZ(I),I=1,N)
500    FORMAT('0',48X,'DATA'/'ONAME',15X,'X',15X,'Y',15X,'Z',11X,
     1 'APPROX',6X,'RESIDUALS'//(' ',A4,10X,5(F10.4,5X)))
501    FORMAT('1THE RESULTS OF PASS NUMBER ',I2,' ARE AS FOLLOWS:')
506    FORMAT(' ',A4,5(F10.4,1X))
508    FORMAT('1',18A4,//)
590    FORMAT('0MAXIMUM CORRELATION ON ITERATION ',I1,
     1 ' OCCURRED AT ',I2,',',I2/' THE VALUE OF THE CORRELATION ',
     1 'COEFFICIENT WAS ',F8.5/' THE GRID SIZE WAS  ',F10.5)
600    FORMAT(20(' ',F6.4))
601    FORMAT('0MATRIX OF CORRELATION COEFFICIENTS'//)
610    FORMAT(//'0THE MAP COORDINATES OF HIGH CORRELATION ARE '
     1 ,2(F10.5,2X)/' THE EXPLAINED VARIANCE IS    ',E11.4/
     2 ' THE UNEXPLAINED VARIANCE IS  ',E11.4)
620    FORMAT('0MEAN',10X,5(E11.4,4X)/' VARIANCE',6X,5(E11.4,4X))
630    FORMAT(///' THE FUNCTION IS '//)
640    FORMAT(' BASE E LOGARITHM OF ')
641    FORMAT(' Z= A + B*(1.0/DISTANCE) ')
642    FORMAT(' Z = A + B*LOGE(DISTANCE) ')
643    FORMAT(' Z = A + B*EXP(DISTANCE) ')
644    FORMAT(' Z = A + B*DISTANCE ' )
645    FORMAT(' Z = A + B*(DISTANCE**2) ')
1010   FORMAT(I2,2I1,F5.2)
1020   FORMAT(2F10.0)
1030   FORMAT(15A4)
1050   FORMAT('1',24X,'ORIGINAL DATA'//' NAME',17X,'X',16X,'Y',16X,'Z'/
     &//(' ',A4,5X,3(5X,F12.4)))
1060   FORMAT('OMEAN',5X,3(5X,E12.5)/' VARIANCE ',3(5X,E12.5))
1999   FORMAT(5X,I1,36X,E12.5,5X,2(F7.3,8X))
2000   FORMAT('1SUMMARY'//71X,'CUMULATIVE'/' ITERATION',13X,'ORIGIN',
     & 15X,'RESIDUAL',6X,'EXPLAINED',5X,'EXPLAINED',5X,'CONSTANTS'/
     & '  NUMBER',11X,'X',12X,'Y',11X,'VARIANCE',5X,'VARIANCE(%)',
     & 3X,'VARIANCE(%)'//)
2010   FORMAT(5X,I1,7X,F10.4,5X,F10.4,4X,E12.5,5X,2(F7.3,8X),E12.5,
     & 2X,E12.5)
2020   FORMAT('0THE PERCENTAGE OF THE TOTAL VARIANCE EXPLAINED ON THIS'
     & ,' ITERATION IS ',F6.3,'%.'/' THE CUMULATIVE EXPLAINED VARIANCE'
```

```
     & ,' IS ',F6.3,'%.')
2030  FORMAT('1THE RESULTS OF THE CORRELATION SUBPROGRAM FOR PASS ',
     & 'NUMBER ',I2,' ARE AS FOLLOWS:')
999   CONTINUE
      WRITE(6,508) (TIT(I),I=1,18)
      GO TO 1
      END
      SUBROUTINE STATS(IROWS,ICOLS,Z,ZBAR,ZSIG,ZMAX,ZMIN,NDIMX,NDIMY)
      REAL*8 ZBAR,ZSIG,TOT
      REAL Z(NDIMX,NDIMY)
      ZMAX=Z(1,1)
      ZMIN=Z(1,1)
      DO 20 I=1,IROWS
      DO 20 J=1,ICOLS
      ZMAX=AMAX1(ZMAX,Z(I,J))
20    ZMIN=AMIN1(ZMIN,Z(I,J))
      ZBAR=0.
      ZSIG=0.
      DO 10 I=1,IROWS
      DO 10 J=1,ICOLS
      ZBAR=ZBAR+Z(I,J)
10    ZSIG=ZSIG+Z(I,J)*Z(I,J)
      TOT=IROWS*ICOLS
      ZBAR=ZBAR/TOT
      ZSIG=ZSIG/TOT-ZBAR*ZBAR
      RETURN
      END
      SUBROUTINE COMFCN(X,Y,N,A,XX,YY,IKOT)
C     THIS SUBROUTINE CALCULATES DISTANCES ,
C     AND APPLIES TRANSFORMATIONS TO THESE DISTANCES
C     COMPUTATION OF DIRECTION AT THIS POINT WOULD
C     ALLOW CONSIDERATION OF ANISOTROPIC TRENDS
      DIMENSION X(500),Y(500),A(500)
      DATA EPS /0.1/
      DO 10 I=1,N
10    A(I)=(X(I)-XX)**2 +(Y(I)-YY)**2
      IF (IKOT.EQ.5) GO TO 60
      GO TO (20,40,51,55,60),IKOT
20    CONTINUE
      DO 30 I=1,N
      IF (A(I).LE.EPS) A(I)=1.0
30    A(I)=1.0/SQRT(A(I))
      GO TO 60
40    CONTINUE
      DO 50  I=1,N
      IF (A(I).LE.EPS) A(I)=1.0
50    A(I)=ALOG(SQRT(A(I)))
      GO TO 60
51    CONTINUE
      DO 52 I=1,N
52    A(I)= 1.0 *EXP(A(I))
      GO TO 60
55    DO 56 I=1,N
56    A(I)=SQRT(A(I))
60    CONTINUE
      RETURN
```

```
      END
      SUBROUTINE CORR(N,Z,ZAV,ZVA,NN,VA,R,RBIG,RX,RY,X,
     & XMIN,Y,YMIN,IKOT)
      DIMENSION R(100,100),A(500),Z(500),X(500),Y(500)
      INTEGER RX,RY
      DOUBLE PRECISION ZVA,ZAV,AVA,AAV
C
C
C     CORR COMPUTES A MATRIX OF CORRELATION COEFFICIENTS AND FINDS THE
C     MAXIMUM OF THE COEFFICIENTS
C
C     A FUNCTION OF THE DISTANCES BETWEEN POINTS ON A GRID (DEFINED
C     BY XMIN,YMIN, AND VA) AND THE COORDINATES OF THE Z VECTOR
C     IS COMPUTED AND CORRELATED WITH THE ACTUAL VALUES OF Z.
C
      XX=XMIN
      YY=YMIN
C
C     N IS THE NUMBER OF DATA POINTS IN THE VECTOR,Z, WHICH IS TO BE
C     CORRELATED WITH THE VECTOR A.  ZAV AND ZVA ARE THE MEAN AND
C     STANDARD DEVIATION OF Z.
C     NN IS THE NUMBER OF ROWS(OR COLUMNS) OF THE COEFFICIENT MATRIX,R.
C     RBIG IS THE LARGEST CORRELATION COEFFICIENT, THE COORDINATES
C     OF RBIG ARE RX AND RY.
C     X AND Y ARE THE COORDINATES OF THE POINTS IN THE Z VECTOR.
C     XMIN AND YMIN ARE THE MINIMUM VALUES OF THESE COORDINATES.
C     IKOT DEFINES THE FUNCTION TO BE USED IN THE FIT.
C     SCALE IS A CONSTANT SCALE FACTOR.
C     VA IS THE GRID SIZE.
C
      RA=N
      DO 90 LL=1,NN
      L=NN-LL+1
      DO 80  M=1,NN
C
C     COMPUTE THE DISTANCES BETWEEN THE Z COORDINATES AND (XX,YY)
C
C
C     COMPUTE THE FUNCTION OF THE DISTANCES AS SPECIFIED BY IKOT
C
      CALL COMFCN(X,Y,N,A,XX,YY,IKOT)
60    IAV=1
C
C     FIND THE MEAN AND VARIANCE OF THE FUNCTIONAL POINTS
C
      CALL STATS(N,IAV,A,AAV,AVA,AMAX,ZMIN,500,1)
      DO 65 I=1,N
65    A(I)=A(I)-AAV
      AAV=0.
      AVA=DSQRT(AVA)
      R(L,M)=0.
C
C     FIND THE CORRELATION BETWEEN THE FUNCTIONAL AND THE OBSERVED VALUES
C
      DO 70 I=1,N
70    R(L,M)=R(L,M)+A(I)*Z(I)
```

```
          R(L,M)=R(L,M)/(RA*AVA*ZVA)
C
C     INCREMENT THE GRID VALUES
C
80        XX=XMIN+M*VA
          XX=XMIN
90        YY=YMIN+LL*VA
C
C     FIND THE MAXIMUM COEFFICIENT AND ITS COORDINATES
C
          RBIG=ABS(R(1,1))
          RX=1
          RY=1
          DO 110 I=1,NN
          DO 110 J=1,NN
          IF(ABS(R(I,J))-RBIG) 110,110,100
100       RBIG=ABS(R(I,J))
          RX=J
          RY=I
110       CONTINUE
          RBIG=R(RY,RX)
          RETURN
          END
          SUBROUTINE SCON(Z,ROWS,COLS,TITL,CON,TOUR,LINES,INCHES,DRANG,
        & ZMIN,ZMAX,XMIN,YMIN,VA)
          DIMENSION FORM(10),TITL(18)   ,SYMTAB(21),PSYM(131),Z(100,100),
        1 INCON(15),DRANG(25),TABE(25),ZPREV(131),SYM(131)
          DIMENSION CUMH(27)
          REAL*8 ZBAR,ZSIG
          INTEGER ROWS,COLS,CON,TOUR,COLMAX,COL,PSYM,PS,GRID,TOUT,TLOW,
        1 SYM,BLANK,PLUS,MINUS,STAR,SYMTAB
          INTEGER XSTOP,YSTOP
          REAL MINZ,MAXZ,INCHES,INCON
          DATA SYMTAB/'1','2','3','4','5','6','7','8','9','A','B','C',
        1 'D','E','F','G','H','I','J','K'/,BLANK,PLUS,MINUS,STAR/' ',
        2 '+','-','*'/
C     ******************************************
C     ZERO COUNTERS AND ARRAYS USED IN FREQUENCY TABLE
C     ******************************************
          WRITE(6,1290)
1290      FORMAT('1THE CONTOUR MAP WILL BE DEVELOPED FROM THE   ',
        & 'FOLLOWING DATA'///)
          DO 4 I=1,ROWS
4         WRITE(6,1300) (Z(I,J),J=1,COLS)
1300      FORMAT(15(' ',F7.4))
          WRITE(6,1241) XMIN,YMIN,VA
1241      FORMAT(///' THE RELATIVE ORIGIN IS ',F10.4,' , ',F10.4,' .'/
        & 'THE GRID SIZE IS ',F10.4,' .')
          JJ=0
          MM=0
          SUM=0.0
          TLOW=0
          TOUT=0
          DO 5 I=1,25
5         TABE(I)=0.0
          N=ROWS*COLS
```

```
C     ****************************************
C      N = NUMBER OF DATA POINTS
C     ****************************************
      WRITE(6,1030) TITL,N,COLS,ROWS
1030   FORMAT(1H1,18A4//' MAP IS DEVELOPED FROM',I5,' GRIDDED VALUES',
     1 ' CONSISTING OF ',I3,' COLUMNS BY ',I3,' ROWS.'//)
       CALL STATS(ROWS,COLS,Z,ZBAR,ZSIG,MAXZ,MINZ,100,100)
       ZSIG=DSQRT(ZSIG)
C     ****************************************
C   DETERMINE CONTOUR LINE ELEVATIONS
C   TOUR VALUE DETERMINES METHOD USED IN DEFINING CONTOURS
C     ****************************************
      IF (TOUR .GT. 3 .OR. TOUR .LE. 0) GO TO 60
      GO TO (30,40,50),TOUR
30    WRITE (6,1060)
1060  FORMAT (25H CONTOUR LEVELS SPECIFIED)
      GO TO 70
40     WRITE(6,1080)
1080  FORMAT (37H VARIABLE CONTOUR INTERVALS SPECIFIED)
      ZMIN=CRANG(1)
      ZMAX=CRANG(CON)
      GO TO 70
50    ZMIN=ZBAR-3.0*ZSIG
      ZMAX=ZBAR+3.0*ZSIG
      CON=13
      WRITE (6,1090)
1090   FORMAT(30H CONTOUR INTERVAL = 0.5 SIGMA
      GO TO 70
60    WRITE (6,1100)
1100  FORMAT (26H CONTOUR LEVELS CALCULATED)
       ZMAX=MAXZ
      ZMIN=MINZ
70    WRITE(6,1110) ROWS,COLS,ZBAR,ZSIG,MAXZ,MINZ
1110  FORMAT (13H GRIDDED DATA/,42H LINEAR INTERPOLATION WITHIN GRID SQU
     1ARES /,17H OBSERVED VALUES /,  8H ROWS = ,I3,10H   COLS = ,
     2I3,/,8H ZBAR = ,F10.3,10H   ZSIG = ,F10.3,10H   MAXZ = ,F10.3,10H
     3  MINZ = ,F10.3)
C     ****************************************
C   CALCULATE MAP WIDTH IN CHARACTERS (COL)
C     ****************************************
       IF (INCHES.LE.0.0) COL=100
       IF (INCHES .GT. 12.7) COL=127
       IF (INCHES.GT.0.0.AND.INCHES.LE.12.7) COL=INCHES*10.0
      PS=(COL-COLS)/(COLS-1)
C     ****************************************
C   TEST GRID SIZE--IF GRID SMALLER THAN 2 CHARACTERS BETWEEN COLUMNS,
C   AN ERROR STATEMENT IS PRINTED AND PLOT ABORTED
C     ****************************************
       IF (PS.GT.1) GO TO 80
      WRITE (6,1120)
1120  FORMAT (25H TOO MANY COLUMNS IN GRID)
      GO TO 999
C
C   CALCULATE REQUIRED CONSTANTS
C
80     COLMAX=PS*(COLS-1)+COLS
```

```
        YINC=1.0/((PS+1.0)*0.6)
        GRID=PS+1
        XINC=1.0/GRID
        RANGZ=ZMAX-ZMIN
        RHIGH=ZMAX+RANGZ
        RLOW=ZMIN-RANGZ
        IF (CON.EQ.0.OR.CON.GT.19) CON=5
        FCON=CON
        CI=RANGZ/FCON
        IF (TOUR.EQ.2) GO TO 100
        JJJ=CON+1
        DO 90 I=1,JJJ
90      DRANG(I)=ZMIN+(I-1)*CI
100     CONTINUE
        WRITE (6,1130)
1130    FORMAT (1H /,16H VALUES EMPLOYED)
        WRITE (6,1140) ZMIN,ZMAX,CI,GRID,XINC,YINC,CON,PS,COLMAX
1140    FORMAT (8H ZMIN = ,F10.3,10H  ZMAX = ,F10.3,8H   CI = ,F10.3/,8H
       1GRID = ,I5,10H   XINC = ,F10.3,10H   YINC = ,F10.3,/,7H CON = I5,
       2 8H   PS = ,I5,12H   COLMAX = ,I5)
C       ****************************************
C     WRITE OUT CONTOUR SYMBOL TABLE
C       ****************************************
        WRITE (6,1150)
1150    FORMAT (35H4CONTOUR SYMBOL TABLE IS AS FOLLOWS)
        IF (LINES.EQ.1) WRITE (6,1160)
1160    FORMAT (43H0(ONLY EVEN SYMBOLS PRINTED IN BODY OF MAP) )
        WRITE (6,1170)
1170    FORMAT (7HOSYMBOL,11X,5HVALUE)
        IF (LINES.NE.0) GO TO 110
        WRITE (6,1180)
1180    FORMAT (5H0   *,6X,7HNO DATA,/,5H    -,4X,10HBELOW  LOW)
        WRITE (6,1190) (SYMTAB(I),DRANG(I),I=1,CON)
1190    FORMAT (4X,A1,2X,F12.4)
        WRITE (6,1200)
1200    FORMAT (4X,1H+,4X,10HABOVE HIGH)
        GO TO 120
110     WRITE (1,1210)
1210    FORMAT(5H0   *,25X,7HNO DATA,/,5H    -,23X,10HBELOW  LOW)
        WRITE (6,1220) (SYMTAB(I),DRANG(I),DRANG(I+1),I=1,CON)
1220    FORMAT (4X,A1,2X,F12.4,6H  TO  ,F12.4)
        WRITE (6,1230) DRANG(CON+1)
1230    FORMAT (4X,1H+,2X,F12.4,10H  OR  OVER)
120     WRITE(6,1030) TITL,N,COLS,ROWS
        XSTOP = GRID
        GRID=COLS-1
        COL=3
        DO 140 J=1,GRID
        DO 130 I = 2,XSTOP
        SYM(COL)=BLANK
130     COL=COL+1
        SYM(COL)=PLUS
140     COL=COL+1
        N=COLMAX+2
        SYM(2)=PLUS
        SYM(1)=BLANK
```

```
      SYM(N)=BLANK
      WRITE (6,1240) (SYM(I),I=1,N)
1240  FORMAT (1H9,129A1)
C
C   ESTIMATE VALUE OF ALL PRINTING POSITIONS ON MAP
C
C     LOOP THROUGH MAP LENGTH, ROW BY ROW
C
      ITOP=1
      IROW = ROWS + 1
      DO 340 MM = 2,IROW
      YSTOP = 1./YINC + .5
      IF (MM.EQ.IROW) YSTOP=1
      M=MM-1
145   YP=0.0
      PSYM(1) = PLUS
      PSYM(N) = PLUS
      DO 340 KJ=1,YSTOP
150   COL=1
      Z1 = YP*(Z(MM,1)-Z(M,1))+Z(M,1)
C
C     LOOP THROUGH MAP WIDTH,COLUMN BY COLUMN
C
190   II=COLS-1
      JLFT=2
      DO 330 J=1,II
      JJ=J+1
      Z2=YP*(Z(MM,JJ)-Z(M,JJ))+Z(M,JJ)
      ZDELT=Z2-Z1
      XP=0.0
      DO 480 KK = 1,XSTOP
      JLFT=JLFT-1
195   COL=COL+1
200   ZEST=XP*ZDELT+Z1
      IF(ZEST.GE.ZMIN.AND.ZEST.LE.ZMAX) GO TO 325
      IF(ZEST.GT.ZMAX) GO TO 203
      IF (ZEST.LE.RLOW) GO TO 206
      IF(ITOP.EQ.1.OR.MM.EQ.IROW.OR.JLFT.EQ.1) GO TO201
      PSYM(COL)=BLANK
      GO TO 202
201   PSYM(COL)=MINUS
202   TLOW=TLOW+1
      ZPREV(COL)=ZEST
      ZLAST=ZEST
      GO TO 124
203   IF(ZEST.GE.RHIGH) GO TO 206
      IF(ITOP.EQ.1.OR.MM.EQ.IROW.OR.JLFT.EQ.1) GO TO204
      PSYM(COL)=BLANK
      GO TO 205
204   PSYM(COL)=PLUS
205   TABE(CON+1)=TABE(CON+1)+1.0
      ZPREV(COL)=ZEST
      ZLAST=ZEST
      GO TO 124
206   IF(ITOP.EQ.1.OR.MM.EQ.IROW.OR.JLFT.EQ.1) GO TO207
      PSYM(COL)=BLANK
```

```
      GO TO 208
207   PSYM(COL)=STAR
208   IOUT=IOUT+1
      ZPREV(COL)=ZEST
      ZLAST=ZEST
      GO TO 480
325    CONTINUE
       IF(ICUP.NE.2) GO TO 312
       DO 311 LMN=2,JJJ
       I=LMN-1
       IF(ZEST.LT.DRANG(LMN)) GO TO 313
311    CONTINUE
312    I=((ZEST-ZMIN)/CI)+1.
313    TABE(I)=TABE(I)+1.
      LVAL=LINES+1
      IF(ITCP.EQ.1.OR.MM.EQ.IROW.OR.JLFT.EQ.1)LVAL=3
      GO TO (326,327,328) ,LVAL
326   DO 309 I=1,CON
      ZSTEP=DRANG(I)
      SW1=1.
      SW2=1.
      SW3=1.
      SW4=1.
      IF (ZSTEP.GE.ZLAST.OR.ZSTEP.GE.ZPREV(COL)) SW1=0
      IF (ZSTEP.LE.ZEST) SW2=0.
      IF (ZSTEP.LT.ZLAST.OR.ZSTEP.LT.ZPREV(COL)) SW3=0.
      IF (ZSTEP.GE.ZEST) SW4=0.
      IF((SW1.EQ.0..AND.SW2.EQ.0.).OR.(SW3.EQ.0..AND.SW4.EQ.0.))GO TO310
309   CONTINUE
      PSYM(COL)=BLANK
      GO TO 320
310   PSYM(COL)=SYMTAB(I)
320   ZLAST=ZEST
      ZPREV(COL)=ZEST
      GO TO 124
327   K=I/2
      IF((K*2).NE.I) PSYM(COL)=SYMTAB(I)
      IF((K*2).EQ.I) PSYM(COL)=BLANK
      GO TO 124
328   PSYM(COL)=SYMTAB(I)
      ZPREV(COL)=ZEST
124   SUM=SUM+1.0
  480 XP=XP+XINC
      Z1=Z2
330   JLFT=0
      COL = COL + 1
      IF(Z(M,COLS).GE.ZMIN.AND.Z(M,COLS).LE.ZMAX) GO TO 333
      IF(Z(M,COLS).GT.ZMAX) GO TO 335
      IF(Z(M,COLS).LE.RLOW) GO TO 336
      PSYM(COL)=MINUS
      TLOW=TLOW+1
      GO TO 334
335   IF(Z(M,COLS).GE.RHIGH) GO TO 336
      PSYM(COL)=PLUS
      TABE(CON+1)=TABE(CON+1)+1.0
      GO TO 334
```

```
336    PSYM(COL)=STAR
       TOUT=TOUT+1
       GC TO 332
333     CONTINUE
        IF(TOUR.NE.2) GO TO 338
        DO 337 LMN=2,JJJ
        I=LMN-1
        IF(Z(M,COLS).LT.CRANG(LMN)) GO TO 339
337     CONTINUE
338     I=((Z(M,COLS)-ZMIN)/CI)+1.
339     TABE(I)=TABE(I)+1.
       PSYM(COL)=SYMTAB(I)
334    SUM=SUM+1.0
332    WRITE (6,1240) (PSYM(L),L=1,N)
        YP=YP+YINC
       PSYM(1) = BLANK
       PSYM(N)=BLANK
340    ITOP=0
       WRITE (6,1240) (SYM(L),L=1,N)
C     ****************************************
C    CALCULATE AND PRINT FREQUENCY TABLE
C     ****************************************
       WRITE (6,1250) TOUT
1250   FORMAT (16H1FREQUENCY TABLE,/,12X,8HINTERVAL ,16X,5HUNITS ,6X,10HP
      1ERCENTAGE ,4X,10HCUMULATIVE ,/,16X,7HNO DATA ,11X,I8)
        PTAB=(TLOW/SUM)*100.0
       CTAB=PTAB
       WRITE (6,1260) TLOW,PTAB,CTAB
1260   FORMAT (14X,9HBELOW LOW,11X,I8,2(3X,F12.4))
       DO 350 I=1,CON
       VALUE=DRANG(I)
       VAL1=CRANG(I+1)
       VAL1=CRANG(I+1)
       PTAB=(TABE(I)/SUM)*100.0
       CTAB=CTAB+PTAB
350    WRITE (6,1270) VALUE,VAL1,TABE(I),PTAB,CTAB
1270   FORMAT (1X,F12.4,6H  TO  ,F12.4,3X,F8.0,2(3X,F12.4))
       I=CON+1
       PTAB=(TABE(I)/SUM)*100.0
       CTAB=PTAB+CTAB
       WRITE (6,1280) VAL1,TABE(I),PTAB,CTAB,SUM,CTAB,CTAB
1280   FORMAT (1X,F12.4,10H  OR  OVER ,11X,F8.0,2(3X,F12.4),/,18X,5HTOTAL
      1,11X,F8.0,2(3X,F12.4))
999    CONTINUE
       RETURN
       END
```

PLOTTING OF BIVARIATE STANDARD DEVIATIONS

Purpose: This program may be used to provide some summary measures of geographical distributions.

Description: The program first calculates the weighted mean center of the input point distribution:

$$\bar{X} = \frac{\sum X_i W_i}{\sum W_i} \qquad \text{and} \qquad \bar{Y} = \frac{\sum Y_i W_i}{\sum W_i}$$

where $\sum W_i$ is the sum of weights (all weights may be one) and X_i and Y_i are the coordinates of the respective points. The variance, standard deviation, and standard distance are then calculated in the normal statistical manner. The eigenvalues of the covariance matrix are the squares of the semi-major and semi-minor axes of the standard deviational ellipse. The counterclockwise angle of inclination of the major axis from East-West is given, as well as the total area of the ellipse. The coefficient of circularity (the ratio of the minor axis to the major axis) is also calculated. The ellipse is then plotted; dots are used for the center and end points of the principle axes. The data are plotted as circles with radii given by $r = aW^b$, where r is the radius of the circle, W is the weight while a and b are empirically defined coefficients. For the United States if W represents population then a = 0.0219 and b = 0.44, and these are the values used in the program.

Comments: The program is currently dimensioned to plot up to 2000 observations per run. This may be changed by altering the value of X in the dimension statement. Multiple data sets may be run by repeating the data deck. The program reads the observation deck from unit 4, reads controls from 5, writes onto 6, and plots on 9. Output is planned for 30" CALCOMP plotter and standard CALCOMP plotting library is assumed. The origin for the X,Y coordinates is, as usual, defined to be the Southwest (lower left) of the study area. This program is designed so that it may be easily used at a remote terminal which has access to shared files.

Data Deck:
1) Control Card
2) Title Card
3) Variable Format Card
4) Observation Deck

Data Card Composition:

1) <u>Control Card</u>

column

1-4	NO	Number of observation (≤2000) I4
5-11	XMIN	Minimum X value (F7.2)
12-18	XMAX	Maximum X value (F7.2)
19-25	YMIN	Minimum Y value (F7.2)
26-32	YMAX	Maximum Y value (F7.2)
33-39	SCAL	Denominator of the representative fraction of the map scale (F12.0)

2) <u>Title Card</u>

Any title (≤ 72 characters)

3) <u>Variable Format Card</u>

E or F type FORTRAN format for one observation.

4) <u>Observation Deck</u>

Observations as described by the variable format card.

The input data must be: X coordinate, Y coordinate, Weight.

Programmer: Waldo R. Tobler

References: Lefever, D. Weltz, "Measuring Geographic Concentration by Means of the Standard Deviational Ellipse," American Journal of Sociology, Vol. 32, #1, July 1926, pp. 88-94.

Lee, Douglass B., Jr., Analysis of Residential Segregation, Division of Urban Studies, Center for Housing and Environmental Studies, Cornell University, Ithaca, New York, 1966.

```
C STANDARD DEVIATIONAL ELLIPSES
C CONCATENATE WITH *PLOTSYS
C TO SKIP PLOTTING SET 9=*DUMMY*
      DIMENSION X(3,2000),FMT(18),TITL(18),S(3),B(3),DEV(3),SX(3),
     1 SX2(3),S2(3)
 1000 FORMAT(18A4)
 1001 FORMAT('1',18A4)
1002  FORMAT(18A4)
1003  FORMAT(2X,18A4)
1004  FORMAT(2X,I4,3(F10.3,2X))
1005  FORMAT('1 SUMS, MEANS, SUMS OF SQUARES ')
1006  FORMAT(2X,I4,3(2X,F10.3))
1007  FORMAT(2X, ' VARIANCES, STANDARD DEVIATIONS')
1008  FORMAT(2X,I4,2X,F10.2,2X,F10.4)
1009  FORMAT(2X, ' STANDARD DISTANCE = ', F10.4)
1010  FORMAT(2X,' NO OF OBSERVATIONS, THEN EIGENVALUES')
1011  FORMAT(2X,'NO OBSNS, ANGLE, AREA ')
 1012 FORMAT(I4,4F7.2,F12.0)
 1013 FORMAT('NO=',I4,'XMIN=',F7.2,'XMAX=',F7.2,'YMIN=',F7.2,'YMAX=',
     1 F7.2,'SCAL=',F12.0)
1014  FORMAT(1H1'OBSERVATIONS'/)
1016  FORMAT(' NOBS, END POINTS OF PRINCIPAL AXES')
 1017 FORMAT('COEFFICIENT OF CIRCULARITY =',F15.8,'(I.E. 1 / ECCENTRICIT
     1Y)')
C NECESSARY CONSTANTS
      RAD=174532925E-10
      PI=3.1415926
      PI2=PI/2.0
      SIZE=0.1
      THO=0.0
       THF=360.
      DL=0.0
       SC=1.0
      PY=1.0
      ALPH=0.0219
      BETA=0.44
    1 READ (5,1012) NO,XMIN,XMAX,YMIN,YMAX,SCAL
      WRITE (6,1013) NO,XMIN,XMAX,YMIN,YMAX,SCAL


      DO 2 I=1,3
      S(I)=0.0
      B(I)=0.0
      S2(I)=0.0
      SX(I)=0.0
      SX2(I)=0.0
2     CONTINUE
      PX=1.0
      RHO=0.01*SCAL
      READ (5,1000) TITL
      WRITE(6,1001) TITL
      READ (5,1002) FMT
      WRITE (6,1003) FMT
      FNO=NO*1.0
      WRITE (6,1014)
```

```
      DO 3 I=1,NO
      READ(4,FMT) X(1,I),X(2,I),X(3,I)
      IF(X(3,I)) 5,5,6
5     X(3,I)=1.0
6     S(1)=S(1)+X(1,I)*X(3,I)
      S(2)=S(2)+X(2,I)*X(3,I)
      S(3)=S(3)+X(3,I)
      WRITE(6,1004) I,X(1,I),X(2,I),X(3,I)
3     CONTINUE
      B(1)=S(1)/S(3)
      B(2)=S(2)/S(3)
      B(3)=S(3)/FNO
      SXY=0.0
      DO 7 I=1,NO
      W=X(3,I)
      DO 8 J=1,2
      DEV(J)=X(J,I)-B(J)
      SX(J)=SX(J)+DEV(J)*W
8     SX2(J)=SX2(J)+DEV(J)*DEV(J)*W
7     SXY=SXY+DEV(1)*DEV(2)*W
      SX2(3)=SXY
      WRITE(6,1005)
      DO 9 I=1,3
9     WRITE (6,1006) I,S(I),B(I),SX2(I)
      WRITE(6,1007)
      DO 10 I=1,3
      S2(I)=SX2(I)/S(3)
      S(I)=SQRT(ABS(S2(I)))
10    WRITE(6,1008) I,S2(I),S(I)
      BB=S2(1)+S2(2)
      SDIS=SQRT(BB)
      WRITE(6,1009) SDIS
      C=-(S2(1)*S2(2)-S2(3)*S2(3))
      ROOT=SQRT(ABS(BB*BB+C*4.0))
      E1=0.5*(BB+ROOT)
      E2=0.5*(BB-ROOT)
      WRITE(6,1010)
      WRITE(6,1008) NO,E1,E2
      E1=SQRT(E1)
      E2=SQRT(E2)
      WRITE (6,1008) NO, E1,E2
      AREA=E1*E2*FI
      DEN=SX2(1)-SX2(2)
      ANG=0.5*ATAN(2.0*SX2(3)/DEN)
      IF(DEN)30,30,32
30    ANG=ANG+FI2
32    WRITE(6,1016)
      IF(ANG-FI) 34,34,33
33    ANG=ANG-FI
34    ANG2=ANG+FI2
C CALCULATE END POINTS
      XA=B(1)+E1*COS(ANG)
      YA=B(2)+E1*SIN(ANG)
      WRITE(6,1008) NO,XA,YA
      XB=B(1)+E2*COS(ANG2)
      YB=B(2)+E2*SIN(ANG2)
```

```
      WRITE(6,1008) NO,XB,YB
      ANG=ANG/RAD
      WRITE(6,1011)
      WRITE(6,1008) NO,ANG,AREA
      EXCEN = E2/E1
      WRITE(6,1017) EXCEN
      WRITE(6,1001) NO,(TITL(I),I=1,17)
C PLOT STANDARD DEVIATIONAL ELLIPSES
      CALL PLTXMX (30.0)
      CALL PLTOFS(XMIN,SCAL,YMIN,SCAL,1.0,3.0)
      CALL PELIPS(B(1),B(2),E1,E2,ANG,THO,THF,SC)
      CALL PSYMB (B(1),B(2),0.1,3,ANG,-1,SC)
      CALL PCIRCL(XA , YA , THO,THF,RHO,RHO,DL,SC)
      CALL PCIRCL(XB,YB,    THO,THF,RHO,RHO,DL,SC)
      CALL PCIRCL (XMIN,YMAX, THO,THF,RHO,RHO,DL,SC)
      CALL PCIRCL (XMAX,YMAX, THO,THF,RHO,RHO,DL,SC)
      CALL PCIRCL (XMAX,YMIN, THO,THF,RHO,RHO,DL,SC)
      CALL PCIRCL (XMIN,YMIN, THO,THF,RHO,RHO,DL,SC)
C PLOT OBSERVATIONS
      DO 20 I=1,NO
      X1=X(1,I)
      Y1=X(2,I)
      RHO=ALPH*(X(3,I)**BETA)
      IF ((RHO/SCAL).LT.0.01) RHO=0.01*SCAL
   20 CALL PCIRCL(X1,Y1,THO,THF,RHO,RHO,DL,SC)
C PLOT TITLES
      CALL PSYMB(PX,PY,SIZE,'GEOGRAPHY DEPARTMENT UNIVERSITY OF MICHIGAN
     1 1968   ',THO,59)
      PX=PX+6.0
      CALL PSYMB(PX,PY,SIZE,TITL,THO,32)
      CALL PLTEND
      GO TO 1
      END
  20    0.0  420.0     0.0  280.0          40.0
020 CENTROID     DETROIT   1837
(47X, F3.0, 1X, F3.0, T70, F1.0, T71)
1837                                                     225 155
1837                                                     225 145
1837                                                     235 155
1837                                                     235 145
1837                                                     235 135
1837                                                     235 125
1937                                                     235 115
1837                                                     245 165
1837                                                     245 155
1937                                                     245 145
1837                                                     245 135
1837                                                     245 125
1837                                                     245 115
1837                                                     245 105
1837                                                     255 135
1837                                                     255 125
1837                                                     265 125
1830                                                     255 115
1830                                                     255 105
1830                                                     265 115
```

PELTO'S D-FUNCTION AND RELATIVE ENTROPY

Purpose: This program calculates two measures of the degree of mixing in multicomponent systems.

Description: If one assumes that each observation consists of V variables, then the procedure is restricted to those situations in which it is meaningful to make the assumption that the totality of variables, for each observation, sum to 100 percent.

The D-function measures the strength of dominance of variables of components; tho lower the D value (and closer to the theoretical minimum), the stronger the dominance. The minimum of the D-function is 100 x (1 - (1/CLASS)). The relative entropy measure may be taken to be an indicator of mixing of the components and would be 100 when all components appear in equal proportions. Since entropy may be taken as a measure of mixing (uncertainty), zones of high entropy should form boundaries between regions, whereas zones of low entropy should form regional areas. Relative entropy (H_r) is calculated by the following formula and expressed as a percent:

$$100\ H_r = \frac{-100\ P_i \log P_i}{H_m}$$

where P_i are the individual proportions and H_m is the theoretical maximum for the entropy. This of course assumes that the probability of occurance is the same for all components.

Comments: The program will handle up to 999 observations for each set of data. Since each observation is processed individually, data sets larger than 999 may be processed by splitting the set and run separately. Many sets of data may be run by repeating the data deck. As written the program will process observations of up to 20 components. This may be expanded by increasing the size of the arrays of D, P, MEMBER, RD, and PD. Their sizes should always be the number of components + 5. The components need not be in proportions; these will be calculated by the program. The results for each observation may be punched out if desired. Input data may include X,Y coordinates for subsequent map plotting. The order of input data is X coordinate, Y coordinate (if coordinates are specified), then the components 1 through NCOMP.

Data Deck: 1) Control Card

2) Variable format card

3) Observation Deck

Data Card Composition: 1) <u>Control Card</u>

column

1-3	NOBS	Number of observations
4-5	NCOMP	Number of components (up to 20)
6	PRO	One (1) if proportions have not yet been calculated, zero (0) otherwise (if in doubt set equal to one).
7	PUN	One (1) if results are to be punched, zero (0) otherwise.
8-9	COORD	If X,Y coordinates are to be read and punched for output in other programs

they must be scaled to fit an output format of F6.2. Enter power of ten to scale coordinates appropriately: (-9 to +9) otherwise leave blank (no coordinates read).

10-69 any title

2) Variable Format Card

E or F type FORTRAN for each observation (≤ 80 characters)

3) Observation Deck

The observations as described in the format.

Reference: The program is a verbatim translation of: C.R. Pelto, "Mapping of Multicomponent Systems," Journal of Geology, 62 (1954), pp. 501-511; Bobbs Merrill Geography reprint #G-178.

```
C PROGRAM TO CALCULATE PELTO'S D-FUNCTION AND RELATIVE ENTROPY
      INTEGER      PRO,RD,CLASS,PUN,GROUP,COORD
      DIMENSION DATA(20),C(25),P(25),MEMBER(25),RD(25),DP(25),TITL(17)
      DATA OUT/'    '/
   98 FORMAT(/////20X,'OBSERVATION #',I5/(1X,8E15.8))
   99 FORMAT('0PELTO''S D-FUNCTION AND RELATIVE ENTROPY'//'CLASS INDICAT
     1ES NO. OF COMPONENTS WHICH DOMINATE'//'MEMBER GIVES COMPONENTS INV
     2OLVED IN ORDER OF IMPORTANCE'//'DF MEASURES STRENGTH OF DOMINANCE
     3    (HIGH DF IMPLIES WEAKER DOMINANCE)'//'MINIMUM DF IS 100.*(1-(1
     4/CLASS))'//'RELATIVE ENTROPY MEASURES DEGREE OF MIXING    (HIGH EN
     5TROPY IMPLIES CONSIDERABLE MIXING)    (LOW ENTROPY IMPLIES LITTLE
     6MIXING)'//)
  100 FORMAT (I3,I2,2I1,I2,T8,A2,17A4)
  101 FORMAT(/////'CONTROL CARD IS  ',45X,'TITLE IS'//5X,'NOBS=',I3,3X,
     1 'NCOMP=',I2,3X,'PRO=',I2,3X,'PUN=',I2,3X,'COORD=',I2, 5X,17A4)
  102 FORMAT('0',2I10)
  103 FORMAT ('1',17A4)
  104 FORMAT(/////'VARIABLE FORMAT FOR ONE OBSERVATION IS',5X,20A4/
     1 '1GROUP IS A BINARY CODE FOR CLASS MEMBERS'//11X,'I',7X,'GROUP')
  105 FORMAT(    '0NO=',I5,5X,'ENTROP=',F15.8,5X,'D-FUNCTION=',F15.8,
     1 5X,'CLASS=',I5,5X,'GROUP=',I10)
  106 FORMAT('0MEMBER MATRIX'//20I5)
  107 FORMAT('0PROPORTIONS ARE'//(1X,8E15.8))
  108 FORMAT('0','X=',E15.8,10X,'Y=',E15.8,10X,'SCALE=',E15.8)
  109 FORMAT(1X,I3,2F6.2,2E14.7,I4,I9,/(5E15.8))
  110 FORMAT(1X,I3,2E14.7,I4,I9,/(5E15.8))
  111 FORMAT('0+++++++++++++++++++++++++++++++++++++++++++++++++++++++')
  112 FORMAT(20A4)
  113 FORMAT (/////'CONTROL CARD IS  ',45X,'TITLE IS'//3X,'NOBS=',I3,3X,
     1 'NCOMP=',I2,3X,'PRO=',I2,3X,'PUN=',I2,3X,'COORD IS BLANK',5X,
     2 17A4)
      WRITE (6,99)
    5 READ (5,100) NOBS,NCOMP,PRO,PUN,COORD,CRDOUT,TITL
      IF(CRDOUT.NE.OUT) WRITE(6,101) NOBS,NCOMP,PRO,PUN,COORD,TITL
      IF(CRDOUT.EQ.OUT) WRITE(6,113) NOBS,NCOMP,PRO,PUN,TITL
      READ (5,112) DATA
C FORMAT APPLIES TO ONE OBSERVATION OF NCOMP COMPONENTS
      WRITE(6,104) DATA
      NCOMP2 = NCOMP +2
      DO 10 I=1,NCOMP
      J=I-1
      IF (J.EQ.0) GROUP =1
      IF (J.NE.0) GROUP =2**J
   10 WRITE (6,102) I,GROUP
      WRITE (6,103) TITL
      NO=0
      DENOM=ALOG(1.0/NCOMP)
   15 NO=NO+1
      IF(CRDOUT.EQ.OUT) GO TO 20
      READ (5,DATA) X,Y,(P(I),I=1,NCOMP)
      SCALE = 10.0**COORD
      X = X*SCALE
      Y = Y*SCALE
      GO TO 25
   20 READ (5,DATA) (P(I),I=1,NCOMP)
```

```
   25 WRITE (6,98) NO,(P(I),I=1,NCOMP)
C CALCULATE PROPORTIONS
      IF(PRO.NE.1) GO TO 40
      SUM = 0.
      DO 30 I=1,NCOMP
   30 SUM = SUM + P(I)
      DO 35 I=1,NCOMP
   35 P(I) =P(I)/SUM
C RANK PROPORTIONS FROM GREATEST TO LEAST
   40 DO 45 I=1,NCOMP2
   45 D(I) =-1.0
      MEMBER (1)=1
      P(NCOMP+1) =0.0
      LOOK = NCOMP +1
      DO 60 I=1,LOOK
      DO 55 K=1,I
      IF(P(I).LE.D(K)) GO TO 55
      L=I-K+1
      DO 50 JL=1,L
      J=I-JL+1
      M=J+1
      D(M)=D(J)
   50 MEMBER (M)=MEMBER(J)
      D(K)= P(I)
      MEMBER(K)=I
      GO TO 60
   55 CONTINUE
   60 CONTINUE
C CALCULATE DIFFERENCES
      DO 65 I=1,NCOMP
      L=I+1
   65 DP(I) =ABS(P(MEMBER(I))-P(MEMBER(L)))
C RANK DIFFERENCES
      DO 70 I=1,NCOMP2
   70 D(I)=-1.0
      RD(1)=1
      SUM=0.0
C CALCULATE RELATIVE ENTROPY
      DO 85 I=1,NCOMP
      IF (P(I).GT.0.001) SUM=SUM+P(I)*ALOG(P(I))
      DO 80 K=1,I
      IF (DP(I).LE.D(K))  GO TO 80
      L=I-K+1
      DO 75 JL=1,L
      J=I-JL+1
      M=J+1
      D(M)=D(J)
   75 RD(M)=RD(J)
      D(K)=DP(I)
      RD(K)=I
      GO TO 85
   80 CONTINUE
   85 CONTINUE
       ENTROP = 100.0*SUM/DENOM
      DF = 100.0*(1.0-(DP(RD(1))-DP(RD(2))))
      CLASS = RD(1)
```

```
      GROUP =0
      DO 90 I=1,CLASS
      J=MEMBER(I)-1
      IF (J.EQ.0) M=1
      IF (J.NE.0) M=2**J
   90 GROUP = GROUP +M
      WRITE(6,105)    NO,ENTROP,DF,CLASS,GROUP
      WRITE(6,106)   (MEMBER(I),I=1,CLASS)
      WRITE (6,107)   (P(I),I=1,NCOMP)
      IF(CRDOUT.NE.OUT)    WRITE (6,108)   X,Y,SCALE
      IF(PUN.NE.1) GO TO 95
      IF(CRDOUT.NE.OUT)    WRITE(7,109)    NO,X,Y,ENTROP,DF,CLASS,GROUP,
     1   (P(I),I=1,NCOMP)
      IF(CRDOUT.EQ.OUT)    WRITE (7,110)   NO,ENTROP,DF,CLASS,GROUP,
     1   (P(I),I=1,NCOMP)
   95 WRITE(6,111)
      IF (NO.LT.NOBS) GO TO 15
      GO TO 5
      END
```

POPULATION MAPS

Purpose: The program reads rectangular coordinates (x,y) and populations, and then draws population maps, with cities represented as circles.

Description: For cities it has been shown that the built up area is extremely well approximated by a circle of radius $r = aP^b$. For the United States a = 0.0219, b = 0.44 when the radius is in miles. These values are used as the default options in the program. If the population is zero, or if the radius is less than 0.01 inch at the plotting scale, the program produces a dot. Up to 9999 circles may be plotted on a single map.

Comments: Input/output--reads from unit 6, writes on 7, plots on 9. Output is planned for 30" CALCOMP plotter and the standard CALCOMP plotting library is assumed. X, Y, XMIN, YMIN, DV are assumed to be in miles; if in inches set SCAL = 63360. If ALPHA and BETA (see below) are defaulted, both must be defaulted; conversely if one is supplied both must be supplied.

Data Deck:
1) Control card
2) Title card
3) Variable format card
4) Observation deck

Data Card Composition: 1) Control card columns

1-4	NOBS	Number of observations (≤ 9999) (I4)
5-9	XMIN	Minimum value of the x coordinate (F5.0)
10-14	YMIN	Minimum value of the y coordinate (F5.0)
15-24	SCAL	Denominator of the representative fraction (F10.0)
25-27	DV	Spacing of tick marks. (F3.0)
28-33	ALPHA	Coefficient of proportionality, a (F6.0) Default = 0.0219.
34-39	BETA	Exponent, b (F6.0) Default = 0.44

2) Title card

Any title (< 32 characters)

3) Variable format card

E or F FORTRAN type format for one observation.

4) Observation deck

Observations as described by the variable format card.

Programmer: Waldo R. Tobler

Reference: W.R. Tobler, "Satellite Confirmation of Settlement Size Coefficients," Area, Vol. 1, No. 3 (1969), pp. 30-34.

```
C       PROGRAM TO DRAW POPULATION MAPS
        INTEGER T
        DIMENSION TIT(8),FMT(18)
10      FORMAT (I4,2F5.0,F10.0,F3.0,2F6.0)
11      FORMAT (8A4)
12      FORMAT (7H1TIT = ,8A4)
14      FORMAT (8H0ALPH = ,F6.0/8H BETA = ,F6.0)
13      FORMAT (18A4)
15      FORMAT (10H0SUMPOP = ,F15.5)
        MILE = 63360.
        T=0
2       READ (6,10) NOBS,XMIN,YMIN,SCAL,DV,ALPH,BETA
        IF(ALPH.GT.0.0) GO TO 3
        ALPH = 0.0219
        BETA = 0.44
3       READ (6,11) TIT
        WRITE (7,12) TIT
        WRITE (7,14) ALPH,BETA
        PX = 1.0
        PY = 1.0
        SIZE = 0.1
        THO = 0.
        CALL PSYMB(PX,PY,SIZE,'GEOGRAPHY DEPARTMENT UNIVERSITY OF MICHIGAN
       1  1968 ',THO,50)
        PX = PX+6.0
        CALL PSYMB(PX,PY,SIZE,TIT,THO,32)
        SCAL = SCAL/MILE
        SUMPOP = 0.0
        CALL PLTOFS(XMIN,SCAL,YMIN,SCAL,1.0,3.0)
        XMAX = XMIN
        READ (6,13) FMT
        THF = 360.
        K=0
        DL = 0.
        SC = 10.
        K = K+1
        READ (6,FMT) X,Y,POP
        RHO = ALPH*(POP**BETA)
        XMAX = AMAX1(X,XMAX)
        IF((RHO/SCAL).LT.0.01) RHO = 0.01*SCAL
        CALL PCIRCL(X,Y,THO,THF,RHO,RHO,DL,SC)
        SUMPOP = SUMPOP+POP
        IF(K.LT.NOBS) GO TO 4
        DV = DV/SCAL
        AXLTH = -(XMAX-XMIN)/SCAL
        CALL PAXIS(1.0,2.0,T,-0,AXLTH,THO,XMIN,SCAL,DV)
        WRITE (7,15) SUMPOP
        CALL PLTEND
        GO TO 2
        END
```

SPHERICAL DISTANCES

Purpose: To compute distances in kilometers assuming one degree equals 69.172 miles.

Description: The program reads a vector of latitude/longitude coordinates and produces the complete distance matrix; i.e., the distance between all pairs of points specified in the coordinate vector. Latitudes and longitudes are assumed given in degrees and minutes, with the usual convention that South latitudes and West longitudes are negative.

Deck Make-up: 1) Control and Format Card

2) Observations deck

Card Format: Control and format Card

columns

1-3	Number of observations ≤ 100
4-67	Fortran format for one observation

Observations deck (all E or F fields)

latitude in whole degrees

minutes of latitude

longitude in whole degrees

minutes of longitude

References:

W.R. Tobler, "A Comparison of Spherical and Ellipsoidal Measures", Professional Geographer, XVI, 4(1964), pp. 9-12.

W.R. Tobler, Geographical Coordinate Computations", University of Michigan, Contract Nonr 1224 (48), Task 389-137; UM 05824-2,3-T, December, 1964, 107 pages.

```
C FORTRAN PROGRAM TO COMPUTE SPHERICAL DISTANCES
C READ NOBS + FORMAT, THEN DATA
C READS 5, WRITES 6, PUNCHES 7
      DIMENSION A(100,100),C(100,100),FMT(16)
      RAD=174532925E-10
1     EARTH=69.172*1.60935/RAD
      READ(5,10) NO,(FMT(I),I=1,16)
10    FORMAT(I3,16A4)
      WRITE(6,48)
48    FORMAT('1LATITUDES AND LONGITUDES')
      DO 701 I=1,NO
      A(I,I)=0.0
      READ(5,FMT) FLTD,FLTM,FLGD,FLGM
      IF (FLTD) 706,707,707
706   FLT=FLTD-FLTM/60.
      GO TO 708
707   FLT=FLTD+FLTM/60.  <-- 708   IF (FLGD) 709,710,710
709   FLG=FLGD-FLGM/60.
      GO TO 711
710   FLG=FLGD+FLGM/60.
711   C(I,1)=FLT*RAD
      C(I,2)=FLG*RAD
701   WRITE(6,49) I,FLT,FLG
49     FORMAT(1X,I3,2(1X,F8.3))
      WRITE(6,50)
50    FORMAT('1SPHERICAL DISTANCES')
      DO 750 I=1,NO
      K=I+1
      IF (NO-K) 780,740,740
740   DO 749 J=K,NO
      CALL SPHERE(C(I,1),C(I,2),C(J,1),C(J,2),RHO,G)
      A(I,J)=RHO*EARTH
749    A(J,I)=A(I,J)
750   CONTINUE
780   CONTINUE
      DO 760 I=1,NO
760   WRITE(7,43) I,(A(I,J),J=1,NO)
43    FORMAT(1H0,I3,10E12.5/(1H ,3X,10E12.5/))
      WRITE(6,52) (FMT(I),I=1,16)
52    FORMAT(1H1,16A4)
      GO TO 1
      END
```

```
      SUBROUTINE SPHFRE (FLT,FLG,FLAT,FLON,RHO,GA)
C     CONSTANTS
      FI=314159265E-8
      TPI=628318531E-8
      FIOVR2=157089633E-8
      RAD=174532925E-10
      EPS=1E-6
      CNLT=COS(FLT)
      SNLT=SIN(FLT)
      DIF=FLON-FLG
      CDIF=COS(DIF)
      SDIF=SIN(DIF)
      CLT=COS(FLAT)
      SLT=SIN(FLAT)
      Q=SLT*SNLT+CLT*CNLT*CDIF
      IF(Q-1) 16,15,15
15    RHO=0.0
      GO TO 19
16    IF(Q+1) 17,17,18
17    RHO=FI
      GO TO 19
18    RHO=ARCOS(Q)
19    FUM=CLT*SDIF
      DEN=CNLT*SLT-SNLT*CLT*CDIF
      IF(ABS(DEN)-EPS) 20,21,21
20    IF(ABS(FUM)-EPS) 22,23,23
22    GA=0.0
      GO TO 30
23    GA=FIOVR2
      IF(FUM) 24,30,30
24    GA=-GA
      GO TO 30
21    IF(ABS(FUM)-EPS) 25,28,28
25    GA=0.0
      IF(DEN) 26,27,27
26    GA=FI
27    GO TO 30
28    GA=ATAN2(FUM,DEN)
      IF (FI-GA) 29,30,30
29    GA=GA-TPI
30    CONTINUE
      RETURN
      END
```

COORDINATES FROM DISTANCES

Purpose: The program reads a matrix of distances, and then produces a vector of plane coordinates.

Deck Make-up: 1) Title Card

2) Control Card

3) Distance Matrix

Data Cards: Title Card (18A4)

any title

Control Card (2I2, I1, 1X, 16A4)

column

1-2	NRANDC	Number of rows and columns in the distance matrix.
3-4	NREIGN	Number of eigenvectors desired
5	IFPLOT	zero to plot the solutions.
7-70	FMT	Format for one row of the distance matrix.

Distance Matrix

The distance matrix as described by the format. Only the lower half is read.

Reference: W.S. Torgerson, Theory and Methods of Scaling, New York, J. Wiley, 1958, pp. 254-259.

J. Lingoes, "An IBM 7090 Program for Guttman- Lingoes Smallest Space Analysis", Behavioral Science, 10(1965), pp. 183-184.

W. Tobler, H. Mielke, T. Detwyler, "Geobotanical Distance Between New Zealand and Neighboring Islands", Bioscience, May 1970.

```
C
C
C   M A P   C O O R D I N A T E S   F R O M   G E O - D I S T A N C E S
C
C PROGRAM TO COMPUTE EUCLIDEAN COORDINATES FROM GIVEN DISTANCES
C WALDO R. TOBLER               1967
C DEPARTMENT OF GEOGRAPHY UNIVERSITY OF MICHIGAN
C HOUSEHOLDER AND YOUNG THEOREM - TORGERSONS METHOD
C FOR METRIC MULTI-DIMENSIONAL SCALING
C
C     INPUT
C READS 7 WRITES 5 AND 6
C READS TITLE, CONTROLS + FORMAT, THEN LOWER-HALF DISTANCE MATRIX
C     CARD1 TITLE (1-72)
C     CARD # 2
C
C     NRANDC (1-2) ROWS AND COLUMNS
C     NREIGN (3-4) COLS OF EIGEN (<11)
C     IFPLOT (5)   =0 FOR PLOT
C     FORMAT (7-72)
C     CARD # 3 AND ON = LOWER HALF OF DISTANCE MATRIX
      DIMENSION A(70,70), B(70,70), C(70,70), D(70,70), FMT(16), TITL(18
     1), DUMYA(70), EVALU(70), X1(70,2), RSUM(70), CSUM(70)
      EQUIVALENCE (DUMYA,EVALU), (NR,NRANDC), (NC,NREIGN)
      EQUIVALENCE (A,B)
      MD=70
C     READ TITLE
1     READ (7,37) (TITL(I),I=1,18)
      READ(7,38) NRANDC,NREIGN,IFPLOT,(FMT(I),I=1,16)
      IFSPM=0
C     OUTPUT OF INPUT
      WRITE (6,39) (TITL(I),I=1,18),(FMT(I),I=1,16)
      WRITE (6,40)
C     READ AND PRINT MATRIX
      IF ((IFSPM)) 2,3,2
2     WRITE (6,41)
      GO TO 4
3     WRITE (6,42)
4     DO 7 I=1,NRANDC
      IF (I-NRANDC) 5,6,6
5     IP1=I+1
      READ (7,FMT) (A(I,J),J=1,I),(DUMYA(J),J=IP1,NRANDC)
      GO TO 7
6     READ (7,FMT) (A(I,J),J=1,NRANDC)
 7    WRITE(5,43) I,(A(I,J),J=1,I)
      DO 8 I=2,NRANDC
      IM1=I-1
      DO 8 J=1,IM1
8     A(J,I)=A(I,J)
C     CHANGES TO DISTANCE MATRIX CAN BE PERFORMED HERE
C     (I.E., GET GEOBOTANICAL DISTANCES FROM SPECIES SIMILARITIES, ETC.)
      IF (IFSPM) 9,11,9
9     DO 10 I=1,NR
      DO 10 J=1,NR
10    C(I,J)=A(I,J)
```

```
      GO TO 16
C     GET SPM FROM DISTANCE MATRIX
11    DO 12 I=1,NRANDC
      DO 12 J=1,NRANDC
12    C(I,J)=B(I,J)**2
      DO 13 J=1,NRANDC
      RSUM(J)=0.
      CSUM(J)=0.
      DO 13 I=1,NRANDC
      RSUM(J)=C(I,J)+RSUM(J)
13    CSUM(J)=C(J,I)+CSUM(J)
      FNRC=NRANDC
      GSUM=0
      DO 14 I=1,NRANDC
      GSUM=RSUM(I)+GSUM
      RSUM(I)=RSUM(I)/FNRC
14    CSUM(I)=CSUM(I)/FNRC
      GSUM=GSUM/(FNRC*FNRC)
      DO 15 I=1,NRANDC
      DO 15 J=1,NRANDC
15    C(I,J)=0.5*(RSUM(I)+CSUM(J)-GSUM-C(I,J))
16    IF (NREIGN) 17,17,18
17    NREIGN=NRANDC
C     TRACE OF MATRIX "C"
18    TRACE=0
      DO 19 I=1,NRANDC
19    TRACE=TRACE+C(I,I)
      WRITE (6,39) (TITL(I),I=1,18)
C     GET EIGEN RESULTS
      CALL EIGEN(C,D,NRANDC,EVALU,NREIGN,MD)
C     PRINT EIGENVALUES
      WRITE(6,49)
      WRITE(6,50) TRACE,NREIGN
      WRITE(6,51)
C     GET AND PRINT PERCENT AND CUM. PERCENT EIGENVALUES
      PERCVU=0
      DO 22 I=1,NREIGN
      IF (TRACE) 20,21,20
20    PERCVU=PERCVU+EVALU(I)/TRACE
      WRITE (6,44) I,EVALU(I),PERCVU
      GO TO 22
21    WRITE (6,44) I,EVALU(I)
22    CONTINUE
C     APPLY MEAN SCALE FACTOR TO FIRST TWO EIGEN VECTORS
      F=0.0
      SUM=0.0
      DO 26 I=1,NR
      IM1=I
      DO 25 J=1,IM1
      IM2=J
      IF (IM1-IM2) 23,26,23
23    C(I,J)=A(I,J)
      B(I,J)=SQRT((D(I,1)-D(J,1))**2+(D(I,2)-D(J,2))**2)
      IF (C(I,J)) 24,25,24
24    SUM=SUM+(B(I,J)/C(I,J))
      F=F+1.0
```

```
25    CONTINUE
26    CONTINUE
      RMULT=F/SUM
C     PROGRAM NOW SCALES FIRST TWO EIGENVECTORS FOR GEO-MAPS
      DO 27 I=1,NR
      DO 27 J=1,2
27    D(I,J)=D(I,J)*RMULT
      WRITE (6,45) RMULT
C     PRINT OUT EIGENVECTORS
      WRITE (6,39) (TITL(I),I=1,18)
      WRITE (6,46) NRANCC,NREIGN
      DO 28 I=1,NRANCC
28    WRITE (6,47) I,(D(I,J),J=1,NREIGN)
C     PRINT RESULTING DISTANCES, USING FIRST TWO EIGENVECTORS
      WRITE (6,39) (TITL(I),I=1,18)
      WRITE (6,48)
      DO 31 I=1,NR
      IM1=I
      DO 30 J=1,IM1
      IM2=J
      IF (IM1-IM2) 30,29,30
29    B(I,J)=0.0
      GO TO 31
30    B(I,J)=SQRT((D(I,1)-D(J,1))**2+(D(I,2)-D(J,2))**2)
31    WRITE(5,43) I,(B(I,J),J=1,IM1)
C     NORMALIZE FOR PLOTS
      IF (IFPLOT) 32,32,36
32    S=0
      DO 34 K=1,NC
      DO 34 I=1,NR
      IF (S-ABS (D(I,K))) 33,34,34
33    S=ABS (D(I,K))
34    CONTINUE
      DO 35 K=1,NC
      DO 35 I=1,NR
35    D(I,K)=D(I,K)/S
C     PLOT OUT THE VECTORS OF D
      CALL PLOT(D,X1(1,1),X1(1,2),NR,NC,MD)
36    GO TO 1
C
37    FORMAT (18A4)
38    FORMAT(2I2,I1,1X,16A4)
39    FORMAT('1EIGENVALUE AND EIGENVECTOR PROGRAM',/1H ,18A4/5X,18A4)
40    FORMAT('LOWER HALF OF DATA MATRIX')
41    FORMAT('DATA IS SCALER PRODUCT MATRIX')
42    FORMAT('DATA IS RAW DISTANCE MATRIX')
43    FORMAT (1H0,I3,10E12.5/(1H ,3X,10E12.5/))
44    FORMAT (1H0,I2,F17.2,2PF24.2)
45    FORMAT('0MULTIPLIER IS ', F11.5)
46    FORMAT(' EIGENVECTORS'/'ROWS=',I10,10X,'COLUMNS=',I10)
47    FORMAT (1H0,I2,2X,10F11.3)
48    FORMAT('RESULTING DISTANCES')
49    FORMAT('- E I G E N V A L U E S ' /)
50    FORMAT( ' TRACE= ',E12.5, 10X,'N= ',I10, /)
51    FORMAT(' ROWS',10X,'VALUE',10X,'CUM. PERCENT OF TOTAL',/1H0 )
      END
```

```
C     EIGENVALUES AND NORMALIZED EIGENVECTORS OF A REAL SYMMETRIC MATRIX
C     EIGENVALUES ARE RETURNED IN VALU AND NORMALIZED EIGENVECTORS ARE
C     STORED IN B.  NSUB IS ORDER OF MATRICES A AND B AND MSUB IS THE
C     NUMBER OF ROOTS AND VECTORS DESIRED.
C
      SUBROUTINE EIGEN (A,B,NSUB,VALU,MSUB,MD)
C
      DIMENSION A(MD,MD), B(MD,MD), VALU(MD), T(70,3), DIAG(70), SUPERD(
     170), WVEC(70), PVEC(70),QVEC(70), VALL(70), Q(70), U(70), INDEX(70
     21, V(70)
C
      EQUIVALENCE (WVEC,VALL,U),              (PVEC,QVEC,Q,V), (I1,T1),
     1(I2,T2),      (TEMP,TO), (SUM,MATCH), (I,P), (DIV,SCALAR,TAU), (AN
     2ORM2,ANORM), (VTEMP,VNORM2,VNORM)
C
C
C     INITIALIZATION
      N=NSUB
      M=MSUB
      NP1=N+1
      NM1=N-1
       E1=1.E-8
C     GENERATE IDENTITY MATRIX
      DO 3 I=1,N
      DO 3 J=1,N
      IF (I-J) 2,1,2
1     B(I,J)=1.
      GO TO 3
2     B(I,J)=0.
3     CONTINUE
C     HOUSEHOLDER SIMILARITY TRANSFORMATION TO CO-DIAGONAL FORM
C     REDUCE COLUMN OF MATRIX
      DO 14 I=1,NM1
      IF (I-NM1) 4,13,4
4     I1=I+1
      I2=I1+1
      SUM=0.
      DO 5 J=I2,N
5     SUM=SUM+A(J,I)**2
      IF (SUM) 6,13,6
6     J=I1
      TEMP=A(J,I)
C                                                   TERMINAL-F FLAG
      SUM=SQRT(SUM+TEMP**2)
      A(J,I)=-SIGN (SUM,TEMP)
C                                                   TERMINAL-F FLAG
      WVEC(J)=SQRT(1.+ABS (TEMP)/SUM)
      DIV=SIGN (WVEC(J)*SUM,TEMP)
      DO 7 J=I2,N
7     WVEC(J)=A(J,I)/DIV
      SCALAR=0.
      DO 9 J=I1,N
      PVEC(J)=0.
      DO 8 K=I1,N
8     PVEC(J)=PVEC(J)+A(K,J)*WVEC(K)
      SCALAR=SCALAR+PVEC(J)*WVEC(J)
```

```
9     CONTINUE
      SCALAR=SCALAR/2.
      DO 10 J=I1,N
      QVEC(J)=PVEC(J)-SCALAR*WVEC(J)
      DO 10 K=I1,J
      A(K,J)=A(K,J)-(WVEC(K)*QVEC(J)+WVEC(J)*QVEC(K))
      A(J,K)=A(K,J)
10    CONTINUE
C     SAVE ROTATION FOR LATER APPLICATION TO CO-DIAGONAL VECTORS
      DO 12 K=2,N
      TEMP=0.
      DO 11 J=I1,N
11    TEMP=TEMP+WVEC(J)*B(J,K)
      DO 12 J=I1,N
      B(J,K)=E(J,K)-WVFC(J)*TLMP
12    CONTINUE
C     MOVE CO-DIAGONAL FORM ELEMENTS FOR ITERATIVE PROCEDURE
13    J=I
      DIAG(I)=A(J,I)
      SUPERD(I)=A(J+1,I)
14    CONTINUE
      DIAG(N)=A(N,N)
C     DETERMINE EIGENVALUES FROM STURM CHAIN OF CO-DIAGONAL MINORS
C     CALCULATE NORM OF MATRIX AND INITIALIZE EIGENVALUE BOUNDS
      ANORM2=DIAG(1)**2
      DO 15 L=2,N
      Q(L-1)=SUPERD(L-1)**2
      ANORM2=DIAG(L)**2+Q(L-1)+Q(L-1)+ANORM2
15    CONTINUE
C                                                     TERMINAL-F FLAG
      ANORM=SQRT(ANORM2)
      DO 16 L=1,M
      VALU(L)=ANORM
      VALL(L)=-ANORM
16    CONTINUE
      EPS1=ANORM*E1
      IF (EPS1) 17,73,17
C     CHOOSE NEW TRIAL VALUE WHILF TESTING BOUNDS FOR CCNVERGENCE
17    DO 35 L=1,M
      ITER=0
      VTEMP=EPS1
18    TAU=(VALU(L)+VALL(L))/2.
      IF (ITER-10)  20,19,20
19    VTEMP=VTEMP*10.
      ITER=0
20    IF (2.*(TAU-VALL(L))-VTEMP) 35,35,21
C     DETERMINE SIGNS OF PRINCIPAL MINORS
21    MATCH=0
      ITER=ITER+1
      T2=0.
      T1=1.
      DO 30 L1=1,N
      P=DIAG(L1)-TAU
      IF (T2) 23,22,23
22    T1=SIGN (1.,T1)
23    IF (T1) 25,24,25
```

```
24     T0=-SIGN (1.,T2)
       T2=0.
       IF (Q(L1-1)) 26,22,26
25     T0=P-Q(L1-1)*T2/T1
       T2=1.
C      COUNT AGREEMENTS IN SIGN (ZERO CONSIDERED POSITIVE)
26     IF (T0) 29,27,28
27     T2=T1
       IF (T2) 29,28,28
28     MATCH=MATCH+1
29     T1=T0
30     CONTINUE
C      ESTABLISH TIGHTER BOUNDS ON EIGENVALUES
       DO 34 L1=L,M
       IF (L1-MATCH) 33,33,31
31     IF (VALU(L1)-TAU) 18,18,32
32     VALU(L1)=TAU
       GO TO 34
33     VALL(L1)=TAU
34     CONTINUE
       GO TO 18
35     CONTINUE
C      EIGENVECTORS OF CO-DIAGONAL SYMMETRIC MATRIX -- INVERSE ITERATION
C      CHECK FOR REPEATED VALUE
       DO 68 I=1,M
       IF (I-2) 37,36,36
   36  IF (VALU(I-1)-VALU(I)-1.E-3) 38,37,37
37     I1=-1
38     I1=I1+1
C      TRIANGULARIZE CO-DIAGONAL FORM AFTER EIGENVALUE SUBTRACTION
       DO 43 L=1,N
       V(L)=EPS1
       T(L,2)=DIAG(L)-VALU(I)
       IF (L-N) 40,39,40
39     T(L,3)=0.
       GO TO 43
40     T(L,3)=SUPERD(L)
       IF (T(L,3)) 42,41,42
41     T(L,3)=EPS1
42     T(L+1,1)=T(L,3)
43     CONTINUE
       DO 50 J=1,N
       T(J,1)=T(J,2)
       T(J,2)=T(J,3)
       T(J,3)=0.
       VTEMP=ABS (T(J,1))
       IF (J-N) 46,44,46
44     IF (VTEMP) 50,45,50
45     T(J,1)=EPS1
       GO TO 50
46     INDEX(J)=0
       IF (ABS (T(J+1,1))-VTEMP) 49,49,47
47     INDEX(J)=1
       DO 48 K=1,3
       VTEMP=T(J,K)
       T(J,K)=T(J+1,K)
```

```
      T(J+1,K)=VTEMP
48    CONTINUE
49    VTEMP=T(J+1,1)/T(J,1)
      U(J)=VTEMP
      T(J+1,2)=T(J+1,2)-VTEMP*T(J,2)
      T(J+1,3)=T(J+1,3)-VTEMP*T(J,3)
50    CONTINUE
      ITER=1
      IF (I1) 58,51,58
C     BACK SUBSTITUTE TO OBTAIN EIGENVECTOR
51    DO 52 L1=1,N
      L=NP1-L1
      V(L)=(V(L)-T(L,2)*V(L+1)-T(L,3)*V(L+2))/T(L,1)
52    CONTINUE
      GO TO (53,58), ITER
C     PERFORM SECOND ITERATION
53    ITER=2
54    DO 57 L=2,N
      IF (INDEX(L-1))55,56,55
55    VTEMP=V(L-1)
      V(L-1)=V(L)
      V(L)=VTEMP
56    V(L)=V(L)-U(L-1)*V(L-1)
57    CONTINUE
      GO TO 51
C     ORTHOGONALIZE VECTOR TO OTHERS ASSOCIATED WITH REPEATED ROOT
58    IF (I1) 59,62,59
59    DO 61 L1=1,I1
      K=I-L1
      VTEMP=0.
      DO 60 J=1,N
60    VTEMP=VTEMP+A(J,K)*V(J)
      DO 61 J=1,N
61    V(J)=V(J)-A(J,K)*VTEMP
62    GO TO (54,63), ITER
C     NORMALIZE VECTOR TO UNIT LENGTH
63    VNORM2=0.
      SUM=0.
      DO 65 L=1,N
      IF (SUM-ABS (V(L))) 64,65,65
64    SUM=ABS (V(L))
65    CONTINUE
      DO 66 L=1,N
      V(L)=V(L)/SUM
66    VNORM2=VNORM2+V(L)**2
C                                                         TERMINAL-F FLAG
      VNORM=SQRT(VNORM2)
      DO 67 J=1,N
67    A(J,I)=V(J)/VNORM
68    CONTINUE
C     ROTATION OF CO-DIAGONAL VECTORS INTO MATRIX EIGENVECTORS
      DO 70 I=1,M
      DO 69 K=2,N
      U(K)=0.
      DO 69 J=2,N
69    U(K)=U(K)+B(J,K)*A(J,I)
```

```
      DO 70 J=2,N
70    A(J,1)=U(J)
C     NORMALIZE LENGTH OF VECTORS TO EIGENVALUES AND STORE IN B(I,J)
      DO 72 J=1,M
      IF (VALU(J)) 73,73,71
C                                                           TERMINAL-F FLAG
71    VTEMP=SQRT(VALU(J))
      DO 72 I=1,N
72    B(I,J)=A(I,J)*VTEMP
73    RETURN
      END
C     PLOT
C
      SUBROUTINE PLOT(A,IX,IY,NV,NVAC,MD)
      DIMENSION A(MD,MD),IX(MD),IY(MD),IW(51),L1(20),L2(50),FMT(58)
      INTEGER BLANK
      DATA BLANK/4H    /,WORD1/4H A2,/,WORD2/4H I2,/
      DATA FMT(1)/4H(1H /,FMT(2)/4H,5X,/,FMT(3)/4HI5,2/
      DATA FMT(4)/4HX1H*/,FMT(5)/4H,   /,FMT(31)/4H1H*,/
      DATA FMT(57)/4H1H*,/,FMT(58)/4HI4) /
C
C     TAPE ASSIGNMENT
      JTAPE=6
      DO 1 I=1,10
      J=11-I
      L1(J)=-I*10
      K=I+10
1     L1(K)=I*10
      DO 2 I=1,25
      J=26-I
      L2(J)=I*4
      K=25+I
2     L2(K)=-I*4
      NFC=NVAC-1
      DO 18 I=1,NFC
      IP1=I+1
      DO 3 K=1,NV
3     IY(K)=((IFIX(A(K,I)*100.)+100)/4)
      DO 18 J=IP1,NVAC
      DO 4 K=1,NV
4     IX(K)=50-((IFIX(A(K,J)*100.)+100)/4)
      WRITE (JTAPE,19) J,I,J,(L1(K),K=1,20)
      DO 17 KX=1,50
      DO 5 M=1,50
5     IW(M)=BLANK
      DO 11 L=1,NV
      IF (IX(L)-KX) 11,6,11
6     IF (IY(L)-50) 7,7,8
7     IF (IY(L)-1) 9,10,10
8     IY(L)=50
      GO TO 10
9     IY(L)=1
10    N=IY(L)
      IW(N)=L
11    CONTINUE
      DO 13 K=1,25
```

```
      L=K+5
      IF (IW(K).EQ.PLANK) GO TO 12
      FMT(L)=WORD2
      GO TO 13
12    FMT(L)=WORD1
13    CONTINUE
      DO 15 K=26,50
      L=K+6
      IF (IW(K).EQ.PLANK) GO TO 14
      FMT(L)=WORD2
      GO TO 15
14    FMT(L)=WORD1
15    CONTINUE
      IF (KX-26) 17,16,17
16    WRITE (JTAPE,20) I
17    WRITE (JTAPE,FMT) L2(KX),(IW(L),L=1,50),L2(KX)
18    WRITE (JTAPE,21) (L1(K),K=1,20)
      RETURN
C     *** FORMAT STATEMENTS ***
C
19    FORMAT (1H1,12HVECTOR PLOTS/1H ,6HVECTOR,I5,1X22HPLOTTED AGAINST V
     1ECTOR,I5,21X6HVECTOR/1H ,60X14/1H0,11X10I5,2X1H*,I4,9I5/1H ,13X50(
     22H* ),1H*)
20    FORMAT (1H ,6HVECTOR,I3,4X50(2H* ),1H*)
21    FORMAT (1H ,13X50(2H* ),1H*/1H ,11X10I5,2X1H*,I4,9I5)
      END
```

DATA PLOTTING

Purpose: Rapid data screening.

Description: The program produces the $N \cdot (N - 1)/2$ printer plots, linear regressions, and simple correlations for all pairwise combinations of N variables. All operations are done according to single precision standard statistical procedures.

Comments: Current dimensioning allows 2000 observations for each of 30 variables. I/O reads controls and format from unit 7, reads observations from 4, and writes onto 6. Multiple observation decks may be run by repeating the data deck. The UMPLOT subroutines are available through SHARE. The end points of the linear regression are indicated by dots in the margin of the plots.

Data deck:

1) Title card
2) Control card
3) Format card
4) Observation Deck
5) End of observations card(s)

Data Card Composition:

1) <u>Title Card</u>

   Any title (≤ 72 characters)

2) <u>Control Card</u>

   column

   1-2 NVAR The number of variables (≤ 15) I2

3-4 INDEP Independent variable indicator (i.e., 5 yields 5th variable on x axis). Default = last variable.

5-6 IDEP Dependent variable indicator (y axis). Default = 1

3) Format Card

E or F type FORTRAN format for one observation of the N variables. (≤72 characters)

4) Observation Deck

Observations as described by the FORMAT

5) "End of observations" card(s)

Put 9999 in columns 1-4; this avoids having to know the exact number of observations. If the observations are on M cards, follow this last card by M-1 blank cards.

Programmer: Program originally written by Krumbein and Benson of Northwestern University. Revised by Tobler of the University of Michigan, 1962.

References: Comparable programs are now available in the BIMED system.

```
C PROGRAM TO PLOT SCATTER DIAGRAMS
      DIMENSION TRANG(30), VMIN(30), VMAX(30), RANG(30)
      DIMENSION V(30,2000),              X(2000),Y(2000)
      DIMENSION TITL(30), FMT(21), GRAPH(1200), DELX(30), DELY(30), DELX
     1X(30), DELYY(30), SUMX(30), SUMXQ(30), SUMCOV(30,30), SIGX(30)
      DIMENSION XBAR(30), SSX(30), R(30,30), B(30,30), A(30,30), RSQ(30,
     130)
      DIMENSION W(30)
      DATA FMT/'(A4,T1, ',18*'     ',' ) '/,HALT/'9999'/
C     *****SET UP TABLE OF RANGES
      TRANG(1)=0.01
      DO 1 I=2,10
      L=I-1
    1 TRANG(I)=10.0*TRANG(L)
      NSCALE=0
      NHL=6
      NSBH=8
      NVL=8
      NSBV=10
C     *****READ TITLES AND MASTER
    2 READ (7,39) (TITL(I),I=1,18)
      WRITE (6,35)(TITL(I),I=1,18)
      READ (7,34)      NVAR,INDEP,IDEP
      WRITE (6,41) NVAR,INDEP,IDEP
      READ (7,33) (FMT(I),I=3,20)
      WRITE(6,33) (FMT(I),I=3,20)
      NVAR1 = NVAR
      IF (IDEP) 498,499,499
  498 IDEP = -IDEP
      NVAR1 = NVAR + NVAR
C     ** ZERO OUT SUMS
  499 DO 3 I =1,NVAR1
      SUMX(I)=0.0
      SUMXQ(I)=0.0
      DO 3 J =1,NVAR1
    3 SUMCOV(I,J)=0.0
      COUNT=0.0
C     *****READ AND STORE DATA MATRIX
      N=0
    4 N=N+1
      READ(4,FMT)       END,(V(I,N),I=1,NVAR)
      IF (NVAR-NVAR1) 501,505,505
  501 DO 500 K=1,NVAR
      JJ = K
      J = NVAR + JJ
  500 V(J,N)=V(JJ,N)**2
  505 IF(END.EQ.HALT) GO TO 11
    5 IF (INDEP) 7,7,6
    6 DUM1=V(NVAR,N)
      V(NVAR,N)=V(INDEP,N)
      V(INDEP,N)=DUM1
    7 IF (IDEP) 9,9,8
    8 DUM1=V(1,N)
      V(1,N)=V(IDEP,N)
      V(IDEP,N)=DUM1
```

```
9       CONTINUE
C       ** ACCUM SUMS
        DO 10 I =1,NVAR1
        SUMX(I)=SUMX(I)+V(I,N)
        SUMXQ(I)=SUMXQ(I)+V(I,N)**2
        DO 10 J =1,NVAR1
10      SUMCOV(I,J)=SUMCOV(I,J)+V(I,N)*V(J,N)
        COUNT=COUNT+1.0
        GO TO 4
C       *** END OF DATA
11      N=N-1
        NVAR=NVAR1
C       ** GET SAMPLE STATISTICS
        DO 12 I=1,NVAR
        XBAR(I)=SUMX(I)/COUNT
        SSX(I)=SUMXQ(I)-XBAR(I)*SUMX(I)
        VARX=SSX(I)/COUNT
12      SIGX(I)=SQRT(VARX)
        DO 13 I=1,NVAR
        DO 13 J=1,NVAR
        COVXX=SUMCOV(I,J)-XBAR(I)*SUMX(J)
        R(I,J)=(COVXX/(SIGX(I)*SIGX(J)))/COUNT
        B(I,J)=COVXX/SSX(J)
        A(I,J)=XBAR(I)-B(I,J)*XBAR(J)
13      RSQ(I,J)=R(I,J)**2*100.0
C       *****PRINT OUT DATA
        WRITE (6,35) (TITL(I),I=1,18)
        WRITE (6,36) NVAR
        DO 14 M=1,N
14      WRITE (6,37)        M ,(V(I,M),I=1,NVAR)
        WRITE (6,38)
        WRITE (6,37) N,(XBAR(I),I=1,NVAR)
        WRITE (6,37) N,(SIGX(I),I=1,NVAR)
C       ****FIND MIN AND MAX VALUES EACH VECTOR
        DO 24 I=1,NVAR
        VMAX(I)=V(I,1)
        VMIN(I)=V(I,1)
        DO 18 M=2,N
        IF (V(I,M)-VMAX(I)) 16,18,15
15      VMAX(I)=V(I,M)
        GO TO 18
16      IF (V(I,M)-VMIN(I)) 17,18,18
17      VMIN(I)=V(I,M)
18      CONTINUE
        RANG(I)=VMAX(I)-VMIN(I)
C       *****SELECT FROM TABLE A RANGE SLIGHTLY LARGER THAN ACTUAL RANGE.
C       *****FOR EACH VECTOR.
        DO 19 K=3,10
        M=K-2
        IF (RANG(I)-TRANG(K)) 20,19,19
19      CONTINUE
20      DO 21 K=1,N
21      V(I,K)=V(I,K)/TRANG(M)
        VMAX(I)=VMAX(I)/TRANG(M)
        VMIN(I)=VMIN(I)/TRANG(M)
        RANG(I)=RANG(I)/TRANG(M)
```

```
      W(I)=TRANG(M)
      DO 22 K=1,10
      VMARG=FLOAT(K)
      VMARG=VMARG*10.0
      XMIN=RANG(I)-VMARG
      IF (XMIN) 23,22,22
22    CONTINUE
C     *****SPREAD MAX + MIN(SPLIT DIFFERENCE) TO FIT NEW RANGE SELECTED
23    VMARG=-XMIN/2.0
      VMIN(I)=VMIN(I)-VMARG
      VMIN(I)=HFIX(VMIN(I))
      VMAX(I)=VMAX(I)+VMARG+0.5
      VMAX(I)=HFIX(VMAX(I))
      RANG(I)=VMAX(I)-VMIN(I)
C     *****CALC INCREMENTS EACH SPACE ALONG X AND Y AXIS
      DELX(I)=RANG(I)/77.0
      DELY(I)=RANG(I)/45.0
      DELXX(I)=RANG(I)/7.0
24    DELYY(I)=RANG(I)/5.0
C     ***SET UP GRAPHS EACH PAIR OF VECTORS
      DO 32 I=1,NVAR
C     ***ITH VECTOR BECOMES YVECTOR
      DO 25 M=1,N
25    Y(M)=V(I,M)
      YMIN=VMIN(I)
      YMAX=VMAX(I)
      DO 32 J=1,NVAR
      IF (I-J) 26,32,26
C     ***JTH VECTOR BECOMES X VECTOR
26    DO 27 M=1,N
27    X(M)=V(J,M)
      XMIN=VMIN(J)
      XMAX=VMAX(J)
C     ***SET UP GRID WITH AXES AND VALUES EACH XDIV + YDIV
      CALL PLOT1 (NSCALE,NHL,NSBH,NVL,NSBV)
      CALL PLOT2 (GRAPH,XMAX,XMIN,YMAX,YMIN)
C     ***DELETE GRID LINES FROM BODY OF GRAPH
      KX=4
      DO 28 K=1,KX
      YVAL=YMIN+FLOAT(K)*DELYY(I)
      DO 28 L=1,76
28    CALL PLOT3 (1H ,XMIN+FLOAT(L)*DELX(J),YVAL,1,4)
      KX=6
      DO 29 K=1,KX
      XVAL=XMIN+FLOAT(K)*DELXX(J)
      DO 29 L=1,44
29    CALL PLOT3 (1H ,XVAL,YMIN+FLOAT(L)*DELY(I),1,4)
C     ***PLOT END POINTS OF REGRESSION LINE IN MARGIN
      YVAL=(A(I,J)+B(I,J)*W(J)*XMIN)/W(I)
      CALL PLOT3 (1H.,XMIN,YVAL,1,4)
      YVAL=(A(I,J)+B(I,J)*W(J)*XMAX)/W(I)
      CALL PLOT3 (1H.,XMAX,YVAL,1,4)
      ABIJ=ABS(B(I,J))
      IF (ABIJ-0.000001) 31,31,30
30    XVAL=(YMIN*W(I)-A(I,J))/(B(I,J)*W(J))
      CALL PLOT3 (1H.,XVAL,YMIN,1,4)
```

```
      XVAL=(YMAX*W(I)-A(I,J))/(B(I,J)*H(J))
      CALL PLOT3 (1H.,XVAL,YMAX,1.4)
C     *****PLOT ALL DATA POINTS, ITH AND JTH VECTORS
31    CALL PLOT3 (1H*,X(1),Y(1),N,4)
C     ***WRITE OUT GRAPH
      WRITE (6,39) I,J,A(I,J),B(I,J),R(I,J),RSQ(I,J),N
      CALL PLOT4 (25,25H                 ITH VECTOR)
      WRITE (6,40)
32    CONTINUE
      GO TO 2
C           W.R.TOPLER / UNIVERSITY OF MICHIGAN GEOGRAPHY / 1962
C
33    FORMAT (18A4)
   34 FORMAT(3I2)
35    FORMAT (1H1,18A4)
36    FORMAT (/10X,I2,16H VECTORS OF DATA/10X,18H------------------)
37    FORMAT (5X,I4,8F14.2/(9X,8F14.2))
38    FORMAT (//10X,44HN, MEANS, AND STANDARD DEVIATIONS ARE ..... /)
39    FORMAT (1H1,2HI=,I2,3X,2HJ=,I2,5X,5HI I =,F10.4,3H + ,F10.4,6H * J
     1 ),5X,7HR(I,J)=,F10.4,5X,10HPCT SSRED=,F10.4,1H,,5X,I6,1X,6HPOINTS
     2///)
40    FORMAT (/30X,10HJTH VECTOR)
   41 FORMAT('    NVAR=',I3,4X,'INDEP=',I3,4X,'IDEP=',I3)
      END
MORTON'S DATA FOR ITALY
 7 3 5
(7F5.1)
14.9 159.0145.5141.0157.0147.0147.2
.6    62.2 38.7 41.7 35.4 34.6 32.8
8.3   90.7 60.7 47.7 42.5 44.6 39.5
9.7   88.6 69.5 56.8 41.7 39.8 39.8
31.0 150.8142.0131.0133.0130.0139.2
2.6   41.1 44.6 25.7 26.3 28.7 25.2
21.4 104.088.0 79.0 75.0 81.7 78.5
3.9   74.7 54.1 44.2 38.1 34.3 33.7
0.5   72.8 46.1 41.1 41.7 44.6 35.4
2.0   53.2 47.2 35.8 29.4 27.9 28.8
17.0 98.0 93.0 98.3 79.5 83.0 81.1
14.8 72.5 64.6 58.9 48.7 51.9 50.8
10.9 74.3 49.2 39.8 42.2 39.4 37.3
2.2   53.1 42.5 40.3 34.3 34.0 33.2
0.5   56.0 61.0 67.5 51.6 46.1 47.1
9.2   172.8120.0116.0123.9144.6132.6
8.3   90.7 60.7 47.7 44.1 44.6 39.5
16.3 111.877.4 60.5 56.8 58.9 54.0
15.7 90.8 69.0 53.5 49.2 44.1 40.0
6.2   83.8 53.0 41.9 38.8 38.8 35.4
4.9   109.368.5 49.9 44.4 43.1 41.6
11.0 64.7 50.1 41.0 37.3 34.4 34.4
3.1   86.0 64.8 54.3 57.5 42.2 34.2
```

GEOGRAPHICAL INTERPOLATION

Purpose: Interpolation to a square lattice from measures given at scattered geographical (x,y) positions.

Description: The assumption is made that the data are a sample taken from a continuous scalar field, and that values at unobserved locations can be estimated from the observed values. The second assumption clearly requires information concerning the shape of the two-dimensional autocorrelation function, as is discussed by Heiskanen and Moritz; equivalently, the applicability of the sampling theorem in two-dimensions should be recognized.

More specifically, the program establishes a lattice and estimates a value at each matrix point by using a weighted average of the six nearest data points. The weights are the inverses of the squares of the distances of the data points from the lattice point (i.e., linear interpolation). This weighted average is then averaged with the value at the nearest observation point (implying an autocorrelation which has a large negative slope in the vicinity of the origin) to give the final estimate. Alternate procedures are alternate hypotheses about nature.

Deck Make up:
1) First Control Card
2) Second Control Card
3) Title Card
4) Observations
5) Boundary Card (optional)

Data Cards: <u>First Control Card</u>

column		
1-3	N	Number of observations
4-71	FMT	Format for the observations

Second Control Card

1-3	ROWS	Number of rows in the output matrix (I3)
4-6	COLS	Number of columns in the output matrix (I3)
7-10	GSIZE	size of lattice in coordinate units (F4.0)
11	BND	1 to read specific boundaries; default uses the observations to calculate the boundaries.
12	LIST	1 to list original observations 2 to list estimated values 3 to do both of the above.
13	IPUN	1 to punch matrix of resulting values on unit 7.
14-15	NMAP	Number of maps requested. This option requires that subroutine SCON (see program GEOFIT) be called at line 149 of the program. The current listing does not include this option.

Comment: GSIZE, COLS, ROWS (in that precedence order) establish the size of the matrix. All may be left blank and the program will use a value which yields approximately as many lattice points as there are data points.

Title Card

Any title in columns 1-72.

Observations

X, Y, Z in that order punched as described in the format in the first control card.

Boundary Card

Used iff BND = 1 on the second control card: Reads XMAX, XMIN, YMAX, YMIN as 4F10.0 fields. These define a subregion of the data for which the interpolation is desired.

Programmer: W.R. Tobler

References: J.W. Goodman, Introduction to Fourier Optics, New York, McGraw-Hill, 1968, pp. 21-25.

W. Heiskanen, and H. Moritz, Physical Geodesy, San Francisco, Freeman, 1967 (Chapter 7).

H. Freeman, Discrete Time Systems, New York, J. Wiley, 1965, pp. 73-96.

B. Gilchrist, G. Cressman, "An Experiment in Objective Analysis", Tellus, 6, 4 (1954).

D. Shepard, " A Two Dimensional Interpolation Function for Irregularly-Spaced Data", Proceedings, 1968 ACM National Conference, pp. 517-524.

P. Switzer, "Reconstructing Patterns from Sample Data", Am. Math. Stat., 33(1967), pp. 138-154.

N. Wiener, Extrapolation, Interpolation, and Smoothing of Stationary Time Series, MIT Press, 1949.

```
C PROGRAM TO INTERPOLATE FROM SCATTERED VALUES TO A LATTICE
C READS NOBS+FORMAT, CONTROLS, TITLE, DATA (X, Y, Z)
C CONTROLS: (2I3,F4.0,3I1,I2) ARE (ROWS,COLS,GSIZE,BND,LIST,IPUN,NMAP)
C
C READS 5,WRITES 6, PUNCHES 7
      DIMENSION X(1000), Y(1000), W(1000), Z(100,100), D(20), FC
     1RM(10), TITL(12), A(6), B(6), C(6)
      DIMENSION FMT(17),NI(20)
      REAL MAXX,MINX,MAXY,MINY,MAXW,MINW,MAXZ,MINZ
      INTEGER TAPE,YES,BND,ROWS,COLS
       TAPE=7
  105 CONTINUE
      READ(5,99) N,(FMT(I),I=1,17)
 99   FORMAT(I3,17A4)
      READ(5,110) ROWS,COLS,GSIZE,BND,LIST,IPUN,NMAP
      READ(5,115) (TITL(I),I=1,12)
      DO 1 I=1,N
1     READ (5,FMT) X(I),Y(I),W(I)
      IF (LIST.LT.1.OR.LIST.GT.3) GO TO 5
      IF(LIST.NE.2) WRITE(6,120) (TITL(I),I=1,12)
    5 IF (IPUN.EQ.1) WRITE (7,125) (TITL(I),I=1,12)
      DO 10 I=1,N
   10 IF (LIST.EQ.1.OR.LIST.EQ.3) WRITE(6,130) I,X(I),Y(I),W(I)
      CALL STATS (N,1,1000,1,X,XBAR,DUM,XSIG,MAXX,MINX)
      CALL STATS (N,1,1000,1,Y,YBAR,DUM,YSIG,MAXY,MINY)
      CALL STATS (N,1,1000,1,W,WBAR,DUM,WSIG,MAXW,MINW)
      WRITE (6,135) (TITL(I),I=1,12),N
      IF (BND.NE.1) GO TO 15
      WRITE (6,140)
      READ (5,145) XMAX,XMIN,YMAX,YMIN
      GO TO 20
C
   15 WRITE (6,150)
      XMAX=MAXX
      XMIN=MINX
      YMAX=MAXY
      YMIN=MINY
   20 FN=MAXW-MINW
      WMAX=FN+MAXW
      WMIN=MINW-FN
      MAXZ=MAXW
      MINZ=MINW
      WRITE (6,155) NMAP,LIST,BND,TAPE
      M=0
      FN=FLOAT(N)
      DX=XMAX-XMIN
      DY=YMAX-YMIN
      AREA=DX*DY
      BEST=SQRT(AREA/FN)
      AVEDIS=BEST*1.07346
      WRITE (6,160) XBAR,XSIG,MAXX,MINX,YBAR,YSIG,MAXY,MINY,WBAR,WSIG,MA
     1XW,MINW,AREA,AVEDIS,BEST,GSIZE,ROWS,COLS
      IF (GSIZE.LE.0) GO TO 25
      COLS=(2.0+DX/GSIZE)
      ROWS=(2.0+DY/GSIZE)
```

```
      M=1
      GO TO 45
C
   25 IF (COLS.LE.1) GO TO 35
   30 GSIZE=DX/FLOAT(COLS-1)
      ROWS=(2.0+DY/GSIZE)
      M=1
      GO TO 45
C
35    IF (ROWS.LE.1) GO TO 40
      GSIZE=DY/FLOAT(ROWS-1)
      COLS=(2.0+DX/GSIZE)
      M=1
      GO TO 45
C
   40 GSIZE=BEST
      COLS=(2.0+DX/GSIZE)
      ROWS=(2.0+DY/GSIZE)
   45 CONTINUE
      IF (COLS.LE.100) GO TO 50
      COLS=100
      GO TO 30
C
   50 WRITE (6,165)
      IF (M.EQ.1) WRITE (6,170)
      IF (M.NE.1) WRITE (6,175)
      WRITE (6,180) XMAX,XMIN,YMAX,YMIN,GSIZE,ROWS,COLS
      IF (LIST.EQ.2.OR.LIST.EQ.3) WRITE (6,185) (TITL(I),I=1,12)
      DY=(DY-(ROWS-1.)*GSIZE)/2.0
      DX=(DX-(COLS-1.)*GSIZE)/2.0
      YMAX=YMAX-DY
      XMIN=XMIN+DX
      HGRID=GSIZE/25.0
      FN=(WMAX-MAXW)/2.0
      MAXX=MAXW+FN
      MINX=MINW-FN
       WRITE(6,199)
      DO 95 M=1,ROWS
         YP=YMAX-(FLOAT(M-1))*GSIZE
         DO 90 J=1,COLS
            XP=XMIN+(FLOAT(J-1))*GSIZE
            DO 55 I=1,9
   55       D(I)=1.E10
            N1(1)=1
            N1(2)=1
            DO 75 I=1,N
               DIST=((XP-X(I))**2)+((YP-Y(I))**2)
               IF (DIST.GE.HGRID) GO TO 60
               Z(M,J)=W(I)
               GO TO 85
C
60    K=I
      IF(K-8) 61,61,62
61    YES=K
      GO TO 63
62    YES=8
```

```
63    DO 70 K=1,YES
                 IF (DIST.GE.D(K)) GO TO 70
                 DO 65 JJ=K,YES
                    KK=YES+K-JJ
                    MM=KK+1
                    D(MM)=D(KK)
   65          N1(MM)=N1(KK)
               D(K)=DIST
               N1(K)=I
               GO TO 75
C
   70 CONTINUE
   75 CONTINUE
      DUM1=0.0
      PSUM=0.0
      DO 80 I=1,6
      YES=N1(I)
      MAXW=1.0/(SQRT(D(I)))
      PSUM=PSUM+MAXW
   80 DUM1=DUM1+W(YES)*MAXW
      DUM6=DUM1/PSUM
      N11=N1(1)
      DUM2=W(N11)
      DUM=(DUM2+DUM6)/2.0
      IF (DUM.GT.MAXX.OR.DUM.LT.MINX) Z(M,J)=DUM2
      IF (DUM.LE.MAXX.AND.DUM.GE.MINX) Z(M,J)=DUM
   85 CONTINUE
   90 CONTINUE
      IF (LIST.EQ.2.OR.LIST.EQ.3) WRITE(6,190) (Z(M,I),I=1,COLS)
      IF (IPUN.EQ.1) WRITE(7,195) (Z(M,I), I=1,COLS)
   95 CONTINUE
      NEWTOT=ROWS*COLS
      CALL STATS (ROWS,COLS,100,100,Z,ZBAR,XP,ZSIG,ZMAX,ZMIN)
      YMIN=YP
      WRITE (6,200) NEWTOT,XMIN,YMIN,ZMAX,ZMIN,ZBAR,ZSIG
      WRITE (6,205)
      IF (NMAP.LT.1) GO TO 105
C
  110 FORMAT (2I3,F4.0,3I1,I2)
 115  FORMAT(12A4)
120   FORMAT(1H1,12A4,//,3X,24H COORDINATE OBSERVATIONS
125   FORMAT(6X,12A4)
  130 FORMAT (5X,4H I =,I3,7H X(I) =,F10.3,7H Y(I) =,F10.3,7H W(I) =,F10
     1.3)
135   FORMAT(1H1,12A4,//,3X,I5,13H OBSERVATIONS   ,//)
  140 FORMAT (3X,25H BOUNDARIES PREDETERMINED)
  145 FORMAT (4F10.0)
  150 FORMAT (3X,32H BOUNDARIES CALCULATED FROM DATA)
  155 FORMAT (3X,6HNMAP =,I3,7H LIST =,I3,6H BND =,I2,7H TAPE =,I3)
  160 FORMAT (3X,15HOBSERVED VALUES,//,3X,6HXBAR =,F10.3,6HXSIG =,F10.3,
     16HMAXX =,F10.3,6HMINX =,F10.3,//3X,6HYBAR =,F10.3,6HYSIG =,F10.3,6
     2HMAXY =,F10.3,6HMINY =,F10.3,//3X,6HWBAR =,F10.3,6HWSIG =,F10.3,6H
     3MAXW =,F10.3,6HMINW =,F10.3,//3X,6HAREA =,F10.3,8HAVEDIS =,F10.3,6
     4HBEST =,F10.3,//3X,7HGSIZE =,F10.3,6HROWS =,I4,6HCOLS =,I4,/)
165   FORMAT(3X, ' ROWS,COLUMNS, OR GRID SIZE ' )
170   FORMAT(3X,' SPECIFIED A PRIORI ' )
```

```
175   FORMAT(3X, ' CALCULATED FROM THE DATA ', / )
  180 FORMAT(3X,16H VALUES EMPLOYED,/,3X,7H XMAX =,F10.3,3X,7H XMIN =,F
     110.3,3X,7H YMAX =,F10.3,7H YMIN =,F10.3,/,3X,9H GSIZE =,F10.3,3X,7
     2H ROWS=,I4,8X, 7H COLS =,I4)
185   FORMAT(1H1,12A4,//,3X,20H INTERPOLATED VALUES     )
  190 FORMAT (1H0,(10F10.3))
  195 FORMAT(8F10.3)
199   FORMAT('1RESULTING VALUES ')
200   FORMAT(1H0,3X,3HN =,I3,6HXMIN =,F10.3,3X,6HYMIN =,F10.3,//,4X,6HZM
     1AX =,F10.3,3X,6HZMIN =,F10.3,3X,6HZBAR =,F10.3,3X,6HZSIG =,F10.3 )
  205 FORMAT(13H1 END OF DATA)
      END
      SUBROUTINE STATS(ROWS,COLS,II,JJ,X,XBAR,XVAR,XSIG,XMAX,XMIN)
C                              SUBROUTINE STATS
C     ------------------------------------------------------------------
C     PURPOSE
C     TO CALCULATE STATISTICAL MEASURES OF ONE OR TWO DIMENSIONAL DATA
C     ARRAYS. MEASURES DETERMINED ARE 1)MEAN,2)VARIANCE,3)STANDARD DEVIA
C     -TION,4)MAXIMUM VALUE,5)MINIMUM VALUE.
C     ------------------------------------------------------------------
C     USAGE -
C
C          CALL STATS(ROWS,COLS,II,JJ,X,XBAR,XVAR,XSIG,XMAX,XMIN)
C        ROWS - NUMBER OF ROWS IN ARRAY BEING ANALYZED.
C        COLS - NUMBER OF COLUMNS IN ARRAY BEING ANALYZED.
C     II - MAXIMUM NUMBER OF ROWS ALLOWED FOR ARRAY BEING ANALYZED.
C     JJ - MAXIMUM NUMBER OF COLUMNS ALLOWED FOR ARRAY BEING ANALYZED.
C     X - DATA ARRAY BEING ANALYZED.
C     XBAR - MEAN OF ARRAY
C     XVAR - VARIANCE OF ARRAY
C     XSIG - STANDARD DEVIATION OF ARRAY.
C     XMAX - MAXIMUM VALUE OF ARRAY.
C     XMIN - MINIMUM VALUE OF ARRAY.
C     ------------------------------------------------------------------
C
C
      INTEGER ROWS,COLS
      DIMENSION X(II,JJ)
      FNO=ROWS*COLS
      XSQR=0.0
      XSUM=0.0
      XMAX=X(1,1)
      XMIN=X(1,1)
      DO 5 I=1,ROWS
      DO 5 J=1,COLS
         IF (X(I,J).GT.XMAX) XMAX=X(I,J)
      IF (X(I,J).LT.XMIN) XMIN=X(I,J)
         XSUM=XSUM+X(I,J)
5     XSQR=XSQR+X(I,J)*X(I,J)
      XBAR=XSUM/FNO
      XVAR=(XSQR-2.0*XBAR*XSUM+FNO*XBAR*XBAR)/FNO
      XSIG=SQRT(XVAR)
      RETURN
      END
```

MAP PLOTTING AND CONTOURING

Purpose The program produces isarithmic maps on the line printer from scattered observations by first interpolating to a lattice and then contouring these regular values. A location map of the observations, a list of the values at the grid intersections, and the contoured map are returned.

Description: For each lattice point the program examines every observation, deleting from consideration all but the nearest NIPT (<10), regardless of direction from the grid point. The only exception to this search procedure occurs when an observation within IMIN (real) of the point is encountered, at which time its value is accepted as the lattice value and no further observations are tested. In the general case the value assigned to the lattice point is

$$\frac{1}{2}\left[z_c + \frac{\sum_i z_i/d_i}{\sum_i 1/d_i}\right]$$

where z_c is the value of the nearest observation, d is distance, and the sums are carried from the nearest to the NIPTth nearest observation. The lattice values are then printed row by row in a table, after which it is possible to terminate the program.

The location map and contour map are constructed simultaneously. Values appropriate to the latter are stored in a scratch file for later printing, while the contents of the former are printed as the contouring algorithm is executed. This routine determines values along horizontal lines by linear interpolation of the values at the bounding intersections. The values of the printer elements within each row of cells are interpolated linearly from the appropriate elements of the two bounding horizontal lines.

The location map shows grid intersections by ticks and the observations by asterisks followed by the Z values. The contour map prints a different symbol for each contour interval and denotes observation locations by asterisks alone. Both have X and Y scales printed in the margins.

Comments: Within the restrictions that the observations be given in rectangular coordinates and that the map produced be isarithmic the user has great flexibility. The spacing of ticks on the map is fixed both horizontally and vertically at one inch, with the result that a large map is produced in sections of twelve inch width and unlimited length. The size of the map is determined by the user through the variable SCALE. This is the interval along both the X and Y axes which will correspond to one inch on the map. This value should be chosen with care, as it is in the interpolation of grid values that much time is consumed. The program examines each of the N observations, regardless of the section of the map into which it falls, to find the NIPT nearest observations to each grid point. SCALE may be chosen as zero, the default value, in which case the program assigns to it the value of

$$\left[(X_{max} - X_{min}) * (Y_{max} - Y_{min}) / N\right]^{1/2}$$

a rough approximation of the number of map cells to the number of observations.

The user may read in the maxima and minima of the X, Y, and Z values over which the map is to range, or he may default to the extrema of the observations. He may request a certain contour interval; default is one tenth the range of Z, whether read in or determined by the program. If the interval chosen yields more than 37 contours the Z increment is doubled. The user may state the maximum distance from a grid intersection at which an observation's Z value is acceptable for the grid value. It is advisable to make this distance DMIN small relative to SCALE; the default is SCALE/25. He may also define NIPT as the number of nearest neighboring observations from which the grid values are to be interpolated. Maximum is nine, default is six. There are also options to suppress printing of certain output features and to punch the values at the lattice points.

The isarithmic map comes in one of three styles. The user may choose to have the entire field filled with symbols, a completely contoured map with different symbols representing different Z intervals; contours are denoted by a change of symbol. Or alternate contour intervals may be suppressed, leaving white bands between the printed intervals. This will usually improve the visual impact. The third choice is an actual isarithmic map, although the program does not calculate points along the line but uses instead the lowest ring of characters in each interval as an approximation to the contour. All maps have the locating asterisks for the observations.

I/O units 8 and 9 are used for storage during execution and should be set to temporary files. Unit 7 is used for punch output; if no cards are desired set this to a temporary file. The first control card reads the units to be used for input and output. These are usually 5 and 6. 500 observations can be handled by current version of the program.

Deck Make-up: The main program reads two control cards and then the observations:

1) I/O Card

2) Variable Format Card

3) Observations

The subroutine Grid reads three more cards, Grid Cards I, II, and III.

Card Formats: <u>I/O Card</u>

1-4	N	Number of observations
8	INPUT	I/O unit to be used for input (reading)
12	OUTPUT	I/O unit to be used for output (writing)

<u>Variable Format Card</u>

This describes the N groups of X, Y, and Z values. As many of these triples may be placed on the same observation card as desired; the format card will include between parentheses a description of the arrangement.

Observation deck: X, Y, Z values

Grid Card I: This reads an alphanumeric title of fewer than 81 spaces.

Grid Card II:

1-10	SCALE	Data units per lattice cell (floating point); default: leave blank
11-22	UNITS	As many as 12 characters describing the units of the observation measurement, e.g., as "inches."

The following eight options are available; a 1 selects the option and a 0 or blank suppresses it:

23	PIN	Print a table of input values
24	PMXMN	Do not print input extrema
25	PGRID	Do not print the gridded values
26	PHGRID	Punch gridded values (0 is recommended)
27	PLMAP	Do not print the location map
28	PCMAP	Do not print the contour map
29	MTYPE	Map style: 0 is alternate bands, 1 is contour lines, and 2 is all contour bands.
30	PCNTRL	Print control values

The following four values are floating point; default is blanks:

31-40	MAXX	Right margin value
41-50	MINX	Left margin value
51-60	MAXY	Top margin value
61-70	MINY	Bottom margin value
72	IFLIM	If the margin values have been read in, set to 1; if blank and defaulted

to extrema of the observations, set to 0

Grid Card III:		The first four values are floating point, while the last two are integer.
1-10	MAXZ	Highest contour altitude
11-20	MINZ	Lowest contour altitude
21-30	ZINC	Contour interval; default leave blank
31-40	DMIN	Maximum acceptance distance for grid point; default leave blank
42	NIPT	Number of interpolation points; default leave blank
44	IFZLIM	If the Z extrema have been read in, set to 1; if not and the default is chosen, set to 0

Programmer: Original program (GRID, SCON, STATS, in Mad) by Waldo R. Tobler, The University of Michigan, 1965. Modifications and translation to FORTRAN IV by D. Gill, The University of Michigan, May 1967. Further modifications by D. Bowman and D. Rhynsburger, 1969.

```
C                  ISARITHMIC MAPPING
      DIMENSION X(500),Y(500),Z(500),FMT(18)
      INTEGER OUTPUT
  101 FORMAT(' NUMBER OF DATA POINTS (N) = ',I3,' INPUT UNIT = "',I1,'"
     1OUTPUT UNIT = "',I1,'"')
  102 FORMAT (18A4)
  103 FORMAT(1X,'DATA IS IN: SUBROUTINE WILL NOW BE CALLED')
  104 FORMAT(1X,10(2X,E12.4))
  105 FORMAT('1')
  106 FORMAT(' DATA FORMAT IS: ',18A4)
  107 FORMAT (20I4)
    2 READ (5,107,END=3)  N, INPUT, OUTPUT
      WRITE(6,101) N,INPUT,OUTPUT
      IF (N) 3,3,4
 4    READ(INPUT,102) (FMT(I),I=1,18)
      WRITE(6,106)(FMT(I),I=1,18)
      IF (INPUT)  10,10,11
10    INPUT=5
      OUTPUT=6
 11   READ(5,FMT) (X(I),Y(I),Z(I),I=1,N)
      WRITE(6,104)(X(I),Y(I),Z(I),I=1,N)
      WRITE(6,103)
      CALL GRID (N,X(1),Y(1),Z(1),INPUT,OUTPUT)
      GO TO 2
3     WRITE(6,105)
      CALL SYSTEM
      END
      SUBROUTINE GRID(N,X,Y,Z,INPUT,OUTPUT)
      IMPLICIT REAL(I-N)
      DIMENSION DELV(45), DUMMY(2), INDEX(1000), INEXT(130), IPREV(130)
      DIMENSION PSYM(131), PSYMA(131), TITLE(20), UNITS(3), VNOW(45)
      DIMENSION X(1), Y(1), Z(1), ZROW1(13)
      DATA DUMMY/'A','B'/, ASTRK/'*'/, BLANK/' '/, MINUS/'-'/
      DATA NINE/'9'/, PLUS/'+'/, VBAR/'|'/
      INTEGER COL, COLS, COLSM1, CON, I, INDEX, INEXT, INPUT, IPREV
      INTEGER J, K, M, N, NCOL, NIPT, NSECT, NSECTC, NUM, N1, SCRCH
      INTEGER OUTPUT, ROWS, ROWSM1, TPS, NIP, IFLIM, IFZLIM
      REAL SYMTAB(36)/'0','1','2','3','4','5','6','7','8','9','A','B',
     1 'C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R',
     1  'S','T','U','V','W','X','Y','Z'/
101   FORMAT (E10.0,3A4,8F1.0,4E10.0,I2)
102   FORMAT(1H-54X,'INPUT CHARACTERISTICS ',1X/1H033X,'COORDINATE',19X,
     1'MAXIMUM',18X,'MINIMUM',1X/1H038X,'X',F30.3,F25.3/1H038X,'Y',F30.3
     2,F25.3/1H038X,'Z',F30.3,F25.3)
103   FORMAT ((1X,3(F20.3,2F10.3)))
105   FORMAT (20A4)
106   FORMAT (E15.3,2I5)
107   FORMAT('0',60X,'ROW',I5/1X/(6(I7,E14.5)))
108   FORMAT(1H045X,'CONTOUR SYMBOL TABLE',1X/1H042X,'SYMBOL',15X,'VALUE
     1')
109   FORMAT (8F10.3)
110   FORMAT (4E10.0,2I2)
111   FORMAT(' CONTOUR INCREMENT OF ',E10.2,'  IS TOO SMALL.  IT WILL BE
     1 DOUBLED')
112   FORMAT (132A1)
```

```
113   FORMAT (1H1,25X,20A4)
114   FORMAT('0',45X,'-',17X,'BELOW',6X,F12.4)
115   FORMAT(46X,A1,10X,F12.4,3X,'TO',F13.4)
116   FORMAT (46X,A1,10X,F12.4)
117   FORMAT(46X,'+',17X,'ABOVE',6X,F12.4)
118   FORMAT (1H0,59X,12HINPUT VALUES  )
119   FORMAT (1H0,3(15X,1HX,9X,1HY,9X,1HZ,4X))
120   FORMAT (1H0,6(3X'COL'7X'VALUE'3X))
121   FORMAT('0',25X,'LOCATION MAP..SCALE IS',E12.4,2X,3A4,'PER INCH')
122   FORMAT('0',25X,'CONTOUR MAP....SCALE IS',E12.4,2X,3A4,'PER INCH')
123   FORMAT('OMAXX = ',E12.5,4X,'MINX= ',E12.5,4X,'MAXY = 'E12.5,4X,'MI
     1NY = ', E12.5,4X,'MAXZ = ',F12.5,4X,'MINZ = ',E12.5)
124   FORMAT('OSCALE IS ',E12.5,1X,3A4,' PER INCH. ALTITUDE INCREMENT IS
     1 ',E12.5,'.  MAXIMUM ACCEPTANCE DISTANCE IS ',E12.5)
125   FORMAT('OOPTIONS ARE:  PRINT INPUT VALUES (',F2.0,')  SUPPRESS CHA
     1RACTERISTICS TABLE (',F2.0,')  SUPPRESS PRINTING OF GRIDDED VALUES
     2 (',F2.0,')')
126   FORMAT('OPUNCH GRIDDED VALUES (',F2.0,')  SUPPRESS LOCATION MAP  (
     1',F2.0,')  SUPPRESS CONTOUR MAP  (',F2.0,')  CONTOUR MAP TYPE  (',
     2F2.0,')')
127   FORMAT('ONUMBER OF INTERPOLATION POINTS  (',F2.0,')  PRINT CONTROL
     1 VALUES  (',F2.0,')  NUMBER OF DATA POINTS  (',I3,')')
128   FORMAT('0',25X,'SECTION ',I3,' OF ',I3)
129   FORMAT('0',57X,'INITIAL CONTROL INPUT')
130   FORMAT('0',57X,'FINAL CONTROL VALUES')
131   FORMAT('ONUMBER OF:  INTERPOLATION POINTS  (',I1,')   VERTICAL MAP
     1 CELLS  (',I2,')  HORIZONTAL MAP CELLS  (',I2,')   MAP VERTICAL S
     2ECTIONS  (',I2,')')
133   FORMAT (1H ,I2,'INTERPOLATION POINTS IS TOO MANY. NINE WILL BE USE
     1D')
134   FORMAT(' ',I1,' DATA POINTS ARE TOO FEW.  RETURN FROM GRID.')
      READ (INPUT,105) TITLE
      READ (INPUT,101) SCALE,UNITS,PIN,PMXMN,PGRID,PHGRID,PLMAP,PCMAP,
     1  MTYPE,PCNTRL,MAXX,MINX,MAXY,MINY, IFLIM
      READ (INPUT,110) MAXZ, MINZ,ZINC,IMIN,NIPT,IFZLIM
      IF (PCNTRL.EQ.0.) GO TO 76
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,129)
      WRITE (OUTPUT,123) MAXX,MINX,MAXY,MINY,MAXZ,MINZ
      WRITE (OUTPUT,124) SCALE,UNITS,ZINC,IMIN
      WRITE (OUTPUT,125) PIN,PMXMN,PGRID
      WRITE (OUTPUT,126) PHGRID,PLMAP,PCMAP,MTYPE
      WRITE (OUTPUT,127) NIPT,PCNTRL,N
   76 IF (PIN.EQ.0.) GO TO 4
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,118)
      WRITE (OUTPUT,119)
      WRITE (OUTPUT,103) (X(I),Y(I),Z(I),I=1,N)
C                               FIND MAXIMA AND MINIMA
    4 IF (IFLIM .EQ. 1)  GO TO 501
      XMAX=X(1)
      XMIN=XMAX
      YMAX=Y(1)
      YMIN=YMAX
      DO 3 I=2,N
      IF (XMAX.LT.X(I)) XMAX=X(I)
```

```
      IF (YMAX.LT.Y(I)) YMAX=Y(I)
      IF (XMIN.GT.X(I)) XMIN=X(I)
      IF (YMIN.GT.Y(I)) YMIN=Y(I)
      CONTINUE
      GC TO 502
  501 XMAX = MAXX
      XMIN = MINX
      YMAX = MAXY
      YMIN = MINY
  502 IF (IFZLIM .EQ. 1)  GO TO 504
      ZMAX=Z(1)
      ZMIN=ZMAX
      DO 503 I = 2,N
      IF (ZMAX.LT.Z(I)) ZMAX=Z(I)
      IF (ZMIN.GT.Z(I)) ZMIN=Z(I)
  503 CONTINUE
      GC TO 505
  504 ZMAX = MAXZ
      ZMIN = MINZ
  505 IF (PYXMN .NE. 0)  GO TO 7
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,102) XMAX,XMIN,YMAX,YMIN,ZMAX,ZMIN
C                             CALCULATE CONTOUR INTERVALS
    7 DZ=ZMAX-ZMIN
      IF (ZINC.EQ.0.) ZINC=DZ/10.
   30 CCN=DZ/ZINC
      IF (CON.LE.36) GO TO 29
      WRITE (OUTPUT,111) ZINC
      ZINC=2.*ZINC
      GC TO 30
C                             CALCULATE CONSTRUCTION PARAMETERS
C                             NCOL IS TOTAL NUMBER OF HCRIZONTAL CELLS
C                             NSECT IS TOTAL NUMBER OF VERTICAL MAP
C                               SECTIONS PRODUCED
C                             NSECTC DETERMINES WHICH SECTION WE'RE IN
C                             ROWS IS VERTICAL NUMBER OF CELL BCUNDARIES
C                             ROWSM1 IS VERTICAL NUMBER OF CELLS
   29 IF (SCALE.EQ.0.) SCALF=SQRT((XMAX-XMIN)*(YMAX-YMIN)/N)
      NCOL=(XMAX-XMIN)/SCALE+.99
      NSECT=(NCOL-1)/12+1
      NSECTC=0
      ROWS=(YMAX-YMIN)/SCALE +1.99
      ROWSM1=ROWS-1
      MAXX=XMIN
      IF (IMIN.EQ.0.) IMIN=SCALE/25.
      IMIN = IMIN * IMIN
      IF (NIPT.EQ.0) NIPT=6
      IF (NIPT.LT.10) GO TO 12
      WRITE (OUTPUT,133) NIPT
      NIPT=9
   12 IF (N.GE.NIPT) GO TO 80
      WRITE (OUTPUT,134) N
      RETURN
C                             CYCLIC SETUP
C                             INDEX IS AN ARRAY WHICH CORRESPONDS TO
C                             THE INPUT DATA ARRAY.  VALUES MEAN:
```

```
C                             1 - DATA POINT IS IN CURRENT SECTION
C                             0 - DATA POINT IS IN A FUTURE SECTION
C                            -1 - DATA POINT IS IN A PREVIOUS SECTION
   80 DO 77 I=1,N
   77 INDEX(I)=0
      XINC=3.
      YINC=5.
      PS=30.
      IF (PCNTRL.EQ.0.) GO TO 79
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,130)
      WRITE (OUTPUT,123) XMAX,XMIN,YMAX,YMIN,ZMAX,ZMIN
      WRITE (OUTPUT,124) SCALE,UNITS,ZINC,IMIN
      WRITE (OUTPUT,131) NIPT,ROWSM1,NCOL,NSECT
C                             HERE BEGINS THE CYCLIC PROCEDURE WHICH
C                             PRODUCES VERTICAL MAP SECTIONS
   79 CALL REWIND(8)
      CALL REWIND(9)
C                             DECIDE MAP WIDTH
C                             COLS IS THE HORIZONTAL NUMBER OF CELL
C                             BOUNDARIES IN THE CURRENT SECTION
C                             COLSM1 IS THE HORIZONTAL NUMBER OF CELLS
C                             IN THE CURRENT SECTION
C                             NUM IS WHERE MARGIN VALUE PRINTING STARTS
      IF (NCOL.GT.12) GO TO 13
      COLS=NCOL+1
      NCOL=0
      GO TO 14
   13 COLS=13
      NCOL=NCOL-12
   14 COLSM1=COLS-1
      NUM=10*COLSM1+4
      IF (PHGRID.EQ.0.) GO TO 1
      WRITE(7,105) TITLE
      WRITE(7,106) SCALE,ROWS,COLS
C                             FIND HORIZONTAL MAX AND MIN FOR
C                             THIS SECTION
    1 MINX=MAXX
      MAXX=MINX+(COLS-1)*SCALE
C                             ELIMINATE OUT-OF-SECTION POINTS
      DO 24 I=1,N
      IF (INDEX(I).LT.0) GO TO 24
      IF (X(I).LE.MAXX.AND.X(I).GE.MINX.AND.Y(I).LE.YMAX.AND.Y(I).GE.
     1   YMIN) INDEX(I)=1
   24 CONTINUE
      IF (PGRID.NE.0.) GO TO 78
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,120)
C                             SET UP ITERATION WHICH DOES GRIDDING.
C                             FOR EACH GRID ROW:
   78 YP=YMAX
      NIP = NIPT + 1
      DO 15 M=1,ROWS
C                             FOR EACH GRID COLUMN WITHIN THE ROW
C                             XP AND YP ARE COORDINATES OF GRID POINT
      XP=MINX
```

```
      DC 16 J=1,COLS
      DXM = 1.E10
      DYM = 1.E10
      DO 20 I = 1,NIP
   20 DELV(I) = 1.E10
C                                    FIND 'NIPT' DATA POINTS NEAREST GRID POINT
      DC 19 I=1,N
      TEMP=XP-X(I)
      TEMP1=YP-Y(I)
      IF (TEMP .GT. DXM .AND. TEMP1 .GT. DYM)  GO TO 19
      DIST=TEMP*TEMP+TEMP1*TEMP1
      K = NIP
      IF (DIST .GE. IMIN)  GO TO 21
      ZROW1(J)=Z(I)
      GO TO 25
   23 K=K-1
      DELV(K+1)=DELV(K)
      IPREV(K+1)=IPREV(K)
   21 IF (K.EQ.1) GO TO 22
      IF (DIST.LT.DELV(K-1) ) GO TO 23
      IF (K .NE. NIPT)  GO TO 22
      DXM = TEMP
      DYM = TEMP1
   22 DELV(K)=DIST
      IPREV(K)=I
19    CONTINUE
C                                    INTERPOLATE WITH 'NIPT' NEAREST POINTS
      TEMP=0.0
      TEMP1=0.0
      DO 17 I=1,NIPT
      TEMP2=1./SQRT(DELV(I))
      TEMP1=TEMP1+TEMP2
   17 TEMP=TEMP+Z(IPREV(I))*TEMP2
      ZROW1(J)=(Z(IPREV(1))+TEMP/TEMP1)/2.
   25 XP=XP+SCALE
16    CONTINUE
      YP=YP-SCALE
      IF (PHGRID .EQ. 1)  WRITE (7,109) (ZROW1(I),I=1,COLS)
      IF (PGRID .EQ. 0)  WRITE (OUTPUT,107) M, (I, ZROW1(I),I=1,COLS)
      IF (PLMAP.NE.0..AND.PCMAP.NE.0.) GO TO 15
      WRITE(8,109) (ZROW1(I),I=1,COLS)
15    CONTINUE
C
C                                    DONE WITH GRIDDING
      IF (PLMAP.NE.O..AND.PCMAP.NE.0.0) GO TO 72.
      NSECTC=NSECTC+1
      IF (PLMAP.NE.0.) GO TO 74
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,121) SCALE,UNITS
      WRITE (OUTPUT,128) NSECTC,NSECT
   74 IF (PCMAP.NE.0.) GO TO 75
      WRITE(9,113) TITLE
      WRITE(9,122) SCALE, UNITS
      WRITE(9,128) NSECTC, NSECT
C                                    SET UP THE TOP ROW OF NUMBERS
   75 DO 32 I=1,131
```

```
      PSYM(I)=BLANK
   32 PSYMA(I)=BLANK
      TEMP=MINX
      DO 31 I=1,COLS
      K=10*(I-1)+1
      CALL BNBCD(TEMP,PSYM(K) )
   31 TEMP=TEMP+SCALE
      IF (PLMAP .EQ. 0)  WRITE (OUTPUT,112) MINUS, PSYM
      IF (PCMAP .EQ. 0)  WRITE (9,112) MINUS, PSYM
C                                SET UP THE TOP MARGIN
      PSYM(1)=BLANK
      DO 38 I=1,NUM,10
      PSYM(I+1)=VBAR
      DO 38 J=2,10
   38 PSYM(I+J)=BLANK
      IF (PLMAP .EQ. 0)  WRITE (OUTPUT,112) NINE, PSYM
      IF (PCMAP .EQ. 0)  WRITE (9,112) NINE, PSYM
C                                SET UP THE FIRST ROW
      CALL REWIND(8)
      READ(8,109) (ZROW1(I),I=1,COLS)
      COL=1
      DO 36 J=1,COLSM1
      DELH=ZROW1(J+1)-ZROW1(J)
      VNOW(J)=ZROW1(J)
      XP=-XINC
   37 XP=XP+XINC
      IF (XP.GE.PS) GO TO 36
      COL=COL+1
      INEXT(COL) = (ZROW1(J) + XP * DELH / PS - ZMIN) / ZINC + 1.
      PSYM(COL)=BLANK
      PSYMA(COL)=BLANK
      GO TO 37
36    CONTINUE
      COL=COL+1
      VNOW(COLS)=ZROW1(COLS)
      INEXT(COL) = (ZROW1(COLS) - ZMIN) / ZINC + 1.
      PSYM(COL)=BLANK
      PSYMA(COL)=BLANK
      YVAL=YMAX
C                                DETERMINATION OF Z VALUES FOR INTERIOR
      DO 42 M=1,ROWSM1
      READ(8,109) (ZROW1(I),I=1,COLS)
C                                CALCULATE ALTITUDE DIFFERENCES
C                                BETWEEN ROWS
      DO 40 I=1,COLS
   40 DELV(I)=ZROW1(I)-VNOW(I)
      PSYM(1)=MINUS
      PSYMA(1)=MINUS
      K=NUM-1
      DO 57 I=2,K,10
   57 PSYMA(I)=PLUS
      PSYMA(NUM-1)=MINUS
      PSYM(NUM-1)=MINUS
      CALL BNBCD(YVAL,PSYM(NUM) )
      CALL BNBCD(YVAL,PSYMA(NUM) )
C                                SET UP INTERMEDIATE ROW INCREMENTS
```

```
      YP=-YINC
   41 YP=YP+YINC
      IF (YP.GE.PS) GO TO 42
      COL=1
      VNOW(1)=YINC*DELV(1)/PS+VNOW(1)
C                              FOR EACH GRID COLUMN
      DO 43 J=1,COLSM1
      VNOW(J+1)=YINC*DELV(J+1)/PS+VNOW(J+1)
      DELH=VNOW(J+1)-VNOW(J)
C                              SET UP INTERMEDIATE COLUMN INCREMENTS
      XP=-XINC
   45 XP=XP+XINC
      IF (XP.GE.PS) GO TO 43
      IPREV(COL)=I
      COL=COL+1
      I=INEXT(COL)
      INEXT(COL) = (VNOW(J) + XP * DELH / PS - ZMIN) / ZINC + 1.
C                              CHOOSE THE PRINT SYMBOL
      IF (I.GE.1) GO TO 46
      PSYM(COL) =MINUS
      GO TO 45
   46 IF (I.LE.CON) GO TO 47
      PSYM(COL)=PLUS
      GO TO 45
   47 PSYM(COL)=SYMTAB(I)
      IF ((M.EQ.1.AND.YP.EQ.0.).OR.COL.EQ.2) GO TO 45
      IF (MTYPE-1.) 49,48,45
   49 IF (I/2*2.EQ.I) PSYM(COL)=BLANK
      GO TO 45
   48 IF (I.GT.IPREV(COL).OR.I.GT.IPREV(COL-1).OR.I.GT.INEXT(COL+1).OR.
     1  I.GT.INEXT(COL)) GO TO 45
      PSYM(COL)=BLANK
      GO TO 45
43    CONTINUE
      IPREV(COL)=I
      COL=COL+1
      I=INEXT(COL)
      INEXT(COL) = (VNOW(COLS) - ZMIN) / ZINC + 1.
      IF (I.GE.1) GO TO 60
      PSYM(COL)=MINUS
      GO TO 61
   60 IF (I.LE.CON)  GO TO 50
      PSYM(COL)=PLUS
      GO TO 61
   50 PSYM(COL)=SYMTAB(I)
   61 COL=1
   62 DO 51 I=1,N
      IF (INDEX(I).LE.0) GO TO 51
      IF (ABS(Y(I)-YVAL).GT.SCALE/12.) GO TO 51
      INDEX(I)=-1
      J=10.*(X(I)-MINX)/SCALE+2.5
      PSYM(J)=ASTRK
      PSYMA(J) = ASTRK
      CALL BNBCD(Z(I),PSYMA(J+1))
51    CONTINUE
      YVAL=YVAL-SCALE/6.
```

```
      IF (PLMAP .EQ. 0)  WRITE (OUTPUT,112) NINE, PSYMA
      IF (PCMAP .EQ. 0)  WRITE (9.112) NINE, PSYM
      DO 52 I=1,131
      PSYM(I)=BLANK
   52 PSYMA(I)=BLANK
      GO TO (41,58),COL
42    CONTINUE
C                                 END INTERIOR COMPUTATIONS
      K=NUM-2
      DO 55 COL=2,K
      I=INEXT(COL)
      IF (I.GE.1) GO TO 54
      PSYM(COL)=MINUS
      GO TO 55
   54 IF (I.LE.CON) GO TO 56
      PSYM(COL)=PLUS
      GO TO 55
   56 PSYM(COL)=SYMTAB(I)
55    CONTINUE
      FVAL = YMAX - SCALE * FLOAT(ROWSM1)
      PSYM(1)=MINUS
      PSYMA(1)=MINUS
      PSYM(NUM-1)=MINUS
      PSYMA(NUM-1)=MINUS
      CALL BNBCD(FVAL,PSYM(NUM))
      CALL BNBCD(FVAL,PSYMA(NUM))
      DO 73 I=2,K,10
   73 PSYMA(I)=PLUS
      COL=2
      GO TO 62
   58 DO 59 I=2,K,10
   59 PSYM(I)=VBAR
      IF (PLMAP .EQ. 0)  WRITE (OUTPUT,112) NINE, PSYM
      IF (PCMAP .EQ. 0)  WRITE (9.112) NINE, PSYM
      IF (PCMAP.NE.0.) GO TO 72
C                                 NOW PRINT OUT THE CONTOUR MAP
      CALL REWIND(9)
      WRITE (OUTPUT,113) TITLE
      WRITE (OUTPUT,108)
      WRITE (OUTPUT,114) ZMIN
      TEMP=ZMIN
      IF (MTYPE.EQ.1.) GO TO 34
      DO 35 I=1,CON
      TEMP1=TEMP+ZINC
      WRITE (OUTPUT,115) SYMTAB(I),TEMP,TEMP1
   35 TEMP=TEMP1
      GO TO 9
   34 DO 8 I=1,CON
      WRITE (OUTPUT,116) SYMTAB(I),TEMP
    8 TEMP=TEMP+ZINC
    9 WRITE (OUTPUT,117) TEMP
   63 READ(9,112,END=72) TEMP,PSYM
      WRITE (OUTPUT,112) TEMP,PSYM
      GO TO 63
C                                 SET UP FOR THE NEXT MAP SECTION
   72 IF (INCOL.EQ.0) RETURN
```

```
      GO TO 79
      END
      SUBROUTINE BNBCD(VAL,ST)
      DIMENSION ST(1), TS(8)
      REAL X/'X'/, POINT/'.'/, BLANK/' '/, QMINU/'-'/
      REAL SYM(10)/'0','1','2','3','4','5','6','7','8','9'/
      MAX = 5
      L = 1
      IF (VAL .GE. 0.)  GO TO 1
      ST(L) = QMINU
      L = 2
    1 V = ABS(VAL)
      I = MAX
      ST(L) = SYM(1)
      IF (V .LT. 10.**MAX)  GO TO 3
      DO 2 L = L,MAX
    2 ST(L) = X
      RETURN
C           INTEGRAL PART
    3 IF (V .LT. 1.)  GO TO 7
    4 IF (V .LT. 1.)  GO TO 5
      TS(I) = SYM(V - 10. * AINT(V / 10.) + 1.)
      V = V / 10.
      I = I - 1
      IF (I .GT. 0)  GO TO 4
    5 I = I + 1
      DO 6 I = I,MAX
      ST(L) = TS(I)
    6 L = L + 1
      L = L - 1
      IF (L .GE. MAX)  RETURN
C           FRACTIONAL PART
    7 L = L + 1
      V = ABS(VAL - AINT(VAL))
      ST(L) = POINT
      IF (V .EQ. 0.)  RETURN
      L = L + 1
    8 V = 10. * V
      ST(L) = SYM(V + 1.)
      L = L + 1
      IF (L .GT. MAX)  RETURN
      V = V - AINT(V)
      IF (V .NE. 0.)  GO TO 8
      RETURN
      END
```

GEOGRAPHICAL NEIGHBORS

Purpose: Given NC points identified by rectangular coordinates X_i, Y_i, $i = 1 \ldots NC$ the program produces the adjacency matrix of neighbors of order $NN \leq 9$. These can then be used to find the neighbors to points of a regular lattice.

Description: Subroutine THIEBS calculates the adjacency matrix. Neighbors are defined on the basis of adjacent Thiessen polygons: Let A_i be the area closer to point i than to any other point, and similarly for A_j with respect to point j. If A_i and A_j touch (contact > 1) then i and j are first order neighbors. Higher order neighbors to a point are defined by deleting neighbors of one lower order. The search algorithm is from a program by Gambini, and requires examination of the lines of equilibrium between all pairs of observations. Specifically, the search proceeds by incrementing along the perpendicular bisector of the line connecting each pair until a position is reached for which the "attraction" is greatest. If the equilibrium point is not attained by the time the search reaches the boundary, another pair is considered. The algorithm is slow and tedious, but no better procedure is known.

A rectangular lattice can now be superimposed on the original point distribution, and the points of the original set which are the neighbors to the lattice points are calculated. The variable SCALE determines the number of coordinate units between the lattice points. If the adjacency matrix of the original set of points is known in advance, this may be entered as data.

Deck Make-up: 1) Control Card

2) Limits Card

3) Format Card

4) Observations

5) Adjacency Matrix (optional)

Data Cards: Control Card (2I5, 4X, I1, 4X, I1)

columns		
1-5	NC	number of observations
6-10	NN	maximum number of neighbors desired.
15	INNBR	0 if neighbors are to be computed 1 if adjacency matrix is to be read as data
20	IFPCH	1 if adjacency matrix is to be punched.

Limits Card

1-10	XO	minimum X coordinate
11-20	X1	maximum X coordinate
21-30	YO	minimum Y coordinate
31-40	Y1	maximum Y coordinate The above limits all apply to the lattice region.
41-50	DMAX	Search increment; blank yields valid default of SCALE/100. Small value of DMAX results in excessive computation

		time; large value increases the probability of missing a short boundary between adjacent polygons (i.e., missing a neighbor).
51-60	RLIM	maximum search radius; blank yields valid default.
61-70	SCALE	unit lattice size; blank yeilds default of $[(XMAX - XMIN)*(YMAX - YMIN)/N]^{\frac{1}{2}}$

Format Card (20A4)

Format for the X, Y coordinates of the observations.

Observations

X, Y coordinates punched as described on the format card.

Adjacency Matrix

Read iff INNBR = 1. Punched as 80 I 1; the number is the order of the adjacency, zero or blank implies non-adjacent.

Programmer: Dierk Rhynsburger, University of Michigan.

Reference: R. Gambini, "A Computer Program for Calculating Lines of Equilibrium between Multiple Center of Attraction," Center for Regional Studies, University of Kansas, Lawrence, no date.

```
C         THIESSEN POLYGONS OF GRID INTERSECTIONS
      DIMENSION XC(100),YC(100), UC(100), VC(100), IZ(100,100), IFMT(20)
      COMMON XC, YC, NC
      COMMON /LIM/ XO, X1, YO, Y1
      COMMON /MGT/NN, IZ, DMAX, RLIM
   10 FORMAT (16I5)
   11 FORMAT (80I1)
   12 FORMAT (1H1,15HLIMITS OF DATA:/1H0,10X,6HXMAX =,F10.3,10X,6HYMAX =
     &   F10.3/1H0,10X,6HXMIN =,F10.3,10X,6HYMIN =,F10.3)
   13 FORMAT (1H-,8HSCALE IS,F8.3,25H DATA UNITS PER GRID CELL)
   14 FORMAT (8E10.3)
   15 FORMAT (20A4)
   16 FORMAT (1H-,23HINCREMENT FOR SEARCH IS,E18.6)
   17 FORMAT (1H-,16HSEARCH RADIUS IS,E18.6)
   18 FORMAT (1H-,39HNEIGHBORS OF OBSERVATIONS READ AS INPUT)
      READ 10, NC, NN, INNBR, IFPCH
      READ 14, XO, X1, YO, Y1, DMAX, RLIM, SCALE
      READ 15, (IFMT(I),I=1,20)
      READ (5,IFMT) (XC(I), YC(I), I=1,NC)
      IF (RLIM .EQ. 0.)  RLIM = SQRT((X1 - XO)**2 + (Y1 - YO)**2)
      IF (SCALE .EQ. 0.)  SCALE = SQRT((X1 - XO) * (Y1 - YO) / NC)
      IF (DMAX .EQ. 0.)  DMAX = SCALE / 100.
      PRINT 12, X1, Y1, XO, YO
      PRINT 13, SCALE
      PRINT 16, DMAX
      PRINT 17, RLIM
      IF (INNBR .EQ. 0)  GO TO 98
      READ 11, ((IZ(I,J),J=1,NC),I=1,NC)
      PRINT 18
      GO TO 99
   98 CALL THIESS(IFPCH)
   99 NB = (X1 - XO) / SCALE + 1.99
      NA = (Y1 - YO) / SCALE + 1.99
      YP = Y1 + SCALE
      DO 100 I = 1,NA
      YP = YP - SCALE
      XP = XO - SCALE
      DO 100 J = 1,NB
      XP = XP + SCALE
  100 CALL GNBR(XP,YP)
      CALL EXIT
      END
      SUBROUTINE THIESS(IFP)
      DIMENSION XC(500), YC(500), IX(500), IS(500), IY(500,15)
      DIMENSION IZ(50,50), IR(500), IN(500,10)
      COMMON XC, YC, NC
      COMMON /LIM/ XO, X1, YO, Y1
      COMMON /MGT/NN, IZ, DMAX, RLIM
   25 FORMAT (1H1)
   26 FORMAT (1H ,50I2)
   27 FORMAT (1H1,35HFIRST NEIGHBORS OF THE OBSERVATIONS)
   28 FORMAT (1H1,9HNEIGHBORS,I4)
   29 FORMAT (1H0,11X,1HX,9X,1HY)
   30 FORMAT (1H-,I4,2F10.2,I8,4X,20I4)
   31 FORMAT (1H-,I4,2F10.2)
```

```
   32 FORMAT (1HO,I8,19H NEIGHBORS OF ORDER,I2,1H:,20I4)
   33 FORMAT (1H-,16HHIGHER NEIGHBORS)
   51 FORMAT (1H ,65I2)
   52 FORMAT (80I1)
      NS = NC - 1
      NL = NN - 1
      DO 95 I = 1,NC
      DO 93 J = 1,NC
   93 IZ(I,J) = 0
   95 IS(I) = 0
      DO 1000 I=1,NS
      NFC=I+1
      II = 0
  105 DO 900 J=NFC,NC
      DX = XC(I) - XC(J)
      DY = YC(I) - YC(J)
      AX = (XC(I)+XC(J))/2.
      AY = (YC(I)+YC(J))/2.
      CALL LMT(DX,DY,AX,AY,DMAX,U,NSTEP)
      ITEST = 2
      DO 600 K = 1,NSTEP
      U = U + DMAX
      X = AX + U * DY
      Y = AY - U * DX
      IP = ITEST
      ITEST = JFUN(X,Y,I,J)
      IF(ITEST.NE.IP) GO TO 700
  600 CONTINUE
      GO TO 900
  700 II = II + 1
      IX(II) = J
      IS(J) = IS(J) + 1
      IY(J,IS(J)) = I
  900 CONTINUE
  910 IT = IS(I)
      IF (IT)  1000,918,912
  912 DO 914 J = 1,IT
      IZ(I,IY(I,J)) = 1
  914 IN(I,J) = IY(I,J)
  918 DO 920 J = 1,II
      IZ(I,IX(J)) = 1
  920 IN(I,J+IT) = IX(J)
      IR(I) = II + IT
 1000 CONTINUE
      IT = IS(NC)
      IR(NC) = IT
      DO 1005 J = 1,IT
      IN(NC,J) = IY(NC,J)
 1005 IZ(NC,IY(NC,J)) = 1
      IK = 1
      PRINT 27
      PRINT 29
      DO 1009 I = 1,NC
      JK = IR(I)
 1009 PRINT 30, I, XC(I), YC(I), JK, (IN(I,K),K=1,JK)
      PRINT 28, IK
```

```
      DO 1010 I = 1,NC
 1010 PRINT 26, (IZ(I,J),J=1,NC)
      IF (NN .EQ. 1)  RETURN
      DO 2000 I = 1,NL
      DO 2000 J = 1,NS
      JJ = J + 1
      DO 1995 K = JJ,NC
      IF (IZ(J,K) .NE. 1)  GO TO 1995
      DO 1990 L = 1,NC
      IF (IZ(K,L) .NE. 1)  GO TO 1990
      IF (IZ(J,L) .NE. 0)  GO TO 1990
      IF (J .EQ. L)  GO TO 1990
      IZ(J,L) = I + 1
      IZ(L,J) = I + 1
 1990 CONTINUE
 1995 CONTINUE
 2000 CONTINUE
      PRINT 33
      DO 2050 I = 1,NC
 2050 PRINT 51, (IZ(I,J),J=1,NC)
      PRINT 25
      IF (IFP .EQ. 1)  WRITE (7,52) ((IZ(I,J),J=1,NC),I=1,NC)
      DO 2100 I = 1,NC
      PRINT 31, I, XC(I), YC(I)
      DO 2100 J = 1,NN
      IW = 0
      DO 2090 K = 1,NC
      IF (IZ(I,K) .NE. J)  GO TO 2090
      IW = IW  +1
      IS(IW) = K
 2090 CONTINUE
      IF (IW .EQ. 0)  GO TO 2100
      PRINT 32, IW, J, (IS(K),K=1,IW)
 2100 CONTINUE
      PRINT 25
 9999 RETURN
      END
      SUBROUTINE GNBR(UC,VC)
      DIMENSION XC(500), YC(500), AA(500), BB(500)
      DIMENSION IZ(50,50), IR(100), IX(20), IY(20)
      COMMON XC, YC, NC
      COMMON /LIM/ XO, X1, YO, Y1
      COMMON /MGT/NN, IZ, DMAX, RLIM
   12 FORMAT (1H1,12HNEIGHBORS TO,F9.2,1H,,F7.2)
   13 FORMAT (1H-,9HNEIGHBORS,I4,I15)
   14 FORMAT (1H ,15X,I4,2F8.2)
   23 FORMAT (1H ,21HRLIM HAS BEEN DOUBLED)
   24 FORMAT (1H-,30HRLIM DOUBLED: CHANGE GRID SIZE)
      II = 0
      IO = 1
      IRL = 1
      PRINT 12, UC, VC
      DO 100 J = 1,NC
      A = UC - XC(J)
      B = VC - YC(J)
      IF (A .EQ. 0. .AND. B .EQ. 0.)  GO TO 920
```

```
      AA(J) = A
  100 BB(J) = B
  105 DO 900 J = 1,NC
      DX = AA(J)
      DY = BB(J)
      DEN = SQRT(DX * DX + DY * DY)
      IF (DEN .GT. RLIM)  GO TO 900
      IRL = 0
      AX = (UC + XC(J)) / 2.
      AY = (VC + YC(J)) / 2.
      CALL LMT(DX,DY,AX,AY,DMAX,U,NSTEP)
      ITEST = 2
      DO 600 K = 1,NSTEP
      U = U + DMAX
      X = AX + U * DY
      Y = AY - U * DX
      IP = ITEST
      ITEST = IFUN(X,Y,UC,VC,J)
      IF (ITEST .NE. IP)  GO TO 700
  600 CONTINUE
      GO TO 900
  700 II = II + 1
      IX(II) = J
  900 CONTINUE
      IF (IRL .EQ. 0)  GO TO 940
      PRINT 23, UC, VC
      RLIM = RLIM + RLIM
      IF (IRL .EQ. 3)  GO TO 9998
      IRL = IRL + 1
      GO TO 105
  920 II = 1
      IX(1) = J
  940 PRINT 13, IO, II
      PRINT 14, (IX(I), XC(IX(I)), YC(IX(I)), I=1,II)
      IF (NN .EQ. 1)  RETURN
      IT = II
      DO 1050 I = 1,II
      IY(I) = IX(I)
 1050 IR(I) = IX(I)
      DO 1500 I = 2,NN
      IO = 0
      DO 1070 J = 1,II
 1070 IX(J) = IY(J)
      DO 1300 J = 1,II
      DO 1200 K = 1,NC
      IF (IZ(IX(J),K) .NE. 1)  GO TO 1200
      DO 1100 L = 1,IT
      IF (IR(L) .EQ. K)  GO TO 1200
 1100 CONTINUE
      IO = IO + 1
      IT = IT + 1
      IY(IO) = K
      IR(IT) = K
 1200 CONTINUE
 1300 CONTINUE
      IF (IO .EQ. 0)  RETURN
```

```
      II = 10
      PRINT 13, I, II
 1500 PRINT 14, (IY(J), XC(IY(J)), YC(IY(J)), J=1,II)
      RETURN
 9998 PRINT 24
 9999 RETURN
      END
      SUBROUTINE LMT(DX,DY,AX,AY,USTEP,U,NSTEP)
      COMMON /LIM/ X0, X1, Y0, Y1
      IF (DY) 111,112,113
  111 IF (DX) 114,115,116
  112 IF (DX) 117,9999,118
  113 IF (DX) 119,120,121
  114 U = -AMIN1((AX - X1) / DY, (YC - AY) / DX)
      UL = AMIN1((X0 - AX) / DY, (AY - Y1) / DX)
      GO TO 124
  115 U = (X1 - AX) / DY
      UL = (X0 - AX) / DY
      GO TO 124
  116 U = -AMIN1((AX - X1) / DY, (Y1 - AY) / DX)
      UL = AMIN1((X0 - AX) / DY, (AY - Y0) / DX)
      GO TO 124
  117 U = (AY - Y0) / DX
      UL = (AY - Y1) / DX
      GO TO 124
  118 U = (AY - Y1) / DX
      UL = (AY - Y0) / DX
      GO TO 124
  119 U = -AMIN1((AX - X0) / DY, (Y0 - AY) / DX)
      UL = AMIN1((X1 - AX) / DY, (AY - Y1) / DX)
      GO TO 124
  120 U = (X0 - AX) / DY
      UL = (X1 - AX) / DY
      GO TO 124
  121 U = -AMIN1((AX - X0) / DY, (Y1 - AY) / DX)
      UL = AMIN1((X1 - AX) / DY, (AY - Y0) / DX)
  124 NSTEP = (UL - U) / USTEP + 1.
      U = U + USTEP
 9999 RETURN
      END
      FUNCTION IFUN(X,Y,UC,VC,J)
      DIMENSION XC(500), YC(500)
      COMMON XC, YC, NC
      COMMON /LIM/ X0, X1, Y0, Y1
      IFUN = 2
      ATT = 1. / SQRT((X - UC)**2 + (Y - VC)**2)
      DO 100 L = 1,NC
      IF (J .EQ. L)  GO TO 100
      IF (X .EQ. XC(L) .AND. Y .EQ. YC(L))  RETURN
      AT = 1. / SQRT((X - XC(L))**2 + (Y - YC(L))**2)
      IF (ATT .LT. AT)  RETURN
  100 CONTINUE
      IFUN = 3
      RETURN
      END
      FUNCTION JFUN(X,Y,I,J)
```

```
      DIMENSION XC(500), YC(500), A(500), ATTR(500)
      COMMON XC, YC, NC
      COMMON /LIM/ X0, X1, Y0, Y1
      JFUN = 2
      DO 100 L=1,NC
      IF (X .EQ. XC(L) .AND. Y .EQ. YC(L))  RETURN
      ATTR(L) = 1. / SQRT((X-XC(L))**2 + (Y-YC(L))**2)
  100 CONTINUE
      DO 200 M=1,NC
      IF((M.EQ.I).OR.(M.EQ.J)) GO TO 200
      IF (ATTR(I) .LT. ATTR(M))  RETURN
  200 CONTINUE
      JFUN = 3
  400 RETURN
      END
```

CONTOUR PLOTTING

Purpose: The program uses the 30 inch Calcomp plotter to draw contour maps from data given in the form of geographical matrices. Stereograms and perspective contours can also be obtained. An option allows conversion of the contours to a map projection before plotting.

Description: The contouring algorithm is that described by Dayhoff. The perspective plotting is based on that of Puckett.

Comments: File 2 must be used as a scratch tape. Input is from unit 7, comments are written onto unit 6, plotting is done via unit 9, and execution requires concatenation with *PLØTSYS.

Deck make-up: The controls are specified by an integer code punched in columns 1 and 2. The code also specifies whether further data are required on the same card, or whether additional cards are to be read. Reading of controls continues until plot code (20) is read. The number of control cards may vary from run to run. A basic sequence has been indicated by asterisks. The plot control card may be called repeatedly.

Map Projection Control Card:

Punch 05 in columns 1-2. This calls a map projection subroutine PROJ (X, Y, XMAX, YMAX, SCALE) which may read additional parameters. In the present instance the first call on PROJ is activated immediately following the first plot (code 20) which follows an Ø5 code, and the program then expects to read one card as follows:

Center latitude (decimal degrees) of the map in columns 1-7.

Center longitude of the map in 8-14.

Size of quadrilateral if different from one degree in columns 15-21.

Center latitude of an oblique stereographic projection in columns 22-28.

Center longitude of the oblique stereographic projection in columns 29-35.

Subsequent plots must repeat the 05 code if conversion to map projection coordinates is desired, and, as currently written, must use the same projection parameters. The width of the map using the 05 code is defined to be the width to scale at the center latitude on the square projection. Other map projections may be used by changing the subroutine PROJ.

*Title Control Card

Punch 12 in columns 1-2.

Follow this with any single card title (which will be plotted at the top of the drawing). Repeat as desired for all runs.

*Format Control Card

Punch 19 in columns 1-2.

Follow this with a single card (18A4) describing the E or F Fortran format for one row of the observation matrix.

*<u>Constant Contour Interval Card</u>

Punch 15 in columns 1-2

Punch lowest contour (with decimal point) in 3-12

Punch contour increment in 13-22

Punch highest contour in 23-32.

*<u>Data Definition Card</u>

Punch 01 in columns 1-2

Punch number of rows (LE.100) in data matrix as an E or F number in columns 3-12.

Punch number of columns (LE.100) in data matrix in columns 13-22.

Punch desired width (inches LE.28) of contour map in columns 23-32.

Punch denominator of vertical scale transformation in columns 33-42.

Follow this card by the data matrix as described by the earlier format control card.

*<u>Plot Control Card</u>

Punch 20 in columns 1-2

Plotting with the current controls is initiated. The remaining control cards (below) could also have been read before the Plot Control Card, with obvious exceptions.

<u>Width Change Card</u>

Punch 21 in columns 1-2

Punch desired width in columns 3-12.

Change Constant Contour Interval Card

Punch 16 in columns 1-2

Columns 3-12, 13-22, 23-32 are punched as for the Constant Contour Interval Card.

Variable Contour Interval Card

Punch 17 in columns 1-2

Punch the number of contour intervals desired in columns 3-12.

Follow this by one card which gives the format for the variable contours.

Follow this by the definition of the variable contours, from lowest to highest, on the appropriate number of of cards (as specified by the format).

Change Variable Contour Interval Card

Punch 18 in columns 1-2

Columns 3-12 and remaining cards punched as described under the Variable Contour Interval Card.

Non-Standard Plot File Card

Punch 11 in columns 1-2.

Punch the file unit onto which plots are to be written in columns 3-12.

Extra Labels Card

Punch 14 in columns 1-2.

Punch starting x-coordinate of the label in columns 3-12.

Punch starting y-coordinate of the label in columns 13-22.

Punch starting z-coordinate of the label in columns 23-32.

Height of the lettering in inches goes into columns 33-42.

Follow this by one card containing any label in columns 1-36.

Repeat the entire sequence up to 10 times.

Translation Card

Punch 10 in columns 1-2.

Punch DX (inches) in columns 3-12.

Punch DY in columns 13-22.

Punch DZ in columns 23-32.

Perspective Card

Punch 08 in columns 1-2

Punch distance to viewing plane in columns 3-12. This should be larger than max (2).

Punch distance to the object in columns 13-22.

X-Rotation Card

Punch 02 in columns 1-2

Punch degrees rotation relative to the X-axis in columns 3-12.

Y-Rotation Card

Punch 03 in columns 1-2

Punch degrees rotation relative to the Y-axis in columns 3-12.

Z-Rotation Card

Punch 04 in columns 1-2

Punch degrees rotation relative to the Z-axis in columns 3-12.

*Termination Card

Punch 13 in columns 1-2.

This terminates the entire program.

Programmer: F.J. Reno.

References:

M.O. Dayhoff, "A Contour-Map Program for X-ray Crystallography," Communications, ACM, 6, 10 (October, 1963), pp. 620-622.

H.R. Puckett, "Computer Method for Perspective Drawing," Journal of Spacecraft and Rockets, 1,1(January 1964), pp. 44-48.

R. Osborn, " An Automated Procedure for Producing Contour Charts", IM No. 67-4, Oceanographic Office, Washington D.C.

B. Kubert, "Two Computer Programs for Contour Plotting", Aerospace Corp., 1969. AD 682307.

G. McCue, H. DuPrie, " Improved Fortran IV Function Contouring Program", North Am. Aviation, SID-65-672, 1965.

D.L. Light, "Ranger Mapping by Analytics", Photogrammetric Engineering, XXXII, (1966), pp. 792-800.

R.F. Walters, "Contouring by Machine", Bulletin Am. Assn. Petroleum Geologists, V. 53, No. 11 (1969) pp. 2343-2340

B. Bengtsson, S. Nordbeck, "Construction of Isarithms and Isarithmic Maps by Computers", BIT, 4 (1964), pp. 87-105.

D. McIntyre, D. Pollard, R. Smith, "Computer Programs for Automatic Contouring", Kansas Geological Survey Computer Contribution #22, 1968.

W. Tobler, "l'automation dans la préparation des cartes thématiques", International Yearbook of Cartography, 1966, pp

A. Noll, "Sterographic Projections by Computer", Computers and Automation, May 1965, pp. 32-34.

```
C      CONTOUR-PERSPECTIVE PLOTTING / FRANK J RENS / GEOG..COMP.CNTR. / L
C     2 IS USED AS A SCRATCH FILE
C
      COMMON MT,NT,NI,IX,IY,IDX,IDY,ISS,IT,IV,NP,N,JT,PY,REC,CV,X,Y,IPT,
     1INX,INY,DL,AM,IOP,OP,DO,ILAB,XL,YL,ZL,LAB,DX,DY,DZ,XMAX,HT,SCALE,Y
     2MAX,CTHETA,STHETA,FM2,FNN,NN,M,CL,D,NLINES,NCURV,Z
      COMMON TIT,IZX
      DIMENSION AM(100,100), REC(800), X(1500), Y(1500), IPT(3,3), INX(8
     1), INY(8), IOP(23), XL(10), YL(10), ZL(10), LAB(10,6), DX(2), DY(2
     2), DZ(2), HT(10), CTHETA(4), STHETA(4)
      DIMENSION TEMP(6), FMT(18)
      DIMENSION Z(1500)
      DIMENSION TIT(18)
      DIMENSION CFMT(18), VCON(20)
      CALL TIME(0)
      CALL PLTXMX(30.0)
      NPLOT=1
      ISTAP=0
      IOP(1)=0
      IOP(23)=0
      D=0.
    1 ILAB=0
      CALL REWIND(2)
      DO 2 I=2,23
    2 IOP(I)=0
      NBOUND=0
C     REAL CONTROL INSTRUCTIONS ACCORDING TO WRITE UP
    3 READ (7,78) NCON,(TEMP(I),I=1,6)
      WRITE (6,79) NCON,(TEMP(I),I=1,6)
      IF(NCON.EQ.1) NLINES=0
      IF(NCON.GT.14.AND.NCON.LT.19) NLINES=0
C*    CALL CPER(NCON)
      IF (NCON) 74,74,4
    4 IF (NCON-23) 5,5,74
    5 GO TO (6,21,21,21,22,23,24,25,25,28,29,30,31,32,36,37,38,39,40,41,
     161,62,63), NCON
C     NCON=1
    6 CONTINUE
      CALL TIME(3,1)
C     M= # OF ROWS       NN= # OF COLS
      M=TEMP(1)
      NN=TEMP(2)
      TEMP(2)=TEMP(2)-1.
      SCALE=TEMP(2)/TEMP(3)
      IF (TEMP(4)) 8,7,8
    7 TEMP(4)=1.
    8 ZSCALE=TEMP(4)
      FM2=FLOAT(M)/2.
      FNN=FLOAT(NN)/2.
      XMAX=(FNN/SCALE)+2.5
      YMAX=(FM2/SCALE)+2.5
      IF (XMAX-14.) 9,9,70
    9 IF (YMAX-14.) 10,70,70
   10 IF (IOP(1)) 11,12,11
   11 IOP(15)=1
```

```
      IOP(19)=1
12    ICP(1)=1
      IF (IOP(19)) 13,71,13
C     READ MATRIX
C PROGRAM READS ALONG ROWS CORRECTLY
13    READ (7,FMT) ((AM(J,I),J=1,NN),I=1,M)
      WRITE(6,96)
96    FORMAT('1',50X,'**DATA MATRIX**'/21X,'-------------------------
     1------------------------------------------------------------------
     1-----------')
      DO 14 I=1,M
  14  WRITE(6,1002)       (AM(J,I),J=1,NN)
1002  FORMAT('0',10(1X,E12.4)/(10(1X,E12.4)))
      IZX=1
      IF (IOP(15)) 15,72,15
15    IF (IOPV) 16,18,16
C     VARIABLE CONTOURS
16    DO 17 I=1,NVCON
      CL=VCON(I)
      CV=CL
17    CALL SCAN
      GC TO 20
C     LOOP THROUGH CONTOURING ROUTINES
18    CV=CBGN
19    CL=CV
      CALL SCAN
      CV=CV+CINC
      IF (CV-CEND) 19,19,20
20    END FILE 2
      CALL REWIND(2)
      CALL TIME(3,1)
      GO TO 3
C     NCON=2,3,4
C     DEFINE THETA FOR ROTATION
21    IOP(NCON)=1
      NBOUND=1
      THETA=-TEMP(1)/57.295795
      STHETA(NCON)=SIN(THETA)
      CTHETA(NCON)=COS(THETA)
      GO TO 3
C     NCON=5
22    IOP(NCON)=1
      GO TO 3
C     NCON=6
C     SLCW DOWN THE PEN
23    CCNTINUE
      GO TO 3
C     NCON=7
C     SPEED UP THE PEN
24    CONTINUE
      GO TO 3
C     NCON=8,9
C     DEFINE PERSPECTIVE CONSTANTS
25    IOP(NCON)=1
      IF (NCON-8) 27,26,27
26    NBOUND=1
```

```
27    CONTINUE
      DP=TEMP(1)
      DO=TEMP(2)
      GO TO 3
C     NCON=10
C     DEFINE TRANSLATION
28    DX(1)=TEMP(1)
      DY(1)=TEMP(2)
      DZ(1)=TEMP(3)
      ICP(NCON)=1
      GO TO 3
C     NCON=11
29    PAUSE 1
      ISTAP=TEMP(1)
      GO TO 3
C     NCON=12
30    ICP(12)=1
      READ (7,81) (TIT(I),I=1,12)
      WRITE(6,97) (TIT(I),I=1,12)
97    FORMAT(70X,'TITLE : ',12A4)
      GO TO 3
C     NCON=13
C     END PLOT TAPE
31    CALL PLTTRM (0)
      CALL SYSTEM (0)
C     NCON=14
C     READ LABELS AND THEIR CORRESPONDING COORDINATES
32    IOP(14)=1
      ILAB=ILAB+1
      XL(ILAB)=TEMP(1)
      YL(ILAB)=TEMP(2)
      IF (ZSCALE) 34,33,34
33    ZL(ILAB)=TEMP(3)
      GO TO 35
34    ZL(J)=TEMP(3)/ZSCALE
35    HT(ILAB)=TEMP(4)
      READ (7,83) (LAB(I,ILAB),I=1,6)
      WRITE (6,86) (LAB(I,ILAB),I=1,6)
      GO TO 3
C     NCON=15
C     DEFINE CONTOUR INCREMENTS
36    CBGN=TEMP(1)
      CINC=TEMP(2)
      CEND=TEMP(3)
      IOP(15)=1
      IOPV=0
      GO TO 3
C     NCON=16
37    CBGN=TEMP(1)
      CINC=TEMP(2)
      CEND=TEMP(3)
      IOPV=0
      GO TO 18
C     NCON=17
38    NVCON=TEMP(1)
      IOP(15)=1
```

```
      IOPV=1
      READ (7,85)  CFMT
      READ (7,CFMT) (VCON(I),I=1,NVCON)
      WRITE (6,84) (CFMT(I),I=1,18),(VCON(I),I=1,NVCON)
      GO TO 3
C     NCON=18
39    NVCON=TEMP(1)
      IOPV=1
      READ (7,85)  CFMT
      READ (7,CFMT) (VCON(I),I=1,NVCON)
      WRITE (6,84) (CFMT(I),I=1,18),(VCON(I),I=1,NVCON)
      GO TO 16
C     NCON=19
C     READ FORMAT FOR MATRIX
40    READ (7,85)  FMT
      WRITE (6,86)  FMT
      IOP(19)=1
      GO TO 3
C     NCON=20
C     BEGIN PLOTTING
 41   CONTINUE
      CALL TIME(3,1)
      IF (IOP(23)) 64,42,64
42    CONTINUE
43    IF (IOP(8)) 45,44,45
44    IOP(9)=1
45    IF (IOP(1)) 95,73,95
95    CONTINUE
C     SAFETY VALVE
      ASSIGN 46 TO NNL
      GO TO NNL, (53,46)
46    ASSIGN 53 TO NNL
C*  CALL SETEFL (NNL)
C     CALL DRAFT TO PLOT PERSPECTIVELY
      CBGN=-53139.E15
      NCURV=0
      NCOUNT=0
      NCMAX=0
      IF (NLINES) 52,52,47
47    READ(2,1000) N,CV
1000  FORMAT(I5,E12.4)
      IF (CBGN+53139.E15) 49,48,49
48    CBGN=CV
49    READ(2,1001)(X(I),Y(I),I=1,N)
1001  FORMAT(6E12.4)
      NCURV=NCURV+1
      NCOUNT=NCOUNT+N
      IF (N-NCMAX) 51,50,50
50    NCMAX=N
      CLMAX=CV
51    CV=CV/ZSCALE
      CALL DRAFT
      IF (NCURV-NLINES) 47,53,53
52    CBGN=0.
53    WRITE (6,88) NPLOT,NLINES,NCOUNT,NCMAX,CLMAX,CBGN
C     MODIFY BOUNDARY FOR MAP PROJECTION
```

```
      IF (IOP(5)) 59,59,54
54    CV=CBGN
      N=1
      X(1)=-FNN+1.
      Y(1)=-FM2+1.
      DO 55 I=2,NN
      X(I)=X(I-1)+1.
      Y(I)=Y(1)
55    N=N+1
      CALL DRAFT
      X(1)=-FNN+1.
      Y(1)=FM2
      DO 56 I=2,NN
      X(I)=X(I-1)+1.
56    Y(I)=Y(1)
      CALL DRAFT
      N=1
      X(1)=FNN
      Y(1)=FM2
      DO 57 I=2,M
      X(I)=X(1)
      Y(I)=Y(I-1)-1.
57    N=N+1
      CALL DRAFT
      X(1)=-FNN+1.
      Y(1)=FM2
      DO 58 I=2,M
      X(I)=X(1)
58    Y(I)=Y(I-1)-1.
      CALL DRAFT
      GO TO 60
59    N=5
      CV=CBGN
      X(1)=FNN
      X(2)=-X(1)+1.
      X(3)=X(2)
      X(4)=X(1)
      X(5)=X(1)
      Y(1)=FM2
      Y(2)=Y(1)
      Y(3)=-Y(2)+1.
      Y(4)=Y(3)
      Y(5)=Y(1)
      CALL DRAFT
60    IF (IOP(23)) 67,62,67
C     NCON=21
C     CHANGE SCALE
 61   SCALE=(FLOAT(NN)-1.)/TEMP(1)
      GO TO 3
C     NCON = 22
C     END THE PLOT
62    CALL PLTEND
      IOP(23)=0
      WRITE (6,89)
89    FORMAT(' PLOTTING SYSTEM LINE FILE GENERATION HAS TERMINATED')
      NPLOT=NPLOT+1
```

```
      CALL TIME(3,1)
      GO TO 1
C     NCON = 2?
C     RED AND GREEN BUSINESS
63    CONTINUE
      FLABGT=PSYMLN(.2,99)
C     MTS SYSTEM DOES NOT INCORPORATE COLORED PENS AT THIS TIME - SUMMER
C     CALL PSYMB (.2,FLABGT+2.,-.2,99HPLEASE INSERT RED PEN NOW - AT NEX
C    1T 999 STOP INSERT THE GREEN PEN - AT 3RD 999 STOP INSERT BPBLK   .
C    2270.,99)
C     CALL PLTSTP
      CALL PLTEND
      IOP(23)=3
      GO TO 3
64    IF (IOP(23)-3) 65,66,66
65    CONTINUE
66    IOP(23)=IOP(23)-1
      GO TO 43
67    IF (IOP(23)-1) 69,68,69
68    CONTINUE
      GO TO 62
69    GO TO 1
70    XMAX=XMAX*2.
      YMAX=YMAX*2.
      WRITE (6,90) YMAX,XMAX
      GO TO 75
71    WRITE (6,91)
      GO TO 75
72    WRITE (6,92)
      GO TO 75
73    WRITE (6,93)
      GO TO 75
74    WRITE (6,94) NCON
75    IF (NPLOT-1) 76,76,77
76    CALL PLTTRM (0)
77    CALL ERROR (0)
C
78    FORMAT (I2,6F10.0)
   79 FORMAT(1H0,I2,6F10.2)
80    FORMAT (1H0,(10F12.6))
   81 FORMAT(12A4)
   83 FORMAT(9A4)
   84 FORMAT(1H0,18A4//(10X,10E12.6))
   85 FORMAT(18A4)
86    FORMAT(70X,'THE FORMAT FOR THE DATA MATRIX IS : ',18A4)
88    FORMAT(' PLOT NUMBER ',I3,'  NUMBER OF LINES = ',I6,',NUMBER OF PO
     1INTS = ',I10/' MAXIMUM NUMBER OF POINTS = ',I10,' WHICH FORMED A C
     2URVE WITH CONTOUR LEVEL OF ',F10.2,', BOUNDARY AT ',F10.2)
90    FORMAT (14H4****A MAP OF ,F10.1,4H BY ,F10.1,14H WAS REQUESTED)
91    FORMAT (32H4****FORMAT NOT DEFINED FOR DATA)
92    FORMAT (31H4****CONTOUR LEVELS NOT DEFINED)
93    FORMAT (12H4****NO DATA)
94    FORMAT (27H4****NCON NOT LEGAL, NCON =,I5)
      END
C     CONTOUR-PERSPECTIVE PLOTTING / FRANK J RENS / GEOG.,COMP.CNTR. / 1
      SUBROUTINE SCAN
```

```
C     ADAPTED FOR U OF M SYSTEM BY FRANK J RENS
      COMMON MT,NT,NI,IX,IY,IDX,IDY,ISS,IT,IV,NP,NQ,JT,PY,REC,CV,X,Y,IPT
     1,INX,INY,DL,AM,IOP,DP,DD,ILAB,XL,YL,ZL,LAB,DX,DY,DZ,XMAX,HT,SCALE,
     2YMAX,CTHETA,STHETA,FM2,FNN,NN,M,CL,D,NLINES,NCURV,Z
      COMMON TIT,IZX
      DIMENSION TIT(18)
      DIMENSION AM(100,100), REC(800), X(1500), Y(1500), IPT(3,3), INX(8
     1), INY(8), IOP(23), XL(10), YL(10), ZL(10), LAB(10,6), DX(2), DY(2
     2), DZ(2), HT(10), CTHETA(4), STHETA(4)
      DIMENSION Z(1500)
      NP=0
      DL=D
      NT=M
      MT=NN
      IF (IZX) 2,2,1
1     IPT(1,1)=8
      IPT(1,2)=1
      IPT(1,3)=2
      IPT(2,1)=7
      IPT(2,3)=3
      IPT(3,1)=6
      IPT(3,2)=5
      IPT(3,3)=4
      INX(1)=-1
      INX(2)=-1
      INX(3)=0
      INX(4)=1
      INX(5)=1
      INX(6)=1
      INX(7)=0
      INX(8)=-1
      INY(1)=0
      INY(2)=1
      INY(3)=+1
      INY(4)=+1
      INY(5)=0
      INY(6)=-1
      INY(7)=-1
      INY(8)=-1
      IZX=0
2     XT=MT
      DO 3 J=1,800
3     REC(J)=0.
      ISS=0
      MT1=MT-1
      DO 6 I=1,MT1
      IF (AM(I,1)-CV) 4,6,6
4     IF (AM(I+1,1)-CV) 6,5,5
5     IX=I+1
      IY=1
      IDX=-1
      IDY=0
      CALL TRACE
6     CONTINUE
      NT1=NT-1
      DO 9 I=1,NT1
```

```
      IF (AM(MT,I)-CV) 7,9,9
7     IF (AM(MT,I+1)-CV) 9,8,8
8     IX=MT
      IY=I+1
      IDX=0
      IDY=-1
      CALL TRACE
9     CONTINUE
      DO 12 I=1,MT1
      MT2=MT+1-I
      IF (AM(MT2,NT)-CV) 10,12,12
10    IF (AM(MT2-1,NT)-CV) 12,11,11
11    IX=MT2-1
      IY=NT
      IDX=1
      IDY=0
      CALL TRACE
12    CONTINUE
      DO 15 I=1,NT1
      NT2=NT+1-I
      IF (AM(1,NT2)-CV) 13,15,15
13    IF (AM(1,NT2-1)-CV) 15,14,14
14    IX=1
      IY=NT2-1
      IDX=0
      IDY=1
      CALL TRACE
15    CONTINUE
      ISS=1
      NT1=NT-1
      MT1=MT-1
      DO 21 J=2,NT1
      DO 21 I=1,MT1
      IF (AM(I,J)-CV) 16,21,21
16    IF (AM(I+1,J)-CV) 21,17,17
17    COM=100*(I+1)+J
      IF (NP) 18,20,18
18    DO 19 ID=1,NP
      IF (REC(ID)-COM) 19,21,19
19    CONTINUE
20    IX=I+1
      IY=J
      IDX=-1
      IDY=0
      CALL TRACE
21    CONTINUE
      RETURN
      END
C     CONTOUR-PERSPECTIVE PLOTTING / FRANK J RENS / GEOG..COMP.CNTR. /
      SUBROUTINE TRACE
C     ADAPTED FOR U OF M SYSTEM BY FRANK J RENS
      COMMON MT,NT,NI,IX,IY,IDX,IDY,ISS,IT,IV,NP,N,JT,PY,REC,CV,X,Y,IPT,
     1INX,INY,DL,AM,IOP,OP,DO,ILAB,XL,YL,ZL,LAB,OX,OY,OZ,XMAX,HT,SCALE,Y
     2MAX,CTHETA,STHETA,FM2,FNN,NN,M,CL,D,NLINES,NCURV,Z
      COMMON TIT
      DIMENSION TIT(18)
```

```
      DIMENSION AM(100,100), REC(800), X(1500), Y(1500), IPT(3,3), INX(8
     1), INY(8), IOP(23), XL(10), YL(10), ZL(10), LAB(10,6), DX(2), DY(2
     2), DZ(2), HT(10), CTHETA(4), STHETA(4)
      DIMENSION Z(1500)
      PY=0.0
      JT=0
      N=0
      IX0=IX
      IY0=IY
      ISX=IDX+2
      ISY=IDY+2
      IS=IPT(ISX,ISY)
      JTB=0
      IS0=IS
      IF (IS0-8) 2,2,1
1     IS0=IS0-8
2     IT=0
3     CALL CALC
      NZ=N
      N=NZ
      IF (IT+JT-1) 5,5,4
4     XS=X(N-1)
      YS=Y(N-1)
      X(N-1)=X(N)
      Y(N-1)=Y(N)
      X(N)=XS
      Y(N)=YS
5     IS=IS+1
      JT=IT
6     IF (IS-9) 8,7,7
7     IS=IS-8
8     IDX=INX(IS)
      IDY=INY(IS)
      IX2=IX+IDX
      IY2=IY+IDY
      JTB=JTB+1
      IF (JTB-1500) 10,10,9
9     WRITE (6,34) CV,X(N),Y(N)
      RETURN
10    IF (ISS) 15,15,11
11    IF (IX-IX0) 19,12,19
12    IF (IY-IY0) 19,13,19
13    IF (IS-IS0) 19,14,19
14    CALL CALC
      GO TO 30
15    IF (IX2) 16,28,16
16    IF (IX2-MT) 17,17,28
17    IF (IY2) 18,28,18
18    IF (IY2-NT) 19,19,28
19    IF (CV-AM(IX2,IY2)) 20,20,3
20    IF (IDX**2+IDY**2-1) 21,25,21
21    DCP=(AM(IX,IY)+AM(IX2,IY)+AM(IX,IY2)+AM(IX2,IY2))/4.0
      IF (DCP-CV) 3,22,22
22    IF (INX(IS-1)) 23,24,23
23    IX=IX+IDX
      IDX=-IDX
```

```
      PY=2.0
      CALL CALC
      IX=IX+IDX
      GO TO 25
24    IY=IY+IDY
      IDY=-IDY
      PY=2.0
      CALL CALC
      IY=IY+IDY
25    IF (AM(IX-1,IY)-CV) 26,27,27
26    NP=NP+1
      REC(NP)=100*IX+IY
27    IS=IS+5
      IX=IX2
      IY=IY2
      GO TO 6
28    XT=MT
      IF (AM(IX-1,IY)-CV) 29,30,30
29    NP=NP+1
      REC(NP)=100*IX+IY
30    NLINES=NLINES+1
      DO 31 K=1,N
      X(K)=X(K)-FNN
      Y(K)=Y(K)-FM2
31    CONTINUE
C     STORE CURVE ON TAPE 2  (B2)
      IF (N) 33,33,32
32    WRITE(2,1000) N,CL
1000  FORMAT(I5,E12.4)
      WRITE(2,1001)(X(I),Y(I),I=1,N)
1001  FORMAT(6E12.4)
33    N=-1
      RETURN
C
   34 FORMAT('0ACONTOURATLEVEL',E11.5,'WASTERMINATEDATX=',F5.1,'Y=',F5.1
     1,'BECAUSEITCONTAINEDMORETHAN1500PLOTPOINTS')
      END
C     CONTOUR-PERSPECTIVE PLOTTING / FRANK J RENS / GEOG..COMP.CNTR. / 1
      SUBROUTINE CALC
C     ADAPTED FOR U OF M SYSTEM BY FRANK J RENS
      COMMON MT,NT,NI,IX,IY,IDX,IDY,ISS,IT,IV,NP,N,JT,PY,REC,CV,X,Y,IPT,
     1INX,INY,DL,AM,IOP,DP,DO,ILAB,XL,YL,ZL,LAB,DX,DY,DZ,XMAX,HT,SCALE,Y
     2MAX,CTHETA,STHETA,FM2,FNN,NN,M,CL,D,NLINES,NCURV,Z
      COMMON TIT
      DIMENSION TIT(18)
      DIMENSION AM(100,100), REC(800), X(1500), Y(1500), IPT(3,3), INX(8
     1), INY(8), IOP(23), XL(10), YL(10), ZL(10), LAB(10,6), DX(2), DY(2
     2), DZ(2), HT(10), CTHETA(4), STHETA(4)
      DIMENSION Z(1500)
      IT=0
      N=N+1
      IF (IDX**2+IDY**2-1) 4,1,4
1     IF (IDX) 3,2,3
  2   X(N)=IX
      Z(N)=IY
      IY2=IY+IDY
```

```
      DY(2)=IDY
      Y(N)=((AM(IX,IY)-CV)/(AM(IX,IY)-AM(IX,IY2)))*DY(2)+Z(N)
      RETURN
3     Y(N)=IY
      W=IX
      DX(2)=IDX
      IX2=IX+IDX
      X(N)=((AM(IX,IY)-CV)/(AM(IX,IY)-AM(IX2,IY)))*DX(2)+W
      RETURN
4     IX2=IX+IDX
      IY2=IY+IDY
      W=IX
      Z(N)=IY
      DX(2)=IDX
      DY(2)=IDY
      DCP=(AM(IX,IY)+AM(IX2,IY)+AM(IX,IY2)+AM(IX2,IY2))/4.0
      IF (PY-2.0) 5,6,5
5     IF (DCP-CV) 6,6,7
6     AL=AM(IX,IY)-DCP
      V=.5*(AL+DCP-CV)/AL
      X(N)=V*DX(2)+W
      Y(N)=V*DY(2)+Z(N)
      PY=0.0
      RETURN
7     IT=1
      AL=AM(IX2,IY2)-DCP
      V=.5*(AL+DCP-CV)/AL
      X(N)=-V*DX(2)+W+DX(2)
      Y(N)=-V*DY(2)+Z(N)+DY(2)
      RETURN
      END
C     CONTOUR-PERSPECTIVE PLOTTING / FRANK J RENS / GEOG..COMP.CNTR. / 1
      SUBROUTINE DRAFT
C     WRITTEN BY FRANK J RENS/ GEOGRAPHY / 1966
      COMMON MT,NT,NI,IX,IY,IDX,IDY,ISS,IT,IV,NP,N,JT,PY,REC,CV,X,Y,IPT,
     1INX,INY,DL,AM,IOP,DP,DO,ILAB,XL,YL,ZL,LAB,DX,DY,DZ,XMAX,HT,SCALF,Y
     2MAX,CTHETA,STHETA,FM2,FNN,NN,M,CL,D,NLINES,NCURV,Z
      COMMON TIT
      DIMENSION TIT(18)
      DIMENSION AM(100,100), REC(800), X(1500), Y(1500), IPT(3,3), INX(8
     1), INY(8), IOP(23), XL(10), YL(10), ZL(10), LAB(10,6), DX(2), DY(2
     2), DZ(2), HT(10), CTHETA(4), STHETA(4)
      DIMENSION Z(1500)
      DIMENSION IRES(3)
1     KTYPE=3
      DO 21 I=2,14
      IF (IOP(I)) 21,21,2
2     GO TO (3,3,4,5,21,21,21,8,8,19,21,21,21,14), I
C     ROTATION ABOUT X
3     D1=1.
      D2=0.
      D3=0.
      D4=0.
      D5=CTHETA(2)
      D8=STHETA(2)
      D7=0.
```

```
      D6=-D8
      D9=D5
      GO TO 6
C     ROTATION ABOUT Y
4     D1=CTHETA(3)
      D2=0.
      D3=STHETA(3)
      D4=0.
      D5=1.
      D6=0.
      D7=-D3
      D8=0.
      D9=D1
      GO TO 6
C     ROTATION ABOUT Z
5     D1=CTHETA(4)
      D4=STHETA(4)
      D3=0.
      D2=-D4
      D5=D1
      D6=0.
      D7=0.
      D8=0.
      D9=1.
C     ROTATE
6     CVD3=CV*D3
      CVD6=CV*D6
      CVD9=CV*D9
      DO 7 J=1,N
      XX=D1*X(J)+D2*Y(J)+CVD3
      YY=D4*X(J)+D5*Y(J)+CVD6
      Z(J)=D7*X(J)+D8*Y(J)+CVD9
      X(J)=XX
7     Y(J)=YY
      GO TO 21
C     PRINT HEADING INFORMATION
    8 CONTINUE
      TITCOD=.9
      IF (I-8) 9,9,10
9     NN1=2
      WRITE (6,30) DP,DO
      GO TO 11
10    NN1=1
11    CALL PSYMB (.2,1.,-.1,63HUNIVERSITY OF MICHIGAN COMPUTING CENTER/D
     1EPARTMENT OF GEOGRAPHY,90.,63)
      CALL TIME(5,-1,IRES)
      CALL PSYMB(.4,3.,.1,IRES,90.,12)
      CALL PSYMB (.6,2.,-.1,28HSCALE, COORDINATE UNITS/INCH,90.,28)
      CALL PENMBR(.6,4.7,.1,SCALE,90.,'WF6.2*')
      IF (IOP(12)) 12,13,12
12    CALL PSYMB (TITCOD,3.,.14,TIT(1),90.,48)
13    IOP(8)=0
      IOP(9)=0
      GO TO 21
C     PLOT LABEL ON GRAPH
14    DO 18 J=1,ILAB
```

```
      GO TO (16,15), NN1
15    E=(DO-ZL(J))/DP*SCALE
      F=(DO/DP)*SCALE
      C1=(XL(J)-FNN)/E+XMAX
      C2=(YL(J)-FM2)/E+YMAX
      GO TO 17
16    C1=XL(J)/SCALE+XMAX
      C2=YL(J)/SCALE+YMAX
      F=SCALE
17    HT(J)=HT(J)/F
18    CALL PSYMB (C2,C1,HT(J),LAB(1,J),90.0,36)
      IOP(14)=0
      GO TO 21
C     TRANSLATE DATA
19    DO 20 J=1,N
      X(J)=X(J)+DX(1)
      Y(J)=Y(J)+DY(1)
20    Z(J)=Z(J)+DZ(1)
C     PLOT PLANE OR PERSPECTIVE VIEW OF SURFACE
21    CONTINUE
      IOPC=IOP(2)+IOP(3)+IOP(4)
      DO 29 J=1,N
      IF (IOPC) 22,22,23
22    Z(J)=CV
23    GO TO (24,25), NN1
24    C1=X(J)/SCALE+XMAX
      C2=Y(J)/SCALE+YMAX
      GO TO 26
25    E=(DO-Z(J))/DP*SCALE
      C1=X(J)/E+XMAX
      C2=Y(J)/E+YMAX
26    IF (IOP(5)) 28,28,27
27    CALL PROJ (C1,C2,XMAX,YMAX,SCALE)
28    CALL PLOTCC (C2,C1,KTYPE)
      KTYPE=2
29    CONTINUE
      RETURN
30    FCRMAT('0      CISTANCE TO PLANE =',F9.2,', DISTANCE TO OBJECT = '
     1,F9.2)
      END
      SUBROUTINE PROJ (X,Y,XMAX,YMAX,SCALE)
      DATA K/'1'/
C     MAP PROJECTION SUBROUTINE
C     OBLIQUE STEREOGRAPHIC PROJECTION
C     INSERT CARD CONTAINING PROJECTION CONSTANTS IMMEDIATELY AFTER
C     "20" CARD WHICH FOLLOWS THE FIRST USE OF A "05" CARD.
C     PROJECTION CONSTANTS MUST INCLUDE THE LAT/LONG OF THE CENTER OF
C     THE MATRIX TO BE CONTOURED AND THE GRID SIZE IF DIFFERENT FROM
C     1 DEGREE. OTHER PROJECTION CONTROLS DEPEND ON THE SPECIFIC
C     PROJECTION SUBROUTINE. ALL DATA USING "05" IN ONE RUN SHOULD
C     HAVE THE SAME CENTER LAT/LON AND GRID SIZE, AND WILL PLOT ON THE
C     SAME PROJECTION.
C     WIDTH OF MAP WHEN USING "05" SHOULD REFER TO THE WIDTH AT THE
C     CENTER LATITUDE ON THE SQUARE PROJECTION (ASSUMES NORTH
C     ORIENTATION).
      IF (K) 1,4,1
```

```
1       CONTINUE
        RAD=1.0/57.2958
C       READ CENTER OF MAP, INCREMENT, AND PROJECTION CONSTANTS
        READ (7,5) CMLT,CMLG,F,CPLT,CPLG
        IF (F) 3,2,3
2       F=1.0
3       CONTINUE
        WRITE (6,6) CMLT,CMLG,F,CPLT,CPLG
        FSCAL=SCALE*F
        TSCAL=FSCAL/2.0
        CMLTR=CMLT*RAD
        CMLGR=CMLG*RAD
        CPLG=CPLG*RAD
        CPLT=CPLT*RAD
        SNLT=SIN(CPLT)
        CSNLT=COS(CPLT)
        DLG=CMLGR-CPLG
        SLT=SIN(CMLTR)
        CLT=COS(CMLTR)
        CDLG=COS(DLG)
        DEN=1.0+SLT*SNLT+CLT*CSNLT*CDLG
        SXR=CLT*SIN(DLG)/DEN
        SYR=(SLT*CSNLT-SNLT*CLT*CDLG)/DEN
        CMLGR=SXR/RAD
        CMLTR=SYR/RAD
        K=0
4       FLT=CMLT-(Y-YMAX)*FSCAL
        FLG=((X-XMAX)*FSCAL)+CMLG
        FLT=FLT*RAD
        FLG=FLG*RAD
        DLG=FLG-CPLG
        SLT=SIN(FLT)
        CLT=COS(FLT)
        CDLG=COS(DLG)
        DEN=1.0+SLT*SNLT+CLT*CSNLT*CDLG
        SXR=CLT*SIN(DLG)/DEN
        SYR=(SLT*CSNLT-SNLT*CLT*CDLG)/DEN
        SX=SXR/RAD
        SY=SYR/RAD
        X=XMAX+(SX-CMLGR)/TSCAL
        Y=YMAX+(CMLTR-SY)/TSCAL
        RETURN
C
5       FORMAT (5F7.0)
      6 FORMAT('OMAPCENTERLAT/LON',2(F10.4,1X),'INCREMENT',F10.4/'OPROJECT
       1IONCENTERLAT/LON',2(F10.4,1X))
        END
05
15         2.0        1.0        19.0
19
(18F4.1/2F4.1)
01         8.         20.    8.32712
08.308.608.409.107.007.308.407.107.508.207.206.603.704.102.304.806.810.1
11.613.1
09.409.309.209.811.212.612.110.810.411.111.009.207.703.904.704.408.213.4
```

```
15.115.1
12.011.611.913.114.214.713.813.313.413.213.111.510.310.808.808.210.215.9
15.717.2
13.715.115.814.814.915.616.516.616.413.915.613.514.714.216.113.211.814.2
10.711.7
15.416.415.917.615.717.217.917.717.916.817.114.316.814.613.914.216.416.3
17.916.8
18.217.117.917.416.916.719.318.018.416.814.616.417.417.717.115.409.416.9
15.013.7
18.818.717.818.217.518.419.018.418.617.917.818.118.718.418.917.817.318.3
17.818.4
18.718.218.018.017.118.318.218.018.418.117.917.918.018.217.617.617.818.2
18.218.1
12
TEMP AT 200M*SAME/FOLIO2/PLATE2/MP6
20
  +40.0  -54.0    1.0  +54.0  -38.0
13
```

BLOCK DIAGRAM PLOTTING

Purpose: This program is used for rapid plotting of isometric profiles from a rectangular matrix of data.

Description: The block diagrams are drawn from a ROWS x COLS matrix of floating point Z values. Data is read in row by row from the top down (i.e., the origin is (1,1) in the upper left, or Northwest). By simply changing the ROWS parameter to a minus number, the block diagram will be drawn as viewed from the North. Controls and format are read from unit 5; the title and observation deck from unit 4, and plotting is done via unit 9. The program is easy to use since it has many default options, the only essential control variables being ROWS and COLS. This version of the program does not delete hidden lines.

Comments: The program is designed for use with 30 inch CALCOMP plotter. Current dimensioning allows an input matrix of 100 x 100; this may be adjusted by changing the dimensioning of X, Y, XL, and YL. Multiple sets of data may be run using the same vertical exaggeration by setting SAME parameter. Plotting calls are from the standard CALCOMP library.

Data Deck:
1) Control card
2) Variable Format card
3) Title card
4) Observation deck

Data Card Composition: 1) Control card

columns		
1-3	ROWS	Number of rows in input matrix (100 or less). If negative, -ROWS is the number of rows, and a view from the North is plotted. (I3)
4-6	COLS	Number of columns in input matrix (100 or less) (I3)
7-9	RINC	Spacing increment of row profiles Default = 1. (I3)
10-12	CINC	Spacing increment of column profiles. Default = Cols -1, i.e., only the edges are drawn (I3)
13-15	SAME	Number of matrices of input to be run with same controls and format, and same vertical exaggeration. Default = 0 (I3)
16-20	MAXHGT	Floating point height of diagram in inches. Default = 0.5 inches (F5.0)
21-25	MAX	
26-30	MIN	Control vertical exaggeration, in floating point (Default = MAX and MIN of data) (F5.0)
31-35	WIDTH	
36-40	FRONT	Specify horizontal size of diagram in floating point. Default uses EPS to calculate these val-

		ues. (F5.0)
41-45	EPS	Inches between columns of diagram. floating point. Default = 0.1 inches. (F5.0)
46-50	ALF	Isometric viewing angle in degrees floating point. Default = 35. (F5.0)

2) Variable format card

E or F type FORTRAN format for one row of data

3) Title card

Any title (≤ 72 characters)

4) Observation Deck

Observations by rows as described by variable format card.

References: B. Kubert, J. Szabo, and S. Giulieri, "The Perspective Representation of Functions of Two Variables", Journal, ACM, 15,2 (April 1968), pp. 193-204.

R.L. Mitchell, "A Computerized 3-D Plotting Program", Los Angeles, 1967, AD 658857.

```
C BLOCK DIAGRAM PLOTTING PROGRAM
C C FORTRAN TRANSLATION OF A 1965 MAD PROGRAM BY W.R. TOBLER
            DIMENSION FMT(18),TITL(18),X(100,100),Y(100,100),XL(5000),YL
     1(5000)
      INTEGER ROWS,COLS,RINC,CINC,SAME
      INTEGER ROWS1
      REAL MAXHGT,MAX,MIN
999   FORMAT(5I3,7F5.0)
1000  FORMAT(18A4)
1001  FORMAT (2X,18A4)
1003  FORMAT (2X,5I4,2X,6(F10.3,2X))
1004  FORMAT(2X,F10.3)
1005  FORMAT(1H ,I5)
      RAD=174532925E-10
5     READ(5,999) ROWS,COLS,RINC,CINC,SAME,MAXHGT,MAX,MIN,WIDTH,FRONT,
     1EPS,ALF
C READS FORMAT FOR ONE ROW OF DATA MATRIX
      READ(5,1000) FMT
      WRITE(6,1001) (FMT(I),I=1,18)
      IVU =0
      IF (ROWS) 500,500,501
 500  IVU=1
      ROWS=-ROWS
      ICOLS=COLS+1
 501  GRID=ROWS*COLS
      IF(RINC) 50,50,52
50    IF(CINC) 51,51,52
51    RINC=1
52    KK=0
      IF (ALF) 56,55,56
55    ALF=35.0
56    IF(ALF-90) 57,57,55
57    ALF = ALF*RAD
      SALF = SIN(ALF)
      CALF = COS(ALF)
10    CONTINUE
      READ(4,1000)(TITL(I),I=1,18)
      WRITE(6,1001) (TITL(I),I=1,18)
      WRITE(6,1005)SAME
      CALL PLTXMX(30.0)
      ROWS1 = ROWS+1
      DO 60 II=1,ROWS
      I = ROWS1-II
C READS DATA MATRIX
      IF (IVU ) 502,502,503
 502  READ (4,FMT) (X(I,J),J=1,COLS)
      GO TO 505
 503   READ(4,FMT) (Y(II,J),J=1,COLS)
      DO 504 J=1,COLS
      JJ=ICOLS-J
 504  X(II,JJ)=Y(II,J)
 505  IF (I-1) 3001,3000,3001
3000  WRITE(6,1003)ROWS,I,II,ROWS1,SAME,X(I,1),X(I,2),
     1X(I,3),X(I,4),X(I,5),X(I,6)
3001  IF(I-ROWS)3003,3002,3003
```

```
3002  WRITE(6,1003)ROWS,I,II,ROWS1,SAME,X(I,1),X(I,2),
     1X(I,3),X(I,4),X(I,5),X(I,6)
3003  CONTINUE
60    CONTINUE
      IF(RINC) 61,61,62
61    RINC = ROWS-1
62    IF(CINC) 63,63,64
63    CINC = COLS-1
64    IF(SAME) 67,67,65
65    IF(KK) 66,66,20
66    KK = KK+1
67    CALL STATS(ROWS,COLS,X,ZBAR,ZVAR,ZSIG,ZMAX,ZMIN,ROWS1)
      IF(MAX-MIN) 68,69,68
68    ZMAX = MAX
      ZMIN = MIN
C     HORIZONTAL SCALES
69    IF(WIDTH) 70,71,70
70    EPS = (WIDTH-1)/(COLS-1+(ROWS-1)*CALF)
      GO TO 75
71    IF (EPS) 75, 72, 75
72    IF(FRONT) 73,74,73
73    EPS = FRONT/(COLS-1)
      GO TO 75
74    EPS = 0.1
75    WIDTH = EPS*((COLS-1)+(ROWS-1)*CALF)+1.0
      FRONT = EPS*(COLS-1)
C     VERTICAL SCALES
      RANGE = ZMAX-ZMIN
      IF(MAXHGT) 76,76,77
76    MAXHGT=0.5
77    DZ = MAXHGT/RANGE
      XCON=0.5
      YCON=1.5
      SUM=YCON-0.1
      SF = SALF*EPS
20    CONTINUE
      WRITE (6,1003) ROWS,COLS,RINC,CINC,SAME,FRONT,EPS,SALF,MAXHGT,
     1 ZMAX,ZMIN
      CALLPSYMB(0.5,0.1,.1,' GEOGRAPHY DEPARTMENT, UNIVERSITY OF MICHIGA
     1N    ',0.0,45)
      CALL PSYMB(0.5,0.5,.1,TITL,0.,72)
C     BEGIN PLOTTING
C     COMPUTE THE DIAGRAM LATTICE POINT COORDINATES
      DO 30 I=1,ROWS
      M = I-1
      RC1 = SF*M+YCON
      RC2 = M*CALF
      DO 30 J=1,COLS
      Y(I,J)=(X(I,J)-ZMIN)*DZ+RC1
      X(I,J) = EPS*((J-1)+RC2)+XCON
30    CONTINUE
      BOT=0.0
       WRITE (6,1004) BOT
C     DRAW PROFILES ACROSS THE ROWS
      DO 40 I=1,ROWS,RINC
        DO 2000 J=1,COLS
```

```
      XL(J)=X(I,J)
2000  YL(J)=Y(I,J)
      CALL PLINE (XL(1),YL(1),COLS,1.0,0,0)
40    CONTINUE
       ACT=1.0
       WRITE (6,1004) BOT
C     DRAW PROFILES DOWN THE COLUMNS
      DO 42 J=1,COLS,CINC
      DO 41 I=1,ROWS
      XL(I) = X(I,J)
      YL(I) = Y(I,J)
41    CONTINUE
      CALL PLINE(XL(1),YL(1),ROWS.1,0,0,0)
42    CONTINUE
      BOT=BOT+2.0
       WRITE (6,1004) BOT
C     PUT BOTTOM ON THE DIAGRAM
      XL(1)=XCON
      YL(1)=Y(1,1)
      XL(2)=XCON
      YL(2)=SUM
      XL(3)=X(1,COLS)
      YL(3)=SUM
      XL(4)=XL(3)
      YL(4)=Y(1,COLS)
      CALL PLINE (XL(1),YL(1),4,1.0,0,0)
      XL(1)=X(1,COLS)
      XL(2)=X(ROWS,COLS)
      YL(3)=Y(ROWS,COLS)
      YL(1)=SUM
       YL(2)=RC1-0.1
      XL(3)=XL(2)
      CALL PLINE(XL(1),YL(1),3,1,0,0,0)
      BCT=BOT+4.0
       WRITE (6,1004) BOT
      SAME = SAME-1
      CALL PLTEND
      WRITE(6,114)(TITL(I),I=1,18)
114    FORMAT(1H1,18A4)
      IF(SAME) 5,5,10
      END
      SUBROUTINESTATS(NR,NC,X,XBAR,XVAR,XSIG,XMAX,XMIN,PRINT)
      DIMENSION X(100,100)
      INTEGER PRINT
200   FORMAT(1H ,2(I5,1X),5(F10.3,1X))
      FN=NR*NC
      XMAX=X(1,1)
      XMIN=XMAX
      XSUM=0.
      XSQR=0.
      DO10I=1,NR
      DO10J=1,NC
      XX=X(I,J)
      XMAX=AMAX1(XMAX,XX)
      XMIN=AMIN1(XMIN,XX)
      XSUM=XSUM+XX
```

```
10  XSQR=XSQR+XX*XX
    XBAR=XSUM/FN
    XVAR=(XSQR-2.*XBAR*XSUM+FN*XBAR*XBAR)/FN
    XSIG=SQRT(XVAR)
    IF(PRINT)30,30,20
20  WRITE(6,200)NR,NC,XBAR,XVAR,XSIG,XMAX,XMIN
30  RETURN
    END
```

SPATIAL DERIVATIVE PROGRAM

Purpose: This program calculates the absolute value of the gradient ("slope") for data given as a geographical matrix.

Description: The spatial derivative is calculated by the equation

$$\nabla(x,y) = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2}$$

Where $\nabla(x,y)$ is the magnitude of the gradient obtained from a pair of orthogonal partial derivatives of Z, approximated by finite differences. That is, ∇_{ij} is calculated from its four neighbors ($Z_{i\pm1,\ j\pm1}$). Edge effects appear at the boundaries, where the three neighboring points are used, and in the corners where the two neighboring points are used in the calculations. The program will calculate as high as nine derivatives. The results may also be punched out.

Comments: Currently the program will accomodate an input matrix of Z values of 100 x 100. This may be adjusted by changing the dimensioning of values Z and W. This program may be used at a remote terminal which has access to shared files with no difficulty. Punched output may be used directly (with proper control cards) in the block profile and contour mapping programs.

Data Deck:
1) Control Card
2) Title Card
3) Variable Format Card
4) Observation Deck

Data Card Composition: 1) Control Card

columns

1-3	ROWS	Number of rows of input matrix
4-6	COLS	Number of columns of input matrix
7	G	One for first derivative, two for second derivative, etc.
8	PUN	One if any derivatives are to be punched out, zero otherwise
9-14	SCALE	Distance between matrix points (F6.0)
15	SPUN	If zero punches out all derivatives (PUN =1), if 1-9 will only punch out derivative number indicated (convenient for remote terminal usage).

2) Title Card

Any title (≤ 80 characters)

3) Variable Format Card

E or F type FORTRAN format for one row of data (≤ 80 characters)

4) Observation Deck

Observations by rows as described by variable format card.

Reference: C.M. Davis, "A Study of the Land Type", University of Michigan, Contract DA-31-124-ARO-D-456, UM 08055-2-F, March 1969, pp. 59-85.

```
C PROGRAM TO COMPUTE THE K'TH(MAX 9) DERIVATIVE AT GRID POINTS
C W.R. TOBLER / UNIVERSITY OF MICHIGAN / GEOGRAPHY
C ADAPTED TO FORTRAN BY H. MOELLERING
      INTEGER ROWS,COLS,G,PUN,SPUN
      DIMENSION TITL(20),FMT(20),Z(100,100),W(100,100),DIV(9)
      DATA DIV/'1ST','2ND','3RD','4TH','5TH','6TH','7TH','8TH','9TH'/
  100 FORMAT(2I3,2I1,F6.0,I1)
  101 FORMAT(20A4)
  102 FORMAT(1H1,20A4)
  103 FORMAT('0','N=',I5,5X,'ROWS=',I3,5X,'COLS=',I3,5X,'K=',I2,5X,
     1'SCALE=',F6.0,5X,'G=',I2,2X,'SPUN=',I2///55X,'THE',1X,A3,1X,
     2'DERIVATIVE IS')
  104 FORMAT(8F10.3)
  105 FORMAT(10X,'END OF DATA SET')
  106 FORMAT(1H0,' END OF DERIVATIVES')
  108 FORMAT('0',12F10.3/(' ',12F10.3))
  109 FORMAT('1',20A4)
  110 FORMAT ('0  PROGRAM IS PUNCHING OUT RESULTS OF ALL DERIVATIVES REQ
     1UESTED')
  111 FORMAT('0  PROGRAM IS PUNCHING OUT RESULTS OF ONLY DERIVATIVE #',
     1I2)
    5 READ(5,100) ROWS,COLS,G,PUN,SCALE,SPUN
      READ(5,101)  TITL
      WRITE(6,109)  TITL
      READ(5,101)  FMT
      WRITE (6,101)  FMT
      IF(SCALE.LE.0.0) XH2=2.0
      IF(SCALE.GT.0.0) XH2=SCALE*2.0
      IF(G.LE.0) G=1
      DO 70 I=1,ROWS
   70 READ (5,FMT)  (Z(I,K),K=1,COLS)
      DO 60 I=1,ROWS
   60 WRITE(6,108) (Z(I,K),K=1,COLS)
      CALL STATS(ROWS,COLS,Z,ZBAR,DUM,ZSIG,ZMAX,ZMIN,K)
      IF(PUN.GE.1 .AND. SPUN.EQ.0) WRITE(6,110)
      IF(PUN.GE.1 .AND. SPUN.GT.0) WRITE(6,111)  SPUN
      DO 50 K=1,G
      N=ROWS*COLS
      WRITE(6,102)  TITL
      WRITE(6,103) N,ROWS,COLS,K,SCALE,G,SPUN,DIV(K)
      DO 30 M=1,ROWS
      MM=M-1
      M1=M+1
      DO 20 J=1,COLS
      JJ=J-1
      J1=J+1
      IF(JJ.LT.1) DX=((Z(M,J1)-Z(M,J))*2.0)/XH2
      IF(J1.GT.COLS) DX=((Z(M,J)-Z(M,JJ))*2.0)/XH2
      IF(JJ.GE.1 .AND. J1.LE.COLS) DX=(Z(M,J1)-Z( M,JJ))/XH2
      IF(MM.LT.1) DY=((Z(M,J)-Z(M1,J))*2.0)/XH2
      IF(M1.GT.ROWS) DY=((Z(MM,J)-Z(M,J))*2.0)/XH2
      IF(MM.GE.1 .AND. M1.LE.ROWS) DY=(Z(MM,J)-Z(M1,J))/XH2
      W(M,J)=SQRT(DX*DX+DY*DY)
   20 CONTINUE
      IF(PUN.GE.1 .AND. SPUN.EQ.0) WRITE(7,104) (W(M,J),J=1,COLS)
```

```
      IF(PUN.GE.1.AND.SPUN.EQ.K)  WRITE(7,104) (W(M,J),J=1,COLS)
   30 CONTINUE
      DO 32 I=1,ROWS
   32 WRITE(6,108) (W(I,J),J=1,COLS)
      CALL STATS(ROWS,COLS,W,WBAR,DUM,WSIG,WMAX,WMIN,K)
      IF(G.LE.1) GO TO 42
      DO 40 M=1,ROWS
      DO 40 J=1,COLS
      Z(M,J)=W(M,J)
   40 CONTINUE
   42 IF(PUN.GE.1 .AND. SPUN.EQ.0) WRITE (7,105)
   50 CONTINUE
      WRITE(6,106)
      GO TO 5
      END
      SUBROUTINESTATS(NR,NC,X,XBAR,XVAR,XSIG,XMAX,XMIN,PRINT)
      DIMENSION X(100,100)
      INTEGER PRINT
   15 FORMAT('0','NR=',I3,1X,'NC=',I3,1X,'XBAR=',E15.8,   'XVAR=',E15.8,
     1    'XSIG=',E15.8,    'XMAX=',E15.8,    'XMIN=',E15.8)
      FN=NR*NC
      XMAX=X(1,1)
      XMIN=XMAX
      XSUM=0.
      XSQR=0.
      DO10I=1,NR
      DO10J=1,NC
      XX=X(I,J)
      XMAX=AMAX1(XMAX,XX)
      XMIN=AMIN1(XMIN,XX)
      XSUM=XSUM+XX
   10 XSQR=XSQR+XX*XX
      XBAR=XSUM/FN
       XVAR=(XSQR-2.*XBAR*XSUM+FN*XBAR*XBAR)/FN
      XSIG=SQRT(XVAR)
      IF(PRINT.GT.0) WRITE (6,15) NR,NC,XBAR,XVAR,XSIG,XMAX,XMIN
      RETURN
      END
  5 1031     1.2
MOFILLERING'S FOLLY
(10F5.1)
  1.0  1.0  1.0  3.4  0.0  4.2  4.2  4.2  0.0  0.0
  1.0  1.0  1.0  2.0  4.2  3.1  4.2  0.0  0.0  0.0
  0.0  0.0  3.0  4.0  4.9  0.0  4.2  0.0  0.0  0.0
  0.0  0.0  0.0  0.0  3.2  1.6  4.2  3.1  0.0  1.2
  0.0  0.0  1.2  3.9  4.2  1.0  1.0  2.1  1.2  1.2
```

TREND SURFACES BY EIGENVECTOR DYADS

Purpose: This program produces a series of trend surfaces each consecutive member of which better approximates an original input distribution.

Description: It is assumed that the geographical data are given in the form of an m by n geographical matrix $Z = (z_{ij})$. This matrix is decomposed by the program into the sum of k matrices of order m by n. Each of these trend matrices is in turn separated into the product of an m by one vector f_k multiplied by a one by n vector g_k, i.e., the model is

$$Z = f_1(x)g_1(y) + \ldots + f_k(x)g_k(y)$$

where x can be interpreted as the column coordinate j, and y the row coordinate. The column vector f_k is the k^{th} eigenvector of ZZ^t, and the row vector g_k is obtained as $g_k = Z^t f_k / f_k^t f_k$. The program orders the dyads (f, g) from the largest eigenvalue to the smallest, each dyad accounting for the maximum amount of variance which can be extracted by any product of the form $f(x)g(y)$. The number of eigenvector dyads required for a complete representation of Z is at most the lesser of m,n. The current dimensioning allows up to m = n = 70. The program reads from 7, writes on 6, and punches on 5.

Data Deck:
1) Control Card.
2) Format Card.
3) Observation deck.

Repeat 1 through 3 as many times as desired.

Data Card Composition: 1) Control Card

column

1-2	NR	number of rows
3-4	NC	number of columns
5-6	MAXE	Maximum number of dyads to be extracted
7	IFPCH	2 if for each dyad the X matrix and the cumulative Y matrix are to be punched. 1 if for each dyad only the cumulative Y matrix is to be punched. 0 if no punching.

2) Format Card

The variable format, including left and right parentheses, is punched in the first 72 columns of this card in E or F type FORTRAN notation.

References: Waldo R. Tobler, "Geographical Filters and their Inverses," Geographical Analysis, Vol 1, (1969), pp. 243-253.

Peter R. Gould, "On the Geographical Interpretation of Eigenvalues," Transactions, Institute of British Geographers, (1967), pp. 53-86.

```
C      TREND SURFACES VIA EIGENVECTOR DYADS
       DIMENSION Z(70,70), ZT(70,70), ZZT(70,70), FMT(18), EVAL(70), A(70
      1), B(70), EVEC(70,70)
31     FORMAT (3I2,I1)
32     FORMAT (18A4)
33     FORMAT (10H1CONTROLS ,4(2X,I2)/1H 18A4)
34     FORMAT ('ORIGINAL MATRIX')
35     FORMAT (1H 10F10.2/)
36     FORMAT ('-EIGENVALUES'/'TRACE=',E15.6,10X,'N=',I5/'VECTOR',10X,
      1'VALUE',10X,'CUMULATIVE PERCENT'/1H0)
37     FORMAT (1H0,I2,E15.6,2PF24.2)
38     FORMAT ('1 DYAD NUMBER',I5)
    39 FORMAT (1H0,'EIGENVECTOR TRANSPOSED')
    40 FORMAT (1H0,'THE OTHER VECTOR')
    41 FORMAT (1H0,'THE PRODUCT MATRIX')
42     FORMAT ('1 CUMULATIVE EXPANSION, THRU DYAD',I5)
    43 FORMAT (1H-,'EXPANSION COMPLETED')
    44 FORMAT (1H0,34HMATRIX DIMENSIONS EXCEED ALLOTMENT)
1      READ (7,31) NR,NC,MAXE,IFPCH
       READ (7,32) (FMT(J),J=1,18)
       WRITE (6,33) NR,NC,MAXE,IFPCH,(FMT(K),K=1,18)
       WRITE (6,34)
       IF (MAX0(NR,NC) .GT. 70)  GO TO 51
C      READ AND PRINT DATA MATRIX
       DO 2 I=1,NR
       READ (7,FMT) (Z(I,J),J=1,NC)
2      WRITE (6,35)  (Z(I,J),J=1,NC)
C      FORM TRANSPOSE
       DO 3 I=1,NR
       DO 3 J=1,NC
3      ZT(J,I)=Z(I,J)
       DO 4 I=1,NR
       DO 4 J=1,NR
       ZZT(I,J)=0.0
4      EVEC(I,J)=0.0
C      FORM PRODUCT
       DO 5 K=1,NC
       DO 5 I=1,NR
       DO 5 J=1,NR
5      ZZT(I,J)=ZZT(I,J)+Z(I,K)*ZT(K,J)
       MAXE = MIN0(NR,NC)
11     TRACE=0.0
       DO 12 K=1,NR
12     TRACE=TRACE+ZZT(K,K)
C      PRINT EIGENVALUES
       WRITE (6,36) TRACE,MAXE
       PERC=0
       CALL EIGEN (ZZT,EVEC,NR,EVAL,MAXE,70)
       DO 15 K=1,MAXE
       IF (TRACE) 13,14,13
13     PERC=PERC+EVAL(K)/TRACE
       WRITE (6,37) K,EVAL(K),PERC
       GO TO 15
14     WRITE (6,37) K,EVAL(K)
15     CONTINUE
```

```
      DO 16 I=1,NR
      DO 16 J=1,NC
16    Z(I,J)=0.0
C     OBTAIN DYADS
      DO 29 M=1,MAXE
      WRITE (6,38) M
      DO 17 I=1,NC
17    B(I)=0.0
      ENORM=0.0
      DO 18 I=1,NR
      A(I)=EVEC(I,M)
18    ENORM=ENORM+A(I)**2
      IF (ENORM) 19,30,19
19    WRITE (6,39)
      WRITE (6,35)  (A(K),K=1,NR)
      DO 20 K=1,NR
      DO 20 I=1,NC
20    B(I)=B(I)+ZT(I,K)*A(K)
      DO 21 I=1,NC
21    B(I)=B(I)/ENORM
      WRITE (6,40)
      WRITE (6,35) (B(K),K=1,NC)
C     OBTAIN PRODUCT
      DO 22 I=1,NR
      DO 22 J=1,NC
22    ZZT(I,J)=0.0
      DO 23 I=1,NR
      DO 23 J=1,NC
23    ZZT(I,J)=ZZT(I,J)+A(I)*B(J)
      WRITE (6,41)
      DO 25 K=1,NR
      WRITE (6,35) (ZZT(K,J),J=1,NC)
      IF (IFPCH - 1)  25,25,24
24    WRITE (5,FMT) (ZZT(K,J),J=1,NC)
25    CONTINUE
C     PRINT CUMULATIVE PRODUCTS
      WRITE (6,42) M
      DO 28 I=1,NR
      DO 26 J=1,NC
26    Z(I,J)=Z(I,J)+ZZT(I,J)
      WRITE (6,35) (Z(I,K),K=1,NC)
      IF (IFPCH) 28,28,27
27    WRITE (5,FMT) (Z(I,K),K=1,NC)
28    CONTINUE
29    CONTINUE
30    WRITE (6,43)
      GO TO 1
   51 PRINT 44
      END
C     EIGENVALUES AND NORMALIZED EIGENVECTORS OF A REAL SYMMETRIC MATRIX
C     EIGENVALUES ARE RETURNED IN VALU AND NORMALIZED EIGENVECTORS ARE
C     STORED IN B.  NSUB IS ORDER OF MATRICES A AND B AND MSUB IS THE
C     NUMBER OF ROOTS AND VECTORS DESIRED.
C
      SUBROUTINE EIGEN (A,B,NSUB,VALU,MSUB,MD)
C
```

```
      DIMENSION A(MD,MD), B(MD,MD), VALU(MD), T(70,3), DIAG(70), SUPERD(
     170), WVEC(70), PVEC(70),QVEC(70), VALL(70), Q(70), U(70), INDEX(70
     2), V(70)
C
      EQUIVALENCE (WVEC,VALL,U),            (PVEC,QVEC,Q,V), (I1,T1),
     1(I2,T2),       (TEMP,TO), (SUM,MATCH), (I,P), (DIV,SCALAR,TAU), (AN
     2CRM2,ANORM), (VTEMP,VNORM2,VNORM)
C
C
C      INITIALIZATION
       N=NSUP
       M=MSUB
       NP1=N+1
       NM1=N-1
        E1=1.E-8
C      GENERATE IDENTITY MATRIX
       DO 3 I=1,N
       DO 3 J=1,N
       IF (I-J) 2,1,2
 1     B(I,J)=1.
       GO TO 3
 2     B(I,J)=0.
 3     CONTINUE
C      HOUSEHOLDER SIMILARITY TRANSFORMATION TO CO-DIAGONAL FORM
C      REDUCE COLUMN OF MATRIX
       DO 14 I=1,NM1
       IF (I-NM1) 4,13,4
 4     I1=I+1
       I2=I1+1
       SUM=0.
       DO 5 J=I2,N
 5     SUM=SUM+A(J,I)**2
       IF (SUM) 6,13,6
 6     J=I1
       TEMP=A(J,I)
C                                                          TERMINAL-F FLAG
       SUM=SQRT(SUM+TEMP**2)
       A(J,I)=-SIGN (SUM,TEMP)
C                                                          TERMINAL-F FLAG
       WVEC(J)=SQRT(1.+ABS (TEMP)/SUM)
       DIV=SIGN (WVEC(J)*SUM,TEMP)
       DO 7 J=I2,N
 7     WVEC(J)=A(J,I)/DIV
       SCALAR=0.
       DO 9 J=I1,N
       PVEC(J)=0.
       DO 8 K=I1,N
 8     PVEC(J)=PVEC(J)+A(K,J)*WVEC(K)
       SCALAR=SCALAR+PVEC(J)*WVEC(J)
 9     CONTINUE
       SCALAR=SCALAR/2.
       DO 10 J=I1,N
       QVEC(J)=PVEC(J)-SCALAR*WVEC(J)
       DO 10 K=I1,J
       A(K,J)=A(K,J)-(WVEC(K)*QVEC(J)+WVEC(J)*QVEC(K))
       A(J,K)=A(K,J)
```

```
10    CONTINUE
C     SAVE ROTATION FOR LATER APPLICATION TO CO-DIAGONAL VECTORS
      DO 12 K=2,N
      TEMP=0.
      DO 11 J=I1,N
11    TEMP=TEMP+WVEC(J)*B(J,K)
      DO 12 J=I1,N
      B(J,K)=B(J,K)-WVEC(J)*TEMP
12    CONTINUE
C     MOVE CO-DIAGONAL FORM ELEMENTS FOR ITERATIVE PROCEDURE
13    J=I
      DIAG(I)=A(J,I)
      SUPERD(I)=A(J+1,I)
14    CONTINUE
      DIAG(N)=A(N,N)
C     DETERMINE EIGENVALUES FROM STURM CHAIN OF CO-DIAGONAL MINORS
C     CALCULATE NORM OF MATRIX AND INITIALIZE EIGENVALUE BOUNDS
      ANORM2=DIAG(1)**2
      DO 15 L=2,N
      Q(L-1)=SUPERD(L-1)**2
      ANORM2=DIAG(L)**2+Q(L-1)+Q(L-1)+ANORM2
15    CONTINUE
C                                                         TERMINAL-F FLAG
      ANORM=SQRT(ANORM2)
      DO 16 L=1,M
      VALU(L)=ANORM
      VALL(L)=-ANORM
16    CONTINUE
      EPS1=ANORM*E1
      IF (EPS1) 17,73,17
C     CHOOSE NEW TRIAL VALUE WHILE TESTING BOUNDS FOR CONVERGENCE
17    DO 35 L=1,M
      ITER=0
      VTEMP=EPS1
18    TAU=(VALU(L)+VALL(L))/2.
      IF (ITER-10) 20,19,20
19    VTEMP=VTEMP*10.
      ITER=0
20    IF (2.*(TAU-VALL(L))-VTEMP) 35,35,21
C     DETERMINE SIGNS OF PRINCIPAL MINORS
21    MATCH=0
      ITER=ITER+1
      T2=0.
      T1=1.
      DO 30 L1=1,N
      P=DIAG(L1)-TAU
      IF (T2) 23,22,23
22    T1=SIGN (1.,T1)
23    IF (T1) 25,24,25
24    T0=-SIGN (1.,T2)
      T2=0.
      IF (Q(L1-1)) 26,22,26
25    T0=P-Q(L1-1)*T2/T1
      T2=1.
C     COUNT AGREEMENTS IN SIGN (ZERO CONSIDERED POSITIVE)
26    IF (T0) 29,27,28
```

```
27    T2=T1
      IF (T2) 29,28,28
28    MATCH=MATCH+1
29    T1=T0
30    CONTINUE
C     ESTABLISH TIGHTER BOUNDS ON EIGENVALUES
      DO 34 L1=L,M
      IF (L1-MATCH) 33,33,31
31    IF (VALU(L1)-TAU) 18,18,32
32    VALU(L1)=TAU
      GO TO 34
33    VALL(L1)=TAU
34    CONTINUE
      GO TO 18
35    CONTINUE
C     EIGENVECTORS OF CO-DIAGONAL SYMMETRIC MATRIX -- INVERSE ITERATION
C     CHECK FOR REPEATED VALUE
      DO 68 I=1,M
      IF (I-2) 37,36,36
36    IF (VALU(I-1)-VALU(I)-1.E-3) 38,37,37
37    I1=-1
38    I1=I1+1
C     TRIANGULARIZE CO-DIAGONAL FORM AFTER EIGENVALUE SUBTRACTION
      DO 43 L=1,N
      V(L)=EPSI
      T(L,2)=DIAG(L)-VALU(I)
      IF (L-N) 40,39,40
39    T(L,3)=0.
      GO TO 43
40    T(L,3)=SUPERD(L)
      IF (T(L,3)) 42,41,42
41    T(L,3)=EPSI
42    T(L+1,1)=T(L,3)
43    CONTINUE
      DO 50 J=1,N
      T(J,1)=T(J,2)
      T(J,2)=T(J,3)
      T(J,3)=0.
      VTEMP=ABS (T(J,1))
      IF (J-N) 46,44,46
44    IF (VTEMP) 50,45,50
45    T(J,1)=EPSI
      GO TO 50
46    INDEX(J)=0
      IF (ABS (T(J+1,1))-VTEMP) 49,49,47
47    INDEX(J)=1
      DO 48 K=1,3
      VTEMP=T(J,K)
      T(J,K)=T(J+1,K)
      T(J+1,K)=VTEMP
48    CONTINUE
49    VTEMP=T(J+1,1)/T(J,1)
      U(J)=VTEMP
      T(J+1,2)=T(J+1,2)-VTEMP*T(J,2)
      T(J+1,3)=T(J+1,3)-VTEMP*T(J,3)
50    CONTINUE
```

```
      ITER=1
      IF (I1) 58,51,58
C     BACK SUBSTITUTE TO OBTAIN EIGENVECTOR
51    DO 52 L1=1,N
      L=NP1-L1
      V(L)=(V(L)-T(L,2)*V(L+1)-T(L,3)*V(L+2))/T(L,1)
52    CONTINUE
      GO TO (53,58), ITER
C     PERFORM SECOND ITERATION
53    ITER=2
54    DO 57 L=2,N
      IF (INDEX(L-1))55,56,55
55    VTEMP=V(L-1)
      V(L-1)=V(L)
      V(L)=VTEMP
56    V(L)=V(L)-U(L-1)*V(L-1)
57    CONTINUE
      GO TO 51
C     ORTHOGONALIZE VECTOR TO OTHERS ASSOCIATED WITH REPEATED ROOT
58    IF (I1) 59,62,59
59    DO 61 L1=1,I1
      K=I-L1
      VTEMP=0.
      DO 60 J=1,N
60    VTEMP=VTEMP+A(J,K)*V(J)
      DO 61 J=1,N
61    V(J)=V(J)-A(J,K)*VTEMP
62    GO TO (54,63), ITER
C     NORMALIZE VECTOR TO UNIT LENGTH
63    VNORM2=0.
      SUM=0.
      DO 65 L=1,N
      IF (SUM-ABS (V(L))) 64,65,65
64    SUM=ABS (V(L))
65    CONTINUE
      DO 66 L=1,N
      V(L)=V(L)/SUM
66    VNORM2=VNORM2+V(L)**2
C                                                      TERMINAL-F FLAG
      VNORM=SQRT(VNORM2)
      DO 67 J=1,N
67    A(J,I)=V(J)/VNORM
68    CONTINUE
C     ROTATION OF CO-DIAGONAL VECTORS INTO MATRIX EIGENVECTORS
      DO 70 I=1,M
      DO 69 K=2,N
      U(K)=0.
      DO 69 J=2,N
69    U(K)=U(K)+B(J,K)*A(J,I)
      DO 70 J=2,N
70    A(J,I)=U(J)
C     NORMALIZE LENGTH OF VECTORS TO EIGENVALUES AND STORE IN B(I,J)
      DO 72 J=1,M
      IF (VALU(J)) 73,73,71
C                                                      TERMINAL-F FLAG
71    VTEMP=SQRT(VALU(J))
```

```
      DO 72 I=1,N
72    B(I,J)=A(I,J)*VTEMP
73    RETURN
      END
 4 4 40
(4F4.0)
 138   0 -10 -80
 156  -5 -16 -99
 117 -13 -27-101
 102 -16 -36-110
```

GEOGRAPHICAL GROUPING

Purpose: The program groups observations using the criterion of Euclidean proximity in a p dimensional vector space; if the data are given in the form of geographical matrices $Z = (z_{ij})$ the program can automatically impose a geographical contiguity constraint on the grouping.

Description: The program is basically a modification of an earlier program by Neely and Mazukelli (see references below). The major modification consists of (1) an option to read data in the form of geographical matrices, and (2) restriction to only one of the possible grouping algorithms. The use of data not in geographic matrix form is still permitted, and contiguity matrices can be prespecified in these cases. Each of the n observations consists of p variables.

The original n observations are initially considered to consist of n p-dimensional groups each containing one element. The grouping procedure examines all of the n (n - 1) / 2 squared distances among these n observations and joins the two groups separated by the minimum distance. Each step of the grouping procedure decreases by one the number of groups still to be consolidated. Each grouping replaces two joined groups by a new group located at the center of mass of the pair, and this group then contains all the elements of the pair. Before a new step is initiated the distances from each of the remaining groups to the new group are calculated; these replace the distances to both of the component groups, with a consequent reduction by one of the distances to be examined at the next step. This aggregation requires n - 1 steps, after which all observations are in the final group. The minimum distance is added to the value of SUM at each step, thus providing a measure of the efficacy of each grouping. If the contiguity option is specified only distances between contiguous groups will be considered in the minimum proximity comparison.

Limits: The current version of the program will accept as many as 400 observations for each of the p variables, where p is defined by

$$n \times p \leq 80{,}200$$

Although subroutine GRPING is called only once, it is here that, for larger problems, the computer time is used. The program searches through the upper half of the distance squared triangle at each step of the grouping algorithm, an operation which requires examination of a number of distances varying from one to the triangle of n. The number of searches for the entire problem of n observations is given by

$$N = (n - 1)\ n\ (n + 1)\ /\ 6$$

The trivial case of two observations produces one search, while 85 observations require over 100,000, and 183 require more than one million. The maximum requires 10,666-600 searches to produce the 399 groupings.

Although 400 observations may seem a reasonably large number, it unfortunately yields geographical matrices of only 20 rows by 20 columns, a rather trivially small size. A nine inch by nine inch aerial photograph could be digitalized to a resolution of only circa one half inch, and a 20 inch topographical map to only a one inch resolution. It appears that other algorithms will be required for geographical grouping; i.e., capable of operating on the circa 1,000,000 observations contained in a single still coarsely digitalized geographical map or aerial photograph. The severity of the problem does not grow as rapidly with increases in p, the number of variables (e.g., for multispectral imagery or map overlays) and the contiguity constraint, essentially a local neighborhood operator, should allow greater efficiencies than are given in the present program.

There is an option not to print (IDSP = 1) the original distance matrix; it requires n/10 pages to print this. Forty pages, each containing up to five hundred E-format numbers, are best left in memory, although smaller problems and illustrative examples may utilize this form of output. Option KODE is also best left at 0; the distance matrices at the subsequent steps have similar paper requirements.

Deck Make-Up:

1) Problem card
2) Format card(s)
3) Data cards
4) Contiguity cards (optional)
5) Finish card.

As many sets of cards 1 through 4 as desired may be placed before the Finish card. If IDS (defined below) is 2, repetition of cards 2 and 3 is required for each of the variables specified on the Problem card.

Card Format: Problem Card:

Columns

1-6	PROBLM	
8-10	NOOB	Number of observations
11-14	MVAR	Number of variables
18	NFMT	Number of format cards
22	IDS	Input code:

2 if the variables are in geographical matrix form. In this case each of the p matrices, of n observations each, is read separately, always preceeded by a format card. This option may also be used to read one variable at a time.

1 To read the upper triangular matrix of distances squared, rather than raw data.

0 To read one observation of p variables.

24-26	NR	Number of rows (required only if both IDS and NCLUS are 2)
28-30	NC	Number of columns (required only if both IDS and NCLUS are 2)

34	NCLUS	Contiguities requested
		2 to be computed from matrix data
		1 to be read in from data cards (see below)
		0 to be ignored
38	IDSP	Initial printing of the distances squared (upper triangular matrix)
		1 to be suppressed
		0 to be effected
42	KODE	Stepwise printing of the distances squared.
		1 to be effected
		0 to be suppressed
48	IP	Page control; set equal to one for best results.
49-80	TITLE	An alphanumeric title of fewer than 33 characters.

Format Cards:

There may be as many as nine E or F format cards, including initial and final parentheses. These will describe one complete observation of p variables or n observations on one variable. The word FORMAT must be punched in the first six columns of each card. If IDS is 2 there must be an identical number (NFMT) of format cards placed before the data for each of the p variables, although the data need not be punched in discrete rows if in matrix form and the content of the format cards may vary from variable to variable.

Contiguity Cards:

The contiguity matrix is symmetric, having n rows and n columns. This optional input requires the contiguity matrix in its entirety, except for the last row, although the contiguity states are read only from the upper triangle. Cards are punched according to format 80I1 for each row, with each row beginning on a new card. There will be an integral multiple of (n - 1) cards in the set. A 1 denotes contiguous observations, a blank the converse.

If contiguities are requested for data in geographical matrix form, the neighbors to an observation are defined as those observations which have a side in common with the cell in question: diagonally adjacent cells are not considered neighbors.

Output: In addition to the optional distances, at each step of the algorithm the program provides a statement of which groups were joined and the increment to the sum of squared distances. At each step the groups are labelled by the smaller identification number of the two in the pair. A dendrogram of the grouping is provided, and this depicts the groups present at a particular step by printing a plus sign. Each line of this graph includes one plus sign which is approximately halfway between two plus signs of the previous line, and this represents the two groups which have been joined at this step. The two groups will always be adjacent on the graph. This output permits division of the observations into any number of exclusive groups (partitions), although it is impossible to specify the total number of observations contained in any group at any step other than the first and last.

Programmer: Dierk Rhynsburger; subroutines TRI, ERROR, FORMAT, and PAGE adapted from a similar program at the University of Chicago by Neely and Mazukelli.

Reference: D.F. Marble, Some Computer Programs for Geographic Research, Department of Geography, Northwestern University, Evanston, 1967; program Congroup, pp. 23-34.

```
      DIMENSION A(80200), NO(400), FMT(162), TITLE(8), IFMT(162)
      DIMENSION IB(2), IPA(7), IPB(6), IPC(6), IPE(6)
      COMMON A
      INTEGER WORD, TITLE
      EQUIVALENCE (FMT,IFMT)
      DATA IB/'PROB','FINI'/, IEA/'MAIN'/, IEB/' NOP'/
      DATA IPA/'STAR','T OF',' PRO','BLEM',' - G','ROUP','ING '/
      DATA IPB/'FORM','AT F','OR I','NPUT',' DAT','A   '/
      DATA IPC/'DIST','ANCE',' SQU','ARED',' MAT','RIX '/
      DATA IPE/'INPU','T FO','RMAT',', VA','RIAB','LE  '/
 1001 FORMAT (A4,2X,9I4,5X,I1,8A4)
 2001 FORMAT(23H0 NUMBER OF VARIABLES = I9, 70X,17HSUBROUTINE MAIN    /
     1 26H0 NUMBER OF OBSERVATIONS =  I6,70X,19HDATED 30 JUNE 1969 ///)
 2002 FORMAT (1H0,3HTHE,I6,40H  ELEMENTS OF THE D SQUARED MATRIX HAVE ,
     1    17HBEEN READ AS DATA)
 2003 FORMAT (1H0,3HTHE,I3,40H MATRICES OF VARIABLES HAVE BEEN READ AS,
     1    5H DATA)
 2004 FORMAT (1H0,3HTHE,I3,37H MATRIX OF VARIABLES HAS BEEN READ AS,
     1    5H DATA)
 2005 FORMAT(1H0/1H0 20X A6)
      NPAGE = 1
      NSYZ = 80200
    1 READ 1001, WORD, NOOB, MVAR, NFMT, IDS, NR, NC, NCLUS, IDSP,
     1    KODE, IP, (TITLE(I),I=1,8)
      IF (WORD .EQ. IB(1))  GO TO 4
      IF (WORD .EQ. IB(2))  GO TO 3
    2 CALL ERROR(IEA,2,WORD)
    3 WRITE (6,2005) WORD
      GO TO 101
    4 IF (MVAR * NOOB)  6,6,5
    5 NTR = NOOB * (NOOB + 1) / 2
      IF (NTR - NSYZ)  7,7,6
    6 CALL ERROR(IEA,6,IEB)
    7 CALL PAGE(NPAGE,IP,TITLE,7,IPA)
      WRITE (6,2001) MVAR, NOOB
      DO 8 I = 1,NOOB
    8 NO(I) = I
C          THREE WAY BRANCH FOR INPUT
      IF (IDS - 1)  9,25,11
    9 I1 = NSYZ - NOOB * MVAR + 1 - MVAR
      I2 = I1 + MVAR - 1
      CALL FORMAT(NFMT,IFMT,162,6,IPB,0)
      DO 10 I = 1,NOOB
      I1 = I1 + MVAR
      I2 = I2 + MVAR
   10 READ (5,IFMT) (A(J),J=I1,I2)
      GO TO 15
   11 DO 12 I = 1,MVAR
      II = I
      CALL FORMAT(NFMT,IFMT,162,6,IPE,II)
      K = NSYZ - NOOB * MVAR + I
   12 READ (5,IFMT) (A(L),L=K,NSYZ,MVAR)
      IF (MVAR - 1)  15,13,14
   13 PRINT 2004, MVAR
      GO TO 15
```

```
   14 PRINT 2003, MVAR
   15 K = 0
      I1 = NSYZ - NOOB * MVAR + 1
      DO 24 I = 1,NOOB
      J1 = I1
      DO 22 J = I,NOOB
      K = K + 1
      A(K) = 0.
      I2 = I1
      J2 = J1
      DO 20 L = 1,MVAR
      A(K) = A(K) + (A(I2) - A(J2)) **2
      I2 = I2 + 1
   20 J2 = J2 + 1
   22 J1 = J1 + MVAR
   24 I1 = I1 + MVAR
      GO TO 26
   25 CALL FORMAT(NFMT,IFMT,162,6,IP8,0)
      READ (5,IFMT) (A(I),I=1,NTR)
      PRINT 2002, NTR
   26 CONTINUE
C           END OF INPUT AND DISTANCE CALCULATIONS
      IF (NCLUS)  101,29,28
   28 CALL CNTGTY(NOOB,NCLUS,NR,NC)
   29 IF (IDSP .EQ. 0)  CALL TRI(A,ND,NOOB,NPAGE,IP,6,IPC,0,TITLE)
   60 CALL GRPING(NOOB,MVAR,KODE,NCLUS,NPAGE,TITLE)
      GO TO 1
  101 STOP
      END
      SUBROUTINE GRPING(NOOB,MVAR,KODE,NCLUS,NPAGE,TITLE)
      DIMENSION TITLE(8), IB(4), IA(2), IPC(6), IPD(4), IPF(6)
      DIMENSION A(80200), IS(400,2), NZ(799), NX(799), AIR(400)
      DIMENSION NY(400), NC(400), NO(400), WG(400), TR(400), PCT(400)
      COMMON A
      INTEGER BLANK,CATA, TITLE
      DATA IB/' ','+','I','-'/, IEC/'GRUP'/, IED/'KODE'/
      DATA IPC/'DIST','ANCE',' SQU','ARED',' MAT','RIX '/
      DATA IPD/'GROU','PING',' SUM','MARY'/
      DATA IPF/'STEP',' GRA','PH O','F GR','OUPI','NG  '/
 2000 FORMAT(23HINUMBER OF OBSERVATIONS I8,50X,39HSUBROUTINE GRPING
    1 DATED 21 JUNE 69 //)
 2001 FORMAT (I1,4HSTEP,I4,3X,5HSUM =,E15.6,13H   MIN DIST =,E15.6,
    1 11H  FOR ITEMS I5,5H  AND I5,6H  WITH 2I5,9H  MEMBERS)
 2006 FORMAT(1H0)
 2008 FORMAT(19H0     ITEMS GROUPED 5X,12I8)
 2009 FORMAT(1X,I4,2I5,E14.5,2H I,101A1)
 2010 FORMAT (1H ,29X,1H-,101A1)
 2012 FORMAT (1H ,25X,12I6)
 2014 FORMAT (1H ,29X,12I8)
 2015 FORMAT(25H STEP    I     J     VALUE   3X,12I8)
 5006 FORMAT (1H1,37HSTEP    I     J        %        WITHIN DSQ,7X,
    1    8HBETW DSQ/1H ,3X,1H0,16X,4H0.00,4X,11H0.00000E 00,E15.5)
 5007 FORMAT (1H ,I4,2I5,F10.2,2E15.5)
 5070 FORMAT (1H ,3X,1H0,13X,13H0.00000E 00 I,101A1)
      AVOG = 6.02257 * 10. ** 23
      NBB = NOOB - 1
```

```
      DO 502 I = 1,NOOB
      NZ(I) = 0
      NC(I) = I
  502 NO(I) = 1
      IC = 0
      TRACE = 0.
      IF (NOOB .GT. 16 .AND. KODE .EQ. 1)  IO = 1
      IF (KODE)  15,17,17
   15 CALL ERROR(IEC,15,IED)
   17 WRITE (6,2000) NOOB
      CALL PAGE(NPAGE,0,TITLE,4,IPD)
      WRITE (6,2006)
      DO 550 NCYC = 1,NBB
      AMIN = AVOG
      N = NOOB - NCYC + 1
      NB = N - 1
      NTR = N * (N + 1) / 2
C          SEARCH WITH DIAGONAL DELETION
      K = 0
      DO 514 I = 1,NB
      K = K + 1
      II = I + 1
      DO 514 J = II,N
      K = K + 1
      IF (A(K))  514,516,512
  512 IF (A(K) .GE. AMIN)  GO TO 514
      AMIN = A(K)
      MR = I
      MC = J
  514 CONTINUE
      GO TO 518
  516 MR = I
      MC = J
      AMIN = 0.
C          THE SEARCH HAVING BEEN COMPLETED WITH THE LOCATION OF AMIN, HC
C          KEEPING PROCEEDS
  518 IR = NC(MR)
      IC = NC(MC)
      IS(NCYC,1) = IR
      IS(NCYC,2) = IC
      WG(NCYC) = AMIN
      TRACE = TRACE + AMIN
      TR(NCYC) = TRACE
      WR = NO(IR)
      WC = NO(IC)
      PRINT 2001, IO, NCYC, TRACE, AMIN, IR, IC, NO(IR), NO(IC)
      IF (KODE .EQ. 1)  CALL TRI(A,NC,N,NPAGE,0,6,IPC,1,TITLE)
      IF (AMIN .EQ. 0.)  GO TO 707
C          COMBINE ELEMENTS OF THE CLOSEST PAIR
      MRR = MR - 1
      MPR = MR + 1
      MCC = MC + 1
      IF (MR .EQ. 1)  GO TO 703
      DO 702 I = 1,MRR
      II = (I - 1) * (2 * N - I) / 2 + MR
      JJ = (I - 1) * (2 * N - I) / 2 + MC
```

```
  702 CALL AA(NCLUS,II,JJ,WR,WC,AMIN)
  703 A(MR + MRR * N - MR * MRR / 2) = 0.
      DO 704 I = MRR,MC
      II = MRR * (2 * N - MR) / 2 + I
      JJ = (I - 1) * (2 * N - I) / 2 + MC
  704 CALL AA(NCLUS,II,JJ,WR,WC,AMIN)
      IF (MC .EQ. N)  GO TO 707
      DO 706 I = MCC,N
      II = MRR * (2 * N - MR) / 2 + I
      JJ = (MC - 1) * (2 * N - MC) / 2 + I
  706 CALL AA(NCLUS,II,JJ,WR,WC,AMIN)
  707 DO 708 I = MC,NP
  708 NC(I) = NC(I+1)
      NC(IR) = WR + WC
      NC(N) = 0
      NO(IC) = 0
C           CONDENSE THE VECTOR
  710 KK = MC - 1
      K = KK
      DO 714 I = 1,KK
      NN = N - I - 1
      IF (NN .EQ. 0)  GO TO 550
      DO 712 J = 1,NN
  712 A(J + K) = A(I + J + K)
  714 K = K + NN
      K = K + 1
      JJ = N * (N - 1) / 2
      DO 716 I = K,JJ
  716 A(I) = A(I + N)
  550 CONTINUE
C           PRINT THE GROUPING GRAPH
      NZ(1) = IS(NBB,1)
      NZ(2) = IS(NBB,2)
      DO 566 K = 2,NBB
      KK = NBB - K + 1
      DO 564 I = 1,K
      IF (IS(KK,1) .NE. NZ(I))  GO TO 564
      II = I + 1
      DO 562 J = II,K
      JJ = K - J + II
  562 NZ(JJ+1) = NZ(JJ)
      NZ(II) = IS(KK,2)
      GO TO 566
  564 CONTINUE
  566 CONTINUE
      JX = IB(1)
      JY = IB(4)
      I5 = 0
      LIM = NOUB / 49 + 1
      DO 587 L = 1,LIM
      I1 = I5 + 1
      I2 = I5 + 2
      I3 = I5 + 3
      I4 = I5 + 4
      II5 = I5 + 48
      I5 = MINO(II5,NOOB)
```

```
      N1 = 2 * I1 - 1
      N5 = 2 * I5
      IF (I5 .EQ. NOOP)  N5 = N5 - 1
      IF (I5 .EQ. NOOB)  JX = IB(3)
      NOB = 2 * NOOB - 1
      DO 572 I = 1,NOB
      NY(I) = 2 * I - 1
  572 NX(I) = IB(2)
      DO 574 I = 2,NOB,2
  574 NX(I) = IB(1)
      CALL PAGE(NPAGE, 1,TITLE,6,IPF)
      PRINT 2008, (NZ(I),I=I1,I5,4)
      PRINT 2012, (NZ(I),I=I2,I5,4)
      PRINT 2015, (NZ(I),I=I3,I5,4)
      PRINT 2014, (NZ(I),I=I4,I5,4)
      PRINT 5070, (NX(I),I=N1,N5), JX
      DO 586 K = 1,NBB
      DO 584 J = 1,2
      DO 582 I = 1,NOOB
      IF (NZ(I) .NE. IS(K,J))  GO TO 582
      IF (J .EQ. 1)  IX = I
      II = NY(I)
      NX(II) = IB(1)
      IA(J) = II
      GO TO 584
  582 CONTINUE
  584 CONTINUE
      II = (IA(1) + IA(2)) / 2
      NY(IX) = II
      NX(II) = IB(2)
      PRINT 2009, K, IS(K,1), IS(K,2), TR(K), (NX(I),I=N1,N5), JX
  586 CONTINUE
      PRINT 2010, (JY,I=N1,N5)
  587 CONTINUE
      DO 588 K = 1,NBB
      PCT(K) = TR(K) / TR(NBB) * 100.
  588 ATR(K) = TR(NBB) - TR(K)
      PRINT 5006, TR(NBB)
      PRINT 5007, (I, IS(I,1), IS(I,2), PCT(I), TR(I), ATR(I), I=1,NBB)
      RETURN
      END
      SUBROUTINE AA(NCLUS,I,J,WR,WC,D)
      DIMENSION A(80200)
      COMMON A
      WS = WR + WC
      WP = WR * WC
      IF (NCLUS - 1)  11,10,10
   10 IF (SIGN(1.,A(I)) + SIGN(1.,A(J)))  13,12,11
   11 A(I) = (WS * (WR * A(I) + WC * A(J)) - WP * D) / WS / WS
      RETURN
   12 A(I) = (WS * ABS(WR * A(I) - WC * A(J)) - WP * D) / WS / WS
      RETURN
   13 A(I) = (WS * (WR * A(I) + WC * A(J)) + WP * D) / WS / WS
   20 RETURN
      END
      SUBROUTINE CNTGTY(NOOB,NCLUS,NR,NC)
```

```
      DIMENSION A(80200), C(400), IC(400)
      COMMON A
 1001 FORMAT (80I1)
      K = 0
      NB = NOOB - 1
C          TWO WAY BRANCH FOR CONTIGUITY OPTIONS
      IF (NCLUS - 1)  51,11,21
   11 DO 14 I = 1,NB
      II = I + 1
      K = K + 1
      READ 1001, (IC(J),J=1,NOOB)
      DO 14 J = II,NOOB
      K = K + 1
      C(J) = 2 * IC(J) - 1
      IF (A(K))  51,12,13
   12 A(K) = SIGN(1.E-20,C(J))
      GO TO 14
   13 A(K) = SIGN(A(K),C(J))
   14 CONTINUE
      RETURN
   21 NN = NOOB - NC
      NM = NN + 1
      DO 27 I = 1,NN
      II = I + 1
      K = K + 1
      DO 22 J = I,NOOB
   22 IC(J) = 0
      IC(I + NC) = 1
      IF (MOD(I,NC))  23,24,23
   23 IC(I + 1) = 1
   24 DO 27 J = II,NOOB
      K = K + 1
      C(J) = 2 * IC(J) - 1
      IF (A(K))  51,25,26
   25 A(K) = SIGN(1.E-20,C(J))
      GO TO 27
   26 A(K) = SIGN(A(K),C(J))
   27 CONTINUE
      DO 31 I = NM,NB
      II = I + 1
      K = K + 1
      DO 28 J = I,NOOB
   28 IC(J) = 0
      IC(I + 1) = 1
      DO 31 J = II,NOOB
      K = K + 1
      C(J) = 2 * IC(J) - 1
      IF (A(K))  51,29,30
   29 A(K) = SIGN(1.E-20,C(J))
      GO TO 31
   30 A(K) = SIGN(A(K),C(J))
   31 CONTINUE
   51 RETURN
      END
      SUBROUTINE TRI(A,NC,NOOB,NPAGE,IP,IQ,IR,IS,TITLE)
      DIMENSION A(80200), NA(400), NB(400), NC(400), ND(400), B(400)
```

```
      DIMENSION TITLE(8), IR(10)
      INTEGER TITLE
 2002 FORMAT (1H0,8HOBS. NO.,I9,9I12)
 2004 FORMAT (1H ,I4,4X,10E12.4)
C         PRINTS A TRIANGLE OF A VALUES
      DO 9 I = 1,NOOB
      LL = NOOB - I + 1
      NB(LL) = NC(I)
    9 NA(LL) = I
      DO 10 I = 1,NOOB
   10 ND(I) = NB(I)
      NN = ((NOOB - 1) / 10) * 10 + 1
      DO 16 I = 1,NN,10
      LL = MIN0(I + 9,NOOB)
      MM = NOOB - I + 1
      DO 15 M = 1,MM
      IF (IS .EQ. 1)  GO TO 12
      IF (MOD(M - 1,50))  13,11,13
   11 CALL PAGE(NPAGE,1,TITLE,IO,IR)
      PRINT 2002, (NC(K),K=I,LL)
      GO TO 13
   12 IF (M .EQ. 1)  PRINT 2002, (ND(K),K=I,LL)
   13 LL = MIN0(NOOB-M+1,LL)
      L = NA(I) + ((2 * NOOB - M) * (M - 1)) / 2
      DO 14 K = I,LL
      B(K) = A(L)
   14 L = L - 1
   15 PRINT 2004, NC(M), (B(K),K=I,LL)
   16 CONTINUE
      RETURN
      END
      SUBROUTINE FORMAT(N,F,L,M,W,II)
      DIMENSION F(162), H(6), W(6), IB(2)
      INTEGER F, H, W, WORD
      DATA IB/'    ','FORM'/, IEF/' LIM'/
 1001 FORMAT (A4,2X,18A4)
 2001 FORMAT (1H0,6A4,31X,18A4/(46X,18A4))
 2002 FORMAT (1H0,6A4,I4,27X,18A4/(46X,18A4))
      IF (N)  11,11,1
    1 DO 2 I = 1,L
    2 F(I) = IB(1)
      I2 = 0
      IF (9 - N)  3,4,4
    3 CALL ERROR(IB(2),3,IEF)
    4 DO 6 J = 1,N
      I1 = I2 + 1
      I2 = I2 + 18
      READ 1001, WORD, (F(I),I=I1,I2)
      IF (WORD .EQ. IB(2))  GO TO 6
    5 CALL ERROR(IB(2),5,WORD)
    6 CONTINUE
      DO 7 I = 1,6
    7 H(I) = IB(1)
      DO 8 I = 1,M
    8 H(I) = W(I)
      K = 18 * N
```

```
      IF (II)  9,9,10
    9 PRINT 2001, (H(I),I=1,6), (F(I),I=1,K)
      GO TO 11
   10 PRINT 2002, (H(I),I=1,6), II, (F(I),I=1,K)
   11 RETURN
      END
      SUBROUTINE PAGE(NP,IP,II,NW,WD)
      DIMENSION HEAD(10), WD(10), II(8)
      INTEGER HEAD,II,WD
      DATA IB/'    '/
 2001 FORMAT (I1,8A4,15X,10A4,32X,4HPAGE,I4)
      DO 2 I = 1,10
    2 HEAD(I) = IB
      IF (NW)  5,5,3
    3 DO 4 I = 1,NW
    4 HEAD(I) = WD(I)
    5 PRINT 2001, IP, II, HEAD, NP
      NP = NP + 1
      RETURN
      END
      SUBROUTINE ERROR(WORD,NUMBER,WD)
      INTEGER WORD,WD
 2001 FORMAT (1H0,23HERROR STOP,    ROUTINE    ,A4,15H    AT STATEMENT,I6,
     1    18X,10HTROUBLE IS,10X,A4)
      WRITE (6,2001) WORD, NUMBER, WD
      CALL EXIT
      END
```

GEOGRAPHICAL MATRIX DISAGGREGATION

Purpose: To prepare data from pairs of geographical matrices for input to a multiple regression program.

Description: Given two geographical matrices $A = (a_{ij})$ and $B = (b_{ij})$ both n x m, the assumption is made that $B = f(A)$. This program disaggregates A and B so that b_{ij}, and $a_{i+p,\ j+q}$ are repunched onto the same card, ready to be inserted into a multiple regression program (e.g., BMD 02R) with b_{ij} as the dependent variable. This provides a quick procedure for empirically estimating two dimensional weighting functions. If $b_{ij} = 0 = a_{ij}$, the observation is skipped. The program reads data from unit, punches on 7, and writes on 6, read from unit 5.

Deck Make-up:
1) Control Card
2) Format Card
3) Title Card
4) A matrix
5) Title Card
6) B matrix

Card Format: 1) Control Card

columns	
1-2	Number of rows in the matrices
3-4	Number of columns in the matrices
5	Number of rows in the $\pm p$ neighborhood
6	Number of columns in the $\pm q$ neighborhood. The result is $(2p + 1)\cdot(2q + 1)$ independent variables.

2) Format Card

1-80	FORTRAN format for one row of the input matrices

3) Title Card

1-80	title of the A matrix

4) A matrix

Data Cards

5) Title Card

1-80	title of the B matrix

6) B matrix

Data Cards

Repeating of cards (5) and (6) causes the B matrix to replace the A matrix, and a new "B" matrix is read in, ad infinitum.

Reference: W.R. Tobler, "Geographical Filters and their Inverses," Geographical Analysis, Vol. I, No. 3 (1969), pp. 234-253; and the references therein. Also see program EXTRAP.

```
C PROGRAM PREPARING DATA FROM GEOGRAPHICAL MATRICES
C FOR INPUT TO A MULTIPLE REGRESSION PROGRAM
C TO ESTIMATE NEIGHBORHOOD WEIGHTING FUNCTIONS
C (EMPIRICAL TWO-DIMENSIONAL CONVOLUTION WEIGHTS)
C B(I,J) = F(   A(I+P,J+Q)   )
C    WHERE THE INPUT MATRICES A AND B ARE OF
C    SIZE NROWS BY NCOLS, AND THE MATRIX OF NEIGHBORS IS
C    NR BY NC.
C    READ FROM UNIT FIVE.
C READS: CONTROLS:FORMAT OF THE MATRICES: THEN
C A'S TITLE, A MATRIX, B'S TITLE, B MATRIX,
C DOES COMPUTATIONS AND PUNCHING, THEN
C REPLCES A BY B, READS C'S TITLE, AND LOADS B WITH C MATRIX, ETC.
C
C    WRITE ON UNIT SIX.
C    PUNCH ON UNIT SEVEN.
C    ORDER OF OUTPUT FROM UNIT SEVEN.
C      (1) THE INDEX NUMBER OF THE POINT.
C      (2) (NR*NC-1) NEIGHBORS FROM THE A MATRIX BEGINNING
C          AT THE POSITION CORRESPONDING TO (1,1) IN THE MATRIX
C          OF NEIGHBORS AND SKIPPING THE CENTRAL POINT.
C      (3) THE CENTRAL POINT FROM THE A MATRIX.
C      (4) THE CENTRAL POINT FROM THE B MATRIX.
C    OUTPUT FORMAT FROM UNIT SEVEN IS 19F4.0
C    COLUMNS 77-80 HAVE BEEN LEFT BLANK FOR PURPOSES OF NUMBERING.
C
      DIMENSION A(100,100),B(100,100),FMT(18), T(18),W(100)
C
      READ(5,1000) NROWS,NCOLS,NR,NC
      READ(5,1010) (FMT(I),I=1,18)
      READ(5,1010) (T(I),I=1,18)
      WRITE(6,1020) (FMT(I),I=1,18),(T(J),J=1,18)
      DO 10 I=1,NROWS
10    READ(5,FMT)(A(I,J),J=1,NCOLS)
20    READ(5,1010,END=100) (T(I),I=1,18)
      WRITE(6,1030) (T(I),I=1,18)
      DO 30 I=1,NROWS
30    READ(5,FMT) (B(I,J),J=1,NCOLS)
      MINR=NR/2+1
      MINC=NC/2 +1
      MAXR=NROWS-MINR+1
      MAXC=NCOLS-MINC+1
      K=0
      M=NR*NC
      N=M+1
      DO 80 I=MINR,MAXR
      DO 80 J=MINC,MAXC
      IF((A(I,J).LT..00001).AND. (B(I,J).LT..00001)) GO TO 80
      K=K+1
      KK=0
      DO 60 IP=1,NR
      IPI=I+IP-MINR
      DO 50 JQ=1,NC
      JQJ=J+JQ-MINC
      IF((IP .EQ.MINR).AND.(JQ .EQ.MINC))GO TO50
```

```
      KK=KK+1
      W(KK)=A(IPI,JQJ)
      IF(K.NE.1) GO TO 50
      WRITE(6,1040) I,IP,J,JQ
50    CONTINUE
60    CONTINUE
      IF(K.NE.1) GO TO 70
      IP=0
      JQ=0
      WRITE(6,1040) I,IP,J,JQ
70    W(M)=A(I,J)
      W(N)=B(I,J)
      FC=K
      WRITE(7,1050) FC,(W(L),L=1,N),I,J
80    CONTINUE
      WRITE(6,1060) K,M,N
      WRITE(6,1030)(T(J),J=1,18)
      DO 90 I=1,NROWS
      DO 90 J=1,NCOLS
90    A(I,J)=B(I,J)
      GO TO 20
100   CONTINUE
C
C    I/C FORMATS
C
1000  FORMAT(2I2,2I1)
1010  FORMAT(18A4)
1020  FORMAT('1MATRIX ESTIMATORS'//('0',18A4))
1030  FORMAT('0',18A4)
1040  FORMAT('0',4(I4,' '))
1050  FORMAT(19F4.0/19F4.0/13F4.0,2(' ',I2))
1060  FORMAT(/3(' ',I4)/'2NEW PAIR')
      END
```

UNIVARIATE GEOGRAPHICAL FORECASTING

Purpose: Extrapolation of geographical matrices in the time domain using a positionally invariant, time varying, linear, local operator.

Description: Data from an observation matrix, D, and matrices of weights are used to extrapolate the data with respect to time. Three options are available:

Option I: One weight matrix is used, and the results are time invariant.

$$Z_{i,j} = \sum_{p=1}^{K} \sum_{q=1}^{L} A_{p,q} D_{i+r,\ j+s}$$

where

$$r = \begin{cases} p - \dfrac{(k+1)}{2} & \text{if k is odd.} \\ p - \dfrac{(k+2)}{2} & \text{if k is even.} \end{cases}$$

and

$$s = \begin{cases} q - (l+1) & \text{if l is odd.} \\ q - \dfrac{(l+2)}{2} & \text{if l is even.} \end{cases}$$

Option II: Two weight matrices are used, and a time variant linear equation is employed.

$$Z_{i,j}^{t+\Delta t} = \sum_{p=1}^{K} \sum_{q=1}^{L} (A_{p,q} + B_{p,q}\Delta t) D_{i+r,\ j+s}$$

Option III: Three weight matrices are used, and a time variant quadratic equation is employed.

$$z_{i,j}^{t+\Delta t} = \sum_{p=1}^{K} \sum_{q=1}^{L} (A_{p,q} + B_{p,q}\Delta t + C_{p,q}(\Delta t)^2) D_{i+r,\ j+s}$$

If there are an even number of rows or columns in the weight matrices, an asymmetrical calculation will result. Calculation of the weight matrices may be performed using the DECOMP program. Predicted negative values are set to zero.

Deck makeup: 1) Card 1. FORTRAN format (2I3, 2I2, I1)

columns

1-3 The number of rows in the observation matrix. (<100)

4-6 The number of columns in the observation matrix. (<100)

7-8 The number of rows in the weight matrices. (≤15)

9-10 The number of columns in the weight matrices. (≤15)

11 NUM, an integer which denotes the option desired.

If NUM = 1, only the A weight matrix is read and option I is chosen.

If NUM = 2, the A and B matrices are read, and the linear option is chosen.

If NUM = 3, the A, B, and C matrices are read, and option III is chosen.

If NUM = 4, the A, B, and C matrices are read, and results are printed for both option II and and option III.

It is assumed that all of the weight matrices are of the same size. Neither the weight matrices nor the observation matrix need be square arrays.

2) Card 2. FORTRAN format (4F8.3)

columns

1-8 The earliest time for which the results are desired.

9-16 The time assigned to the observation matrix.

17-24 The final time for which the results are desired.

25-32 The time increment.

Any unit of time may be used, as long as all times are expressed in a common unit.

3) The weight matrices, FORTRAN format (5F8.6)

If NUM = 1, one weight matrix must be provided.

If NUM = 2, two matrices must be provided and the A matrix should precede the B matrix.

If NUM = 3, or NUM = 4, three weight matrices must be provided, A first, B second, and C third.

4) The observation matrix, FORTRAN format (20F4.0)

Output: 1) The weights used, five numbers per line.

2) The time and the corresponding results for that time, fifteen numbers per line.

Reading and printing are done on logical devices 5 and 6, respectively. Writing on tape or on files is accomplished through logical device 7.

Programmer: L. Trevillyan

References:

R.G. Brown, Smoothing, Forecasting and Prediction of Discrete Time Series, Prentice Hall, 1962.

W. Tobler, "Spectral Analysis of Spatial Series", Proceedings, Fourth Annual Conference on Urban Planning Information Systems, University of California, Berkeley, 1966, pp. 179-186.

W. Tobler, "A Computer Movie Simulating Urban Growth in the Detroit Region", Economic Geography, IGU Proceedings, June 1970.

```
C PROGRAM FOR THE TIME DOMAIN EXTRAPOLATION
C OF GEOGRAPHICAL MATRICES, Z(I,J) = F (  D(I,J), T  )
C WALDO R. TOBLER, GEOGRAPHY, U. OF MICHIGAN, 1969
C
C READS FROM 5, WRITES ON 6 AND 7
C NROWS, NCOLS = # OF ROWS, COLUMNS (<101)
C IN THE OBSERVATION MATRIX D AND THE PREDICTED MATRIX Z
C IROWS, ICOLS = # OF ROWS, COLUMNS (<16)
C IN THE WEIGHTING MATRICES A, B, C.
C THE EXTRAPOLATION EQUATION IS
C Z(I,J) = THE SUM OVER P AND Q OF W(P,Q)*D(I+P,J+Q)
C WITH W(P,Q) = A(P,Q) + B(P,Q)*T + C(P,Q)*T*T
C  INPUT:
C     CARD 1.  NROWS,NCOLS,IROWS,ICOLS,NUM, FORMAT2I3,2I2,I1
C     NUM DENOTES THE EQUATION TO BE USED.
C     CARD 2.  TO,TI,TF,DT , FORMAT 4F8.3
C     TO IS THE EARLIST TIME FOR WHICH EXTRAPOLATION IS DESIRED
C     TI IS THE TIME FOR WHICH DATA IS PROVIDED
C     TF IS THE LATEST TIME FOR WHICH EXTRAPOLATION IS DESIRED
C     DT IS THE TIME INCREMENT
C     (TIMES MAY BE GIVEN IN ANY CONSISTENT UNITS)
C
C     THE A WEIGHT MATRIX, FORMATTED 5F8.6
C     THE B WEIGHT MATRIX, 5F8.6
C     THE C WEIGHT MATRIX, FORMATTED 5F8.6
C     THE D MATRIX, FORMAT 20F4.0
C     A,B,C ARE WEIGHT MATRICES OF SIZE (IROWS,ICOLS)
C     D IS THE OBSERVATION MATRIX . SIZE NROWS X NCOLS
C
       DIMENSION Z(100,100),D(100,100),A(15,15),B(15,15),AZ(100,100),
     1 BZ(100,100),C(15,15),CZ(100,100)
1     READ(5,10,END=200) NROWS,NCOLS,IROWS,ICOLS,NUM
10     FORMAT(2I3,2I2,I1)
       READ(5,20) TO,TI,TF,DT
20     FORMAT(4F8.3)
       READ(5,40)  ((A(I,J),J=1,ICOLS),I=1,IROWS)
       WRITE(6,55)  ((A(I,J),J=1,ICOLS),I=1,IROWS)
       IF (NUM.EQ.1) GO TO 30
       READ(5,40)  ((B(I,J),J=1,ICOLS),I=1,IROWS)
       WRITE(6,56)  ((B(I,J),J=1,ICOLS),I=1,IROWS)
       IF (NUM.EQ.2) GO TO 30
       READ(5,40) (( C(I,J),J=1,ICOLS),I=1,IROWS)
       WRITE(6,57)  ((C(I,J),J=1,ICOLS),I=1,IROWS)
30     READ(5,50)  ((D(I,J),J=1,NCOLS),I=1,NROWS)
40     FORMAT(5F8.6)
50     FORMAT((20F4.0))
55     FORMAT('1 THE A WEIGHT MATRIX'  //5(' ',F8.6,3X))
56     FORMAT('0 THE B WEIGHT MATRIX'//5(' ',F8.6,3X))
57     FORMAT('0 THE C WEIGHT MATRIX'//5(' ',F8.6,3X))
       JROW=IROWS/2
       JCOL=ICOLS/2
       LROW=NROWS-2*JROW
       LCOL=NCOLS-2*JCOL
       DO 60 I=1,NROWS
       DO 60 J=1,NCOLS
```

```
      Z(I,J)=0.
60    AZ(I,J)=0.
      IF(NUM.EQ.1) GO TO 65
      DO 61 I=1,NROWS
      DO 61 J=1,NCOLS
61    BZ(I,J)=0.
      IF(NUM.EQ.2) GO TO 65
      DO 62 I=1,NROWS
      DO 62 J=1,NCOLS
62    CZ(I,J)=0.
65    CONTINUE
C
C     CALCULATION OF AZ,BZ,CZ
C
      DO 73 I=1,LROW
      IR=I+JROW
      DO 73 J=1,LCOL
      JR=J+JCOL
      DO 73 K=1,IROWS
      DO 73 L=1,ICOLS
      M=IR+K-JROW-1
      N=JR+L-JCOL-1
      GO TO (73,70,69),NUM
69    CZ(IR,JR)=CZ(IR,JR)+C(K,L)*D(M,N)
70    BZ(IR,JR)=BZ(IR,JR)+B(K,L)*D(M,N)
73    AZ(IR,JR)=AZ(IR,JR)+A(K,L)*D(M,N)
C
C    BEGIN EXTRAPOLATION
C
      IF(NUM.NE.1) GO TO 72
      DO 71 I=1,NROWS
      DO 71 J=1,NCOLS
      IF(AZ(I,J).LT.0. .OR.D(I,J).EQ.0.) AZ(I,J)=0.
71    Z(I,J)=AZ(I,J)
      GO TO 200
72    T=T0-TI
      DO 130 JKL=1,5001
      XM=T+(JKL-1)*DT
      TK=TI+XM
      IF(TK .GT. TF) GO TO 135
      DO 100 I=1,LROW
      IR=I+JROW
      DO 100 J=1,LCOL
      JR=J+JCOL
      IF (D(IR,JR)) 90,90,80
80    CONTINUE
      IF(NUM.EQ.3) GO TO 81
      Z(IR,JR)=AZ(IR,JR)+XM*BZ(IR,JR)
      GO TO 82
81    Z(IR,JR)=AZ(IR,JR)+XM*BZ(IR,JR)+XM*XM*CZ(IR,JR)
82    CONTINUE
      IF(Z(IR,JR)) 90,100,100
90    Z(IR,JR)=0.
100   CONTINUE
      WRITE(6,110) TK
110   FORMAT('1THE COMPUTED VALUES IN ',F8.3)
```

```
      WRITE(6,120) ((Z(I,J),J=1,NCOLS),I=1,NROWS)
120   FORMAT(15(' ',F4.0,3X))
      WRITE(7) NROWS,NCOLS,TK
      DO 125 I=1,NROWS
125   WRITE(7) (Z(I,J),J=1,NCOLS)
130   CONTINUE
135   CONTINUE
      IF (NUM .NE.4) GO TO 1
      NUM=3
      WRITE(6,140)
140   FORMAT('0 RECALCULATION USING THE QUADRATIC EQUATION')
      GO TO 72
200   CONTINUE
C
C     SET NROWS AND NCOLS EQUAL TO NEGATIVE NUMBERS AS A SWITCH FOR *PIB
C
      NROWS=-3
      NCOLS=-3
      WRITE(7) NROWS,NCOLS,TK
      END
```

BINOMIALLY WEIGHTED SMOOTHING

Purpose: The program removes high frequency spatial components from a matrix of geographical data by use of a nine point binomially weighted local smoothing operation.

Description: The local smoothing operator is passed over each A value (input) to produce a matrix of B values (output). This operation may be repeated many times. The fundamental equation is:

$$B_{ij} = \sum_{m=-1}^{1} \sum_{n=-1}^{1} A_{i+m\ j+n} W_{mn}$$

where W is a set of weights (3 x 3) and A_{ij} is the original data. Boundary effects are noted as edges and corners are reached. The weight fields are as follows:

in interior

0.06	0.125	0.06
0.125	0.25	0.125
0.06	0.125	0.06

at boundary

0.167	0.08
0.33	0.167
0.167	0.08

at corner

0.22	0.11
0.44	0.22

Although there is no mathematical limit to the number of times the observations may be smoothed, edge effects will creep inward towards the center of the matrix with each consecutive smoothing operation (i.e., observations smoothed 3 times, boundary effects felt three rows in). Currently the program will handle an input matrix of 48 x 48. This may be adjusted by changing the dimensioning of A and B to (ROWS + 2 by COLS + 2). In the current version of the program the results will be punched out after all smoothings. The observation deck is read by rows beginning in the NW (upper left).

Data Deck: 1) Control Card

2) Title Card

3) Variable Format Card

4) Observation Deck

Data Card Composition: 1) Control Card

columns

1-2	MB	Number of Rows
3-4	NB	Number of Columns
5-6	NO	Number of smoothings requested

2) Title Card

Any title ($\leq$ 72 characters)

3) Variable Format Card

Any E or F type FORTRAN type format for one row of data ($\leq$ 72 characters)

4) Observation Deck

Observations as described by the format card

Programmer: D. Rhynsburger and H. Moellering, after a more general MAD program by W.R. Tobler

References:
W.R. Tobler, "Numerical Map Generalization," MICMOG Paper No. 7, University Microfilms, OP - 33067.

W. Tobler, "Of Maps and Matrices", Journal of Regional Science, 7 (1967), pp. 275-280.

```
C BINOMIALLY WEIGHTED SMOOTHING PROGRAM
C READ=5  WRITE=6   PUNCH=7
C WRITTEN BY D. RHYNSBURGER, MODIFIED BY H. MOELLERING
      DIMENSION A(50,50), B(50,50), C(18), D(18)
   81 FORMAT (18A4)
   82 FORMAT (40I2)
  100 FORMAT(1X,'INPUT DATA')
  101 FORMAT('1OUTPUT SMOOTHED ',I2,' TIMES')
  102 FORMAT('0',13F10.4/' '13F10.4)
  104 FORMAT(8F10.2)
  105 FORMAT('1ROWS=',I5,'COLS=',I5,'WITH ',I5,' ITERATIONS')
      READ (5,82)  MB, NB, NO
      WRITE (6,105) MB,NB,NO
      READ (5,81) D
      WRITE (6,81) D
      READ(5,81) C
      MA = MB + 2
      MC = MB + 1
      MD = MB - 1
      NA = NB + 2
      NC = NB + 1
      ND = NB - 1
      DO 4 I = 2,MC
    4 READ (5,C)  (A(I,J),J=2,NC)
      WRITE(6,100)
      DO 3 I=2,MC
    3 WRITE (6,102)  (A(I,J),J=2,NC)
      DO 15 K = 1,NO
C          EVALUATION OF THE INTERIOR
      DO 11 I = 3,MB
      DO 11 J = 3,NB
   11 B(I,J) = (A(I-1,J-1) + A(I-1,J+1) + A(I+1,J+1) + A(I+1,J-1))/16.
     1    + (A(I,J-1) + A(I,J+1) + A(I-1,J) + A(I+1,J))/8. + A(I,J)/4.
C          EVALUATION OF THE HORIZONTAL EDGES
      DO 12 J = 3,NB
      B(2,J) = (A(2,J-1) + A(2,J+1) + A(3,J))/6. +
     1    (A(3,J-1) + A(3,J+1))/12. + A(2,J)/3.
   12 B(MC,J) = (A(MC,J-1) + A(MC,J+1) + A(MB,J))/6. +
     1    (A(MB,J-1) + A(MB,J+1))/12. + A(MC,J)/3.
C          EVALUATION OF THE VERTICAL EDGES
      DO 13 I = 3,MB
      B(I,2) = (A(I-1,2) + A(I+1,2) + A(I,3))/6.  +
     1    (A(I-1,3) + A(I+1,3))/12. + A(I,2)/3.
   13 B(I,NC) = (A(I-1,NC) + A(I+1,NC) + A(I,NB))/6.  +
     1    (A(I-1,NB) + A(I+1,NB))/12. + A(I,NC)/3.
C          EVALUATION OF THE CORNERS
      B(2,2) = (4. * A(2,2) + 2. * (A(2,3) + A(3,2)) + A(3,3)) / 9.
      B(2,NC) = (4. * A(2,NC) + 2. * (A(2,NB) + A(3,NC)) + A(3,NB)) / 9.
      B(MC,2) = (4. * A(MC,2) + 2. * (A(MC,3) + A(MB,2)) + A(MB,3)) / 9.
      B(MC,NC) = (4. * A(MC,NC) + 2. * (A(MB,NC) + A(MC,NB)) +
     1    A(MB,NB)) / 9.
      WRITE(6,101) K
      WRITE (6,81) D
      DO 15 I=2,MC
      DO 14 J = 2,NC
```

```
   14 A(I,J) = B(I,J)
   15 WRITE (6,102)  (B(I,J),J=2,NC)
      DO 5 I=2,MC
    5 WRITE(7,104) (B(I,J),J=2,NC)
      STOP
      END
 820 4
PINTHER'S BATHYTHERMAL DATA
(18F4.1/2F4.1)
08.308.608.409.107.007.308.407.107.508.207.206.603.704.102.304.806.810.1
11.613.1
09.409.009.209.811.212.612.110.810.411.111.009.207.703.904.704.408.213.4
15.115.1
12.011.611.913.114.214.713.813.313.413.213.111.510.310.808.808.210.215.9
15.717.2
13.715.115.814.814.915.616.516.616.413.915.613.514.714.216.113.211.814.2
10.711.7
15.416.415.917.615.717.217.917.717.916.817.114.316.814.613.914.216.416.3
17.916.8
18.217.117.917.416.916.719.318.018.416.814.616.417.417.717.115.409.416.9
15.013.7
19.818.717.818.217.518.419.018.418.617.917.818.118.718.418.917.817.318.3
17.818.4
18.718.218.018.017.118.318.218.018.418.117.917.918.018.217.617.617.818.2
18.218.1
```

Michigan Geographical Publications

University of Michigan
Department of Geography
Ann Arbor, Michigan 48104

Papers are issued several times a year. Priced variously; standing subscriptions granted twenty per cent reduction (payment must accompany single title orders).

1. Tobler, Waldo R., editor, Selected Computer Programs, 1970, 162 pp., $5.50. Second printing 1973.

2. Yuill, Robert S., General Model for Urban Growth: A Spatial Simulation, 1970, 221pp., $4.00.

3. Shannon, Gary W., Spatial Diffusion of an Innovative Health Care Plan, 1970, 166 pp., $4.00.

4. Tilmann, Sister Jean Paul, O.P., An Appraisal of the Geographical Works of Albertus Magnus, 1971, 190 pp., $4.00.

5. Deskins, Donald R., Jr., Residential Mobility of Negroes in Detroit, 1837-1965, 1972, 298 pp., $4.00.

6. Ma, Laurence J.C., Commercial Development and Urban Change in Sung China (960-1279), 1971, 196 pp., $4.00.

7. Jacoby, Louis R., Perception of Noise, Air, and Water Pollution in Detroit, 1972, 286 pp., $4.00.

8. Tobler, Waldo R., Lambert's Notes on Maps (1772), a translation with introduction of J.H. Lambert's "Notes and Comments on the Composition of Terrestrial and Celestial Maps," 1972, 125 pp., $4.00.

Manuscripts to be considered by the Department for publication may be submitted to Professor Thomas R. Detwyler or Professor John F. Kolars, editors.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

PROGRAMAS DE GEO-INFORMATICA

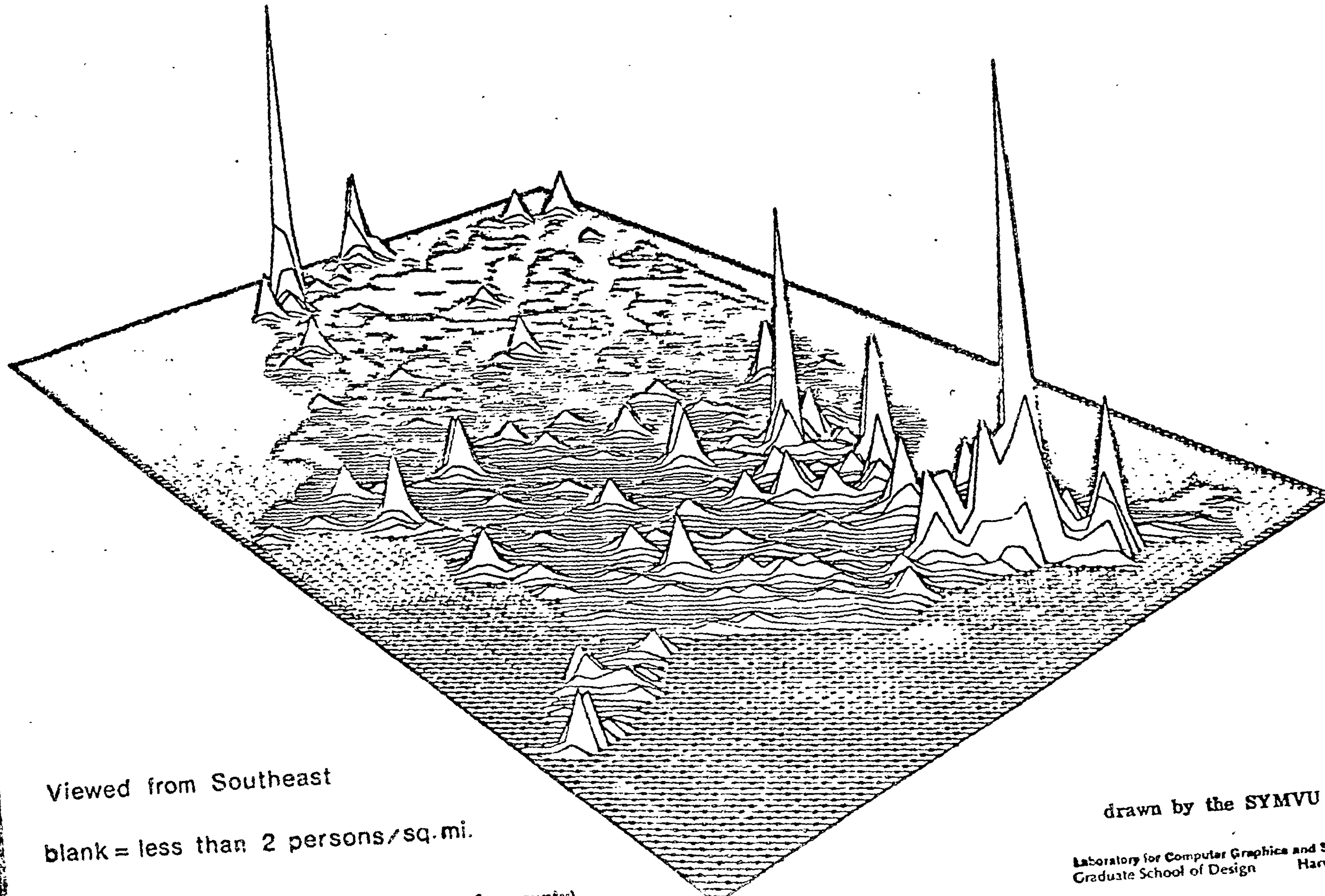
LABORATORIO DE COMPUTACION GRAFICA

Y ANALISIS ESPACIAL

UNIVERSIDAD DE HARVARD

JULIO, 1978.

UNITED STATES POPULATION DENSITIES, 1970



Viewed from Southeast

blank = less than 2 persons/sq. mi.

source: U.S. 1970 Census of Population (by county)

drawn by the SYMVU program

Laboratory for Computer Graphics and Spatial Analysis
Graduate School of Design Harvard University

CONTENTS

Brief History of the Laboratory

The Laboratory for Computer Graphics and Spatial Analysis was established within the Graduate School of Design at Harvard University in the Spring of 1965. In December of that year it received a major grant from the Ford Foundation.

The original and continuing goals of the Laboratory are:

1. To design and develop computer software for the analysis and graphic display of spatial data.
2. To distribute the resulting software to governmental agencies, educational organizations and interested professionals.
3. To conduct research concerning the definition and analysis of spatial structure and process.

The Laboratory was founded by Howard T. Fisher, who served as its Director until June 1968. During this time the SYMAP, SYMVU, and GRID programs were developed and made available to practicing professionals. In addition, several short courses and conferences were sponsored. Succeeding Fisher, William Warntz was Director of the Laboratory until June 1970. Under his direction, the Laboratory staff focused on research in spatial structure, and began to publish their findings in a newly initiated series of Theoretical Geography Papers.

In 1970 Allan Schmidt assumed responsibility for the Laboratory. Under his direction several additional software packages including CALFORM and POLYVRT were introduced. Extensive work was also begun on issues concerning cartographic data structures (POLYVRT and ODYSSEY) and interactive graphics (INPOM and ASPEX). In July 1976 Brian Berry became Director of the Laboratory. A substantial increase in emphasis on research relating to spatial process and geographic information systems is planned.

Research within the Laboratory covers a wide range of activities related to the analysis and display of spatially variable phenomena. Of particular interest are:

1. Theories and techniques applicable to the display of spatial data,
2. Theories intended to provide a better understanding of spatial structure and process, and
3. The distribution of resulting materials to members of planning and design professions as well as cartographers and geographers.

Display of Spatial Data (Computer Graphics)

An initial goal of the Laboratory was to develop low cost capabilities for producing computer maps. This is still an objective of the Laboratory, both in terms of updating earlier systems and developing new ones.

The original product, SYMAP, has continued and developed. Its system of display on a line printer led to a second generation of line plotter programs (SYMVU and CALFORM). Recently, a third generation (ASPEX and INPOM) has evolved towards an interactive cathode-ray tube environment. In the future, graphic software developed by the Laboratory will be increasingly interactive and intended for use with small to medium size computers.

Understanding Spatial Structure (Spatial Analysis)

Recent progress in the display of spatial data has depended increasingly on an understanding of spatial models and their impact upon the storage, manipulation and display of x-y coordinate cartographic data bases. These files contain the basic locational information required to produce a map. If a map is to represent a spatial concept, locational attributes of that concept must first be transmitted to the display program by means of a cartographic data base.

Initial efforts in cartographic data base research resulted in design and development of the POLYVRT program. This program has the ability to translate files from one data structure and format to another — preserving the inherent topology of each file yet eliminating errors and redundant detail.

Work on this project is currently being funded under a 3-year grant from the National Science Foundation. The title of the project is *Topological Information Systems for Urban and Environmental Research.*

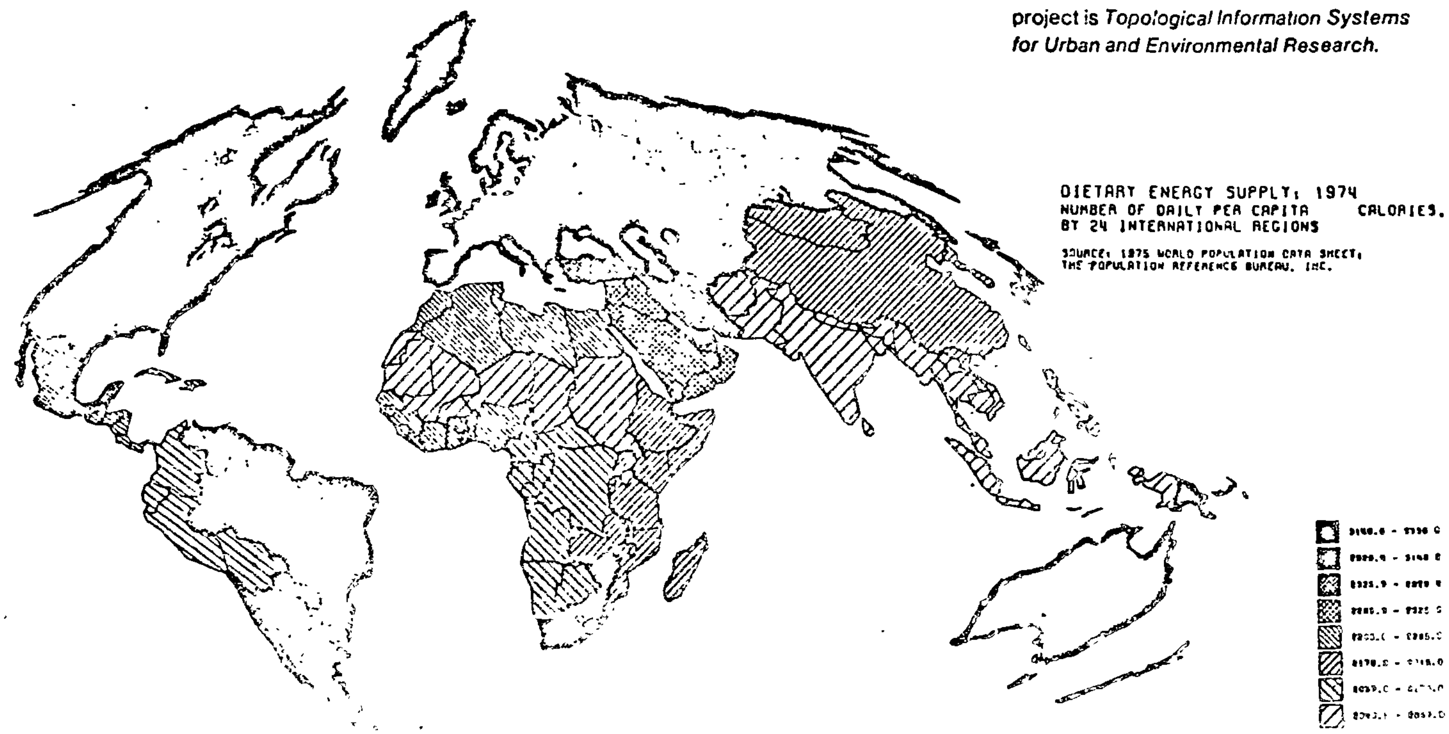
Recent Laboratory research on cartographic data structures has focused on topological analysis of spatial entities where basic relationships of connectedness provide the organization for a data structure. Such a data structure provides efficient data storage, flexible data retrieval and extensive error checking capability.

A topological data structure also allows for the creation of least common geographic units (LCGUs) which result from the superimposition of two or more partitionings of a region. As a result, direct overlay of arbitrary data zones in a region becomes feasible. This allows immediate comparison of different data bases without the complications of more traditional procedures for uniform gridding of a region. Current activities include utilization of this approach in the fields of land use and environmental planning and thematic mapping.

The Laboratory has acquired expertise in the manipulation of global cartographic files (WDB-I), U.S. Census DIME files for metropolitan areas and counties, U.S Census Urban Atlas files of census tracts, and a number of locally generated files. This research embodies both thematic mapping concerns of display research and other issues related to automated cartography and geographic information systems.

While each program can be characterized in terms of hardware requirements, these technological factors are less important than the conceptual model of spatial structure which influences its design. The design of a program is largely a function of how one describes and classifies measurable entities on the earth's surface. As a result, each program embodies a model of spatial structure which is reflected in the requirements and capabilities of that program.

The development and dissemination of cartographic data bases by the Laboratory is intended to facilitate the preparation of maps by users of the Laboratory's computer mapping programs.



Tools for the Profession

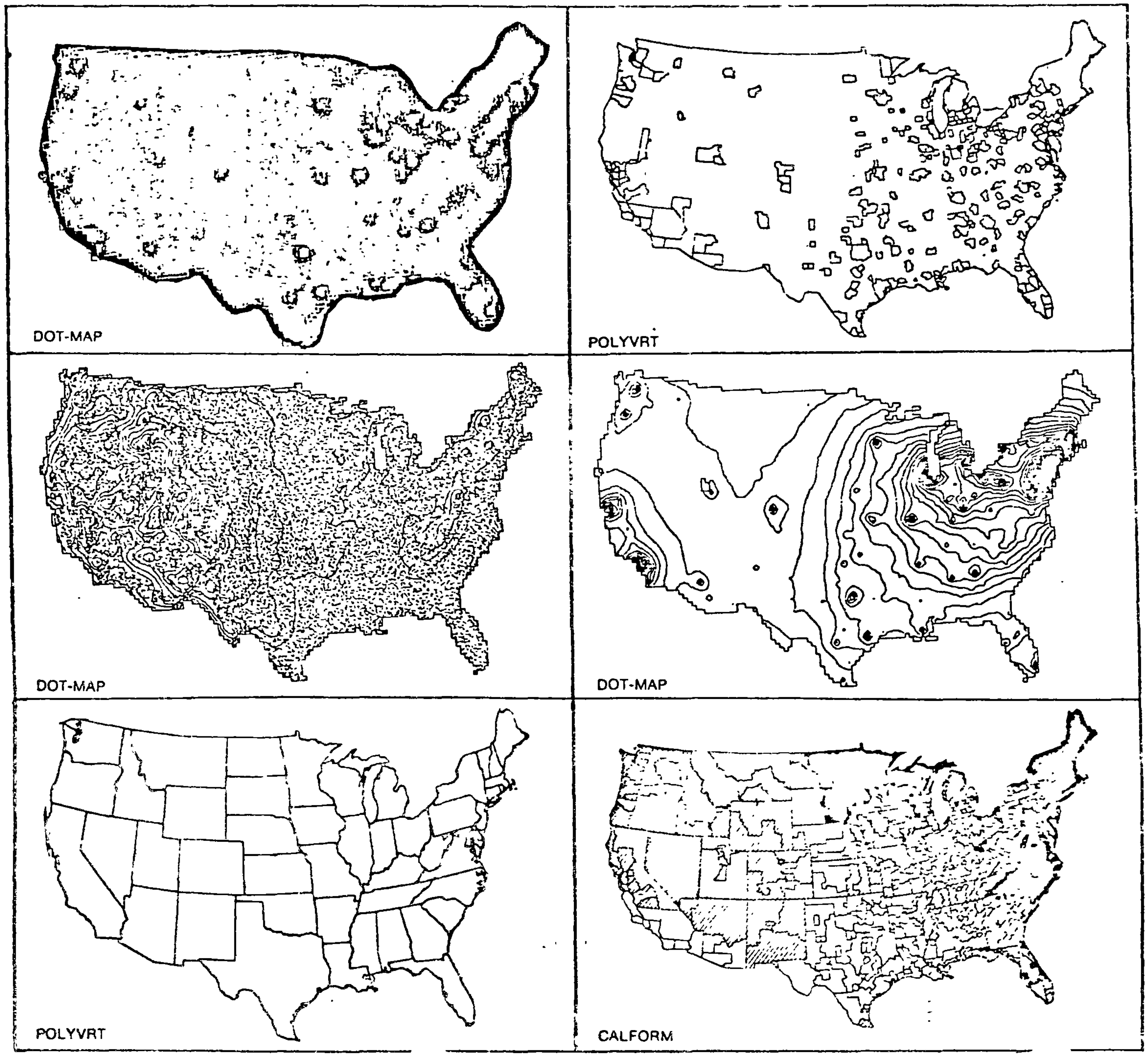
The Laboratory distributes the results of its research to academic, governmental, and commercial organizations interested in the application of spatial analytic techniques to planning and design. Available programs and publications are described in this catalog. Recent activities and developments are reported in the Laboratory's newsletter, CONTEXT, which is distributed free of charge. Summarys of research projects with appropriate illustrations are published in the Laboratory's "Red Book," a cumulative report of selected projects.

The Laboratory's applied research efforts focus upon selected projects that allow for the immediate application of new theories and techniques in a working environment. Such projects provide opportunities for evaluating and refining Laboratory products in realistic test situations. As a result, user requirements are directly reflected in the design of a spatial model as soon as possible. Feedback from our software user community also provides a significant contribution to our on-going program development efforts.

Though the Laboratory's programs are designed for use by individuals without prior training or experience in computer science, the software allows increasing sophistication and application by those with a computer science background or automated cartography experience. The programs are user oriented with considerable flexibility and numerous options within each program There is also a default procedure to satisfy most common requirements of a user's data base and also to minimize difficulty of obtaining an initial computer output for which sample test data is provided with the programs

A recent addition to the list of materials available from the Laboratory are a relatively complete set of Cartographic Data Bases for use with a conformant mapping program like SYMAP, CALFORM or INPOM Such data bases are currently available for a wide variety of geographic locations

Selected views of the United States



OPERATING POLICIES

The Laboratory's relationship with its user community has resulted in the establishment of operating policies concerning software distribution, support services, computer hardware, standards, research grants and contracts, the Harvard Mapping Service, and pricing.

Software Distribution

The Laboratory distributes a set of computer programs described later in this publication. In the future, new programs will be added to this list as a result of research efforts outlined above. Distribution of these programs is subject to the following agreement between the Laboratory and users of its software:

1. Neither this software, its documentation, nor adaptations thereof shall, except with prior written consent of the Laboratory, be sold, leased or otherwise distributed in any form to any individual, business entity, academic institution or governmental body whatsoever.
2. This software is not to be installed at a commercial computing installation (service bureau) in a manner which would allow for its use by individuals other than employees of the purchasing organization without prior written consent of the Laboratory.
3. Upon acceptance of these terms and conditions, as indicated by signature of an officer having authority to enter into such agreements, the Laboratory grants the recipient a royalty free, non-exclusive license to use the subject material at a single computer facility.

Support Services

With the sale of a program, the Laboratory undertakes to provide assistance and support for the program's installation and use. Through increasing adoption of software standards and assistance of software conversion centers (see below), installation effort is minimized.

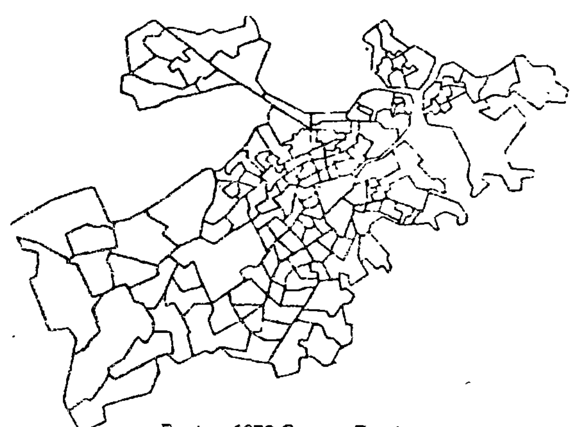
Questions concerning use of a program, however, will always exist. The utility of programs for a large user community depends on early identification and diagnosis of problems and a corresponding development of corrections and/or modifications. Such a process benefits the entire user community by eliminating errors and extending the capabilities of a total program package. Support provided free of charge by the laboratory for its programs consists of:

1. Responses to written or telephone communications concerning installation or operation of Laboratory software.
2. Modifications to the program, printed and distributed to all users. Provision for a complete program replacement tape is made at a charge equal to 10% of a program's current purchase price. This option is frequently desirable when modifications to an existing program involve several hundred changes within the original source code.

Computer Hardware

All of the Laboratory's programs are distributed for use on IBM hardware in FORTRAN source code. However, most of the Laboratory's programs also have been converted by certain other users to operate on a wide variety of non-IBM hardware. For most non-IBM hardware the Laboratory has available names and addresses of users who already have performed conversions and have agreed to make available a copy of their versions of the Laboratory's programs to authorized users.

The Laboratory is interested in establishing formal relationships with organizations willing to act as conversion centers for the Laboratory's software with respect to a specific hardware manufacturer. Two such conversion centers have recently been established, the University of Massachusetts for all CDC users and the University of Delaware for all Burroughs users. Each designated conversion center receives a free copy of all Laboratory software. These centers prepare copies of the Laboratory's software for their respective hardware and will be able to respond to user questions concerning its installation on these machines. As outlined above, however, the Laboratory retains sole right to control program distribution by these conversion centers.



Boston 1970 Census Tracts

Atlanta 1970 Census Tracts

Standards

The position of the Laboratory in the profession of automated cartography is unique in terms of standards. Decisions concerning software conventions affect a large number of users and have maximum utility only if applicable to a large segment of the user community.

Specific areas which the Laboratory has selected for establishing conventions for its software include:

1. Glossary of terms related to automated cartography
2. Standardized subset of transferrable FORTRAN and recommended procedures for localized groupings of non-standard features. For example, machine specific constructs to be avoided or input/output units to be assigned at a specific installation.
3. User command language for flexible and consistent naming of program control parameters.

Research Grants and Contracts

Support for the Laboratory's activities is derived solely from research grants and contracts plus program and publication sales. The Laboratory actively solicits support for its various research interests, described above. Organizations which have sponsored work of the Laboratory include:

1. The Ford Foundation
2. The Office of Naval Research
3. The National Science Foundation
4. The U.S. Departments of Housing and Urban Development, Health, Education and Welfare, Commerce, Interior, and Defense.

Research which is funded by outside organizations must satisfy criteria administered by Harvard University's Office for Research Contracts These criteria include the freedom to publish research findings subject only to established safeguards for protection of privacy or confidentiality of personal data.

Harvard Mapping Service

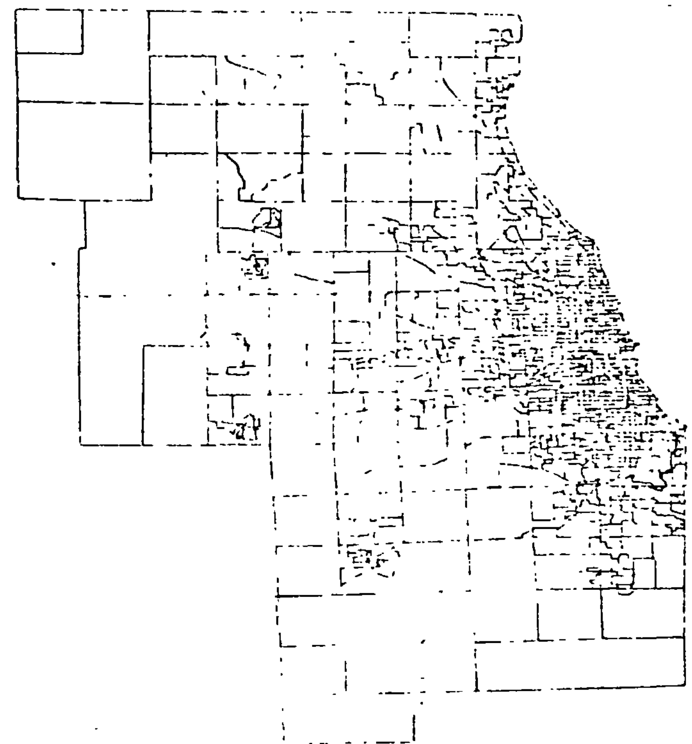
Occasionally the Laboratory has prepared maps, either by hand or by computer, to meet the needs of Harvard and MIT faculty and graduate students. When the diversity of experience thus gained can contribute to the general objectives of the Laboratory, mapping work may be undertaken for others as well Assignments of this kind have been carried out for such publications as *Life Magazine, Scientific American, The New York Times, National Geographic,* and for individual authors of various books and articles.

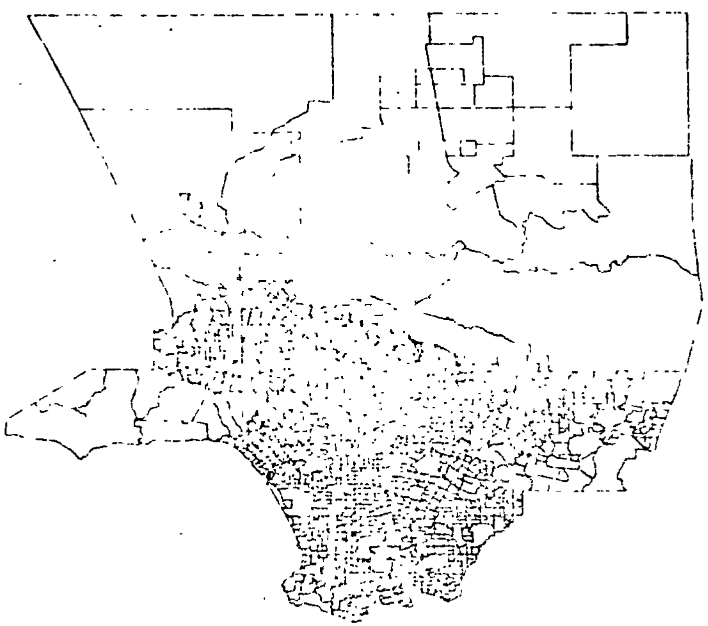
Interested organizations are encouraged to contact the Laboratory describing in as much detail as possible the nature of their needs and any applicable deadlines. Enclose if practicable, copies of data tabulations, base maps, or any other material that could aid in the understanding of what is required.

Pricing

Program products of the Laboratory are distributed at prices described in this catalog. Substantial discounts are available to governmental agencies and educational institutions. Prices are subject to change without notice. Purchase of a Laboratory program entitles a user to a copy of the current version of the program, sample data for test purposes, a user's reference manual, future updates and written instructions concerning the installation and use of the software. Software prices are established to cover program development, documentation, future enhancement, user assistance plus distribution.

Chicago 1970 Census Tracts →





← Los Angeles 1970 Census Tracts

PROGRAM DESCRIPTIONS

Program Name ..... **SYMAP, Version 5.20**

Computer Language .... FORTRAN IV
Computer Requirements
  Machine .... IBM 370
  Memory .... 128K bytes
  Mode .... Batch processing
Peripheral Devices
  Four temporary data sets
  Standard line printer

Description

SYMAP (SYnagraphic MAPping system) is a general purpose graphic display program for the mapping of spatially disposed quantitative and qualitative data. It is the most widely distributed program of its kind and is used by city planners, economists, geographers, and others.

SYMAP produces maps on a standard line printer. Varying shades of gray representing value class intervals are created by overprinting.

SYMAP creates five basic types of maps:

1. *Conformant* — Maps that display data by predefined zones, e.g., counties or census tracts. Input required consists of a set of x-y coordinates defining the outline of each zone and a data value for each zone.
2. *Contour* — Maps that graphically represent a continuous surface which has been computed by interpolation from data values at specified data points. The value computed for each point on the surface is influenced by the values at the nearest data points and the weights assigned to these points. These weights are based on $1/d^2$ where d is the distance from a data point to the point for which a value is being interpolated. The weights are modified as a function of the spatial distribution of the data points around the interpolation point. Input required consists of an x-y coordinate and a data value for each data point. Papers describing the contouring algorithm in greater detail are available from the Laboratory.
3. *Trend Surface* — Maps that graphically depict a polynomial surface of a specified order fitted to a set of data points and their data values. A surface is constructed by minimizing the sum of the squares of the differences between the known data values and the computed value of the surface at each data point. Surfaces from a first order ($z=ax+by+c$) to a sixth order ($z=ax^6+bx^5y+cx^4y^2+\cdots+iy^2+gx+hy+i$) polynomial can be calculated. Input required is the same as for contour maps.
4. *Residual* — Maps that describe a surface created by subtracting a trend surface from an interpolated contour surface where both surfaces were derived from the same set of data values. Input required is the same as for contour maps.
5. *Proximal (or Thiessen polygon)* — Maps that depict zones which are created from data points by the nearest neighbor method, i.e., the value of any point on the surface is the same as the data value of the closest data point. Input required is the same as for contour maps.

In addition to the required input, SYMAP includes options for creating map cosmetics (legends) such as: graphic scale, place names, rivers, bodies of water, transport routes, city locations, and other point, line or area symbols. There is also an option for specification of a study area outline which delineates a geographic boundary and displays interpolated data only within that outline.

In SYMAP, numerous electives provide control over virtually every visual aspect of the output map. For example, electives are used to specify the physical size of the map to be produced, coordinates of the display window (which allow close-ups or inset maps of sub-areas), the number, range, and symbolism of value class intervals, and other features.

Electives also are used to influence the computation of the map by controlling the interpolation or calculation methods that produce proximal, contour, trend surface, residual or conformant map output. Other electives modify the contour interpolation process by determining the number of data points which the program should search for within the vicinity of each interpolation point, the maximum distance to search for data points, and the amount of extrapolation to allow if the spatial distribution seems to warrant it.

SYMAP reads the x-y coordinate and value data in a standard fixed format, but it is possible to supply a subroutine FLEXIN (FLEXible INput) at program load time to allow for reading data in non-standard formats, extracting data from a data bank, or manipulation of data if desired.

SYMAP can also produce a binary file of the data zone outlines of conformant maps in matrix format. Each record consists of the zone number at each line printer character location for a given row on the output map. SYMAP can later read these files as input, allowing for the creation of inexpensive conformant maps by omitting processing required for the initial definition of zone boundaries.

In addition to line printer output maps, SYMAP can also produce binary files in a matrix format. The resulting files may be used as input to the Laboratory's SYMVU and ASPEX programs. These files have one record for each row of symbols on the line printer output map. Each record consists of the interpolated values at each line printer location (column).

Application Notes

The most common application of SYMAP involves the display of selected data from the U.S. Census of Population and Housing. Typical subjects mapped include median family income, population density, population change, and housing quality. More complex applications include use of the program for analytic, as well as descriptive purposes. For example, one may wish to investigate the nature of the relationship between a number of variables using statistical methods. The more highly related variables could then be mapped in an attempt to identify spatial trends or document a spatial process.

SYMAP has been used in a wide variety of applications including:

1. *Market Research* — To delineate patterns of ethnicity and socio-economic character in determining the most cost-effective approach for product testing and potential market penetration.
2. *City Planning* — To highlight urban social and economic problems.
3. *Coronary Care* — To identify inadequate hospital coronary care facilities and to determine if hospitals with coronary

Program Name .... GRID, Version 3.0

Computer Language .... FORTRAN IV
Computer Requirements
Machine .... IBM 370
Memory .... 150K bytes
Mode .... Batch processing
Peripheral Devices .... Standard line printer

Description

GRID is a special purpose program designed to display data values which have been collected on the basis of gridded data collection zones. Since it was designed specifically for use with gridded data bases, the program is able to display such data more efficiently than would be possible with a general purpose program like SYMAP. The program is frequently used to map natural resource data derived from aerial photos using a grid overlay to define data collection zones.

Like SYMAP, various options may be used to control the scale, symbolism, and value range. Other options allow the user to define grid cell size and shape, prescale the data, obtain grid numbering or perform dot mapping. Users prepare their own data input subroutine (FLEXIN) for selecting specific subject items from a data bank. FLEXIN may also be used to define a mathematical expression involving two or more variables. The resulting values are then displayed by use of shading symbolism. The result is a graphic matrix composed of one or more print symbols for each cell of the data grid. The density of a print symbol indicates the data value or category assigned to a given cell.

Although GRID normally is used with data based upon rectangular grids, there is also a method for specifying irregular outlines. In the program as distributed the data grid is restricted to 10,000 cells but minor modifications to a DIMENSION statement can greatly increase the program's internal storage capability. Larger data matrices also may be processed as a series of adjacent panels which the program can produce automatically.

Application Notes

The GRID program is commonly used as

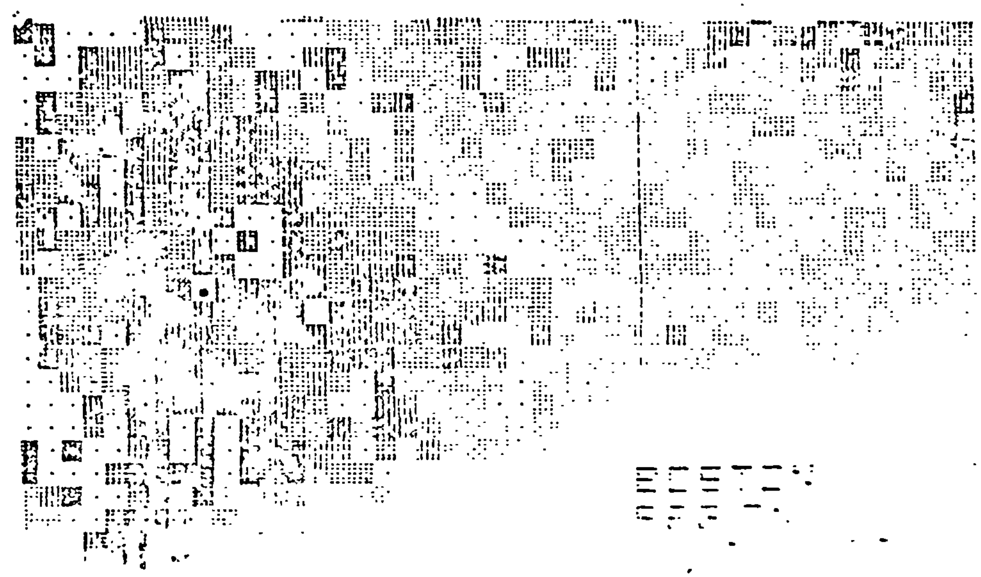
necessary to record data for a region without reference to pre-existing data zones or other boundaries. Examples of such data include land use, soil type, ground water, vegetation type, zoning constraints and topography. Since features of this type rarely exist neatly within predefined geographic zones, such as census tracts, it may be necessary to establish a geographic data collection unit such as a grid over a study area and then record one or more data values for each grid cell. Data obtained by remote sensing techniques such as LANDSAT satellite imagery is recorded directly in a gridded (raster) format.

In addition to being able to display data from a single matrix, it is also possible to read data from two or more matrices using subroutine FLEXIN and perform user defined arithmetic or logical operations upon the values within each grid cell. The resulting values may then be displayed as a new matrix of values. This technique is particularly helpful when evaluating a variety of alternative combinations or weightings of data matrices for several different subjects. Such an operation has been used to identify those locations (grid cells) having a maximum (or minimum) attractiveness for future development, conservation or other use. GRID also allows for the evaluation of alternative sites by different analysts. Each of their professional preferences may be expressed in terms of selected subjects and their weightings. The resulting maps may then be interpreted as alternative solutions which reflect the judgment or preferences of each analyst in response to a given set of objectives.

Materials Available

1. GRID Version 3 Computer Mapping Program
2. GRID User's Reference Manual.

Copies of the FORTRAN-IV source program for an IBM 370 computer are available on new, unlabeled, 200 foot reels of 9 track tape written at 800 BPI.



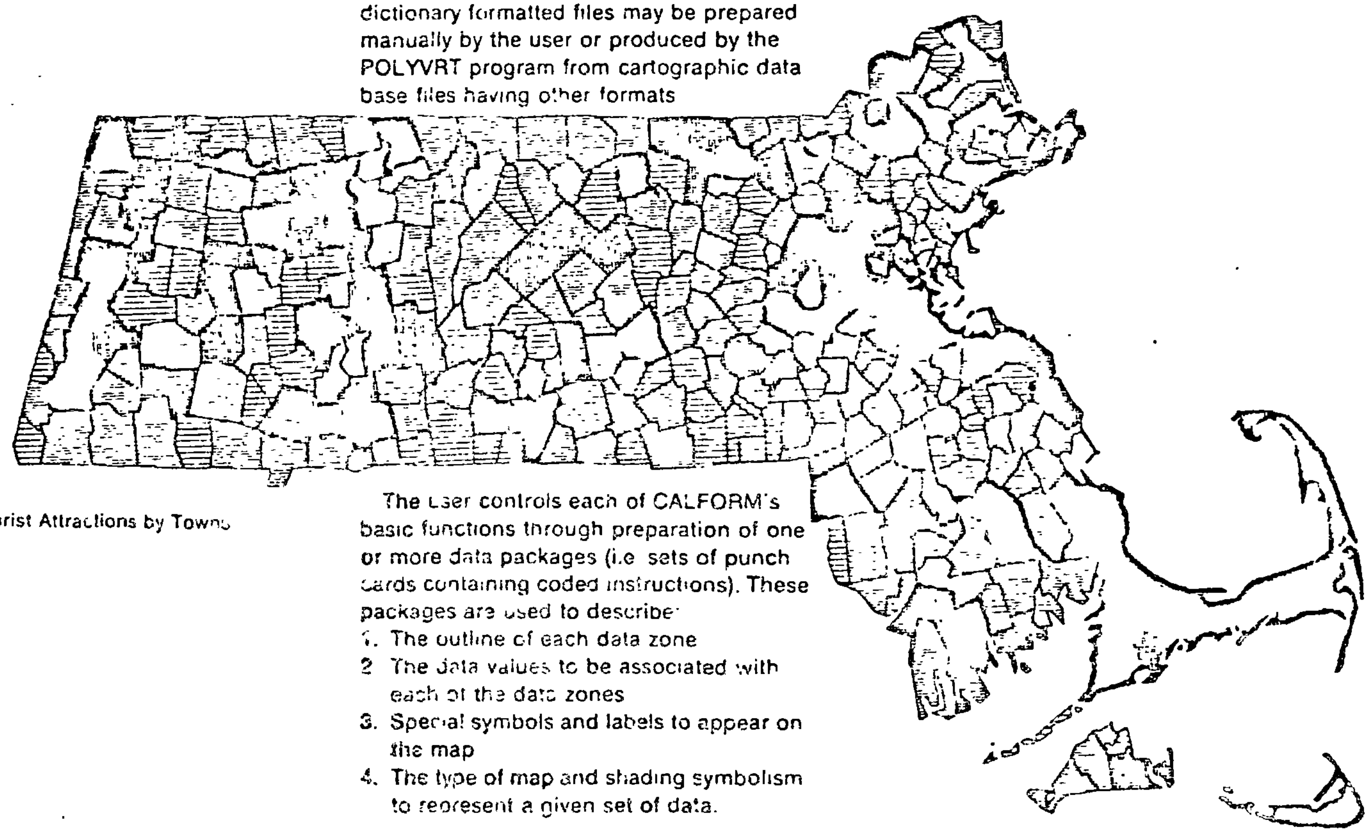
Program Name . . . CALFORM, Version 1.2

Computer Language . . . . FORTRAN IV
Computer Requirements
Machine . . . . .IBM 370
Memory .150K bytes
Mode . . . . Batch processing
Peripheral Devices
Three temporary data sets
Pen plotter or CRT plotter

Description

CALFORM (CALcomp FORMs) is a computer program which uses a line plotter to produce a shaded map based upon data zones. The program is typically used to display data by administrative zones which have been defined as irregularly-shaped polygons. The resulting conformant maps graphically depict data values for data zones such as census tracts, municipalities and counties. This program may be used to produce graphics of high quality, suitable for publication.

CALFORM produces two basic types of conformant maps, outline maps and shaded thematic maps. A map may contain as many as 500 data zones, each of which can be described as a polygon composed of up to 600 line segments. Required input data is in a point dictionary format which ensures that each point is uniquely defined thereby eliminating graphic "sliver errors." point dictionary formatted files may be prepared manually by the user or produced by the POLYVRT program from cartographic data base files having other formats.



Mass. Tourist Attractions by Towns

The user controls each of CALFORM's basic functions through preparation of one or more data packages (i.e. sets of punch cards containing coded instructions). These packages are used to describe:

1. The outline of each data zone
2. The data values to be associated with each of the data zones
3. Special symbols and labels to appear on the map
4. The type of map and shading symbolism to represent a given set of data.

Once the user has defined an initial map size and type, data zone outlines, the number and range of data categories and the shading symbolism associated with these categories, numerous maps can be produced by providing a new set of data values and/or output specifications for each additional map.

Application Notes

CALFORM is used to produce conformant maps of a study area which is divided into a number of data zones. Each zone may have associated with it a value to be mapped. By grouping the values into a number of value class intervals and selecting a line or character symbol for each interval, one or more maps may then be produced depicting the spatial pattern of the data.

This program is designed to be used on a device having high resolution such as a line plotter. It has a number of features to allow legends, keys and titles to be included in the map. Hence, it can be used for maps of publication quality.

Through the use of the FLEXIN option data values for each zone can be manipulated prior to their display. For example, FLEXIN may be used to consolidate many small zones to a number of larger zones. The values for all newly aggregated zones could be derived by weighting the contribution of each constituent zone as a function of its contribution to the new zone aggregate value.

For identifying zones which are shaded according to their data value, a smaller inset outline map of each zone with its name can be included and positioned in an appropriate part of the larger map.

Although the program's algorithm for creating data value intervals is designed for data measured on an interval or ratio scale, data which has an inherently ordinal or even nominal measurement scale may be mapped as well. For example, symbols could be chosen to represent several different types of agricultural production. The counties of a state could then be mapped by the predominant type of agriculture in each county.

Other applications of CALFORM involve mapping of data values related to point or line locations rather than areal zones. By coding the points as compact areas (e.g., squares) of small size, or lines as linear strips of limited width, shading may be employed to illustrate values pertaining to the points or lines.

Materials Available

1. CALFORM Mapping Program
2. CALFORM User's Reference Manual

Copies of the FORTRAN-IV source program for an IBM 370 computer are available on new, unlabeled, 200 foot reels of 9 track tape written at 800 BPI. A sample set of input data is furnished as a second file

Program Name . SYMVU, Version 1.2

Computer Language .... FORTRAN IV
Computer Requirements
Machine .... IBM 370
Memory .... 220K bytes
Mode .. .. Batch processing
Peripheral Devices
Four temporary data sets including plot tape
Pen plotter

Description

The SYMVU computer mapping program uses a line plotter to represent gridded spatial data as a three-dimensional surface. Surface locations hidden from view are not drawn. SYMVU representations have been used to portray such variables as topography, income, population, air pollution and bathorhythms. Users have a great deal of flexibility in specifying how the surface is to be drawn including: rotation, tilt, elevation, vertical scale, base and background symbols as well as locational symbols, titles and text.

SYMVU accepts a matrix of data values containing up to 130 × 130 elements. The data may be generated in either of two ways. If the data values are irregularly spaced, SYMAP can interpolate between the data points to produce a contour map and at the same time generate a matrix formatted tape file specifically designed to be processed by SYMVU. If users furnish their own input subroutine to SYMVU (Subroutine DATA) SYMVU may be used to read a user provided matrix of data values in a non-standard input format or manipulate values according to user specifications. SYMVU contains several user aids such as the automatic computation of minimum and maximum data values. It also has a data smoothing routine to reduce minor fluctuations and diminish extreme variations within the data.

Application Notes

Although SYMVU can represent any data provided in a matrix format (up to its dimensional capacity of 130 × 130 data units), it is most effective for displaying data which can be represented as a continuous surface. Such surfaces are commonly represented on a standard two-dimensional map as a series of contours. When viewed as a SYMVU perspective drawing, a three dimensional surface yields substantially greater detail and visual impact.

In addition to displaying values which represent a continuous surface, SYMVU may also be used to depict a matrix of values representing a discontinuous data surface. Such surfaces correspond to conformant map data for data zones such as census tracts, towns, states or countries. In such cases the data zones appear as raised plateaus whose elevation corresponds to the data value of each zone. The outline of each plateau retains the general shape of the data zone. SYMAP may be used to generate a matr formatted tape file of a conformant map for input to SYMVU.

SYMVU surfaces have been effectively used to present data relating to such subjects as landforms (terrain elevation), population densities, air pollution concentrations, income distribution and related social statistics. Other applications of SYMVU have included:

1. *Animated Time-Series Films* — Spatially distributed data showing changes in population growth or air quality.
2. *Stereograms* — Images achieved by plotting two views of the same surface and changing the viewing azimuth by a few degrees.
3. *Cartograms* — Representations of geographic regions depicted as volumetric prisms. Height is proportional to one variable (e.g., population size) and the area of each data zone is defined proportional to a second variable (e.g., per capita income). The volume becomes proportional to the product of the two variables (total income, in this case).
4. *Two-Color Overlays* — Combination data display in which one surface, such as topographic relief, is plotted with reference to a related data grid indicating forested areas. With slight program modification, the lines over the forested portion of the grid are drawn as a separate plot. The two plots are then registered and printed together in two colors using standard printing techniques.

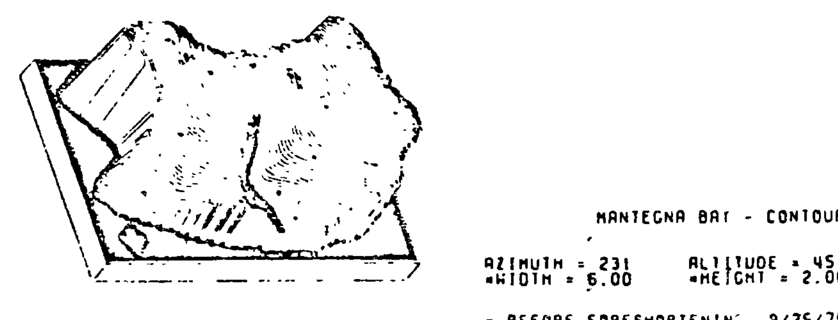
Materials Available

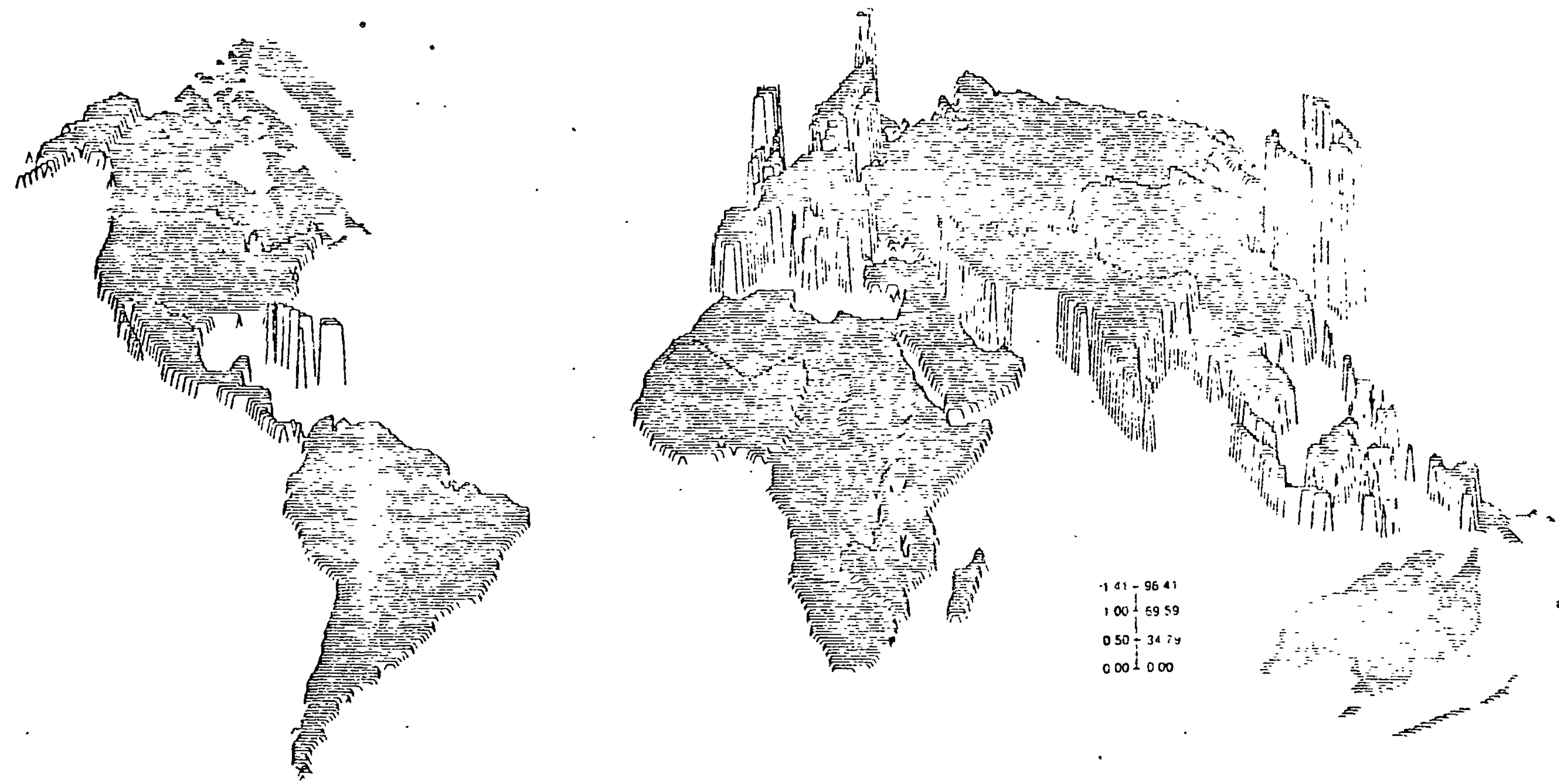
1. SYMVU Program with sample data
2. SYMVU User's Reference Manual

Copies of the FORTRAN-IV source program for an IBM 370 computer are available on new, unlabeled, 200 foot reels of 9 track tape written at 800 BPI.

← SYMAP Contour Map

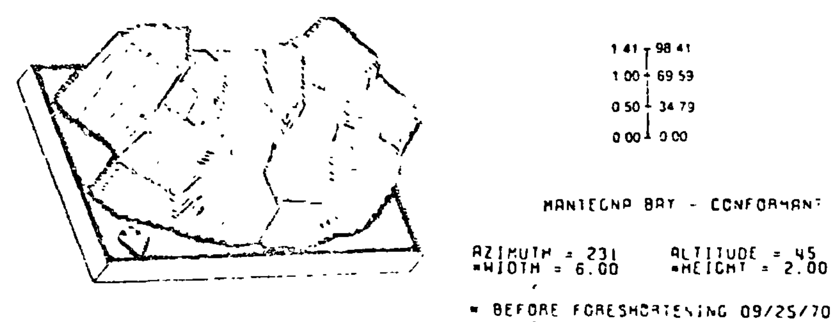
SYMVU Contour Map →





← SYMAP Conformant Map

SYMVU Conformant Map →



Program Name .....POLYVRT, Version 1.1

Computer Language ....FORTRAN IV
Computer Requirements
Machine ....IBM 370
Memory ....160K bytes
Mode ....Batch
Peripheral Devices
Two temporary data sets
Pen plotter

Description

POLYVRT is a general purpose utility program designed for use with polygon oriented cartographic data bases (CDB's). Coordinate data describing polygons may be input and output in a variety of data structures and formats. Such files are normally required for use with conformant mapping programs.

POLYVRT's capabilities include:

1. *Conversion of a CDB File* — Translation from one data structure or format to another.
2. *Internal Topological Data Structure* — Automatic detection of errors in polygon definition
3. *Retrieval of Selected Polygons* — Retrieval from a CDB file using FORTRAN-like logical expressions.
4. *Update and Correction of Coordinates* — Revision of coordinates and topological attributes of chains and polygons.
5. *Map Projections* — Creation of specific projections, rotations and scalings.
6. *Assignment of Levels of Line Detail* — Representation based on an automated line generalization technique.
7. *Simple Line Plotting* — Visual verification of a file.

POLYVRT relies on its topological data structure — the chain file — to provide the capabilities outlined above. A "chain" consists of a series of points which form a boundary between the polygons on its left and right. The first and last points of a chain are referred to as nodes. Together these three elements (nodes, points, and left/right polygon identifiers) define the network of a cartographic data base. The chain file concept has a generality and efficiency that offers many advantages for the creation and maintenance of cartographic data bases

Any of the following file structures may be accepted as input and converted to an internal chain file data structure. In addition, the following file structures may be output from the internal chain file of POLYVRT.

1. *DIME (Dual Independent Map Encoding)* Files — CDB's where each line segment is identified by its nodes, (end points) and the polygon names on each side of the line. DIME files which describe the census blocks and tracts for most U.S metropolitan areas plus a file of county outlines are available from the Census Bureau. A county outline file for the U.S. in DIME format is provided with the POLYVRT program.
2. *World Data Bank I* — A file containing 120,000 points which describe the outlines of the countries of the world. The Laboratory has enhanced the basic WDB-I file available from the U.S. National Technical Information Service by adding the topological information required by a POLYVRT chain file This modified WDB-I chain file is provided along with the POLYVRT program.
3. *CALFORM File* — A cartographic data base structure based on a common point dictionary with which polygons are described. The CALFORM program for conformant mapping is distributed separately by the Laboratory.
4. *SYMAP File* — A very simple file in which each polygon is individually defined as a string of coordinates is available as output from POLYVRT. At a future date POLYVRT will be modified to also accept a SYMAP file as input.
5. *LUDA (Land Use Data and Analysis)* — Files generated by the U.S. Geological Survey. These files share the 'chain' concept with POLYVRT although they use the term 'arc.' Conversion of these files will be a new feature of POLYVRT once formats for dissemination of LUDA files have been established.
6. Other file structures will be handled in response to user requests and included in future updates of the program.

POLYVRT provides capabilities to perform a variety of operations upon a cartographic data base by use of optional input packages. A specific subset of an input CDB may be requested through English-like IF statements. Topological error detection may be used to identify and also to correct certain file errors. Correction of errors and additions a file also may be effected by use of the program's comprehensive update package. Chains, parts of chains and their topological characteristics may be added, deleted or replaced. A coordinate transformation package including ten standard map projections (with provision for a user-supplied routine to define non-standard projection types) is also provided.

The program includes an efficient recursive algorithm that measures the detail level (geometric significance) of each point in each chain By specifying a set of criterion bands, the user may create a line detail measure for each point and this information can be stored with the file. Files may be output with any degree of line generalization making it possible to produce the smallest CDB compatible with the purpose for which it will be used. Issues relevant to the selection of an appropriate level of line generalization include map size and scale plus the line drawing precision of output devices such as printers, CRT's and line plotters.

Output may be displayed on a line plotter. A chain structured output file can be produced for input to INPOM and CELLMAP. The topological chain will be the basic data structure for much of the Laboratory's future mapping software.

The distribution tape includes source code, the county DIME file, and the Laboratory revision of WDB-1. *The POLYVRT Users Manual* includes program documentation along with control card examples and instructions. IBM versions include a separate file containing the Job Control Language to compile, overlay, and execute the program from the tape provided.

Application Notes

Before using a conformant mapping program (such as SYMAP, CALFORM, INPOM, CELLMAP), a user must create a cartographic data base file which describes the data collection zones. Generally, this description consists of x-y coordinate outlines of polygons, but there are significant differences in the record and file format requirements for various mapping programs. Creating a new CDB involves the expenditure of significant amounts of tim[illegible] [illegible]d energy.

One goal of POLYVRT is to provide a universal, archival description of data zones based upon a topological data structure. POLYVRT can then use this data structure to generate any one of the record and file formats required by a specific mapping program. As a result, POLYVRT provides a great deal of flexibility to the individual who wishes to create only one CDB which can be used to produce input to a variety of different mapping programs.

A major obstacle to the widespread use of computer mapping software is the initial cost of preparing a CDB for a given study area. This process can be avoided in many applications by using existing, publicly available files. However, the conflicting formats and structures of these files reflect the lack of standards in the design of computer mapping systems. Without POLYVRT a user is required to write a different program to extract useful information from a variety of sources. The difficulties often outweigh the benefits.

The growing library of available CDB's describe geographic units from the city block level in Metropolitan GBF-DIME files to the country level in WDB-I. With POLYVRT, a user can generate a CDB specifically tailored from any one of these sources. As a result, the effort required to produce a CDB and related thematic maps of an area can be significantly reduced.

When an available file does not provide the information required, POLYVRT can aid in the process of constructing a CDB. The chain file format offers the following advantages in the digitizing process:

- Allows use of stream input modes
- Avoids ambiguity and sliver errors
- Reduces the number of keystrokes required to identify polygons.

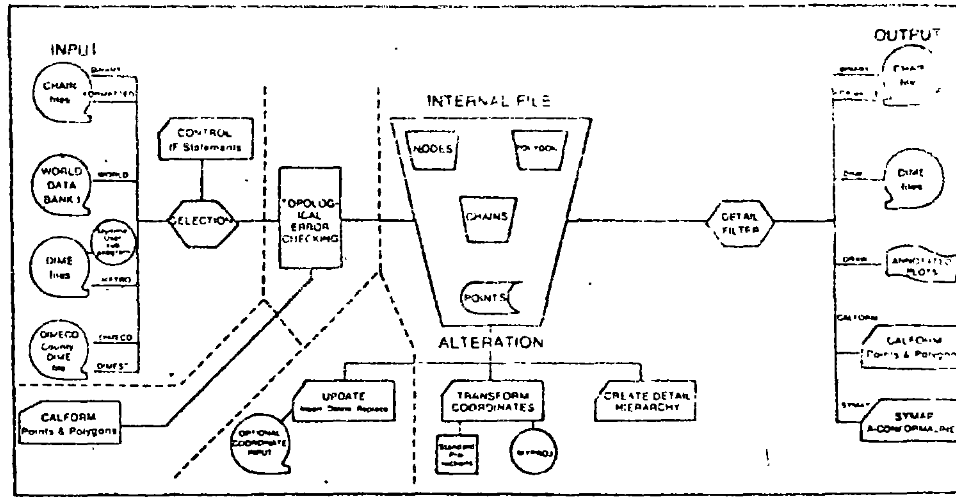
Furthermore, POLYVRT's topological error checking capability serves an important function in detecting errors. The program's updating facility allows corrections to be made without redigitizing. By creating a CDB through POLYVRT, the quality of the information is likely to be quite high and the user acquires a great deal of flexibility in the subsequent uses of the file.

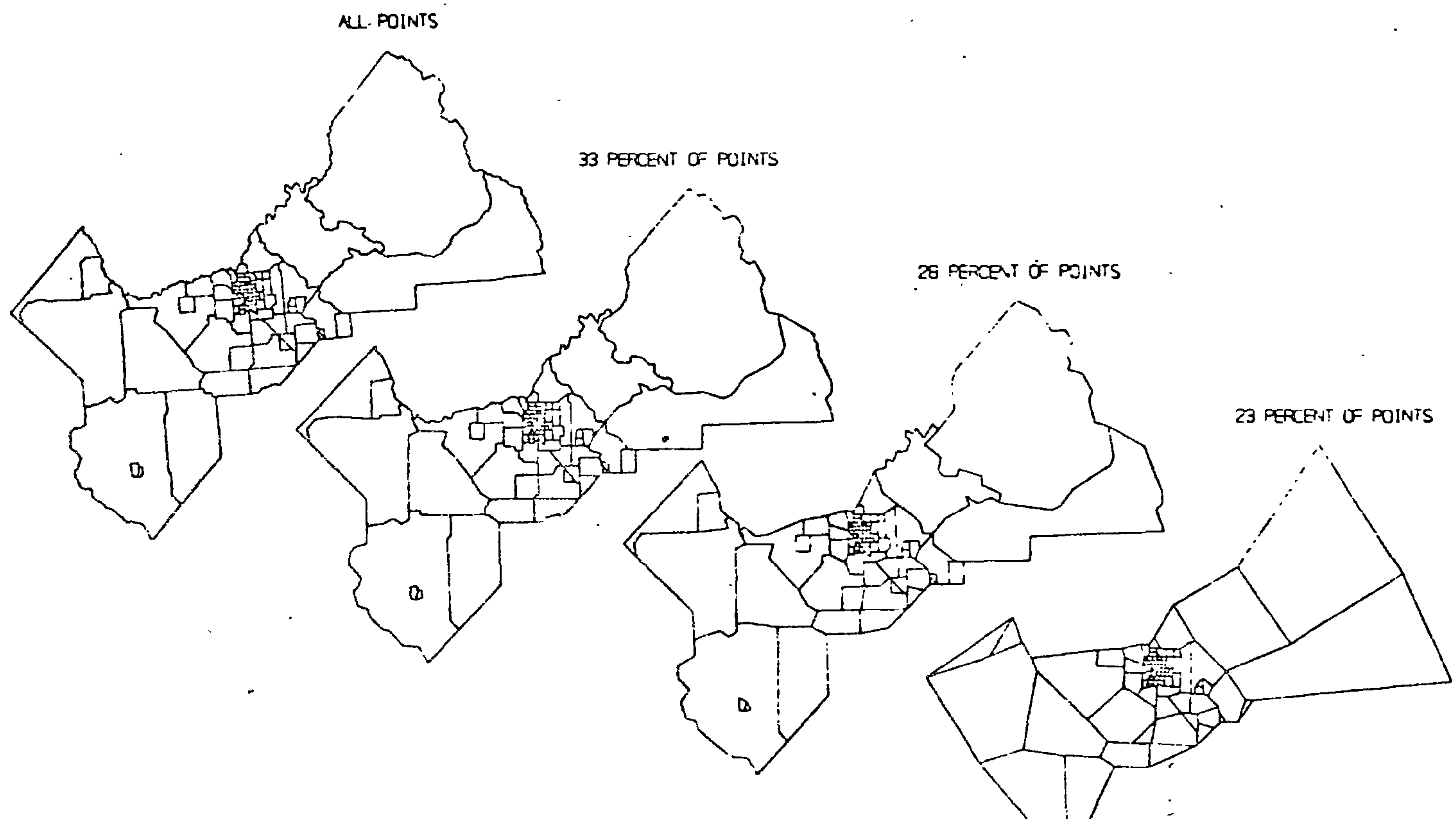
Materials Available

1. POLYVRT program with sample data
2. POLYVRT User's Reference Manual
3. U.S. County DIME File (as created by and available from the U.S. Census Bureau)
4. World Data Bank I File (as created by the U.S. Government and modified by the Laboratory)

Copies of the FORTRAN-IV source program for an IBM 370 computer are available on new, unlabeled, 200 foot reels of 9-track tape written at 800 BPI.

POLYVRT SYSTEM COMPONENTS





Fresno SMSA 1970 Census Tracts

Program Name . . . . . INPOM, Version 1.0

Computer Language . . . . . . FORTRAN IV
Computer Requirements.
Machine . . . . PDP-10, IBM 370
Memory . . . . . 20Kwords, 150K bytes
Mode . . . . Interactive or Batch
Peripheral Devices
Seven data sets plus plot tape for offline plots (Optional)
Tektronix 4000 series cathode ray tube
Pen plotter

★ NOTE ★
INPOM will be available in 1977

Description

INPOM (INteractive POlygon Mapping) is a computer mapping program which produces conformant outline or shaded maps on a cathode-ray tube A study area is partitioned into data zones such as continents, countries, states, counties, census tracts, city blocks or any user-defined areal units. The user provides a cartographic data base (CDB) which contains x-y coordinates describing the data zones of interest

Data values are assigned to each data zone by the user Many variables can be mapped in succession using the same CDB for a given study area INPOM may be used to extract data values from a file or they may be entered from the user's terminal and saved as a file for later use

Ten categories of graphic symbolism may be used to graphically represent the values Graphic symbols are constructed using parallel (hatched) lines, intersecting (cross-hatched) lines, or regularly spaced character symbols Angles and densities of lines, plus selection and spacing of characters are defined within the program However, they may be redefined by the user for any particular map.

A user interacts with INPOM by use of an English-like command language which consists of verbs and direct objects The language may be entered in a free field format using terms which are completely spelled-out or by using any unambiguous abbreviations. The user also may specify parameters affecting the size, scale, and geographical limits of a map to be produced It is also possible to enter into a dialogue with the program when a more complicated series of specifications are required, such as definitions of shading, value ranges, and titles Map output is sent directly to the user's CRT terminal in response to a 'MAP' command or to a pen plotter following a 'PLOT' command

INPOM's cartographic data base defines a study area as lists of x-y coordinates called "chains." Each chain serves the topological functions of connecting two end points (nodes) and bounding two adjacent data zones. Chains usually contain other points between their two end points The actual number of points in a chain is a function of the length and complexity of the boundary represented by the chain Any cartographic data base composed of data zones can be converted by the POLYVRT program to a chain file format for input to INPOM. POLYVRT can accept as input DIME files, CALFORM files or chain files (such as the World Data Bank I file supplied with POLYVRT) and produce a chain file for subsequent input to INPOM CDB coordinate data may also be entered directly into the program from the keyboard of a graphic terminal, or recorded using a digitizing tablet or other graphic input device.

Once structured as an INPOM chain file, the cartographic data base must be stored on disk Using such a working file allows the program to be smaller and achieves flexibility in defining and displaying a map image Random access reading of the disk file also allows for user selection of sub-regions, legends and titles from a large data base plus a high degree of efficiency in plotting of maps

Other features of INPOM include

1. *Line Generalization* — Selectively reduces the number of points used to describe data zone outlines. When displaying a map a user may select any one of 10 detail levels which have been defined for each point on the chain file at the same time it was created by POLYVRT, or INPOM will create (or re-define) detail levels according to the user's specifications
2. *Polygon Selection and Windowing* — Enables a subregion of a CDB to be isolated and mapped The area of interest may be specified by naming the polygons to be included or by providing a set of coordinates which essentially describe a window overlayed on the base map
3. *Titling Capability* — Allows any number of titles, each having up to 5 lines of text to be defined and positioned within a user-defined space on the output map Titles can be stored as files and any combination of titles from a given file can be retrieved for display on a given map
4. *Cartographic Data Base and Data Values Editing Capability* — Allows points on a chain to be modified or new chains and polygons to be created Data values can be input from the terminal or from a disk file and individually modified by the user Edited data values and CDB's can be saved offline. Terminals equipped for graphic input (including tablets) may be used for recording chain formatted coordinates directly from a base map
5. *Lines* — Makes use of chains to define lines as well as polygon boundaries Lines (e.g. roads, rivers) may be mapped in combination with polygon outlines or displayed independently of polygon outlines
6. *Legends* — Defines and positions strings of characters as legends (e.g., place names) on the maps.
7. *Graphic Symbol Key* — Relates graphic symbols to each value class interval and may be displayed alone or inserted within a map Each element of the key may be positioned individually, if desired

INPOM is written in machine-independent FORTRAN IV for PDP-10 or IBM-370 computers. When used at other installations it may require alteration of certain of its modules which handle random access files or address graphic terminals and plotters Graphic subroutine calls are compatible with Tektronix PLOT-10 software

Application Notes

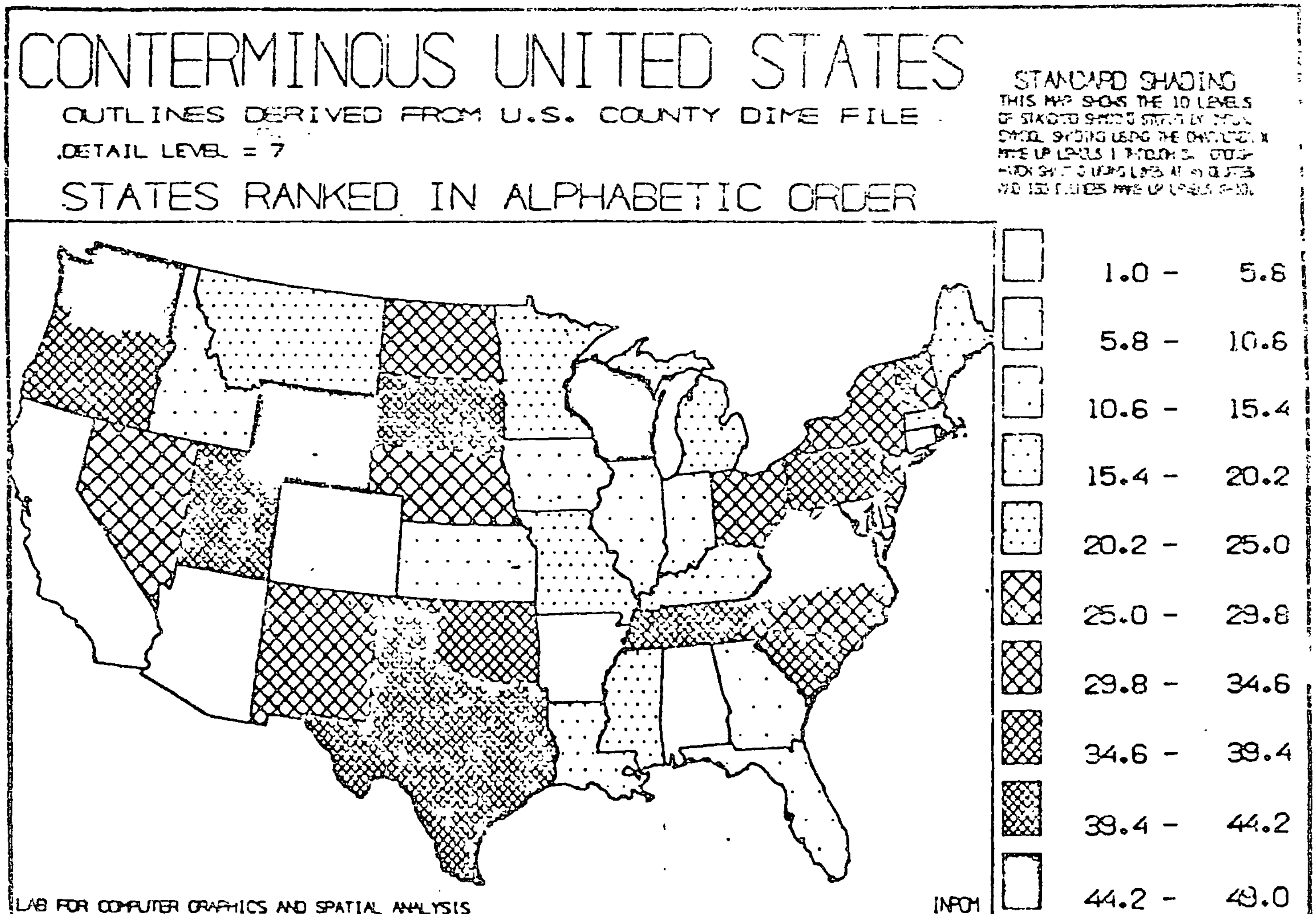
INPOM's ability to display a map on a Cathode Ray Tube or pen plotter provide it with a degree of flexibility unequalled by batch mode graphic programs. A user may very quickly examine a large number of alternative maps, varying either in terms of subject matter, study area or graphic symbolism The idea of using computer generated maps as analytic as well as descriptive tools assumes new meaning when it becomes possible to produce maps within minutes rather than hours or days.

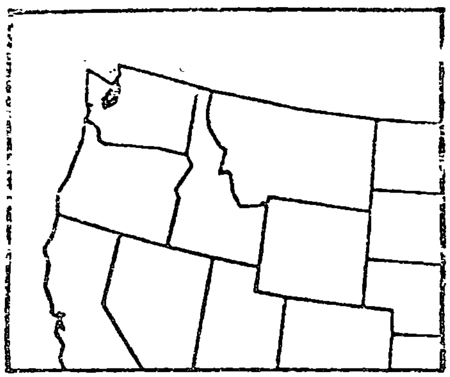
Materials Available

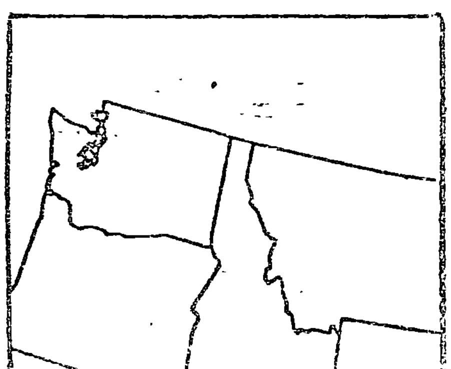
1. INPOM Version 1.0 computer mapping program
2. INPOM User's Reference Manual
3. World Data Bank — I file (as created by the U.S Government and modified by the Laboratory)

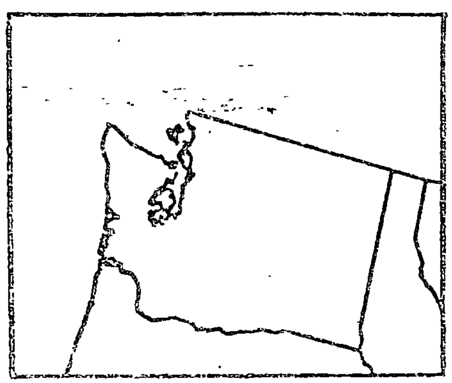
Copies of the FORTRAN-IV source program for an IBM 370 computer are available on new, unlabeled, 200 foot reels of 9 track tape written at 800 BPI.

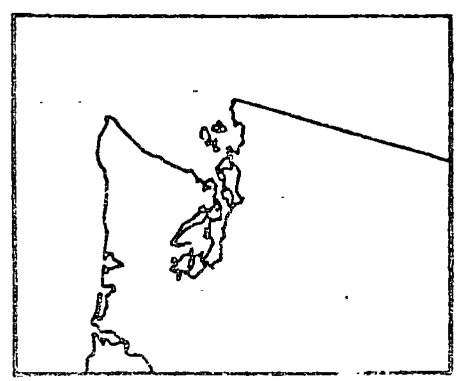
Copies of the FORTRAN-IV source program for a DEC PDP-10 computer are available on a DEC tape in compressed format with DEC specific command files and MACRO-10 files included

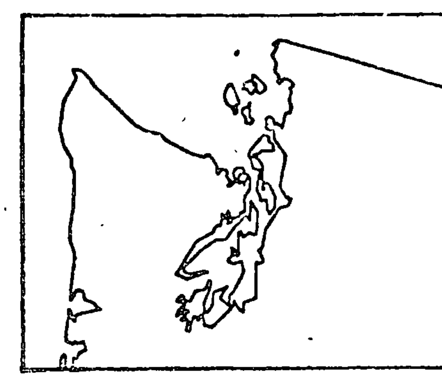












Program Name . . . . . . ASPEX, Version 1.0

Computer Language . . . . FORTRAN IV
Computer Requirements
Machine . . . . PDP-10, IBM 370
Memory . . . . . . 20Kwords, 150K bytes
Mode . . . . Interactive or batch
Peripheral Devices
Three data sets
Tektronix 4000 series cathode ray tube
Pen plotter

Description

ASPEX (Automated Surface Perspectives) is a computer program for producing oblique perspective views of three dimensional surfaces on a cathode ray tube (CRT). The program is an outgrowth of the SYMVU program described elsewhere in this publication. ASPEX, however, is a significantly more powerful tool for displaying surfaces because it operates interactively and is able to accept much larger files of input data while requiring less core memory

Input data for ASPEX consists of a matrix of values identical to that used by SMYVU ASPEX also uses the same method of relief representation, i.e elevation lines drawn along rows, columns, or diagonals including an algorithm for deleting hidden line segments.

ASPEX can be run either in an interactive or batch mode. Interactive mode accepts input of control commands from a CRT keyboard with output to the CRT or to a plotter. Batch mode involves control information being read from cards with output being produced on a line plotter.

APEX's user language involves commands composed of verbs and objects. A verb describes an action to be taken, and an object identifies the entity or value to be acted upon. An object may be a single word, or a word set equal to a value. Furthermore, a value may be a number, a set of numbers, a string of characters, or a symbol for which a value has been previously defined In interactive mode, 'HELP', 'SHOW', and 'LIST' commands are available to assist the user in selecting desired verbs and objects ASPEX currently can accept a data matrix containing up to 500x500 cells (i.e , any number of rows and columns whose product is less than 250,000). Input data is processed by ASPEX and output onto a random access disk file with only a small portion of the data in core memory at any time As a result the program is able to run in a smaller amount of core memory than SYMVU while displaying larger sets of data.

ASPEX allows the user to define the azimuth, altitude and viewing angle relative to the data surface The program also has an alternative mode for defining a view of the surface A user may specify two points in three-dimensional cartesian space, the eye-point of the observer and the center point of the area being viewed. These points may be defined to be at any position in three-space, including positions on the surface

A flexible method for scaling and selecting a "window" of the matrix is also provided. In addition, effects similar to a variable focal length (zoom) lens can be achieved by specifying 1) a "cone of vision" which determines how much of the matrix around the center of vision will be included, and 2) the "focal ratio" which determines where the picture plane (the projected image of the surface) is placed between the two viewing points (eyepoint and centerpoint)

Other features include.

1 A *Subroutine FLEXIN* -- To manipulate the data values of the input matrix prior to their display
2. An *Option to Generalize Relief Lines* — To reduce the number of points thereby allowing reduction of the time and cost required to produce a plot
3 A *Variety of Graphic Cosmetics* — To create endlines at the matrix edges, blocks of varying sizes at the matrix base, graphic scales for data interpretation and special background shading for regions within the square matrix but outside the study area
4. *The Ability to Produce Three Different Surface Projections* — To achieve isometric, planometric, and perspective variations

Application Notes

ASPEX is able to effectively display data surfaces of all kinds This includes data derived from digital terrain models as well as contour, conformant, proximal and trend surface maps produced by SYMAP. The resulting graphics provide a great deal of visual impact The user has a broad range of control over how a data surface is e viewed. For example, one may move across the surface by producing a series of graphic displays. Such a technique would be particularly useful for purposes of film animation where each successive plot could be used to produce several seconds of viewing time on film

Individuals involved in the preparation of data surfaces frequently express a desire to select a viewing angle which provides maximum visibility of the surface This concern reflects the fact that some portion of a surface is usually hidden by "peaks" which rise up and block from view areas behind them The interactive capabilities of ASPEX minimize this problem by allowing the user to freely experiment with various viewing angles, distances, windows, and vertical scalings.

A surface drawing program such as ASPEX allows thematic map data to be displayed without the necessity for data value classification inherent in traditional thematic map symbolism Since each value on the surface is shown raised to a height which is continuously proportional to the value scale, a user is free of the necessity to establish value class intervals and then select corresponding symbolism A three dimensional surface is literally classless thereby allowing minor as well as major fluctuations within the data to be shown on the same drawing. There is a need to establish a vertical scale for a data surface but associated limitations are far less significant than those which result from having to classify and symbolize data on a traditional two dimensional shaded map.

Materials Available

1. ASPEX program with sample data
2. ASPEX User's Reference Manual

Copies of the FORTRAN IV source program for an IBM 370 computer are available on new, unlabeled 200 foot reels of 9 track tape written at 800 BPI.

For installation of the program on a DEC PDP-10 computer a user alternatively may request a DEC tape in compressed format with DEC specific command files and MACRO-10 files included

Selected Maps of U.S. Population Density from "Manifested Destiny"

CARTOGRAPHIC DATA BASES

Introduction

A cartographic data base (CDB) is a necessary part of the input data required by the Laboratory's conformant mapping programs. These data bases define the x-y coordinate locations for each data zone which is to appear on a computer generated map. A CDB may be digitized locally but it is usually more economical to obtain a copy of an existing CDB if one already has been prepared for the area of interest.

CDBs Available

The Laboratory has available a selected set of CDBs and will consider requests to prepare others to a user's specifications. CDBs distributed by the Laboratory are available in many different record and file formats, including those required by the SYMAP, CALFORM and INPOM computer mapping programs. Alternatively, an individual may prefer to acquire a CDB in a topological chain format and use the Laboratory's POLYVRT software to create a variety of CDBs for use with one or more mapping programs.

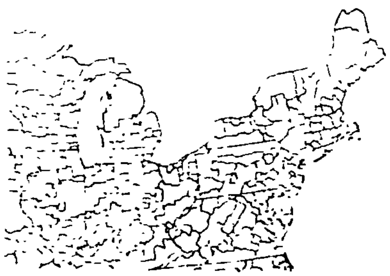
Original Data Sources

CDBs offered by the Laboratory frequently have been derived from other files such as the CIA's World Data Bank-1 and the Census Bureau's County DIME, Metropolitan DIME, and Urban Atlas files. In most instances the content and/or structure of these files have been modified by the Laboratory. Silver error, overlapping polygons and similar difficulties inherent in the original data have been corrected by the Laboratory.

Centroid (Data Point) Files

For users of SYMAP or other programs which utilize x-y coordinate data for point locations, files of centroids (central points representative of given areal units) are available. Files currently being distributed include:

1. Centroids of U.S. counties
2. Centroids of U.S. States
3. Centroids of Massachusetts cities and towns.



Standard Economic Areas (SEAS)

Standard Metropolitan Statistical Areas (SMSAS)

Countries of the World

Study areas and their geographic subdivision (data zones) for which CDBs are available include the following:

Study Area	Geographic Units	Standard Format	Original Source	Coordinate System	Recommended Maximum Mapping Scale	Maximum Ground Resolution per 0.1"	1mm on map
Planet Earth	Countries	Chain	WDB-I	Radians, Lat. & Long	1:10,000,000	16 mi.	10 km
United States	States	"	County DIME File	"	1:5,000,000	8.0 mi.	5 km
United States	SMSAs	"	"	"	"	"	"
United States	SEAs	"	"	"	"	"	"
United States	Counties	"	"	"	"	"	"
Any State	Counties	"	"	"	"	"	"
Any SMSA	Census tracts, 1970	"	Urban Atlas Files	Degrees, Lat. & Long	1:15,000	125 ft.	15 m
Any SMSA	Census blocks, 1970	Metro-DIME	Metro-DIME	Radians, Lat. & Long	1:5,000	40 ft.	5 m
Massachusetts	Cities & Towns	Chain	Files locally prepared	Inches, Cartesian	1:250,000	2,100 ft.	250 m

Line Generalization (Detail)

The amount of line detail (number of points) contained within a file may be specified at the time that an order is placed. Alternatively the Laboratory will provide a degree of line detail appropriate to

1. The mapping package with which the CDB is to be used,
2. The maximum size at which all or some part of the file is to be displayed and
3. The line resolution of the device on which the output is to appear

All of the Laboratory's CDB files contain a finite amount of detail. However, when preparing a CDB for mapping purposes, it is desirable to generalize each line in order to retain only those points which can be seen on the final graphic output. The actual number of useful (visible) points depends upon the scale of the final map output plus the graphic resolution of the display hardware. Clearly, a line printer map can portray less detail than a cathode ray tube which in turn has less revolving power than most pen plotters for a map at a given scale. Other issues to be considered when selecting a detail level include the maximum number of points which can be processed by a given display program plus the relative cost of processing large files.

The standard version of each CDB whose standard format is "chain" includes with each x,y coordinate pair a third number which is the measure of that point's deviation from the trend line. As a result, each point may be either included or excluded from a given plot depending upon the amount of line resolution desired in a given situation. This capability provides a great deal of flexibility to the user and allows for a variety of different CDB's, varying in line detail, to be produced from the original file. For a description of the algorithm which determines deviations and a discussion of their use in "detail filtering" see Douglas, David and Thomas Peucker, "Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature" in the *Canadian Cartographer*, Vol 10, No 2, Dec 73, pp 112-122

Record Formats

CDB's distributed by the Laboratory are available in any one of four standard formats or in a format defined by the user. Standard formats include files prepared specifically for input to the SYMAP, CALFORM or INPOM computer mapping programs. Files also may be obtained in a topological chain file format for input to the POLYVRT program and other mapping software to be distributed by the Laboratory.

Record Content

CDB's prepared for input to a computer mapping program include x-y coordinates plus a geographic identification code for each polygon. Codes used are Federal Information Processing Standards (FIPS) or Census Bureau geographic identifiers. INPOM and POLYVRT files also include a detail level code for each point coordinate in the file. For CDB's other than those described which are prepared to a user's specifications, geographic codes appropriate to the locations of interest can also be provided

POLYVRT Software

All CDB's offered by the Laboratory are directly compatible with the computer mapping software for which they are intended. However, for those individuals wishing to acquire a high degree of flexibility in working with CDB's for one or more study areas, it may be advisable to request a CDB in a POLYVRT chain file format plus a copy of the POLYVRT software. By doing so one acquires the capabilities to:

1. Generate CDB's in a variety of formats (e.g., CALFORM, SYMAP, INPOM, etc.)
2. Select only those geographic regions of immediate interest from a larger file
3. Specify any one of several map projections
4. Specify any level of line generalization desired
5. Edit or modify the content of a CDB file in order to satisfy the users' specific requirement.

Map Projections

WDB-I, the County DIME, METRO-DIME and the Urban Atlas files contain coordinates expressed in terms of latitude and longitude. These coordinates usually are projected to a cartesian coordinate system before being used with a computer mapping package. Files describing a region no larger than a Standard Metropolitan Statistical Area (SMSA) are normally created using a standard equi-rectangular projection. For larger regions within the U.S. (e.g. one or more states) a user may wish to request a different projection. CDB files for regions as large as the entire United States will normally require projection. Projections available include Albers, Lambert conformal, Mercator, Miller, Equi-Rectangular, Sinusoidal and Orthographic. Other types can be furnished upon special request. CDB's prepared from the DIME and World Data Bank files will be provided with conic (usually Albers) or Sinusoidal projections, unless specific projection instructions are provided. Such instructions should include necessary projection parameters.

Documentation

Each CDB includes documentation describing the following.

1. Number of polygons and number of points
2. Number of points per polygon
3. Number of chains (for chain files)
4. Identification codes for each geographic unit (FIPS code and name)
5. Minimum and maximum x-y coordinate values for the CDB
6. Map projection parameters (when known)
7. Test plot of polygon outlines
8. Record and file descriptions

File Characteristics

Files will normally be furnished on new, unlabeled 9-track tape, written in EBCDIC mode at 800 BPI, using a blocksize of 4000 bytes or less. If this is not suitable, specify the character set, density, parity and blocking factor appropriate for your installation

Cost

The cost of a CDB is a function of the effort involved in preparing it to a user's specifications. A CDB which can be prepared from WDB-I or the County DIME files is likely to be less costly per point than are similar files derived from Urban Atlas or Metro DIME files. The reason for this is that the latter files frequently include errors which must be identified and corrected prior to their conversion to a CDB file. However, if the file for a region has already been corrected this will be reflected in its cost. Because of the many different variables and user specifications which must be taken into account, those interested in acquiring a specific CDB should request a price quotation.

When requesting a CDB price quotation, the following information must be provided

1. Geographic region of interest
2. Geographic data zones desired (type, number and names as appropriate)
3. Computer mapping software package to be used. If other than a Laboratory package, specify the data format required, restrictions concerning maximum number of points or polygons and the device on which graphic output is to be displayed.
4. Map projection desired (if any)
5. Line generalization required in terms of:
   - Smallest scale output map
   - Positional accuracy
   - Precision of output display device
6. File characteristics (if other than as described above)

Price estimates range from a minimum of $75 for a copy of an existing file to $100+ for a file derived from WDB-I, the Urban Atlas, County DIME or METRO-DIME files

Delivery Time

CDB's derived from WDB-I or the County DIME file can normally be shipped within 14 days from receipt of an order. Urban Atlas and Metro-DIME files may require somewhat longer due to additional processing involved. However all orders are normally shipped within 2-4 weeks

PUBLICATIONS

RED BOOK

This is an illustrated synopsis of projects undertaken since the Lab's organization in 1965. The "Red Book" describes the Laboratory's applications of computer graphics and spatial analysis to such fields as architecture, city planning, landscape architecture and theoretical geography. Descriptions of how other research communities have applied the Lab's programs are also included. "Red Book" is in 8½ x 11 loose-leaf format, chronologically organized for easy reference and convenient updating. Research entries for successive years are separated by dated dividers.

THEORETICAL CARTOGRAPHY PAPERS

"Issues in Thematic Map Design" Series

In 1976 the Harvard Laboratory for Computer Graphics and Spatial Analysis began publication of a series of discussion papers dealing with new concepts and problems in thematic mapping. Consideration is given to maps made by hand as well as by computer. Based on extensive Ford Foundation supported research begun in 1969 by Howard T. Fisher and various associates, these papers are of three principal types

1. Papers dealing with fundamental cartographic principles without reference to specific mapping situations
2. Papers illustrating and discussing a variety of alternative solutions to representative mapping problems, and
3. Papers analyzing in detail alternative approaches to thematic cartography as presented by leading cartographic texts in English, French, and German

The papers of the second type, which assume no previous knowledge of cartography, include numerous specially prepared maps designed to show the relative advantages of different design approaches. The first of these illustrative studies presents more than 50 maps showing some 14 different basic types of symbolism as applied to the problem under consideration.

As with past Harvard Papers, these are of an informal and exploratory nature with a view of eliciting reader comments. It is believed that this series offers important new insights and information capable of contributing to both the theoretical and practical advance of thematic cartography. To [illegible] up-dated information regarding specific papers as issued, interested persons should request placement on our mailing list. Arrangements can be made for the papers in this series to be sent automatically as they become available. Those requesting this service will be sent without charge a periodic abstract of such comment as may be received regarding the papers.

Below is a tentative list of titles, pursuant to the first two categories mentioned

Fundamental Cartographic Principles

1. Thematic Maps: What They Are and Who Needs Them?

2. Wanted: An Improved Theoretical Construct to Aid in the Design of Thematic Maps

3. Hypotheses for the Mapping of Qualitative and Quantative Information

4. Thematic Map Titles — What Should They Contain?

5. Types of Study Spaces in Thematic Mapping — Distinctions with a Difference

6. Types of Locations in Thematic Mapping — Distinctions with a Difference

7. Types of Values in Thematic Mapping — Distinctions with a Difference

8. Value Keys in Thematic Mapping — What Should They Contain?

9. A Manual of Value Symbolism — Spot Type

10. A Manual of Value Symbolism — Band Type

11. A Manual of Value Symbolism — Field Type

12. A Manual of Value Symbolism — Cyclical Type

13. Multi-Subject Mapping — With Interlocking Subjects

14. Multi-Subject Mapping — With Diverse Subjects

15. The Use of Color as a Quantitative Analogue in Thematic Mapping

16. A Glossary of Terms Used in Thematic Mapping

Alternative Solutions to Representative Problems

1. Foursquare I — An Introductory Problem in Single-Layer Thematic Mapping

2. Foursquare II — Illustrating Basic Symbolisms for the Representation of Values

3. Foursquare III — Value Curves and Classing Procedures

4. Sparsely Populated France — An Illustrative Problem with Alternative Solutions

5. Densely Populated France — An Illustrative Problem with Alternative Solutions

6. All France I — An Illustrative Problem with Alternative Solutions

7. All France II — An Illustrative Problem with Alternative Solutions

8. The Washington-Boston Corridor — An Illustrative Problem with Alternative Solutions

9. U.S. Standard Metropolitan Statistical Areas (SMSAs) With Over One Million Inhabitants — An Illustrative Problem with Alternative Solutions

10. The White House Area, Washington, D.C. — An Illustrative Problem with Alternative Solutions

11. Twelvesquare I — An Introductory Problem in Multi-Layer Thematic Mapping

12. Twelvesquare II — The Thematic Mapping of Non-Layered Space

AUTOMATED CARTOGRAPHY PAPERS

NOTE: The first 12 papers report on a Laboratory study that demonstrates the role of computer graphics in presenting data related to the source, distribution, and effect of air pollution.

1. SUMMARY REPORT 150 pp., $9.00. There are 3 sections: 1) a summary of the entire project, 2) a discussion of each individual study area, 3) a technical report and a discussion of computer graphic outputs.

2. SUMMARY GRAPHICS 162 pp., $9.75. This is a complete set of graphics for all individual case studies

3. COMPUTER EQUIPMENT AND PROGRAMS 83 pp., $5.00. This report discusses computing machinery (hardware), computer programs (software), plus types of maps and data for graphic display. Also, the introduction considers computer mapping criteria, operating principles, and current and future applications

SYMAP INTERPOLATION CHARACTERISTICS 70 pp., $4.20. There are four sections: 1) the SYMAP interpolation algorithm, 2) sensitivity and accuracy of interpolation, 3) use of electives to affect interpolation and 4) use of barriers to affect interpolation. A comparison of SYMAP's algorithm with other methods of interpolation is also included

CASE STUDY REPORTS These are detailed descriptions of individual case study plus technical reports relating to other research undertaken. Relevant graphics are included with each report.

5. ST. LOUIS REGION CASE STUDY 3 vol. 326 pp., $19.75. This includes comparisons between air quality and socio-economic data, examination of diurnal variations in air quality, modeling of air quality from power plant emissions and statistical examination of the data.

6. KANSAS CITY REGION CASE STUDY 112 pp., $6.75. Air quality data is manipulated and mapped with particular emphasis on background values at the periphery of the study area. These include defining coherent areas for mapping, time-averaging of data and comparison with air quality standards. Also emission densities are mapped and air quality is compared with socio-economic data

7. MONTREAL REGION CASE STUDY 59 pp., $3.75. An Index of Atmospheric Purity (I.A.P.) is determined and mapped in this study. The coverage and extent of various lichens and mosses are statistically examined and mapped to achieve the best interpretation for the limited amount of data gathering. In addition, the I.A.P. was compared to measured air quality

8. CONNECTICUT STUDY 44 pp., $3.00. Air quality mapped and compared at various time periods. Comparisons are then made using mathematical models

9. SOUTHERN NEW ENGLAND 55 pp., $3.50. Measured air quality supplemented with estimated background values is compared to socio-economic surfaces

10. PUGET SOUND REGION 32 pp., $2.00. Air pollution emissions from future transportation and urban development options are mapped. Transportation-related air qualities reflect the influence of mountain barriers to pollutant dispersion

11. CALIFORNIA REGION 49 pp., $3.00. Investigations were made of carbon monoxide concentrations, the total annual suspended dust concentrations and air quality relative to statewide air quality standards.

12. BOSTON REGION 51 pp., $3.25. This report illustrates differences in statistical surfaces based on daily, monthly, seasonal and [illegible] averages.

13. The Use of Computer Graphics in Planning, by Howard Fisher, presented at the 1970 National Conference of the American Society of Planning Officials, April 6, 1970. 9 pp. $1.00

"Those involved in urban planning are not likely to be able to come to sound conclusions and be able to make sound recommendations unless the facts regarding the human community under study are readily available in easily understood form."

14. Maps ala Computers, reprint of a March 11, 1970 article from the Christian Science Monitor. $1.00

"It's hard to know where to begin solving such amorphous challenges as crime and air pollution in cities. A group at Harvard, with the help of a computer, is mapping concentrations of these problems. These computer maps lend a clarifying visual perspective."

15. Computer Cartography, by Thomas K. Peucker, Commission on College Geography, Resource Paper No. 17, Association of American Geographers, 1972. 75 pp. $3.00

A valuable tutorial covering such topics as theory of data surface processing and representation of surfaces. The publication also contains numerous illustrations involving over twenty computer mapping programs.

16. Thematic Cartography, by Phillip Muehrcke, Commission on College Geography, Resource Paper No. 19, Association of American Geographers, 1972. 66 pp. $3.00

This report describes the various steps involved in thematic mapping: data collection, mapping and analysis. Specific subjects discussed include spatial sampling, measurement, symbolic representation, visual analysis and quantitative map analysis.

17. Management by Computer Graphics, by Kenneth Shostack and Charles Eddy, Harvard Business Review, November-December 1972. 12 pp. $1.00

"The tremendous output of the high-speed computer has far outstripped the modern executive's ability to examine, absorb, and use all the information generated in his day-to-day decision making. But this imbalance between the machine and the manager is now being corrected through the development of computer graphics."

18. Color in Art, by Howard T. Fisher [illegible] Carpenter Center for [illegible] Harvard University, [illegible] 134 pp. $11.00

An excellent introduction to color and color theory presented from the special viewpoint of those concerned with the use of color in art and design work, based upon a study originally undertaken with specific reference to the use of color in thematic map design. Includes among its many black and white and color illustrations numerous charts and diagrams, of which four of particular reference value are in color.

19. Manifested Destiny. A Graphic Account of the Settlement and Growth of America 1790-1970, by Geoffrey Dutton, Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University, 1975. 30 pp. $2.00

Surfaces of population distribution for the conterminous United States are displayed (using the ASPEX program) for each census date from 1790 through 1970. Total population by county were used as input data, allocated to a grid for display. The spreading of settlement and the growth of cities are dramatically illustrated by the changing population surface.

20. Programming for Transportability. A Guide to Machine Independent FORTRAN, by Nicholas Chrisman and Denis White, Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University, March 1976. 40 pp. $2.50

This paper addresses the problem of creating a "transportable" subset of FORTRAN. Four aspects of the problem are dealt with:

1. Syntactic limitations
2. Semantic problems
3. Necessary evils
4. Suggestions about good practice

This publication is based on experience with IBM, CDC, PDP, UNIVAC, Honeywell and Burroughs operating systems.

21. Computer Cartography: World-Wide Technology and Markets, by Eric Teicholz and Julius Dorfman, International Technology Marketing, Newton, Massachusetts, May 1976. 427 pp. $275.00

The study covers all aspects of automated cartography including the automated cartography process, users and sources of automated cartography in the United States, Canada, Europe, Latin America and Japan, the status and forecast of automated cartography technology and a market survey and forecast.

THEORETICAL GEOGRAPHY PAPERS

1. Concepts and Applications—Spatial Order, by William Warntz and Michael Woldenberg, 16 May 1967. 135 pp. $12.00

2. A Nomographic Representation of the Geoid, by Walter Messcher, 28 August 1967. 13 pp. $1.75

3. Implicit Map Projections in Computer Print-Outs, by William Warntz, 15 September 1967. 24 pp. $2.25

4. Out of Print.

5. Superceded by paper No. 6.

6. Superceded by paper No. 25.

7. The Geometry of Mixed Hexagonal Hierarchies in the Context of Central Place Theory, by C. Ernesto S. Lindgren, 22 December 1967. 44 pp. $3.00

8. Energy Flow and Spatial Order, with Special References to Mixed Hexagonal Central Place Hierarchies, by Michael J. Woldenberg, 3 January 1968. 46 pp. $3.25

9. The Continent Problem — Geography and Spatial Variance, by Christopher W. Warntz, 32 pp. $2.25

10. Distances and Land Values as Data for Introducing Problems Associated with Spatially Continuous Fields of Correlation Coefficients, by William Warntz, 29 January 1968. 18 pp. $1.50

11. Space Straightening and Flattening, by C. Ernesto S. Lindgren, 2 February 1968. 6 pp. $1.50

12. Out of Print.

13. Spatial Order in Fluvial Systems: Horton's Laws Derived from Mixed Hexagonal Hierarchies of Drainage Basin Areas, by Michael J. Woldenberg, 14 February 1968. 37 pp. $3.60

14. A Nomographic Representation of Trajectories, by Walter Messcher, 8 March 1968. 11 pp. $1.50

15. A Two-Dimensional Interpolation Function for Computer Mapping of Irregularly Spaced Data, by Donald Shepard, 20 March 1968. 20 pp. $1.50

16. N-Dimensional Spatial Analysis and Computer Graphics: Part A, by C. Ernesto S. Lindgren, 14 April 1968. 60 pp. $4.20

17. Hyper-surfaces and Geodesic Lines in 4-D Euclidean Space, by C. Ernesto S. Lindgren, 17 May 1968. 15 pp. $1.50

18. A Note on Stream Ordering and Contour Mapping, by William Warntz, 1 July 1968. 30 pp. $2.00

19. Hierarchical Systems: Cities, Rivers, Alpine Glaciers, Bovine Livers and Trees, by Michael J. Woldenberg, 5 July 1968. 160 pp. $10.00

20. A Study of Rivers and Other Branching Systems, by Gordon Howie, 31 July 1968. 19 pp. [illegible]

21. Geography and an Existence Theorem: A Cartographic Computer Solution to the Localization on a Sphere of Sets of Equal-Valued Antipodal Points for Two Continuous Distributions with Practical Applications to the Real Earth, by Stephen E. Sczkowitz, 5 August 1968. 64 pp. $4.50

22. A Mathematical Representation of the Altitude Relationships on the Surface of the Earth, Using Spherical Functions to the 16th Order, by Albert Prey, translated by Bonnie Binder, August 1968. 62 pp. $4.50

23. Plane Globe Projection — A Linnean System of Map Projection, Translation of Min. Rat. Prof. Dr. Hans Maurer's "Ebene Kugelbilder" forwarded and edited by William Warntz, 19 August 1968. 337 pp. $18.00

24. The Descriptive Geometry (or Representative) of a Collection of Points Fixed by N Coordinate Numbers or of N-Dimensions, by Felippe dos Santos Reis, translated by C. Ernesto S. Lindgren, 30 September 1968. 31 pp. $2.25

25. The Law of Travel and Its Application to Rail Traffic, by Eduard Lill, translated and forwarded by Thomas K. Peucker, 10 January 1969. 123 pp. $7.75

26. Minimum Time Paths and the Migration of the Arctic Tern, by Robert C. Eckhardt, 7 February 1969. 26 pp. $2.00

27. Notes on the Methodology for Generation of the Representative of a Set, by C. Ernesto S. Lindgren, 14 February 1969. 39 pp. $2.75

28. A Minimum Path Problem Reconsidered, by C. Ernesto S. Lindgren, 21 February 1969. 11 pp. $1.50

29. The Use of the Geodesic Curvature in the Determination of Geodesic Lines, by C. Ernesto S. Lindgren, 26 February 1969. 13 pp. $1.50

30. Numerical-Geometrical Techniques for Information Storage and Retrieval, by C. Ernesto S. Lindgren, 1 October 1969. 8 pp. $1.50

31. Some Reflections on Concepts Based on Three-Dimensional Geometry, by C. Ernesto S. Lindgren, 8 October 1969. 13 pp. $1.50

32. Algorithms and Models Based on Projective Transformations in Spatial Location, Regional Planning, and Central Place Theory, by C. Ernesto S. Lindgren, 17 October 1969. 14 pp. $1.50

33. Graphical Representation of a Matrix with Applications in Spatial Location, by C. Ernesto S. Lindgren and Carl Steinitz, 30 October 1969. 53 pp. $3.75

34 Some Thought on Optimal Mapping and Coding of Surfaces, by Thomas K Peucker, Ph D , 3 November 1969 14 pp , $1 50

35. Homological Transformations in Four-Dimensional Space, by C Ernesto S Lindgren, 25 November 1969 20 pp , $1 50

36 A Study of the Movement of a Point on a Plane and in Space, by C Ernesto S Lindgren, 7 December 1969, 16 pp , $1 50

37 Set of Equal-Value Antipodal Points for Two Continuous Distributions, by C Ernesto S Lindgren 12 December 1969, 11 pp , $1 50

38 An Outline for the Theory of Man-Made Space — Essays in Urbanology, Number One, by Kozmas Balkus, 26 February 1970 55 pp , $4 00

39 A Structural Taxonomy of Spatial Hierarchies, by Michael J Woldenberg, 24 March 1970, 53 pp., $4 00

40. The Determination of Fixed-Points in Finite Dimensional Spaces, by C Ernesto S Lindgren, 3 June 1970, 17 pp , $1 50.

41. Law and Order in the Human Lung, by Michael J Woldenberg, Gordon Cumming, Keith Harding, Keith Horsfield, Keith Prowse and Shiam Singhal 29 July 1970, 58 pp , $4 00

42. The Hexagon as a Spatial Average, by Michael J Woldenberg, 15 October 1970, 26 pp $2 00

43. The Two Dimensional Spatial Organization of Clear Creek and Old Man Creek, Iowa, by Michael J Woldenberg, 23 June 1971, 34 pp , $2 50

44. The Sandwich Theorem — A Basic One for Geography, by William Warntz, C Ernesto S Lindgren, Katharine Kiernan, Louisa Bonfiglioli, Eduardo Lozano, 30 June 1971, 88 pp , $5 25

45. Relations Between Horton's Laws and Hydraulic Geometry as Applied to Tidal Networks, by Michael J Woldenberg, 10 June 1972, 40 pp , $3 00

46. The Two-Dimensional Spatial Analysis of the Pecatonica River in Wisconsin, by Michael J Woldenberg and Larry Onesti, 1 July 1972, 57 pp , $4 00

47. A Computer Program for Mixed Hexagonal Hierarchies, by Rachel Thurston, Michael Woldenberg and David Barer, 15 June 1975, 214 pp , $13 00

48. Fred K. Schaefer and the Science of Geography, by William Bunge, Detroit Geographical Expedition, 1, Detroit, Michigan, 1 November 1968, 26 pp , $1 50.

49 Sparking Potential, Personal Interaction and Social Distance: Directions for a Theory, by Geoffrey Dutton, Graduate Student, Department of City and Regional Planning, Harvard University, Cambridge, Massachusetts 1 August 1969, 11 pp $1 50

50. Macroscopic Aspects of Metropolitan Evolution, by Geoffrey H Dutton, 11 March 1970, 116 pp , $7 25

51. Notes on the Friction of Distance in the U.S. Telephone Network, 1935-1995, by Geoffrey H Dutton, November 1971, 11 pp , $1 50

52 Tabulations of Data on Area, Population, Income and Certain Derived Quantities for the 3070 Counties of the 48 Conterminous States of the United States, 1967, by Geoffrey Dutton Katharine Kiernan, Douglas Kingsbury and William Warntz 19 May 1971, 324 pp , $19 25

53. The Geographical Distribution of Income in the Conterminous United States, 1967-68, and the Income Fronts by States, by William Warntz, 24 May 1971, 20 pp $1 50

54. A Description of the 1967-68 United States Income Potential Surface, by Douglas Kingsbury, 20 June 1971, 13 pp , $1 50

55. National and Regional Parameters of Growth and Distribution of Urban Population In the United States, 1790-1970, by Geoffrey H Dutton, 9 July 1971, 30 pp , $2 25

56. Allometric Growth in Social Systems, by Michael J Woldenberg 15 pp , $1 50

57 Allometry in Micro-Environment Morphology, by Ranko Bon Graduate Student, Department of City and Regional Planning, Harvard University, June 1972 32 pp , $2 50

OTHER PAPERS

1. The Potential of Video Tape Recorders for the Design Professions, Eric Teicholz, Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University, July 1972, 78 pp., $1.50.

Video tape recorder technology is one of the most rapidly developing communication technologies This paper looks at aspects of new technology, talks about the differences between film and video tape, discusses the relationship between VTR and computer technology and suggests some potential applications to the design professions.

2. CPED Computer Programs in Environmental Design, Kaiman Lee Center for Environmental Research, Boston, Massachusetts 1974, 1300 pp , $210 00.

CPED is a 5 volume set of reference manuals containing documented and illustrated computer program abstracts A reliable design and planning data source for architects, researchers, and others concerned with computer technology. Contents include feasibility studies architectural programming, site planning, relational planning, two and three dimensional graphics, cost control, environmental control, circulation analysis, text manipulation, project control, office management and evaluation.

3. Computer Aided Space Planning by Eric Teicholz, a paper presented at the First Bi-National (U.S./Australian) Urban Systems Symposium, September 1974 23 pp , $1 50

This paper discusses general approaches to using a computer for space planning aid in an architectural, urban and regional context Programs are classified and various techniques for space planning are described and illustrated

4. Interactive Mapping of Urban Data by Eric Teicholz a reprint of a paper that appeared in the Proceedings of the Second General Assembly of the *World Future Society*, June 1975, 12 pp., $1.50

The paper describes several factors contributing to the explosion of statistical and geographical data related to urban areas. It also deals with corresponding interest in automated procedures for the input, analysis and display of spatial data. INPOM and ASPEX, the Laboratory's two and three-dimensional interactive mapping programs, are briefly described

5 Interactive Graphics Comes of Age by Eric Teicholz, a reprint of the *Datamation* article appearing in December 1975, 4 pp., $1.50.

The paper is a brief history of interactive computer graphics that shows technical trends in industries and universities There is a comparison of the turnkey integrated interactive graphics systems along with market figures and application areas for this newly emerging field.

6. Computer Graphics. A Perspective by Eric Teicholz, a reprint of a paper appearing in *Biosciences Communications*, S. Karger AG, Basel, Switzerland, January 1976, 17 pp , $1 50.

The paper surveys basic ideas and approaches to computer graphics in general and to interactive computer graphics in particular It is an historical overview including speculations on the future of interactive graphics with reports on various hardware systems and their programming implications.

7. How to Talk to a Computer Applications Consultant by Eric Teicholz, a paper presented at the National Computer Conference New York City, June 1976, 8 pp , $1 00 A discussion of the first (and last?) ten years of computer aided architectural design in relation to evolving technologies and architectural practices

8. Graphic Technology and the Display of Spatial Data by Eric Teicholz, a paper presented at the Wescon Conference, Los Angeles, Calif , November 1976, 10 pp $1 00 Describes the unique characteristics of spatial data as a subset of graphics Hardware and software considerations related to the display of spatial are also discussed

EDUCATIONAL AIDS

Self-Study

Self-study instructional materials are available concerning use of the SYMAP program They are designed for students having neither knowledge of geography nor experience with statistical and quantitative mapping The materials include five lessons to acquaint the user with the various capabilities of SYMAP Each lesson successively requires greater use of various input packages and mapping electives plus sample problems to be completed by the reader

Video Tapes

Through a grant from the Alfred P Sloan Foundation, the Laboratory distributes half-inch BAJ-standard video tapes of the SYMAP, SYMVU, and GRID programs which introduce the prospective user to specific computer mapping programs Each tape presents the various output devices used to produce computer maps, and shows the step-by-step preparation of a computer map using the program Because of their general nature, the tapes are intended only as supplements to the user reference manuals which are available for each program.

Order Form

Computer Programs and Related Materials

Program Distribution — General Information

Dissemination of Computer Programs
An organization which requests a program from the Laboratory must agree in writing that the program only will be used for its own internal operations and will not be made available to others without prior written consent of the Laboratory. A University computing installation and all its users are defined as a single user, a commercial service bureau or time sharing company offering use of the Laboratory's programs is *not* considered to be a single user. A commercial vendor such as a service bureau or time sharing company wishing to provide access to the Laboratory's programs for its customers should contact the Laboratory for leasing arrangements.

Postage and Handling
Computer programs and other items recorded on magnetic tape or punched cards are shipped to locations within the U.S. by first class mail, postage and handling prepaid. Documentation materials normally provided with a program are included as part of the shipment. All other publications are sent as third class mail, printed matter.

Computer programs and publications sent outside the U.S. are shipped via surface mail unless other arrangements have been made in advance. Price estimates for air mail shipment of specific items to another country will be sent upon request. Alternatively, additional funds to cover the cost of air mail shipment may be enclosed with an order. Any funds in excess of the amount required will be refunded.

Telephone Orders
Telephone orders will be accepted from organizations within the U.S. A purchase order number is required. Orders may be placed during the hours of 9:30 AM to 4:30 PM by calling area code (617) 495-2526.

Payment
Payment should accompany an order unless a purchase order number has been included. Payment must accompany all orders of less than $5.00. Orders to be shiped outside the U.S. must include advance payment in the form of a bank draft payable in U.S. dollars.

Prices
Prices marked with an asterisk (*) apply to educational institutions and governmental agencies only.

Computer Program	Quantity	Cost (US$)	*	Amount
SYMAP				
1. Mapping Program incl. items 2 through 6 below	____	$1,000.00	$665.00*	____
2. User's Reference Manual	____	10.00		____
3. Self Study Course (5 lessons)	____	25.00		____
4. Interpolation Characteristics Report	____	5.00		____
5. Grid Sheet	____	1.00		____
6. SYMAP Ruler	____	.50		____
7. SYMAP Replacement Tape	____	100.00		____
8. SYMAP Video Tape	____	40.00		____
SYMVU				
1. Mapping Program incl. User's Reference Manual	____	715.00	$475.00*	____
2. User's Reference Manual	____	5.00		____
3. SYMVU Replacement Tape	____	71.50		____
4. SYMVU Video Tape	____		40.00	____
GRID				
1. Mapping Program incl. User's Reference Manual	____	575.00	$380.00*	____
2. User's Reference Manual	____	4.00		____
3. GRID Replacement Tape	____	57.50		____
4. GRID Video Tape	____	40.00		____
CALFORM				
1. Mapping Program incl. User's Reference Manual	____	715.00	$475.00*	____
2. User's Reference Manual	____	5.00		____
3. CALFORM Replacement Tape	____	71.50		____
POLYVRT				
1. Cartographic Data Base (CDB) Utility Program incl. User's Reference Manual plus CDB's for U.S. Counties and nations of the world	____	1,100.00	$735.00*	____
2. User's Reference Manual	____	6.00		____
3. POLYVRT Replacement Tape	____	110.00		____
ASPEX				
1. ASPEX Interactive mapping program incl. User's Reference Manual	____	$1,200.00	$800.00*	____
2. User's Reference Manual	____	5.00		____
3. ASPEX Replacement Tape	____	120.00		____
INPOM TO BE AVAILABLE IN 1977				
1. INPOM Interactive mapping program incl. User's Reference Manual and a CDB for nations of the world	____	$1,200.00	$800.00*	____
2. User's Reference Manual	____	5.00		____
3. INPOM Replacement Tape	____	120.00		____

Computer programs and data bases for use on IBM computers are distributed on unlabeled 9-track magnetic tape recorded in EBCDIC mode (odd parity) at 800 BPI. Logical record length is 80 and physical record length is 8,000.

If a non-IBM computer has been listed on side two, the program to be sent will be,
(1) a version which has been converted by others for use on the computer specified, or
(2) a standard IBM version plus a list of users with similar hardware who have expressed a willingness to assist others in conversion.

For further information concerning the current availability of programs for specific non-IBM hardware, contact the Laboratory.

(over)

Those requesting a program must complete the following:

CONTACT PERSON to receive future program changes and announcements (sent free of charge):

Name ______________________ Title ______________________

Organization ______________________ Phone ______________________

Address ____

____

HARDWARE INFORMATION

Computer to be used: ______________ Manufacturer______________ Model______________

TERMS OF AGREEMENT

It is the policy of the Laboratory for Computer Graphics and Spatial Analysis, Graduate School of Design, Harvard University, to make its programming systems available only under the following conditions.

1. Neither this software, its documentation, nor adaptations thereof shall, except with prior written consent of the Laboratory, be sold, leased or otherwise distributed in any form to any individual, business entity, academic institution or governmental body whatsoever.
2. Upon acceptance of these terms and conditions, as indicated by signature of an officer having authority to enter into such agreements, the Laboratory grants the recipient a royalty free, non-exclusive license to use the subject material at a single computer facility.

I, an authorized official, understand and accept the stated conditions for use of the material to be provided by the Laboratory.

(Signature) ______________________________ (Date)______________

(Name, printed)______________________________

(Title) ______________________________

(Organization) ______________________________

(Address)______________________________

____

(Telephone) ______________________________

PAYMENT

Checks should be made payable to Harvard University, Lab for Computer Graphics.

Total Amount: $ ______________ Check Enclosed: ______________ Purchase Order #: ______________ Date: ______________

Orders should be sent to: The Lab for Computer Graphics, 520 Gund Hall, Harvard University, Cambridge, Mass. 02138.

SHIP TO:

Name ______________________

Organization ______________________

Address ______________________

____

BILL TO:

Name ______________________

Organization ______________________

Address ______________________

____

CUT ON DOTTED LINE

Order Form

Publications

Publication Distribution — General Information

Postage and Handling

Publications are sent as third class mail, printed matter via surface mail unless other arrangements have been made in advance. Price estimates for air mail shipment of specific items to another country will be sent on request. Alternatively, additional funds to cover the cost of air mail shipment may be enclosed with an order. Any funds in excess of the amount required will be refunded.

Telephone Orders

Telephone orders will be accepted from organizations within the U.S. A purchase order number is required. Orders may be placed during the hours of 9 30 AM to 4 30 PM by calling area code (617) 495-2526.

Payment

Payment should accompany an order unless a purchase order number has been included. Payment must accompany all orders of less than $5 00. Orders to be shipped ouside the U.S. must include advance payment in the form of a bank draft payable in U.S. dollars.

AUTOMATED CARTOGRAPHY PAPERS	Quantity	Price	Amount
1. Summary Report		$ 9.00	
2. Summary Graphics		$ 9.75	
3. Computer Equipment and Programs		$ 5 00	
4. Symap Interpolation Characteristics		$ 4.20	
5. St Louis Region Case Study		$ 19.75	
6 Kansas City Region Case Study		$ 6.75	
7. Montreal Region Case Study		$ 3.75	
8. Connecticut Case Study		$ 3.00	
9. Southern New England Case Study		$ 3 50	
10. Puget Sound Region Case Study		$ 2 00	
11. California Region Case Study		$ 3.00	
12. Boston Region Case Study		$ 3.25	
13. The Use of Computer Graphics in Planning		$ 1.00	
14. Maps a la Computers		$ 1.00	
15. Computer Cartography		$ 3.00	
16. Thematic Cartography		$ 3.00	
17. Management by Computer Graphics		$ 1.00	
18. Color In Art		$ 11.00	
19. Manifested Destiny: A Graphic Account of the Settlement, etc.		$ 2 00	
20 Programming for Transportability, etc.		$ 2 50	
21. Computer Cartography: World-Wide etc.		$375.00	

THEORETICAL GEOGRAPHY PAPERS	Quantity	Price	Amount
1. Concepts and Applications—Spatial Order		$12 00	
2. A Nomagraphic Representation of the Geoid		$ 1 75	
3. Implicit Map Projections in Computer Printouts		$ 2 25	
7. The Geometry of Mixed Hexagonal Hierarchies, etc.		$ 3.00	
8. Energy Flow and Spatial Order, etc.		$ 3.25	
9. The Continent Problem, etc.		$ 2.25	
10. Distances and Land Values, etc.		$ 1.50	
11. Space Straightening and Flattening		$ 1 50	
13. Spatial Order in Fluvial Systems, etc.		$ 3.60	
14. A Nomographic Representation of Trajectories		$ 1.50	
15. A Two-Dimensional Interpolation Function, etc.		$ 1.50	
16. N-Dimensional Spatial Analysis, etc.		$ 4.20	
17. Hyper-surfaces and Geodesic Lines, etc.		$ 1 50	
18. A Note on Stream Ordering and Contour Mapping		$ 2.00	
19 Hierarchical Systems. Cities, Rivers, etc.		$10.00	
20. A Study of Rivers and Other Branching Systems		$ 1 50	
21. Geography and an Existence Theorem, etc.		$ 4.50	
22. A Mathematical Representation, etc.		$ 4 50	
23. Plane Globe Projection, etc.		$18.00	
24. The Descriptive Geometry of a Collection, etc.		$ 2.25	
25. The Law of Travel and Its Application to Rail Traffic		$ 7.75	
26. Minimum Time Paths and the Migration of Arctic Tern		$ 2.00	
27. Notes on the Methodology, etc.		$ 2.75	
28. A Minimum Path Problem Reconsidered		$ 1.50	
29 The Use of the Geodesic Curvature, etc.		$ 1.50	

(over)

	Quantity	Price	Amount
30. Numerical-Geometrical Techniques, etc.		$ 1 50	
31. Some Reflections on Three-Dimensional Geometry Concepts		$ 1.50	
32 Algorithms and Models Based on Projective Transformations, etc.		$ 1 50	
33 Graphical Representation of a Matrix with Applications, etc.		$ 3 75	
34 Some Thought on Optimal Mapping and Coding of Surfaces		$ 1.50	
35 Homological Transformations in Four-Dimensional Space		$ 1 50	
36. A Study of the Movement of a Point on a Plane, etc.		$ 1 50	
37 Set of Equal-Value Antipodal Points, etc		$ 1 50	
38. An Outline for the Theory of Man-Made Space, etc.		$ 4.00	
39 A Structural Taxonomy of Spatial Hierarchies		$ 4 00	
40. The Determination of Fixed-Points, etc		$ 1 50	
41. Law and Order in the Human Lung		$ 4 00	
42. The Hexagon as a Spatial Average		$ 2 00	
43. The Two Dimensional Spatial Organization, etc		$ 2 50	
44. The Sandwich Theorem — A Basic One for Geography		$ 5 25	
45 Relations Between Horton's Laws and Hydraulic Geometry, etc.		$ 3.00	
46 The Two-Dimensional Spatial Analysis, etc.		$ 4.00	
47. A Computer Program for Mixed Hexagonal Hierarchies		$13.00	
48. Fred K. Schaefer and the Science of Geography		$ 1.50	
49 Sparking Potential, Personal Interaction and Social Distance, etc		$ 1 50	
50 Macroscopic Aspects of Metropolitan Evolution		$ 7 25	
51 Notes on the Friction of Distance		$ 1 50	
52 Tabulations of Data on Area, Population, Income, etc.		$19 25	
53. The Geographical Distribution of Income, etc.		$ 1 50	
54. A Description of the 1967-68 United States Income, etc.		$ 1 50	
55 National and Regional Parameters of Growth and Distribution, etc.		$ 2 25	
56 Allometric Growth in Social Systems		$ 1 50	
57. Allometry in Micro-Environment Morphology		$ 2 50	

OTHER PAPERS	Quantity	Price	Amou
1. The Potential of Video Tape Recorders, etc.		$ 1 50	
2. CPED: Computer Programs in Environmental Design		$210.00	
3. Computer Aided Space Planning		$ 1.50	
4. Interactive Mapping of Urban Data		$ 1.50	
5. Interactive Graphics Comes of Age		$ 1.50	
6 Computer Graphics. A Perspective		$ 1.50	
7. How to Talk to a Computer Consultant		$ 1 00	
8 Graphic Technology and the Display etc.		$ 1 00	

PAYMENT

Checks should be made payable to Harvard University, Lab for Computer Graphics.

Total Amount: $________ ________ Check Enclosed:____________ Purchase Order #:__________ Date:______________

Orders should be sent to The Lab for Computer Graphics, 520 Gund Hall, Harvard University, Cambridge, Mass. 02138.

SHIP TO:

Name ____________________

..ganization ____________________

Address ____________________

____

BILL TO:

Name ____________________

Organization ____________________

Address ____________________

____

CUT ON DOTTED LINE

ILLUSTRATION CREDITS

(Where there are two or more illustrations, titles read upper to lower, left to right)

Page	Subject	Data Source	Computer Program	Prepared By
Front Cover	Per Capita Rate of GNP Growth: 1965-1972	WDB-I 1974 World Bank Atlas	CALFORM	David Sheehan Geoffrey Dutton Peter Corbin
2	Dietery Energy Supply: 1974	1975 World Population Data Sheet	CALFORM	Geoffrey Dutton
3	Upper Left: 1970 U.S. Population Density Middle Left: U.S. Land Elevations Lower Left: U.S. State Outlines Upper Right: 1970 SMSA Outlines Middle Right: 1960 U.S. Population Potential Lower Right: % Homes With Oil Heat	U.S. Census Standard Topographic Map County DIME File County DIME File U.S. Census Arthur D. Little, Inc.	DOT-MAP* DOT-MAP* POLYVRT POLYVRT DOT-MAP* CALFORM	Geoffrey Dutton Geoffrey Dutton Nicholas Chrisman Dennis White and Nicholas Chrisman Geoffrey Dutton Nicholas Chrisman
4	Boston 1970 Census Tracts Atlanta 1970 Census Tracts	Urban Atlas Files Urban Atlas Files	CALFORM CALFORM	Denis White Denis White
5	Los Angeles 1970 Census Tracts Chicago 1970 Census Tracts	Urban Atlas Files Urban Atlas Files	POLYVRT POLYVRT	Denis White Denis White
7	Locally prepared SYMAPS	Boston Local Maps	SYMAP	David Sheehan
8	Locally prepared GRID	Boston Local Maps	GRID	David Sinton
9	Massachusetts Tourism Attraction by Towns	Mass. Dept. of Commerce	CALFORM	David Sheehan
10	Montegna Bay SYMAP and SYMVU Contour maps	Symap Self Study Course	SYMAP and SYMVU	David Sheehan
11	1970 Population by Nations (top) Montegna Bay Symap Symvu Conformant maps	United Nations Yearbook Symap Self Study Course	SYMVU SYMAP and SYMVU	David Sheehan Geoffrey Dutton

*Program under development, for further information contact the Laboratory.

ILLUSTRATION CREDITS

(Where there are two or more illustrations, titles read upper to lower, left to right)

Page	Subject	Data Source	Computer Program	Prepared By
13	Diagram of POLYVRT Components Fresno SMSA 1970 Census Tracts	POLYVRT Manual Urban Atlas	POLYVRT POLYVRT	Nicholas Chrisman Denis White
15	Conterminous United States	County DIME File	INPOM	Geoffrey Dutton
17	Selected U.S. Population Densities from "Manifested Destiny"	U.S. Census of Population 1790-1970	ASPEX	Geoffrey Dutton
18	Left: Standard Economic Area (SEA) Center: Standard Metropolitan Statistical Areas (SMSA) Right: Countries of the World	County DIME File County DIME File WDB-I	POLYVRT/INPOM POLYVRT POLYVRT	Nicholas Chrisman Nicholas Chrisman Nicholas Chrisman
Inside Front Cover	U.S. 1970 Population Density	U.S. 1970 Population Census	SYMVU	Geoffrey Dutton James Little

Staff Members of the Laboratory for Computer Graphics & Spatial Analysis

Brian J.L. Berry, Director
Allan H. Schmidt, Assoc. Director
Eric Teicholz, Assoc. Director
Howard T. Fisher, Director Emeritus

Richard Carling, Computer Programmer
Nicholas Chrisman, Sr. Research Analyst
James Dougenik, Research Analyst
Geoffrey Dutton, Research Analyst
Randolph Franklin, Research Assistant
Denis White, Research Analyst
David Sheehan, Consultant

Janice Kostka, Admin. Asst.
Barbara Petersen, Program Distribution

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

GEO-CODIFICACION

APENDICE BIBLIOGRAFICO

JULIO, 1978.

Palacio de Minería   Calle de Tacuba 5,   primer piso.   México 1, D. F.

"Urban and Regional Information Systems for Social Programs", Papers from the 5th Annual Conf. of the URISA, 1967, 207-218

Donald F. Cooke, staff
William H. Maxfield, staff
New Haven Census Use Study

THE DEVELOPMENT OF A GEOGRAPHIC BASE FILE AND ITS USES FOR MAPPING

ABSTRACT: The Address Coding Guide (ACG) of the Census Bureau was first used as the geographic base file. But, weaknesses in the ACG system lead to an experimental mapping data base, Dual Independent Map Encoding (DIME), which employs redundant encoding of map features which allows machine detection of errors.

Details of coding and diagrams make both the theory, file preparation, file editing and actual use in mapping very explicit.

INTRODUCTION

One of the objectives of the New Haven Census Use Study is to evaluate methods of computer mapping of small area data. The Study began experimenting with computer-produced maps in February of 1967. Several methods of linking geographic coordinates to data identified by Census block or by address were studied.

The Census Bureau Address Coding Guide (ACG) was first used for the geographic base file, with the ACG's "block-faces" as reference units for data aggregation and display. However, several weaknesses in the ACG-based system led to the proposal of an experimental mapping data base, (called "DIME" for Dual Independant Map Encoding) employing redundant encoding of map features which allows machine detection of errors. The mapping file thus created has been successfully edited, has proven useful and accurate for mapping, and shows promise for use in many areas of city planning.

The following sections outline the reasons for the choice of redundant encoding, the system's theory, the techniques for developing and editing the file, discussions of the file's use, its cost, and recommendations for its use in other cities.

BACKGROUND

The Census Bureau's Address Coding Guide - part of the mechanism of the mail-out/mail-back Census proposed for 1970 appeared to be the most logical starting point for preparing a geographic base file for computer mapping. The address coding guide is a listing of block faces, which are, as the name suggests, sides or "faces" of blocks. A block face record in the ACG contains the following information:

State—16 009 150 06510 Andrews St. 21 103 2-48

County

Place

Zip Code

Street Name

Census Tract

Census Block

Address Range

In exploring the possibilities of a base file using the coding guide, block face terminals were digitized - that is, coordinates of the endpoints of each block face were measured and recorded, using a semi-automatic "Coordinate Locator" built by the Census Bureau. With the addition of the block face terminal coordinates, the format of the geographic base file became:

State—16 009 150 06510 Andrews St. 21 103 2-48 (x,y) (x',y')

County

Place

Zip Code

Street Name

Census Tract

Census Block

Address Range

Coordinate pair for one block face terminal

Coordinate pair for other block face terminal

However, several difficulties appeared in the use of this file:

1) Digitizing was inefficient, because each cross-street intersection was read eight times.

2) Non-street block boundaries such as shorelines, railroads, and rivers were not included (since there are no addresses on them).

3) Areas of blocks could not be calculated.

4) Winding roads and circular streets could not be described.

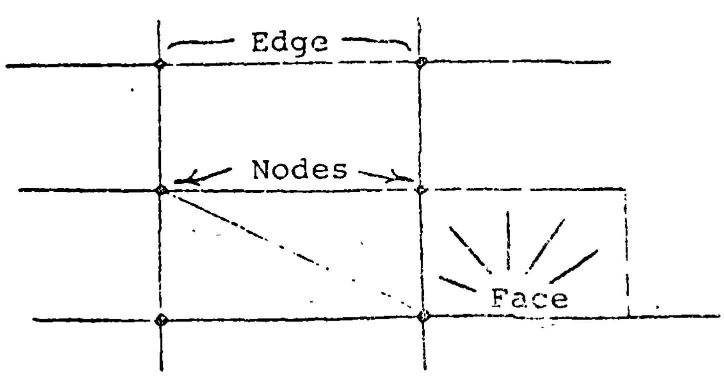
5) Clerical errors were present, and could not be detected by machine edits.

Editing the file proved very difficult and time consuming. As these problems compounded, it became apparent that the Address Coding Guide, though ideal for a mail-out/mail-back Census, could not easily be adapted to computerized mapping.

Therefore, an alternative Mapping Data Base was proposed by James Corbett of the Bureau's Operations Research Division and George Farnsworth, a Census Statistical Intern. The method was tested in New Haven and has yielded a geographic base file which is useful for mapping, and, in addition, may have far wider applicability in traffic analysis, routing of city services, creation of special urban districts, and in the placement of future city facilities. The system is based on graph theory - a branch of mathematics with direct applicability to mapping.

THEORY

The approach suggested by Corbett and Farnsworth is derived from the fact the Metropolitan Maps of the Census

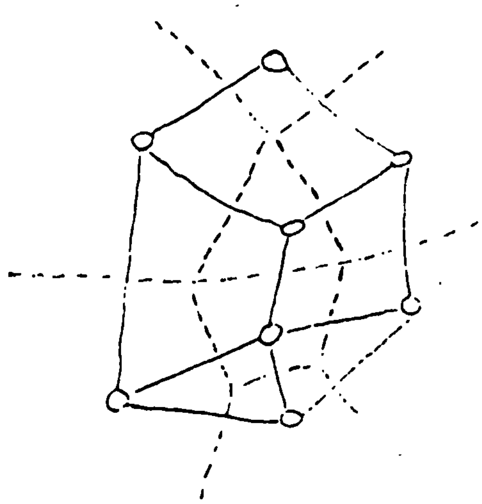


Node = Intersection
Edge = Street Segment
Face = Block

Bureau can be considered mathematically to be linear graphs.

A linear graph is a set of nodes, edges, and faces on a plane, which form a network - such as a street pattern.

Such a network can be described by any of three "incidence matrixes" which define the relationships between the edges and nodes, or nodes and faces, or faces and edges. Whereas only one of the matrixes is sufficient to describe the network, a redundant encoding of the incidence matrixes allows error detection routines to be run on computers until the street network is described perfectly. The network described by one matrix is compared to the supposedly identical network described by another matrix. Differences in the network descriptions can be detected by machine and corrected by hand. Not only the primal network, but also its dual can be edited in this fashion.



Primal Network (solid lines) and its duel (dotted lines)

PREPARATION OF THE FILE

The materials used to create a geographic mapping base file may vary. In New Haven, the materials used were a Metropolitan Map set (which has street names and block numbers labeled) and a city directory to supply address range information for each block. All elements not specifically named on the maps were labeled in a straight forward manner. Nodes were given numerical identifiers and names such as "shoreline" or "Connecticut Turnpike Ramp" or "Mill River Centerline" were assigned to their corresponding line segments.

Node numbering was done sequentially and systematically to facilitate digitizing. Nodes which were numbered included: street intersections, ends of dead-end streets, and places where railroads, shorelines or rivers constituted Census block boundaries and crossed streets or each other. Places where lines on the map curved appreciably were assigned node identifiers also. The criterion for "appreciable" bending of a street depends on the needs of the user, that is, whether he needs very accurate area calculations or wants to reproduce the street network extremely accurately.

Once all nodes were numbered, encoding of the incidence relationships were done in the following fixed length format, each record in the file describing a line segment - or section of a line between two nodes.

Street—Andrews St. 73 129 1 102 1 101 06510 1-25 1 0

Node (1)

Node (2)

Tract (L)

Block (L)

Tract (R)

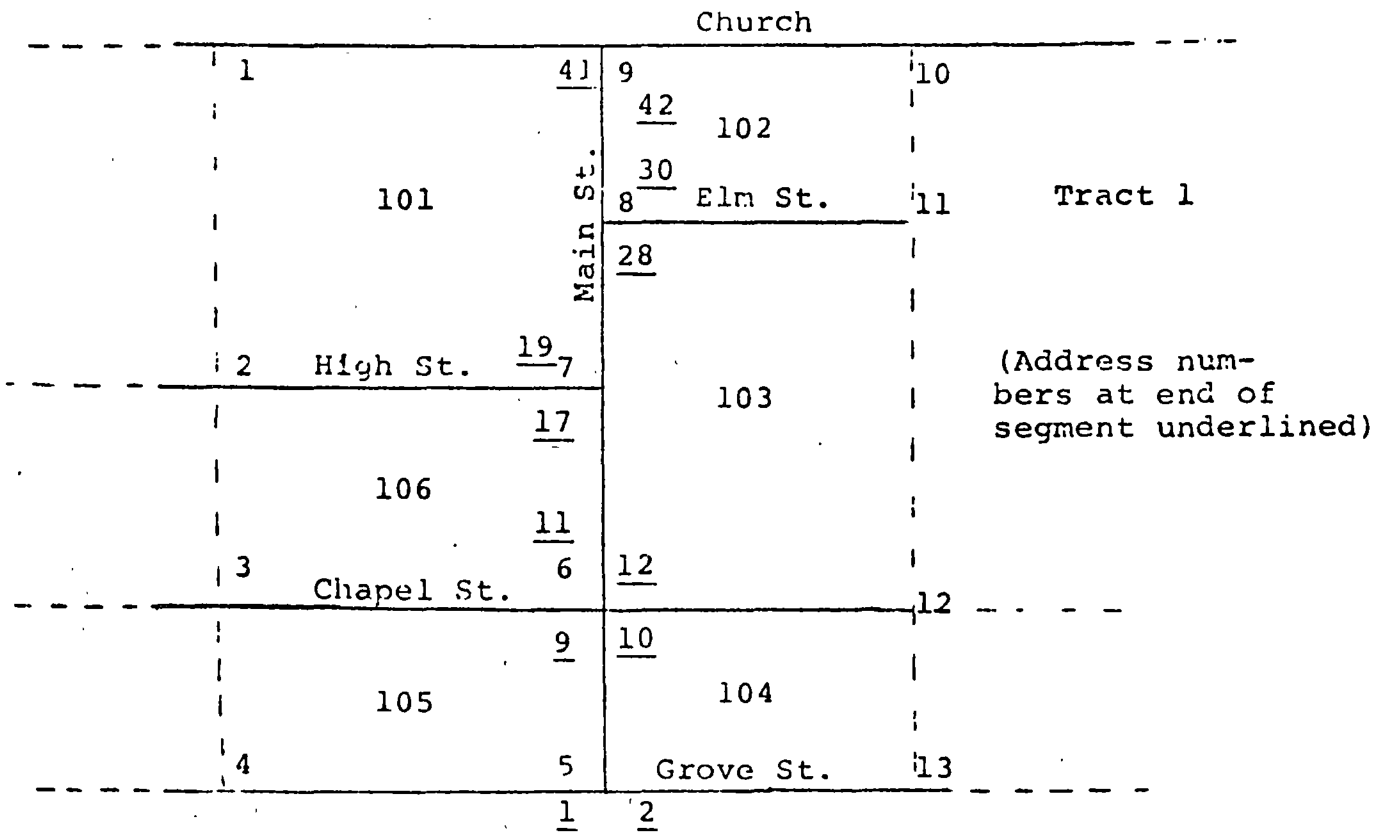
Block (R)

Zip Code

Address Range

Address Code (1 means odd left)

Segment Code (0 means street segment)



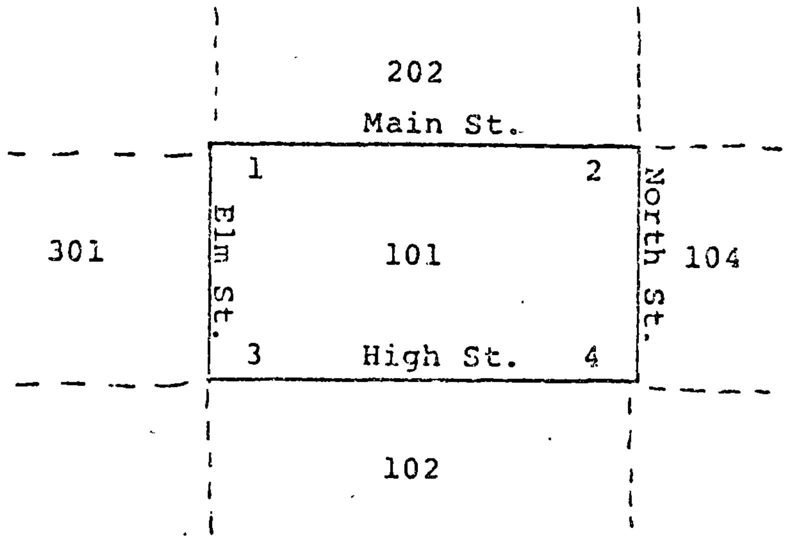
Census Address Coding Guide records for Main Street

Street	Tract	Block	Low Add	High Add
Main St.	1	102	30	42
Main St.	1	103	12	28
Main St.	1	104	2	10
Main St.	1	105	1	9
Main St.	1	106	11	17
Main St.	1	101	19	41

DIME Street Segment Records for Main Street

Street	Node	Node	Tract	Block	Tract	Block	Low Add	High Add	
Main St.	5	6	1	105	1	104	1	10	*
Main St.	6	7	1	106	1	103	11	18	*
Main St.	7	8	1	101	1	103	19	29	*
Main St.	8	9	1	101	1	102	30	42	*

* = Odd left



Primal edit sorts for boundary segments of block (101)

Street	Node[1]	Node[2]	Block[L]	Block[R]
Main St.	1	2	202	101
Elm St.	1	3	101	301
High St.	3	4	101	102
North St.	4	2	101	104

The edit attempts to traverse the boundary by placing the segments in order by matching on Node numbers.

Street	Node[1]	Node[2]	Block[L]	Block[R]
Main St.	1	2	202	101
North St.	2	4	104	101
High St.	4	3	102	101
Elm St.	3	1	301	101

Block L and Block R have been switched if the nodes had to be reordered to effect a traverse. (This occurs for all streets except Main)

In this example, the traverse is successful, and the block numbers have been encoded correctly. Note the relationship of the node numbers and block numbers.

Node 1 and Node 2 are the node numbers defining the line segment. Their order is such that Node 1 is at the low address end of the segment. Tract L and Block L are the Census track and block numbers to the left of the segment as seen by an observer standing at Node 1 and looking towards Node 2. The address range is derived from the city directory and the Address Code indicates odd numbered left side of the street or odd numbered right side of the street. Note that this is an inclusive address range for both sides of the street while each record in the Census Address Coding Guide gives address range information for one side of the street. Segment Code is a number assigned to indicate whether the record is a street, railroad, shoreline, or other type of segment. It was also used to indicate problems encountered by the encoding clerks. Coordinates of the nodes defining the segment may be incorporated into the record, or may be carried separately in a list which can be indexed by node number.

EDITING THE FILE

Street segment records were keypunched, and the first of several computer edits was run. This edit sorts out for each block all segment records associated with the block, then further sorts out all boundary segments. Note that a boundary segment can be identified as having different block numbers in the Block L and Block R fields.

An interior street such as a dead end will have the same block number on each side. The edit program attempts to "traverse" the boundary of the block by linking the segments together in order by matching node numbers. Success or failure messages are written out, and are used to guide clerks in correcting whatever errors or omissions are present. The edit program is run and corrections and additions are made until no traverse failures are reported.

The second edit is a test of the accuracy of the encoding of the Block L - Block R information. Again a traverse of the boundary is made, this time with the Block L and Block R fields being switched if the nodes had to be reordered to effect a traverse. The program then checks to see if the block number of the block being checked appears entirely in the Block R or entirely in the Block L fields. If this condition fails, there is an error in encoding the block pair. It may be of interest to note that if the coordinates of the nodes are digitized and inserted into the segment records by matching on the node numbers, then the areas of the blocks can be calculated as soon as a block can be traversed. The sign of the calculated area will indicate the direction of the traverse. Knowing the direction of the traverse enables one to know which of the Block L or Block R fields should contain the block number of the block being edited.

The implication of this is that the clerks who perform the encoding process need not even be able to tell their right hand from their left. However, in New Haven we had excellent clerical help, and very few errors in the Block L - Block R encoding were encountered.

A final machine edit sorts out all segments records associated with each intersection. The module two sum of the block numbers associated with any node should be zero. If this is not the case, an error message is written.

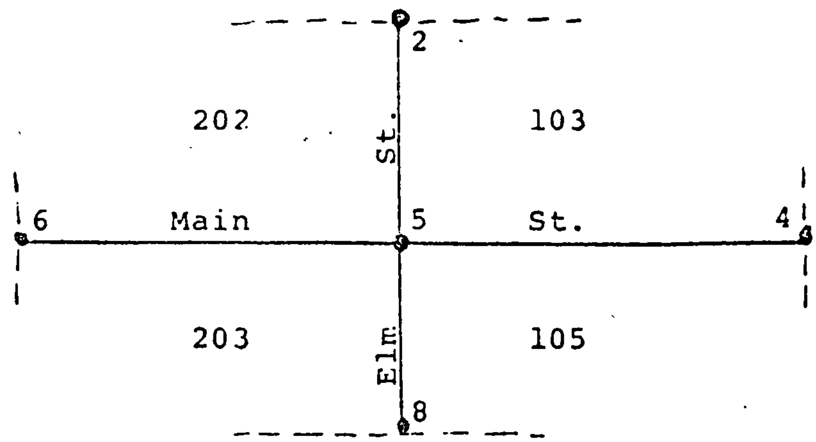
Another useful edit is a visual check of the street network as reconstructed from the segment file using digitized coordinates. An almost equivalent edit is to plot mechanically the node numbers on an overlay to the source map. However, the former technique exploits the pattern recognition abilities of the human eye, and erroneous digitizing is easily discovered.

Edits which we hope to run as soon as possible will check the address range information for internal consistency. Any gaps or overlaps of address ranges will be detected and corrected by hand. When the address range checks have been completed, the data base file will be complete for New Haven and West Haven, Connecticut. We hope sometime to expand the directory to include the rest of the S.M.S.A. The final mapping file will include calculated block centers and areas.

USE OF THE FILE IN NEW HAVEN MAPPING

The file without coordinates is usable for address coding, and for many functions which require knowledge of the contiguity relationships of blocks. Mr. Farnsworth mentioned some of these last night: studying the "neighborhood" concept, moving or mosaic averaging, aggregation of statistics of any area by specifying the boundary of the area, districting, and avoiding disclosure of confidential information. We have coordinate information in New Haven, which allows the further ability of graphic display, area and density calculation, calculation of distances along streets, and aggregation of statistics for districts specified by coordinates, for instance, a count of all people living within one mile of City Hall.

Our primary interest lies in graphic display, however, and experiments indicate that we have a very flexible tool for this purpose. In testing the applicability of the mapping file, we have produced maps of housing units per acre from block statistics using SYMAP, a printer mapping program de-



Dual edit sorts for all segment records associated with intersection (5)

Street	Node[1]	Node[2]	Block[L]	Block[R]
Main St.	6	5	202	203
Main St.	5	4	103	105
Elm St.	8	5	203	105
Elm St.	5	2	202	103

If the intersection being edited (5) appears in the Node field, multiply Block L by -1. Otherwise, multiply Block R by -1.

Street	Node[1]	Node[2]	Block[L]	Block[R]
Main St.	6	5	202	-203
Main St.	5	4	-103	105
Elm St.	8	5	203	-105
Elm St.	5	2	-202	103

Next, all the block numbers are added:

```
   202
  -103
   203
  -202
  -203
   105
  -105
+  103
------
     0
```

If all segments are encoded correctly, the sum will be zero.

veloped by Howard Fisher of the Harvard Laboratory for Computer Graphics. The SYMAP options we have tested require that the boundary of the map section to be plotted be described by coordinate pairs of vertices in order around the boundary. Retrieving this information from the Geographic Base File was easily done and can be done by machine. Block center coordinates, and the area of the blocks (needed for the density calculation) were calculated directly from the segment file.

We are testing another printer mapping package - MAP 01, for use with the mapping file to map Census data. We are also writing a mapping program to plot data maps on CRT, pen plotter, or Geospace plotter output devices. The figure of merit of any maps produced by the Census Use Study will be the usefulness, if any, to local planners who use Census data. Hopefully some of the techniques we develop in New Haven will be significant enough and cheap enough to be adopted by the Census Bureau for their publication problams sometime in the future.

EXTENSION OF THE SYSTEM TO OTHER CITIES

In recommending adoption of this system in other cities, we would modify the foregoing procedure as follows: we would stress the importance of accurate source materials. Much time was lost trying to read detailed information off maps where the scale (1" = 800') made the features ambiguous. We would suggest using 1" = 400' or even larger scale maps, preferably ones with address numbers on the maps. We would further suggest that two address ranges be carried for each segment record - one for each side of the street. Working in this manner from maps such as the Sanborn series would enhance the address coding accuracy of the file (which we consider to be the weakest part of the New Haven file).

Once the source maps are encoded, machine plotting of the street segments, block numbers, and street lables allows rapid, accurate and inexpensive production of work maps at any desired scale. Aside from the size of the file, there is no technical reason why the system of network encoding cannot be extended to the parcel level, where parcel boundary segments, rather than street segments, are the units of the directory. An experiment to test this is being run on a four tract section of New Haven, primarily to produce a highly accurate address file to test the address coding capability of the New Haven mapping file. Extension of the system to the parcel level is straightforward, and not as time-consuming as might be expected. A file such as this, matched to tax records, could yield extremely accurate, detailed, land use information. In addition, the file could yield the percentage of land that is residential, commercial, street and even the area of the sidewalks.

FILE DEVELOPMENT COSTS

One final note on the New Haven street segment base file - the cost. Even using very generous pay scales, and overestimating computer and digitization usage, we found it difficult to push the cost estimate for the New Haven file over $3,000.00, including writing of the edit programs. This figure works out to about 2 cents per person, or $3.00 per block. We cannot extrapolate the cost to other cities at the present time, but we do feel that this method yields an inexpensive and accurate geographic base file with broad application as a tool for urban analysis.

R. G. Loomis and J. J. Lorenzo
IBM Corporation
New York Scientific Center

EXPERIMENTS IN MAPPING WITH A GEO SPACE PLOTTER

ABSTRACT: The Geo Space Plotter is a computer mapping system using wet photographic process. It is versatile in that images are stored on disk, transferred to tape for reading and restored on disk for updating or overlay with additional image components to produce a photographic image of a special purpose map.

Using the New Haven Census Use Study data, it proved to be capable of plotting block numbers in irregular blocks and base maps depicting streets, tract boundaries and railroads. Experiments in block shading for density studies were less promising.

Future experiments in equipment and programming include use of a "Fast ALPACA" system allowing construction of entire image before plotting computer storage is used, and the development of a faster dry photographic process.

INTRODUCTION

Within the preceding year the IBM New York Scientific Center initiated a study to investigate advanced computer applications in the area of urban affairs. At a very early state of this study it became quite clear that one of the distinguishing characteristics of many urban planning and operational activities is their significant dependence on spatially organized data - a fact which requires no further substantiation for those present at this meeting. As a consequence, it was decided to undertake exploratory studies directed to what were judged to be three major aspects of computer processing of spatial data; namely

1) techniques for storing and manipulating spatial data within a computer,

2) systems which would permit planners and other users to directly interact with such data by means of online graphic terminals,

3) methods for providing hard copy graphical represent-

ALGORITHMS FOR THE REDUCTION OF THE NUMBER OF POINTS REQUIRED TO REPRESENT A DIGITIZED LINE OR ITS CARICATURE

DAVID H DOUGLAS AND THOMAS K PEUCKER

University of Ottawa/Simon Fraser University, British Columbia

ABSTRACT. All digitizing methods, as a general rule, record lines with far more data than is necessary for accurate graphic reproduction or for computer analysis. Two algorithms to reduce the number of points required to represent the line and, if desired, produce caricatures, are presented and compared with the most promising methods so far suggested. Line reduction will form a major part of automated generalization

Lines from maps and photographs are recorded numerically for cartographic manipulation to facilitate their reproduction at different scales and projections, and to allow map compilation with other geographic data bases. Usually lines are approximated by straight line segments, and end points of which are recorded by a pair of co-ordinates in either polar or orthogonal measure. The other more important methods by which lines are recorded are chain encoding and skeleton encoding. Chains approximate lines by a sequence of end to end vectors, where the length and direction of the vectors are selected from a fixed, usually four or eight, number of possibilities.[1] Skeleton encoding is directed more at recording closed areas or polygons by filling the area with circles or rhombi of different sizes. The lines forming the boundaries are recorded by implication.[2] The conversion of graphic data to computer readable numerical forms is effected with a co-ordinate digitizer, a bit plane scanner or an automatic line follower. A co-ordinate digitizer converts a pointer's location on a table to x-y values which can be written on punched cards or magnetic devices. Polar co-ordinate digitizers, which consist of a slide in a rotating anchor head, record a radius and an angle from a base vector. Another digitizing device consists of a pointer suspended from a pair of retracting wires which activate potentiometers. Conversion of values in one recording co-ordinate system to another can be performed easily with small computer programs.

Drum scanners superimpose a vast and very fine grid over the document to be digitized recording a "yes-no" or "on-off" value for each cell location, depending on whether that cell covers a line or not. A trade-off is introduced between the fineness of the mesh, implying more computer processing time to reduce the data to forms which are easily handled, and coarseness of the image recorded. On the other hand, the mesh density, being dependent on hardware, is fixed at the time of manufacture and is usually set to be somewhat smaller than the minimum line width. In all cases, the reduction of a bit plane scan, in which lines are represented by clouds of cells containing numerous discontinuities, to chain or vector encoded lines, is a complex process requiring processing time and resources which could only be described as being quite substantial.

With a co-ordinate digitizer lines may be recorded in point mode, time or increment automatic modes. Lines recorded in point mode are effectively generalized by the operator who subjectively selects points which best approximate the line to the degree he desires. This presumes, among other things, that he is his own customer. Point digitizing is extremely tedious however, and is unsuitable for anything but the simplest data sets, such as the generalized

David H. Douglas is a lecturer in Geography at the University of Ottawa and is presently on a study leave at Simon Fraser University, Burnaby, British Columbia. Thomas K. Peucker is an Associate Professor at Simon Fraser University, Burnaby, British Columbia

MS submitted June 1973

[1] H. Freeman, "On the Encoding of Arbitrary Geometric Configurations", *Institute of Radio Engineers; Transactions on Electronic Computers*, Vol. EC-10, 1961, pp. 26–268.
[2] J. R. Pfaltz and A. Rosenfeld, "Computer Representation of Planar Regions by Their Skeletons", *Communications of the ACM*, Vol. 10, No. 2, February 1967, pp. 119–122 and 125.

outlines of counties or census tracts. Most coordinate x-y digitizers on the market possess, as options, time or increment automatic recording modes. Points are recorded automatically in a given time interval, or after the cursor has moved a preset distance along the x and/or y axis. The prime limiting factor on the speed of recording is the speed of the output device. Magnetic tape transports which record up to 300 characters per second are commonly available, allowing up to 20 or 30 points to be recorded each second. To record coastlines, contour lines, or other lines of high frequency oscillation it is evident that the minimum speed required, given the speed at which an operator can follow a line, is in the order of 5 to 10 points per second, which effectively eliminates paper tape and punched cards as output media. Digitizing onto magnetic tape has more than its share of problems, primarily because there are no foolproof means to ensure the data are correctly recorded at the time of digitizing, and because of the inordinately frequent occurrence of non-confirmable digitizing errors, such as line ends which should, but do not meet, ... lines recorded twice and so forth. The editing procedures necessary are time consuming and clumsy. These problems have been met by elaborate on-line procedures where a mini-computer interfaced to the digitizing table oversees the whole operation, checks and double checks the data recorded, closes loops and signals when it senses a great many errors, such as cursor movement too fast to be accurate.[3]

All digitizing methods, except perhaps for the possible exclusion of point digitizing on a co-ordinate digitizer, record, as a general rule, far more points than necessary to reproduce the line on most graphic devices, even at the scale and resolution of the original line. The elimination of data representing unnecessary points, such as duplicates,

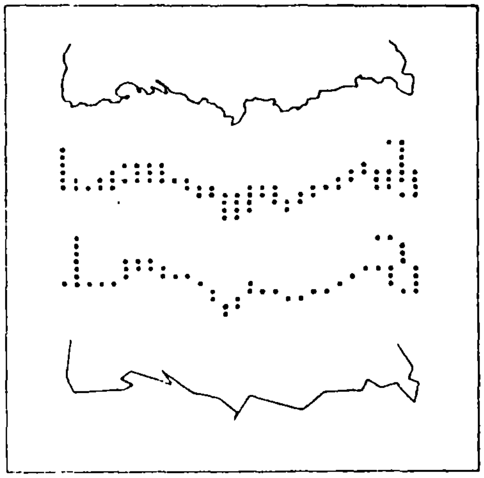


Figure 1. Line represented by 140 points on the plotter and the printer, and the same line represented by 25 points.

and points along a straight line, can be of significance, simply because of the diminished storage requirements. As well, the operating speed of many spatial analysis programs and the plotting speed of many graphic devices are related inversely to the number of points to be processed or plotted. Reduction of a line by elimination of unnecessary points representing it assumes a more positive advantage if the line is to be abstracted or caricatured purposely, if the scale of reproduction is to be smaller, or if the output device, such as some Cathode Ray Tube plotters, has a cruder resolution than represented by the original digitized line. Lines which have a higher frequency of oscillation than can be represented within the resolution capability of the graphic device become fuzzy and weak.[4] Figure 1 illustrates line data at the resolution of recording, its reproduction on the computer printer, and the reproduction of a greatly generalized version of the line. Given the crudeness of the printer as a graphic device it is evident that the simplified version of the line is preferable to the unsimplified one mainly because of the elimination of most

[3]A. R. Boyle: Computer Aided Compilation, Hydrographic Conference, Ottawa, January, 1970.
[4]and are similar in effect to the data clouds recorded by a bit plane scanner.

of the double lines and data clouds. Since this line was better represented by 25 points than it was by the original 140 obviously some computer pre-processing was justified.

There have been a great many approaches suggested and algorithms programmed to reduce the number of points required to represent numerically recorded lines. Some of these are in regular use within planning agencies and cartographic units. Not all of the methods have been exhaustively tested to measure or judge their cartographic usefulness and there have been few, if any, studies to compare the methods with each other. The methods can be classed broadly into the categories of: elimination of points along the line by one or more of a multitude of criteria; approximation of the line with a mathematical function; and deletion of specific cartographic features represented by the line. Of these categories, it would seem that the last one would come closest to duplicating the task as performed by an experienced cartographer as he generalizes.

The cartographer attempts to maintain the character and overall impression of an empirically defined, or hand drawn line by selective deletion of some of the details. A fjorded coast is represented by only a few of the actual number of fjords, a delta by only a few of the actual number of channels and so forth. The automation of this approach would rely therefore on the ability to program the computer to recognize specific cartographic features. One attempt is based on an interactive computer program which has the ability to "learn" from the actions of an operator.[5] The operator generalizes a line plotted on a cathode ray screen by signaling the deletion or maintenance of points. As the computer "learns" from what the operator selects it attempts to recognize similar features on its own. This system at its present level of development concentrates on the angular and length relationships of a very small number of segments, but the number of possible ways to represent a single simple class of feature, such as a peninsula, is simply staggering. This interactive system, therefore, represents but a small step towards the solution of a fantastically complex problem.

The second group seeks to approximate the points along a line with mathematical functions. This can be done for the whole line at once or it can be done in some piece-wise order taking a small number of connected points at a time. There are several different methods fitting into the latter category. One developed by A. R. Boyle for the Hydrographic Survey of Canada (1972) computes a first order least squares line through a fixed number of points and then steps forward in that direction by a predetermined distance. Two other approaches begin by defining the ends of segments as averages of a fixed number of points along the line. Koeman and Vander Weiden[6] suggest taking the mean while Jancaitis and Junkins[7] take the distance weighted centroid. When these central points are joined the results simulate a piece-wise approximation with functions of the first order. It must be mentioned, however, that the stated purpose of Jancaitus and Junkins was to smooth and not necessarily to reduce the line.

The resulting data sets of extracted functions are economical in terms of storage

[5]Andrew H. Clement, "The Application of Interactive Graphics and Pattern Recognition to the Reduction of Map Outlines", Master's Thesis, University of British Columbia, 1973.
[6]C. Koeman and F. L. Vander Weiden, "The Application of Computation and Automatic Drawing Instruments to Structural Generalization", *Cartographic Journal*, Vol. 7, No. 1, June 1970, pp. 47–49.
[7]James R. Jancaitis and John L. Junkins, *Mathematical Techniques for Cartography*, Final Contract Report for U.S. Army Engineers Topographic Laboratory, Fort Belvoir, Virginia, Contract No. DAAK02-72-C-0256, February 1973, pp. 15–20.

space required, but are relatively time consuming in the processing stage. The greater the number of points, the more costly and complex the operation These functions reproduce lines which are typically much smoother than the lines they represent In the main they are probably much better suited for smoothing than reduction and have to be considered of limited value for generalizing. Functions extracted in a piece-wise fashion tend to under-represent erratic curves and over-represent smoother curves Methods which look for central tendencies are inclined to depress the effect of extreme points. Unfortunately, these are often the very points which give character to the line.

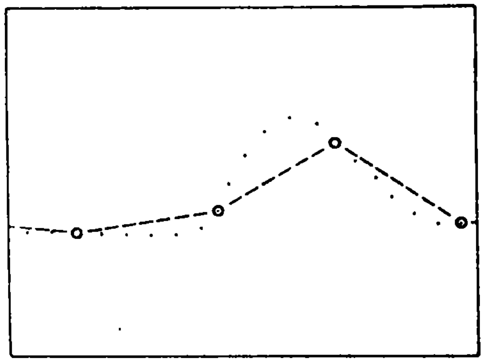


Figure 2 Line reduction by the selection of every sixth point.

fore it cannot represent a significant step towards automated generalization.

Of the group of methods which eliminate points, some concentrate on the points which are to be deleted while others are directed towards selecting those points which are to be maintained. The algorithms directed at deleting points are usually the simplest. In the case of data recorded by time-automatic digitizing a simple test to drop those closer than one resolution unit can eliminate a large percentage of the points recorded. This method can be extended by purposely decreasing the numerical resolution or by establishing a threshold distance. Points closer than this distance to neighbours are dropped.[8] For chain encoding a simple compression on the basis of consecutively equal vectors can also result in significant savings. This can be extended as well for other types of encoding by dropping points whenever the direction of the line is not changed through a threshold angle by the segments subtended on it The underlying purpose of these methods is to eliminate wasted data space but since the line plotted after this kind of processing would look very much the same as it would be-

The simplest and most often used method of line reduction is to delete all but every n^{th} point along the line where n is a fixed integer based upon the desired degree of reduction[9] The method does not require much in the way of computing resources and it furnishes acceptable results if the digitizing was extremely dense. The primary disadvantage is the frequent elimination or misrepresentation of important features along the line such as promontories, indentations, sharp angles and so forth. A secondary limitation is that straight lines are still over-represented. These shortcomings are made obvious in Figure 2.

The alternative to deleting points is to select them. In the special case of monotonically increasing lines (for instance, just one value of "y" for every "x"), crests and troughs may be selected. The obvious disadvantage here is the omission of points where there is a change of direction but which nonetheless are not crests or troughs. For irregular planar curves, the problem is more difficult. Jarvis con-

[8]W. R. Tobler, "Numerical Map Generalization", *Michigan Inter-University Community of Mathematical Geographers, Discussion* Paper No. 8, Department of Geography, University of Michigan, January 1966.
[9]Experimental Cartographic Unit, Royal College of Art: *Automatic Cartography and Planning*, London, Architectural Press, 1971.

verts the Cartesian to polar co-ordinates and then looks for crests and troughs.[10] This is useful for curves which can be made monotonic by this conversion, but, as for Cartesian measure, the solution cannot be considered general.

One alternative to line generalization which seemed to hold conceptual promise was that method provided by the German firm, A. E. G. which supplied the Experimental Cartographic Unit with its GEAGRAPH 4000 plotter, and was described by T. Lang in 1969.[11] This method was reported as producing acceptable results but was eventually rejected as a general purpose technique by the Experimental Cartographic Unit on the grounds that it required far too much computer time for the on-line processing system being operated at the time. The objective of the procedure was to delete points if they were found to lie within a tolerance distance of a straight line segment being tested to represent a portion of the line. From one representative point it constructs straight lines to subsequent points until one point between the representative point and the sub-point is further away from the line linking the two than a pre-set tolerance value. As soon as this condition is satisfied, the point before the sub-point becomes a new representative point and the procedure is repeated. The method gives acceptable results in the case of smooth curves but it does not detect the best representative points on sharp curves and the results are particularly unsatisfying where sharp angles are numerous.

The methods proposed in this paper are based on a concept somewhat similar to the pre-set tolerance ideas described by Lang but concentrates rather on the selection of points rather than on their deletion

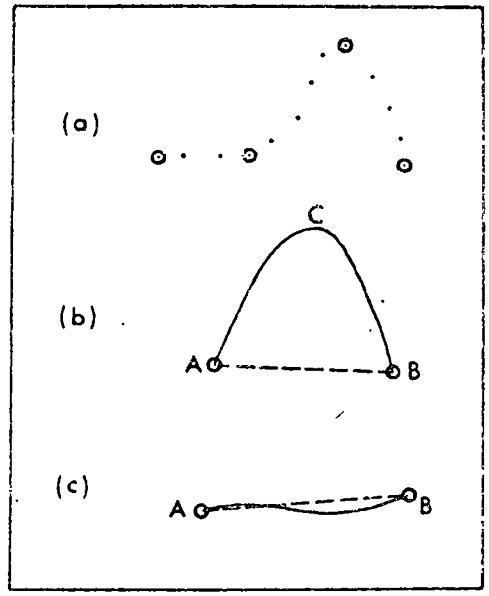


Figure 3. Subjective selection of representative points.

Approaches to a computerized solution to many problems begin with an examination of the way one would solve them subjectively. Consider the line represented by points illustrated in Figure 3(a). One might choose the encircled points as these which represent the original line to our own requirements of accuracy. Perhaps the reason we would select these points and not others might be illuminated by examining the simpler situations in Figure 3(b) and (c). Starting with the obligation to begin with the end points, the question might be: "Why would there be a compulsion to insert a point C in (b), where no such compulsion would exist in (c)?" The perpendicular distance of C from the segment A-B may provide a clue. This suggests that an arbitrary maximum distance could be established. If no point along the line is further than this distance from the straight line segment connecting its end points, then the straight line seg-

[10] C. L. Jarvis, "A Method for Fitting Polygons to Figure Boundary Data", *The Australian Computer Journal*, Vol 3, 1971, pp. 50–54.
[11] T. Lang, "Rules for Robot Draughtsmen", *Geographical Magazine*, Vol. XLII, No. 1, Oct. 1969, pp. 50–51.

ment will suffice to represent the original line. If this condition is not satisfied, then another point along the curved line must be selected and the same test would be carried out with the new segments The next question is: "What point along the curved line should be selected to become the end point of the two new straight segments created?" The obvious answer is the furthest point from the straight segment Although it is possible that this point may be embedded in a long smooth curve, it is more likely that it is the apex of a relatively sharp angle. As well, this point has already been identified as a result of the distance search, therefore, the benefits associated with its selection far outweigh the possible attraction of selecting some other representative point. In the case of closed loops, where the first and the last point do not define a line then the maximum perpendicular distance from the segment is replaced with the maximum distance from the point. The same process would be repeated with the new segments created until the maximum distance requirement is satisfied for all straight segments.

Two different procedures embodying these principles have been encoded in FORTRAN IV and tested. In addition Method 2 has been encoded as a recursive function in ALGOL W.[12]

Method one begins by defining the first point on the line as an anchor and the last as a floating point. These two points define a straight segment. The intervening points along the curved line are examined to find the one with the greatest perpendicular distance between it and the straight line defined by the anchor and the floater. If this distance is less than the maximum tolerance distance, then the straight segment is deemed suitable to represent the whole line. In the case where the condition is not met, the point lying furthest away becomes the new floating point. As the cycle is repeated the floating point advances toward the anchor. When the maximum distance requirement *is* met the anchor is moved to the floater and the last point on the line is reassigned as the new floating point. The repeat of this latter operation comprises the outer cycle of the process. The points which had been assigned as anchor points comprise the generalized line.

Method two is exactly the same as method one except that note is taken of all points which have been assigned as floaters on previous inner cycles. These are stacked in a vector. After the anchor point is moved to the floating point, the new floating point is selected from the top of this stack, thereby avoiding the necessity of re-examining all the points between the floater and the end of the line. This procedure usually results in the selection of a slightly greater number of points than Method 1, but takes approximately 5 per cent of the computing time and is thought to produce better caricatures. This method can also be thought of as taking a logically hierarchical approach to line reduction. On one cycle extreme points are selected and these tested to see if they suffice If they do not, intermediate points are taken and the same question asked about each of the two new segments produced, and then each of the four new segments are examined, ... and so on as if in a branching tree. Each branch is terminated when the offset tolerance criterion is satisfied

To enable valid comparisons four separate subroutines were written on the basis of the procedure described by Lang. One was an exact duplication of that procedure while the other three were combinations of two incorporated modifications.

---

[12]Andrew H. Clement, "The Application of Interactive Graphics and Pattern Recognition to the Reduction of Map Outlines", Master's Thesis, University of British Columbia, 1973.

The program Lang describes starts by assigning the first point as the anchor and third as a floater. The second is tested to see if it lies within tolerance distance of the segment defined by the anchor and the floater. If it does, the fourth is assigned as the floater and the second and third are examined and so on. The first floating point defining a segment which does not allow all intervening points to satisfy the tolerance criteria causes the anchor to move to the point before the floating point. Since selection of the point immediately before the floating point has no cartographic justification, the first modification of the procedure has the anchor point move to the point furthest from the segment. The reasoning behind selecting the furthest point is that it is the one most likely to subtend a sharp angle and would therefore have the best chance of properly representing the line. The second modification attempts to cut computing time by avoiding unnecessary repeated calculations of distance. From Figure 4, it is clear that in most cases, the sum of the distances a + b + c is greater than the greatest distance that P_1, P_2, or P_3 lies from the segment P_0P_4. In other words, if a + b + c is less than the tolerance distance then d also would be less than the tolerance. Only one distance, rather than all of the intervening ones, has to be calculated on each cycle. The inner cycle, intended to find the point lying furthest from the segment, is invoked only in the cases where the accumulated total is greater than the tolerance. Positive and negative accumulations are kept in separate registers to avoid subtractions from their absolute magnitudes in the case of double curves. The maintenance of these running registers is particularly useful when series of points lie along straight lines.

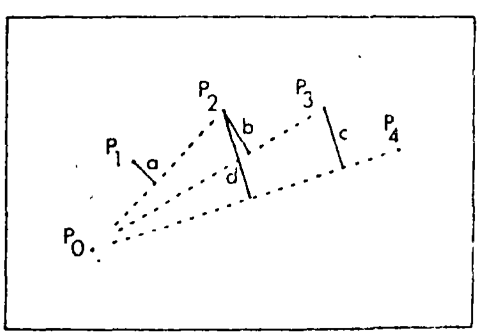


Figure 4. Running registers of accumulated offset distances.

The first modification which attempts to select a point which is more rationally defined than simple convenience, has the expected result of approximately doubling the number of points selected and the processing time required to isolate them. The second modification definitely reduces the time required to process a given line, especially if a great many points are deleted because they lie along relatively straight segments.

All procedures were tested and compared, both for their ability to remove unnecessary points, that is with the offset tolerance set to be less than the resolution of the plotting device (Figure 5), and their ability to produce caricatured representations (Figure 6). All were judged to produce satisfactory results for simple line reduction, however the versions of the A. F. G. procedure without the modification to pick the furthest point from the tested segment did not produce satisfactory caricatures because of the tendency to omit and cut corners. The methods presented in this paper were tested with substantial data sets and found to be operationally suitable both for simple reduction and in the production of satisfactory abstractions (Figures 5 and 6).

Detailed comparisons in computing time required for each sub-routine were made on the basis of a three inch square and a three inch diameter circle, each made up of 4000 points evenly distributed along its periphery. It was felt that the square would give ample opportunity to demonstrate the power of each routine in the

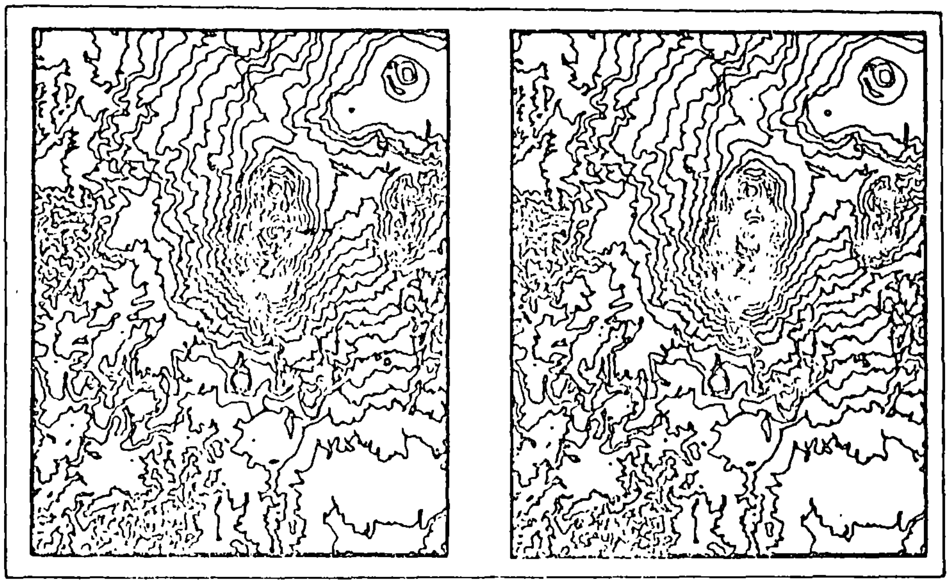


Figure 5 Contours plotted from the original digitized, unduplicated at .001 resolution, 41,311 points (left), and from 7,782 points (right) reduced by Method 2 with a tolerance set to half the resolution of the plotter The reduction procedure added 16.5 seconds to the 64 seconds required to read and write the data to plot the map on the left The images may be compared with a simple stereoscope

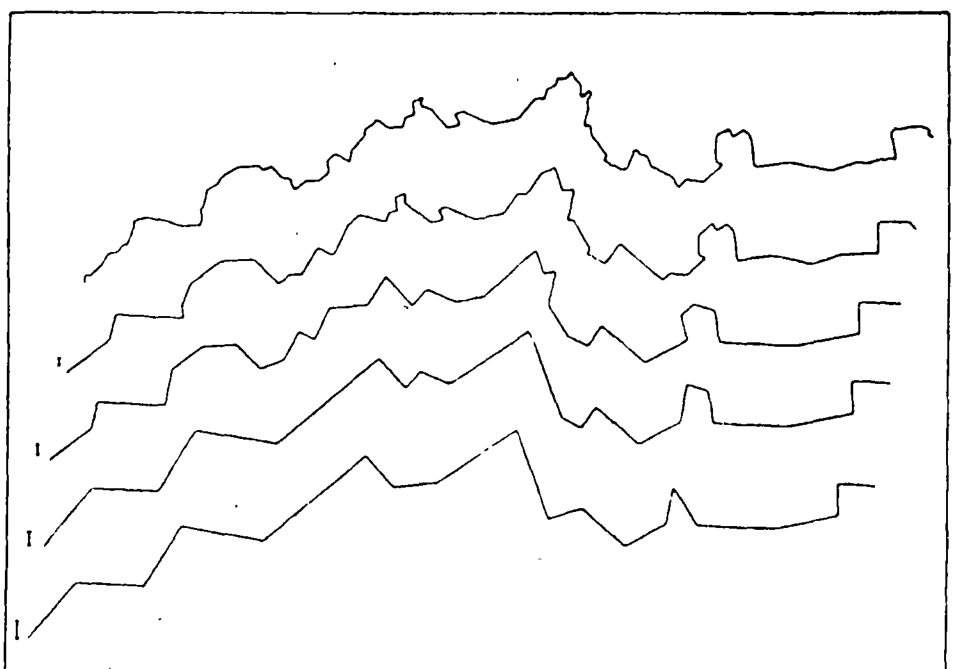


Figure 6 Line reduced and caricatured by Method 2 The tolerance value employed is shown to scale at the left of each caricature which was reduced from the original data represented by the top line

TABLE I

PROCESSING TIME REQUIRED TO REDUCE A 3″ CIRCLE AND A 3″ SQUARE MADE UP OF 4000 POINTS EACH EVENLY SPACED ALONG THE PERIMETER, TO THE NUMBER OF POINTS INDICATED WITH THE GIVEN OFFSET TOLERANCE.

	Offset Tolerance (inches)											
SQUARE	.001		.005		.01		.05		.1		.5	
4000 points	Points	Time	Points	Time	Points	Time	Points	Time	Points	Time	Points	Time
A.E.G. procedure	5	88.4	5	88.6	5	87.3	5	88.3	5	86.4	5	86.9
A.E.G. plus Mod. 1	5	87.8	5	88.9	5	88.6	5	88.3	5	89.8	5	113.9
A.E.G. plus Mod. 2	5	22.6	5	44.5	5	46.0	5	45.8	5	44.5	5	44.7
A.E.G. plus Mods. 1 and 2	5	22.8	5	22.5	5	22.4	5	22.9	5	23.1	5	31.4
Method 1	5	.7	5	.7	5	.8	5	.8	5	.8	5	.7
Method 2	5	.6	5	.6	5	.6	5	.5	5	.5	5	.6
CIRCLE												
4000 points												
A.E.G. procedure	88	5.5	40	11.1	29	14.9	14	32.6	10	42.4	5	97.1
A.E.G. plus Mod. 1	171	10.4	77	20.4	55	28.6	25	60.4	18	84.8	5	109.4
A.E.G. plus Mod. 2	88	5.4	40	10.8	29	15.4	14	32.7	10	41.6	5	92.2
A.E.G. plus Mods 1 and 2	171	10.6	77	21.5	55	30.0	25	60.9	18	87.8	8	170.2
Method 1	127	25.1	56	10.4	39	7.5	18	3.7	13	2.7	6	1.0
Method 2	129	1.8	65	1.5	33	1.2	17	.9	17	1.0	5	.6

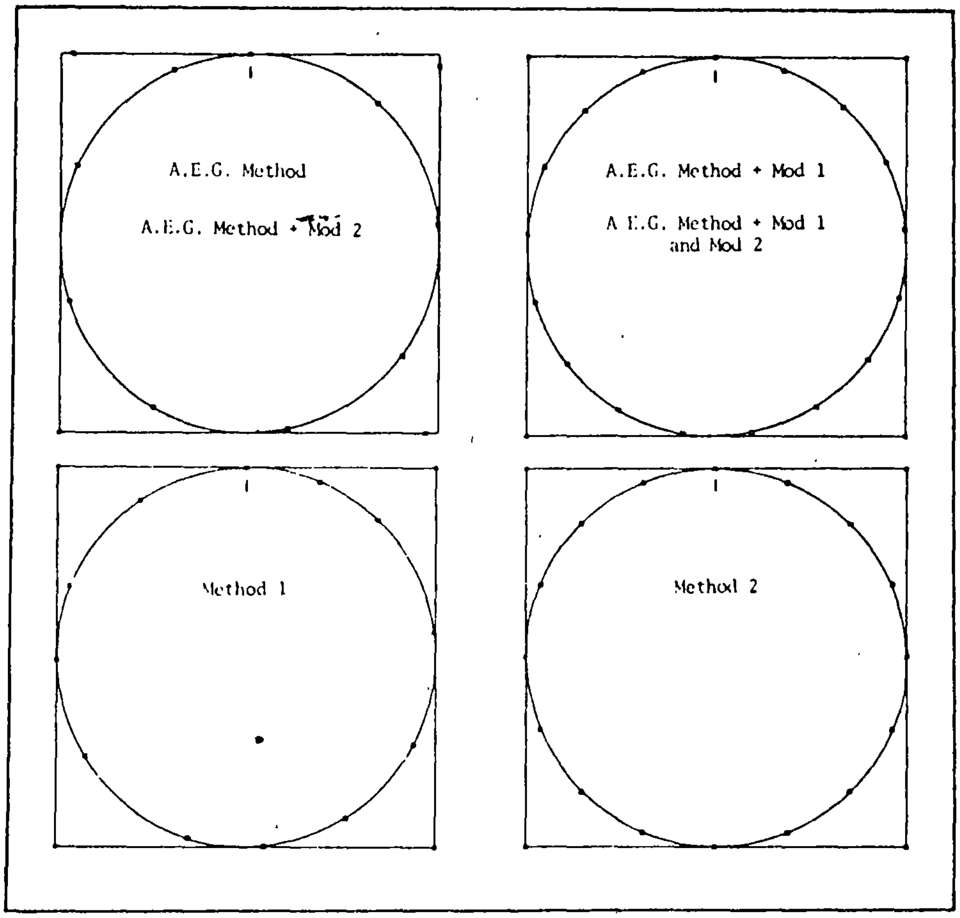


Figure 7 Plotted results for a circle and a square each made up of 4,000 points around its perimeter. The dots on the boundaries indicate the selected points by the indicated procedures. The tolerance value of 1 inch is illustrated to scale at the top centre of each diagram.

case where many points along a line are to be deleted, whereas the circle would be more representative in sinuosity of drawn or empirically recorded lines as far as this timing test was concerned. Table 1 presents results by the established offset tolerance in number of points selected and in seconds central processing unit time required for the reduction procedure (I. B. M. 370/155 under O/S MVT).

Figure 7 illustrates plotted results for a 1/10" offset tolerance with the points selected by each routine marked with a heavy dot. The fact that the routines selected different points and different numbers of points is not unusual or unexpected and similar differences occur in the case of sinuous lines. The shortcoming of the unmodified A. E. G. procedure is evident in the case of points selected to represent the square, ... which were just less than one tolerance unit from the corners for all but the first and last point.

Each routine selected five points to represent the square and each took approximately the same time regardless of the tolerance, except with the second modification. In this case the first step off the straight line caused the inner cycle to be invoked which found the new anchor point on the first iteration. More iterations were required for the other tolerance limits.

In the case of the circle an increase in tolerance limit caused a decrease in the number of points found to represent it for all methods. Those methods which push the examination segment ahead of the anchor points, that is the A. E. G. method with none, one or two modifications, take longer to perform as the offset tolerance is increased This therefore comprises the main reason that they have to be considered unsuitable in an operational context. These procedures are fastest if they are unable to delete any points, because in such cases they would have to examine only one point to come to that decision. On the other hand if a great number of points are found to be deletable, increasingly large inner cycles are invoked for each advance of the floating point. The two methods presented in this paper work in entirely the opposite way and are fastest in the case of lines which are found to be representable with a smaller number of points. Presumably this is the object of the effort. In all cases Method 2 is seen to take as little as 1 per cent of the time required by the others.

The prime purpose of the routines discussed here is to reduce the number of points required to represent a line and to produce abstractions, or caricatures of the line in cases where these will suffice. In many cases these could be considered to be perfectly adequate generalization procedures While the scope of generalization is no doubt much broader, line reduction by means such as those described here, represents an important portion of that topic.

RÉSUMÉ. Règle générale, les méthodes numériques enregistrent des lignes avec beaucoup plus de données qu'il n'est nécessaire à la reproduction graphique précise ou à la recherche par ordinateur. L'auteur présente deux algorithmes pour réduire le nombre de points nécessaires pour représenter la ligne et produire des caricatures, si désiré, et les compare aux méthodes les plus prometteuses suggérées jusqu'ici La réduction de la ligne constituera une partie importante de la généralisation automatique.

ZUSAMMENFASSUNG. Alle Digitalisierungsmethoden zeichnen in der Regel Linien mit bedeutend mehr Daten auf als fur eine genaue graphische Wiedergabe oder für eine Computeranalyse notwendig sind. Zwei spezielle Rechenverfahren zur Reduzierung der Punktezahl, die zur Darstellung einer Linie benotigt werden und die auch falls erwunscht Verzerrungen produzieren, werden vorgestellt und verglichen mit den bisher am meisten versprechenden Methoden. Die Linienreduzierung wird eine grosse Rolle in der automatisierten Generalisierung spielen.

RESUMEN Todos los métodos digitales, como regla general, registran líneas que tienen mucho más datos que los necesarios para la reproducción gráfica correcta o para el análisis por computadora. Se presentan dos algoritmos para reducir el número de puntos necesarios para representar una línea y si se desea, producir caricaturas; estos se comparan con los métodos más pometedores sugeridos hasta ahora La reducción de líneas formará gran parte de la automatización en general

Research Notes and Comments

Choropleth Maps Without Class Intervals?

W. R. Tobler

It is now technologically feasible to produce virtually continuous shades of grey by using automatic map drawing equipment. It is therefore no longer necessary for the cartographer to "quantize" data by combining values into class intervals. As a simple illustration an automatic line plotter can be programmed to draw lines virtually any distance apart (Fig. 1). Thus, one can obtain any desired destiny of inked area to white area. For example, if the geographical data, symbolized by z, are normalized to lie in the range from zero to one, then an appropriate spacing of orthogonal lines of width w is given by

$$s = (w/z^x) \cdot [1 + (1 - z^x)^{1/2}].$$

Here an exponent ($x \approx 1.4$) of z has been chosen to approximate the nonlinear response of the human eye [*13*]. The units of the spacing s are those of w. Comparable equations are easily obtained for dashed lines or for dotted maps. Automatic equipment that produces grey areas by modulation of light intensities can produce even more refined displays. There thus results a choropleth map on which the visual intensity is exactly proportional to the data intensity. Since no class intervals have been introduced, there is no quantization error [*1, 2, 11*]. The much studied [*5, 6, 7, 8, 9, 10, 12*] and difficult problem of optimum class intervals is thus circumvented.

Some cartographers will still wish to group their data into classes and will argue that they do this in order to simplify or enhance the map for the user. This, then, is a problem of map generalization and not necessarily one of chosing class intervals. I assume that, by definition, a generalization of a choropleth map is another choropleth map, not a smooth surface as might be built up from modelling clay.

A choropleth map can be generalized in at least four ways. First, by combining

W. R. Tobler is professor of geography at the University of Michigan.

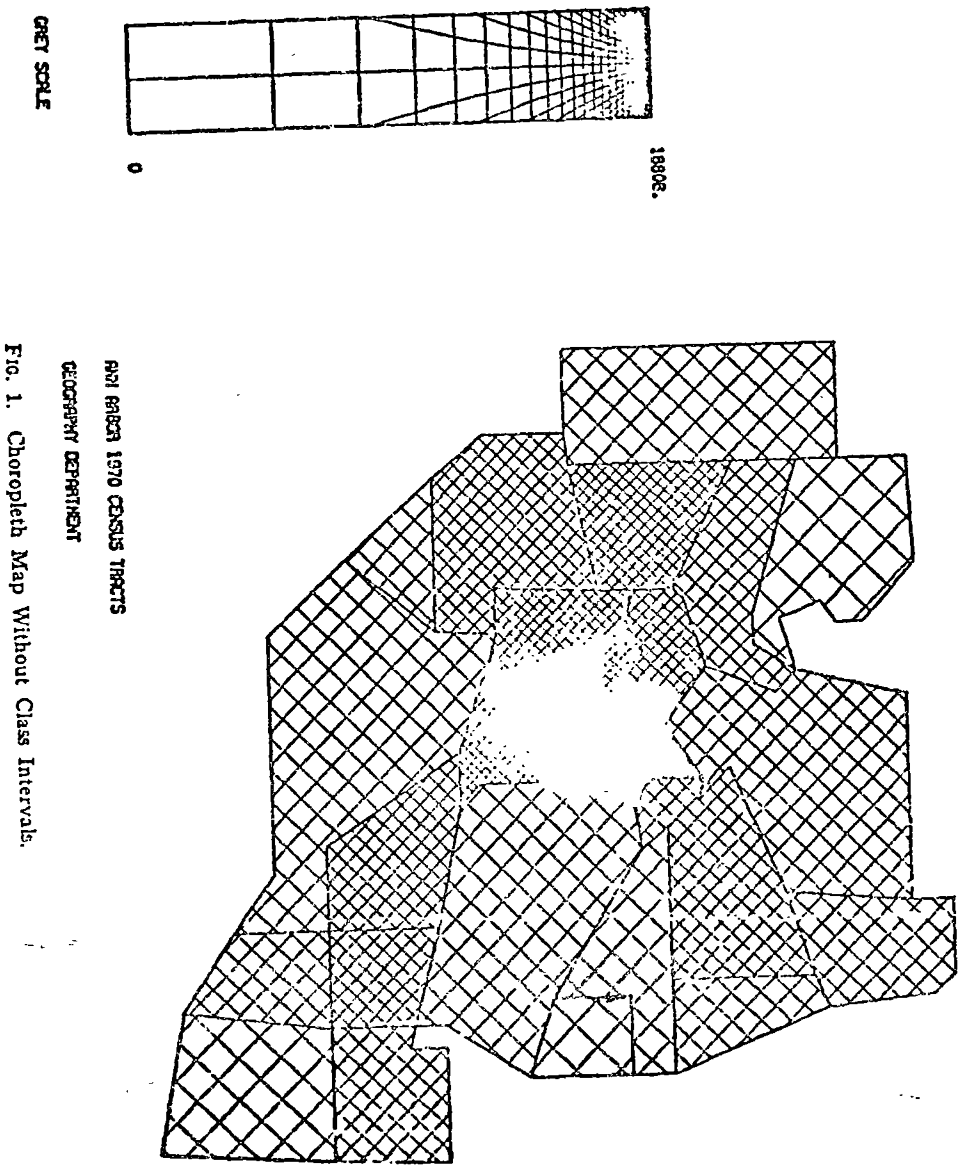


Fig. 1. Choropleth Map Without Class Intervals.

adjacent areal units (units that have similar values are made into new units whose value is some combination of the earlier values, or small units are eliminated, reducing the resolution of the data); secondly, by simplification of the boundaries of the areal units; thirdly, by changing the value of each unit in some manner which depends on the values of the adjacent units [17]; fourthly, by quantizing the data more coarsely, i.e., by picking large class intervals, or by using some nonlinear class intervals. As an analogy, one may consider the ways of generalizing a topographic surface: by varying the spacing of the sampling points, by smoothing with a filter, or by choosing a larger or variable contour interval. The latter method is of course comparable to the choosing of class intervals for a choropleth map. Enlarging or modifying the contour interval, without simplifying the contours, does not necessarily improve the map, but may enlarge the quantization error. Taking samples at larger or different spatial intervals is equivalent to filtering using a different two-dimensional Dirac comb [3] and thus is a type of smoothing. The more general case is to modify the values of each unit in a controlled manner that depends on the values of adjacent units [14, 15]. This is easily achieved by performing the choroplethic equivalent to taking a two-dimensional weighted moving average, as, for example, in binomial filtering [4, 16, 17]. Either smoothing or emphasis can be obtained in this manner.

The main argument in favor of using class intervals seems to be that their use enhances readability. This at least is the assertion. It seems equally plausible that this is also true of the three alternate map generalization methods cited above. If the assertion is in fact valid why then is grouping of greys into classes not also (e.g., in addition to spatial filtering) used to enhance aerial photographs, or television? Formulae for the optimal quantization of images are in fact given in the literature on picture processing, where the main difficulty stems from the conversion of continuous images into discrete signals, or relates to transmission band-width and noise reduction studies [1, 2, 11]. Typically, a large number (2^6) of levels are recommended, compared with the small (2^2 to 2^3) number used for choropleth maps, though somewhat fewer levels are required for equally satisfactory colored pictures. It is thus not clear why the theory for pictures should differ from the theory for choropleth maps, since both have visual information processing as their ultimate objective. Presumably, both have some domain of validity, but the limits need further exploration.

LITERATURE CITED

1. Andrews, H. C. *Computer Techniques in Image Processing.* New York: Academic Press, 1970. Pp. 136–51.
2. DiGiuseppe, J. L. *A Survey of Pictorial Data Compression Techniques.* Technical Report No. 16, Contract N. DA-49-083 OSA-3050, University of Michigan, Ann Arbor, University of Michigan, 1969.
3. Goodman, J. W. *Introduction to Fourier Optics.* New York: McGraw-Hill, 1968. Pp. 14–25.
4. Holloway, J. L. "Smoothing and Filtering of Time Series and Space Fields." *Advances in Geophysics,* 4 (1958), pp. 351–89.
5. Jenks, G. F. "Generalization in Statistical Mapping," *Annals,* Association of American Geographers, 53 (1963), pp. 15–26.
6. Jenks, G. F. and M. R. Coulson. "Class Intervals for Statistical Maps," *International Yearbook for Cartography,* 3 (1963), pp. 119–34.
7. Jenks, G. F. and F. C. Caspall. "Error on Choroplethic Maps," *Annals,* Association of American Geographers, 61 (1971), pp. 217–44.
8. Mackay, J. R. "An Analysis of Isopleth and Choropleth Class Intervals," *Economic Geography,* 31 (1955), pp. 71–81.
9. Monmonier, M. S. "Contiguity-Biased Class-Interval Selection," *Geographical Review,* 62 (1972), pp. 203–28.
10. Robinson, A. H. and R. D. Sale. *Elements of Cartography.* 3rd ed. New York: Wiley, 1968. Pp. 164–70.
11. Rosenfeld, A. *Picture Processing by Computer.* New York: Academic Press, 1969. Pp. 20–28.
12. Scripter, M. W. "Nested-Means Map Classes for Statistical Maps," *Annals,* Association of American Geographers, 60 (1970), pp. 385–93.
13. Stoessel, O. C. "Standard Printing Screen System," *Proceedings,* 1972 Fall Convention, American Congress on Surveying and Mapping, Columbus, Ohio, 1972, pp. 111–49.
14. Tobler, W. R. "Numerical Map Generalization," *Michigan Inter-University Community of Mathematical Geographers Paper No. 11,* Ann Arbor, University Microfilms No. OP-33067, 1966.
15. ———. "Of Maps and Matrices," *Journal of Regional Science,* 7 (1967), pp. 275–80.
16. ———. "Geographical Filters and Their Inverses," *Geographical Analysis,* 1 (1969), pp. 234–53.
17. ———. "CHOROS: A Computer Program to Apply Linear Neighborhood Operators to Choropleth Maps." Ann Arbor, April, 1972. Available from the Geography Program Exchange, Computer Institute, Michigan State University, East Lansing, Michigan 48823, U.S.A.

Cartographic Data Structures

Thomas K. Peucker and Nicholas Chrisman

ABSTRACT. Efficient and flexible data structures are important to the development of computer mapping. Most current data banks are characterized by 1) structures which are convenient at the input stage rather than at the stages of use within computer programs, 2) separate and uncoordinated files for different types of geographic features, and 3) a lack of information about neighboring entities. The term "neighborhood function" may be used to indicate the relative location of a geographic entity and is a concept which is involved in all three of these characteristics. Ongoing research on data structures had led to work on the GEOGRAF system for encoding planar data and the GDS ("Geographic Data Structure") for encoding three-dimensional surfaces. Both involve data manipulation between the digitizing stage and the actual use of the data within computer mapping programs.

INTRODUCTION

A series of ongoing research projects are concerned with efficient and flexible data structures for geographic and cartographic analysis. The three main points of concern in the research can be summarized as follows:

1) In most cartographic data banks, the arrangement of the data is guided by the input stage. In other words, little manipulation of the data is performed after the data have been input into the system from maps.

2) Cartographers and computer scientists have made few attempts to combine different types of cartographic information, for example, height with other cartographic features. Therefore, the different types of cartographic entities are stored in different files and it is usually extremely time-consuming to combine them.

3) The data structure is usually very simple and lacks one facet in particular which is essential for much geographic and cartographic analysis—an indication of the relative location of a geographic entity, i.e., the position of a geographic entity with respect to its neighboring entities.

Thomas K. Peucker is associate professor, Simon Fraser University, Burnaby, Canada. Nicholas Chrisman is senior programmer, Laboratory for Computer Graphics and Spatial Analysis, Harvard University, Cambridge, Massachusetts. The research for this paper has been supported by ONR contract N00014-73-C-0109 and the Laboratory for Computer Graphics and Spatial Analysis, Harvard University.

These three points may be abbreviated with the terms flexibility, comparability, and topology. This paper will characterize types of existing geographic and cartographic data systems for planar and three-dimensional surfaces, especially with respect to these three points. The paper will also describe attempts which have been made by the authors to produce data systems which eliminate some of the problems of existing ones. The term "neighborhood function" will play a major role throughout the paper and will therefore be explained in more detail in the following section.

NEIGHBORHOOD FUNCTION

When asked for the location of a city, we will give the location with respect to a river, a seacoast, a pass, a neighboring larger city, or other feature. Rarely will we use the geographic coordinates of longitude or latitude, nor will we use map coordinates. We are taught in elementary geography that the geographic coordinates will tell us little about either the large-scale (site) or small-scale (situation) characteristics of a place. Similarly, if I describe my position on a piece of terrain, I will not use my map to determine my location within the UTM-grid; rather I will look for nearby relief features (peaks, rivers, slopes, roads) as orientation characteristics.

In contrast, when a geographic data bank of any kind is created, it utilizes some kind of absolute coordinate system. Usually neither the geographic evaluation at the time of digitizing nor the mapping system used with the data allow the inclusion of such cartographic features as streets, rivers, and roads which would give us an indication of relative location.

While the human user can be aided in his orientation on a map through overlays or map comparison, the computer has difficulty in determining relative location. If the relative location in terms of the closest points for each of, say, 5,000 points is to be calculated, the program literally has to compute every point's distance to every other point. Indeed, some widely-used programs do this computation several times within one program run.

Some indication of the relative location of a geographic feature can be very useful. This neighborhood relationship will be referred to as a "neighborhood function." It can be expressed in different ways: as an *explicit* or *implicit function*, or as a discrete function in the form of a *table.*

The *explicit function* can be a polynomial or trigonometric equation set for a discrete grid of surface patches which give the form of the surface at each point within the patch. Typical for this approach is the work of Junkins, et al. (1973). Two-dimensional spline functions also fall into this category (Holroyd and Bhattacharyya, 1970). A much more frequent way of defining a neighborhood is by the explicit function in the form of a sort routine which finds the closest neighbors. This is done in various interpolation algorithms to produce a regular grid of points (Shepard, 1968; Heiskanen and Moritz, 1967). The computations increase close to the square of the increase in the number of points, since the search has to be repeated for every point and all points, or at least a large number of them, must be processed each time.

This search procedure also applies to the case of planar surfaces where neighboring polygons must be found. For example, when contiguity constraints are imposed in problems of factorial ecology and other regional correlations, all polygon points must be searched to find those which are in common for a pair of polygons. Again, the problem increases in complexity according to the square of the number of the items being searched.

The *implicit function* expressing neighborhood relationships is usually a function that describes the coding structure of the geographic entities. One very good case is described in Rosenfeld (1969) for different types of neighborhood relationships within a regular grid in which the point P_{ij} has the four neighbors $(i+1, j)$, $(i, j+1)$, $(i-l, j)$, $(i, j-1)$ and it has the eight neighbors $(i+1, j)$, $(i+1, j+1)$, $(i, j+1)$, $(i-1, j+1)$, $(i-1, j)$, $(i-1, j-1)$, $(i, j-1)$, $(i+1, j-1)$.

Neighborhood relationships in the form of *tables* are very rarely used. This type records the neighborhood function by "pointers" indicating neighboring geographic entities. For example, a structure which is built on the basis of Thiessen polygons could have such a structure by simply having the labels of the neighboring points accompany the record of each point. The most widely-known structure of this type is the DIME file of the U.S. Census which encodes line segments, the names of the polygons to the left and right of each line segment, and the names of the two nodes at either end. The neighborhood relationships used in the DIME development are derived from the discipline of topology (Cooke and Maxfield, 1967).

Neighborhood relationships will be discussed in greater detail in the analysis of various existing data structures. Two types of geographic data bases will be discussed: those defining planar surfaces and those defining three-dimensional surfaces. For both types, a summary of their historical development will be presented, and it will be shown that, although presently at different stages of development, these two types can be treated as special cases of one topological data structure.

DATA STRUCTURES FOR PLANAR SURFACES

Types of Structures

The types of geographic entities on planar surfaces are points, lines, and area-enclosing lines or polygons. The latter are perhaps the most frequently encoded feature in geographic data systems.

The simplest data base system for planar surfaces is that of encoding *entity by entity* with little or no regard for entity overlaps or adjacencies (Fig. 1). In other words, every polygon in a polygon system is encoded and stored without any regard for contiguous polygons, and lines are encoded without regard for the fact that they may intersect or merge with other lines. The results of such an encoding are "sliver lines" (duplication of lines in slightly different positions). These sliver lines are confusing and unaesthetic and, hence, it is virtually impossible to do anything directly with such a data base except an extremely coarse graphic image.

To go beyond the use of such data for the production of coarse images, editing must be performed. This alternative has been attempted in several cases, one being the MAP-MODEL system (Arms, 1970). The editing in this system is guided by the assumption that every segment has to be represented twice except for segments on the outer boundary. For each segment, the editing program sorts through all remaining segments to find its complement (the identical line of the neighboring polygon). Those segments for which it has not found a complement are tagged to indicate potential errors to the user.

To overcome some of the limitations of independently encoded entities, systems have been developed based on a common *location dictionary*. This dictionary contains the coordinates of every boundary point on the map. Polygon boundary lists are then compiled which consist of the labels (location numbers) of these boundary points (Fig. 2). Line information can be handled in the same way. Programs based on this structure include CALFORM from the Laboratory of Computer Graphics and Spatial Analysis. Other programs have subroutines to convert this point dictionary structure to the simple entity-by-entity line list described above. The data can then be used in programs such as SYMAP (Laboratory for Computer Graphics and Spatial Analysis) which are compatible with the entity-by entity structure. Programs have also been developed to simplify data input through automated polygon identification (Douglas, 1973).

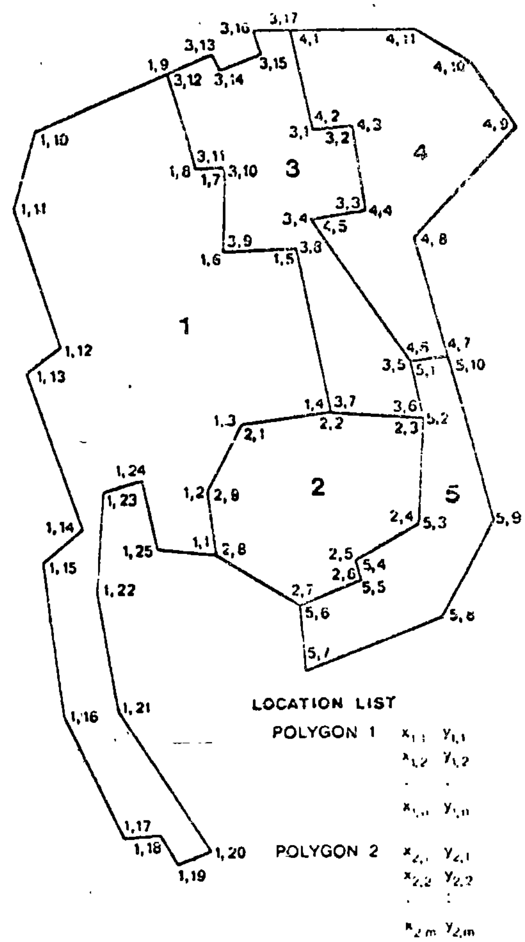


Fig. 1. The simplest encoding is areal entity by areal entity. In this system, most points are recorded twice and some (e.g., $P_{1,4}$; $P_{2,2}$; $P_{3,7}$) three times. A point does not necessarily have identical coordinates in all recordings.

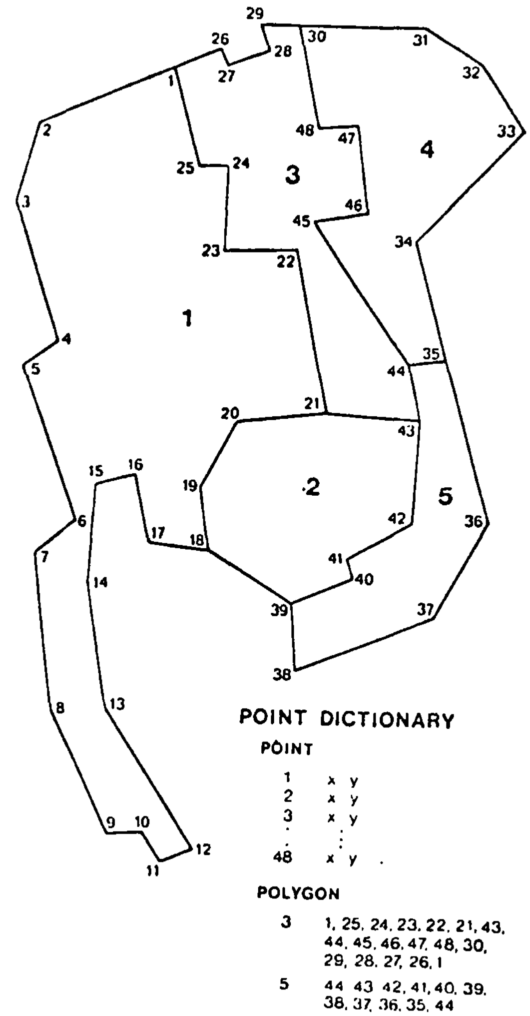


Fig. 2. In the second type of polygon system, the individual points are encoded only once and are stored in the Point Dictionary. For every polygon, a Polygon Boundary List is then established.

The point dictionary data base has the advantage that sliver lines do not occur. However, the problem of neighborhood relationships is not handled any better than with the entity-by-entity approach. The search for common lines is no longer according to the coordinates of points but by their labels; this brings us closer to a solution only by a little less computer time. It also creates difficulties. A point dictionary can and will be accessed in an arbitrary order, since there are no restrictions regarding point placement. The standard response to this problem is to make the dictionary core resident. Unfortunately, this will limit the complexity of the map that can be handled in this manner, since all points must be stored throughout the operation of the program. This shortcoming of such sharing of data is augmented by the continued independence of the entities created by the dictionary; instead of n points with their x and y coordinates, there are simply n references to points.

Some of the objections to the ordinary point dictionary approach can be eliminated by formulating an intermediate object between the entity and the points used as an addressing scheme (Nake and Peucker, 1972; Peucker (ed.) 1973). A geographic entity can be created from a *list of line segments;* these segments are, in turn, created from references to the point dictionary. This system allows for easy definition of the entities with a minimum of pointers, but each entity is still independent in the sense that its neighbors are not known. The direction of access is still from entity to location but not the reverse.

All of the data structures described thus far are of limited flexibility and utility because neighborhood relationships are not known. By adding the topological neighborhood function of each element to a data structure, large improvements in flexibility and scope of applications can be realized.

If one is concerned about the memory capacity which is needed for the storage of explicit neighborhood relationships, one might consider a system with implicit neighborhood functions by *modifying the entity form* in the encoding stage. Many existing geographic information systems store land-use data in grids of rectangular cells (Hsu, 1975). However, a serious problem is encountered with such a regular discrete encoding of planar surfaces. According to the sampling theorem, the sampling interval has to be half the size of the smallest features to be encoded (e.g., Tobler, 1969). Hence, either the size of the cells has to be very small to enable encoding of detailed variations (e.g., urban land uses) or a grosser cell size can be used for encoding only the very slowly changing variations (generalized land use). In the case of the small units, a very large volume of redundant information is created in an area of uniform land use and matrix reduction techniques such as run-length encoding (recording in each row only those cells differing from an immediately previous cell) create only physical and not logical compaction (Amidon and Akin, 1970). In the case of the grosser cell size, highly varying land use features are aggregated to a degree which reduces the usefulness of the whole system.

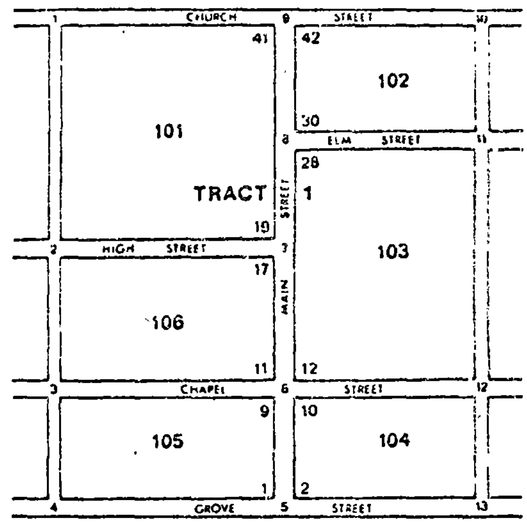
Beyond sampling problems, a grid structure imposes a bias toward specific orientations of features. Diagonals, although physically longer, are given the apparent cell relationships of unit distances along the major axes. The grid structure also creates the illusion of working with a discrete point space, rather than a regular areal partitioning.

Recent studies have developed parameters which determine the degree of inaccuracy of a given sampling mesh (Switzer 1975). In cases where some types of geographic features vary over small areas and others over very large areas, such as the case of statewide land-use patterns, the error is averaged over all features and will affect features with a high-frequency variation more than others.

The variation of the mesh size within the information system would be of only little help. The data management and the manipulation and display routines would become more complex, although probably less than many researchers imagine. Exact figures about the differences cannot be given since no literature about such a system is known to the authors.

Structures Using Explicit Topological Relationships

One of the first known attempts to incorporate explicit topological structure into a geographic data base is the Dual Independent Map Encoding (DIME) system of the U.S. Bureau of the Census (Fig. 3). The DIME files were originally developed as an automated topological error detection



CENSUS ADDRESS CODING GUIDE RECORDS

STREET	TRACT	BLOCK	LOW ADDRESS	HIGH ADDRESS
Main	1	102	30	42
Main	1	103	12	28
Main	1	104	2	10
Main	1	105	1	9
Main	1	106	11	17
Main	1	101	19	41

DIME STREET SEGMENT RECORDS

STREET	NODE START	NODE END	TRACT LEFT	BLOCK LEFT	TRACT RIGHT	BLOCK RIGHT	LOW ADDR	HIGH ADDR
Main	5	6	1	105	1	104	1	10
Main	6	7	1	106	1	103	11	17
Main	7	8	1	101	1	103	19	28
Main	8	9	1	101	1	102	30	42

Fig. 3. The DIME-file base. Represented is a portion of Tract 1 in a hypothetical city (Cooke and Maxfield, 1967). The small numbers at the street intersections are the nodes, the larger numbers on Main Street are the addresses. Two types of records are created as illustrated in the tables.

system for the Address Coding effort of the 1970 census.

The basic element of the DIME file is a line segment defined by two end points. It is assumed that the segment is straight and not crossed by any other line. The metropolitan files usually define this unit as a street block face. Complex lines are represented by a series of segments approximating the line. The segment has two "node" identifiers, along with the coordinates of its two end points and codes for the polygon on each side of the segment.

While DIME topology makes much information accessible to urban researchers, neighborhood relationships are not made explicit. Segments sharing a node, for example, must be found by laborious search procedures. Search is also required to assemble the outline of a polygon. More importantly the DIME structure is cumbersome to use for many cartographic applications involving areas made up of complex lines. For procedures in a one-time checking effort, as is the case of Address Coding and Address Matching in metropolitan areas, it is quite adequate. For efficient computer storage and retrieval and for many applications, however, improvements are desirable. For example, the reliance on the individual line segment makes the reduction of detail for display purposes difficult since line segments cannot be simply deleted without correcting the reference codes for the affected nodes.

At the Laboratory for Computer Graphics and Spatial Analysis, the junior author has developed a data structure, POLYVRT, that is designed to contain all the information needed to construct any of the previously enumerated planar structures. The basic object of POLYVRT is the "chain." Like a DIME segment, a chain has nodes at its two ends, separates two areas, and is assumed to be uncrossed. It differs in that the POLYVRT chain may be made up of many points whereas the basic DIME unit has only two points. A boundary between two polygons can be referenced by a single chain no matter how complicated, because line detail is topologically unimportant (Laboratory for Computer Graphics and Spatial Analysis, 1974).

The coding of a complicated boundary as a unit is not unique to POLYVRT. The data bank used in the project "The Interactive Map in Urban Research" (Nake and Peucker, 1972; Peucker (ed.) 1973) as well as the World Data Bank I (Schmidt, 1969) are composed of "lines." In the latter case some of them contain over 4,000 points. The chain based system of POLYVRT, however, is a different type of structure because of the topological role assigned to the chain and the subsequent construction of a list data structure. Based upon this assignment, the topological information about a chain resembles the information on a DIME record except that the distinction between nodes (i.e., points used for more than one chain) and the points internal to a chain allows internal points to be eliminated without influencing the neighborhood relationships. The main innovation in developing a chain representation is that areas of significant line detail may be efficiently handled. Topological checking is reduced from dependency on the number of points to dependency on the number of boundaries.

In addition to the indication of the relative location of the chain with respect to its neighboring polygons, POLYVRT information is stored in separate lists assembling the bounding chains for every polygon. Thus, searches can take place in two directions, from the chain to the polygon and from the polygon to the chain. This is very important for any type of neighborhood manipulation, since neighboring entities can be found through their "bounding" or "bounded" complements. In other words, to follow along a group of chains one flips through chain to polygon to the next chain, etc., whereas to traverse a series of polygons one tests for adjacent polygons by going through the chain directory for each polygon.

The POLYVRT program places point information in secondary storage. The three higher level objects (chains, nodes, and polygons) are core resident. Only the chain refers directly to the point file in the secondary storage. In addition to indicating the locations of the points in the point file, the chain record incorporates the name of the chain, the labels of the starting and ending nodes, and the left and right polygons. Conversely, the polygon list consists of the bounding chains in proper sequence (Figs. 4 and 5). A list linking nodes to chains could easily be constructed.

DATA STRUCTURES FOR THREE-DIMENSIONAL SURFACES

Boehm (1967) describes in a very detailed analysis the advantages and disad-

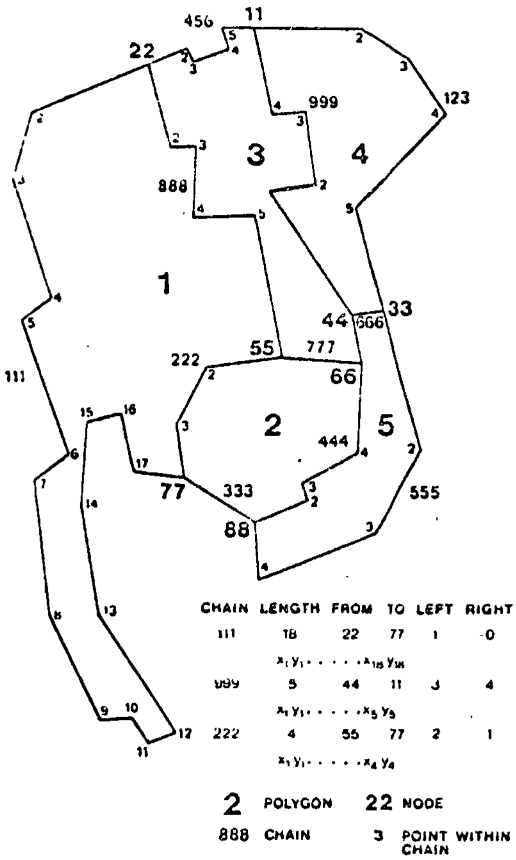


Fig. 4. The external representation of the POLYVRT chain-file. Every chain has a "name," the number of inner points (length), the two limiting nodes, the two boundary polygons, and a series of coordinates for the points.

vantages of different ways of encoding surfaces. He comes to the conclusion that the encoding of surfaces by contours minimizes the storage capacity, whereas a regular grid of surface points minimizes the computing time necessary for several types of manipulations. Boehm did not include in his study a data system with an irregular distribution of points but only contour encodings and regular grid structures of constant and variable mesh width. He did not reflect on the reasons why contoured data minimize storage capacity whereas a regular grid minimizes computing time. If he had done so it is quite possible that the development of geographic data bases would have taken different routes. Surprisingly this topic has produced little discussion despite the fact that extremely large data banks of terrain (digital terrain models) have been developed.

When encoding surfaces, it is necessary to adapt the density of points to the variation in the local terrain. The question of how dense the points have to be can be answered by again using the philosophy of the sampling theorem. For the terrain within a typical map, the variation can change considerably, resulting in a need for frequent adjustments of the sampling interval.

In a contour map the density of contour lines changes with the density of relief variation. Therefore, it fulfills the requirements set by the sampling theorem for a "non-stationary surface," i.e., a surface with changing terrain. For the regular grid, on the other hand, if the smallest object one wishes to detect anywhere within a study area is of size ("wave length") S, then the grid spacing everywhere must be $S/2$ or less (note, for example, Mark, 1974). The regular grid again tends toward redundancy since smooth areas within the study area will contain far more points than are needed to accurately portray their form. To improve the "resolution" of a grid by a factor of f, the grid spacing must be decreased by

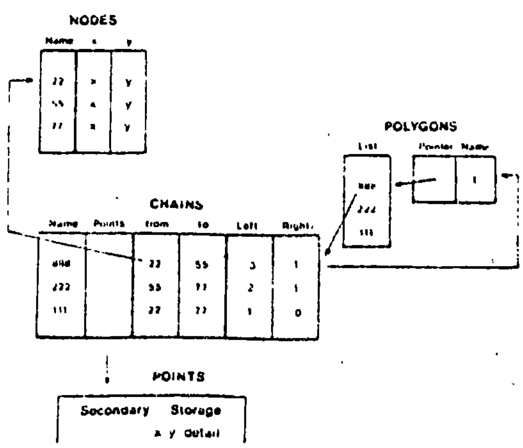


Fig. 5. The internal representation of the POLYVRT chain-file. Reference to polygon chains are positive or negative according to the direction of the chain used.

the reciprocal of this factor and the total number of points is increased by f^2.

The question has been raised many times in photogrammetry whether contours are the best representatives of topographic surfaces. It has been stated that contours do not detect many types of breaks which are frequent on terrain (e.g., Brandstatter, 1957). Therefore, the encoding of surfaces by vertical profiles has been attempted many times in photogrammetry in recent years. Points are encoded only when the slope of the surface changes (e.g., Kraus, 1973). In another case the break lines are encoded in addition to a regular grid (Sima, 1972), and in another approach only the break lines or only the ridges and channel lines of a surface are encoded (Grist, 1972). The amount of detail with these approaches depends on the scale used.

When performing numerical computations on the basis of these digital terrain models, the quantity of data involved will be only one determining factor for the amount of programming and computations needed. Most of the numerical computations on surfaces require some type of neighborhood function, either to compute some surface behavior, such as slope or local variation of relief, or to find the next unit for the drawing of a contour or a vertical profile. For a set of contours it is relatively easy to create a directory which indicates a sequence of contours in a type of tree in which the surrounding contour is the base and the other contours are the branches (Morse, 1968). To find the neighboring points on the two adjacent contours for a given point on an intermediate contour, however, one must search through all the points on the two adjacent contours and compute the distances to the point in question. This procedure is time-consuming if the contour lines are represented by very many points. A regular grid on the other hand has an implicit neighborhood function and finding a neighbor does not involve search nor extra computer time. For a set of irregularly-distributed points as they are represented in very simple data structures (e.g., SYMAP) the creation of a neighborhood function is usually performed by finding the closest small number of points where the number varies around six.

THE GENERAL CASE

It has been noted that although planar surfaces and three-dimensional surfaces look rather different it takes only a few assumptions to treat one as the subset of the other. While three-dimensional surfaces are always based on interval or ratio data, planar surfaces often involve ordinal and nominal data. However, it has been shown that ordinal and nominal data may be treated as interval data (Nordbeck and Rystedt, 1970; Rosenfeld, 1969) and even without this conversion we can combine the two types into one general case by simply using different assumptions about neighborhood.

Given a set of n data points for which x, y, and z coordinates are known, continuously defined surfaces (i.e., surfaces for which one and only one value exists at every point) can be created, using any of three different assumptions about the surface behavior (Peucker, 1972). The first assumption is that of a stepped surface, which says that the surface retains the value of a data point within the neighborhood of that data point, where neighborhood is defined either by a given polygon (the choropleth approach) or by the fact that the area is closer to one data point than to any other one in the neighborhood (the proximal approach). The second assumption is that each data point represents a sample of a single value on a constantly changing surface. Neighborhood is then a number of closest neighboring data points, and intervening values are interpolated with different types of interpolation procedures. The third assumption is that the data point is a sample from a constantly changing surface that may contain errors; thus the data point is not necessarily located on the surface, but close to it. This approach implies the further assumption that the actual surface is smoother than the surface constructed through the sample data points.

With these assumptions about surface behavior, any point or areal distribution of a variable can be treated as a continuous function $z = f(x, y)$, and planar and three-dimensional surfaces can be combined into one type. This has already been done in some computer programs, the most notable being SYMAP. Many cartographic applications must treat both types of surfaces and, therefore, data structures must be developed which can handle both types at one time.

THE PROPOSED DATA STRUCTURES

Planar Surfaces

The need to incorporate different types, or hierarchies, of polygons uncovered one of the limiting assumptions made in POLYVRT. A chain plays a dual role: first, it is the boundary of two areal entities, and secondly, it is the unbroken unit of point retrieval. This is not a problem if one is limited to a single nonoverlapping set of polygons. The following proposed new system, to bear the name GEOGRAF, is an attempt at greater flexibility.

Because of the addition of many layers of complexity involving multiple polygon sets, the chain cannot remain, to the same degree, the controlling object of the data structure. Just as the notion of an unbroken line is important, so is the notion of an unpartitioned space. In a system which must handle overlapping polygon networks, there is a need for a root object which is defined as an area uncut by any further partitioning. This object is termed the Least Common Geographic Unit (LCGU). The LCGU's are constructed as a POLYVRT polygon directly from chains. The relationship of the LCGU's to all other polygon types is hierarchical (Fig. 6).

In turn, the existence of the LCGU allows for the creation of each class of polygon. In order to allow simple coding of the boundary relationships at these higher levels in the structure, the "chain group" was devised. A chain group is a set of chains which form a boundary of two areal units for a given polygon class. These polygons are constructed of chain

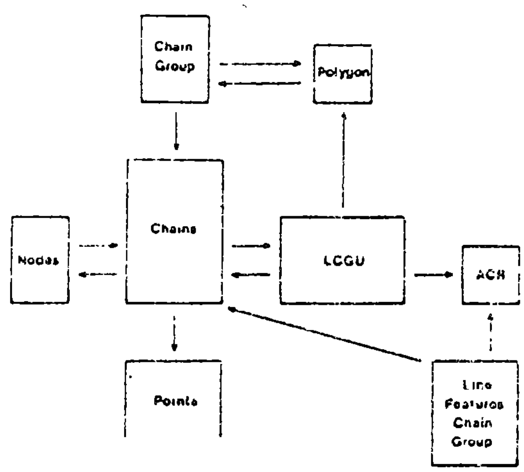


Fig. 6. The relationship between the various parts of the data structure.

groups which, in turn, are constructed from chains.

Line features can be built up of chains in the same manner as chain groups. Note that the chain group listing for each level of polygon and each listing for a line feature reference only the chains themselves. This allows each system to be considered as a separate directory which is core-resident only when that class of objects is retrieved.

The LCGU has other implications and applications that are useful because of the topological data structure. The LCGU, with its coding for each of the polygon sets, can be combined with contiguity information of linear feature types to produce an Attribute Cross Reference (ACR). The ACR is a table in which all objects (in polygon and linear systems) are cross-referenced to each other to determine nesting. By using attributes of chains (lengths) and LCGU's (areas, population densities, etc.) this cross-referencing capability can assign a string of data, collected for one polygon type, to a string of a second type.

Topological manipulation routines are central to the success of this structure. The intersection of geographic features will rely on topological knowledge to realize economies of scale in processing large files. All operations with lines will actually work with bands, built with endpoints of the line, and the furthest deviants to both sides (if the bands become too wide the lines are split, etc.). With this approach, the windowing process uses non-linear windows (which is often the case with geographic coordinates and map projections) and becomes quite elegant. Similarly, intersection procedures such as point-in-polygon, line-across-polygon, and polygon-over-polygon determination allow for gainful application of the topological principle. For point-in-polygon searches, chain groups are constructed which bisect the universe into parts to sort the points (or nodes of polygons) into three groups: left, within the band, and right. Only the second group needs more detailed treatment. The point set is then recursively partitioned until it reaches the level of the LCGU's. For line-oriented problems, the topological connections of the two sets compared would allow intersections to be limited to immediate neighbors.

Another important procedure will be a nested chain-intersection routine. Here, again, the chain band and its recursive segmentation is used. The number of points which define a chain is constantly increased until the intersection test can be determined. The search of a line through a set of polygons will use a graph-search algorithm developed for the GDS (Geographic Data Structure) project discussed in the next section. The neighborhood search routines create records of neighbors for every point or line or polygon at any specified depth of neighborhood.

Three-Dimensional Surfaces

To implement the ideas presented here, the senior author is developing a geographic information system for three-dimensional surfaces. Both systems, the three-dimensional and the planar, are based on data structures with explicit topological neighborhood relationships.

The basic philosophy of both approaches is to separate the data base from the application programs. In the early days of computer cartography, a data set was generally tied to the application program that used it. Now the available data have become so voluminous and the application programs so varied that extra efforts in the preparation of data for more efficient computations seem to be justified. We are well in step with modern computer science to separate the data base from the application programs with the data structure becoming the link between the two.

The creation of a structured data base is nothing new. The interpolation of an irregular grid to a regular grid of height-points provides a data structure through the implicit neighborhood function of the grid. And in the case where polygons are independently defined by a series of points, the neighborhood relationship is replaced by a search algorithm which finds the neighbors for every polygon by searching for matching segments in the boundary files of the other polygons. The idea is to spend computing time before any application has been performed in the anticipation of heavy uses of the data base. An efficient structure of the data base should not only speed up computations considerably, but should also simplify the production of application programs.

The data structure developed under the working title "Geographic Data Structure" (GDS) is based on irregularly distributed points which are assumed to be sample points without sampling errors from a single-valued surface. Two types of structures form the core of the data bases. The first creates neighborhood relationships by "triangulating" the data set and storing for every point the labels of all points which are linked with the point by a triangle edge (Fig. 7). The second structure is produced by selecting those points of the surfaces which lie along lines of high information content, such as ridges and channel lines, and defining them by their nodes, which are peaks, passes, and pits (Fig. 8). This second data structure serves two purposes: First, it is a general representation of the surface for rough computations; second, it is a "directory" into the more detailed first structure.

In the first structure, the creation of the neighborhood relationship is based on the assumption that data-sets are usually of

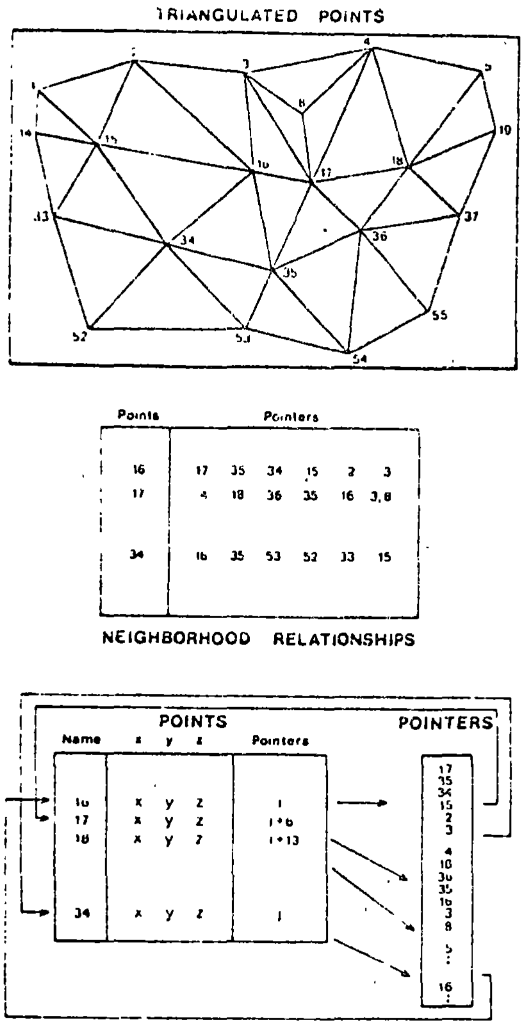


Fig. 7. The GDS-first data structure. The illustration shows the points on the surface with their links to neighbors (edges). The external representation is shown by the neighborhood relationships. The internal representation is composed of the point-file and the pointer-file.

two types: (a) sets of irregularly-distributed points which were digitized with the understanding that every point is significant, and (b) sets of regularly or irregularly-distributed points where it is known that a number of points are redundant and can be eliminated from the set, e.g., regular grids of points and encoded contours.

The first type of data set is linked by some type of triangulation. At least two approaches exist. The first (Dueppe and Gottschalk, 1970) creates all possible links, chooses the shortest, and eliminates all links which intersect the shortest. This procedure is repeated with the next shortest links until no links intersect. The result is the set of links with the minimum cumulative distance between neighboring points.

The procedure has one disadvantage: since $\binom{n}{2}$ links have to be created, the number of points is therefore limited to only several hundred. The first step in our approach therefore limits the links to a number of "potential neighbors," among which the shortest link is chosen and intersected with all other links originating in these potential neighbors. This procedure limits the number of tests for intersections of links to less than $\frac{n \cdot m^2}{4}$ where m is the number of potential neighbors, an arbitrary number between 8 and 14 depending on the density variation of these points. The procedure does not guarantee, however, that only triangles are constructed; polygons with more than three sides can result, although they are relatively rare. The check for such polygons and their elimination is very easy and fast.

The second possibility is to create a triangulated structure through use of Thiessen polygons. A published solution (Rhynsburger, 1973) intersects for every point the links to every other point midway and chooses the smallest polygon created

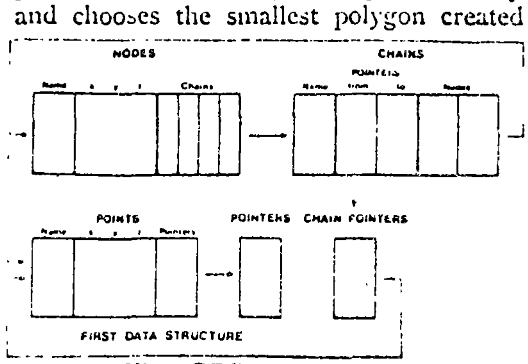


Fig. 8. The GDS-second data structure. Both the node-file and the chain-file have access to the first data structure, the node-file directly and the chain-file through a chain-pointer-file.

by the perpendiculars. Every point which contributes to the Thiessen polygon is a Thiessen neighbor. This procedure can again be simplified by the assumption of a limited set of "potential neighbors." The same checking routines as above have to be applied.

Another approach which limits the number of necessary tests, but is mathematically correct at the same time, has been developed within the project by Kurt Brassel at Harvard University. The procedure is based on "fields of potential neighbors" which converge very rapidly.

The alternative to triangulation is three-dimensional generalization, i.e., to select from a set of points those which define the structure with the least deviation from the original surface. The basic concept, developed by Randolph Franklin of Harvard University and T. K. Peucker, is to approximate the surface of a series of triangles through a selected set of points, where each additional point included into the set is the one which deviates the most from the approximated triangles until the deviations are below a given value (see Peucker, 1974).

In both cases, the triangulation and generalization of the surface, the result is a "linked list" of surface points. The term linked list means that points are linked with one another through pointers. In other words, a point is not only identified by its x, y, z coordinates, but also by a list of the labels of the points which form edges of triangles with the point. In our case each record consists of the x, y, z coordinates of a point and a reference to the start of the pointers to the neighbors in a pointer list. The reason for not having the neighborhood pointers with the point record is that the number of points varies considerably (Mark, 1974). Since the record has to be long enough to include all possible numbers of neighbors, large parts of the pointer sections would be empty most of the time. The pointers are sorted, starting with the pointer the least East of North of a point (Fig. 7).

The use of this type of data-structure is very simple and efficient. For every search (profile, contour, etc.) a criterion for edge-intersection is developed. For contouring, for example, the critical question is whether one point of the edge is above the contour level and the other below. A start is found and one point of the edge is considered a reference point and the other a subpoint. The next subpoint is found by looking up the next neighbor in the pointer list. If the test is positive, the intersection is performed and the process repeated. If the test is negative, the reference and subpoints are switched and the process repeated.

Other procedures are equally simple. To find a triangle, for example, one has only to have a reference and a subpoint. The third point is the next label in the pointer list of the reference point after the subpoint. To find all triangles one goes through the total pointer list leaving out all those edges connecting reference points with subpoints with a smaller label since they would create triangles which had been treated when the subpoint was a reference point.

Although the first data structure, as presented above, seems to be efficient in terms of storage capacity, it does not provide easy access to the data base which is often very large. It is for this reason that we are developing the second data structure to represent the general structure of the surface and to serve as a directory to reach into the first data base (Pfaltz, 1975) (Fig. 8).

The first step in the creation of the second data structure is to find the ridge, channel, and break lines on the surface. If one labels the highest point for every triangle, the unlabeled points are members of the channel line, at least on smooth surfaces. A subsequent search routine deals with the irregularities. Points along ridges are found by eliminating the lowest point of every triangle, using a routine developed for a regular grid by D. M. Douglas. The detection of break lines is somewhat more difficult.

Some theoretical studies of surfaces by Warntz (1966) show that ridge lines and channel lines cross at passes, a useful point of information relative to the development of the second data structure. Practical considerations suggest that in terrain and other surfaces, this regularity is not always present.

Once the topological structure is at hand, it can be used as a directory into the first structure. The line is therefore treated as a chain similar to the chain of the POLYVRT system. The nodes of the chain are the peaks, passes, pits, and other endpoints of chains on the surfaces. These points are stored with their coordinates and the names of the chains which terminate at the nodes. The chains are stored with the labels of the nodes and pointers into the chain lists which consist of labels of points in the first data structure (note that here the chain structure differs from that of POLYVRT).

A third component of the "Geographic Data Structure" should be mentioned since it illustrates very well the logical adaptation of a computer problem solution to geographical data. The problem at hand is the partitioning of the data set. Since with large data sets only portions can be kept in fast memory, the data base is segmented into "pages" which are brought into memory as units. For the "Geographic Data Structure" the paging system can solve several problems inherent in a complex geographic information system.

The boundaries of "patches," as we call the areal extension of a "page," are chains already defined for the second structure. Since detail along the chain is of no topological interest, the density of points along the chain can differ for its two sides. In other words, the density of triangles can change from patch to patch. This allows for very efficient data encoding even in terrain with sudden changes in the surface behavior as at the change from a mountainous area into a plain (Peucker, 1972).

Another advantage of the paging-system is the ease of including topographic and planar information. Linking point, line, and areal data to the triangulated points would lead to high definitional redundancy. The secondary structure could lead to ambiguities where the terrain is very elongated. Since an attempt has been made to keep the shape of the patches as compact as possible, the combination of non-terrain data with patch boundaries seems to be most appropriate.

Since the patch boundaries are again chains, another virtue comes to light: The patches can be treated as polygons of the POLYVRT system with little difficulty. This link between the two systems lends hope that eventually they may be merged.

It is an appropriate question to ask what such a data structure as the GDS will be able to accomplish. A number of display routines have already been developed (Cochrane, 1974) and a series of procedures for surface analysis based on heuristic searches are underway (Fig. 9). Since both levels of data structure are graphs, we will be able to rely on many of the developments connected with operations research, specifically network analysis, for the manipulative treatment of the data.

As both systems, GDS and GEOGRAF, have topological structures, it is possible to merge the two. The creation of polygons from points is the major link from the GDS project to GEOGRAF. The creation of a set of centroids for polygons allows the conversion in the opposite direction. This way, surfaces can be treated as polygonal sets and can be displayed and manipulated by the routines of GEOGRAF. Conversely, polygonal data can be treated as surfaces for GDS. The neighborhood routines are what make the project useful in quantitative geography and planning. Neighborhood searches are extremely expensive without the topological data structure, but they are usually a most important part of urban and environmental analyses once a general overview is obtained from the data.

Although basic research and application development are two sides of one coin and must go together to obtain lasting results, this paper has concentrated on the theoretical parts of the project since their development is ahead of the application routines, a fact which should be expected.

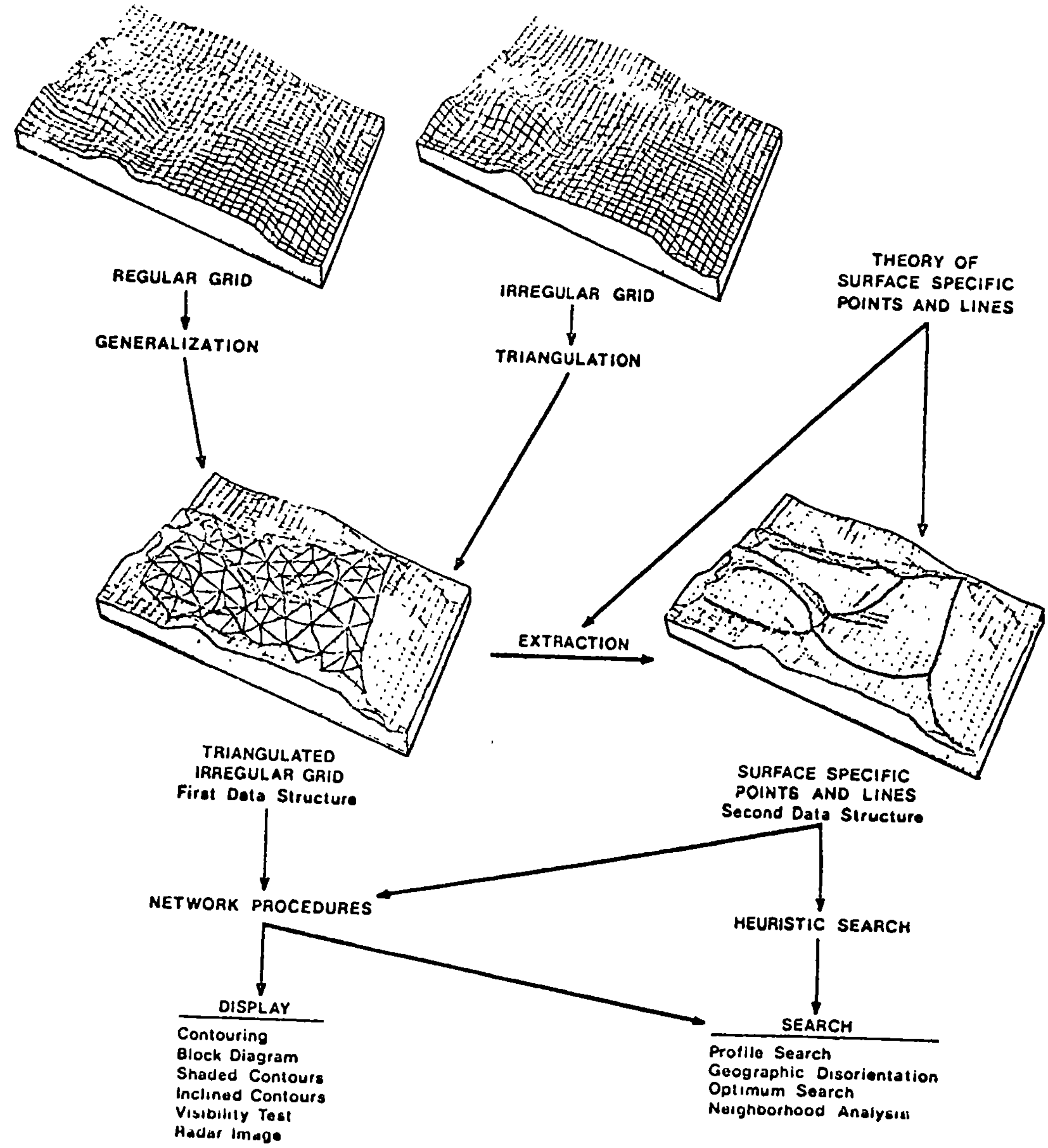


Fig. 9. The problem flow for GDS from the data base, via the extraction of points and chains for the two types of data structures, to their application.

The quintessence of the research so far is the hypothesis that topologically-structured data bases of three-dimensional and planar surfaces can result in reduced efforts in the development and execution of application routines. We have some indication that the hypothesis is correct; the real test will come when the bulk of the application routines is completed.

REFERENCES

Amidon, E. L. and G. S. Aiken (1970), "Algorithmic Selection of the Best Method for Compressing Map Data Strings," *Communications of the Association for Computing Machinery*, Vol. 14, pp. 769–774.

Arms, A. (1970), *MAP/MODEL System: System Description and User's Guide*, Bureau of Governmental Research and Service, University of Oregon, 59 pp.

Boehm, B. M. (1967), "Tabular Representation of Multivariate Functions with Applications to Topographic Modelling," *Proceedings*, 22nd National Conference, Association for Computing Machinery, pp. 403–415.

Brandstaetter, L. (1957), "Exakte Schichtlinien und Topographische Gelaendedarstellung," *Oesterreichische Zeitschrift fuer Vermessungswesen*, Sonderheft 18, Wien, 90 pp.

Cochrane, Douglas (1974), "Cartographic Display Operations on Surface Data Collected in an Irregular Structure," M.A. thesis, Simon Fraser University.

Cooke, D. F. and W. F. Maxfield (1967), "The Development of a Geographic Base File and Its Uses for Mapping," *Urban and Regional Information Systems for Social Programs*, Papers from the 5th Annual Conference of the Urban and Regional Information System Association, pp. 207–218.

Douglas, David H. (1973), "BNDRYNET," Peucker, T. K. (ed.), *The Interactive Map in Urban Research, Final Report After Year One*, University of British Columbia.

Dueppe, R. D. and H. J. Gottschalk (1970), "Automatische Interpolation von Isolinien bei willkuerlich verteilten Stuetzpunkten," *Allgemeine Vermessungs-Nachrichten*, Vol. 129, No. 10 (Oct.), pp. 423–426.

Grist, M. W. (1972), "Digital Ground Models: An Account of Recent Research," *Photogrammetric Record*, Vol. 70, No. 4 (Oct.), pp. 424–441.

Heiskanen, W. A. and H. Moritz (1967), *Physical Geodesy*, W. H. Freeman, San Francisco, Chapter 7.

Holroyd, M. T. and B. K. Bhattacharyya (1970), *Automatic Contouring of Geophysical Data Using Bicubic Spline Interpolation*, Geological Survey of Canada, Paper No. 70–55.

Hsu, M. L., *et al.*, (1975), "Computer Applications in Land Use Mapping and the Minnesota Land Management Information System," Davis, J. C. and M. McCullagh (eds.), *Display and Analysis of Spatial Data*, New York, John Wiley and Sons.

Junkins, J. L., G. M. Miller and J. R. Jancaitis (1973), "A Weighting Function Approach to Modeling of Irregular Surfaces," *Journal of Geophysical Research*, Vol. 78, No. 11 (Apr.), pp. 1794–1803.

Kraus, K. (1973), "A General Digital Terrain Model," translation from an article in Ackermann, F. (1973), *Numerische Photogrammetrie*, Sammlung Wiechmann, Neue Folge, Buchreihe. Bd. 5, Karlsruhe.

Laboratory for Computer Graphics and Spatial Analysis, Harvard University (1974), *POLYVRT Manual*, Cambridge, Mass.

Mark, David M. (1974), "A Comparison of Computer-based Terrain Storage Methods With Respect to Evaluation of Certain Geomorphometric Measures," M.A. thesis, University of British Columbia.

Morse, S. P. (1968), "A Mathematical Model for the Analysis of Contouring Line Data," *Journal of the Association for Computing Machinery*, Vol. 15, No. 2, pp. 205–220.

Nake, F. and T. K. Peucker (1972), *The Interactive Map in Urban Research, Report after Year One*, University of British Columbia.

Nordbeck, S. and B. Rystedt (1970), "Isarithmic Maps and the Continuity of Reference Interval Functions," *Geografiska Annaler*, Vol. 52, Ser. B., pp. 92–123.

Peucker, T. K. (1972), *Computer Cartography*, Association of American Geographers, College Geography Commission, Resource Paper No. 17, Washington, D.C.

Peucker, T. K. (ed.) (1973), *The Interactive Map in Urban Research, Final Report After Year One*," University of British Columbia.

Peucker, T. K. (1974), *Geographical Data Structures Report After Year One*, Simon Fraser University.

Pfaltz, J. L. (1975), "Surface Networks," *Geographical Analysis* (forthcoming).

Rhynsburger, D. (1973), "Analytic Delineation of Thiessen Polygons," *Geographical Analysis*, Vol. 5, No. 2 (Apr.), pp. 133–144.

Rosenfeld, A. (1969), *Picture Processing by Computer*, New York, Academic Press.

Schmidt, W. (1969), "The Automap System," *Surveying and Mapping*, Vol. 29, No. 1 (Mar.), pp. 101–106.

Shepard, D. (1968), "A Two-Dimensional Interpolation Function for Irregularly Spaced Data," *Proceedings*, 23rd National Conference, Association for Computing Machinery, pp. 517–524.

Sima, J. (1972), "Prinzipien des CS digitalen Gelaendemodells," *Vermessungstechnik*, Vol. 20, No. 2, pp. 48–51.

Switzer, P. (1975), in: "Sampling of Planar Surfaces," Davis, J. C. and M. McCullagh (eds.), *Display and Analysis of Spatial Data*, John Wiley and Sons, New York.

Tobler, W. R. (1969), "Geographical Filters and their Inverses," *Geographical Analysis*, Vol. 1, pp. 234–253.

Warntz, W. (1966), "The Topology of Socio-Economic Terrain and Spatial Flows," *Papers of the Regional Science Association*, Vol. 17, pp. 47–61.

A Self Instructional Package:
How to Digitize a Topographic Map

by

David M. Mark

INTRODUCTION

Digitization can be defined as the process by which "analog measures" such as length or location on a map, are converted into "digital computer-usable form" (Peucker, 1972, p.72), in other words, into numbers. When a topographic surface is digitized using surface-specific points, knowledge of the form of the surface being sampled (usually obtained by a visual inspection of a contour map or the land surface itself) is used to select points or lines which contain a maximum amount of "information." The digitization process involves 3 relatively independent phases:

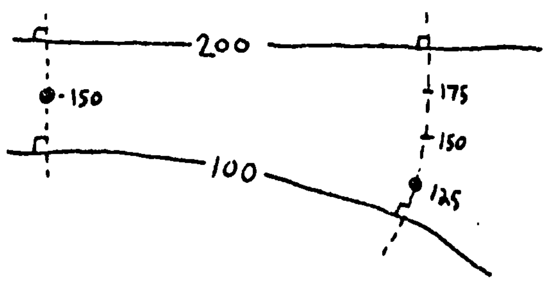
1) The selection of the points which are to be used to represent the surface;

2) the determination of the elevations (Z - co-ordinates) of each of the selected points: and

3) the determination of the planimetric locations (X and Y co-ordinates) of each of the selected points.

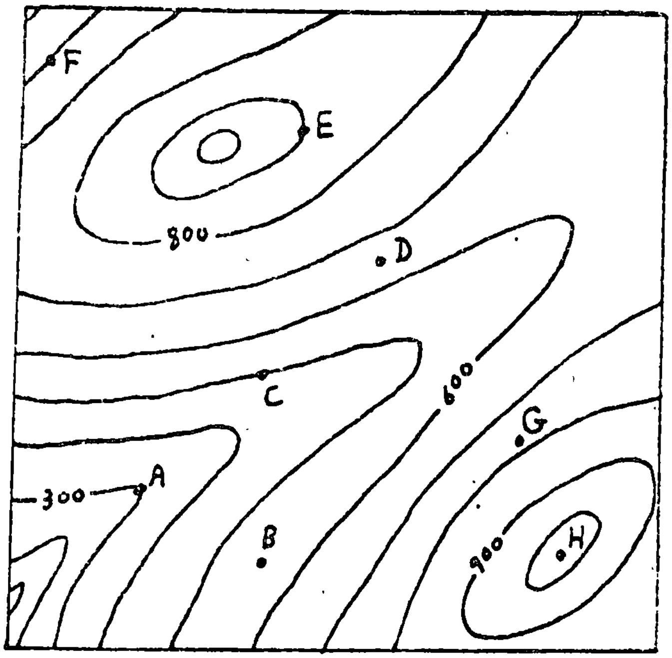
The present package is designed to instruct the reader in how to perform particularly phase 1) and to some extent phase 2) of this process.

A: Reading Contours and Interpolating: When reading a contour map, one should first determine the contour interval, the difference in elevations between adjacent contours. This will usually be the same throughout a map, and is often printed in the margin of the map. If not, it can be determined, simply: Find two nearby contours on a slope which both have their elevations indicated; count the number of spaces between these two labelled contours and divide this number into the elevation difference between the two contours. If one wishes to know the elevation of a point not on a contour, one must use interpolation, generally linear interpolation. First, imagine a straight or curved line perpendicular to both of the neighbouring contours and passing through the point. Next imagine that this line is divided into a number (2, 3, 4, 5, ...) of equal divisions in such a way that one of the divisions falls on the point. Each of the n equal division represents (contour interval / n) units of elevation, and thus the required elevation can be determined. See diagram below.



①

Determine the elevations of the eight points on the contour map below.



(2)

What is the contour interval of the map?

If you got any point wrong,* compare the correct answers with the map and if necessary, re-read section A. If you still do not understand, check with the instructor.

B: PEAKS AND PITS

Perhaps the most [illegible]portand and most easily recognizable surface-specific points are peaks and pits, which are local maxima and minima respectively on the surface. A peak is a point which is higher than all the immediately surrounding points. To put it another way, the land surface slopes downhill away from a peak in all directions. A peak is shown on a contour map as a

---

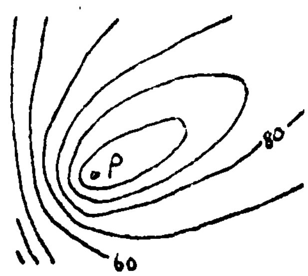
* A = 300 B = [illegible] [illegible] 500 D = 650 E = 900 F = 600 G = 775 (approx)
H = 1050 (approx) [illegible] contour interval = 100 units

closed contour loop which is higher than the surrounding territory. When there is no evidence to the contrary, the peak should be located at the centre of the closed contour (see diagram 3).

• P = peak

③

In some cases, however, the surrounding slopes can be used to determine more precisely where the highest point (peak) is located.

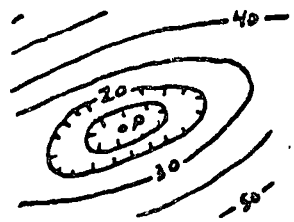


④

Here in diagram 4, the slopes clearly indicate that the peak is most likely to be toward the left end of the closed contour. A peak need not be the highest point on a map, and indeed most topographic maps show many peaks; a peak need only be higher than the immediately surrounding land.

Pits are the exact inverse of peaks; they are points which are lower than the surrounding land and are also shown by a closed contour, in this case lower than the surrounding territory. Usually (but not always) contour maps distinguish pits from peaks by placing small "hachures" along the closed depression contours. These lines point downhill and turn into the pit (see diagram 5).

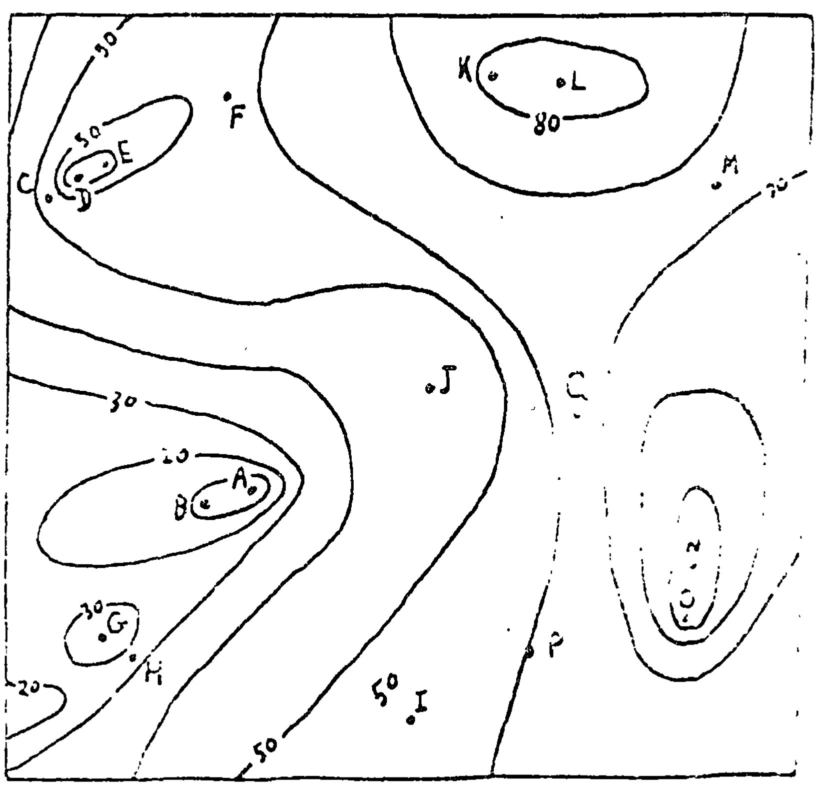
•P = pit



⑤

Once again, the form of the surrounding slopes should be used to determine the exact location of the pit. Except in special types of areas (limestone areas or "karsts" and certain types of glacial topography are examples), pits are rather rare in temperate landscapes.

On the following map, determine which of the lettered points are <u>ideal</u> locations of peaks and of pits, and distinguish between peaks and pits. Depression contours are <u>not</u> hachured in this example.



⑥

Answers

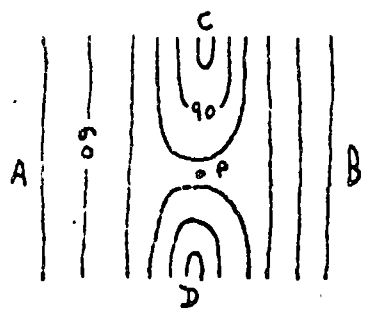
	Peak	Pit	Neither
A		X	
B			X
C			X
D		X	
E			X
F			X
G	X		
H			X
I		X	
J			X
K			X
L	X		
M			X
N			X
O	X		X
P			X
Q			X

Check your answers against the correct answers on the left. If you did not make <u>any</u> mistakes, go to part C.

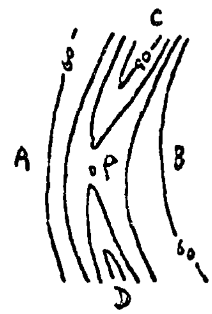
If you thought that G was a pit, or that D or I was a peak, you have mis-read the contours. Consider point I: The area <u>outside</u> the closed 50 contour is between 50 and 60 (the point P is on the 60 contour); thus the area <u>inside</u> the closed 50 contour <u>must be below</u> 50 and I must therefore be a pit. The same argument holds for point D, while the reverse argument can be applied to show that G must be a peak. If you made any other mistakes, please re-read section B, and if you are still unsure, consult the instructor.

C: <u>PASSES</u>: A <u>pass</u> can be defined as a point which is a maximum along a line in one direction and at the same time minimum along a line at right angles. On a contour map, it usually appears as follows:

The point P is a pass: It is a maximum along line A-B and a minimum along C-D. Its elevation is about 85 units. Of course the profile lines A-B and C-D can both be curved, or one set of contours less curved than the other, producing similar but not identical appreances on the contour map:



(8)



(9)

All of these passes have one feature in common: As one goes in a circle around the point, the land surface is, in turn, higher - lower - higher - lower, than the pass itself. The elevation of the pass can be estimated from the relative distances from the pass to the neighbouring higher and lower contours.

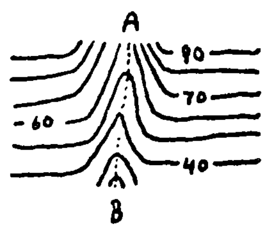
Turn back to the map on page 5 (diagram 6). Which of the lettered points on that diagram is a pass?

Anser: C, H, and M only. While F and Q may <u>look</u> like passes, examine the sequence of elevations as one goes around the point; you should see that these do not show the "higher - lower - higher - lower" sequence characteristic of a pass. If you are unsure of this, see the instructor.

D: <u>Course Lines and Ridge Lines</u>:

So far, we have examined three types of surface-specific <u>points</u>. This section looks at surface-specific <u>lines</u> of two sorts: Course lines and ridge lines

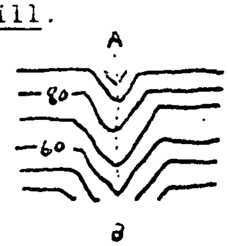
A course line will be failiar to most as the centre-line of a"valley." These valleys have sides higher than the centre-line, and in humid areas, these centre-lines are generally occupied by streams or "water-courses" (hence the name "course-line"). Even where streams are lacking, course-lines can be recognized by "V-shaped" contours in which the points of the "V"s point uphill.



(10)

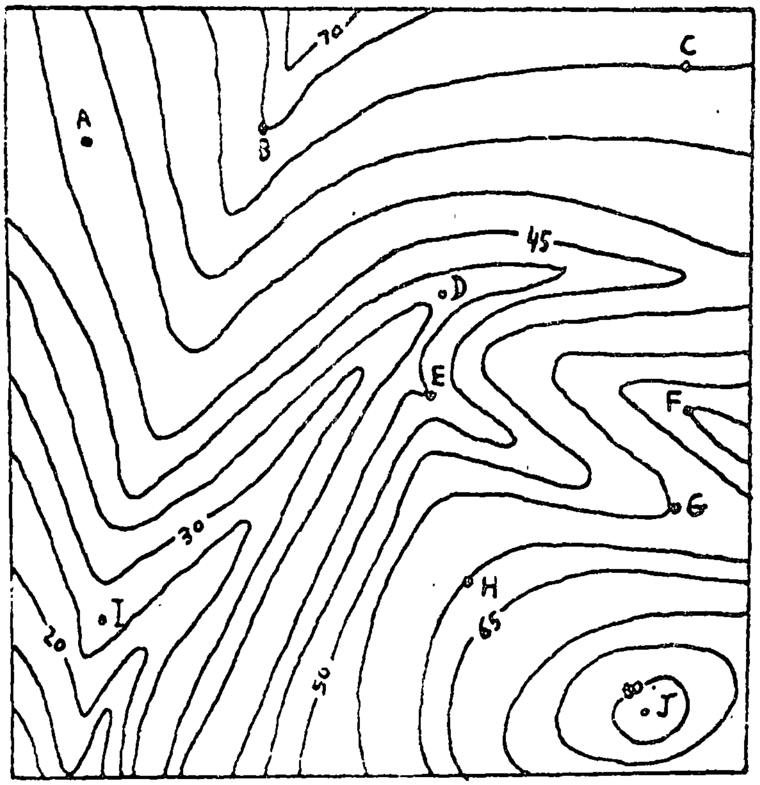
In the above diagram, A-B is a course-line. There are certain points along course-lines which are relatively more significant and which should be specified when a map is digitized. These include course-junctions, i.e., points where two course-lines merge or divide, points where a course-line bends, points where a course-line starts or ends, and points where a course-line enters or leaves a map sheet or study area.

Ridge lines are the exact inverse of course lines, but are not as obvious since they are not marked by obvious features (as are often course lines by streams). Once again, "V-shaped" contours mark a ridge, but in this case the "V"s point downhill.



(11)

In the above diagram A-B is a ridge line. As in the case of streams, ends, junctions and bends form important points along a ridge as well as points where ridges enter or leave the map. On the map below, sketch in the ridge lines and course lines using the indicated symbols, indicate which of the lettered points are ridge lines, course lines or neither.



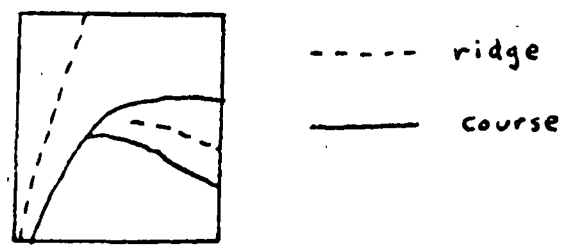
(12)

	Ridge	Course	Neither
A			
B			
C			
D			
E			
F			
G			
H			
I			
J			

Answers

	Ridge	Course	Neither
A			X
B	X		
C			X
D		X	
E		X	
F	X		
G		X	
H			X
I	X		
J			X

If you got any wrong, re-check the map. If you really want to argue, point H might be considered to be on a ridge, although this ridge, if it is a ridge, is not very well marked. If you are unsure of why the points are as they are, re-read section D and, if necessary, consult with the instructor.



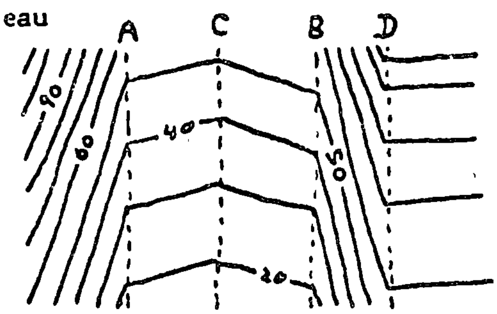
D 1: Relationships between ridges and course lines, and peaks, pits and passes

Course lines, when traced downhill, usually lead to either the ocean or a pit (often they simply lead off the map at hand). They may, especially in arid areas, simply end in an area of flat ground or an alluvial fan. Similarly, when ridge lines are followed uphill they often reach a peak, although sometimes they end on a slope or at a flat plateau. Ridges and course lines often bear a special relationship to passes: The two topographic "lows" encountered as one goes around the pass are often the beginnings of course lines, while the intervening highs are usually the starting points of ridges leading up to peaks. In fact, passes often form minima on continuous ridges passing between two peaks. Refer back to diagrams 7, 8 and 9.

The lines C-P-D form ridges on all three diagrams. PA and PB in 7 form well marked courses, while PB in 9 is a less well developed course line.

E: Breaks of slopes:

Another type of surface specific line is the line marking a break of slope, where the angle of slope of the land changes suddenly. This is reflected in a sharp change in the spacing of the contours. Examples are the lines where the steep slopes of a valley meet a broad, flat valley floor or gently sloping plateau



(13)

Here, C is a course line, while A, B, and D are important breaks of slope.

F: Digitizing a Map:

On the following attached section of a topographic map, do the following:

a) Place dots (•) at all the peaks, pits, and passes.

b) Sketch in the ridges, course lines and breaks of slope (if any).

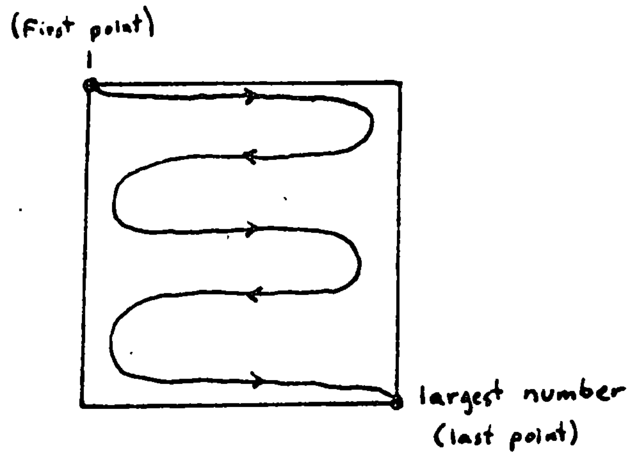
c) Place dots at significant points along the course lines and ridges.

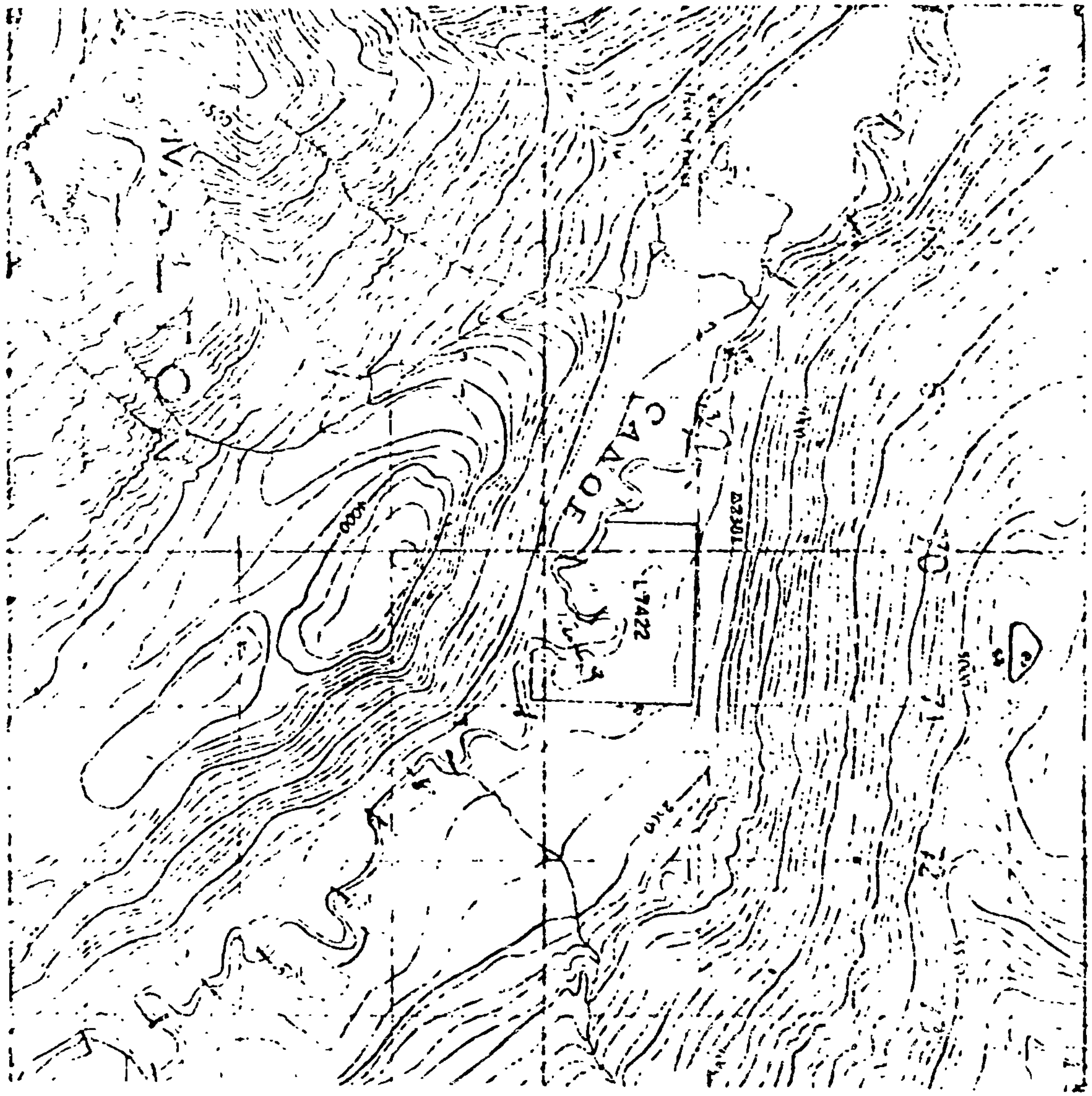
d) If there are any large areas of the map with no points, add one or two points at [illegible] locations within these areas. Include also points at corners or bends in the map boundary.

e) Number the points beginning in one corner and going back and forth in strips (see sketch below).

f) Make a list of the numbers and determine the elevations of each point by interpolation.

g) Hand in the map and a list of points to the instructor.




L7422

An Integrated System of Digitization of Cartographic Data

(1) Introduction

This report describes the work carried out at the University of Saskatchewan on digitization of map data and the conclusions it has reached at this time.

At first all effort was put into making digitization by manual tracing as easy and reliable as possible and a sophisticated on-line (PDP8) digitization program was developed which gave the operator considerable confidence in the results of his work.

However, when very large amounts of digitization were involved the speed limitations of the method (about 0.01 inches/sec on a daily average basis) were too great and although the possibilities of etched-line' sheets were examined, greater attention was paid to automatic line-following.

Automatic line-following could only be applied once the 'follower' had been positioned on the line and care had to be taken to see that complex situations were avoided which might confuse the system. The process adopted therefore was to utilize the manual digitization system to indicate starts and ends of lines as well as 'lengths of confusion.' In the designated lengths automatic following could be used and a method was developed which adequately handled the most irregular line forms.

The production of the 'director' information on the manual digitizer was now much simplified, merely requiring the digitization of points (usually two at the start of a line to indicate the direction vector) and one at the end. No particularly great accuracy was demanded as the line follower (ALF) itself searched for and located the exact position (of course, the error must not be so great as to have dubiety about which line) and any missing points or major errors could be picked up in the ALF process and the problem reported to the operator. The manual digitizer could now become a simplified point digitizer, and need not be on-line as no great advantage would result. Even a system with punched tape output would be satisfactory.

No great advantages result from on-line operation of a manual digitizer if only points are being output. With the Canadian

Hydrographic Service a particular problem arose because of the large number of pronounced depth soundings which had to be digitized 'without error.' Tests showed that pointing to a position and keying in the value resulted in an error rate of 2 - 5% - much too great. Attention was then focussed on the use of optical character recognition (O.C.R.) in an entirely automatic opera-tion. Providing the numerics were well-formed and there were no interfacing lines or other data, very high reliability results could be obtained. Various techniques were used which will be described later.

In both cases, ALF and OCR, the increase in speed was at least 10:1 and mainly because, no operator was present the cost per hour was reduced about 2:1. Thus a reduction in cost per inch of line or per sounding is about 20:1 and when a large amount of data is present this is a very large saving. Added to this an automatic equipment operates 24 hrs/day instead of a more normal 8 hour shift (with breaks) for a manual unit.

In fact it appears that the time and cost of this automatic ALF and OCR part of the work is so relatively low, that it is possible to do the entire work twice and computer compare the results, without excessive cost resulting.

Some of the numerics on survey sheets were so coalesced and difficult to clarify, that work has been started on Audio Character Recognition. Instead of keying in the numeric value, which seems to be prone to error, the digits are spoken, the computer recog-nizes them and answers back the same digits as a check to the operator. The process is of course relatively slow, and as it must be on-line, is relatively expensive.

One of the expensive and tedious processes in any form of digitization is the adding of 'descriptors' to line data. Every few inches of line may require a new variation - as for example on coastline. In on-line manual digitization this creates problems for the operator and even when producing the 'director' information for ALF, can divert the thinking of the operator. Since the development of the Computer Aided Map Compilation system using interactive displays it would seem advisable to only add these descriptors after the lines have been digitized. The operator sees the lines on the display, points to them and adds the descriptor. The possibility of easily adding descriptors to lines now removes one of the problems of using scanners, and tests on scan data are now proceeding. Where there are areas with many small lakes or islands the production of a director 'tape' manually becomes tedious. While it might be possible to designate such a zone and get the ALF to look for islands, a scan input with a scan to line conversion routine might well be preferable.

It would thus appear that a fully integrated system of digitization might consist of:

1) A number of off-line manual digitizers
   a) to produce 'director tapes' for an ALF unit
   b) to digitize names and symbol data
   c) to digitize urban line data (many straight lines and mathematical curves, as a series of points)
   d) to digitize other man-made features e.g. railroads as a series of points
2) One on-line manual digitizer
   a) to digitize lines too difficult for the ALF
   b) to digitize numerics too difficult for the OCR by audio character recognition
3) One combined ALF and OCR unit on-line
4) One or more interactive display system -- on-line
5) One scanning digitizer

(2) Economics of Use

The economics of the system do not at present include the use of scanner digitizers and the costs given below are only the hourly cost of equipment and labour where applicable at $5 per hour.

Off-line manual digitization   10 secs/pt. average, i.e. 360 pts/hr.

Equipment cost $20,000 including maintenance

Cost/hr $2.50 at 1 shift, 2000 hrs/yr, 4 yrs. life.

Labour

Cost/hr say $5.

Total Cost $7.50 for 360 pts. or 2 c per point.

On-line Manual digitization - tracing speed 0.03"/sec for high accuracy work (0.01"/sec on a daily average).

Equipment cost $40,000 including maintenance

Cost/hr. $5. at 1 shift, 2000 hrs/yr., 4 yrs life.

Labour

Cost/hr. say $5.

Total Cost/hr is $10. for 36" line or 30c per inch of line.

On-line ALF - tracing speed 0.5"/sec.

Equipment*cost $40,000 including maintenance

Cost/hr $1.25 at 24 hrs/day, 4 yr. life for 1800" line or .07c per inch of line.

On-line OCR - reading speed 1/sec.

Equipment* cost $40,000 including maintenance

Cost/hr $1.25 at 24 hrs/day, 4 yr. life for 3600 soundings or .04 c per sounding.

On-line interactive display operation

Equipment cost $40,000 including maintenance

Cost/hr $5. at 1 shift 2000 hrs/yr., 4 yr. life.

Labour

Cost/hr say $5

Total cost/hr is $10.

The usage time is discussed more fully later but might be several hours per chart or map.

On-line scanning digitization

No detailed costs available at present.

Overall cost of irregular line digitization

In a completely manual on-line digitization operation the cost for 1000" of line would be about $30 and this would include the addition of simple descriptors.

In a manual 'director' with ALF operation the costs for a similar 1000" of line might be the sum of: -

a) 'Director' specification of end points (say 100 points) is $2.00

b) ALF operation is .70

c) Display addition of descriptors and joining lines etc. 1 hr. is 10.00

$12.70.

*Same equipment will do OCR and ALF work.

To this are additional miscellaneous charges for program running between stages and possibly some manual digitization of very difficult line work.

Note a) the extra cost of repeating the ALF as a check is negligible.

b) regular line work as in urban plans is best done by simple off-line point digitization followed by computer generation of straight lines and mathematical fit curves.

Overall cost of sounding digitization (special to hydrographic work).

In a completely manual operation, sounding digitization costs 2 c per point. For 1000 soundings the cost is therefore $20.

Using an OCR system in areas where suitable (see later description) the cost is 40c for 1000 soundings.

Again it should be noted that the OCR operation can be repeated and the results computer compared with negligible cost.

It should be noted that some areas may not be amenable to OCR and also it is advisable to 'duff out' some line work adjacent to soundings to be read. This work might involve 1 hour of work (labour only, no equipment).

(3) Off-line manual digitization

Many different types of digitizers may be used for this purpose. The costs are relatively similar for high accuracy units. The normally obtained resolution is ± 0.002" to ±0.004" but some manufacturers are claiming ± 0.001".

The digitizers may be of the "free-pencil" type or ones which use a cursor mounted on a 'floating' or X-Y mechanism arm.

Output is relatively slow and may be on punched cards, punched tape or magnetic tape. Some operator check back system is advisable. If labels have to be attached to each point digitized, the speed of operation on average may fall to one in 10 secs. but when only a series of points, as in urban line digitization, is required rates up to one per second, may be obtainable.

The design of the cursor is of importance. For many purposes cross lines scribed on a plastic viewer are adequate, but care must be taken to see that these are properly centered and that excessive parallax does not occur. While the matter of centering may be properly dealt with when received from the manufacturer, frequent checks must be maintained as it is a frequent source of error, particularly in the 'rotatable' free-pencil units. Many manufacturers also do not provide a sufficient fineness of readability of lines or pointer for the operator.

When digitizing points it is important to know which ones have been measured. There are a number of possible aids and if possible all should be used.

A method frequently used, but wasteful in time, is to tick each point with a pencil. A better method is to have a small ink ejector attached to the point, which operates when digitization is called. Both of these should be used with a transparent overlay sheet over the map as it may be necessary to repeat work.

An overlay sheet with coloured strips about 1" wide or coloured squares 1" X 1" can also be very helpful, as the mind's eye can generally remember within such a zone.

A simple program can be written to check for duplicated points but missing ones are more difficult to locate.

When labels have to be attached to points the use of two operators, with voice communication, may be very helpful. Such methods apply to sounding, symbol and name digitization. An alternative is to voice record the labels on an audio tape recorder and key these in afterwards to 'merge' with the digitized points. A more sophisticated method is to use audio character recognition by the computer but this involves on-line work and is really only applicable to numerics as in depth soundings.

The problem of descriptor addition, other than as a label, is not so critical as with line digitization and this matter is left to the next section.

For man-made line work such as in urban street networks, buildings, roads and railways, the method of digitizing appropriate points along the lines appears to be adequate and in fact preferable to continuous line work. The actual points are left to the discretion of the operator; at the straight line sections he will only indicate each end, at smooth curves a few points on the arc and at irregular situations a high density of points. Adequate lines can be regenerated by mathematical spline-fit routines.

The use of a display system on which data can be viewed after periods of digitization is good and provides confidence to the operator. This is particularly so in the last example of line work digitization by point specification as the full lines can be generated when being displayed. A continuous on-line display would not seem to be advantageous cost-wise.

(4) On-line manual digitization

This process is mainly concerned with manual tracing digitization of complex line data, eg. rivers and coastlines. A more sophisticated digitizer than for point work is required. It must not put restraint on the careful tracing operation of the operator and therefore must be of the 'free-pencil' type. The 'pencil' point must be adequately fine, correctly centered to allow for rotation and be very convenient for the operator to hold.

Some method of checking the lines traced and the quality of tracing should be applied, and this appears best met at the present by the use of a 'smoked' overlay sheet upon which the 'pencil' or fine scriber leaves a thin line. Unfortunately the material at one time available, 'Masonscribe', no longer appears to be on the market.

A number of users have proposed making etched lines by printing onto appropriate etch sheets. This provides a gully along which the tracer can move at a much higher speed than when only tracing a black line. However, etching does have its problems and needs some very careful control. One user used etched metal plates and filled the etch lines with wax which was removed by the scriber point. In general the etched method appears able to give higher speed but is likely to reduce accuracy.

The work is very tedious and the digitizer needs to be on-line to a small computer so that the output can be examined continuously. This is important for production work to prevent time being wasted and generally appears to give better results as the confidence and therefore application of the operator is increased and concentrated on the line work in hand. If all is well a smooth 'beeping' tone is provided by the computer.

A sophisticated program for this work was written some years ago by the University of Saskatchewan and is now in use in the Canadian Hydrographic Service. Before describing this in some detail one other matter of prime importance should be mentioned. The output from a line digitization operation should be 'clean.' A number of users have met this problem and now appreciate the computer problems involved in trying to correct tapes which include errors. This has been considered in much detail in the U. of S. on-line program.

The first aspect is that all possible checks are made in the logical quality of the recording and only if this is satisfactory, does the 'beeping' continue. This covers parity errors etc. The second aspect is based on the property of a line. For careful line tracing it is not expected that a certain speed can be exceeded - for standard work this is about 1/30th of an inch per second. At 250 increments/inch this means an average of 8 increments/second in the present version, and as sampling is carried out every 1/100th of a second, it would not be expected that there should be more than one increment between each 1/100th second and the next.

The program is set up to check this and, if more than one increment does occur, the 'beeping' ceases and a routine called backtrack is automatically called. (Actually the program can be set to any value between 1 and 9 increments to allow for accurate to inaccurate work). The operator stops digitization, releases his button or footswitch and goes back about one quarter to one half an inch along the line and restarts digitization. As soon as one of these new coordinates coincides with one of the old ones in this backtrack area, normal 'beeping' resumes and the work goes on, the unwanted line 'tails' being automatically removed and a 'clean' line output obtained.

The same process can be used when the operator wishes to change his hand position without starting a new line. In that case he calls backtrack by pressing a special button. A similar process is used in 'island closure' but that is an even more complicated routine as it is necessary for the computer to remember not only the last half inch of line but also the first half inch. Again a 'clean' island line without gaps or tails results.

The 'backtrack' routine not only picks up operator faults of going too fast, jogging his arm and so on, but also detects digitizer equipment faults. For example if the encoder is faulty, the 'backtrack' will usually be repeatedly called each time that position is reached.

A final aspect of work done in the digitization program is to eliminate spikes. These can occur in the most careful work due to the encoding system and may be at right angles to the line or forward or backward along its length. As an addition the program removes small lumps from the side of an irregular line if these are only of the size of the line width itself. All these 'cleaning' subroutines make the later handling of the data much more satisfactory.

The addition of descriptor information is a serious problem. It has been found that the problems of allocating a full and exact descriptor tend to make the operator concentrate less on his line work or vice versa. In fact the jobs would be better separated; this idea is one being implemented using the interactive display system (see later). In some lines such as coastline, the descriptor may have to be modified every few inches, e.g. rocky to sandy etc.

While keyboard entry of descriptors can be made, it is usually preferable to use a thumbwheel arrangement in which the data can be preset and output by pressing a single button. The proponents of keyboard input have gone to the length of fitting a complex keyboard to the scriber unit itself, making it into a two handed device. While there is little doubt that output from a thumbwheel unit is logically correct, the U. of S. program checks any keyboard entry for validity before accepting it. It also checks that proper end codes are applied and that a descriptor is present.

The on-line program allows for the entry of point data as well as line data and this can be with no labels, short labels or long labels.

Another useful feature of the program is a position check which can be used periodically to see if the map sheet has been moved or has expanded or contracted. The locations are used called 'check coordinates.' These are usually the origin and X and Y limits. These can be digitized at any time at the end of a feature digitization and the old and new values will be output for visual comparison and action can thus be taken.

(5) Digitization of Reference Positions

Before proceeding further in the description it is advisable to say a few words about the reference positioning of digitized data.

It should not be necessary to place a sheet down in an exact position on a digitizing table; it is preferable to read off reference positions and correct the data by a program operation.

It is normal to digitize grid crossing points in some defined order and use these for calculation either to corrected machine coordinates or to a reference such as lat-long. The simpler method is to assume that the sheet is non distorted and use the grid crossovers to calculate an average correction. For distorted sheets it is preferable to use each grid crossovers independently and interpolate between each pair for a different correction.

It is also normal in the U.of S. system to digitize some specified lat - long positions and also the check coordinates as described in the last section.

When ALF is to be used corrections have to be made between the manual digitizing table and the ALF table, as the map sheets are unlikely to be put down on each in a similar position.

(6) Data Bank Incorporation

Work is now proceeding in adding digitized data into a full databank and carrying out the necessary computations as well as join and merge operations. This will be described in a separate report.

(7) Automatic Line Following (ALF)

The device consists of an XY mechanism driven by the on-line computer and which carries a Vidicon camera to any required position.

The vidicon is made to move in steps down each line designated by the 'director' tape(produced on an off-line manual digitizer.)

The view area of the vidicon camera is approximately 1/2" x 1/2" and the image is passed to the on-line computer via a low speed scan unit. Within the 1/2" x 1/2" area the line is tracked in core and an 'emergent' position found. The vidicon is then moved automatically until this position becomes a new'entry' position usually arranged to be at the center of one or other sides of the small square.

Each square operation takes about 1/2 one second and another 1/2 second for movement resulting in an overall speed of approximately 1 inch in 2 seconds.

The resolution of the system is ± 0.004" and it is only necessary to maintain reasonable stability on the vidicon image and orthogonality and precision to this extent on the XY mechanism.

It will be noted that actual line reading is under static conditions and thus the normal dynamic following problems of ALF units do not apply. There are no problems of small radii of curvature or of line thickness. The only requirement is that separate lines are at least 0.004" apart.

The output is provided in the same format as from a manual digitizer and includes the reference data from the 'director tape' input.

The line data is now complete, except for some missing parts e.g. where a name crossed a line, and at complex junctions. These can be added by passing the output tape to the on-line interactive display system (see later).

Some cartographic complexities are now being dealt with by the addition of logical routines. A particularly important one of these is concerned with 'sand' symbols touching the coastline and this can lead to some indecision on the part of the follower.

The ALF appears to work very well and is equally good on black lines on transparent overlays, transparent lines on negatives or the edges of colour mask sheets.

It will be noted that the ALF is physically the same as the OCR unit to be described next - the difference is only one of program and method of use.

The speed of operation of the ALF unit is such that it is feasible to reposition the sheet and carry out a second complete operation, comparing the two results afterwards as a very strong check of reliability.

The operator need only be present for the tracking of tapes and fitting the map sheet to the ALF unit. If the ALF operation detects an unspecified condition, e.g. a name across a line or a junction it will report this to the operator at the end of its work (it leaves the unspecified condition and goes to the next specified line start). The operator deals with this in the best manner, probably by adding a new start coordinate after the problem situation.

Prior to commencement of the ALF operation the operator moves the vidicon to certain reference locations (as used when making the director tape) and when the vidicon is centered (as seen on the monitor) he enters these values which are used to reorientate the 'director' locations to the ones on the ALF.

Programming work is now being carried out for automatic line following of a complex polygon structure without the necessity of redefining each line separately.

(8) Optical Character Recognition (OCR)

This work can become very complex but it is not so if limited to only ten digits and particularly if these are well formed as in numerical depth soundings on charts. No attempt is being made to read alphabetic characters.

The system hardware is the same as that for the ALF work consisting of an XY mechanism and Vidicon camera, both on line to the computer. No director tape is necessary in this case if all line work has been removed. In that case the vidicon is mechanically traversed backwards and forwards across the sheet, stopping to input the data each time a black mark is encountered.

Recognition is carried out in core, digit by digit, by a complex fitting of small rectangular patterns. The program is relatively sophisticated and works on a proportional basis, not requiring an exact masking fit. While the recognition is aimed at 100% reliability it is nevertheless allowable for the system to report 'unrecognized' in doubtful situations. These are reported to the operator at the end of the operation and he can add these manually. Again the operator is only concerned with fitting magnetic tapes, the map sheet to the mechanism, and adding in certain reference locations (in this case they are usually outside the normal map area.)

The system is capable of reading well formed hand-written characters and work is now being extended to field survey sheets. In this case it is useful to specify some 'director' information on a manual digitizer. This defines rectangular areas and indicates the process on each. These might be "do not attempt as too complex," "simple to read," "with suffix digits," etc. The defined rectangular areas remove much of the need for 'duffing out' lines and control the mechanical scan limits of the mechanism.

The accuracy of positioning can be as good as ± 0.004" for soundings, but as this depends on calculation of the 'weighted centre' the true accuracy may be somewhat less.

(9) Interactive Display System

A report is available on the sophisticated interactive manipulation of cartographic data developed at the U. of S. (Computer Aided Map Compilation).

As far as digitization is concerned, sections of lines can be added, deleted or modified. As line addition is not to an exact truth, but merely 'eye-balling' it should not be used for more than about 1/4" of line at normal scale, e.g. the missing section where a name crossed a line. The line addition is carried out at magnification on the screen which gives good smoothing and end joining properties.

It is also possible to point to any line or location and modify the descriptor or label associated with it. This method appears increasingly attractive as an aid to manual line digitization (instead of adding at the time) and is very valuable when scanning digitization is used.

The system uses an on-line Tektronix 611 storage display with interaction by 'write thru' spot and 'mouse' manual control.

It is a most valuable adjunct to digitization. In manual digitization periodic visual checks on the data should be made as well as after ALF or OCR work. It is rarely useful to build up the data on the display as the work is digitized, it is preferable to do this on an internal disc store so that the data can be displayed on request at any magnification and can be modified at the time if desired to do so.

The resolution of the display is 0.008" and the size of the screen is 6 1/4" x 8 1/4".

(10) Scanning Digitizers

Work in this area has only recently been commenced at the U. of S. A digitizer is now available in the department and some tests are being made using the Visicon equipment and programs.

The process was not considered advisable earlier on two counts a) the difficulty and expense of changing scan to line data format and b) the addition of descriptors.

The second problem has now been removed by the easy use of the interactive display program and new ideas and new methods may make the first quite acceptable.

A particular application is being examined at present to digitize areas containing multiple lakes and multiple islands.

(11) Equipment used for work in fully integrated system (not including scanner).

Qty	Item	Approximate U.S. Prices
1	PDP8/E-CA Minicomputer with extra 4K core memory and positive I/O bus interface	8,890
1	Teleprinter LT33-DC	1,500
1	Tektronix 611 storage display	3,500
1	U. of S. designed interactive interface with 'mouse'	2,500
1	RK8 disk system control	2,800
1	RK05 disk drive	5,100
1	TM8-EA DEC magnetic tape unit	9,950
1	Manual digitizer, d-mac, Gradicon, Bendix or Ferranti	20,000
1	Vidicon camera unit and control for ALF and OCR	5,000
1	XY mechanism to carry vidicon for ALF and OCR	15,000

CAMS : Computer Augmented Mapping System

Richard W. Hessdorfer

Bundesforschungsanstalt für
Landeskunde und Raumordnung

Bonn - Bad Godesberg

Abstract:

An interactive computer system to increase the ability of geographers to visualize and manipulate geographical and statistical data in the form of thematic maps is currently being developed in the "Bundesforschungsanstalt für Landeskunde und Raumordnung" in Bonn. This system, named CAMS for Computer Augmented Mapping System, is not only based on the need for the map designer to choose representation techniques and to place legends, symbols and text, but also on the need for the researcher to be able to explore a variety of data combination and grouping techniques to clarify his understanding of the data.

CAMS is being implemented on a Digital Equipment Corporation PDP 11/20 with 32 k-words, including both fixed-head and interchangeable disk storage, connected to a Tektronix 4002 A Graphic Computer Terminal with a joy-stick input device and keyboard, and a Calcomp 738 flatbed plotter. Geometric base data in the form of line segments and coordinates is supplied by a sister system, CADS (Computer Aided Digitizing System), also working online with the PDP 11/20 and a D-MAC-Digitizer.

In short CAMS enables the user to input a master spatial data set and then operate upon this data to create submaps, or to associate the geographical areas with demographic or statistical data. The results of these manipulations, executed in dialog with in a user-friendly command environment, can be previewed and altered on the graphical display, and then output to the plotter as hard copy in various colors, shadings, or line drawings. CAMS processes all

system elements as lists or strings: a map is a list of figures (geographical areas), a figure is a list of contiguous or non-contiguous line segments, a line segment is a list of X and Y coordinates. System data maps (data sets associated with a particular geographical map) are considered as lists of data values, symbol or shading designations, or alphanumeric information, each element corresponding to a figure in the 'mother' map. Special figures (legends, symbols, partial drawings) can be created and stored either as graphical macros, or as subroutine calls in map lists. CAMS allows the user to create and store up to 35 sub maps derived from the base geographical data set in a system maintained map library. These maps can be then accessed at any time, edited, deleted, or output to external bulk storage. Data maps can be operated upon mathematically (+, -, *, /) by constants or other data maps, logically (union, intersection, GT, LT, EQ), or statistically by grouping (eventually by standard statistical analysis functions). Provision is being made for the convenient addition of user models and data manipulation subroutines.

An important factor in the design of the entire system has been the desire to make the command language and interaction procedure simple and self-explanatory so that non-programmers could effectively use CAMS. All user replies are in free-format natural language using key word command cues. The system will allow the advanced user to write lists of complete commands without interrupting when they are error free;the beginner can request an input explanation at each command point by simply typing a blank line. Command menus are displayed at strategic points in the dialog and the user is given the option of menu-picking with the joy-stick cursor or typing a complete text command.

centro de educación continua
división de estudios superiores
facultad de ingeniería, unam

SISTEMAS AVANZADOS DE GEO-INFORMATICA EN PLANEACION

I M G R I D

SISTEMA PARA EL MANEJO DE MALLAS DE INFORMACION EN EL ANALISIS DE LOS USOS DEL SUELO Y LOS RECURSOS NATURALES.

LABORATORIO DE PLANEACION

A. INTRODUCCION

El sistema IMGRID es un conjunto de programas de computadora - para manejar y transformar mallas de información geográfica de acuerdo a especificaciones precisas basadas en el juicio y experiencia de grupos interdisciplinarios de expertos.

En términos generales, el sistema IMGRID agiliza el análisis - referente a la utilización del suelo y los recursos naturales- disponibles y permite el estudio de los efectos que diferentes políticas pueden producir en el medio ambiente.

El sistema está diseñado para ser utilizado por personas sin - conocimientos en programación, dado que se controla mediante - instrucciones sencillas que realizan operaciones específicas. - Una vez que el usuario entiende la naturaleza de las instrucciones, está en posibilidad de manejar el sistema sin mayor conocimiento del mismo. IMGRID puede funcionar como un sub-sistema dentro de un sistema de recopilación, almacenamiento, análisis y recuperación de información geográfica mas amplio.

Componentes del sistema:

1. BANCO DE DATOS.

Almacen de la información relevante al sistema bajo estudio.

2. MAPEO.

Con auxilio de la impresora de líneas de la computadora, produce representaciones gráficas (mapas) de la información contenida en el banco de datos y de la generada por los diferentes modelos del sistema.

3. MODELOS PARA LA LOCALIZACION DE ACTIVIDADES.

Permiten analizar la factibilidad de usos específicos del suelo en cada una de las zonas del área bajo estudio. Es posible incluir 20 diferentes usos del suelo en una sola corrida.

4. MODELOS PARA EL ESTUDIO DEL IMPACTO DEL USO DEL SUELO SOBRE LOS SISTEMAS AMBIENTALES.

Estos modelos permiten analizar los efectos probables sobre --

los sistemas físicos y biológicos, que ocasionarían diferentes usos del suelo. Se dispone de suficiente capacidad para analizar el impacto producido hasta en 30 componentes del sistema ambiental, en una sola corrida de los modelos.

5. MODELOS PARA LA EVALUACION DE PLANES DE UTILIZACION DEL SUELO.

Estos modelos permiten la evaluación de los diferentes planes de utilización del suelo entre sí, de acuerdo con el impacto o efectos que se inducen en cada componente del medio ambiente bajo estudio.

Aplicaciones:

Planeación del uso del suelo en nuevos polos de desarrollo industriales y turísticos.

Dictamen de impacto ambiental para los planes de desarrollo urbano y regional.

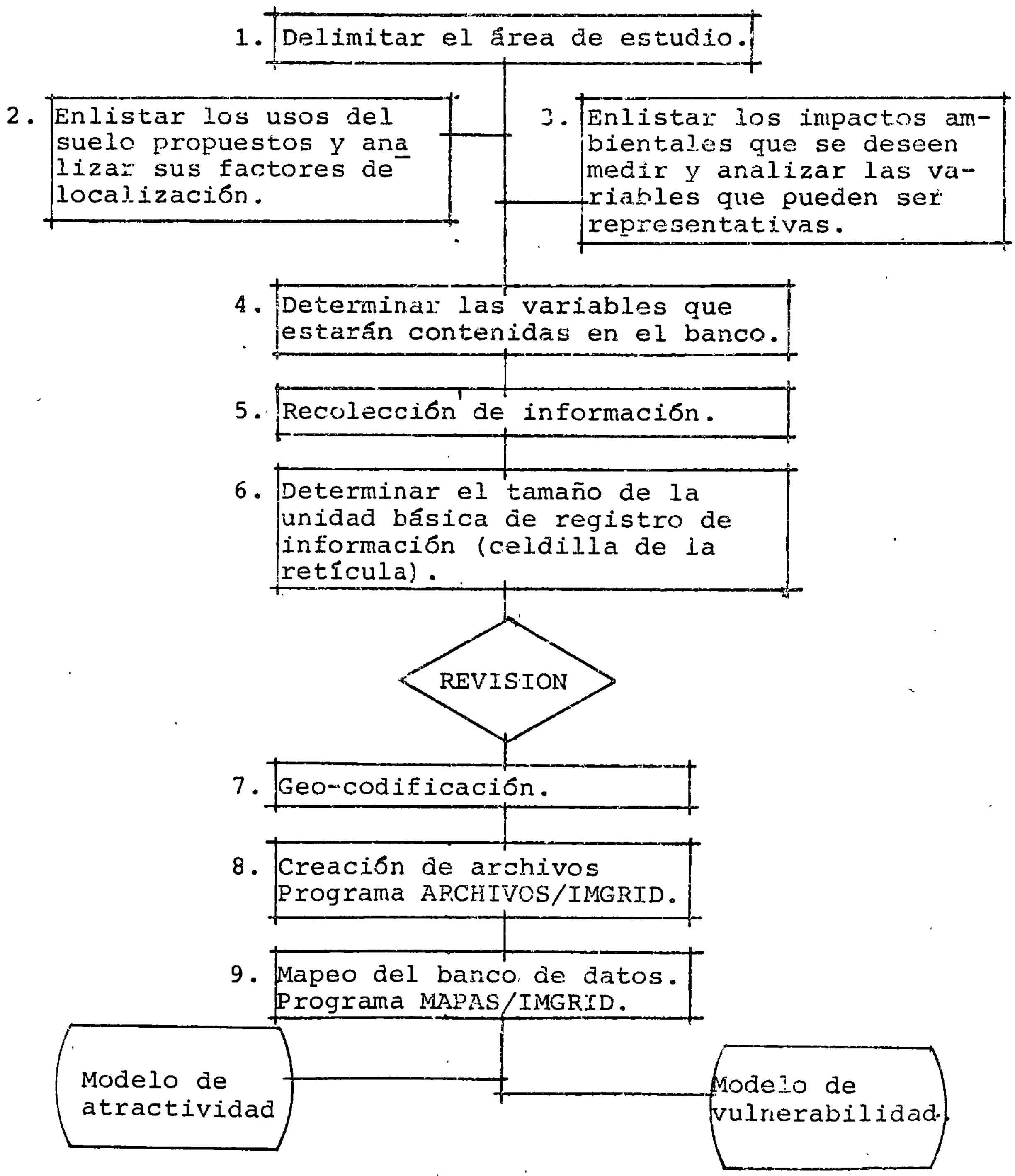
Estudios para la conservación de recursos naturales en parques nacionales y reservas territoriales.

Planeación de cuencas.

Capacitación profesional en planeación y administración del uso del suelo y los recursos naturales.

I. BANCO DE DATOS.

Para la elaboración del banco de datos se siguen los siguientes pasos:



Paso 1.- Delimitación del área de estudio.

El contorno del área bajo estudio debe ser delimitado mediante un rectángulo (contorno regular) ó conjunto de rectángulos (contorno irregular), dependiendo de las condiciones geográficas, jurisdicciones político-administrativas o algún otro criterio.

Paso 2.- Especificación de los usos del suelo.

Debe hacerse una lista de los usos del suelo que se desean localizar en el área de estudio, analizando al mismo tiempo sus factores de localización.

Paso 3.- Identificación de los impactos ambientales que se desean medir.

Debe hacerse una lista de los impactos ambientales (erosión, contaminación del suelo, agua, aire, cambios en el paisaje, etc.) que son de interés, asi como un análisis de la mejor forma de describirlos.

Paso 4.- Determinación del contenido del banco de datos.

Basándose en los análisis de los pasos 3 y 4 y la disponibilidad de información y recursos, se deben seleccionar las variables que estarán contenidas en el banco de datos, así como las categorías en que se subdivide cada una de ellas.

Paso 5.- Determinación del tamaño de las celdillas de la retícula.

El tamaño de la unidad básica de análisis (celdillas de la retícula) depende de los siguientes factores:

- Exactitud y tipo de los datos disponibles
- Tamaño del menor rasgo que se desee registrar
- Propósito para el cual van a usarse los datos
- Tamaño del área de estudio
- Limitaciones en los recursos para geo-codificar la información.

El tamaño de las celdillas debe permanecer constante en toda el área de estudio.

Paso 6.- Geo-codificación del contorno.

Contorno regular:

1a. tarjeta:

Col. 1-5	Número de celdillas de la retícula en sentido vertical. Perforado como un número entero, justificado a la derecha.
Col. 10	'0'
Col. 15	'0'

2a. tarjeta:

Col. 1-5	'99999'

Contorno irregular:

1a. tarjeta:

Col. 1-5	Número de hileras de la retícula en el sentido vertical que tienen el mismo desplazamiento hacia la derecha y la izquierda. Perforado como un número entero y justificado a la derecha.
Col. 5-10	Número de celdillas que se desplazan a la derecha del márgen del contorno regular. Perforado como número entero y justificado a la derecha.
Col. 10-15	Número de celdillas que se desplazan a la izquierda del contorno regular. Perforado como número entero y justificado a la derecha.

Debe repetirse este procedimiento en tantas tarjetas como sea necesario para especificar toda el área de estudio y sus fronteras.

Ultima tarjeta:

Col. 1-5	'99999'

Paso 7.- Geo-codificación de las variables.

La geo-codificación se hace superponiendo la retícula al área de estudio y registrando el valor ó categoría que toma la variable en cada una de las celdillas, anotando el número correspondiente de acuerdo a la sub-división especificada.

Los datos pueden ser registrados como:

a) Datos de punto (vgr. una cascada, un pozo, etc.).
b) Porcentaje de la celdilla con una actividad determinada.
c) Tipo predominante de uso del suelo.
d) Datos de línea (vgr. una carretera, un río, etc.).

Cada columna en una tarjeta representa una celdilla - de la retícula. Cada tarjeta representa una hilera - de celdillas. Si se tienen más de 80 celdillas en -- una hilera se deberá usar dos o más tarjetas.

1a. tarjeta:

Col. 1-2 Número de la variable, perforado como - número entero justificado a la derecha.

Tarjetas siguientes:

Col. 1-80 Valor o categoría que toma la variable- en cada una de las celdillas. Deben -- perforarse como números enteros de una- cifra (0-9) en las columnas correspon-- dientes. Se debe utilizar solamente el número de columnas necesarias. Para registrar todas las celdillas de una hilera de la retícula. Se deben perforar - tantas tarjetas como sea necesario para registrar todas las hileras de la retícula.

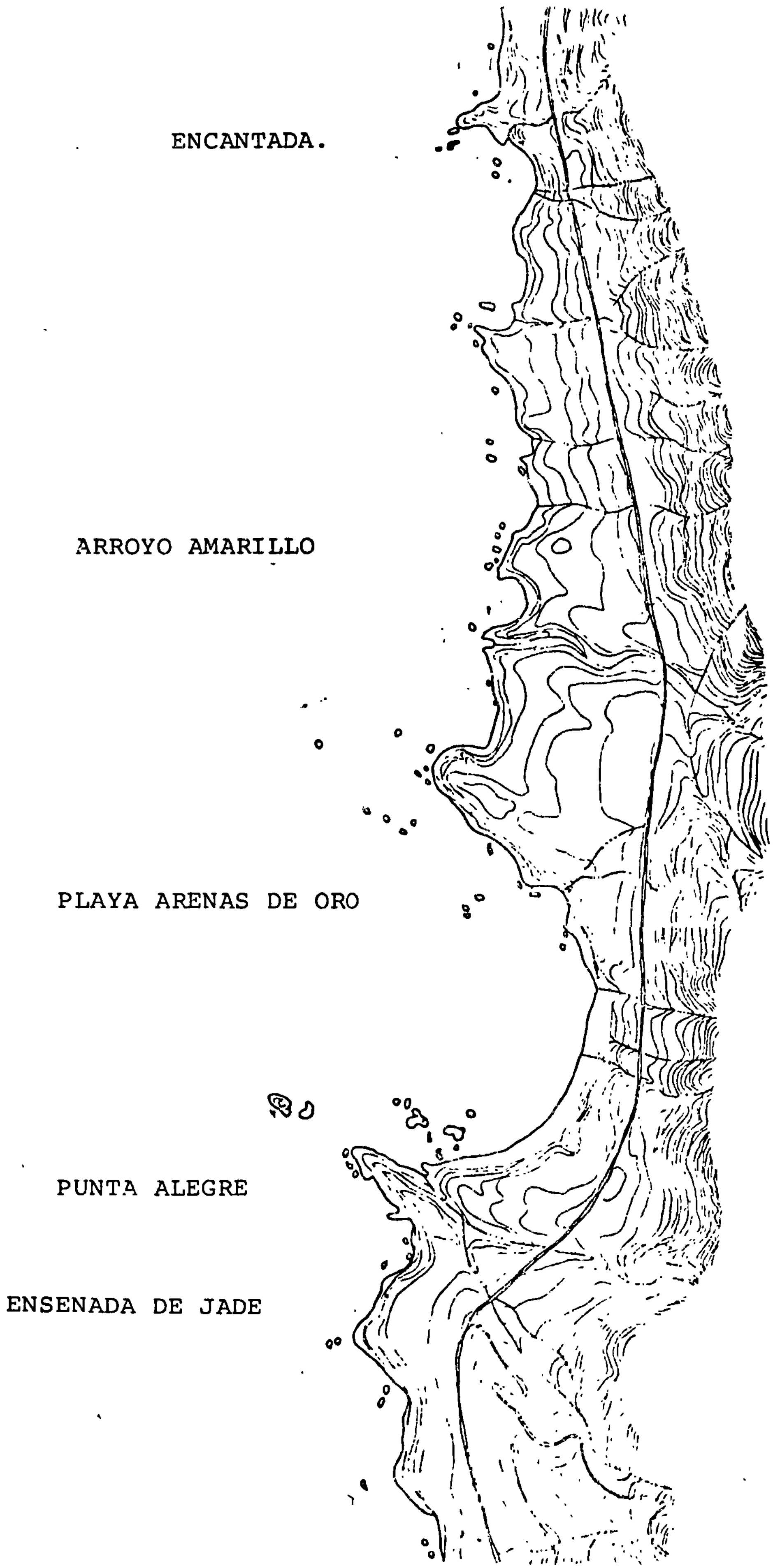
El programa limita el número de variables a 50 y el - de niveles ó categorias para cada variables a 10, numerándolos del 0 al 9.

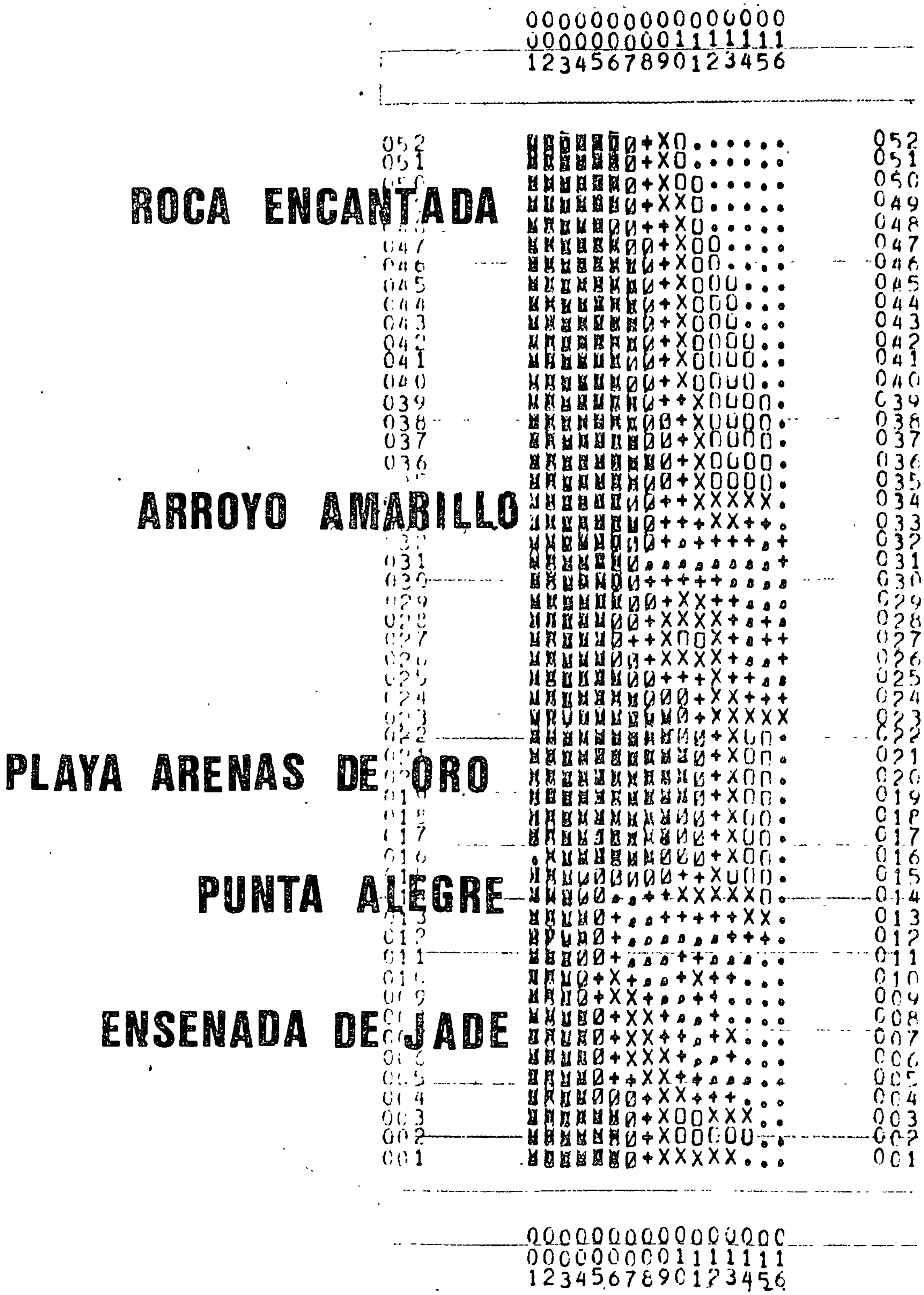
Paso 8.- Creación de archivos en la computadora.

Para almacenar las variables en el banco de datos, se alimenta el programa ARCHIVOS/IMGRID con las tarjetas perforadas en el paso 6 y 7 para crear archivos en -- disco magnético que serán utilizados por los modelos- de localización y de impacto ambiental.

Paso 9.- Mapeo del banco de datos.

Debe producirse un mapa para cada una de las varia--- bles incluidas en el banco de datos utilizando el programa MAPAS/IMGRID, cuya descripción es proporcionada en el capítulo V. Una vez impresos, revisados y co-- rregidos los mapas es posible pasar a la siguiente -- etapa en el proceso de IMGRID.


ENCANTADA.
ARROYO AMARILLO
PLAYA ARENAS DE ORO
PUNTA ALEGRE
ENSENADA DE JADE


ROCA ENCANTADA
ARROYO AMARILLO
PLAYA ARENAS DE ORO
PUNTA ALEGRE
ENSENADA DE JADE

MAPA DE LA VARIABLE 1 DEL INVENTARIO DE DATOS USOS DEL SUELO ACTUALES

AREA DE ESTUDIO VALLE PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

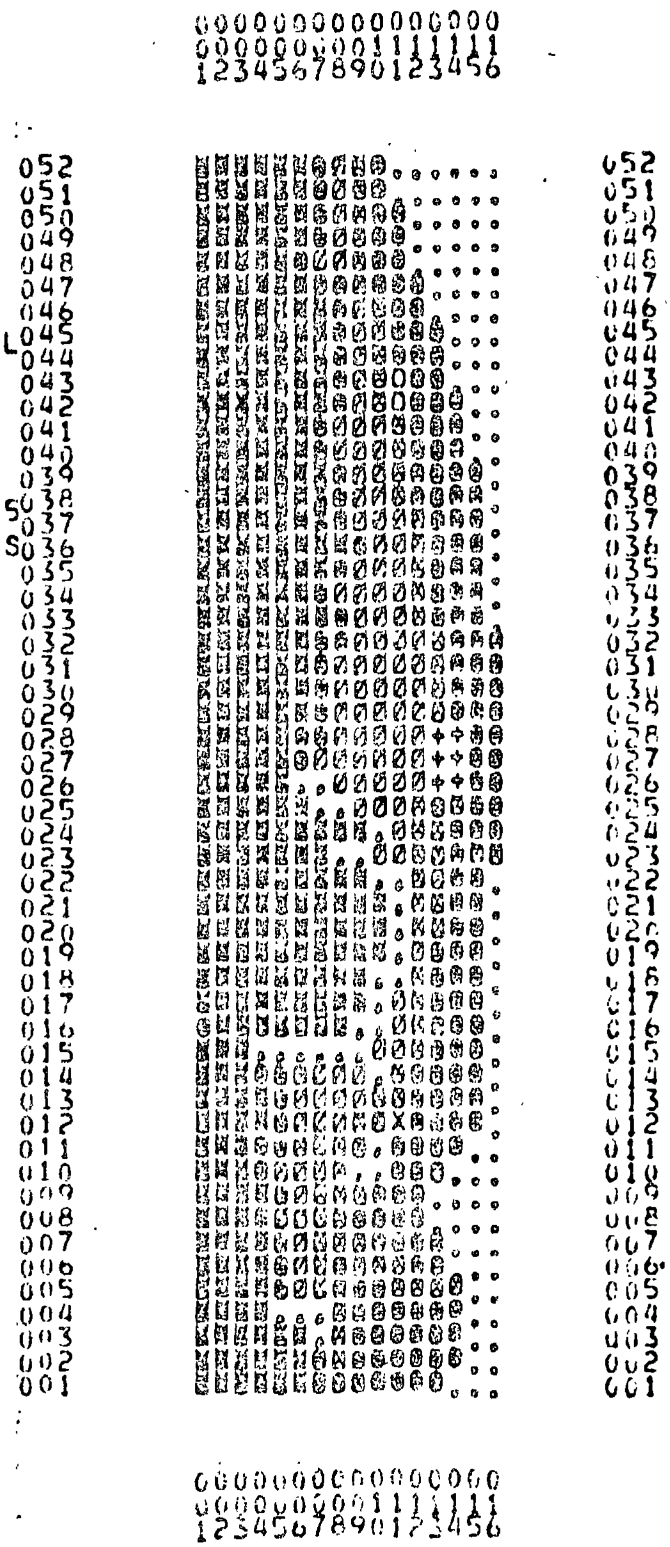
0 = NO HAY DATOS
1 = CAMPAMENTOS
2 = USO DIURNO
3 = ADMINISTRACION DE SERVICIO FORESTAL
4 = ESCUELA
6 = PASTO
7 = CARRETERAS
8 = RESERVA FORESTAL
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2
FRECUENCIA	98	3	28

NIVELES	3	4	5	6
FRECUENCIA	6	1	2	98

NIVELES	7	8	9
FRECUENCIA	76	203	317



MAPA DE LA VARIABLE 2 DEL INVENTARIO DE DATOS PORCIENTO DE PENDIENTE

AREA DE ESTUDIO VALLE DEL PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

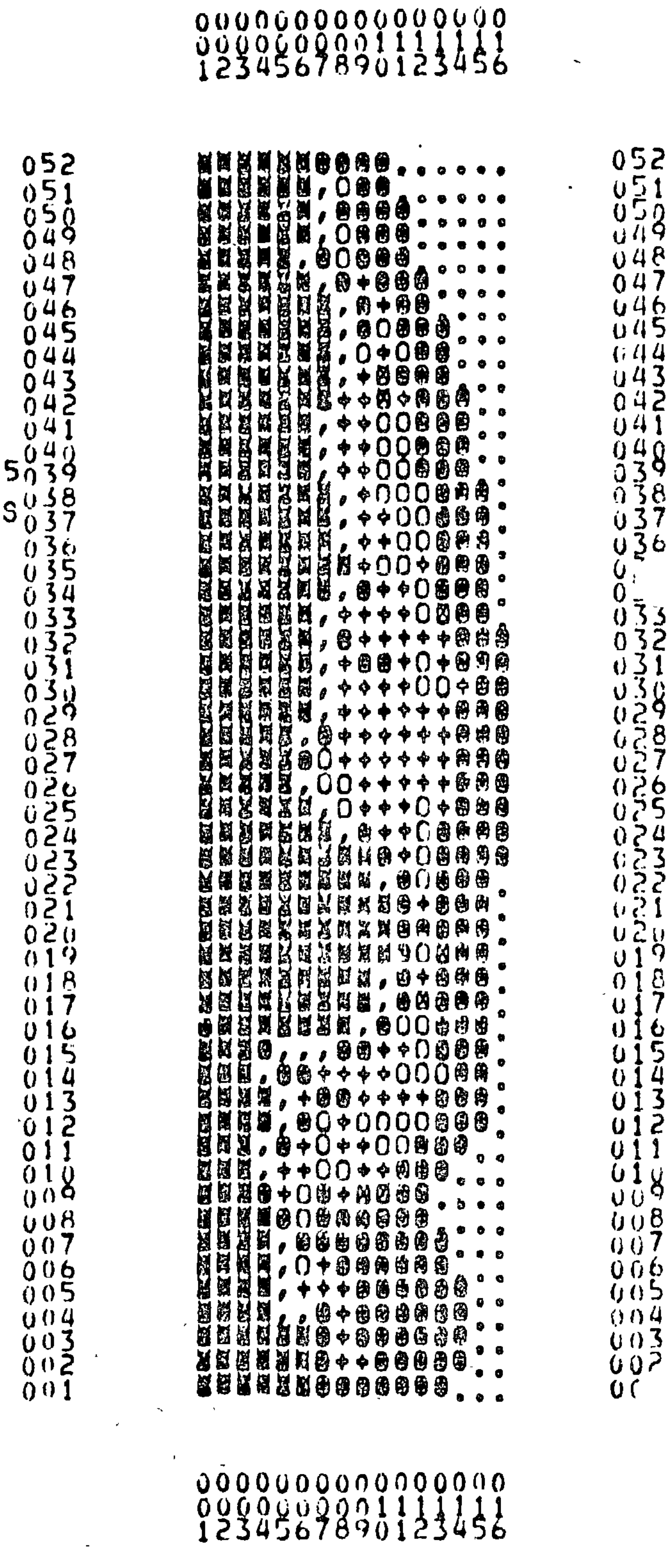
0 = NO HAY DATOS
1 = 100% AGUA
3 = 0-9%
5 = 10-15%
7 = 16-25%
8 = 25%+
9 = OCEANO

EL TAMAÑO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2
FRECUENCIA	99	42	0

NIVELES	3	4	5	6
FRECUENCIA	100	0	54	0

NIVELES	7	8	9
FRECUENCIA	7	214	316



MAPA DE LA VARIABLE 3 DEL INVENTARIO DE DATOS ORIENTACION PENDIENTE

AREA DE ESTUDIO VALLE DEL PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

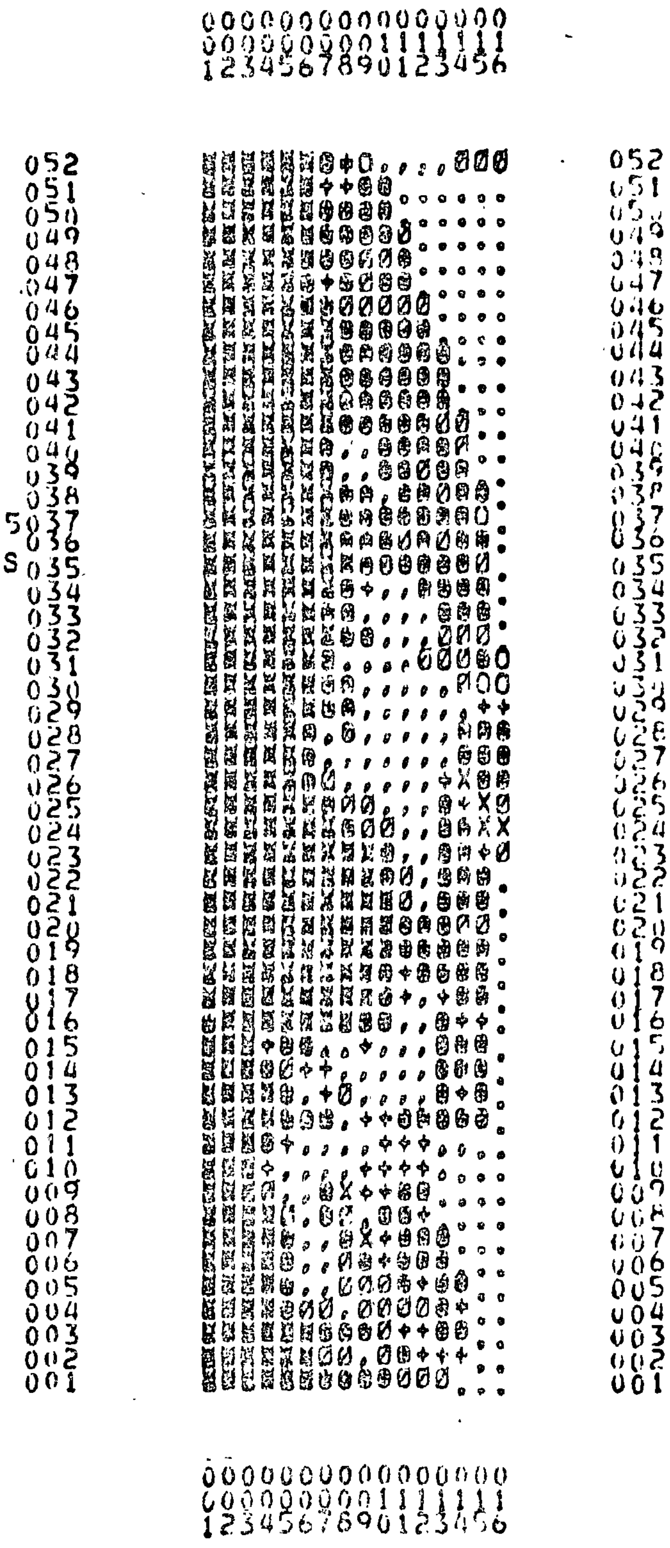
0 = NO HAY DATOS
1 = LLANURA
2 = NORTE
3 = NOROESTE
4 = NORESTE
5 = SUR
6 = SUROESTE
7 = SURESTE
8 = OESTE
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2
SIMBOLOS	0	1	2
FRECUENCIA	98	88	14

NIVELES	3	4	5	6
FRECUENCIA	44	6	5	60

NIVELES	7	8	9
FRECUENCIA	0	200	317



MAPA DE LA VARIABLE 4 DEL INVENTARIO DE DATOS VEGETACION POR ZONA

AREA DE ESTUDIO VALLE DEL PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

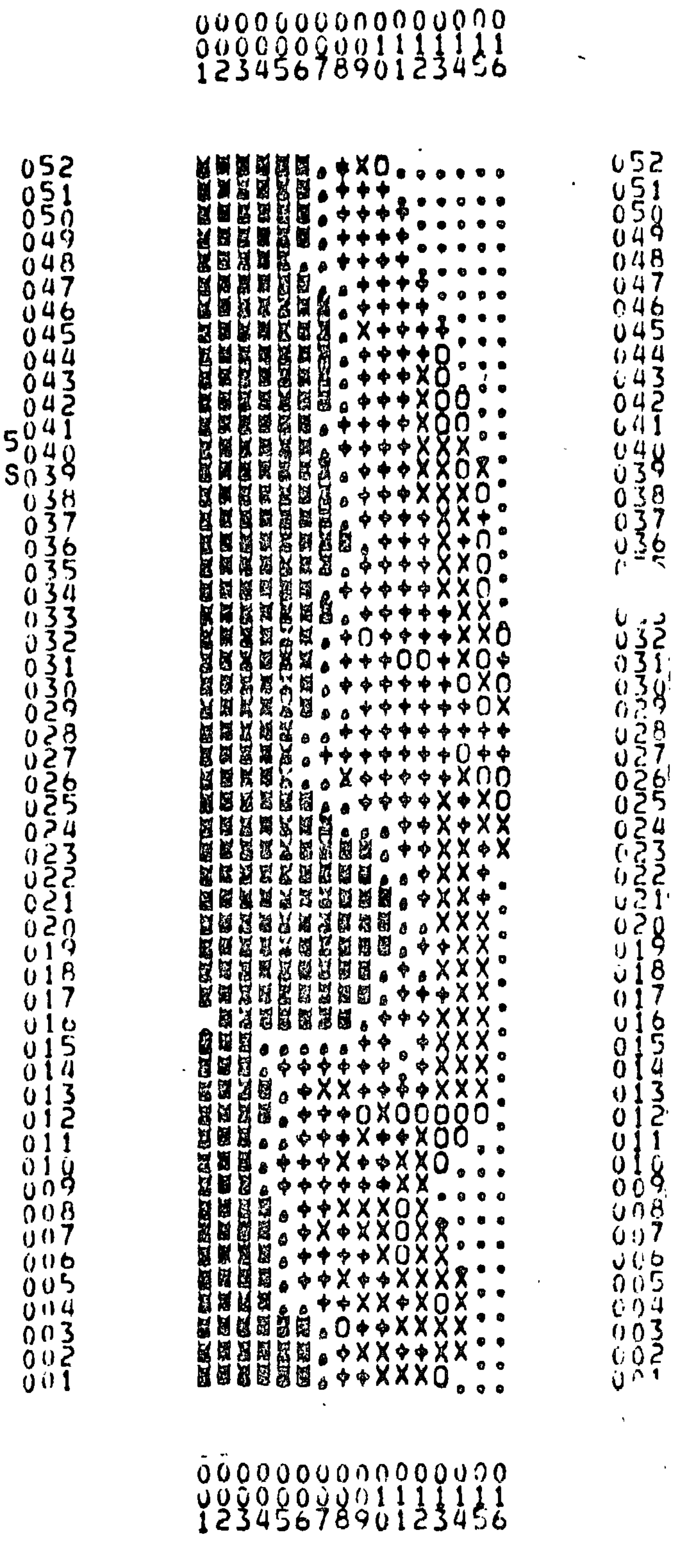
- 0 = NO HAY DATOS
- 2 = ZONA COSTERA
- 3 = SUELO PASTADO
- 4 = VEGETACION CHAPARRAL
- 5 = ZONA DE BOSQUE
- 9 = OCEANO

EL TAMAÑO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2
SIMBOLOS	0	1	2
FRECUENCIA	101	0	69

NIVELES	3	4	5	6
SIMBOLOS	3	4	5	6
FRECUENCIA	200	106	39	0

NIVELES	7	8	9
SIMBOLOS	7	8	9
FRECUENCIA	0	0	317



MAPA DE LA VARIABLE 5 DEL INVENTARIO DE DATOS DENSIDAD DE ARBOLES

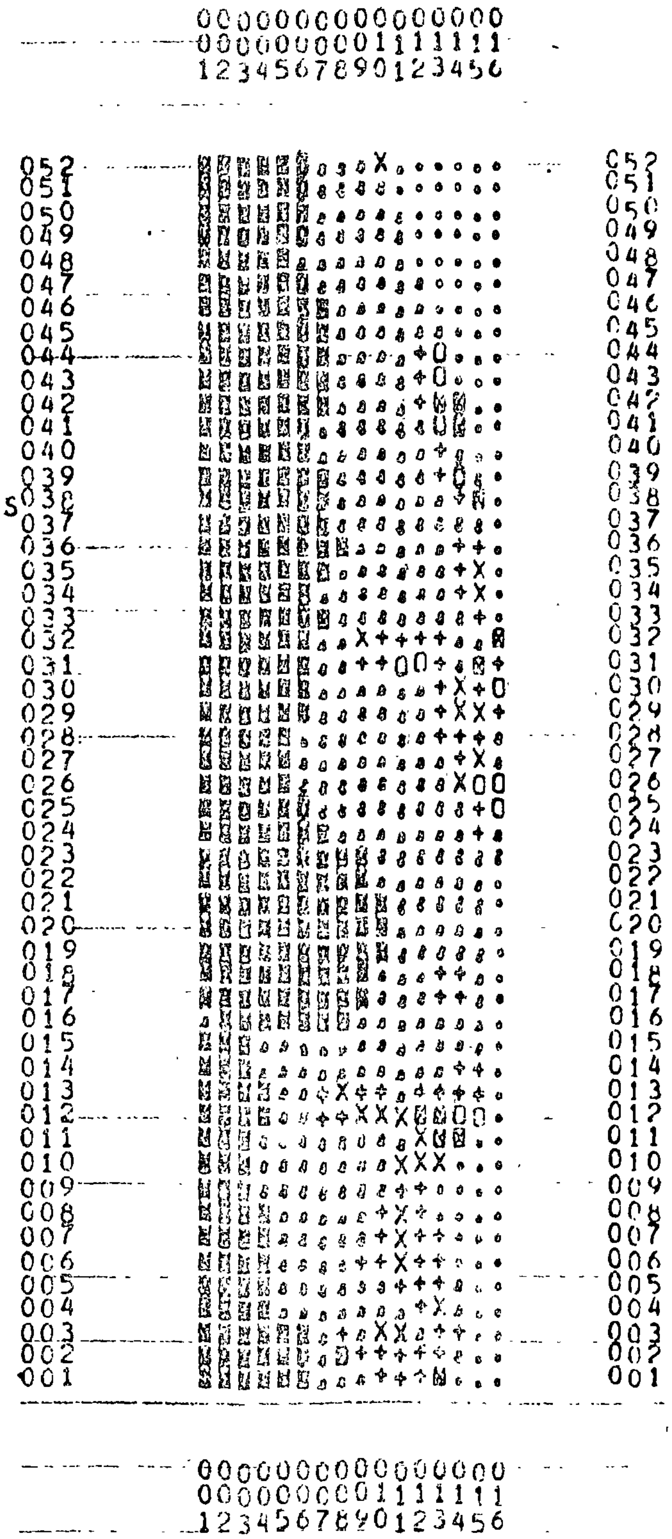
AREA DE ESTUDIO VALLE DEL PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

0 = NO HAY DATOS
2 = NADA
3 = 1-25%
4 = 26-50%
5 = 51-75%
7 = 76-100%
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2. ACRES

NIVELES	0	1	2	3	4	5	6	7	8	9
SIMBOLOS	.	'	a	+	X	O	Ø	M	W	B
FRECUENCIA	98	0	300	70	23	11	0	13	0	317



MAPA DE LA VARIABLE 6 DEL INVENTARIO DE DATOS   ZONAS LLANAS

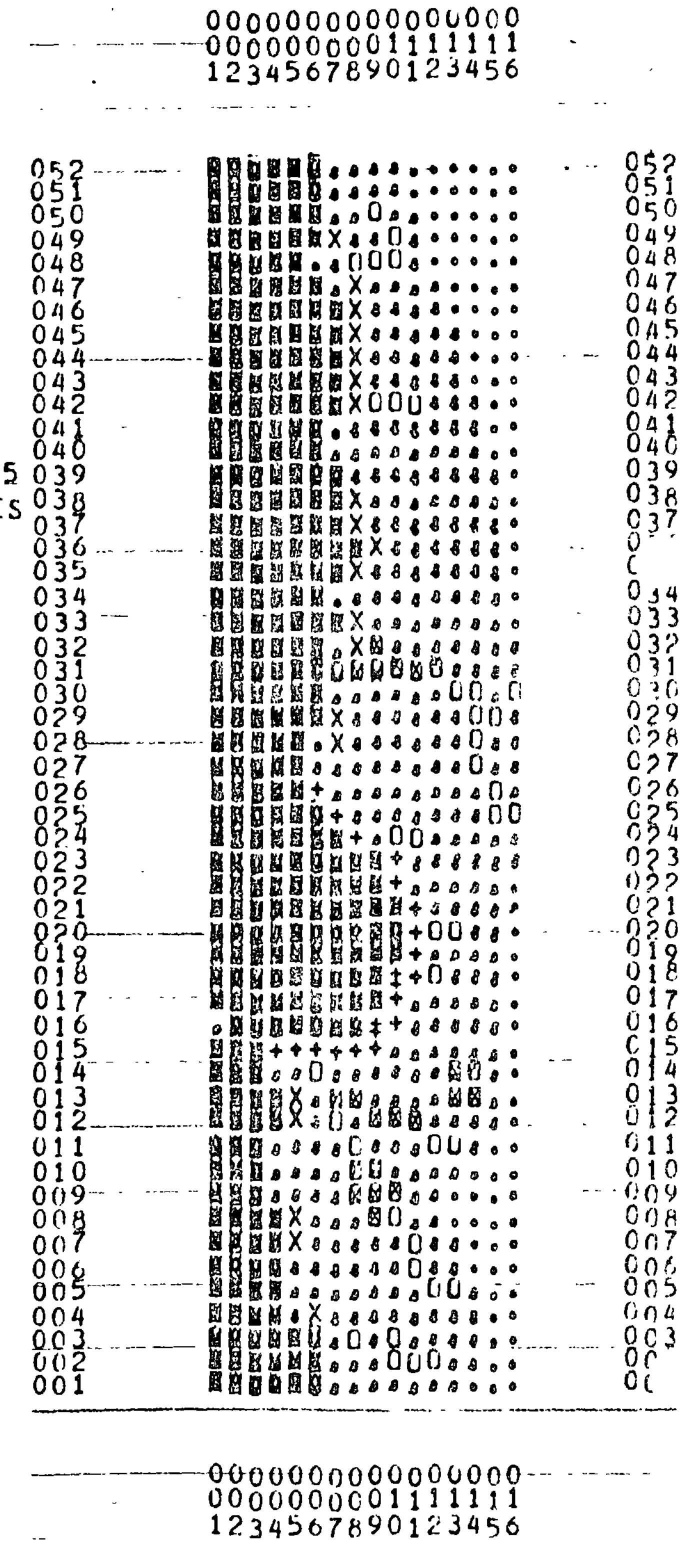
AREA DE ESTUDIO   VALLE DEL PACIFICO   LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

0 = NO HAY DATOS
2 = NO ZONAS LLANAS
3 = LLANOS SOBRE LA COSTA > DE 2000
4 = LLANOS SOBRE LA COSTA < DE 2000
5 = LLANOS TERRESTRES < DE 200
7 = LLANOS TERRESTRES > DE 200
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2	3	4	5	6	7	8	9
SIMBOLOS	.	,	'	+	X	0	Ø	M	B	■
FRECUENCIA	97	1	318	19	20	38	0	22	0	317



MAPA DE LA VARIABLE 7 DEL INVENTARIO DE DATOS ACCIDENTES GEOLOGICOS

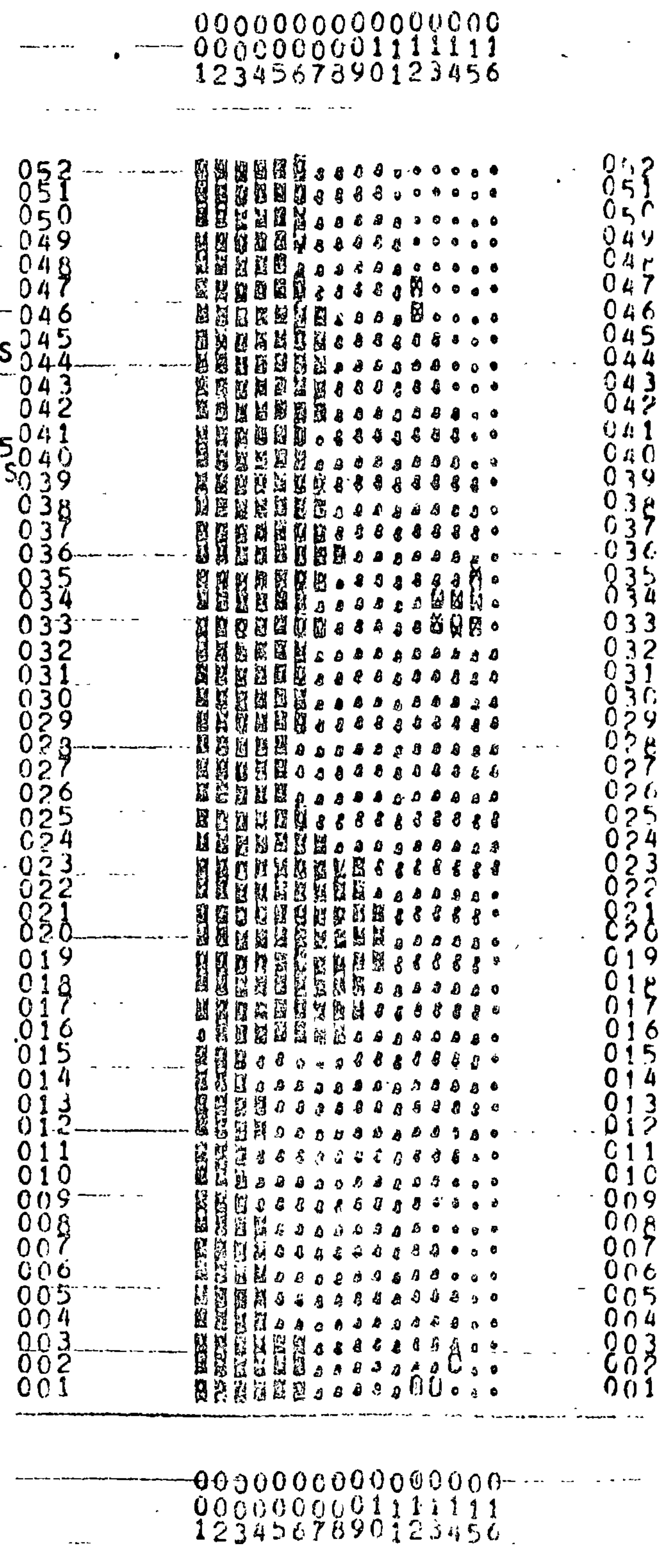
AREA DE ESTUDIO VALLE DEL PACIFICO LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

0 = NO HAY DATOS
2 = AREA ESTABLE
5 = DESLIZAMIENTOS ACTIVOS NO NATURALES
~~7 = DESLIZAMIENTOS ACTIVOS DE TIERRA~~
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

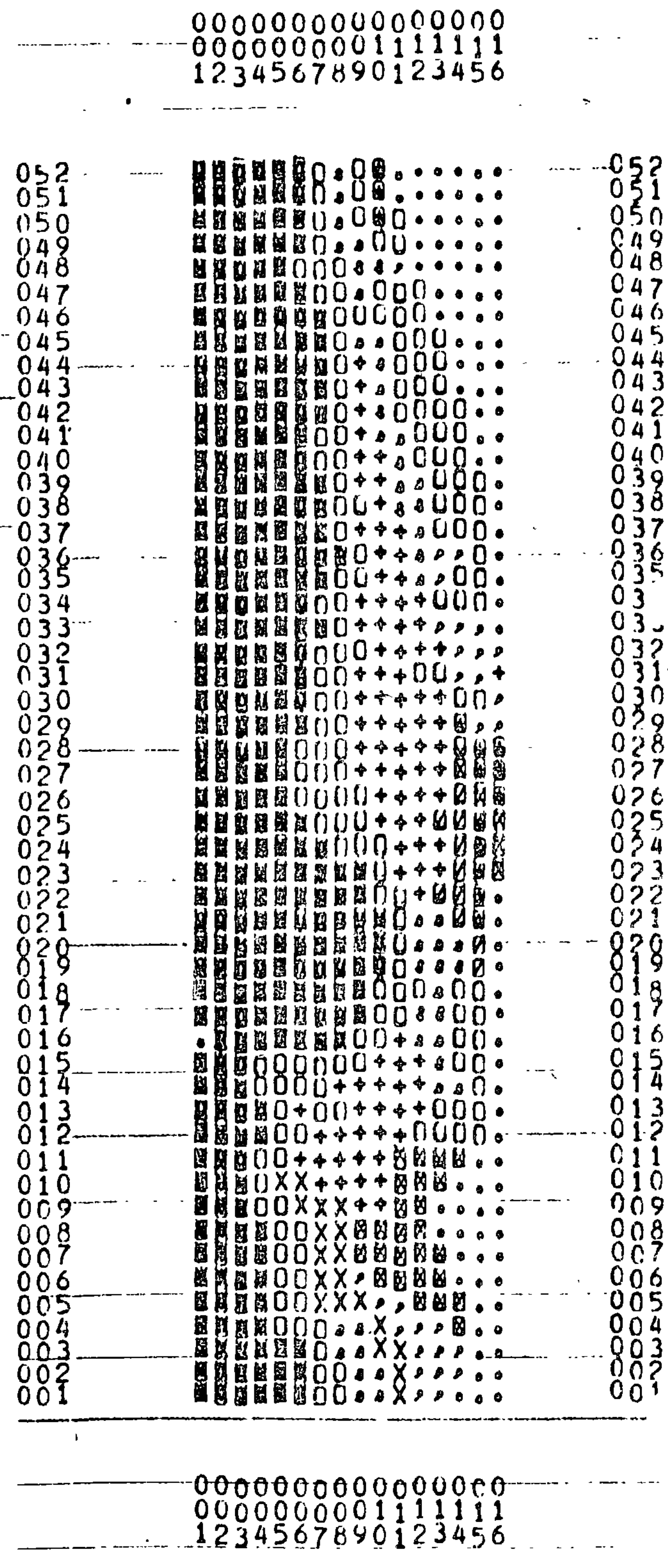
NIVELES	0	1	2	3	4	5	6	7	8	9
SIMBOLOS	0	1	2	3	4	5	6	7	8	9
FRECUENCIA	98	0	405	0	0	3	0	9	0	317



MAPA DE LA VARIABLE 8 DEL INVENTARIO DE DATOS SUELOS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



SUBVARIABLES DATO

0 = NO HAY DATOS
1 = MGFG2
2 = ST
3 = LMBC
4 = CHE
5 = RHGH
6 = GZF
7 = GRGH
8 = GZF
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2	3	4	5	6	7	8	9
FRECUENCIA	99	29	47	96	19	169	10	32	14	317

MAPA DE LA VARIABLE 9 DEL INVENTARIO DE DATOS PROXIMIDAD AL AGUA

AREA DE ESTUDIO VALLE DEL PACIFICO

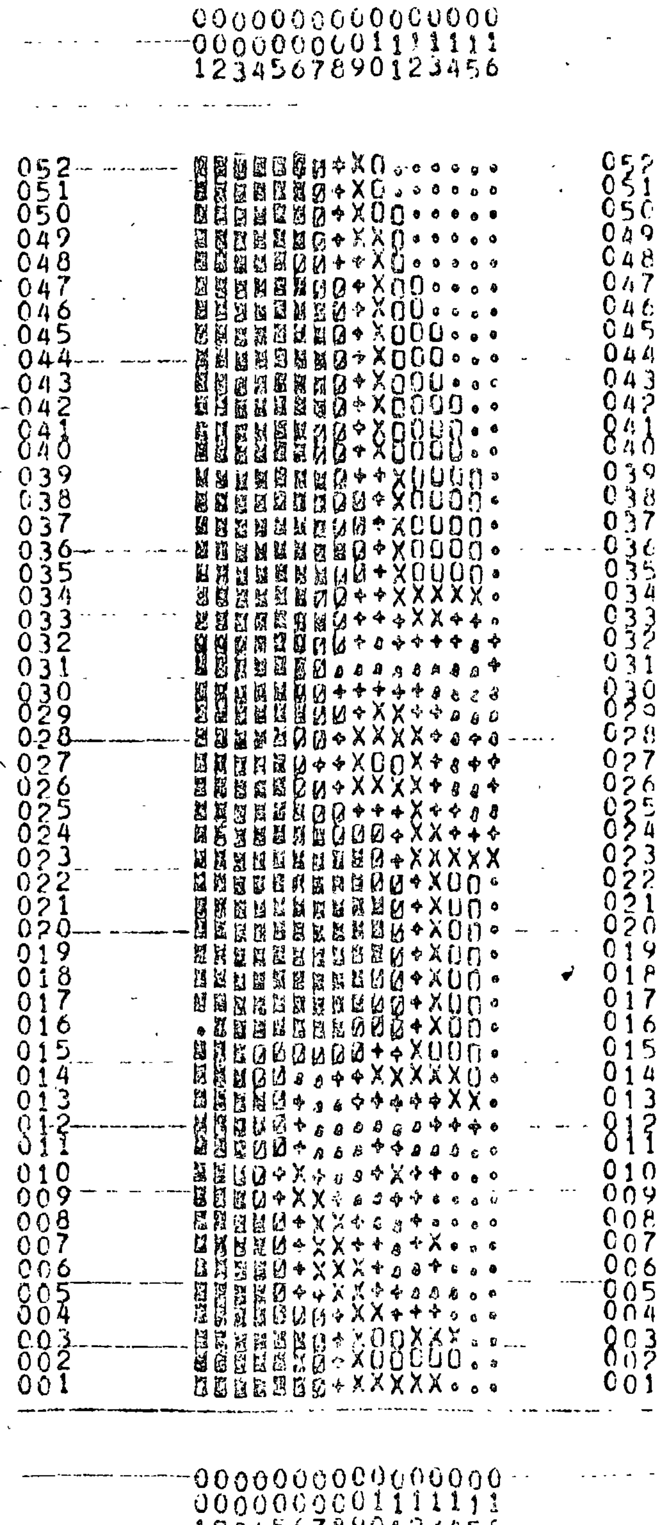
LABORATORIO DE PLANEACION URBANA

SUBVARIABLES DATO

0 = NO HAY DATOS
2 = CORRIENTES DENTRO DE LA CELULA
3 = AGUA EN LAS CELULAS ADYACENTES
4 = AGUA A UNA CELULA DE PROXIMIDAD
5 = AGUA A DOS CELULAS DE PROXIMIDAD
6 = ZONA DE MAREA
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE 2.5 ACRES

NIVELES	0	1	2	3	4	5	6	7	8	9
FRECUENCIA	99	0	53	117	87	78	81	0	0	317



MAPA DE LA VARIABLE 10 DEL INVENTARIO DE DATOS VISTAS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

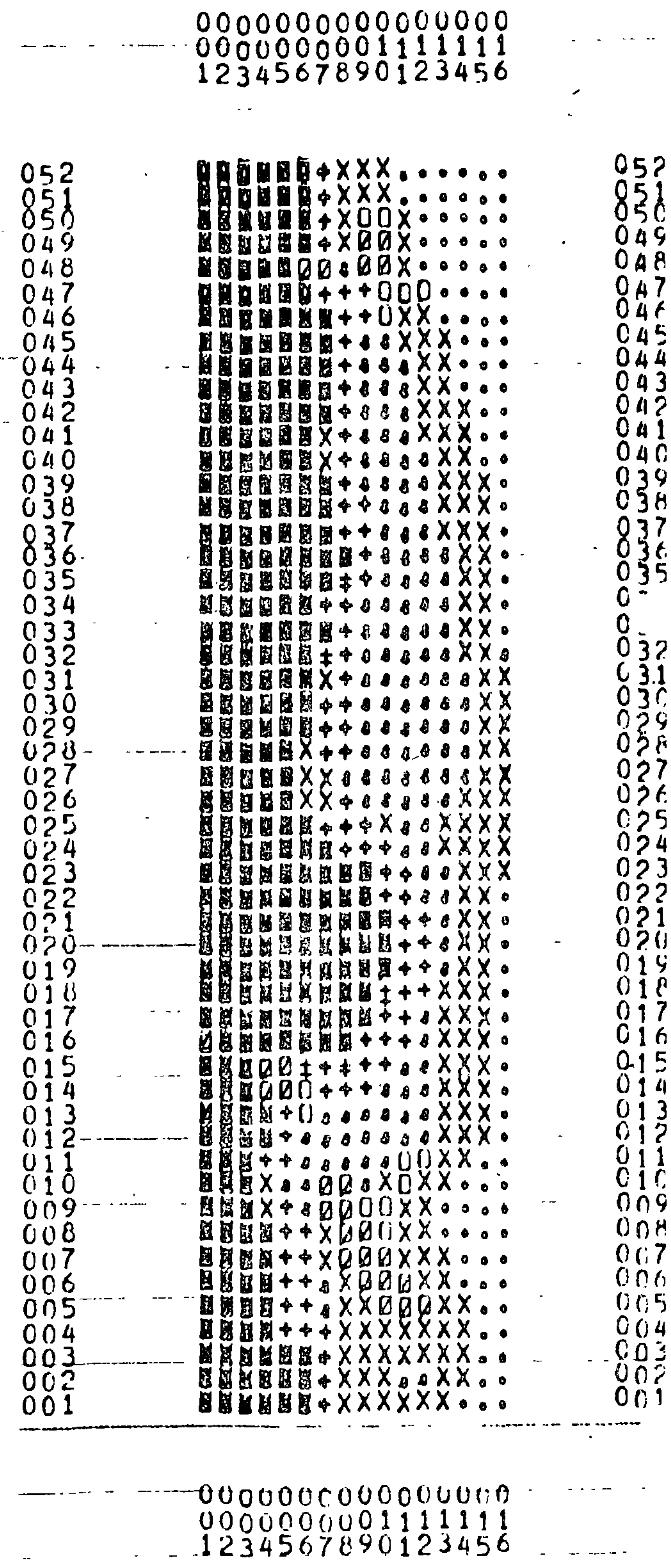
SUBVARIABLES DATO

0 = NO HAY DATOS
2 = NO HAY VISTAS IMPORTANTES
3 = BUENAS VISTAS AL OCEANO
4 = EXCELENTES VISTAS AL OCEANO
5 = BUENAS PANORAMICAS DEL LUGAR
6 = EXCELENTES PANORAMICAS DEL LUGAR
9 = OCEANO

EL TAMANO DE LA CELULA DE LA MALLA ES DE

2.5 ACRES

NIVELES	0	1	2	3	4	5	6	7	8	9
SIMBOLOS	0	1	2	3	4	5	6	7	8	9
FRECUENCIA	98	0	133	87	157	14	26	0	0	317



II. MODELOS DE ATRACTIVIDAD.

Para analizar cuáles son los sitios mas adecuados para localizar cada uno de los usos del suelo, deben seguirse los siguientes pasos:

1. Enlistar las variables contenidas en el banco de datos que representan los factores de localización del uso del suelo considerado. - Pueden considerarse hasta 10 variables.
2. Diferenciar la importancia de los factores de localización, asignándole a cada variable un peso de acuerdo a la siguiente escala:

   1 = menos importante.
   2 = importante.
   3 = muy importante.
3. Asignar preferencias para cada uno de los niveles ó categorias en que se sub-dividen las variables, con una escala del 0 al 9, donde

   0 = nada deseable.
   9 = lo más deseable.
4. Codificación del modelo.
5. Correr los modelos.
   Programa ATRACTIVO/IMGRID.
6. Mapear los resultados
   Programa MAPAS/IMGRID.
7. Analizar los resultados y modificar los modelos.

III. MODELOS DE ATRACTIVIDAD

Estos modelos calculan un índice que representa la atractividad de cada una de las celdillas de la retícula para alojar un uso del suelo determinado. Este índice toma valores del 0 al 9, e indica que tanto se satisfacen los requerimientos de localización del uso considerado en una celdilla específica.

Paso 1.- Escoger las variables relevantes

Deben identificarse las variables del banco de datos que sean mas representativas de los factores de localización del uso del suelo considerado. Puede incluirse un máximo de diez variables en un modelo.

Paso 2.- Ponderar las variables

En la mayoria de los casos, no todas las variables incluidas en el modelo tienen la misma importancia en la localización del uso del suelo considerado.

Puede darse prioridad a más variables (factores de localización) sobre otras ponderándolas de acuerdo a la siguiente escala:

1 = menos importante
2 = importante
3 = muy importante

Paso 3.- Asignar preferencias a los niveles ó categorias de cada variable

Se debe asignar un número del 0 al 9 a cada uno de los niveles ó categorias de las variables, indicando así una mayor ó menor preferencia de acuerdo a los requerimientos específicos del uso del suelo considerado. '0' significa nada deseable, '9' significa lo mas deseable. Ver figura

Paso 4.- Codificación del modelo

Para cada modelo de atractividad se debe perforar las siguientes tarjetas:

1a. tarjeta:

Col. 1-2 — Número del modelo

2a. tarjeta:

Col. 1-2 — Número de variables que intervienen en el modelo.*

3a. tarjeta

Col. 1-2	Número de la primera variable *
Col. 10 - 12 - 14 16 - 18 - 20 22 - 24 - 26 28	Preferencias asignadas a los niveles o categorias. (Paso 3). Dado en orden ascendente *

Tarjetas siguientes:

Lo mismo que la 3a. tarjeta.
Una tarjeta por cada variable incluido en el modelo

Paso 5.- Correr los modelos

La forma de preparar el paquete para alimentar el programa ATRACTIVO/IMGRID esta ilustrado en la figura.

El programa analiza cada una de las celdillas y le asigna un índice de atractividad en función de la importancia y de las preferencias asignadas. Estos índices son almacenados en archivos de disco magnético, los cuales serán utilizados para su mapeo y en la evaluación de planes de uso del suelo.

Una vez ejecutado, el programa manda un mensaje como el de la figura.

Paso 6.- Mapear los resultados

Debe producirse un mapa para cada modelo mediante el programa MAPAS/IMGRID.

La atractividad de cada celdilla se representa con una gama de diez tonos de gris lograda mediante sobre-impresión de carácteres, en la que el tono mas obscuro significa una mayor atractividad.

La forma de especificar los mapas se explica en el capítulo VI. La manera de alimentar el programa se ilustra en la figura.

En las figuras se ejemplifica el mapeo de los modelos de atractividad.

---

* Los números en estas tarjetas deben ser perforados como enteros sustituidos a la derecha.
La figura muestra un ejemplo para forma de codificación de los modelos de atractividad.

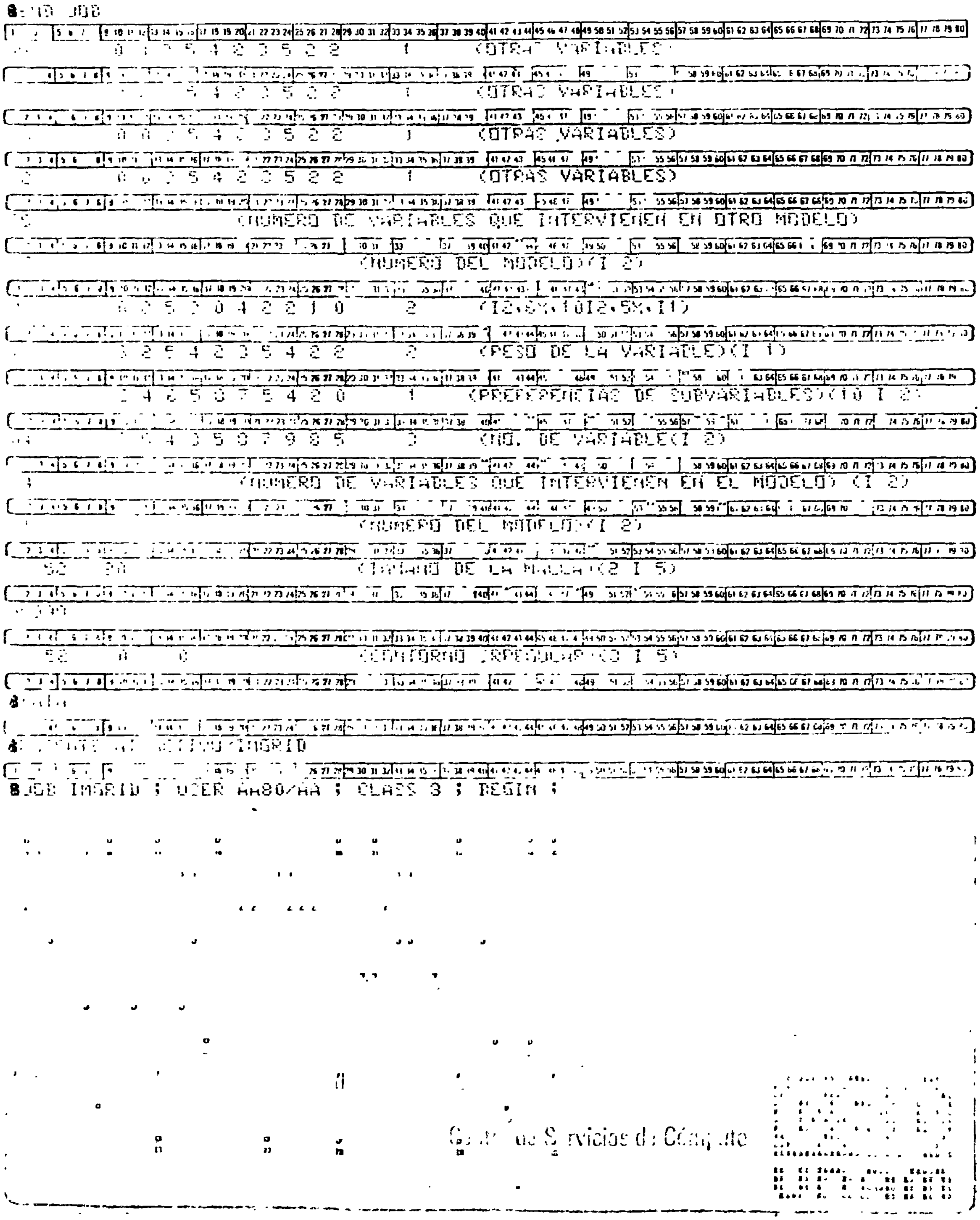


Figura . forma de ordenar su paquete de tarjetas para utilizar el programa ATRACTIVO/IMGRID en el sistema B-6700 del CSC de la UNAM.

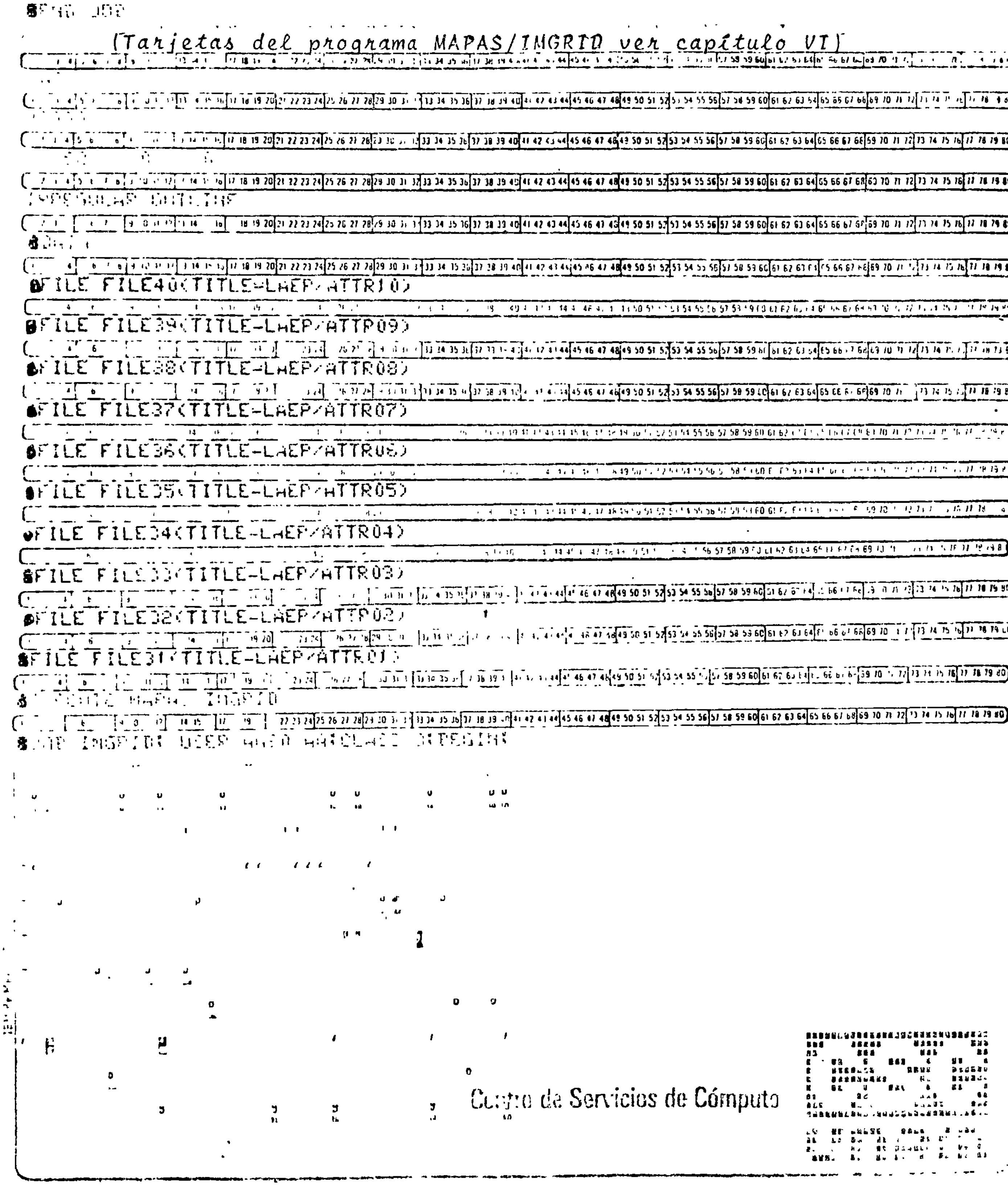


Figura 6.- Forma de obtener los mapas de los "modelos de atractividad" con el programa MAPAS/IMGRID. Este ejemplo muestra la manera de obtener 10 mapas (FILE31,..,40) a partir de los archivos (LAEP/ATTR01,...,10) que fueron generados por el programa ATRACTIVO/IMGRID.

*** MODELO DE ATRACTIVIDAD NUMERO 1***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAEP/ATTR 1

EL NUMERO DE VARIABLES EN EL MODELO ES 4

ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

10	0	0	0	6	8	7	9	0	0	0	2
6	0	0	0	6	6	9	0	9	0	0	2
9	0	0	9	7	4	0	5	0	0	0	2
2	0	0	0	9	0	8	0	5	0	0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 1: ACTIVIDADES DE ESPARCIMIENTO

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

1 EL TAMAÑO DE LA MALLA ES 52 RENGLONES Y 16 COLUMNAS
EL TAMAÑO DE LA CELULA ES 1 CARACTERES EN SENTIDO VERTICAL
Y 1 CARACTERES EN SENTIDO HORIZONTAL

7 LOS SIMBOLOS SON
.,,+XCCOCC123456789
.+XCCOO0
/X*X
+V

10 EL TEXTO DEL MAPA ES ----------
VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
10	VISTAS	2
6	ZONAS LLAGAS	2
9	PROXIMIDAD AL AGUA	2
2	PORCIENTO DE PENDIENTE	1

13 LA NUMERACION DE LA MALLA COMIENZA EN 1 52

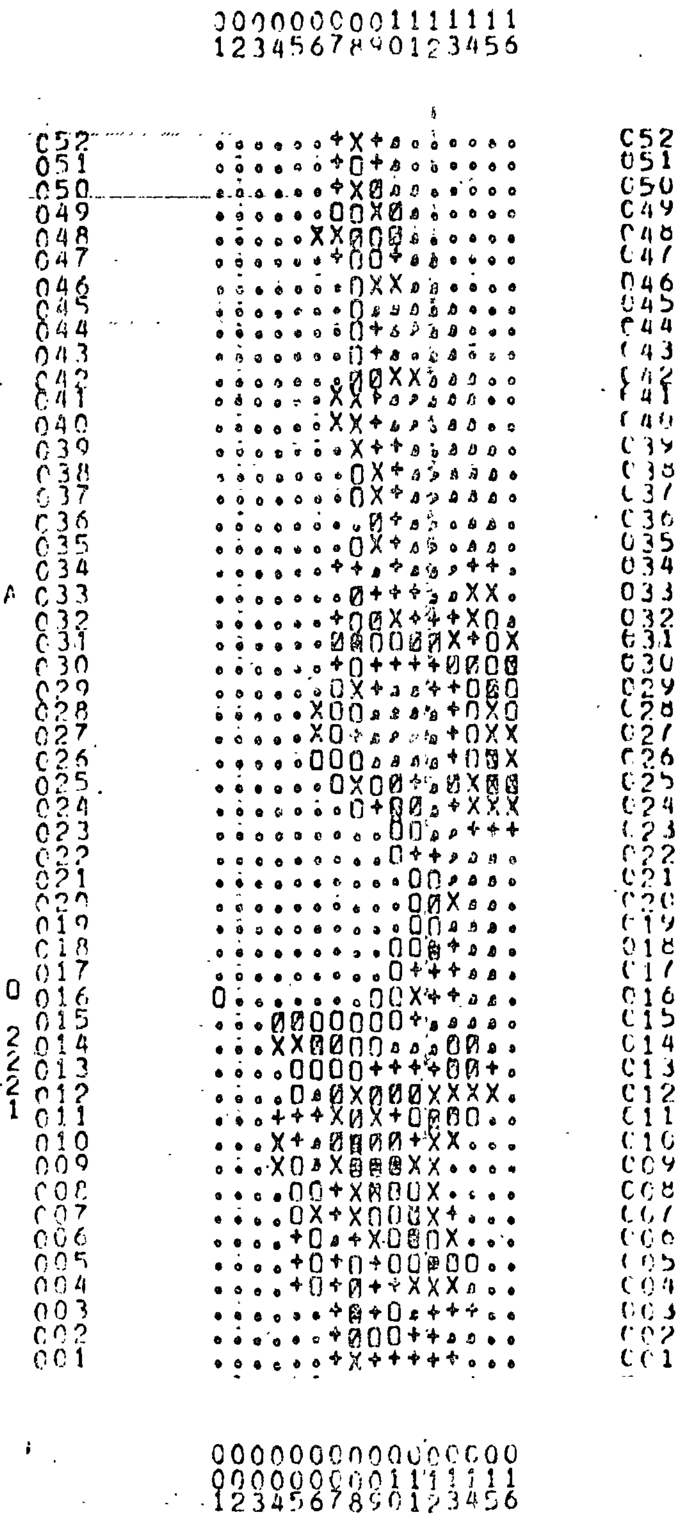
14 SE SUPONE QUE LOS DATOS ESTAN PRE-ESCALADOS

MODELO DE ATRACTIVIDAD # 1

ACTIVIDADES DE ESPARCIMIENTO

AREA DE ESTUDIO   VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

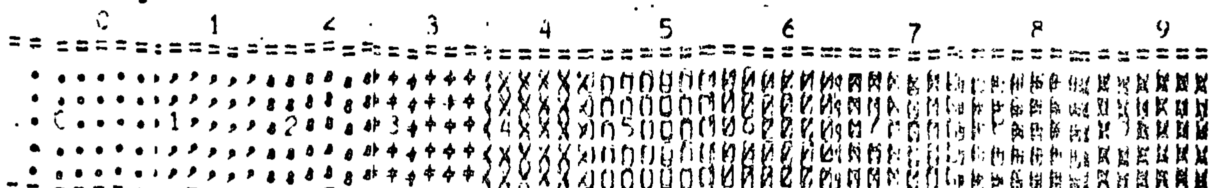


VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
10	VISTAS	2
6	ZONAS LLANAS	2
9	PROXIMIDAD AL AGUA	2
2	PORCIENTO DE PENDIENTE	1

NIVELES 0 1 2 3 4 5 6 7 8 9

SIMBOLOS



FRECUENCIA 420 13 101 94 64 82 33 16 9 1

*** MODELO DE ATRACTIVIDAD NUMERO 2***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAFP/ATTR 2
EL NUMERO DE VARIABLES EN EL MODELO ES 4
ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

2	0 0 0 9 0 2 0 0 0 0	2
5	0 0 0 7 9 6 0 5 0 0	1
6	0 0 3 0 0 6 0 9 0 0	1
8	0 9 0 9 0 9 0 9 0 0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 2: ESTACIONAMIENTOS
AREA DE ESTUDIO VALLE DEL PACIFICO
LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

VARIABLES DATO USADAS EN EL MODELO

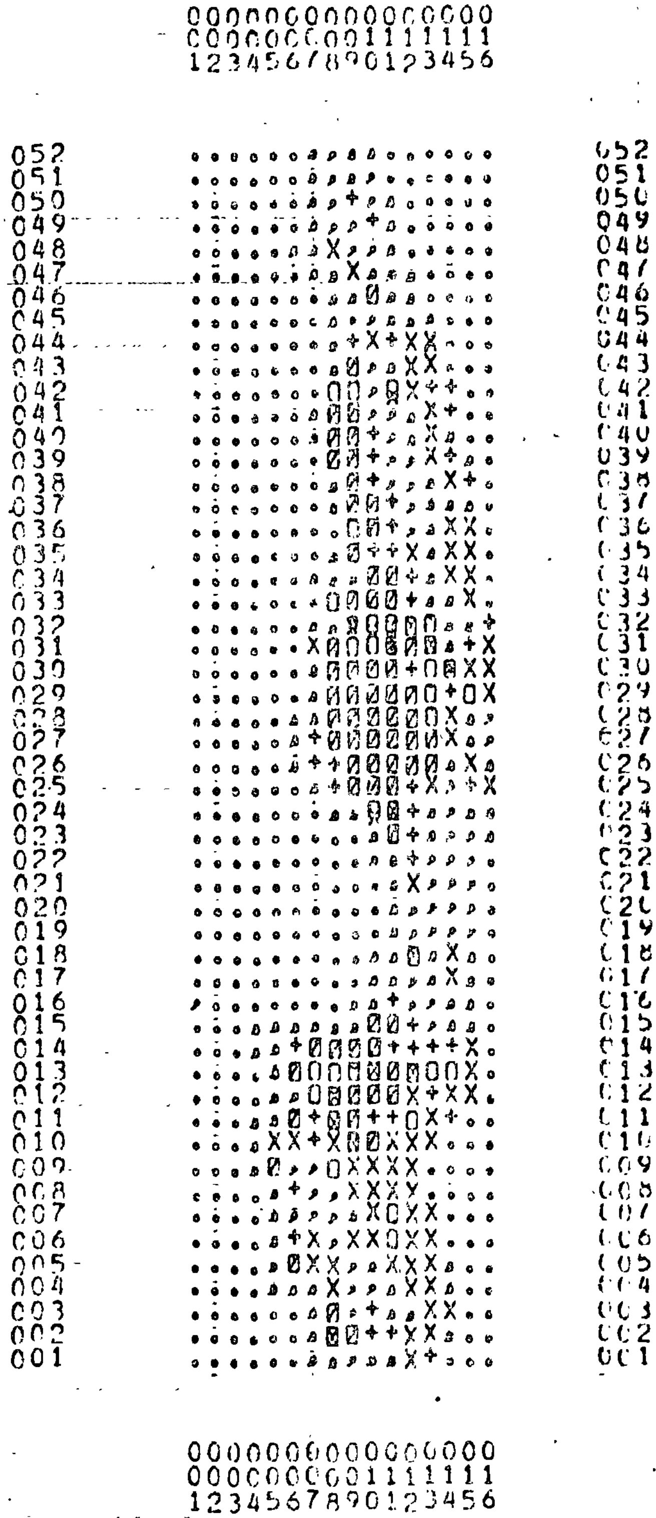
NO	NOMBRE	PESO
3	PORCIENTO DE PENDIENTE	2
5	PORCIENTO DENSIDAD DE ARBOLES	1
6	ZONAS LLANAS	1
8	SUELOS	1

MODELO DE ATRACTIVIDAD # 2

ESTACIONAMIENTOS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
3	PORCIENTO DE PENDIENTE	2
[illegible]	PORCIENTO DENSIDAD DE ARBOLES	1
[illegible]	ZONAS LLANAS	1
8	SUELOS	1

NIVELES	0	1	2	3	4	5	6	7	8	9
FRECUENCIA	419	59	129	56	74	16	66	21	2	1

*** MODELO DE ATRACTIVIDAD NUMERO 3***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAEP/ATTR 3

EL NUMERO DE VARIABLES EN EL MODELO ES 4

ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

2	0	0	0	9	0	7	0	0	0	0	1
5	0	0	0	6	8	9	0	7	0	0	1
6	0	0	0	0	0	9	0	7	0	0	1
7	0	0	9	0	0	0	0	0	0	0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 3: ESTRUCTURAS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

VARIABLES DATO USADAS EN EL MODELO

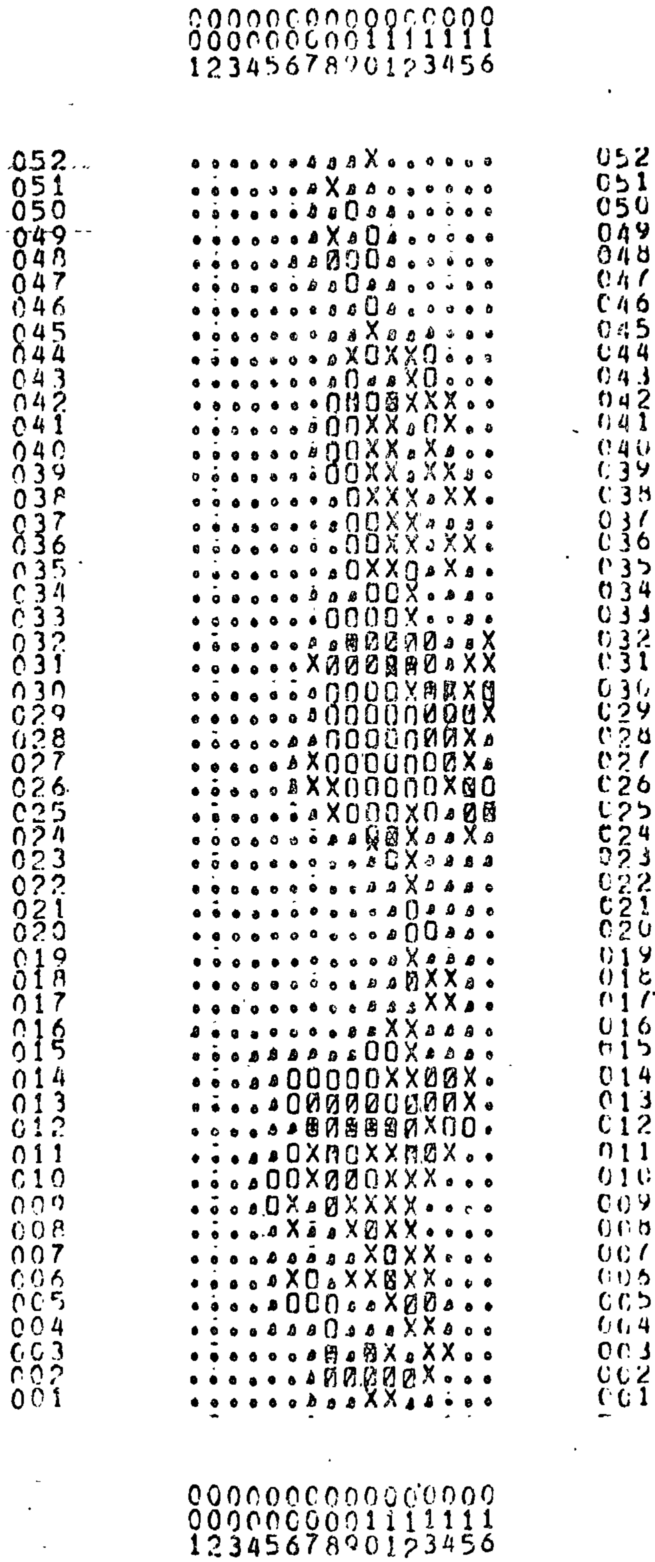
NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	1
5	PORCIENTO DENSIDAD DE ARBOLES	1
6	ZONAS LLANAS	1
7	ACCIDENTES GEOLOGICOS	1

MODELO DE ATRACTIVIDAD # 3

ESTRUCTURAS

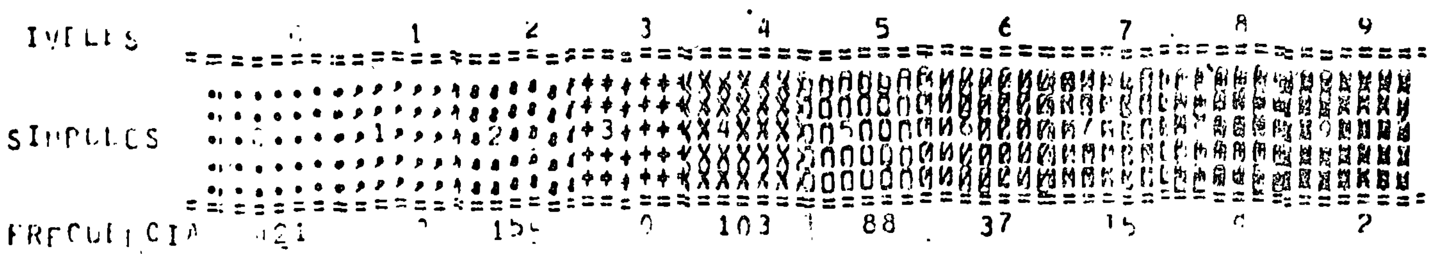
AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	1
5	PORCIENTO DENSIDAD DE ARBOLES	1
6	ZONAS LLANAS	1
7	ACCIDENTES GEOLOGICOS	1



*** MODELO DE ATRACTIVIDAD NUMERO 4***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAEP/ATTR 4

EL NUMERO DE VARIABLES EN EL MODELO ES 2

ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

2	0 0 0 9 0 5 0 0 0 0	1
9	C 0 0 2 5 9 0 0 0 0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 4: CABALLERIZAS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

VARIABLES DATO USADAS EN EL MODELO

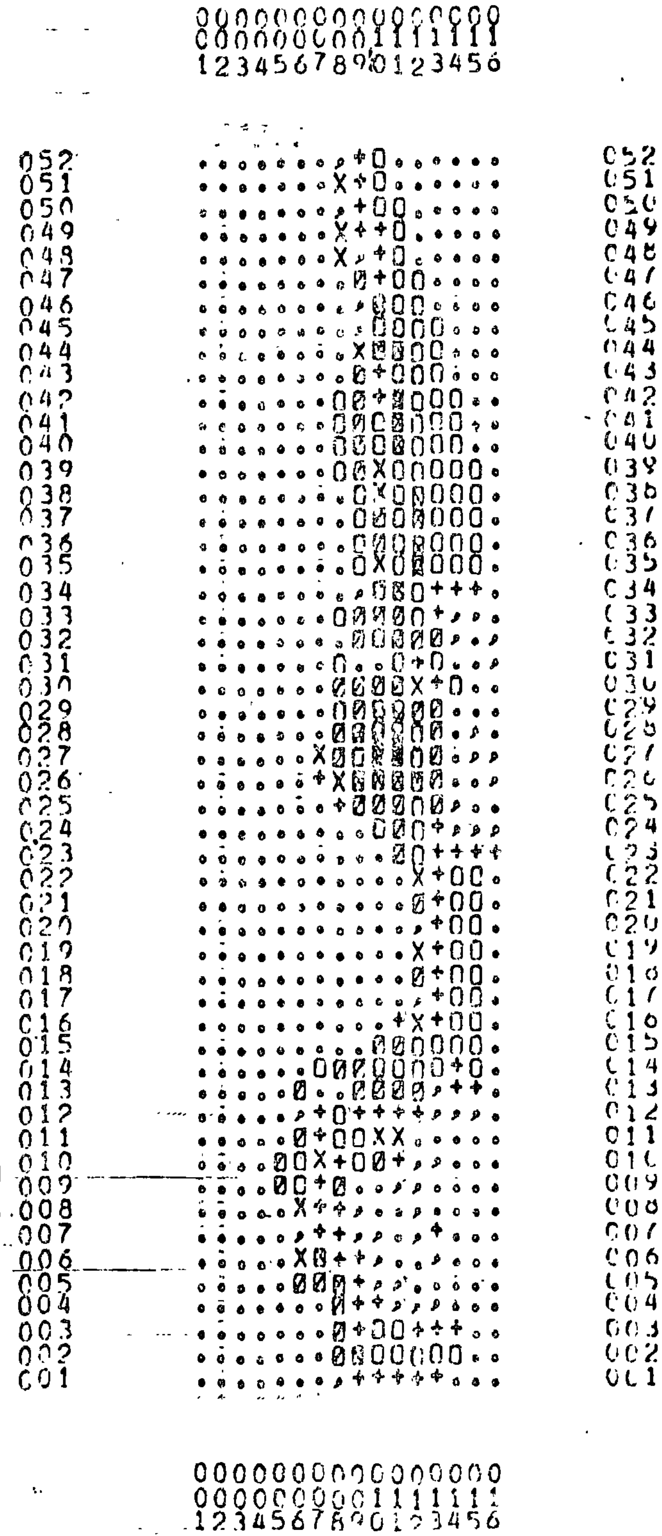
NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
9	PROXIMIDAD AL AGUA	1

MODELO DE ATRACTIVIDAD # 4

CABALLERIZAS

AREA DE ESTUDIO VALLE DEL PACIFICO

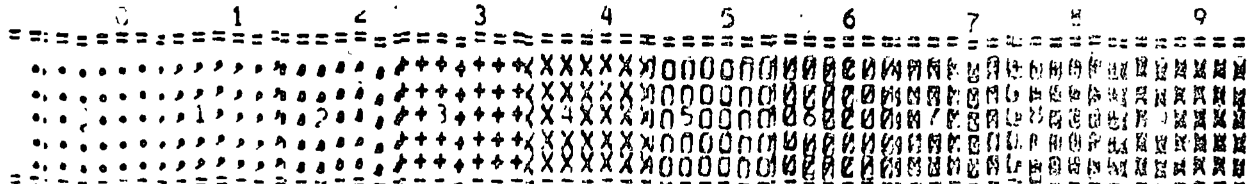
LABORATORIO DE PLANEACION URBANA



VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
9	PROXIMIDAD AL AGUA	1

NIVELES	0	1	2	3	4	5	6	7	8	9
SIMBOLOS										
FRECUENCIA	617	44	9	61	1[illegible]	107	55	26	[illegible]	4



*** MODELO DE ATRACTIVIDAD NUMERO 5***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAFP/ATTR 5

EL NUMERO DE VARIABLES EN EL MODELO ES 6

ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

2	0 0 0 9 0 3 0 0 0 0	2
5	0 0 0 6 8 9 0 8 0 0	2
6	0 0 0 0 0 9 0 9 0 0	2
4	0 0 0 0 6 9 0 0 0 0	1
9	0 0 9 9 7 3 0 0 0 0	1
1	0 9 5 0 0 0 5 0 9 0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 5! ESTACIONAMIENTO PARA TRAILERS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

VARIABLES DATO USADAS EN EL MODELO

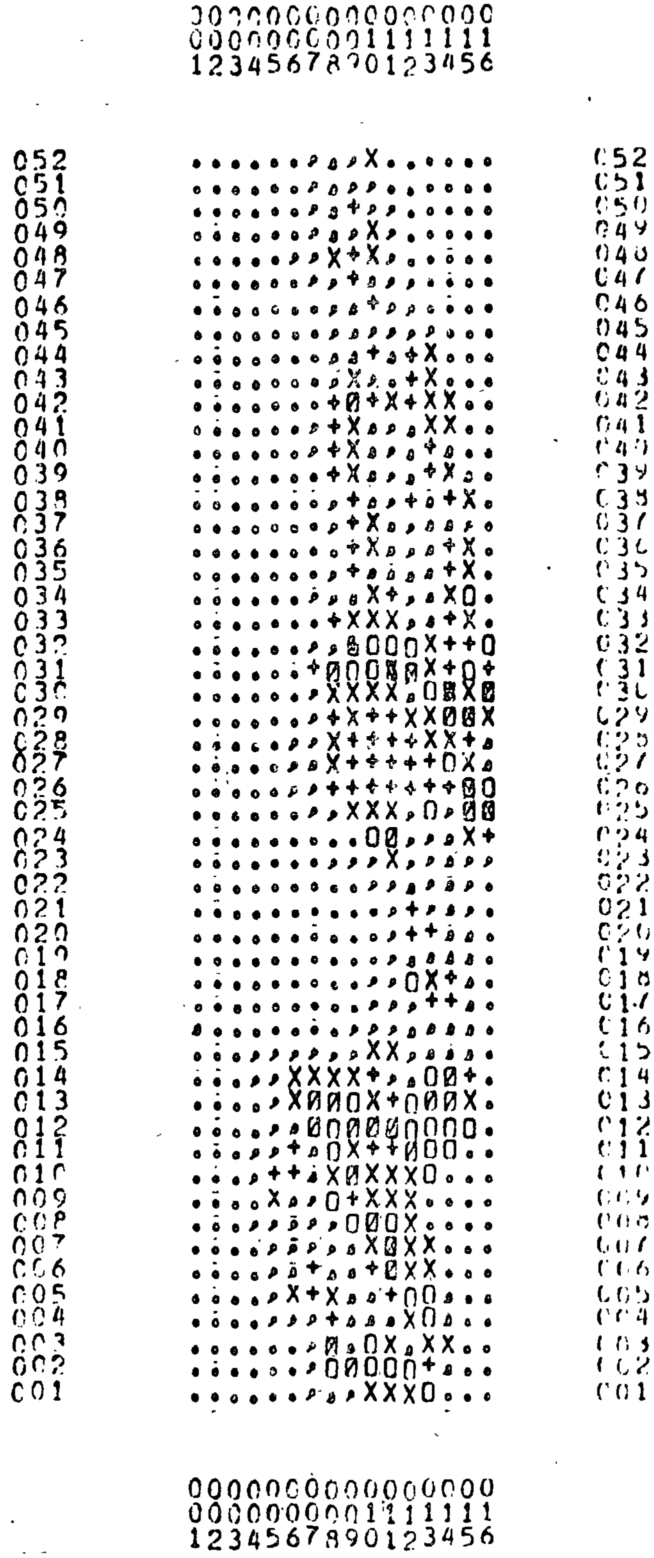
NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
5	PORCIENTO DENSIDAD DE ARBOLES	2
6	ZONAS LLANAS	2
4	VEGETACION POR ZONA	1
9	PROXIMIDAD AL AGUA	1
1	USOS DEL SUELO ACTUALES	1

MODELO DE ATRACTIVIDAD # 5:

ESTACIONAMIENTO PARA TRAILERS

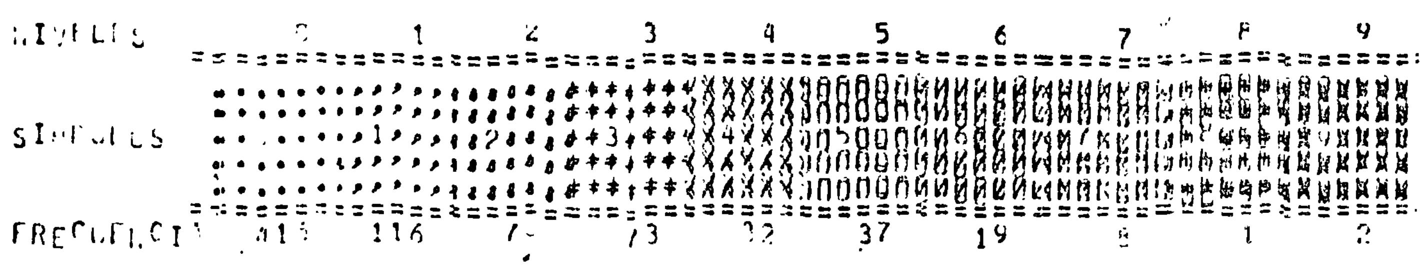
AREA DE ESTUDIO   VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
5	PORCIENTO DENSIDAD DE ARBOLES	2
6	ZONAS LLANAS	2
4	VEGETACION POR ZONA	1
9	PROXIMIDAD AL AGUA	1
1	USOS DEL SUELO ACTUALES	1



*** MODELO DE ATRACTIVIDAD NUMERO 6***

LOS INDICES DE ATRACTIVIDAD SON PUESTOS EN EL ARCHIVO EN DISCOLAEP/ATTR 6

EL NUMERO DE VARIABLES EN EL MODELO ES 5

ESTAS SON LAS TARJETAS DE LA HOJA DE CODIFICACION USADA EN EL MODELO :

2	C 0 0 9 0 8 0 4 0 0	2
6	C 0 C 0 C 9 0 8 0 0	2
5	C 0 0 5 8 9 0 8 0 0	1
9	C 0 9 9 5 0 0 0 0 0	1
4	C 0 0 C 5 9 0 0 0 0	1

TITULO DEL MAPA

MODELO DE ATRACTIVIDAD # 6' CAMTHAIA

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA
----

10 EL TEXTO DEL MAPA ES ------------

VARIABLES DATO USADAS EN EL MODELO

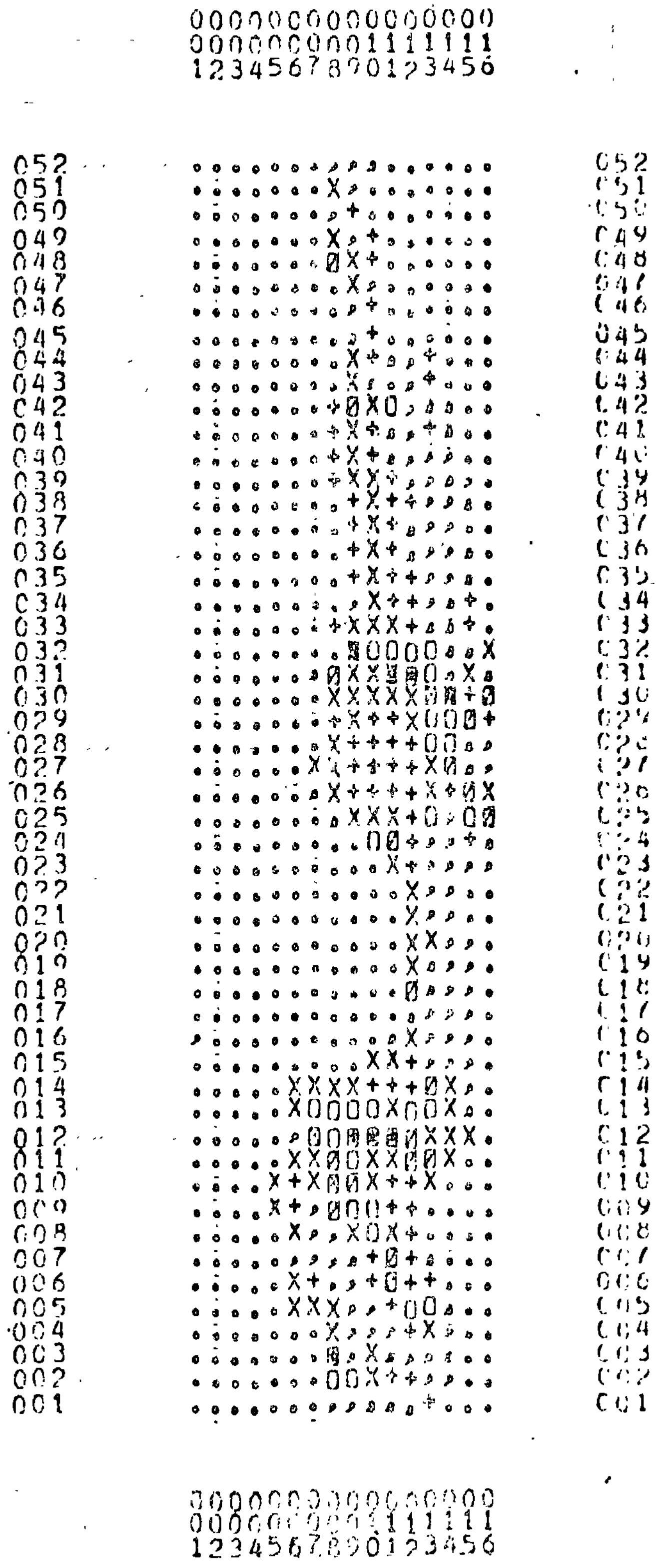
NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
6	ZONAS LLANAS	2
5	PORCIENTO DENSIDAD DE ARBOLES	1
9	PROXIMIDAD AL AGUA	1
4	VEGETACION POR ZONA	1

MODELO DE ATRACTIVIDAD # 6:

CAMINATA

AREA DE ESTUDIO   VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

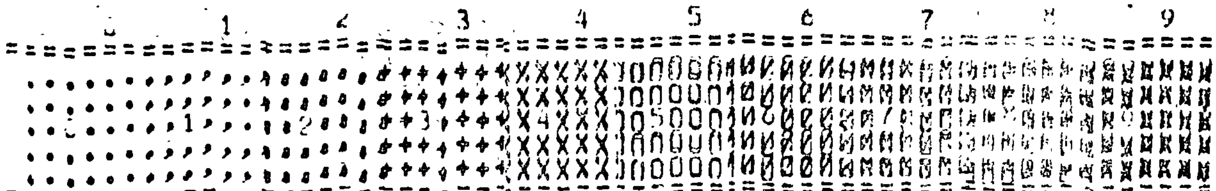


VARIABLES DATO USADAS EN EL MODELO

NO	NOMBRE	PESO
2	PORCIENTO DE PENDIENTE	2
[illegible]	ZONAS LLANAS	2
[illegible]	PORCIENTO DENSIDAD DE ARBOLES	1
[illegible]	PROXIMIDAD AL AGUA	1
4	VEGETACION POR ZONA	1

NIVELES 0 1 2 3 4 5 6 7 8 9

SIMBOLOS



FRECUENCIA 497 73 42 76 31 23 19 3 5 3

IV. MODELOS DE IMPACTO.

Mediante estos modelos es posible analizar los efectos probables que tendrán los usos del suelo propuestos sobre diferentes sistemas biológicos y físicos existentes en el -- área. Estos efectos se miden de acuerdo al grado de impacto negativo, jerarquizándolos en compatible, moderado, severo y terminal. El método consiste principalmente en la- asignación y combinación de sensitividades de las varia--- bles que intervienen en el modelo mediante el uso de matrices.

Paso 1.- Analizar detalladamente las compoentes del sistema y el impacto que se desea medir.

Una descripción detallada del sistema sobre el que se desea evaluar el impacto de los usos del suelo pro--- puestos, ayuda a elegir las variables que mejor lo representan.

Paso 2.- Elegir las variables que intervienen en el modelo

Por razones prácticas, sólo deben considerarse tres - variables para describir un sistema. Estas deben ser enlistadas en orden de imprtancia decreciente: más importante, importante y menos importante.

Paso 3.- Diferenciar la sensitividad de los niveles o categorías de las variables.

Deben agruparse los niveles o categorías de cada va-- riable que interviene en el modelo de acuerdo al grado de sensitividad al impacto considerado. Los gra-- dos de sensitividad en el modelo se especifican de la siguiente manera:

1 = Alta sensitividad
2 = Media
3 = Baja
4 = Nula

En las hojas para codificar los modelos debe escribirse el número de cada nivel o categoría en la casilla- correspondiente a su grado de sensitividad. A continuación, debe anotarse el grado de sensitivad de cada nivel o categoría dibajo de la columna correspondiente

Paso 4.- Combinar las sensitividades de las variables importante y menos importantes.

Se debe asignar un grado de sensitividad conjunta a cada una de las combinaciones posibles de sensitividades de los niveles ó categorías de estas dos variables. La estimación de estos grados de sensitividad "combinada" se hace con la jerarquía:

1 = Alta sensitividad
2 = Media
3 = Baja
0 = Nula

Paso 5.- Agrupar los usos del suelo

Los usos del suelo considerados deben agruparse de acuerdo al grado de impacto potencial ocasionado por su construcción, mantenimiento y actividades de los usuarios. Estos impactos se clasifican como:

1 = bajo
2 = medio
3 = alto

Paso 6.- Determinar la Vulnerabilidad del sistema con respecto a los tres grupos de usos del suelo.

Debe asignarse un índice de vulnerabilidad (grado de impacto negativo) del sistema para todas las combinaciones posibles de los grados de la sensitividad obtenidos en el paso 4 con los grados de sensitividad de la variable más importante. Los indices de vulnerabilidad corresponden a las siguientes categorías:

1 = compatible
2 = moderado
3 = severo
4 = terminal

Debe hacerse un análisis por separado para cada uno de los grupos de usos del suelo.

Paso 7.- Codificación del modelo

Para cada modelo se debe perforar las siguientes tarjetas:

1a. tarjeta:

Col. 1-2 Número del modelo. Perforado como entero justificado a la derecha.

2a. tarjeta:

Col. 1-9 LAEP/IMPT para indicar que se crea un archivo de impactos.

Col. 10-11 Número del modelo. Perforado como entero justificado a la derecha.

Col. 12 1 para indicar que termina el nombre del archivo.

3a. tarjeta:

Col. 1-3 Número de la variable más importante.

Col. 4-6 Número de la variable importante.

Col. 7-9 Número de la variable menos importante. Perforados como enteros y justificados a la derecha.

4a. tarjeta:

Col. 2, 4, 6, 8, 10, 12, 14, 16, 18, 20 Valores de sensitivdad de los niveles o categorías, asignados en el paso 3.

5a. tarjeta: Lo mismo que la 4a. tarjeta para la variable importante.

6a. tarjeta: Lo mismo que la 4a. tarjeta para la variable menos importante.

7a. tarjeta:

Col. 2, 4, 6, 8, 10, 12, 14, 16, 18, Grados de sensitividad conjunta de las variables importante y menos importante. (matriz del paso 4)*.

8a. tarjeta:

Col. 2, 4, 6 8, 10, 12, 14, 16, 18 Grados de impacto negativo (vulnerabilidad) para el primer grupo de usos del suelo. (1a. matríz del paso 6)*.

*Las matrices deben ser codificadas empezando por la primera columna y de arriba hacia abajo, continuando hasta registrar las nueve casillas.

9a. tarjeta: Lo mismo que la 8a. tarjeta para el segundo -- grupo de usos del suelo. (2a. matríz del paso 6)*.

10a. tarjeta: Lo mismo que la 8a. tarjeta para el tercer grupo de usos del suelo. (3a. matríz del paso 6)*

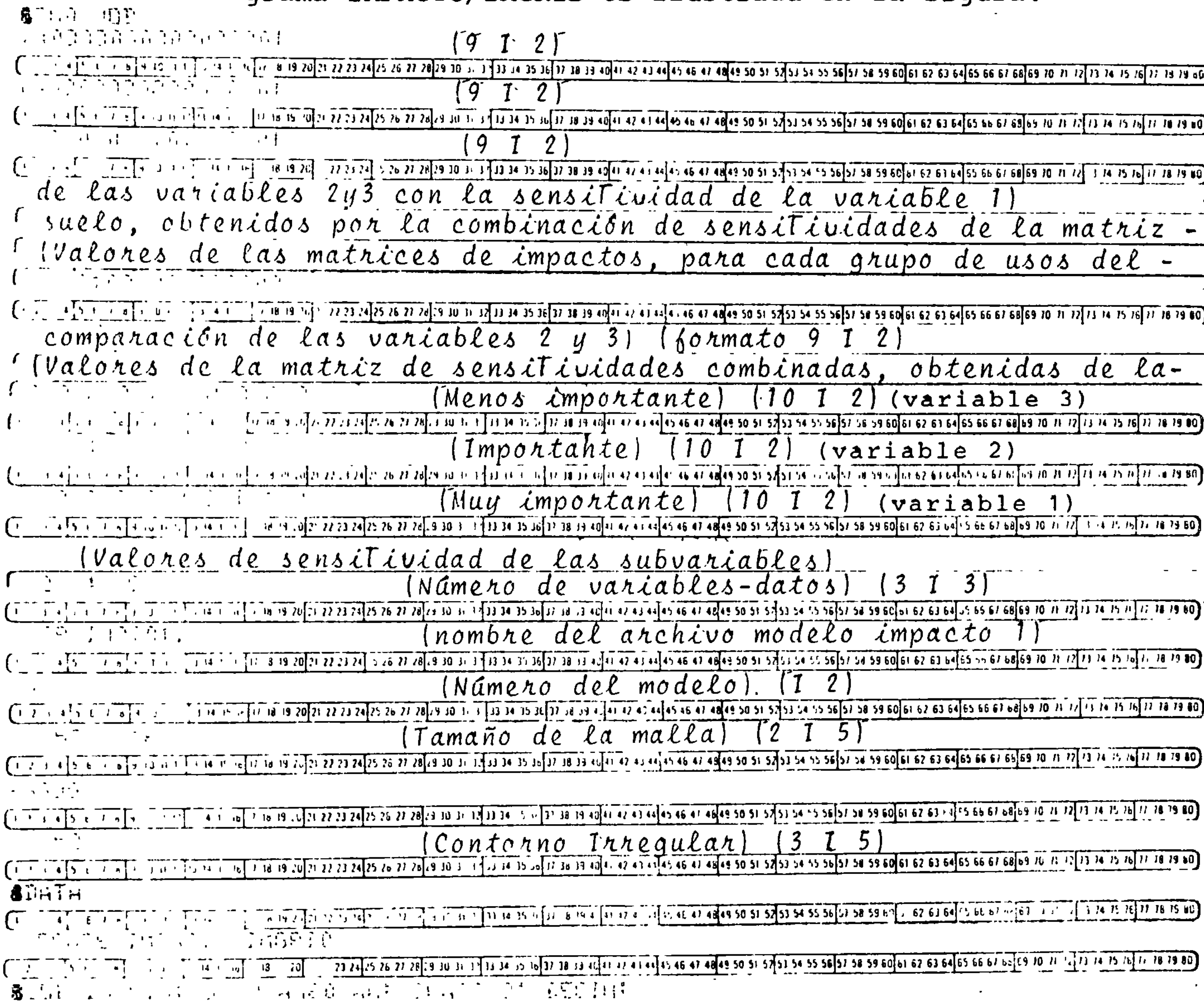
11a. tarjeta:

Col. 2, 4, 6, 8, 10, 12, ..., 40 Número de grupo de usos del suelo al que per-- tenece cada uno de los usos considerados en el estudio. (paso 5) Se deben utilizar solamente el número de columnas necesario para registrar la clasificación de todos los usos. El programa permite como máximo 20 usos diferentes.

*Las matrices deben ser codificadas empezando por la primera columna y de arriba hacia abajo, continuando hasta registrar las nueve casillas.

Paso 8.- Correr los modelos

La forma de preparar un paquete para ejecutar el programa IMPACTO/IMGRID es ilustrada en la figura:



Paso 9.- Mapear los resultados

Mediante el programa MAPAS/IMGRID se debe producir un mapa para cada modelo de impactos. La forma de especificar los mapas se explica en el capítulo VI.

TITULO DEL MAPA

MODELO DE IMPACTO N 1: EROSION

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

```
1     EL TAMAÑO DE LA MALLA ES  52 RENGLONES Y  16 COLUMNAS
      EL TAMAÑO DE LA CELULA ES   1 CARACTERES EN SENTIDO VERTICAL Y
7     LOS SIMBOLOS SON             1 CARACTERES EN SENTIDO HORIZONTAL
       .,,+X000000123456789
         .+X0000A
             /X*X
               +V

10    EL TEXTO DEL MAPA ES       ----------
                VARIABLES DATO USADAS EN EL MODELO

                    2    PORCIENTO DE PENDIENTE
                    4    VEGETACION POR ZONA
                    8    SUELOS POR TIPO
13   LA NUMERACION DE LA MALLA COMIENZA EN    1    52
14   SE SUPONE QUE LOS DATOS ESTAN PRE-ESCALADOS
```

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	CAMINOS	SERVICIO INF. TURISTICA
CAMINATA	ESTACIONAMIENTOS	CABALLERIZAS
	ESTRUCTURAS	ESTAC./ TRAILERS

43

MODELO DE IMPACTO Ñ 1: EROSION

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

VARIABLES DATO USADAS EN EL MODELO

2 PORCIENTO DE PENDIENTE
4 VEGETACION POR ZONA
8 SUELOS POR TIPO

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO
2 GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3 LOS TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6 LOS TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9 LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1

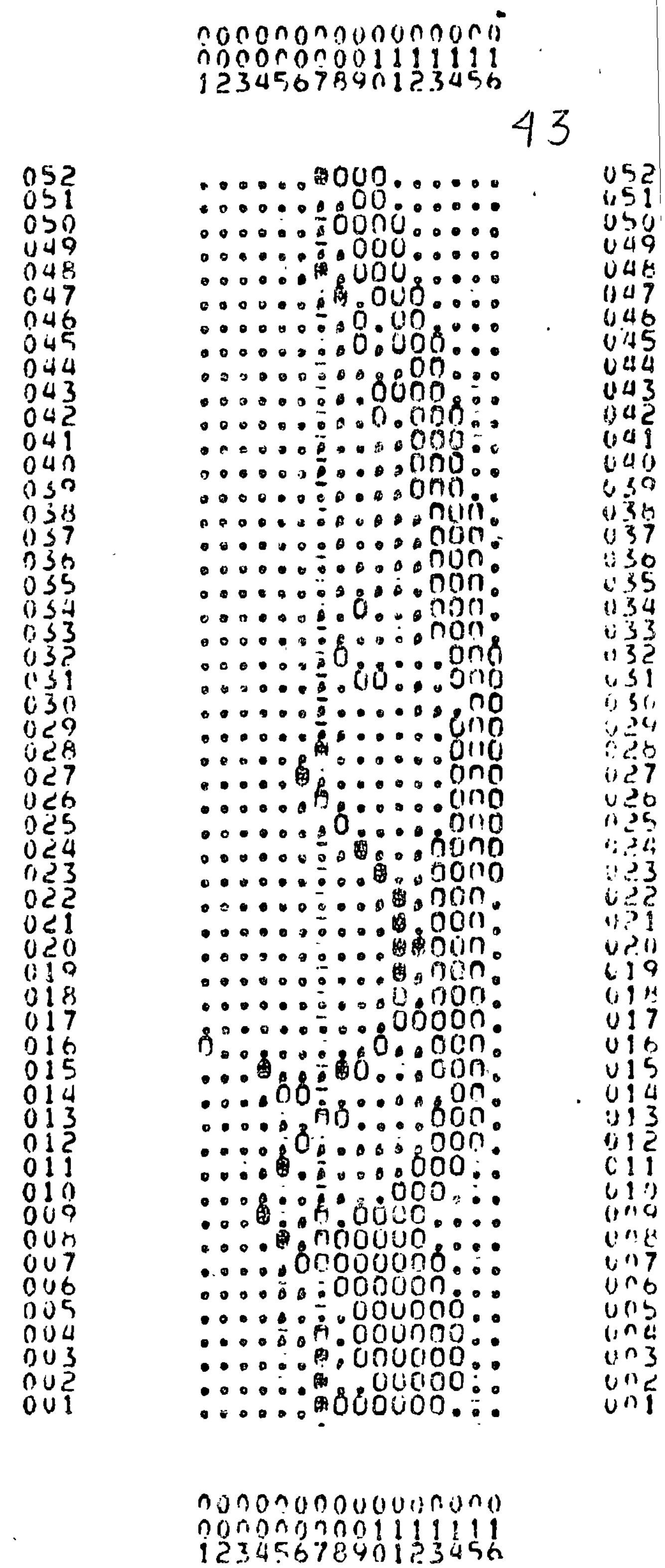
ACTIVIDADES DE ESPARC.
CAMINATA

USO DEL SUELO 11

CAMINOS
ESTACIONAMIENTOS
ESTRUCTURAS

USO DEL SUELO 111

SERVICIO INF. TURISTICA
CABALLERIZAS
ESTAC./ TRAILERS



NIVELES	0	1	2	5	8
SIMBOLOS	·········	/////////	·········	000000000	@@@@@@@@@
FRECUENCIA	50?	1	99	203	20

TITULO DEL MAPA

MODELO DE IMPACTO N 2: CAMBIO EN LA VISUALIDAD

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES ----------
VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
5 PORCIENTO DENSIDAD DE ARBOLES
2 PORCIENTO DE PENDIENTE

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	ESTRUCTURAS	ESTACIONAMIENTOS
CAMINATA	VIS	CAMINOS
CABALLERIZAS		ESTAC./ TRAILERS

MODELO DE IMPACTO N 2:

CAMBIO EN LA VISUALIDAD

AREA DE ESTUDIO   VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

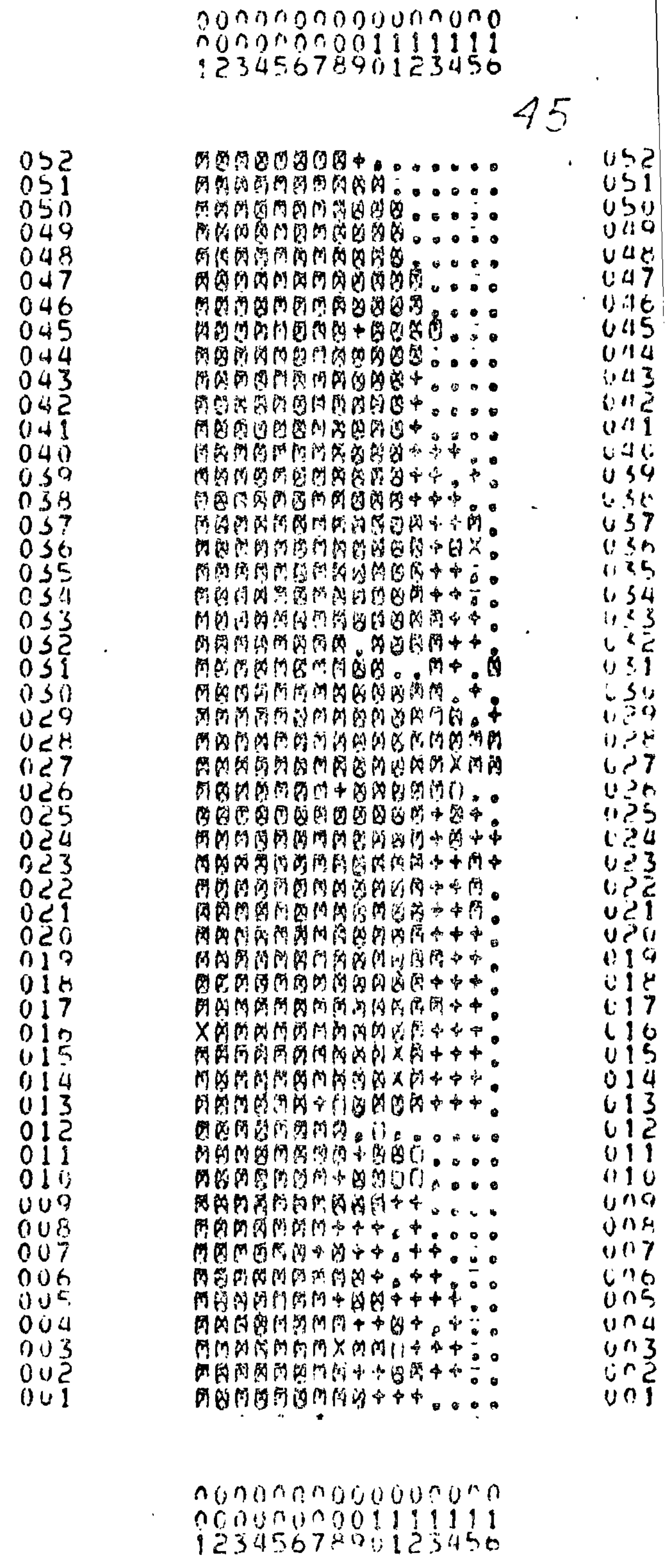
VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
5 PORCIENTO DENSIDAD DE ARBOLES
2 PORCIENTO DE PENDIENTE

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO
2 GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3 LOS TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6 LOS TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9 LOS TRES GRUPOS DE US TERMINAL



USO DEL SUELO 1

ACTIVIDADES DE ESPARC.
CAMINATA
CABALLERIZAS

USO DEL SUELO 11

ESTRUCTURAS
VIS

USO DEL SUELO 111

ESTACIONAMIENTOS
CAMINOS
ESTAC./ TRAILERS

NIVELES	0	2	3	7
SIMBOLOS	.	.	+	@
FRECUENCIA	123	11	99	586

TITULO DEL MAPA

MODELO DE IMPACTO Ñ 3: CONTAMINACION DE LA SUPERFICIE DEL AGUA
AREA DE ESTUDIO VALLE DEL PACIFICO
LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES
VARIABLES DATO USADAS EN EL MODELO

9 PROXIMIDAD AL AGUA
8 SUELOS POR TIPO
4 VEGETACION POR ZONA

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR
LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

ACTIVIDADES DE ESPARC. ESTACIONAMIENTOS CABALLERIZAS
CAMINATA ESTRUCTURAS ESTAC./ TRAILERS
CAMINOS
VIS

47

MODELO DE IMPACTO N 3:

CONTAMINACION DE LA SUPERFICIE DEL AGUA

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

VARIABLES DATO USADAS EN EL MODELO

9 PROXIMIDAD AL AGUA
8 SUELOS POR TIPO
4 VEGETACION POR ZONA

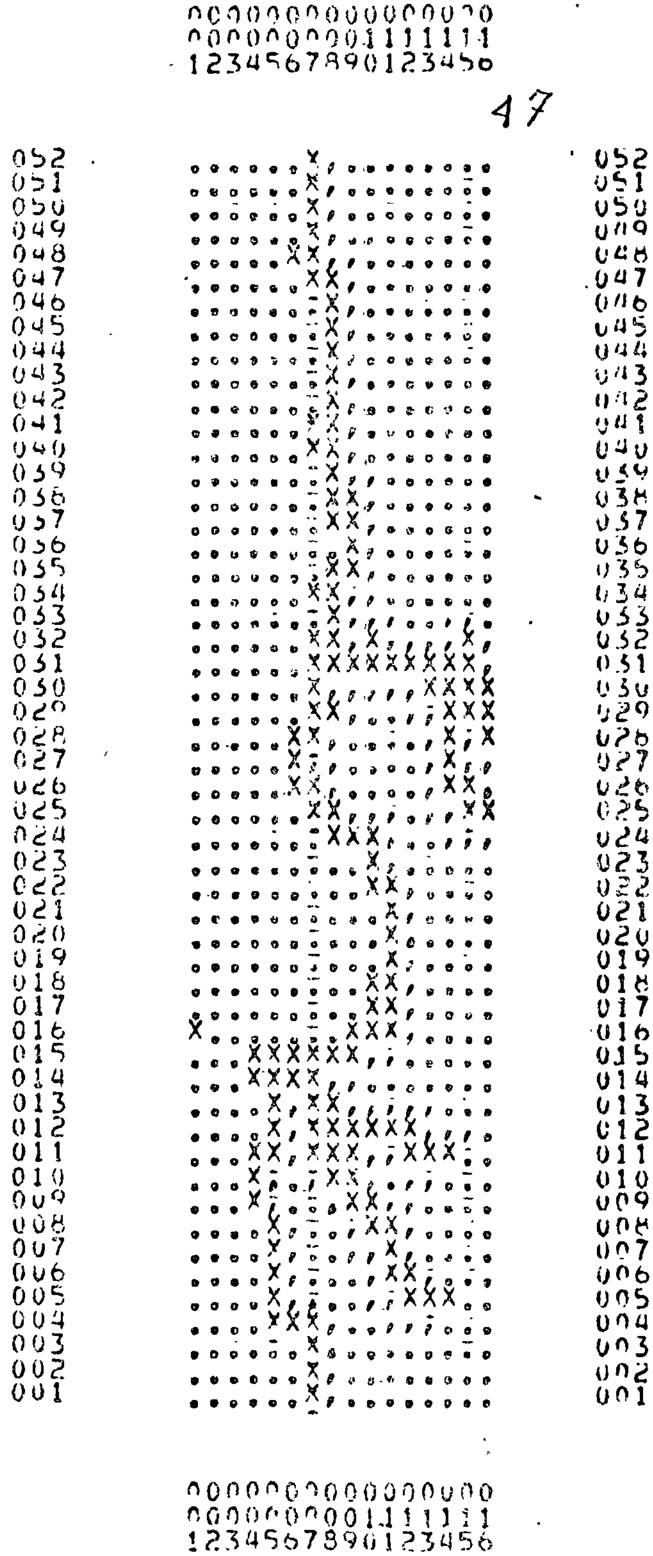
LEGENDA: EL SIMBOLO MAS OBSCURO

CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS

3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO -
2 GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3 LOS-TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6 LOS-TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9 LOS-TRES GRUPOS DE US TERMINAL



ESTACIONAMIENTOS
ESTRUCTURAS
CAMINOS
VIS

CABALLERIZAS
ESTAC./ TRAILERS

ACTIVIDADES DE ESPARC.
CAMINATA

NIVELES	0	1	2	4
SIMBOLOS	•	/	ø	X
FRECUENCIA	581	111	6	134

TITULO DEL MAPA

MODELO DE IMPACTO Ñ 4: VULNERABILIDAD A FUEGOS
AREA DE ESTUDIO VALLE DEL PACIFICO
LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES
VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
3 ORIENTACION
5 PORCIENTO DENSIDAD DE ARBOLES

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR
LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ESTACIONAMIENTOS	ACTIVIDADES DE ESPARC.	ESTAC./ TRAILERS
CABALLERIZAS	ESTRUCTURAS	CAMINATA
CAMINOS	VIS	

LABORATORIO DE PLANEACION URBANA

AREA DE ESTUDIO VALLE DEL PACIFICO.

MODELO DE IMPACTO N 4: VULNERABILIDAD A FUEGOS

VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
3 ORIENTACION
5 PORCIENTO DENSIDAD DE ARBOLES

LEGENDA: EL SIMBOLO MAS OBSCURO

CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS

3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO
GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
LOS TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6 LOS TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9 LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1

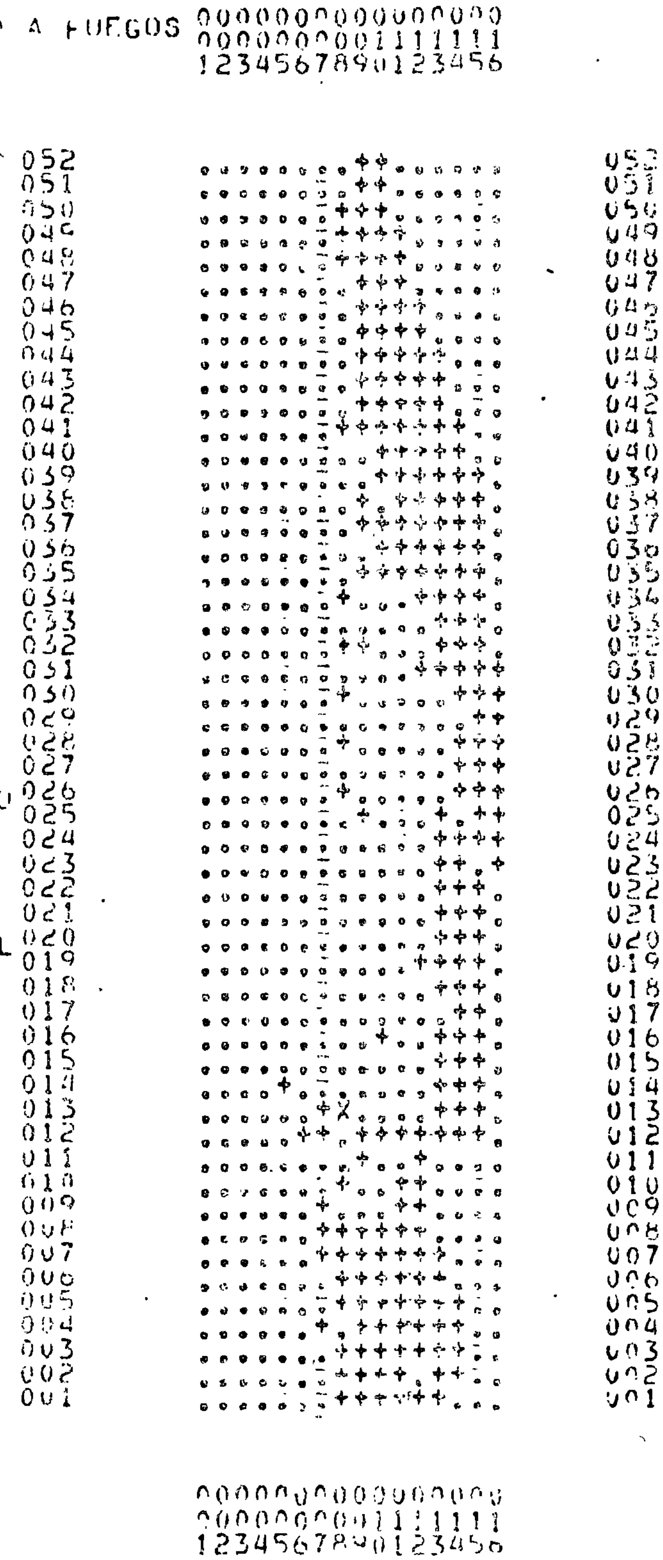
ESTACIONAMIENTOS
CABALLERIZAS
CAMINOS

USO DEL SUELO 11

ACTIVIDADES DE ESPARC.
ESTRUCTURAS
VIS

USO DEL SUELO 111

ESTAC./ TRAILERS
CAMINATA



NIVELES	0	3	4
SIMBOLOS	•	+	X
FRECUENCIA	598	233	1

TITULO DEL MAPA

MODELO DE IMPACTO Ñ 5: DESLIZAMIENTOS DE TIERRA.

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES ----------

VARIABLES DATO USADAS EN EL MODELO.

7	ACCIDENTES GEOLOGICOS
2	PORCIENTO DE PENDIENTE
4	VEGETACION POR ZONA

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

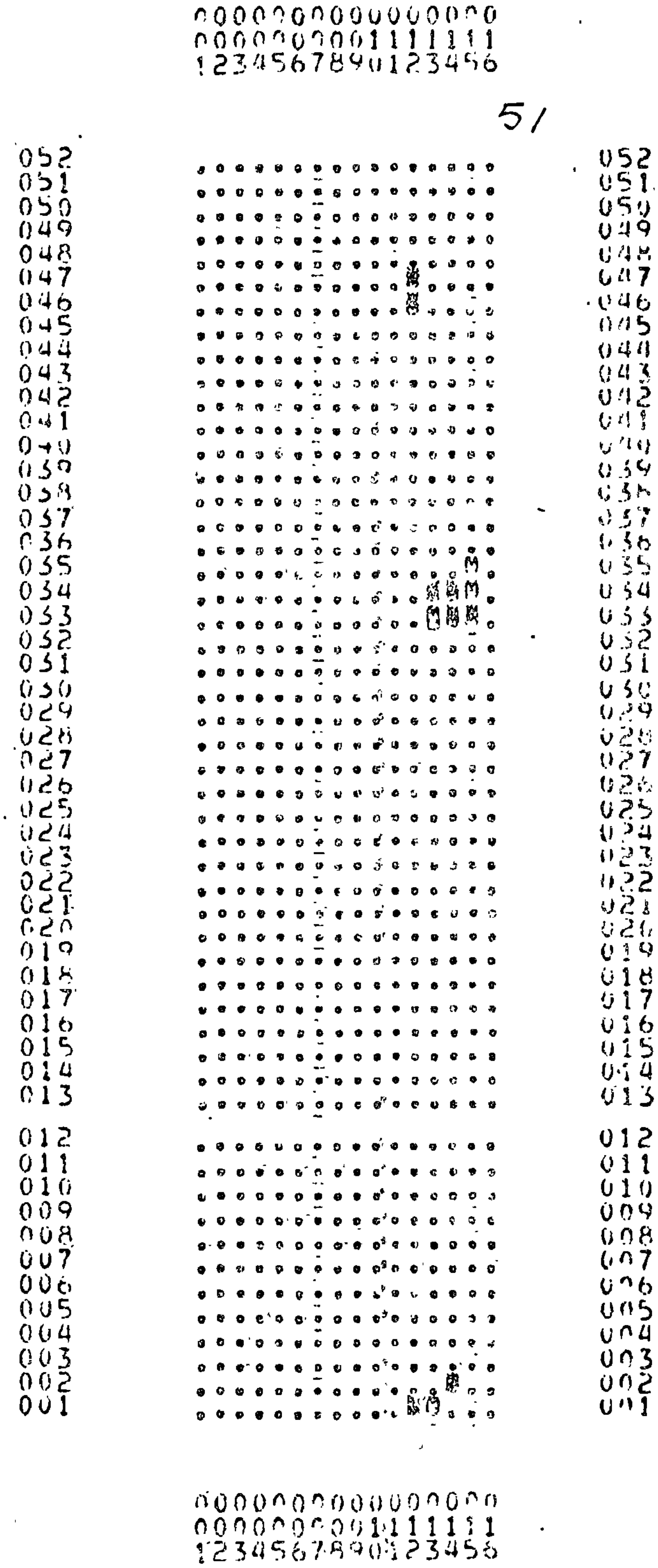
USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	CABALLERIZAS	ESTACIONAMIENTOS
CAMINATA		ESTRUCTURAS
		VIS
		CAMINOS
		ESTAC,/ TRAILERS

MODELO DE IMPACTO N 5:

DESLIZAMIENTOS DE TIERRA

AREA DE ESTUDIO   VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



VARIABLES DATO USADAS EN EL MODELO

7 ACCIDENTES GEOLOGICOS
2 PORCIENTO DE PENDIENTE
4 VEGETACION POR ZONA

LEGENDA: EL SIMBOLO MAS OBSCURO
CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS
3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO
2 GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3 LOS TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6 LOS TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9 LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1

ACTIVIDADES DE ESPARC.
- ... IATA

USO DEL SUELO 11

CABALLERIZAS

USO DEL SUELO 111

ESTACIONAMIENTOS
ESTRUCTURAS
VIS
CAMINOS
ESTAC./ TRAILERS

NIVELES	0	7	9
SIMBOLOS			
FRECUENCIA	820	4	8

TITULO DEL MAPA

MODELO DE IMPACTO Ñ 6: BASURA EN EL MAR

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES : ----------
VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
8 ZONAS LLANAS
9 PROXIMIDAD AL AGUA

LEGENDA: EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

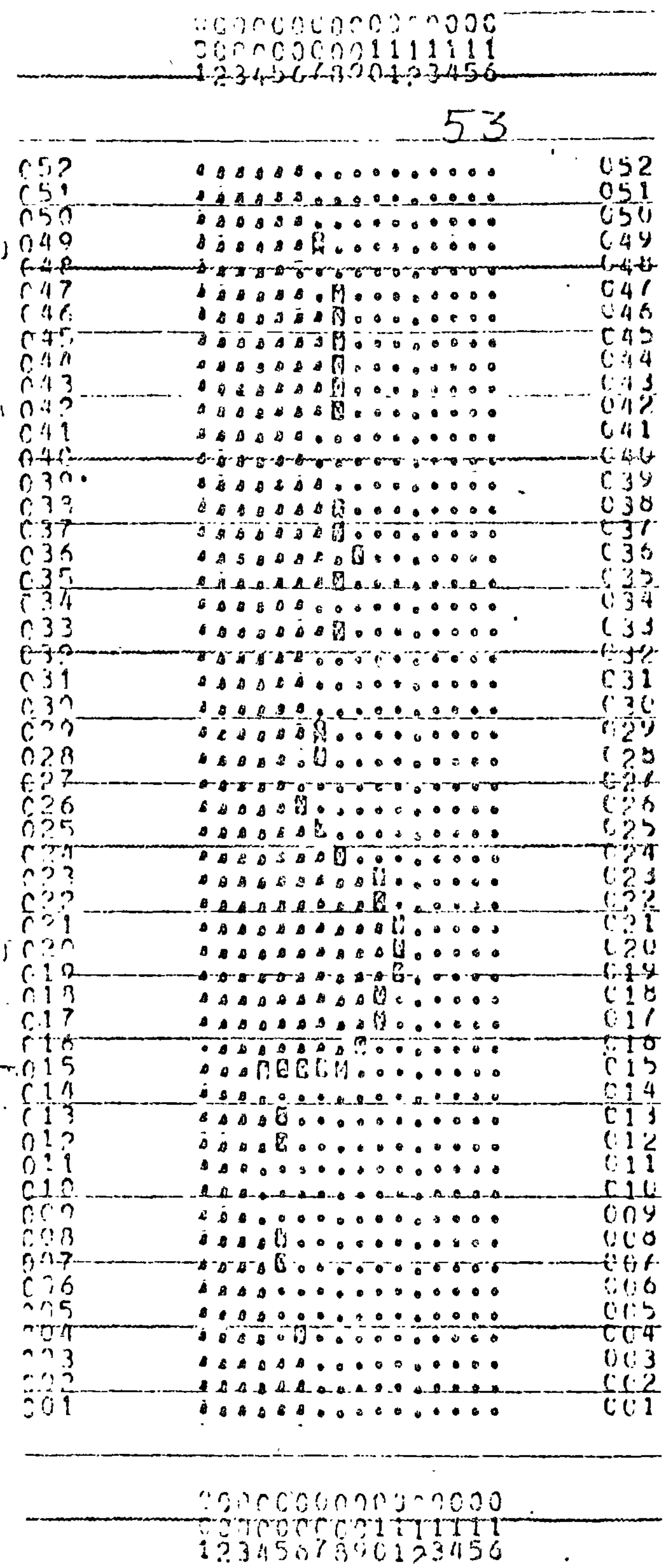
0	LOS TRES GRUPOS DE US COMPATIBLES
1	GRUPO DE US 111 MODERADO
2	GRUPO DE US 111 MODERADO, GRUPO 11 MODERADO
3	LOS TRES GRUPOS DE US MODERADO
4	GRUPO DE US 111 SEVERO
5	GRUPO DE US 111 SEVERO, GRUPO 11 SEVERO
6	LOS TRES GRUPOS DE US SEVERO
7	GRUPO DE US 111 TERMINAL
8	GRUPO DE US 111 TERMINAL, GRUPO 11 TERMINAL
9	LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	ESTRUCTURAS	ESTACIONAMIENTOS
CAMINATA		VIS
CABALLERIZAS		CAMINOS
		ESTAC./ TRAILERS

53

MODELO DE IMPACTO # 6: BASURA_EN EL MAR

AREA DE ESTUDIO VALLE DEL PACIFICO



LABORATORIO DE PLANEACION URBANA

VARIABLES DATO USADAS EN EL MODELO

4 VEGETACION POR ZONA
6 ZONAS LLANAS
9 PROXIMIDAD AL AGUA

LEYENDA: ~~EL SIMBOLO MAS OBSCURO CORRESPONDE AL IMPACTO MAYOR~~

LOS VALORES DEL IMPACTO PARA LOS 3 GRUPOS DEL USO DEL SUELO SON:

0 LOS TRES GRUPOS DE US COMPATIBLES
1 GRUPO DE US 111 MODERADO
2 GRUPO DE US 111 MODERADO, GRUPO II MODERADO
3 LOS TRES GRUPOS DE US MODERADO
4 GRUPO DE US 111 SEVERO
5 GRUPO DE US 111 SEVERO, GRUPO II SEVERO
6 LOS TRES GRUPOS DE US SEVERO
7 GRUPO DE US 111 TERMINAL
8 GRUPO DE US 111 TERMINAL, GRUPO II TERMINAL
9 LOS TRES GRUPOS DE US TERMINAL

USO DEL SUELO I

ACTIVIDADES DE ESPARC.
[illegible]
[illegible]

USO DEL SUELO II

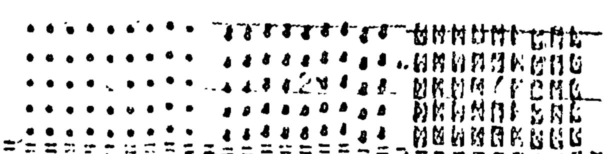
ESTRUCTURAS

USO DEL SUELO III

ESTACIONAMIENTOS
VIAS
[illegible]
ESTAC./ TRAILERS

NIVELES 1 2 7

SIMBOLOS



FRECUENCI [illegible]

V. EVALUACION DE PLANES DE USO DEL SUELO

El objetivo de los planes es localizar los usos del suelo propuestos en los sitios de mayor atractividad y al mismo tiempo minimizar el impacto ambiental en el área.

El resultado de la evaluación se expresa en dos tablas sumario. Una con la atractividad total del plan y la otra con los impactos producidos en los sistemas considerados. Además, se obtienen un mapa de atractividad del plan y uno para cada impacto.

Paso 1.- Localización de los usos del suelo

Mediante una inspección visual de los mapas de atractividad y de impacto se pueden diseñar planes, indicando el uso del suelo que ha sido puesto en cada una de las celdillas de la retícula. Sólo se permite un tipo de uso del suelo en cada celdilla.

Se debe verificar que los usos del suelo esten localizados en el número de celdillas necesario para alojar todo el área propuesta en el estudio.

Paso 2.- Codificación de los planes

El procedimiento es bastante similar al utilizado en la preparación del banco de datos. Se debe comenzar por la orilla superior izquierda del área de estudio y recorrer horizontalmente, celdilla por celdilla e hilera por hilera. El número del uso del suelo debe perforarse en la columna correspondiente, dejando en blanco aquellas en donde no se localice ningún uso. Las hileras de la retícula se registran mediante una o más tarjetas, dejando 2 columnas por cada celdilla ya que algunos números de usos del suelo necesitan dos cifras.

Para cada uno de los planes que se deseen evaluar debe prepararse un paquete

Paso 3.- Revisión de la codificación y perforación de los planes.

Mediante el programa REVISA/IMGRID se obtiene un listado de los planes de uso del suelo para su revisión.

Paso 4.- Almacenar los planes en la computadora.

Cada uno de los planes es almacenado en un archivo de disco magnético mediante el programa PLANES/IMGRID. Estos archivos serán utilizados posteriormente en la evaluación de los planes.

Paso 5.- Evaluación de los planes en la computadora.

Durante la fase de evaluación, la computadora suma todos los atractivos que corresponden a las celdillas designadas para localizar un uso del suelo particular La suma total se divide entre el número total de celdillas distribuidas para ese uso del suelo. El resultado es un indicador de atractividad promedio. Este es multiplicado por 10 para ponerlo en una escala de 1 a 100.

Un marcador final alto indica que el uso del suelo fue puesto en áreas que cumplen muchos de los criterios deseados para ese uso del suelo. Un marcador bajo indica una pobre selección en la localización de los usos del sulo con respecto a la tractividad de las celdillas.

Todos los usos del suelo que se van a incluir en el área de estudio son evaluados en forma similar y reciben un marcador final de atractividad promedio.

La segunda parte en la evaluación de un plan consiste en dar una estimación del grado de impacto negativo creado por el plan de uso del suelo en su totalidad sobre cada uno de los sistemas previamente definidos por los modelos de impacto.

Cuando la computadora encuentra una celdilla a la que se ha asignado un uso del suelo, determina el grado de impacto que tendrá ese uso del suelo sobre el sistema que está siendo analizado. El grado de impacto dependerá de la combinación de niveles y categorías encontrada en la celdilla y del grupo de usos del suelo al que pertenezca.

El marcador final de cada modelo indica el grado de impacto promedio creado por la totalidad del plan sobre el sistema considerado en toda el área de estudio. La media numérica es una media ponderada. El número de celdillas que reciben un indicador de impacto 1, se multiplica por 1. El número de celdillas que reciben un indicarod de impacto 2 (impacto moderado) se multiplica por 2, etc. El total final de los cuatro impactos se divide entre el número total de celdillas donde se localizan los nuevos usos del suelo. La media resultante estará entre un máximo de 4 y un mínimo de 1. Un marcador cercano a 4 indican que los usos del suelo han sido puestos donde crean una gran cantidad de impacto negativo. Un marcador cercado a 1 indica poco impacto en el sistema.

Para la evaluación de los planes deben perforarse las siguientes tarjetas:

1a. tarjeta:

Col. 1-5 Número de hileras de la retícula. *
Col. 6-10 Número de columnas de la retícula.*

2a. tarjeta:

Col. 4-5 Número de modelos de atractividad realizados en el estudio.*
Col. 9-10 Número de modelos de impacto realizados en el estudio.*
Col. 14-15 Número del plan que se desea evaluar.*

3a. tarjeta:

Col. 2, 4, 6, 8, 10, 12, 14, 16 Agrupación de los usos del suelo en el primer modelo de impacto (paso 5). Debe anotarse el número de grupo al que pertenece cada uno de los usos considerados en el estudio dados en orden.*

Tarjetas siguientes: Lo mismo que la 3a. tarjeta.
Una para cada modelo de impacto realizado en el estudio.
Deben darse en el mismo orden en que se corrieron los modelos.

La forma de alimentar el programa EVALUA/IMGRID se ilustra en la figura.

* Deben perforarse como números enteros justificados a la derecha.

Paso 6.- Mapear los resultados de la evaluación.

Se debe producir un mapa de la atractividad del plan y uno de cada impacto considerado.

El mapa de atractividad indica el grado de atractividad logrado al haber localizado los usos del suelo en los sitios que se pensaron convenientes.

La impresión del mapa estará basada en un rango numérico de 1 a 4. El número 1 indica que el uso del suelo es compatible; un número 4 indica que el uso del suelo tiene un impacto terminal sobre el área.

Se obtendrá un mapa impreso para cada sistema descrito por un modelo de impacto. Cada mapa representa un sistema diferente e indica cuantas celdillas del total distribuido tienen impacto compatible, moderado, severo ó terminal.

La forma de especificar los mapas con el programa --- MAPAS/IMGRID se explica en el capítulo VI.

Para producir los mapas de evaluación debe prepararse un paquete como el de la figura.

Paso 7.- Análisis de resultados y generación de nuevos planes

Con esta información a mano y las áreas problema localizadas, se puede preparar un segundo plan.

La preparación de este segundo plan implica la relocalización de los usos del suelo que crean impactos mayores o que reciben bajos indicadores de atractividad.

El objetivo final es elevar la atractividad y disminuir los impactos. Se debe lograr un equilibrio en la atractividad total no puede ser elevada sin elevar también los impactos.

Con este punto se supone que los usos del suelo han sido puestos en las mejores localizaciones dadas las condiciones naturales existentes, hasta que se presente nueva información disponible.

Una vez realizada la evaluación debe ser examinada, identificando las áreas problema cuyos indicadores de atractividad son mas bajos que los esperados ó sus indicadores de impacto son mayores que los deseables.

TITULO DEL MAPA

EVAL DEL IMPACTO DEL PLAN 1 :CONTAMINACION DE LA SUP. DE AGUA.

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES
LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	ESTACIONAMIENTOS	CABALLERIZAS
CAMINATA	ESTRUCTURAS	ESTAC./ TRAILERS
	CAMINOS	
	VIS	

TITULO DEL MAPA

EVALUACION DEL IMPACTO DEL PLAN # 1 EN: EROSION

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES
LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	CAMINOS	SERVICIO INF. TUR
CAMINATA	ESTACIONAMIENTOS	CABALLERIZAS
	ESTRUCTURAS	ESTAC./ TRAILERS

TITULO DEL MAPA

EVALUACION DEL IMPACTO DEL PLAN # 1 EN: CAMBIO EN LA VISUALIDAD

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES
LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 111
ACTIVIDADES DE ESPARC.	ESTRUCTURAS	ESTACIONAMIENTOS
CAMINATA	VIS	CAMINOS
CABALLERIZAS		ESTAC./ TRAILERS

TITULO DEL MAPA

EVALUACION DEL IMPACTO DEL PLAN # 1 EN: VULNERABILIDAD A FUEGOS 60

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 1
ESTACIONAMIENTOS CABALLERIZAS CAMINOS	ACTIVIDADES DE ESPARC. ESTRUCTURAS VIS	ESTAC./ TRAILER CAMINATA

TITULO DEL MAPA

EVALUACION DEL IMPACTO DEL PLAN # 1 EN: DESLIZAMIENTOS DE TIERRA

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 1
ACTIVIDADES DE ESPARC. CAMINATA	CABALLERIZAS	ESTACIONAMIENTO ESTRUCTURAS VIS CAMINOS ESTAC./ TRAILERS

TITULO DEL MAPA

EVALUACION DEL IMPACTO DEL PLAN #1 EN: BASURA EN EL MAR

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

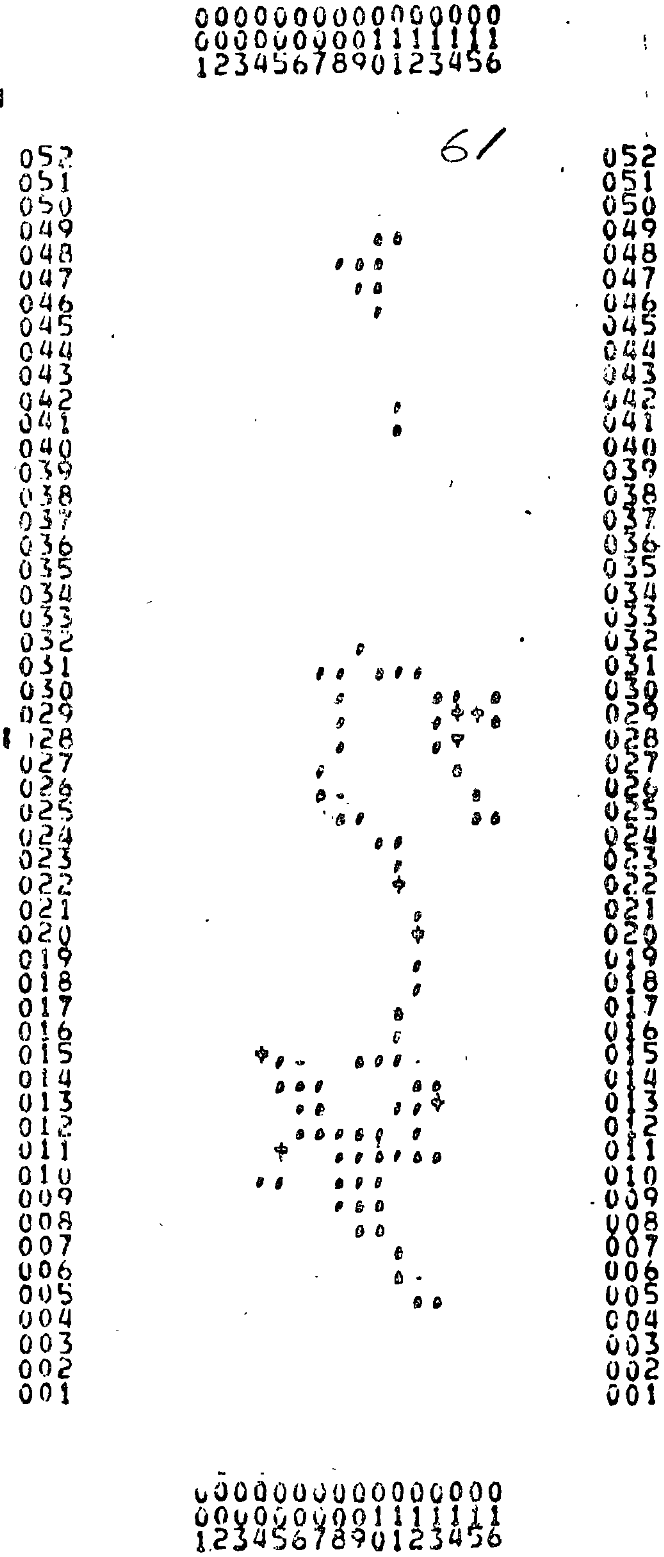
OPCIONES USADAS PARA ESTE MAPA

10 EL TEXTO DEL MAPA ES

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1	USO DEL SUELO 11	USO DEL SUELO 1
ACTIVIDADES DE ESPARC. CAMINATA CABALLERIZAS	ESTRUCTURAS	ESTACIONAM TO VIS CAMINOS ESTAC./ TRAILER

EVALUACION DEL IMPACTO DEL PLAN Ñ 1 EN: EROSION



AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO I

- ACTIVIDADES DE ESPRC.
- CAMINATA

USO DEL SUELO II

- CAMINOS
- ESTACIONAMIENTOS
- ESTRUCTURAS

USO DEL SUELO III

- SERVICIO INF. TURISTICA
- CABALLERIZAS
- ESTAC./ TRAILERS

NIVELES	0	1	2	3	4	5
SIMBOLOS	0	1	2	3	4	5
FRECUENCIA	0	42	38	8	0	0

EVALUACION DEL IMPACTO DEL PLAN Ñ 1 EN:

CAMBIO EN LA VISUALIDAD

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1

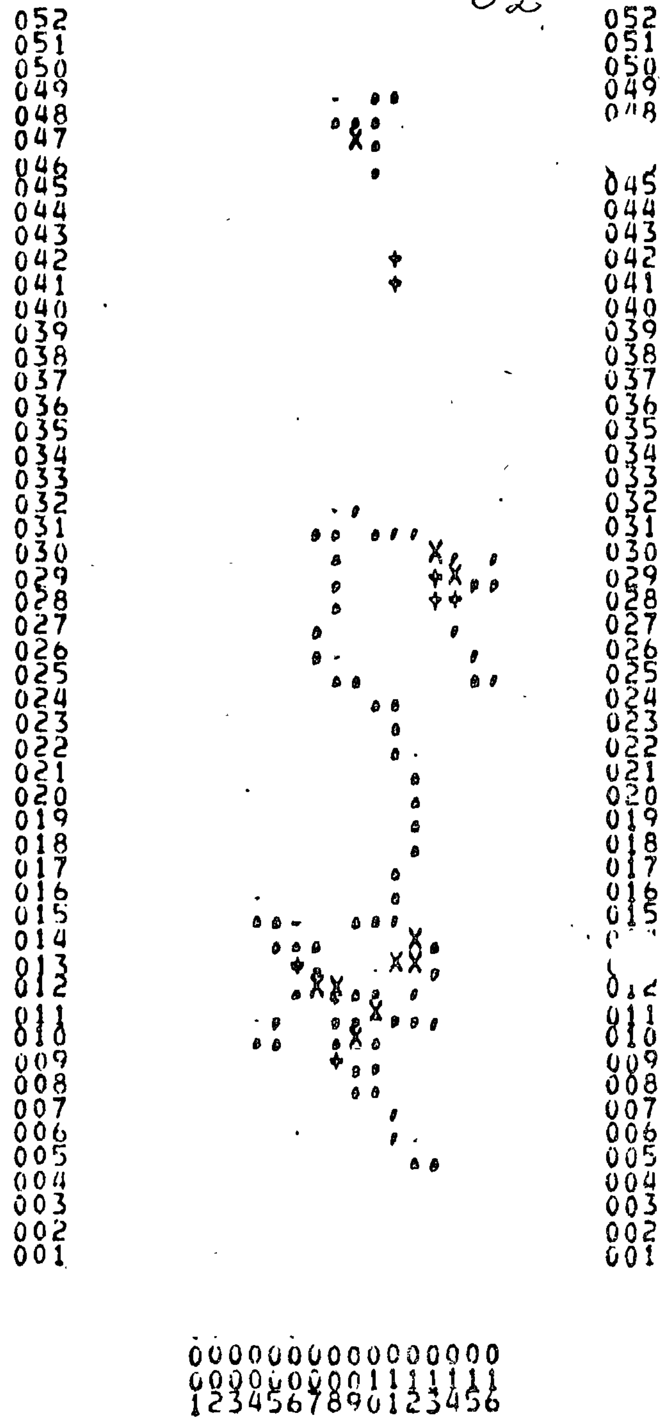
ACTIVIDADES DE ESPARC.
CAMINATA
CABALLERIZAS

USO DEL SUELO 11

ESTRUCTURAS
VIS

USO DEL SUELO 111

ESTACIONAMIENTOS
CAMINOS
ESTAC./ TRAILERS



NIVELES	0	1	2	3	4	5
SIMBOLOS	....0....	''''1''''	''''2''''	++++3++++	XXXX4XXXX	00005000
FRECUENCIA	0	13	58	7	10	0

EVAL DEL IMPACTO DEL PLAN 1 :

CONTAMINACION DE LA SUP. DE AGUA.

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1

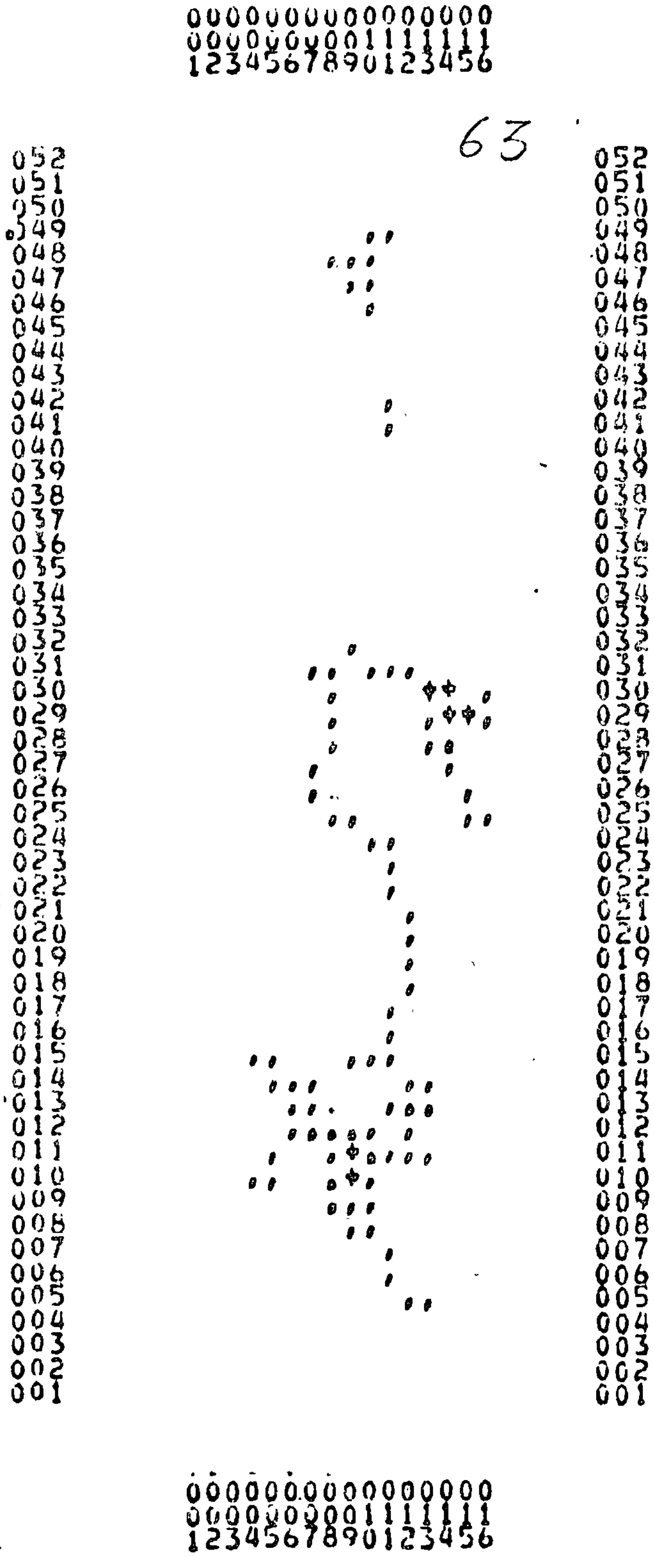
ACTIVIDADES DE ESPARC.
CAMINATA

USO DEL SUELO 11

ESTACIONAMIENTOS
ESTRUCTURAS
CAMINOS
VIS

USO DEL SUELO 111

CABALLERIZAS
ESTAC./ TRAILERS



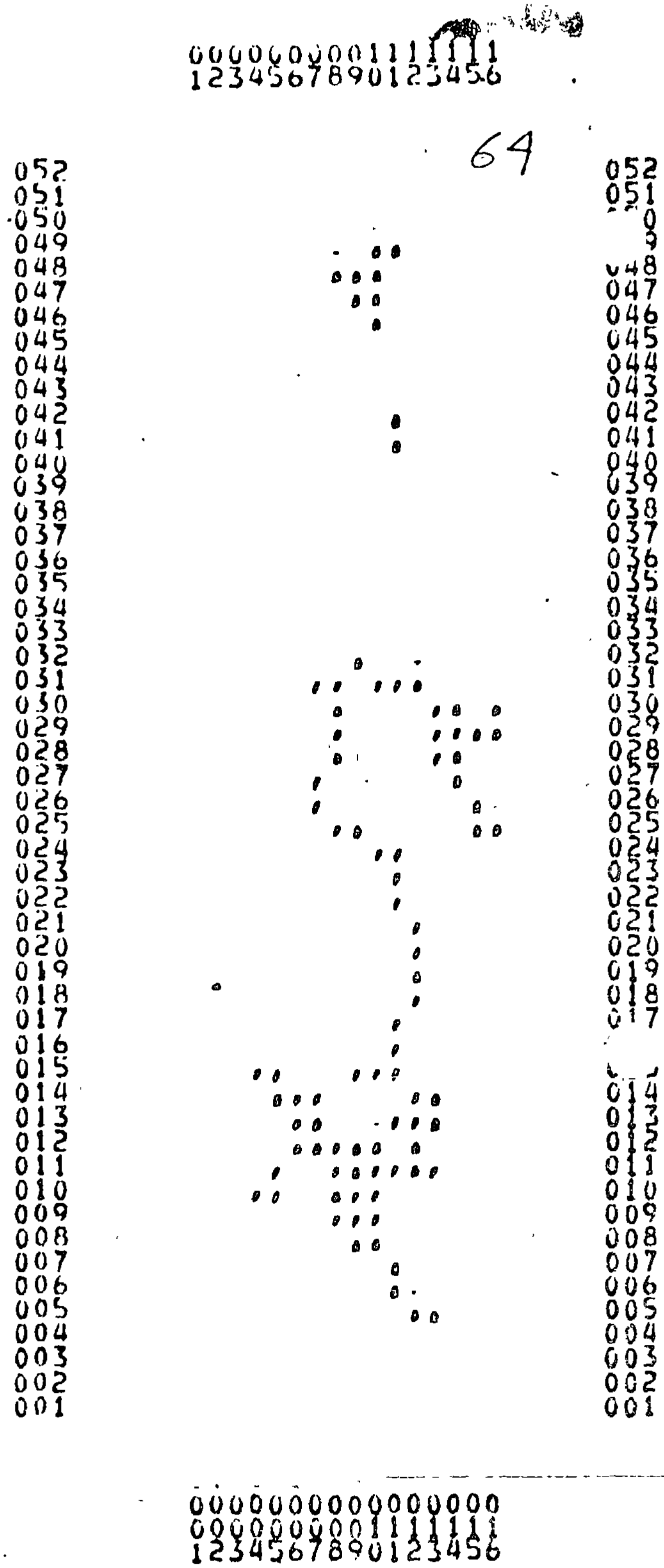
NIVELES	0	1	2	3	4	5
SIMBOLOS	0	1	2	3	4	5
FRECUENCIA	0	73	9	6	0	0

EVALUACION DEL IMPACTO DEL PLAN Ñ 1 EN:

VULNERABILIDAD A FUEGOS

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA



LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1

ESTACIONAMIENTOS
CABALLERIZAS
CAMINOS

USO DEL SUELO 11

ACTIVIDADES DE ESPARC.
ESTRUCTURAS
VIS

USO DEL SUELO 111

ESTAC. / TRAILERS
CAMINATA

NIVELES	0	1	2	3	4	5
SIMBOLOS	.	,	•	+	X	0
FRECUENCIA	0	45	43	0	0	0

EVALUACION DEL IMPACTO DEL PLAN Ñ1 EN:

BASURA EN EL MAR

AREA DE ESTUDIO VALLE DEL PACIFICO

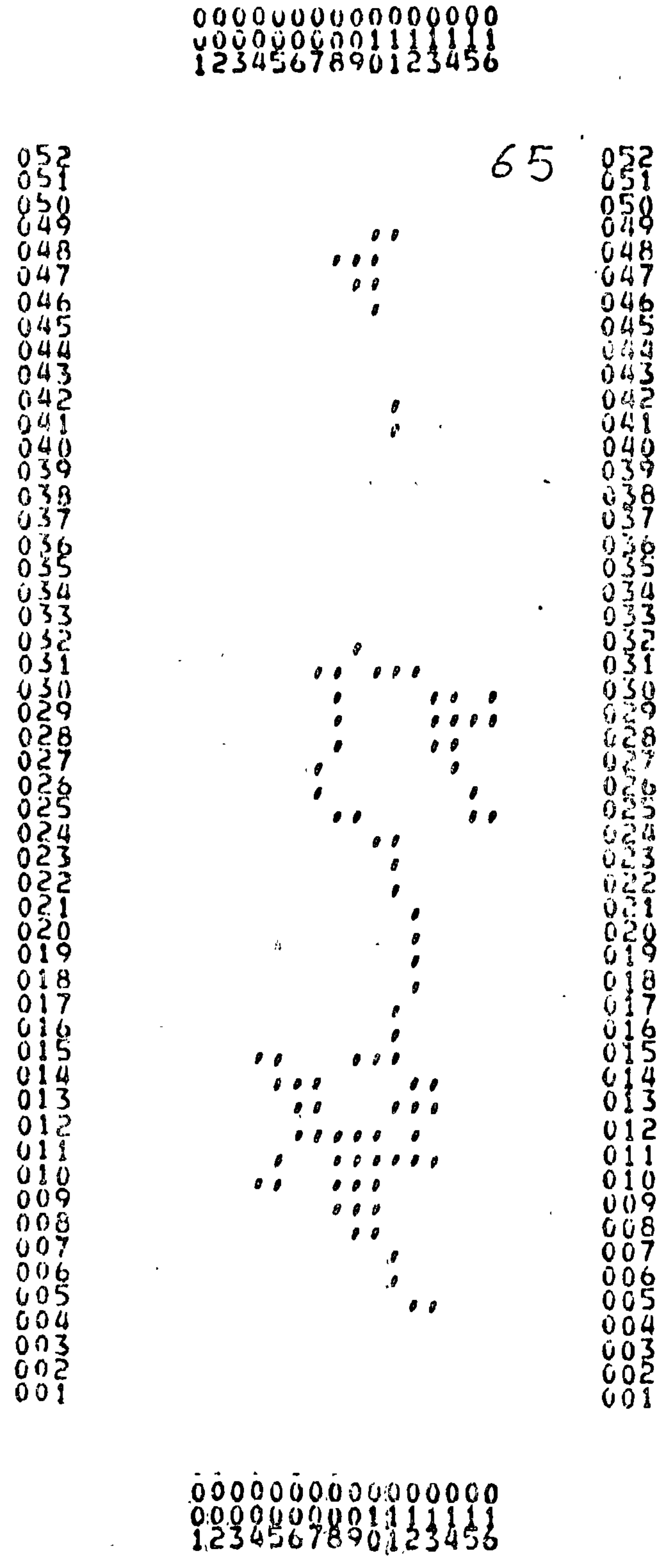
LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1
ACTIVIDADES DE ESPARC.
CAMINATA
CABALLERIZAS

USO DEL SUELO 11
ESTRUCTURAS

USO DEL SUELO 111
ESTACIONAMIENTOS
VIS
CAMINOS
ESTAC./ TRAILERS



NIVELES	0	1	2	3	4	5
SIMBOLOS	0	1	2	3	4	5
FRECUENCIA	0	88	0	0	0	0

EVALUACION DEL IMPACTO DEL PLAN Ñ 1 EN:

DESLIZAMIENTOS DE TIERRA

AREA DE ESTUDIO   VALLE DEL PACIFICO

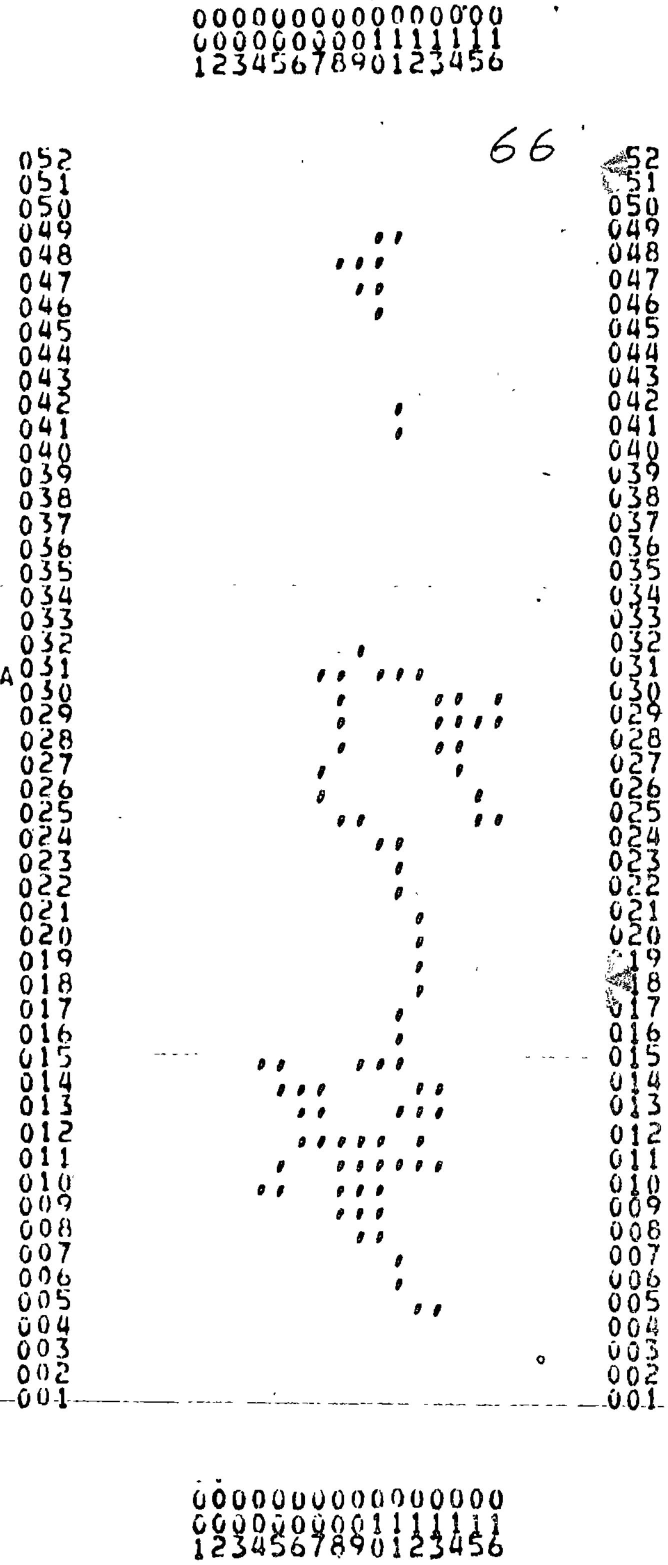
LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1
ACTIVIDADES DE ESPARC.
CAMINATA

USO DEL SUELO 11
CABALLERIZAS

USO DEL SUELO 111
ESTACIONAMIENTOS
ESTRUCTURAS
VIS
CAMINOS
ESTAC./ TRAILERS



NIVELES	0	1	2	3	4	5
SIMBOLOS	0	1	2	3	4	5
FRECUENCIA	0	88	0	0	0	0

EVALUACION DE LA ATRACTIVIDAD DEL PLAN N 2

AREA DE ESTUDIO VALLE DEL PACIFICO

LABORATORIO DE PLANEACION URBANA

LOS USOS DEL SUELO SE AGRUPARON EN LOS GRUPOS:

USO DEL SUELO 1

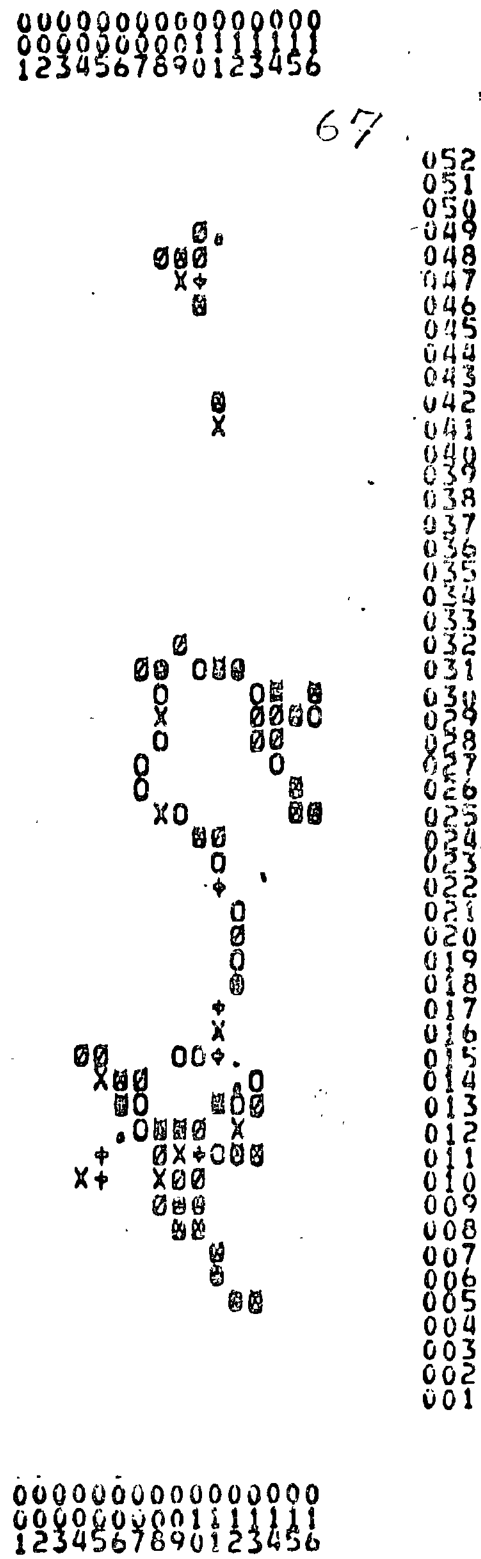
ACTIVIDADES DE ESPARC.
CAMINATA
CABALLERIZAS

USO DEL SUELO 11

ESTRUCTURAS

USO DEL SUELO 111

ESTACIONAMIENTOS
VIS
CAMINOS
ESTAC./ TRAILERS



NIVEL	2	3	4	5	6	7	8	9
SIMBOLO	.....	++++	XXXX	OOOO	ØØØØ	ØØØØ	ØØØØ	ØØØØ
FRECUE	3	7	10	19	20	16	7	6

VI. PRODUCCION DE MAPAS

El mapeo de la información generada en cada una de las etapas del proceso (banco de datos, modelos y evaluación) se realiza mediante el programa MAPAS/IMGRID, el cual es alimentado con uno ó varios sub-paquetes MAP, siendo antecedidos estos por un sub-paquete de contorno irregular en caso de ser necesario (paso 6 del banco de datos).

El sub-paquete MAP maneja una variedad de opciones que controlan el tamaño, el simbolismo y los textos del mapa.

Toda opción una vez, especificada seguirá siendo válida en los mapas subsecuentes a menos que sea cambiada.

Las opciones 1-7 deben ser siempre especificadas ya que el programa no tiene condiciones estándar en caso de no activarlas.

Un sub-paquete MAP consta de las siguientes tarjetas:

Tarjeta 1:

Col. 1-3 — 'MAP' para identificar el sub-paquete.

Tarjetas 2-4:

Col. 1-72 — Información que se desea al pie del mapa. Pueden dejarse en blanco, pero deben incluirse las tres.

Tarjetas siguientes: — Opciones deseadas.

Penúltima tarjeta: — '99999' para indicar que termina el sub-paquete

Ultima tarjeta:

Col. 12 — Número de la variable ó archivo que se va a mapear. Perforado como entero justificado a la derecha.

Tarjeta de terminación:

Col. 1-3 — 'END' Debe colocarse después del último sub-paquete MAP de la corrida.

Opción 1.- Tamaño de la retícula.
(1 tarjeta)

Col. 5 '1' para identificar la opción.
Col. 11-20 Número de celdillas que contiene la retícula en sentido vertical.
Col. 21-30 Número de celdillas que contiene la retícula en sentido horizontal.
Col. 31-40 Número de caracteres con que se desea imprimir cada celdilla en el sentido verticla.
Col. 41-50 Número de caracteres con que se desea imprimir cada celdilla en el sentido horizontal.

Los números en los campos 2, 3, 4, 5, se perforan como números decimales ó como enteros justificados a la derecha.

Opción 2.- Número de niveles ó intervalos de clase.
(1 tarjeta)

Col. 5 '3' para identificar la opción.
Col. 11-20 Número de niveles ó intervalos de clase en que se subdivide la variable. Perforado como número decimal ó entero justificado a la derecha.

Opción 4.- (no se utiliza en esta versión del programa).

Opción 5.- (no se utiliza en esta versión del programa).

Opción 6.- (no se utiliza en esta versión del programa).

Opción 7.- Simbolismo.
(5 tarjetas)

1a. tarjeta:

Col. 5 '7' para identificar la opción.

2a. tarjeta:

Col. 1-10 Caracteres que se desea imprimir ó sobre-imprimir para representar a cada nivel o categoría de la variable, dados en orden y perforados en la columna correspondiente. Sólo debe usarse el número de columnas necesario.
Col. 11-20 Caracteres para especificar el simbolismo de los puntos señal que aparecen al centro de cada celdilla y en la clave del histograma correspondiente a cada uno de los niveles o categorías.
Col. 25 Caracter para especificar el simbolismo que aparece fuera del área de estudio.

Opción 8.- Punto señal.
(1 tarjeta)

Col. 5 '8' para identificar la opción.

Col. 20 — '1' si se desea suprimir la impresión del punto señal.
'0' si se desea restablecer la impresión.

Cuando se hace un mapa en el que un caracter representa una celdilla, se suprime automáticamente el punto señal y debe restablecerse para mapas subsecuentes. Para otros casos, si no se especifica nada, el punto señal aparecerá impreso en el mapa.

Opción 9.- Histograma
(1 tarjeta)

Col. 5 — '9' para identificar la opción.
Col. 20 — '1' para generar un histograma al pie del mapa, el cual muestra las frecuencias de celdillas de la retícula en cada nivel ó categoria.
Col. 30 — '1' para suprimir la información numérica que se imprime junto al simbolismo de los niveles o categorías.

Si no se especifica esta opción, no se imprime el histograma y se incluye la información numérica correspondiente. Para regresar al estándar debe perforarse '0' en los campos 2 y 3.

Opción 10.- Texto explicativo.
(3 a 32 tarjetas)

1a. tarjeta:

Col. 4-5 — '10' para identificar la opción.

Tarjetas restantes:
(no más de treinta)

Col. 1-12 — Cualquier información adicional que se desea aparezca al pie del mapa.

Ultima tarjeta:

Col. 1-7 — 'ENDTEXT' para indicar que termina el texto.

Si no se especifica esta opción no aparece ningún texto. Si se desea eliminar el texto en mapas subsecuentes, deben alimentarse solo la primera y la última de las tarjetas.

Opción 11.- (No se usa en esta versión de GRID)

Opción 12.- Mapa puntual.
(1 tarjeta)

Col. 4-5 '12' para identificar la opción.
Col. 20 '1' para especificar simbolismo puntual.
'10' para restablecer simbolismo gris.

Como una alternativa al simbolismo normal, de la opción se puede producir un mapa puntual representando cada celdilla mediante 4 caracteres en sentido vertical, 5 en el horizontal y utilizando el simbolo Ø.

Se pueden tener hasta 20 niveles ó categorias. El número de caracteres impresos en la celdilla es igual al número del nivel-ó categoría.

Opción 13.- Numeración de la retícula.
(1 tarjeta)

Col. 4-5 '13' para identificar la opción.
Col. 20 '1' para numerar la malla.
'10' para regresar al estándar.
Col. 21-30 Número de columna de la celdilla de referencia. Perforado como número decimal ó entero justificado a la derecha.
Col. 31-40 Número de hilera de la celdilla de referencia.- Perforado como número decimal ó entero justificado a la derecha.

Esta opción imprime número para las hileras y las columnas en-los cuatro lados de la retícula para ayudar al usuario a localizar celdillas individuales sobre el mapa.

La celdilla superior izquierda es denominada celdilla de referencia de la retícula y sus coordenadas sirven de orígen para-numerar hileras y columnas. Si estas coordenadas no son especificadas el programa supone que son: columna = 1, hilera = n, donde n es el número de hileras especificado en la opción 1.

Si no se especifica no se numera la retícula.

Opción 14.- Datos escalados
(1 tarjeta)

Col. 4-5 '14' para identificar la opción.
Col. 20 '1' para indicar que los datos estan escalados.

Esta opción siempre debe activarse en esta versión del programa.

BIBLIOGRAFIA

1. PUBLIC LAND POLICY AND THE ENVIRONMENT.

LANDS CAPES, INC.
Public Land Review Commission.
Washington, DC., 1970

2. DRAPT ENVIRONMENTAL IMPACT STATEMENT INTERSTATE 84, RHODE ISLAND AND PROVIDENCE PLANTATIONS.

U.S. Departament of Commerce
ELS-RI-72-5834-D.

3. OMNIBUS STUDIO 1975-1976

- Atractiveness models.
- Volnerability models.

Departament of Landscape Architecture.
Harvard Graduate School of Design.

4. A GENERAL SYSTEM FOR ENVIRONMENTAL RESOURCE ANALYSIS.

A report to the Public Land Review Commission.

Steinit3, Rogers Associates, Inc.
Cambridge, Mass., 1970.

5. A COMPARATIVE STUDY OF RESOURCE ANALYSIS METHODS.

Steinit3, Murray, Sinton, Way.

Departament of Landscape Architecture.
Harvard Graduate School of Design., 1969.

6. THE ANTILOPE ISLAND STUDY.

Robert Scott.
Departament of Landscape Architecture and Environmental Planning.
Utha State University, 1974.

7. HUDROLOGY.

LAEP 610 - Fall 1.974.
Matt Conover, Dan Green, Rick
Johnson, Marsha Lee.

- Introduction
- Availabillty
- Water Quality
- Interactions
- Implications for Residential growth
- Appendix

8. VEGETATION AND WILDLIFE OF SOUTH CACHE VALLEY.

Steve Smith, Steve Henning,
Chuck Borup, Mark Mc Kenzie

- Vegetation
- Wildlife

9. GEOLOGY OF SOUTHERN CACHE VALLEY

J. T. Fey, Allan Snell, Ann Mollens, Wade Ward.

- Design Considerations.

10. SOUTH CACHE VALLEY: SOILS.

J. T. Fey, Allan Snell, Ann Mullens, Wade Ward.

11. SOUTHERN CACHE VALLEY : A VISUAL ANALYSIS.

B. Hendrickson, B. Hinckley,
S. Holtschneider, C. Mayer.

- Concept Development.
- Visual Inventary
- Data Evaluation and Technical Analysis
- Presentation
- Recommendations for Sity Types
- General Recommendations.

12. CLIMATOLOGICAL SUMMARY

Physiology & Microclimate
LAEP 610, Fall 1.974.
Complied from Prof. R. E. Toth.

- Frost Free Days
- Precipitation

13. POLITICS OF PLANNING AND THE PROBLEM OF PUBLIC CONFIDENCE: A SOCIOLOGY OF CONFLICT APPROACH

    David M. Freeman, Department of Sociology
    Colorado State University

    - The Social Environment of the Public Planner
    - The Criteria of Planning
    - Toward a Sociological Approach to Planning.
    - Summary and Conclusión

14. SUMMARY REPORT ON: HILLS' DEVELOPING A BETTER ENVIRONMENT

    Blaise Grden, Stan Holtschneider

    - Case Study
    - Concepts and Goals
    - Key Terms.
    - Identification of Method

15. CASE STUDY: HANDBOOK: APPLICATIONS OF REMOTE SENSING AND COMPUTER TECHNIQUES FOR RECREATION PLANNING.

    Allan Snell, Dan Green
    LAEP 610, Fall 1.974.

    - Introduction
    - Study Area
    - Proyect Output
    - Methodology
    - Data Classification System.

16. THE PLAN & PROGRAM FOR THE BRANDY WINE - A CASE STUDY

    Bruce D. Hinckley
    LAEP 610, Utah State University
    Fall 1.974

    - The Case Study
    - Goals and Concepts
    - Methods
    - Graphic Outline of approach methodology
    - Conclusions

17. METLAND CASE STUDY REPORT

Charles Barup

- The Case Study
- The Goals and Objectives
- Definition of Key Terms
- Method of Approach

18. THE NEW COMMUNITIES FAMILY MOBILITY SYSTEM

Urban Studes Center
University of Louisville.

- Flow Diagram
- Social Goals
- Physical Goals
- Psychosocial Goals
- Money
- Mobility
- Monagement
- Method
- Locational Analysis
- Regional Delineation
- Program Assumptions
- Overview of the Process
- Regional Analysis
- Subregional Analysis
- Area Analysis

19. HONEYHILL: A SYSTEMS ANALYSIS FOR PLANNING THE MULTIPLE USE OF CONTROLLED WATER AREAS.

Douglas Way, Richard Toth,
Carl Steintz, David Sinton,
US. Dept. of Commerce
AD - 736 343.

Peter Rogers, Timothy Murray
Department of Landscape
Architecture:
Harvard University

- Computer Method
- Creation of a Data Bank
- Basis for Land Use Selection
- Selection of Land Uses for the Proyect
- Attractiveness
- Development of site resource systems

- Model development for each system
- Model framewark
- Activity demand estimate
- Development of a Simulation Model
- Output
- Linear Programming Model
- Seasonal Operation and Maintenance exped; tures.
- Comparative Evaluation
- Innovations
- Assessments

20. MEDFORD - PERFORMANCE REQUIREMENTS FOR THE MAINTENANCE OF SOCIAL VALUES REPRESENTED BY THE NATURAL ENVIRONMENT OF MEDFORD TOWNSHIP, N.J., JUNE JA., 1.974.

Mark Mc. Kenxie, Marsha Lee
LAEP 610., Fall 1.974
Center of Ecological Research in Planning and Design, Department of Landscape Architecture and Regional Planning, University of Pennsylvania

- Background
- Basic Goals and Concepts
- Key Terms
- Identification of Method

21. LAND USE IMPACT AND ACTIVITY ALLOCATION ANALYSIS. PACIFIC VALLEY, California

Mike Johnson, Paul Parker.
Jim Lance
Department of Landscape Architecture and Environmental Planning.

Utah State University, Logam Utah.

- Introduction
- Data Inventary
- Attactiveness Models
- Impact Evaluations
- Centroid
- Plan Stategy and Conclusions
- Strategy Evaluation Process
- Literature Cited and Mayor Sources

22. WELLSVILLE CANYON - DRY LAKE AREA

Toth, Wegkamp

Department of Landscape
Architecture and Environmental Planning
Utah State University

Landscape Architecture - 401
Fall Quarter - 1.974.

- Problem
- Goals
- Procedures
- Phasing
- Papers and Reports

23. A PLANNING AND DESIGN PROCESS

Richard E. Toth
Dep. of Landscape Architecture and Environmental Planning.
Utah State University
September 4, 1.974.

- Introduction
- A Situation
- Process Description

24. AN APPROACH TO PRINCIPLES OF LAND PLANNING AND DESIGN

Richard E. Toth.
Department of Landscape Architecture
Harvard University
June 1.972.

PAQUETE IMGRID

Programa IMPACTO/IMGRID

HOJA DE CODIFICACION PARA LOS MODELOS DE IMPACTOS

Nombre del modelo: ______________________________

Tarjeta 1.- Número del modelo formato (I2): __ __

Tarjeta 2.- Nombre del modelo: LAEP/IMPT01 ,....., LAEP/IMPT30.

Tarjeta 3.- Número de las variables dato (3I3): __ __ __
cols. 3 6 9

Encontrar los valores de sensitividad, para cada una de las subvariables de las 3 variables que intervienen en el modelo, de acuerdo a la siguiente escala:

Subvariable más sensitiva = 1 = A

Subvariable moderada = 2 = B

Subvariable menos sensitiva = 3 = C

	no. de variable	Variable	Subvariables: más sensitiva → menos sensitiva		
más importante					
↓					
menos importante					
			A	B	C

cols.- 2 4 6 8 10 12 14 16 18 20

Tarjeta 4.- __ __ __ __ __ __ __ __ __ __ *Variable 1*

Tarjeta 5.- __ __ __ __ __ __ __ __ __ __ *Variable 2*

Tarjeta 6.- __ __ __ __ __ __ __ __ __ __ *Variable 3*

Reducir las variables 2 y 3 comparando entre si sus grados de sensitividad para obtener la matriz (3,3) de sensitividades combinadas.

Variable 2

Variable 3	A	B	C
A			
B			
C			

Valores de:

X= Más sensitivo =1

Y= Moderado =2

Z= Menos sensitivo=3

Tarjeta 7.- Valores de la matriz de sensitividades combinadas (9 I 2).

Encontrar la matriz de impactos para cada grupo de usos del suelo, estos valores de impactos se obtienen a partir de la combinación - de sensitividades de la matriz de sensitividades combinadas de las variables 2 y 3 con las sensitividades de la variable 1.

Variable 1

	A	B	C
X			
Y			
Z			

Uso del suelo
Grupo I

Variable 1

	A	B	C
X			
Y			
Z			

Uso del suelo
Grupo II

Variable 1

	A	B	C
X			
Y			
Z			

Uso del suelo
Grupo III

Asignación de grupos de usos del suelo

Uso del suelo 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 2

Grupo _

La matriz de impactos tiene tres niveles de impacto por grupo de usos del suelo.

Asignando valores de <u>C</u>- *Impacto Compatible*
<u>M</u>- *Impacto Moderado*
<u>S</u>- *Impacto Severo*
<u>T</u>- *Impacto Terminal*

Tarjeta 8.- Valores de la matriz de impactos U.S. grupo I (9 I 2)
Tarjeta 9.- Valores de la matriz de impactos U.S. grupoII (9 I 2)
Tarjeta10.- Valores de la matriz de impactos U.S. prupoIII(9 I 2)

Figura 7.- Hojas de codificación para el programa Impacto/IMGRID.

PAQUETE IMGRID

Programa ATRACTIVO/IMGRID

HOJA DE CODIFICACION PARA LOS MODELOS DE ATRACTIVIDAD

Nombre del(os) usuario(s):

Nombre de la actividad (usos del suelo) para la que se construyó el modelo:

Número del modelo (formato I2); --

Número de variables que intervienen en el modelo (formato I2); __

Columnas 1 y 2 Cuales son las variables relevantes n0.	Cual es la preferencia de las subvariables (del 0 al 9) Col-10	Col-12	Col-14	Col-16	Col-18	Col-20	Col-22	Col-24	Col-26	Col-28	Cual es la importancia de la variable (1al3) Col-34

Figura 4.- Hoja de codificación para los *MODELOS DE ATRACTIVIDAD*.