

Durham E-Theses

A comparison of the efficiency of techniques for digital encoding of contour maps

Owczarczyk, Jacek K.

How to cite:

Owczarczyk, Jacek K. (1978) *A comparison of the efficiency of techniques for digital encoding of contour maps*, Durham theses, Durham University. Available at Durham E-Theses Online:
<http://etheses.dur.ac.uk/9121/>

Use policy

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a [link](#) is made to the metadata record in Durham E-Theses
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Please consult the [full Durham E-Theses policy](#) for further details.

A COMPARISON OF THE EFFICIENCY
OF TECHNIQUES FOR DIGITAL ENCODING
OF CONTOUR MAPS

by

JACEK K. OWCZARCZYK

Submitted to the Faculty of Science of the University of Durham
in partial fulfillment of the requirements for the degree of Master
of Science.



September 1978

The copyright of this thesis rests with the author.
No quotation from it should be published without
his prior written consent and information derived
from it should be acknowledged.

ABSTRACT

A COMPARISON OF THE EFFICIENCY OF TECHNIQUES FOR DIGITAL ENCODING OF CONTOUR MAPS

There are two digitizing techniques which are operable without human interpretation of contour-mapped data and which thus may be performed semi-automatically; line following and raster scanning. After a comparison of these techniques the author concentrates on raster scanning, as this technique facilitates creation of matrix data, the data structure most widely used in spatial analysis performed by digital computers.

The author proposes the quantization method of computing matrix data and shows that quantized data may be used not only for overall computations but also to determine geometrical properties of surfaces - giving the possibility of error estimation-shown by the contour map. In particular such an approach enables pre-determination of a scanning parameter - track spacing - before digitizing to satisfy the required accuracy of analysis.

The different aspects of utilising data obtained by raster scanning are discussed; storage, display in contour map form - the algorithm to produce the type of map where the contour lines pass exactly through the sample points taken during scanning is included.

TO MY WIFE AND PARENTS

ACKNOWLEDGEMENTS

I wish to express my appreciation for the valuable comments and criticism Dr.I.S. Evans made while supervising my research. I am grateful to N.J.Cox, R.Hindmarsh and Dr.D.W.Rhind for helpful discussions and to J.S.Gill for allowing me to use his hand-digitized altitude matrix. My stay in Durham would not have been nearly so productive and enjoyable but for the congenial atmosphere in the 'Huts'.

I would like to thank The British Council for making my study possible with a generous grant.

LIST OF CONTENTS

	Page
Glossary of Terms	
INTRODUCTION	1
CHAPTER I	3
DIGITIZATION OF CONTOUR MAPS	
1.1 A Contour map as a source of information	3
1.2 Digitizing	4
1.3 Manual methods of digitizing	5
1.4 Semi-automatic techniques of digitizing	6
1.5 A comparison between the line following technique and the raster scanning technique	10
CHAPTER II	14
PROCESSING OF DATA OBTAINED BY RASTER SCANNING	
2.1 Creation of the data matrix	14
2.2 Spatial analysis based on ^{the} quantized matrix data	17
2.3 Display of data when digitizing by raster scanning	30
2.4 Storage of data when digitizing by raster scanning	32
CHAPTER III	34
TECHNICAL IMPLEMENTATION OF DIGITIZING METHODS	
3.1 Digitizing errors	34
3.2 Hardware for digitizing	38
CONCLUSIONS	41
REFERENCES	43
DESCRIPTION OF MAPS	45
APPENDIX A	63
SPLINE INTERPOLATION	
APPENDIX B	65
A MACHINE ALGORITHM TO CREATE A QUANTIZED DATA MATRIX FOR DATA OBTAINED BY RASTER SCANNING (WITH FORTRAN SUBROUTINES)	
APPENDIX C	70
A MACHINE ALGORITHM TO DISPLAY DATA OBTAINED BY RASTER SCANNING IN CONTOUR MAP FORM (WITH FORTRAN SUBROUTINE)	

GLOSSARY OF TERMS

- flying spot scanner** - a system that encodes a picture by scanning in strips across the image and recording the brightness at each incremental point along the scan line.
- gantry** - structure (a part of a digitizer) supporting a travelling arm or bridge to which a recording device is attached.
- line following** - a technique of digitizing of lines by measurement of absolute or incremental rectangular co-ordinates of points along the lines, each measurement being taken at pre-determined intervals of time or distance.
- raster scanning** - a technique of recording or generating an image with an intensity controlled, line-by-line sweep across the entire display surface.
- sample point** - point on the map which is registered (i.e. its co-ordinates and the value at this point) during digitizing; when raster scanning is utilised the sample points are located at places where the tracks cross the contour lines.
- track** - (also scan line or raster line) a line (usually generated by hardware) along which the sample points are located during raster scanning.

'One cannot create something from nothing (although the reverse is true); therefore, it is extremely important that the initial external data structure for digitizing data should contain all of the information necessary to satisfy anticipated internal and external data structure requirements.'

A.H.Schmidt (1976)

INTRODUCTION

In the earth sciences one of the most frequently used media for the display and storage of spatial data is the contour map. Such a map consists of a set of lines, each of them joining the points on the plane where the function describing the physical phenomenon has a constant value.

The development of different methods and techniques of computerised spatial data analysis created the need to have input data in a form readable by digital computers, namely a stream of alpha-numeric characters. Thus, if one wishes to use data mapped by contours for computer analysis the first operation one has to perform is to convert the information contained in the contour maps into a sequence of numbers. In this work some aspects of the digitization of contour maps (i.e. the process of converting them into digital form) are discussed.

To begin with the following question should be asked: is it necessary to use existing maps to produce a digital surface model?

For topographic surfaces, digital data can be obtained directly from air-photographs by use of a stereoplotter (Evans, 1972). At the moment, however, such facilities are not very commonly available and production of data is rather expensive. So, even if Boyle's statement 'The future of automated cartography is dominated by the cost and time of digitization of existing maps' (Boyle, 1976) may seem to be too categorical, the fact is that, even in fields in which there is an alternative way to obtain digital data, the need for a method that enables the creation of a digital surface model from existing maps without special expense and using available hardware is very great.

On the other hand, in sciences like geology or meteorology, one cannot use the orthophotomap method to produce digital data but the question mentioned at the beginning is still relevant. In geology, for example,



the existing maps were produced from field measurements - by definition digital. Is it possible to use these data in further computations and analysis instead of data from contour maps? If the contour map was produced only by means of mathematical operations the answer is, Yes, it is. Such operations can be repeated very easily (if it is necessary) by means of the computer. But more often the contour map is not only the result of an interpolation method but also reflects the experience of the (often very many) people who drew it. In this instance it would be a waste of potential information if we started from the measurement points again.

Without forgetting the alternative ways of obtaining digital data, this work will concentrate on looking at methods of digitizing contour maps which may be performed very quickly using devices no more complicated than the commonly available table-digitizer.

CHAPTER 1

DIGITIZATION OF CONTOUR MAPS

1.1 A contour map as a source of information

A contour map is a graphical image of a function of two variables describing certain physical phenomena. Since a contour map portrays three-dimensional phenomena on a two-dimensional paper sheet it is impossible to describe the function in a full manner and some information is lost. In fact, a contour map is not a representation of any particular surface (function), it is a representation of a class of surfaces (functions). The common property of the functions belonging to this class is that at points on particular contour lines, all functions have the same value. In the intercontour regions the value of the functions may be different.

If such functions are denoted by $f(x,y)$ then the contour lines making up the map satisfy the equation

$$f(x,y) = Z_k$$

where Z_k ($k= 1, 2, \dots, K$) is a given set of values.

Although the only points on the map at which the function has an explicit value are those situated on the contour lines, the values of the function at any point on the map sheet may be estimated. Provided that the function in question $f(x,y)$ is continuous for any arbitrary point (x,y) situated between two contour lines (let us say Z_k and Z_{k+1}) the value of the function at this point must satisfy the following condition:

$$(f(x,y) - Z_k)(f(x,y) - Z_{k+1}) < 0$$

To prove this it is sufficient to notice that this inequality is true when,

$$f(x,y) - Z_k > 0 \text{ and } f(x,y) - Z_{k+1} < 0 \tag{1}$$

or
$$f(x,y) - Z_k < 0 \text{ and } f(x,y) - Z_{k+1} > 0 \tag{2}$$

From (1) we obtain

$$Z_k < f(x,y) < Z_{k+1}$$

and from (2)

$$Z_k > f(x,y) > Z_{k+1}$$

which are the only possible estimations of a value of the function at points situated between the contour lines Z_k and Z_{k+1} .

Up to now it has been implicit in the discussion that determining the value of the function at points not situated on contour lines involves some method of approximation.

As mentioned earlier in this thesis it has been assumed that the function $f(x,y)$ is single-valued and continuous, i.e. the surface is without vertical cliffs or faults and it is determined in the whole area given by the map sheet. Morse (1968) considers a more general mathematical model of contour-mapped data.

1.2 Digitizing

A contour map can be considered as a graphical image of a function of two variables. Before such a function can be processed by a digital computer the contour map needs to be digitized or in other words converted from its analogue structure into a discrete form as a finite string of numbers. This may be done by several methods and may be performed manually or semi-automatically. The choice of the digitizing method should be determined not only by the available hardware but also by the subsequent use of the data.

The general purpose of digitizing is to create a bank of spatial data in machine readable form which may later be searched, displayed (perhaps in its initial form, i.e. as contour maps plotted by computer) or analysed mathematically.

The digitization method must guarantee that, in spite of the loss of information that usually accompanies analogue-digital conversion, our aims will be achieved. In addition it should allow the possibility of error estimation.

A general classification of digitizing methods is shown in Figure 1. The diagram is based on a classification introduced by Wesley (1974). The problems connected with classification of digitizing methods are discussed also by Yoeli (1975) and Türke (1978).

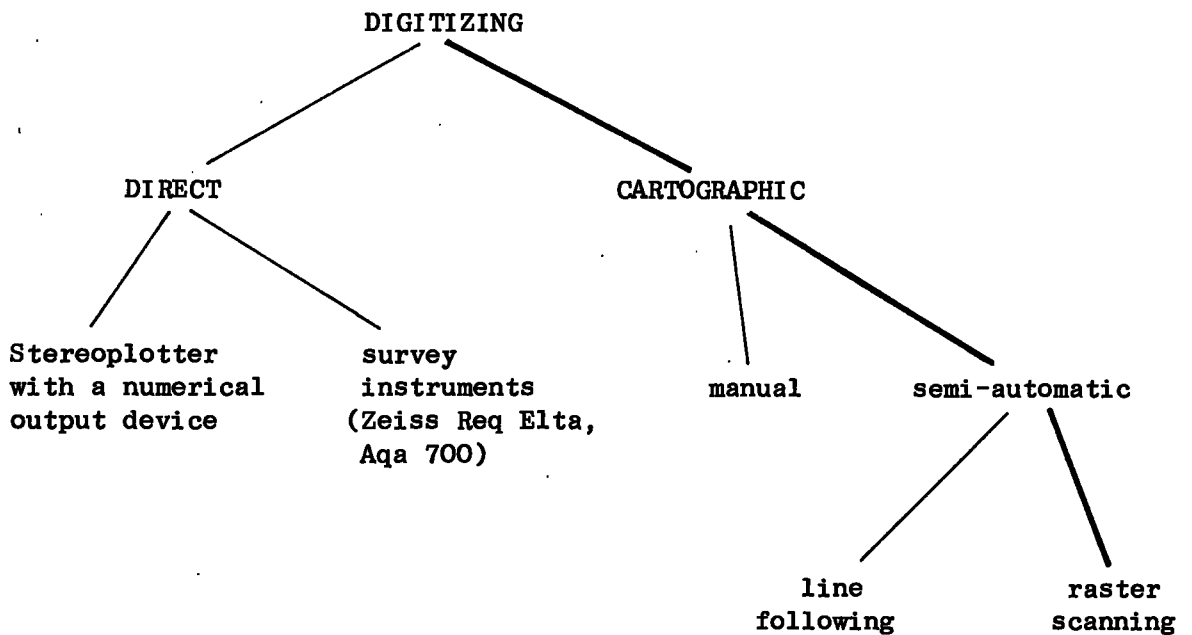


FIGURE 1. Classification of digitizing methods

The thick lines on the diagram show the aspects which are emphasised in this thesis.

1.3 Manual methods of digitizing

The division of cartographic digitizing into manual and semi-automatic methods needs some explanation. Manual methods are those involving human interpretation of contour map data and visual interpolation during digitizing. Clearly such a process cannot be performed by machines. On the other hand, although both of the semi-automatic techniques of digitizing can be performed manually without any special equipment, by using hardware like

the table digitizer the speed and accuracy of digitizing can be increased enormously.

There are several methods of manual digitizing. They can be classified into two groups defined by the positioning of the sample points. The first group consists of methods in which the location of the sample points is not related to the positioning of the contour lines; in other words, the co-ordinates of the sample points are generated and the same sample pattern may be applied to different contour maps.

The most frequently used methods of generating sample points are:

- a) regular grids, usually square grids; triangular grids and rectangular grids may be also used,
- b) random points - in the domain given by the map area a set of randomly distributed points is generated (Mark, 1975; MacDougall, 1976),
- c) mixed patterns like the stratified systematic unaligned pattern (MacDougall, 1976).

The second approach is different, the set of sample points being dependent on the contour map to be digitized. It is called a surface specific point pattern (Mark, 1975). By interpretation of surface behaviour places with special significance are selected, for example ridges, valleys, peaks and pits and so on. The sample points are located in these places.

In all manual methods, however, the result depends ultimately on the operator, for he has to estimate the value of the function at a point by inspection and visual interpolation from surrounding contour lines.

1.4 Semi-automatic techniques of digitizing

There are two semi-automatic techniques of digitizing; line following and raster scanning, and it is often emphasised that for contour digitizing

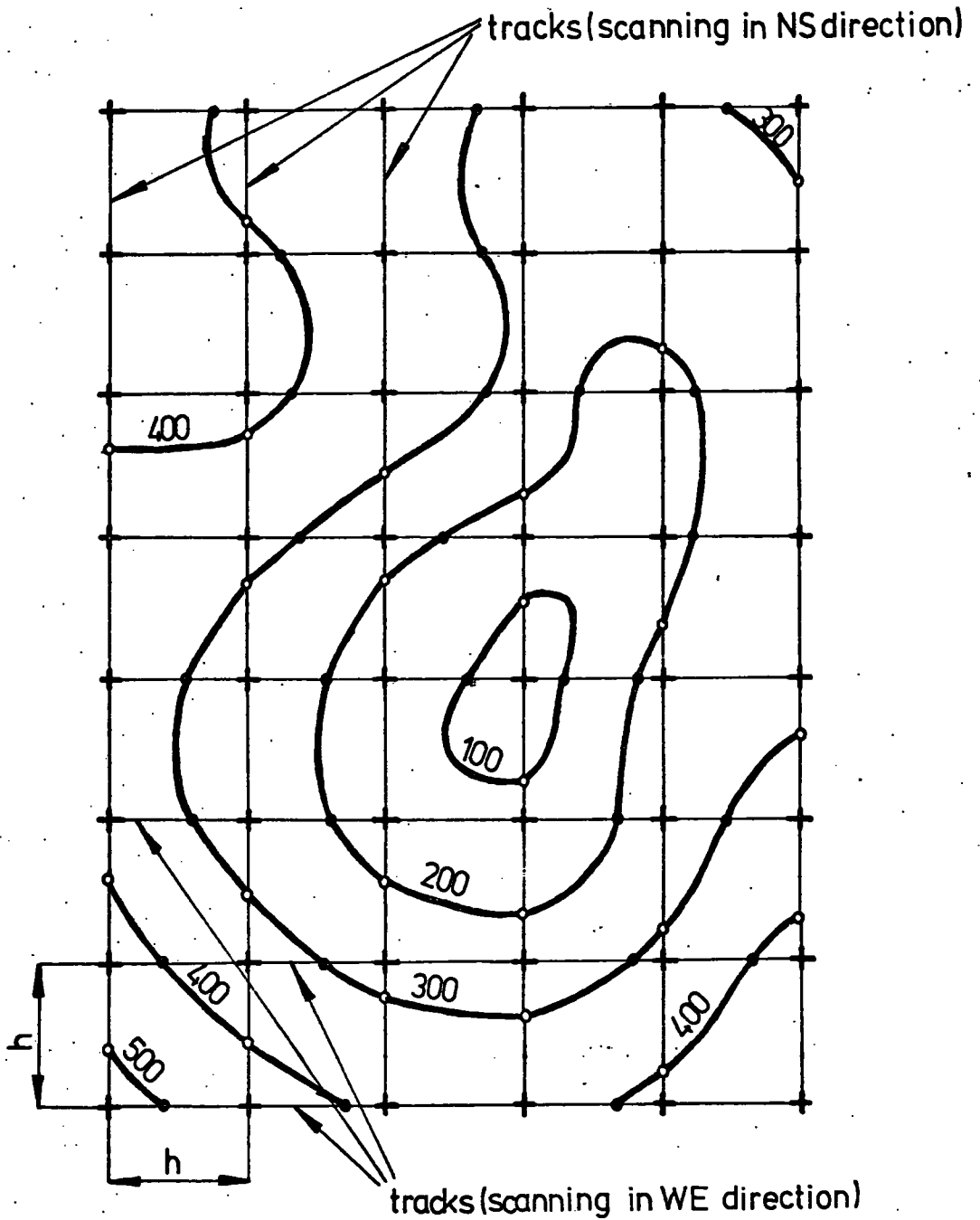


Figure 2. Raster Scanning

closed circles ● indicate points to be registered during raster digitizing in W-E direction.

open circles ○ indicate points to be registered during raster digitizing in N-S direction.

only the line following and raster scanning are practical e.g. Yoeli (1975).

Line following:

When the contour lines on the map to be digitized are followed one by one and points along them are registered, then the map is being digitized by the line following technique. Usually the sample points on the contour lines are taken in predetermined intervals of time or distance and co-ordinates at the points are registered either absolutely or incrementally. To represent the contours efficiently, the sample points are spaced along the contour in inverse proportion to its curvature e.g. Boehm (1967), Rhind (1974). Different algorithms have been developed to generate such a sequence of points; usually it is done using so-called sagittal tolerance, e.g. Hunting Surveys Ltd. (1977), and also MacDougall (1976) - algorithm for digitizing the boundaries of regions.

Raster scanning:

Raster scanning is the process of digitizing along straight lines superimposed on the contour map. The set of lines is called a raster and the lines themselves are called tracks, raster lines or scan lines. The points where the tracks cross contour lines are registered during digitizing (Figure 2). The principle of analysing the picture in raster scanning is similar to the generation of the image on a television screen.

Two parameters have to be determined before digitization by the raster scanning technique; namely the direction and the spacing of the tracks. When using table digitizers the tracks are usually located parallel to one of the co-ordinate axes because of the hardware implementation. When electronic devices are used (Flying Spot Scanners) the direction of the tracks may be different (Figure 3). Although the direction of the tracks is important - to minimise the errors the tracks should be perpendicular to the contour lines - it is practically impossible to determine for a relatively complicated contour map the optimal direction

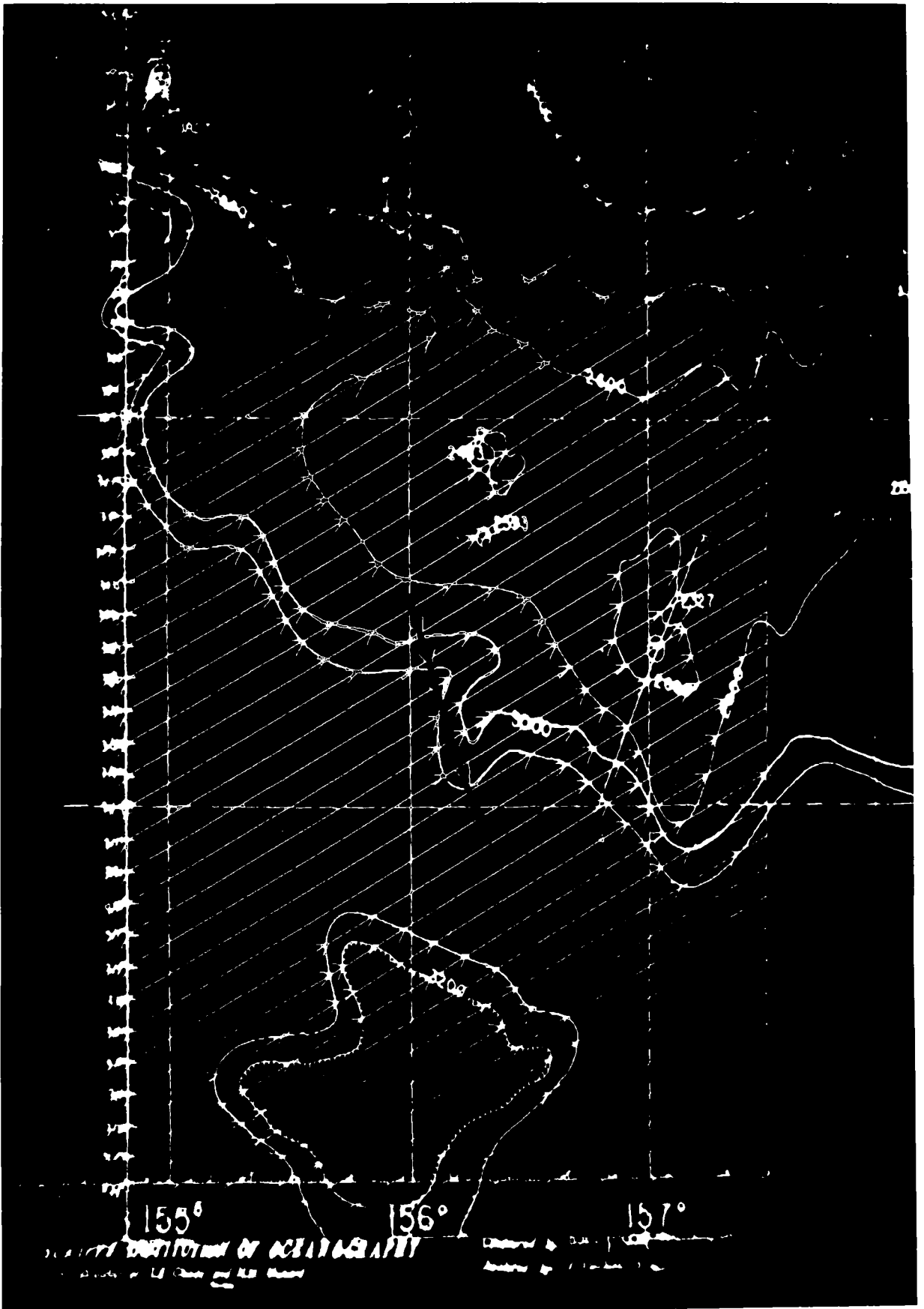


Figure 3. Location of tracks in the SEASCAPE system (from Ryan, 1976).

of scanning. For choropleth map digitization when the data are to be stored in chain mode the right choice of scanning direction may reduce the required memory space. Amidon and Akin (1971) discuss relationships between the direction of scanning and required memory space.

The proper choice of track density is of greater importance. It should be determined by the purpose and scale of further analysis.

1.5 A Comparison between the line following technique and the raster scanning technique

Although line following and raster scanning are both based on the same principle of digitization of contour maps - they both give us a sequence of points situated on contour lines - there is a difference in the amount of information they convey.

The line following technique seems to be a natural method of digitizing line-drawn pictures such as contour maps. It has a long history and its various aspects are discussed in several papers. Freeman (1974) gives a review of applications using data derived by this technique, Evans (1972) describes the creation of inter-contour slope lines, Boyle (1976) gives some information about the mass digitization of contour maps using this technique automatically and Rhind (1974) discusses the choice of location of sample points along a contour line.

The popularity of the line following technique is due to two factors; it is logical in its construction from the formal point of view, i.e. if we were not limited by technical restrictions (memory space, hardware accuracy) it would allow the possibility of a complete reconstruction of the initial contour lines or, in other words, by reducing the distance between sample points we can always produce an adequate approximation to the contour lines. Secondly, the line following technique seems to convey all the information from the contour map. It seems logical that, if the contour map consists of contour lines, then by digitizing them one

digitizes the map. However, the line following technique does not convey the spatial information necessary for much analysis of spatial problems. Consider an arbitrary point on the map sheet not situated on any contour line. The value of the function at this point can be estimated by inspecting the surrounding contour lines: this information is implicit in the contour map. However, a first problem is that the value at that point cannot be estimated from the sequence of points obtained by the line following technique. Increasing the number of sample points along the contour lines does not help. Secondly, nothing can be said regarding the nature of a function along a straight line joining subsequent sample points, even though that information is contained in the contour map.

Before the nature of the spatial information conveyed by the raster scanning technique can be discussed it has to be proved that this method is correct from the formal point of view and allows complete restoration of the initial data. Clearly, if we were not limited by hardware, we could space the tracks at an arbitrarily small distance. The next step is constructing the grid and computing the values at the nodes of a grid square by the quantization method. It may be assumed that the grid size is small enough to satisfy the postulate that no contour line crosses the side of the grid square more than once - otherwise the track spacing could be reduced further.

This situation is shown in Figure 4.

Let R be the distance between the real contour line and the line obtained by linear approximation from quantized matrix data (this line on Figure 4 is shown as a dashed line). The maximum value of R is then $\frac{3\sqrt{2}}{4}h$. As h tends to zero so R tends to zero, or in other words, the lines derived from quantized matrix data tend to the real contour lines.

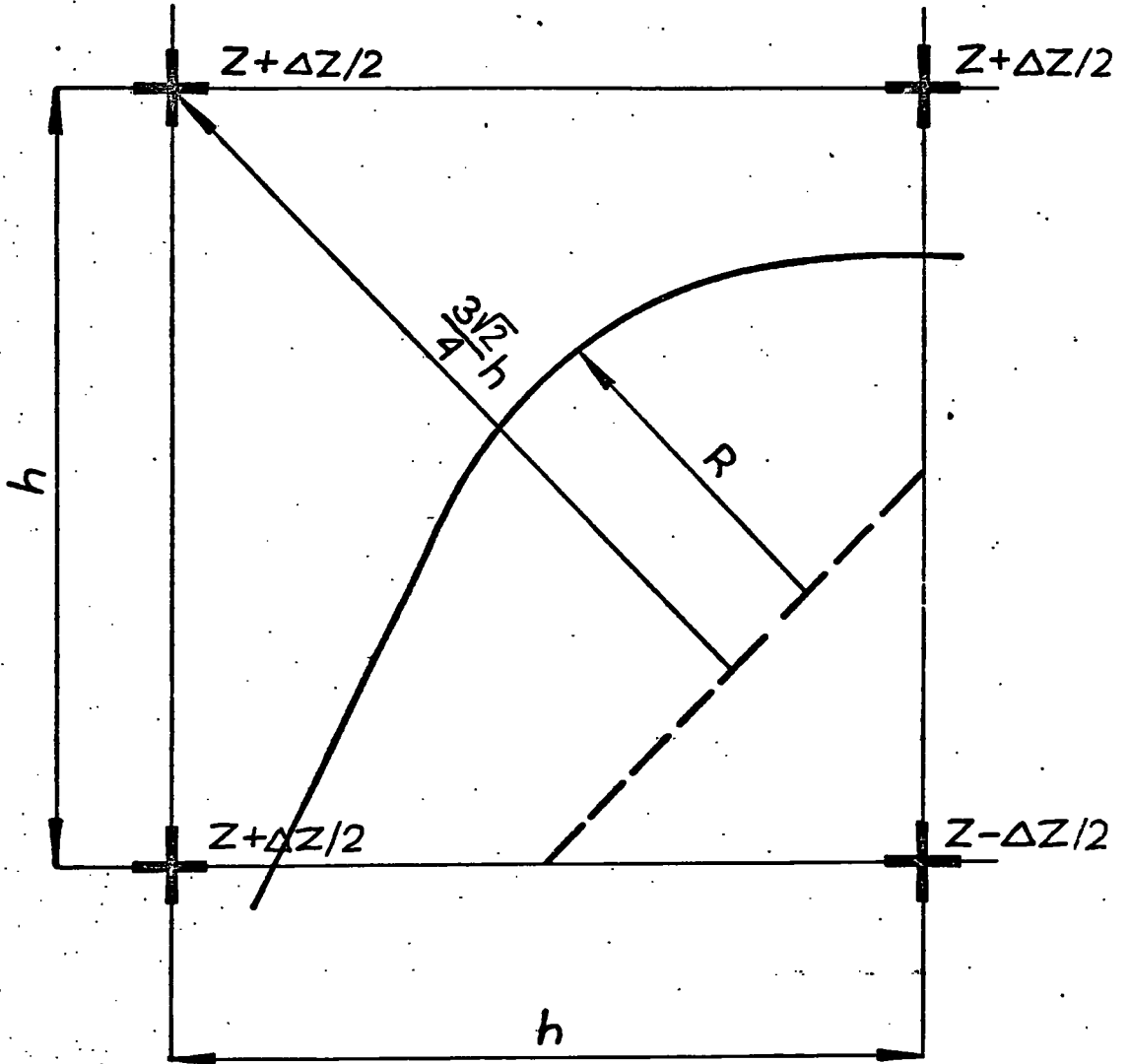


Figure 4. Estimation of maximum distance between the contour line on the map (thick line) and the contour line drawn from quantized data using linear interpolation (dashed line).

Having shown that the raster scanning technique is formally correct, the solution to the first problem posed earlier is that, for any arbitrary point, the track spacing can be chosen so that this point will be situated on the track so the value at this point can be estimated by inspecting surrounding sample points on the track. The solution to the second problem is obvious; the values the function takes between two successive points along the track must lie between the values of the function at those two points.

The additional advantage of using the raster scanning technique is that the process of digitizing may be performed in more than one step. If, for example, it turns out that for certain applications the track spacing h is too wide one can digitize again with new tracks also spaced h apart. By shifting their location by $h/2$ in relation to the previous ones, we obtain a situation equivalent to a digitization with track spacing $h/2$ and we can continue this process if desired. No work or information from the previous steps is wasted.

CHAPTER II

PROCESSING OF DATA OBTAINED BY RASTER SCANNING

2.1 Creation of ^{the} data matrix

The most convenient structure for analysis of a function of two variables by means of digital computers is a data matrix, i.e. the set of values of the function given at the nodes of a square grid. The two main reasons for that are: firstly, the numerical analysis of spatial data is usually based on this particular structure; and secondly, programming of spatial problems is made easier because all the commonly used computer languages allow handling of two-dimensional arrays.

To obtain matrix data when raster scanning has been used for digitizing, the contour map values are approximated at points equally spaced along the scanning tracks. If the distance between the nodes of approximation is the same as the track spacing, we obtain matrix data equal-spaced in x and y.

Three different approaches were considered and the efficiency of their interpolation methods was compared.

Linear interpolation: for each successive pair of sample points (where tracks cross contours) the equation of a straight line is found and from this equation the values at nodes situated between those points are calculated. Using linear interpolation one cannot determine values at the nodes situated between the edge of a map and the first sample point, nor at those situated between the last sample point and the edge. Linear interpolation fails when two successive sample points have the same value; then a constant equal to the value at the sample points is obtained. With this method the value at any node cannot be greater (less) than the greatest (smallest) value of contour lines. In other words, if matrix data obtained by linear interpolation are displayed on a

contour map the reconstituted map will lose the extreme-valued peaks and pits present on the original document. When the values of the function at the nodes are determined by linear interpolation the maximum error is not greater than a contour interval.

Spline interpolation: for each four successive sample points the equation of a spline function (Appendix A) can be found. It is valid for the segment between two middle points. At the nodes situated there, the values are calculated from the equation. For this method the edge effect is even larger; at each end of a track value at nodes situated between the edge and the second sample point cannot be determined. In fact, the more sophisticated the interpolation method, the bigger is the edge effect.

It is difficult to give a quantitative estimate of maximum error for this method, but it may exceed Δz (contour interval). The greatest errors will occur mainly at plateau tops. In fact, if the change in slope is sharp enough, the reconstituted map may well have contour lines which were not on the original map.

Quantization method: the values the function takes between two successive sample points on the track P_k and P_{k+1} with contour values Z_k and Z_{k+1} ($Z_k \neq Z_{k+1}$) are limited by values Z_k, Z_{k+1} . At all nodes situated between P_k and P_{k+1} we estimate values at the function as $(Z_k + Z_{k+1})/2$. If the contour values are the same, i.e. $Z_k = Z_{k+1}$ then we attach to all nodes the value $Z_k + \Delta Z/2$ if the function in the segment (P_{k-2}, P_k) is increasing in value, or $Z_k - \Delta Z/2$ if the function is decreasing in value in this segment. In the same way we can attach values to nodes situated between the edges of a map and the first sample point (they will be $Z_1 + \Delta Z/2$ or $Z_1 - \Delta Z/2$ depending on the behaviour of the function in the segment (P_1, P_2)). A formal description of the algorithm to compute the quantized data is included in Appendix B.

In every case the maximum error generated by this method is not

greater than $\Delta Z/2$. In other words the set of contour lines on the reconstituted map and the initial document will be the same. To aid visualization of the effects of these methods a track with approximated values plotted is shown on Figure 5 by way of example.

To examine the influence of the direction of scanning the test map (Map 1) was scanned twice (from North to South and from West to East). During digitizing only 100 m contours were taken into account. The co-ordinates of sample points were read with an accuracy of ± 0.25 mm.

As a basis for comparison a set of matrix data digitized by hand on a 200 x 200 m grid were used and the values of the function were estimated to the nearest 5 m from 20 m contours. These values were compared with values computed by the three methods mentioned above. Results of the error analysis are presented in Table 1.

As can be seen from the table, only the quantization method permits values to be estimated at all nodes, and it minimizes the maximum error. Although the mean absolute error is smallest for spline interpolation, only about 62% of the nodes have determined values. The results obtained by the quantization method do not depend on the direction of scanning.

The quantization approach, i.e. dividing of the range of values into classes and attaching all members of a class with the same value unique to that class, derives from digital picture processing where the picture must be quantized (Rosenfeld and Kak, 1976). For a contour map the number of quantizing levels needed is normally determined by the contour spacing and cannot be arbitrarily set by the operator unless he can accept much lower resolution than is present in the initial map.

2.2 Spatial analysis based on the quantized matrix data

Although quantization of matrix data seems to yield a direct representation of the contour map and is characterised by the minimization of maximum error, it cannot always be directly used in its original form

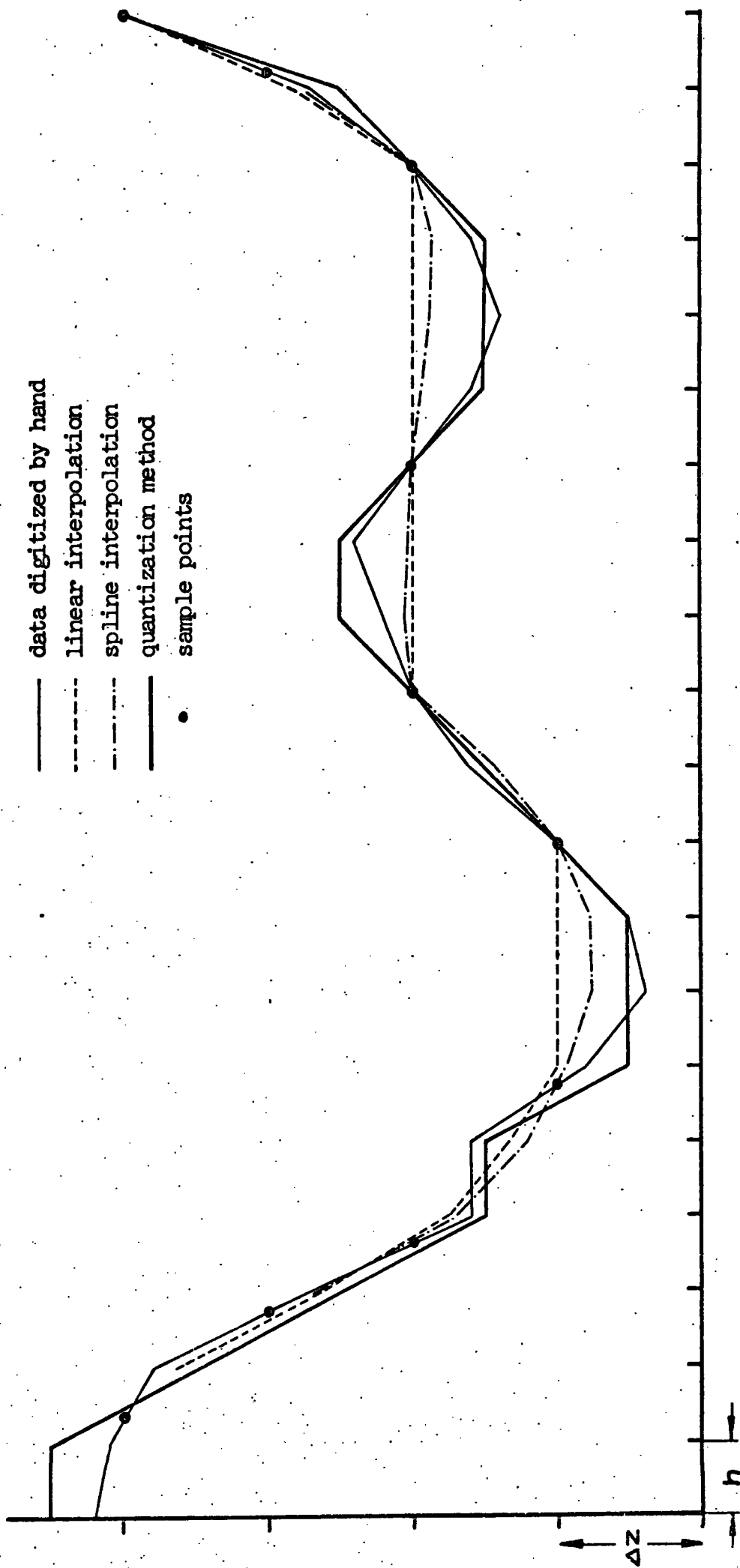


Figure 5. Methods of approximations of values at grid points along track

	Linear interpolation		Spline interpolation		Quantization method	
	W - E	N - S	W - E	N - S	W - E	N - S
Direction of scanning						
Total number of points registered during scanning	256	234	256	234	256	234
Total number of mesh points N	441	441	441	441	441	441
Total number of mesh points with undeterminable value N_W	94	76	164	156	0	0
Ratio $\frac{N - N_W}{N}$	0.787	0.828	0.628	0.646	1.0	1.0
Mean error	0.089	0.478	-0.882	1.378	0.760	0.760
Standard deviation of mean error	29.574	29.285	24.556	22.877	25.689	25.689
Mean absolute error	20.417	20.348	16.824	15.859	21.168	21.168
Standard deviation of mean absolute error	21.367	21.039	17.880	16.520	14.540	14.540
Maximum absolute error	100.0	90.0	91.021	79.410	50.0	50.0

to solve spatial problems. As can be seen from Table 1, the mean error obtained during quantization tends to zero. It may be also assumed that in the domain under question (i.e. given by the map sheet) the errors are randomly distributed. When values of the errors are small relative to values of the function to be processed, the quantized matrix data can be used directly for global computations.

For example, the results of trend surface analysis should not be affected by quantization errors. Maps 4 and 5 both show fourth order polynomials found by the least squares method. The former (Map 4) uses matrix data digitized by hand from 20 m contours, the latter (Map 5) uses quantized matrix data from digitization by raster scanning of 100 m contours. For these, as for lower order polynomials, differences in coefficient value are negligible; moreover, the maps appear visually to be highly similar.

Problems such as computation of the total volume between surfaces given by their contour maps may be solved easily by using quantized matrix data. This sort of computation occurs in the estimation of natural resources and changes over time.

Computation of local geometrical properties of the surface (such as gradient, aspect, convexity) directly from a contour map is based on measurements of distances between contour lines. The process of quantization changes these distances, therefore the values of parameters derived from the quantized data will incorporate some errors. It is obviously important, if one wishes to broaden the scope of the quantization method, to investigate the possibility of estimating these errors knowing only the variability of the contour spacing and the track spacing; determining the relationship between the errors and the track spacing enables choice of an appropriate track spacing before digitizing to ensure that the parameters are estimated to the required accuracy.

To this end, the distances between the 100 m contour lines were measured

on the tracks (i.e. the distances between sample points) and the results are given in Table 2 (for W-E direction of the tracks) and Table 3 (for N-S direction of the tracks). For each track the number of contours, minimum, maximum and mean distance between contours and the standard deviation of mean distance are given. All distances are given as a multiple of the track spacing h i.e. for example 1.5 denotes that distance is 1.5 h .

In determining the contour spacing, allowance must be made for the edge effect, i.e. it is not known where the next contour line would be if the track were to be extended (off the map sheet). The smallest possible maximum error is obtained by assuming that the first contour off the map is situated at a point whose distance from the edge of the map is equal to the distance between the last contour on the map and the edge. If the maximum or minimum distances have been estimated by this method they are indicated by '*' on Tables 2 and 3. Figure 6 shows the histograms of mean distances between contours on one track.

The variance of an arbitrary function $g(x,y)$ may be estimated by variances of the variables in the following way (see, e.g. Davies, 1961)

$$\text{Var}(g) = \left(\frac{\partial g}{\partial x}\right)^2 \text{Var}(x) + \left(\frac{\partial g}{\partial y}\right)^2 \text{Var}(y) \quad (1)$$

where the values of the derivatives are calculated using mean values of x and y . As an example this formula will be applied to estimate the error obtained during computation of gradient S from the quantized data.

$$S(u,v) = \sqrt{\left(\frac{\Delta z}{u}\right)^2 + \left(\frac{\Delta z}{v}\right)^2} \quad (2)$$

where Δz is a contour interval and u, v are the distances between two successive contours in the x and y directions respectively (Figure 6).

Therefore the variance of gradient $\text{Var}(S)$ is given by

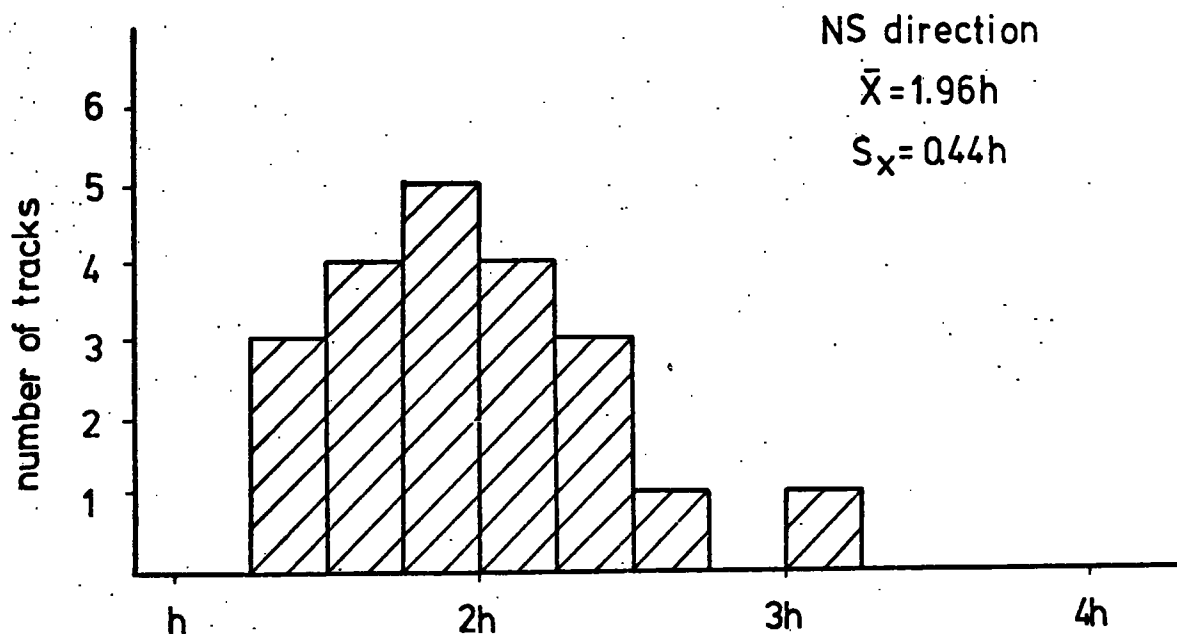
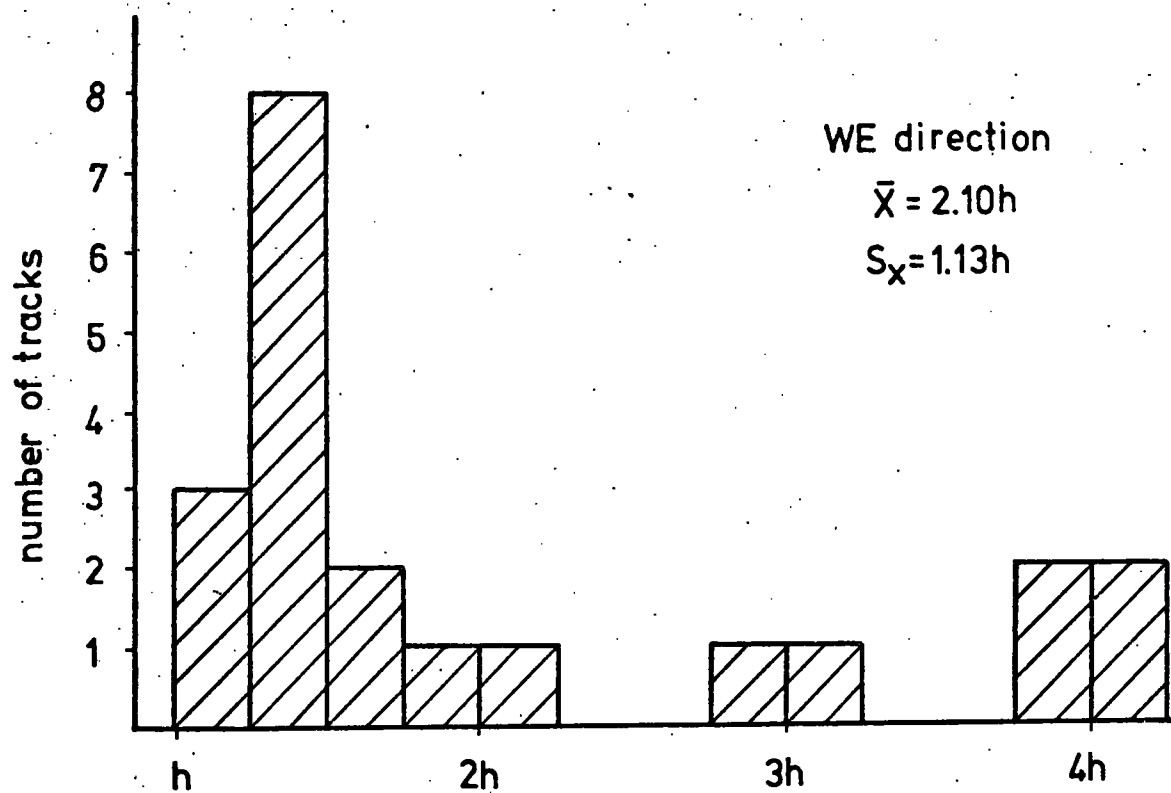
$$\text{Var}(S) = \frac{(\Delta z)^2}{(\bar{u} + \bar{v})} \left(\frac{\bar{v}^2 \text{Var}(u)}{\bar{u}^4} + \frac{\bar{u}^2 \text{Var}(v)}{\bar{v}^4} \right) \quad (3)$$

Table 2

Track number	Number of contours	Minimum distance between contours	Maximum distance between contours	Mean distance between contours	Standard deviation of mean distance	$\overline{U - Uq}$	$\text{Var}(U - Uq)$
1	13	0.25*	9.37	1.48	2.33	0.0	0.13
2	14	0.25*	5.87	1.35	1.59	0.02	0.14
3	13	0.5	5.37	1.47	1.45	-0.01	0.08
4	5	1.25	7.50	4.08	2.34	0.0	0.03
5	5	0.62	14.25*	4.87	4.78	0.18	0.15
6	7	0.75	13.25*	3.60	4.08	0.02	0.21
7	7	0.62	12.50*	3.81	4.36	0.08	0.12
8	8	0.62	6.00*	2.76	1.74	-0.05	0.02
9	10	0.50	7.50*	2.20	2.03	-0.02	0.05
10	13	0.12	2.50	1.49	0.80	-0.03	0.16
11	12	0.37	4.25	1.66	1.26	-0.01	0.10
12	14	0.125	5.25	1.40	1.36	-0.01	0.10
13	13	0.25	2.87	1.50	0.94	-0.02	0.07
14	15	0.50	4.37	1.35	1.09	0.01	0.03
15	17	0.37	4.50	1.13	0.96	0.0	0.06
16	15	0.5	3.75	1.33	0.94	0.0	0.08
17	19	0.12	3.62	1.05	0.78	0.04	0.18
18	17	0.37	4.50	1.14	0.97	0.0	0.11
19	16	0.37	6.00*	1.35	1.39	0.0	0.03
20	14	0.5	15.25*	1.87	3.72	0.0	0.04
21	9	0.37	19.25*	3.21	5.80	-0.01	0.04

TABLE 3

Track number	Number of contours	Minimum distance between contours	Maximum distance between contours	Mean distance between contours	Standard deviation of mean distance	$\frac{U - Uq}{U}$	Var (U - Uq)
1	8	0.62	6.12	2.56	2.03	-0.01	0.11
2	7	0.37	7.25*	3.03	2.11	-0.06	0.06
3	9	0.87	5.00*	2.45	1.42	0.0	0.12
4	13	0.37	5.25*	1.75	1.41	-0.01	0.08
5	11	0.62	4.25	1.93	1.22	-0.02	0.08
6	13	0.50	3.75*	1.65	0.98	-0.01	0.20
7	9	0.37	5.50	2.33	1.78	0.01	0.10
8	9	0.50	5.50	2.31	1.89	0.02	0.11
9	9	1.00	4.75	2.16	1.33	0.04	0.15
10	11	0.50	6.00*	2.00	1.73	0.0	0.07
11	14	0.25	4.62	1.34	1.28	-0.03	0.09
12	10	0.75	6.50	1.97	1.92	-0.02	0.03
13	10	0.25	7.37	2.07	2.60	-0.01	0.14
14	13	0.25	5.50*	1.62	1.54	-0.02	0.09
15	10	0.25*	8.50*	2.21	2.50	0.04	0.05
16	10	0.50	5.00*	2.13	1.47	0.0	0.08
17	11	0.37	6.00*	1.95	1.72	0.0	0.06
18	13	0.37	3.75	1.56	0.90	-0.03	0.12
19	13	0.37	4.50	1.50	1.16	0.0	0.03
20	16	0.25	5.12	1.30	1.25	0.01	0.06
21	15	0.25	3.75	1.38	1.11	-0.02	0.03



average distance between contours on one track

Figure 6. Histograms of average distance between contours on one track

To estimate the variances $\text{Var}(u)$ and $\text{Var}(v)$, the distances between contours obtained by linear interpolation from quantized data (on Figure 7, u_q, v_q) were obtained and subtracted from the distances between the contours (on Figure 7 u, v). In the two last columns of Tables 2 and 3 the mean difference and the variance of mean difference for each track are given for W-E scanning in Table 2 and for N-S scanning in Table 3. The histograms of the variance of mean difference are given on Figure 8.

For the test data $\bar{u} = 2.10h$, $\bar{v} = 1.96h$, $\Delta z = 100$ m, $\text{Var}(u) \approx \text{Var}(v) = 0.09h^2$, $h = 200$ m, thus

$$\text{Var}(S) = 0.0013$$

A ratio of standard deviation $\text{Var}(S)$ to a mean gradient $S(\bar{u}, \bar{v})$

$$\frac{\sqrt{\text{Var}(S)}}{S(\bar{u}, \bar{v})} = 0.105 \quad (4)$$

gives an estimate of average error during computation of gradient from quantized data for the test area. For a track spacing of 200 m the error is about 10%. For a given pair (u, v) where $u, v = 1h, 2h, \dots, 6h$ the error may be estimated from Table 4. For any pair the value of standard deviation (squared value from (3)), the value of gradient computed from (2) and the ratio (4) are given.

The error obtained in computing any parameter from quantized data may be estimated using a similar procedure.

These considerations have a wider aspect, since the variability of spacing of contours may be estimated from the map before digitizing (by measurement of distances between contours in randomly chosen places), by use of Table 4 it is possible to deduce the track spacing necessary to satisfy the required accuracy of computation of gradient. For example, if the error is to be about 5% then the average distance between contours should be about $4h$ in both directions or, in other words, in the case of

TABLE 4

6h	0.1480	0.0356	0.0150	0.0081	0.0053	0.0042
	0.5069	0.2635	0.1863	0.1502	0.1302	0.1179
	0.2919	0.1351	0.0806	0.0542	0.0409	0.0354
5h	0.1471	0.0349	0.0146	0.0082	0.0060	0.0053
	0.5099	0.2693	0.1944	0.1601	0.1414	0.1302
	0.2885	0.1296	0.0752	0.0514	0.0424	0.0409
4h	0.1455	0.0338	0.0145	0.0094	0.0082	0.0081
	0.5154	0.2795	0.2083	0.1768	0.1601	0.1502
	0.2824	0.1209	0.0695	0.0530	0.0514	0.0542
3h	0.1424	0.0325	0.0167	0.0145	0.0146	0.0150
	0.5270	0.3005	0.2357	0.2083	0.1944	0.1863
	0.2702	0.1083	0.0707	0.0695	0.0752	0.0806
2h	0.1352	0.0375	0.0325	0.0338	0.0349	0.0356
	0.5590	0.3536	0.3005	0.2795	0.2693	0.2635
	0.2419	0.1061	0.1083	0.1209	0.1296	0.1351
1h	0.1500	0.1352	0.1424	0.1455	0.1471	0.1480
	0.7071	0.5590	0.5270	0.5154	0.5099	0.5069
	0.2121	0.2419	0.2702	0.2824	0.2885	0.2919
	1h	2h	3h	4h	5h	6h

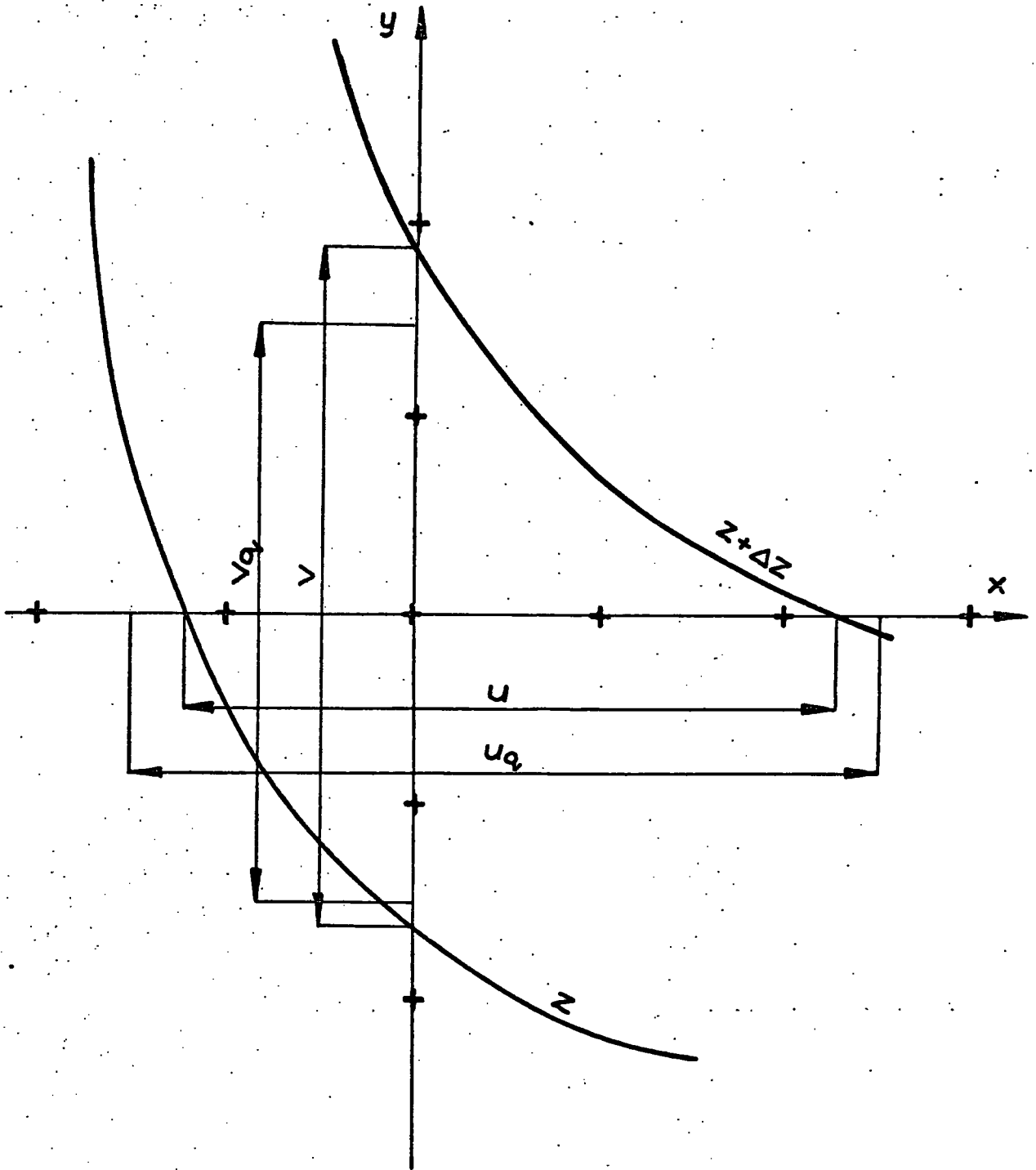


Figure 7. Figure to show how inter-contour distances are measured using quantized data (u_q, v_q) compared with real distances (u, v)

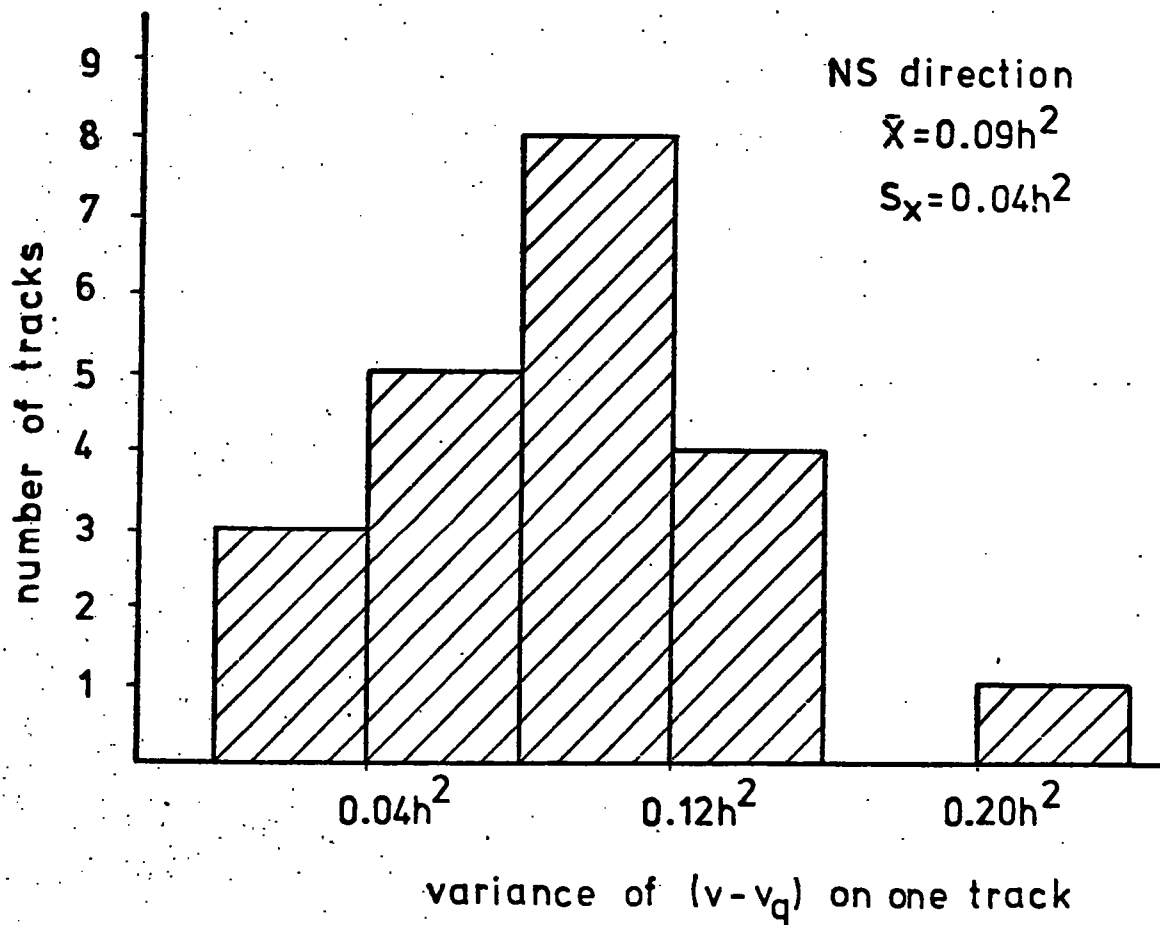
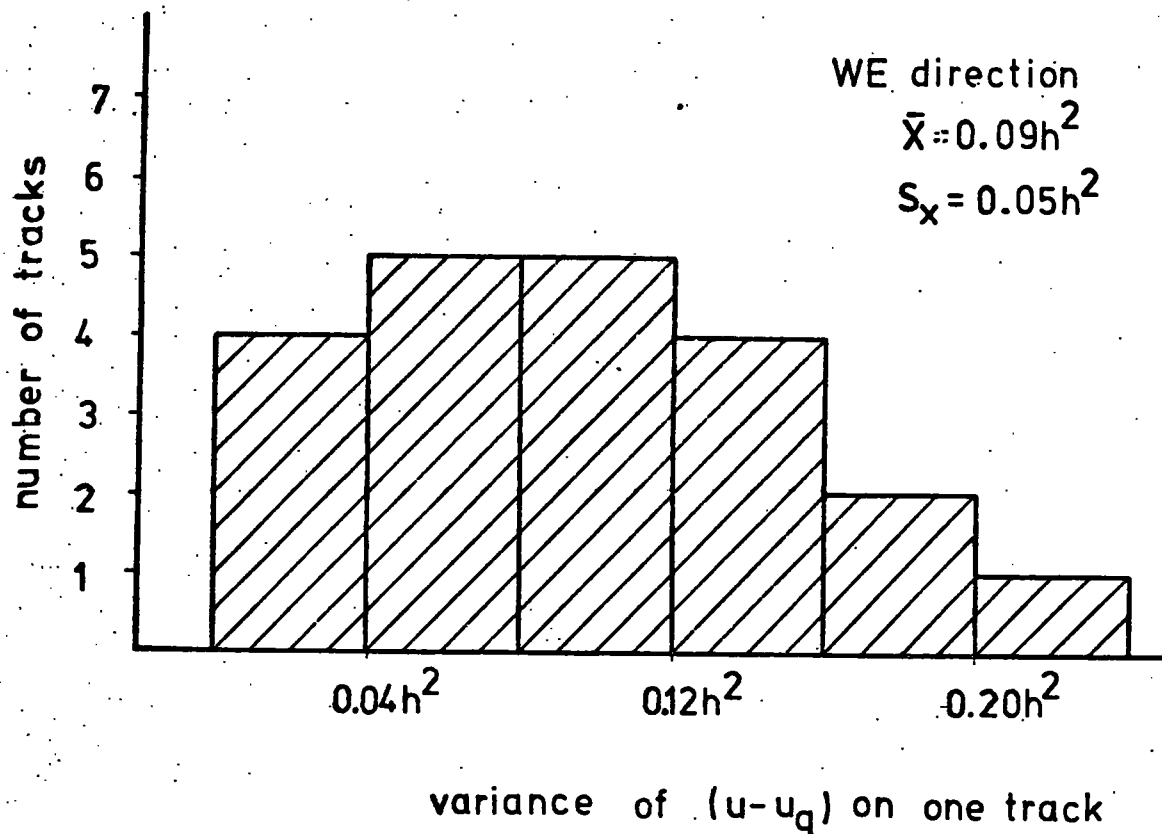


Figure 8. Histograms of the variance of $(u-u_q)$ and $(v-v_q)$

the test data to obtain this accuracy the track spacing should be halved; then the average distance between contours would be 4.20h in W-E direction and 3.92h in N-S direction.

Park and others (1971) give a simplified formula to compute a gradient to avoid taking square roots

$$S(u,v) = \sqrt{u^2 + v^2} \approx \max(u,v) + 0.5 \min(u,v)$$

If this simplification had been used in the test case it would have introduced an additional 6% error.

The geometrical properties of the surface may be calculated in a different way, namely, before computations some generalisation of surface (e.g. smoothing of quantized data to reduce the influence of quantization noise) is done. There are two main strategies available:

- (a) the least squares method is used to determine a polynomial over a certain area. Usually it is a quadratic applied to each 3 x 3 matrix position square. The values of the required parameters may be derived directly from the equation of the polynomial. This method is used by Evans and Young (1977).
- (b) Alternatively, to generalise the surface the local weighted averaging of matrix elements, usually called filtering may be used. A 3 x 3 filter is denoted in the following form:

$$\frac{1}{\sum_{i=n-1}^{n+1} \sum_{j=m-1}^{m+1} w_{ij}} Z_{nm} \left\{ \begin{array}{ccc} w_{n-1,m+1} & w_{n,m+1} & w_{n+1,m+1} \\ w_{n-1,m} & w_{n,m} & w_{n+1,m} \\ w_{n-1,m-1} & w_{n,m-1} & w_{n+1,m-1} \end{array} \right\}$$

where w_{ij} is a weight of value Z_{ij} . In an abbreviated form only a matrix of weights is given.

Values of the derivatives at the point (n,m) in the formulae for required parameters are usually estimated by a finite difference method

as follows:

$$\frac{\partial f}{\partial x} \approx \frac{G_{n+1,m} - G_{n-1,m}}{2h}$$

$$\frac{\partial f}{\partial y} \approx \frac{G_{n,m+1} - G_{n,m-1}}{2h}$$

$$\frac{\partial^2 f}{\partial x^2} \approx \frac{G_{n+1,m} - 2G_{n,m} + G_{n-1,m}}{h^2}$$

$$\frac{\partial^2 f}{\partial y^2} \approx \frac{G_{n,m+1} - 2G_{n,m} + G_{n,m-1}}{h^2}$$

$$\frac{\partial^2 f}{\partial x \partial y} \approx \frac{G_{n+1,m+1} + G_{n-1,m-1} - G_{n+1,m-1} - G_{n-1,m+1}}{4h^2}$$

where h denotes a grid size and $G_{n,m}$ are the smoothed values of the grid points.

For any given number of grid values taken into account during computations the quadratic is more efficient - that is the generalisation is smaller - than filtering because this sort of filtering is in fact the (weighted) least square approximation by the 0-th order polynomial.

Map 6 shows the result of generalisation of matrix data digitized by hand by a quadratic computed for a 3x3 neighbourhood. Map 7 shows the results of the same operation when quantized matrix data were used. As can be seen, some of the quantization noise is still present. The reason for this seems to be the size of the neighbourhood. In fact, for this data 5% of the 3x3 squares do not contain any contour lines, 32% contain only one, 37% two and 26% of squares contain more than two contour lines. In order to calculate parameters dependent upon spatial relationships for each square, at least two contour lines should cross the x-axis and two contour lines should cross the y-axis (the origin of the local co-ordinate system here is in the middle of squares). The correct size of the neighbourhood when data are obtained by the raster scanning technique may be estimated from the average spacing of the contour lines in the directions

of both axes. For the test data, the average spacing of the contour lines in x-direction (W-E) is 2.10h and 1.96 h in y-direction (N-S). It can be expected that the segment with length 4h will be crossed twice by the contour lines. But to obtain the same level of smoothing with an enlarged neighbourhood, the order of the polynomial should be increased.

Map 8 shows the result of generalisation of matrix data digitized by hand by a cubic polynomial computed for the neighbourhood pattern below:

```
      x
     x x x
    x x x x x
     x x x
      x
```

Map 9 shows the results of the same operation for quantized matrix data. A comparison with the original contour map indicates that now the generalisation of the contour lines in both cases is equally good.

Maps 10, 11, 12 show the generalisation of quantized matrix data by three different local filters. For the test data the filter $\begin{pmatrix} 010 \\ 141 \\ 010 \end{pmatrix}$ (Map 11) gives the most satisfactory results, in the two remaining cases the smoothing causes a loss of the extreme-valued contour lines.

2.3 Display of data when digitizing by raster scanning

One of the functions of a spatial data base is the provision of data to create a display in contour map form. If the data in this bank are derived from the digitization of contour maps, it is important to know the difference between the initial contour map and the map constructed by the computer from stored values. Very often the statement that both maps are similar is not enough and it has to be determined at which places (or points) both maps are identical. For example, if the contour map constructed by computer is to be used in the location of a new borehole a set of control points, i.e. the points at which the contour lines on

both maps are the same has to be known. This set of control points consists of the sample points taken during digitization.

The display of data in contour map form when they were obtained by the line following technique and have been stored in ordered form is a relatively easy task. When the raster scanning technique is applied to digitize the contour map the matter is more complicated. It has already been stated that if quantized matrix data are used to produce a contour map the quantization causes deformation of the contour lines (Map 3) and it can only be said that the contour lines on the reconstructed map are situated not more than $\frac{3\sqrt{2}}{4} h$ (where h denotes scan-line spacing) from the original contour lines. To visualise the quantization noise more effectively, a test picture (Map 13) was scanned (with the same track spacing as was used for scanning Map 1) and quantized, then reconstructed by the Calcomp GPCP (General Purpose Contouring Program) considering the data as a regular grid. The result is shown on Map 14. The effect of quantization noise depends on the track spacing (the grid size), the angle between the data lines and the direction of scanning.

Although the quantization noise can be reduced by generalisation of the surface, at the same time the estimation of the maximum error, i.e. the maximum distance between the contour lines on original and reconstructed map $\left(\frac{3\sqrt{2}}{4} h \right)$ is lost.

The next question which should be answered is that, given a set of points obtained by digitization, and given contouring programs that can operate on irregularly spaced data, what sort of contour map will these programs produce? Map 15 was produced by using GPCP on data obtained by raster scanning and treated as a set of irregularly spaced points. Comparison with the original map (Map 1) shows that the differences (in location, shape and number of the contour lines) are large.

To overcome the problem of how to display data in contour map form when the initial document was digitized by raster scanning a special algorithm was written. A formal description of this algorithm is included in Appendix C. The algorithm permits production of a contour map directly from the sequences of sample points along the tracks and guarantees that contour lines on the reconstructed map pass through the sample points. Map 16 shows the result of the program for the test data.

2.4 Storage of data when digitizing by raster scanning

Although matrix data (i.e. data taken over a square grid) are the most convenient form of representing a surface in terms both of computation - this is because numerical analysis of two-variable functions is based on that structure of data - and of programming - that is because commonly used computer languages allow the handling of two-subscripted arrays - the problem of how to store the spatial data is still open. If, for example, the cost of storage of a large matrix is higher than the cost of generating it, plus the cost of the storage of the original data, it would be unwise to store data in the matrix form.

In the case of spatial data obtained by raster scanning the best method of storage seems to be the raw data, i.e. the sequences of sample points (where the tracks cross the contour lines). First of all the data carry very accurate information about the original document whereas the matrix data are 'second hand' and it is impossible to extract from them information about exact positions of the contour lines. When it is desired to increase the accuracy of the data base by further scanning, creating a new denser grid, the initial data are necessary.

The matrix form of storage is convenient for displaying spatial data in contour map form because that structure of data is used by most of the contouring programs. Regarding the display of spatial data when the contour

map has been digitized by raster scanning, storage in matrix form has no advantage since it is possible to display the data in contour map form directly from the raw sample points (see Appendix C).

The memory space required for storing the data from raster scanning may be reduced in the following ways:.

- (a) it is not necessary to store the co-ordinates of the tracks; knowing the co-ordinate of the first track and the track spacing the co-ordinates of all the tracks can be determined,
- (b) the set of contour line values can be transformed into a set of integer numbers. Let us say that Z_0 is the lowest contour line on the map sheet to be digitized and ΔZ is the contour interval then $Z_1 = Z_0 + \Delta Z$, $Z_2 = Z_0 + 2\Delta Z, \dots$
 $Z_k = Z_0 + k\Delta Z, \dots$ are the values of contours on the map.

The transformation

$$T(Z_k) = \frac{Z_k - Z_0}{\Delta Z}$$

gives the set of the integer numbers $0, 1, \dots, K$ where Z_k is a maximum value of the contour lines on the map sheet. Now, instead of using the whole machine word to store the real value of the contour line, one can use a part of it to store an integer value. Let us say that the total number of the different values of the contour lines on the map sheet is not greater than 256 which is a reasonable assumption for a great many maps. Any integer in the range $0, 255$ can be represented with only 8 bits. Most computer systems permit operations on parts of the machine word (so-called bytes) consisting of 6 or 8 bits.

A similar principle of storage reduction is used by Pfaltz (1975) in bit plane representation of surfaces.

CHAPTER III

TECHNICAL IMPLEMENTATION OF DIGITIZING METHODS

3.1 Digitizing errors

There are two main problems causing errors during digitizing:

(a) the thickness of the contour lines as a source of errors

Up to now it has been assumed that the contour lines are dimensionless and hence that the raster scanning technique provides exactly one point from each crossing of a contour line by a track. Usually it can be assumed that the thickness of the tracks is negligible but very often this assumption cannot be made about the thickness of the contour lines; in this case a segment rather than a point is being sampled. The problem then arises as to which point from that segment should be chosen as a sample point. Although the middle one tends to be chosen it seems unlikely that the 'real' contour lines go exactly through the middle of the contour lines on the map. If it is assumed that the contour lines on the map cover the 'real' contours, taking the middle point of the segment will cause the smallest error. The amount of error caused by the thickness of the contour lines depends on local gradients and the thickness of the contour line itself. If, for example, the thickness of the 100 m contour line corresponds to a height difference of 1 m then the error is insignificant. On the other hand, it may be relatively big. On the test map (Map 1) the thickness of the 100 m contours corresponds in certain parts of the map to a 20 m height difference. Assuming that the 'real' contour lines are underneath the contours on the map and that one samples in the middle of the contour line, is still possible to have errors of plus or minus half the height difference of the area covered by the contour lines.

The thickness of contour lines may be a source of error not only when the raster scanning technique is applied, but also when the line following technique is used. The following problems, however, only arise

during raster scanning.

(b) the tangency problem

So far it has been assumed that the scanning tracks cross the contour lines. However, sometimes situations arise when the track and the contour line are collinear for a certain distance (as was mentioned previously, the thickness of the track is relatively small compared with the thickness of the contour line): for example, the situations shown on Figure 9b or when the track touches a contour line but does not cross it (Figure 9c). In fact, both cases may occur simultaneously as on Figure 9a.

The case when the track follows the contour line is referred to by Mor and Lamdon (1972) as the tangency problem. In their paper, they describe a two-level test to detect whether the track passes from one intercontour region to another (crossing type) or remains within the same intercontour region (tangent type). These tests may be used when the raster scanning technique is computer controlled. When a digitizing table is used the type can be deduced directly from the map. A general rule is that, with the crossing type an odd number of sample points has to be registered along the track, whereas with the tangent type there are an even number of sample points.

(c) mismatching the tracks

Before digitizing by the raster scanning technique, the spacing of the tracks must be decided. Usually equal spacing of the tracks is assumed. However, the physical location of the tracks may be different from that planned and thus sample points may be incorrectly located (Figure 10a). The amount of deviation depends firstly on the error in location of the track, and secondly, on the angle between the contour line and the track (Figure 10b). Usually errors caused by mismatching the tracks are not large. Hardware accuracy enables fitting of the tracks to an accuracy of a few hundredths of a millimetre.

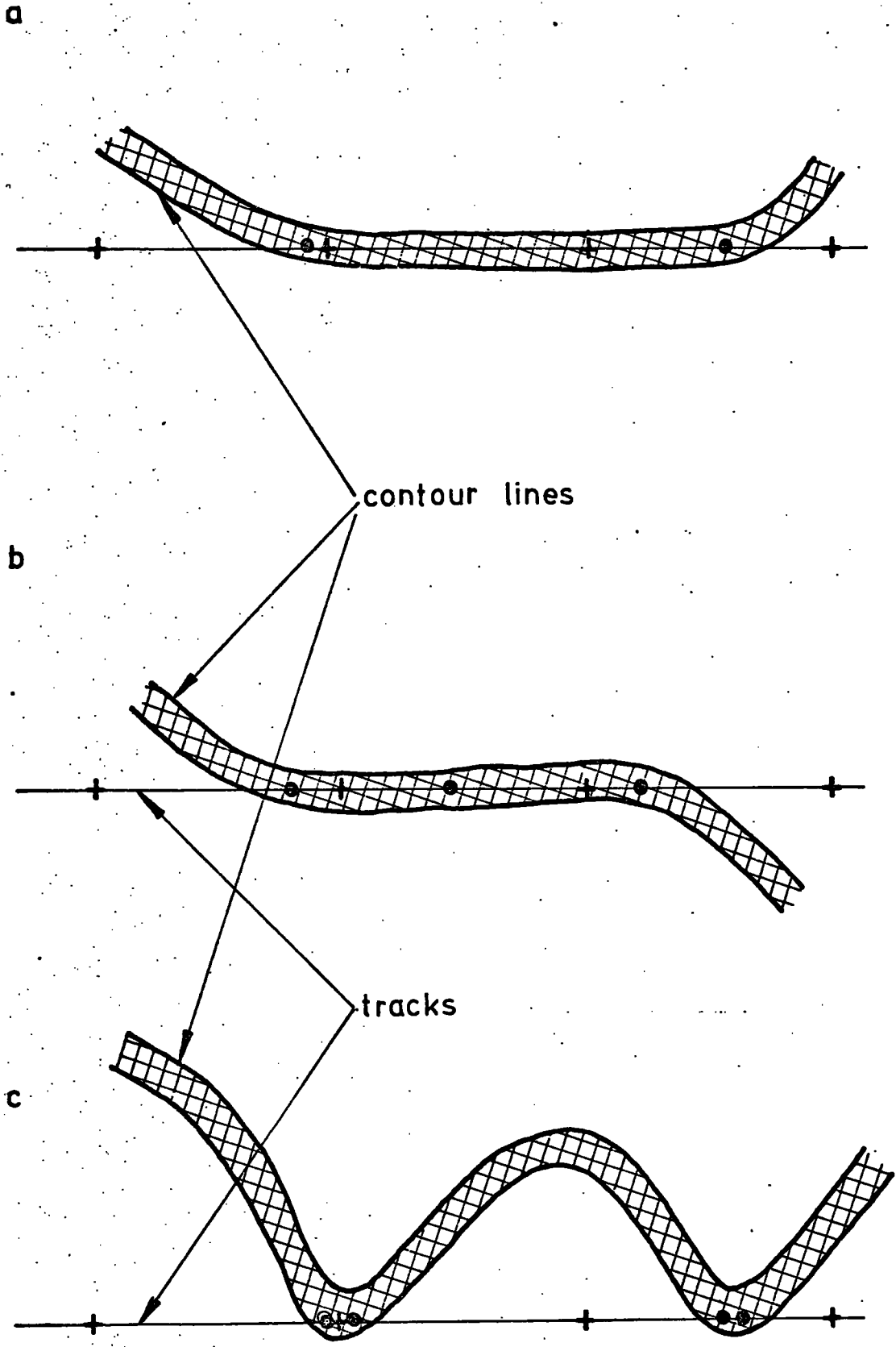


Figure 9. The tangency problem

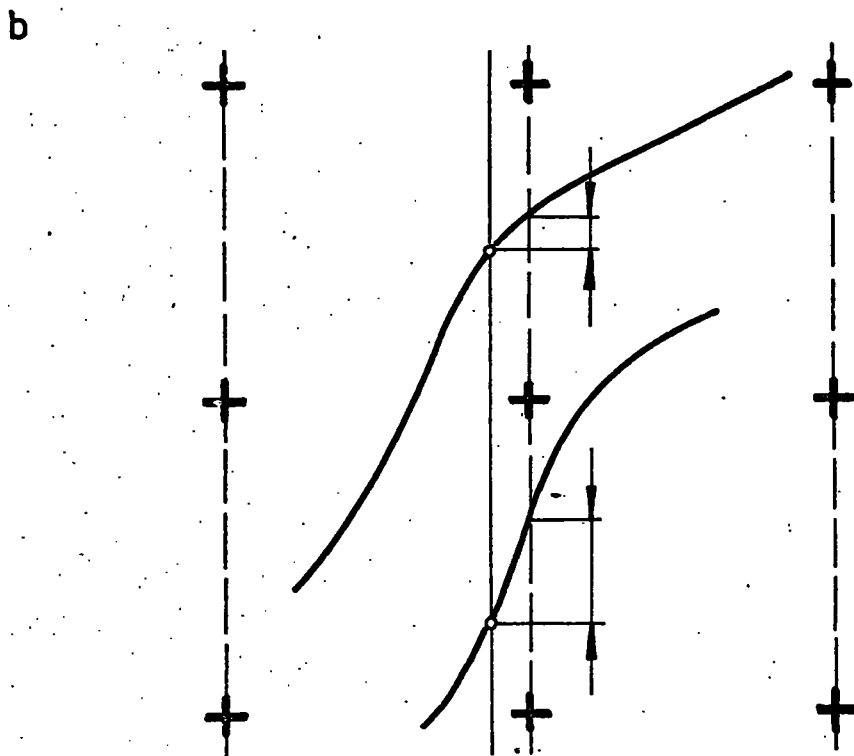
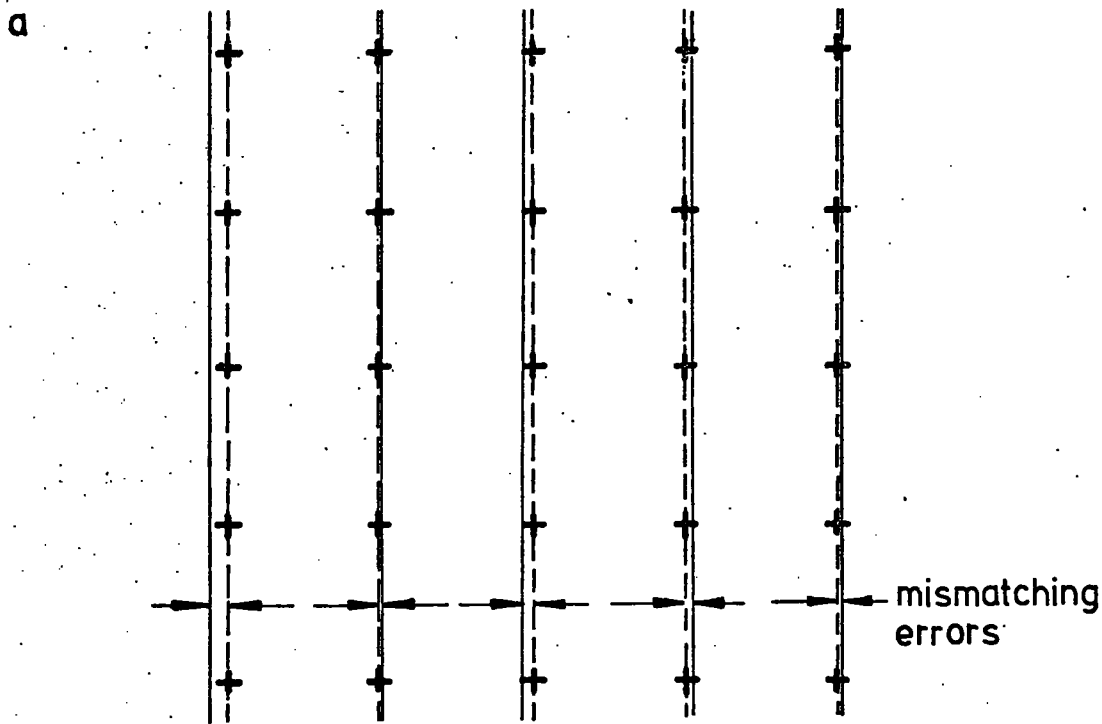


Figure 10. Mismatching the tracks

This kind of error may also arise when the contour map is scanned again, i.e. when new tracks are located to extend the digitization and the initial alignment of the scanner differs from that in the second scanning.

3.2 Hardware for digitizing

Although the term 'automatic digitizing system' is commonly used to describe the various systems used for digitizing of contour maps, in fact there is no system which could be used entirely automatically. The problem is that, during digitizing, it is necessary not only to locate certain lines, i.e. determine their co-ordinates, but also to determine the values attached to these lines. This last step cannot be performed without a human operator. First, not all contour lines on the map have an explicit value, as usually only selected contour lines (every fourth, fifth or tenth) are labelled. Secondly, even if the line has a printed value, the instrumentation to read it is very complex and so far has not been entirely successful in cartographic implementation. Lastly, even the location of lines sometimes requires an operator decision since there may be dashed lines on the map sheet.

The hardware for digitizing can be broken down into two main groups:

(a) table digitizers which enable automatic read out of the co-ordinates of points on the map to be digitized. Usually the sequence of points is registered on magnetic or paper tape. They may be classified as follows:

1. pencil follower digitizers
2. solid state tables
3. gantry tables

(b) automatic systems which enable automatic location of lines on the document to be digitized. Very often the document



Figure 11. Calspan's SEASCAPE digitizing facility (from Ryan, 1976)

has been photocopied onto transparent film so that lines may be detected by variations in the transmitted light. Depending on the technique involved, automatic systems may be classified thus:

1. scanners - flying spot scanners (Figure 11)

(Ryan, 1976; Davey et al, 1974)

drum scanners (McDonough, 1976)

2. automatic line followers (Laserscan, 1977)

Kelk (1974) gave a description of different types of table digitizers. An excellent review of all automatic systems available in Europe and North America is given by Davey et al (1977).

The problem of attaching the values to the lines might be overcome by using coloured lines. Mor and Lamdon (1972) described the logic of such a method based on two-coloured lines, where one colour is used for even-valued contour lines and the other for odd-valued ones. Knowing the value of the first line on the track and the initial trend, the values of all the lines on the track can be determined by noting their colours. The disadvantage of this method is that the map must be preprocessed by hand before digitizing. However, according to Davey et al (1977) there is no system for cartographic digitizing making use of coloured lines.

Among table digitizers, the most convenient type for raster scanning seems to be the gantry table, i.e. with the gantry above the table top and the cursor attached to it, especially when the position of the gantry (i.e. location of the tracks) may be automatically changed.

CONCLUSIONS

The aim of this thesis was to determine and compare the efficiency of techniques for the digital encoding of contour maps. Various aspects were considered: problems in the creation of matrix data, e.g. the most convenient structure for computations, the facility and accuracy of reconstructing digitized data in contour map form, storage requirements and error estimations. In particular, the raster scanning technique was stressed: this technique can be a good alternative to manual methods which are very time consuming and always subjective. Raster scanning can also be utilised without any special hardware.

The quantization method seems to be very useful for producing matrix data. Compared with linear and spline interpolation it minimizes the maximum error and permits the estimation of values at all nodes.

The specially written algorithm enables the reconstruction of digitized data obtained by raster scanning into contour map form, giving a set of control points at which the contour lines on the reconstructed map and the initial document are the same.

Although in the creation of quantized matrix data it is sufficient to scan a contour map once it seems worthwhile to stress that two-fold scanning in orthogonal directions creates new possibilities:

1. having the set of sample points on the tracks from two perpendicular directions one can estimate all parameters at the nodes with the same accuracy as is obtained when estimating directly from a map;
2. because the values encoded by quantization methods do not depend on the direction of scanning, the two data matrices obtained by orthogonal scanning may be compared with one another for a check on possible mistakes, be they on the map or due to the operator;

3. a comparison of the distributions of density and spacing of contours crossing the tracks in both directions may improve understanding of surface properties.

Although this thesis concentrates on digitizing of contour maps, there is an alternative way to obtain a digital terrain model, namely the orthophotomap method. The state of ^{the} art in orthophot^omap techniques is presented by Petrie (1977). However, because many map features, such as public land surveys and political boundaries, do not appear in stereophotographs, in addition to that technique, thousands of existing maps must be digitized either manually or semi-automatically (Southard, 1978). For example, in Sweden the high density terrain elevation data base is created from three different sources (Ottoson, 1978): photogram^metric measurements of aerial photos; digitizing of existing contour lines; and digitizing of existing profile plates. Also in West Germany data for cartographic data bases are partly captured by digitizing of contour maps (Uhrig, 1978).

The process of digitizing of contour maps cannot be performed entirely automatically. Although operator influence during semi-automatic digitizing is significantly less than in manual digitizing, one should remember that a contour map is a product of human subjectivity and fallibility.

REFERENCES

- A.C.S.M. (1976) Proceedings of the International Conference on Automation in Cartography, 1974, Am. Congr. Surv. Mapp.
- Amidon, E.L. and Akin, G.S. (1971) Algorithmic selection of the best method for compressing map data strings. Communications of the ACM, 14, 769-774.
- Boehm, B.W. (1967) Tabular representation of multivariate functions - with applications to topographic modeling. Proceedings ACM National Meeting, 22, 403-415
- Boyle, A.R. (1976) Automatic line digitization. Paper presented to XIII Congress of the International Society of Photogrammetry, Helsinki, 1976.
- Dale, P.F. (ed) (1974) Automated Cartography. British Cartographic Society, Spec. Publ. 1.
- Davey, P.G. Harris, J.F. and Loken, J.G. (1974) PEPR techniques for reading microfilmed rain charts. Proceedings Oxford Conference on Computer Scanning.
- Davey, P.G., Harris, J.F. and Preston, G. (1977) Automatic boundary digitizing for the Department of the Environment. Report of Nuclear Physics laboratory of Oxford University.
- Davies, O.L. (ed) (1961) Statistical Methods in Research and Production, Oliver and Boyd, Edinburgh.
- Evans, I.S. (1972) General geomorphometry, derivatives of altitude and descriptive statistics; in Chorley, R.J. (ed) Spatial analysis in geomorphology, Methuen, London, 17-90
- Evans, I.S. and Young, M. (1977) Terrain analysis - program to derive the descriptive characteristics of gradient, aspect and profile and plan convexity.
- Freeman, H. (1974) Computer processing of line-drawing images. Computer Surveys, 6, 57-97.
- Hunting Surveys Ltd. (1977) A digital mapping system designed for the commercial air survey market. Paper presented by Kier, K.M. to XIII Congress of the International Society for Photogrammetry, Helsinki, 1976.
- Kelk, B. (1974) Hardware for automated cartography, in Dale, op. cit. 25-43.
- Laserscan Ltd. (1977) the HRD1 Laser Display, Plotter, the Automatic Line Follower, Cambridge.
- MacDougall, E.B. (1976) Computer programming for spatial problems, Arnold, London, 106-115.

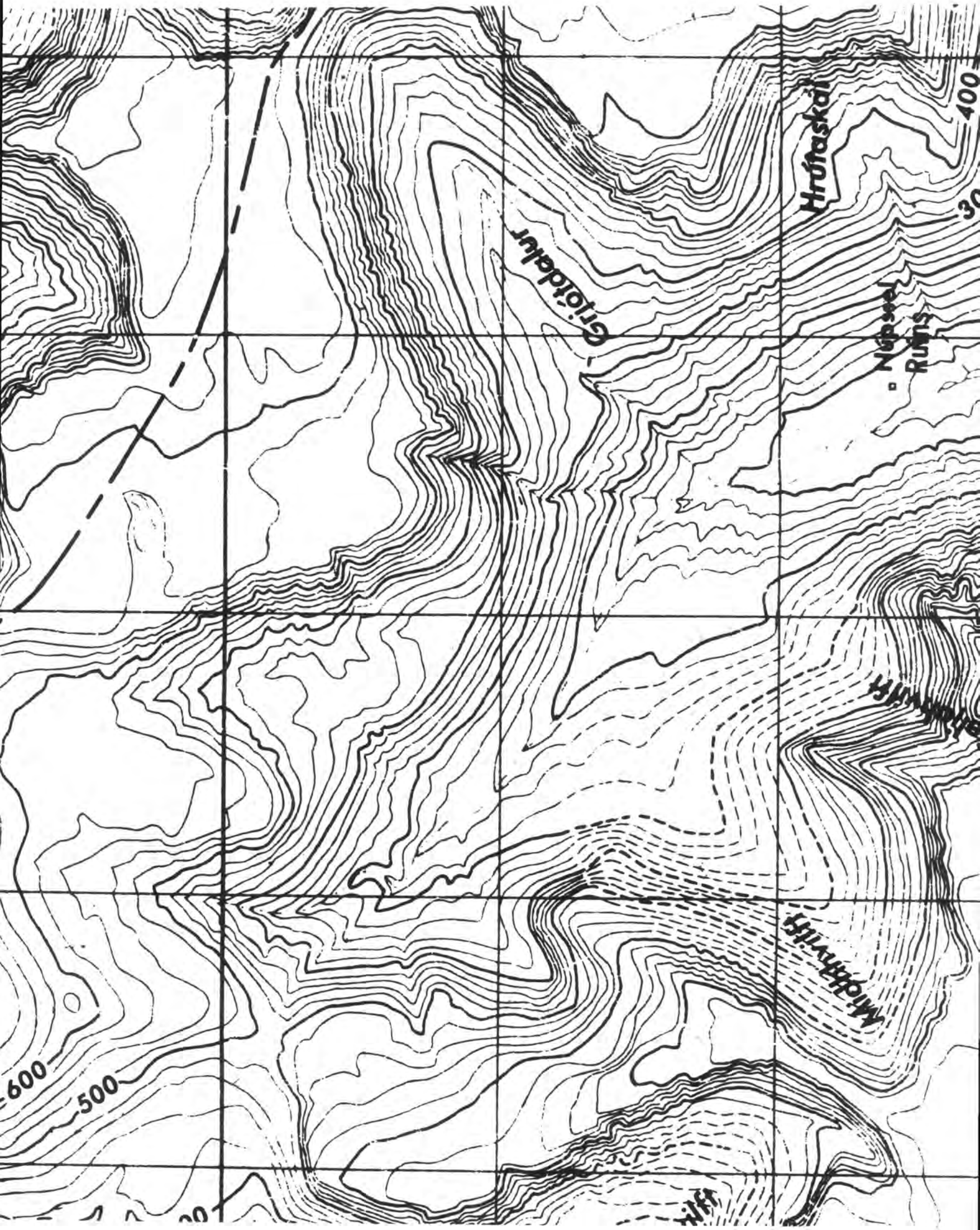
- Mark, D.M. (1975) Computer analysis of topography: a comparison of terrain storage methods. Geografiska Annaler, 57A, 179-188.
- McDonough, J.O. (1976) Input Methods Panel, in A.C.S.M., op. cit., 97-113
- Mor, M. and Lamdon, T. (1972) A new approach to automatic scanning of contour maps. Communications of the ACM, 15, 809-12.
- Morse, S.P. (1968) A mathematical model for the analysis of contour-line data. Journal of the ACM, 15, 205-220.
- Ottoson, L. (1978) Establishment of a high density terrain elevation data base in Sweden. Paper presented to the IX International Conference on Cartography, ICA, Maryland, 1978.
- Park, G., Lee, Y.H., and Scheps, B.B. (1971) Slope measurement from contour maps. Photogrammetric Engineering, 37(3), 277-283.
- Petrie, G. (1977) Orthophotomaps. Transactions of Institute of British Geographers, 2, 49-70.
- Pfaltz, J.L. (1975) Representation of geographic surfaces within computers, in Davis, J.C. and McCullagh, M.J. (eds.) Display and analysis of spatial data, John Wiley, London, 210-230.
- Rhind, D.W. (1974) An introduction to the digitizing and editing of mapped data, in Dale, op. cit., 50-68.
- Rosenfeld, A. and Kak, A.C. (1976) Digital picture processing, Academic Press, London, 1-8, 64-107.
- Ryan, H.F. (1976) Input Methods Panel, in A.C.S.M., op. cit., 97-113.
- Schmidt, A.H. (1976) Cartographic Data Structures Panel, in A.C.S.M., op. cit., 165-182.
- Southard, R.B. (1978) Development of a digital cartographic capability in the National Mapping Program. Paper presented to the IX International Conference on Cartography, ICA, Maryland, 1978.
- Türke, K. (1978) Digitizing of geometric data for thematic mapping: state-of-art and future developments. Nachrichten aus dem Karten-und Vermessungswesen, 35, 85-92
- Uhrig, H. (1978) Data capture for cartographic data bases, Nachrichten aus dem Karten-und Vermessungswesen, 35, 107 -113.
- Wesley, P.W. (1974) Basic concepts in automation in cartography, in Dale, op. cit., 1-14.
- Yoeli, P. (1975) Compilation of data for computer assisted relief cartography, in Davis, J.C. and McCullagh, M.J. (eds.) Display and analysis of spatial data, John Wiley, London, 352-367

Description of maps

Contour lines on the maps 4 -12, 14 were drawn using Calcomp's GPCP (General Purpose Contouring Program) from matrix data and copied on the test map (Map 1).

- Map 1. Test map - Nupur area (Iceland) enlarged from 1:50 000 scale, contour interval 20 m.
- Map 2. A map obtained by using GPCP from data digitized by hand at 200 x 200 m. grid.
- Map 3. A map obtained by using GPCP from quantized matrix data.
- Map 4. A polynomial of 4th order computed on data digitized by hand in 200 x 200 m grid.
- Map 5. A polynomial of 4th order computed on quantized matrix data.
- Map 6. A map of the surface smoothed by a quadratic (input data were the same as for Map 2.)
- Map 7. A map of the surface smoothed by a quadratic (input data were the same as for Map 3).
- Map 8. A map of the surface smoothed by a cubic (input data were the same as for Map 2).
- Map 9. A map of the surface smoothed by a cubic (input data were the same as for Map 3).
- Map 10. A map of the surface smoothed by the filter
1 2 1
2 4 2 (data as in Map 3)
1 2 1
- Map 11. A map of the surface smoothed by the filter
0 1 0
1 4 1 (data as in Map 3)
0 1 0
- Map 12. A map of the surface smoothed by the filter
0 0 1 0 0
0 1 2 1 0
1 2 8 2 1 (data as in Map 3)
0 1 2 1 0
0 0 1 0 0
- Map 13. Test data to visualize quantization noise.

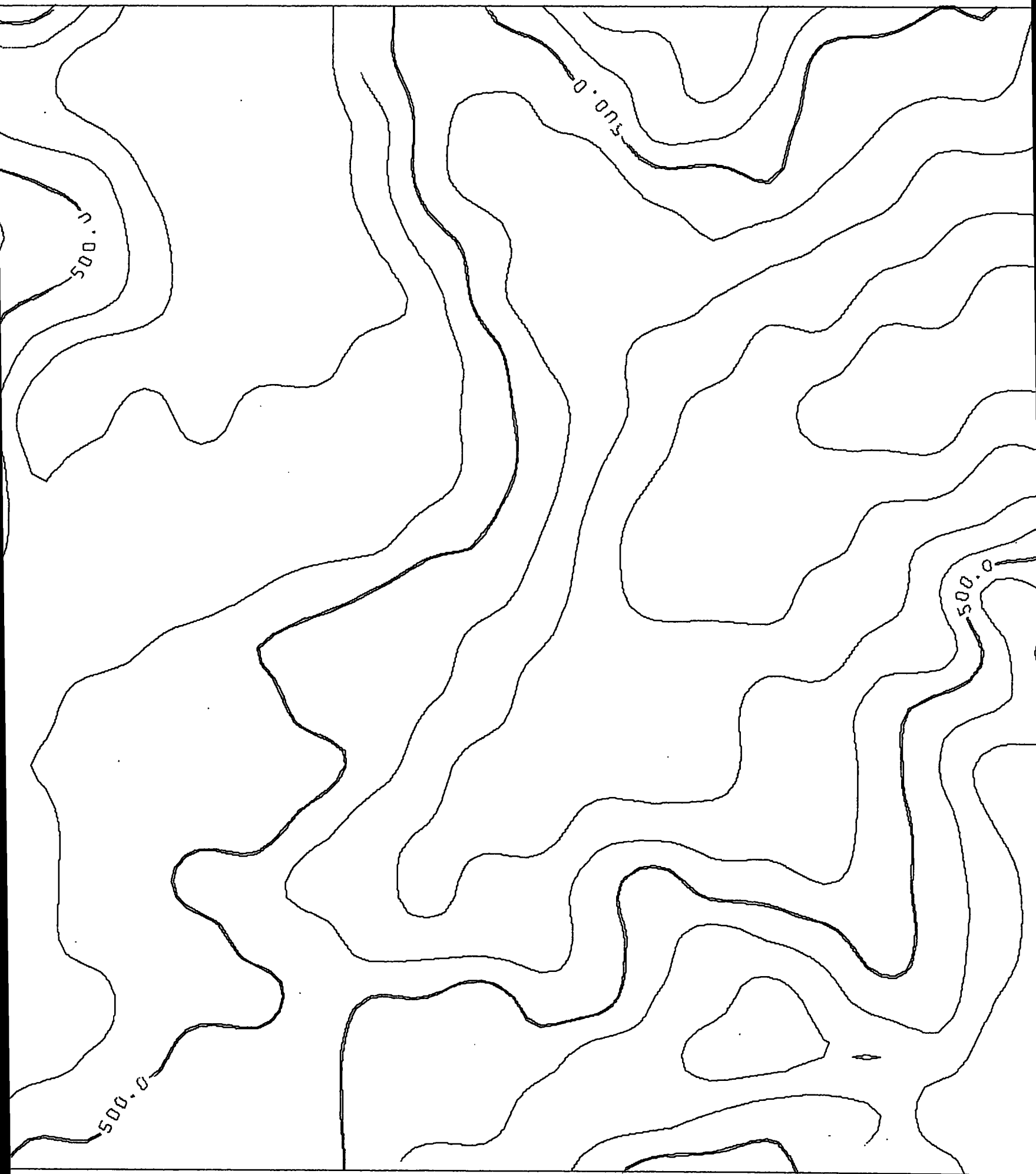
- Map 14. Quantization noise: the test contour map shown as Map 13 was scanned and quantized.
- Map 15. A map plotted by GPCP from irregularly spaced data (sample points)
- Map 16. A map drawn using the algorithm to display data obtained by raster scanning in contour map form (Appendix C).



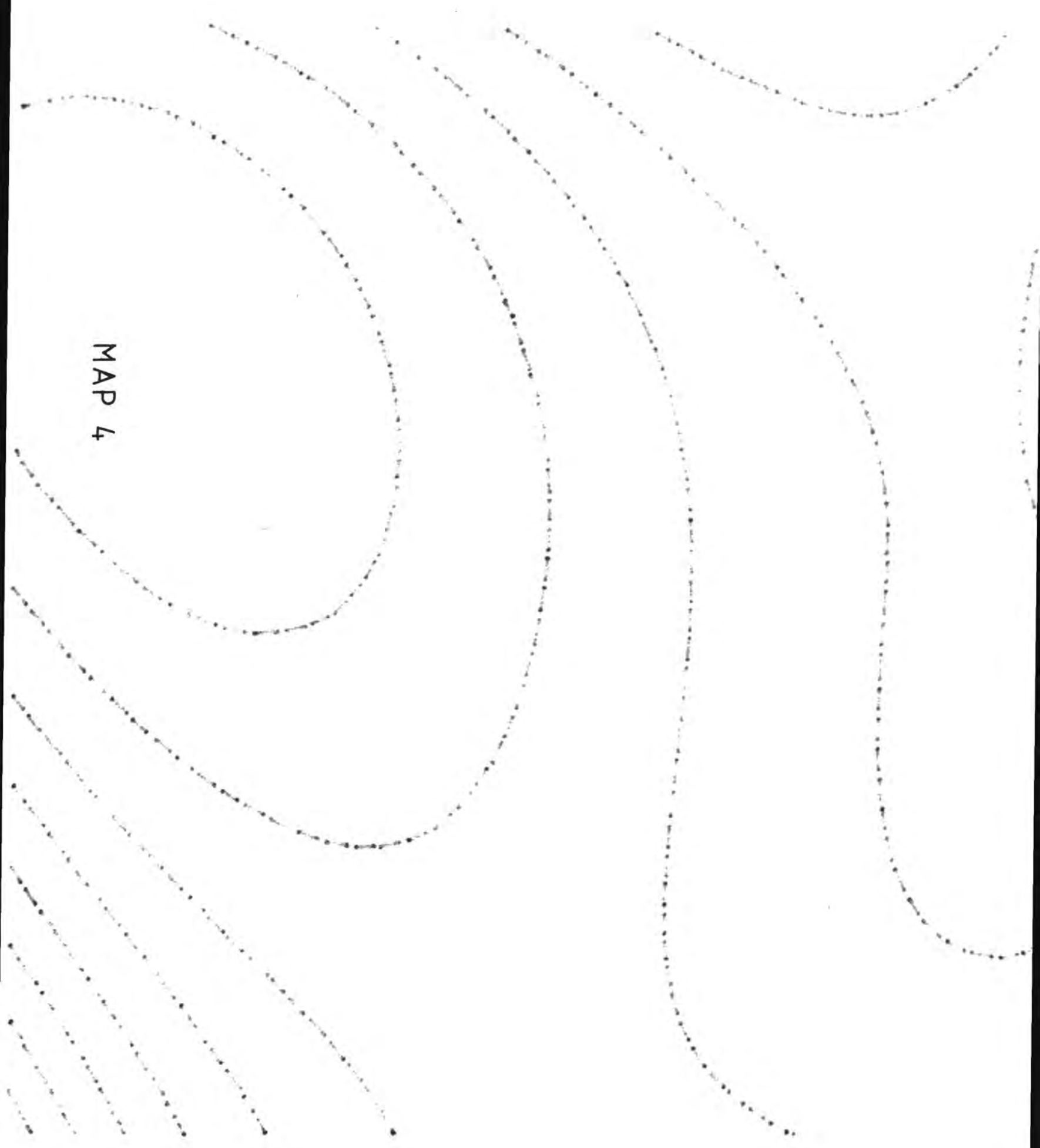
MAP 2

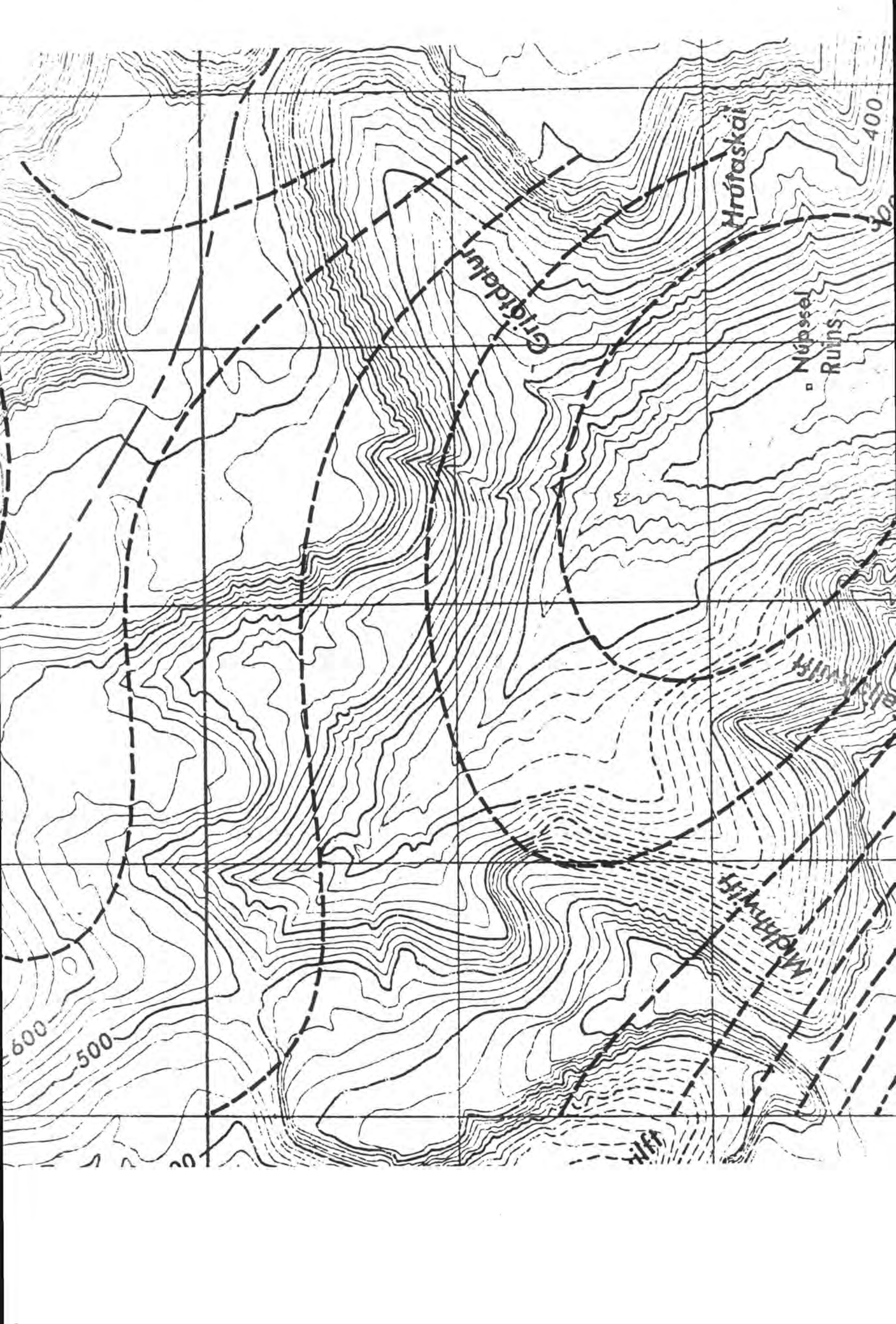


MAP 3



MAP 4





Hrófaskál

Griðdalur

Núpsel
Ruins

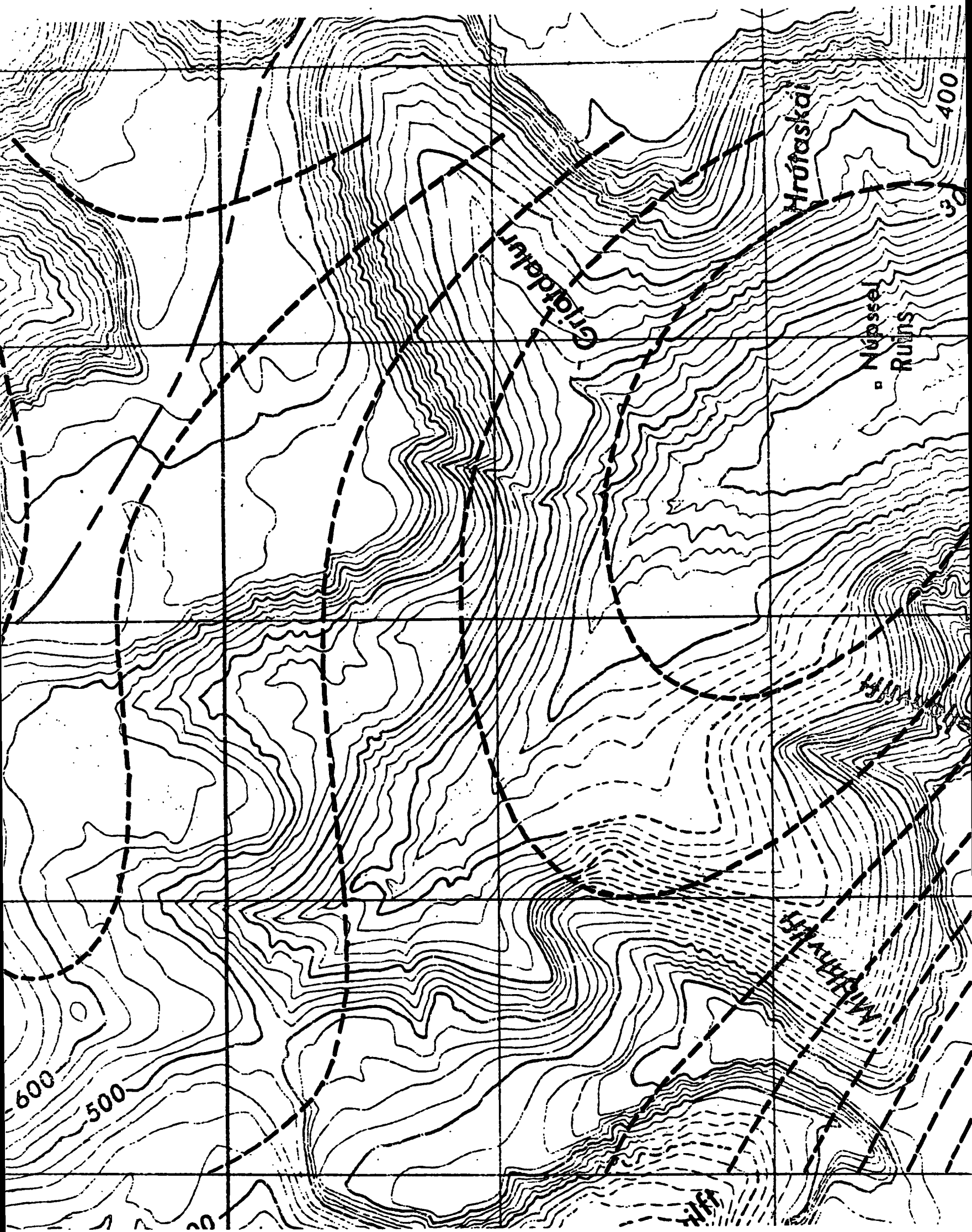
Málmur

Hlífi

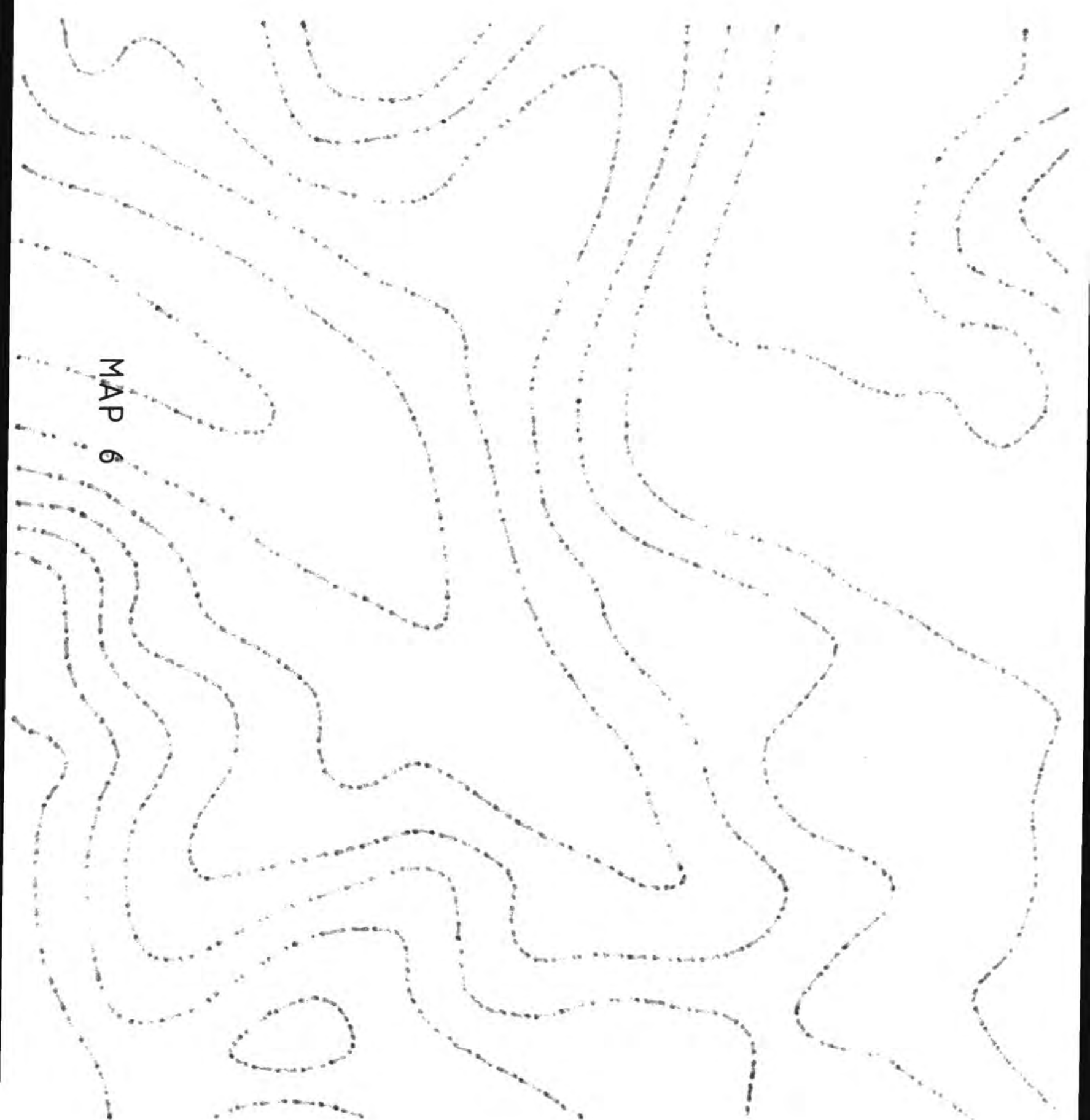
600
500

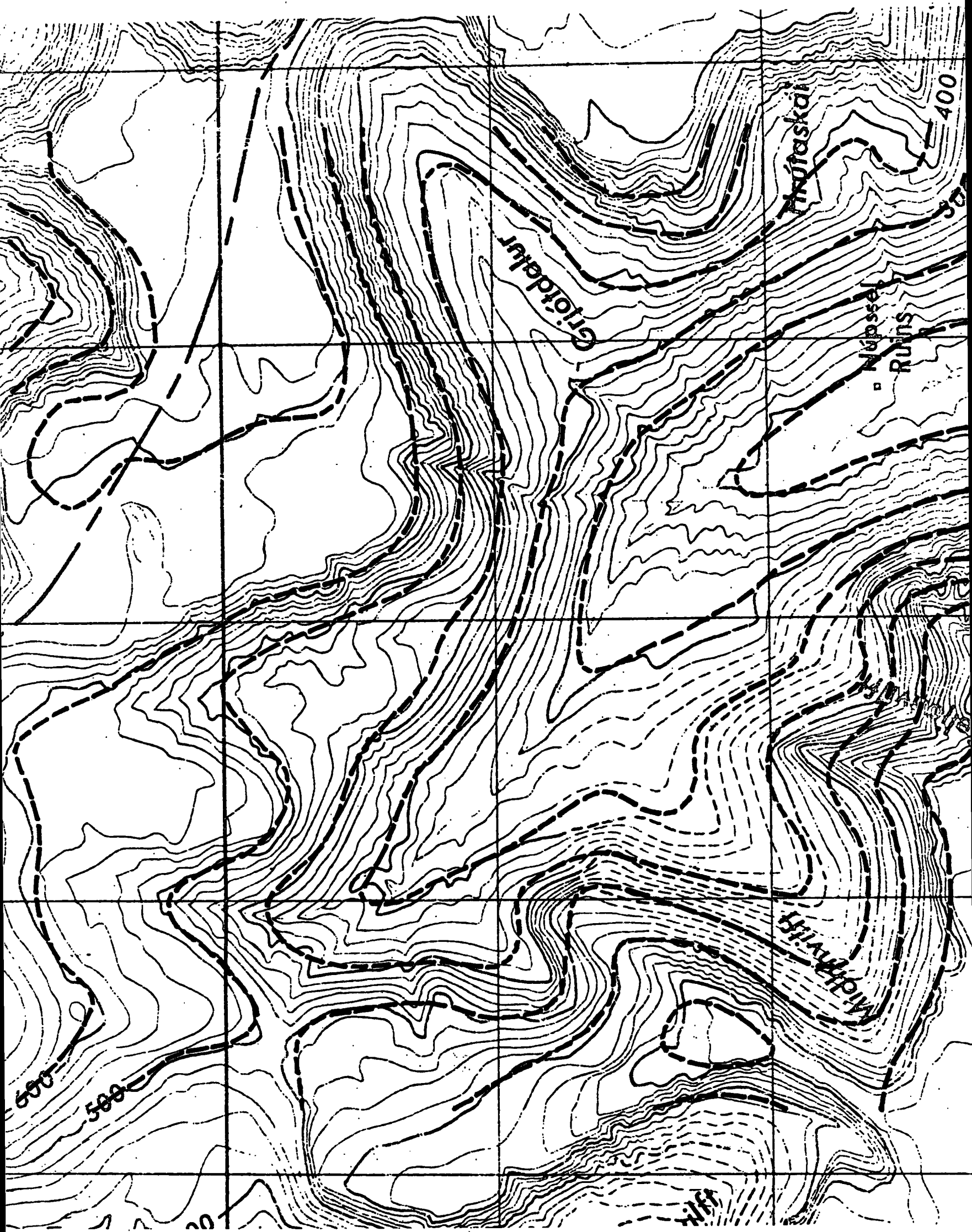
400

MAP 5

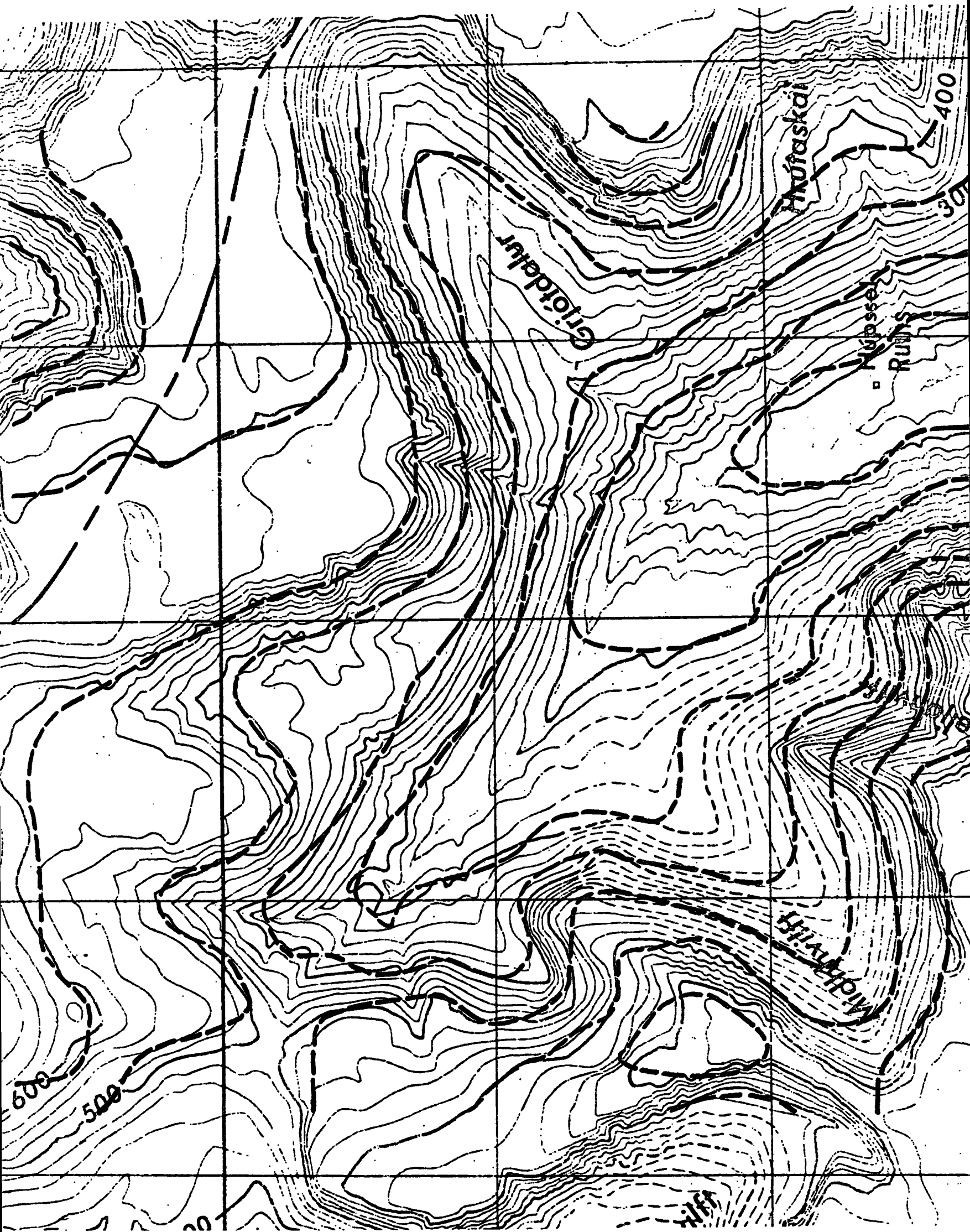


MAP 6

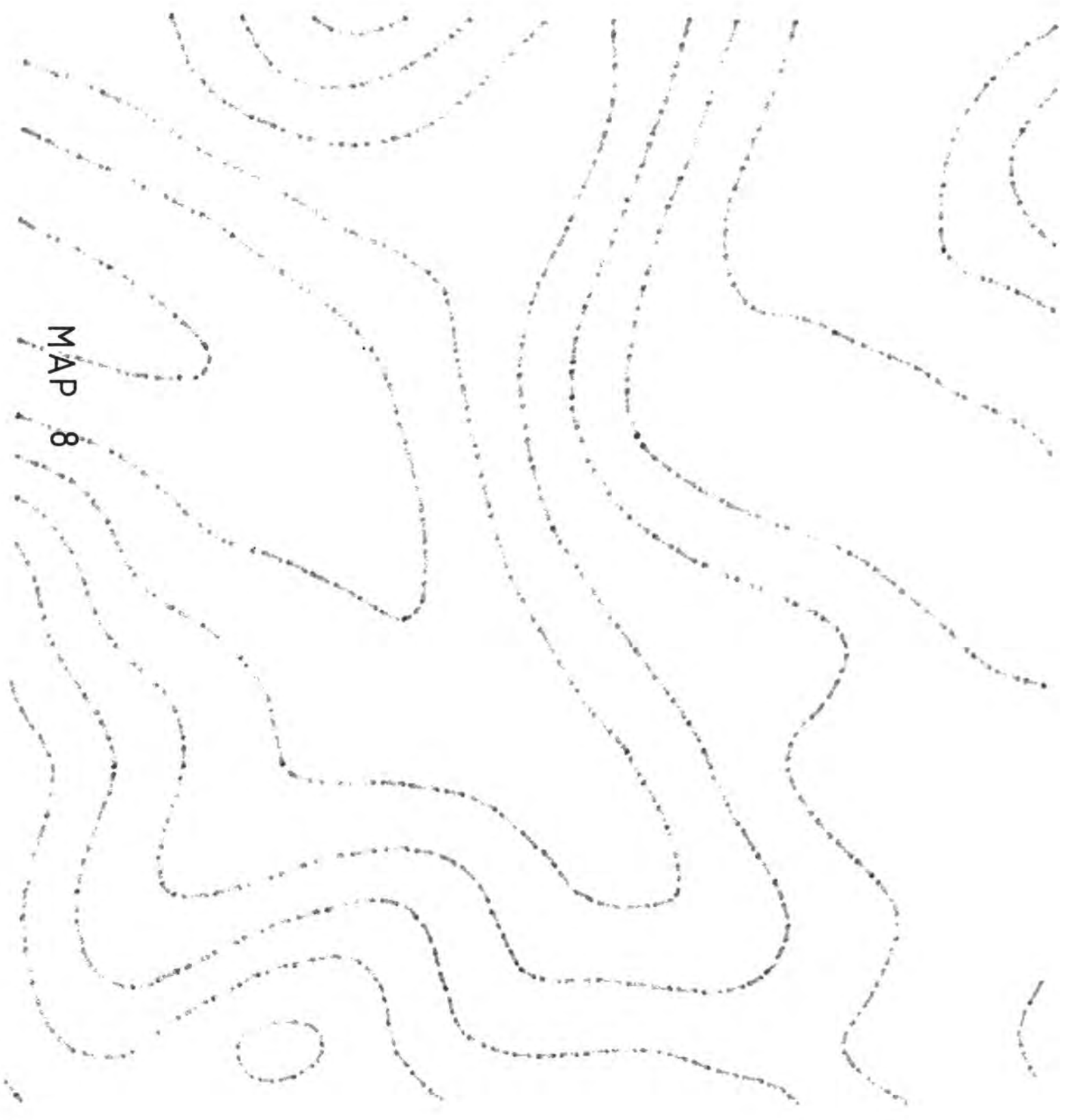


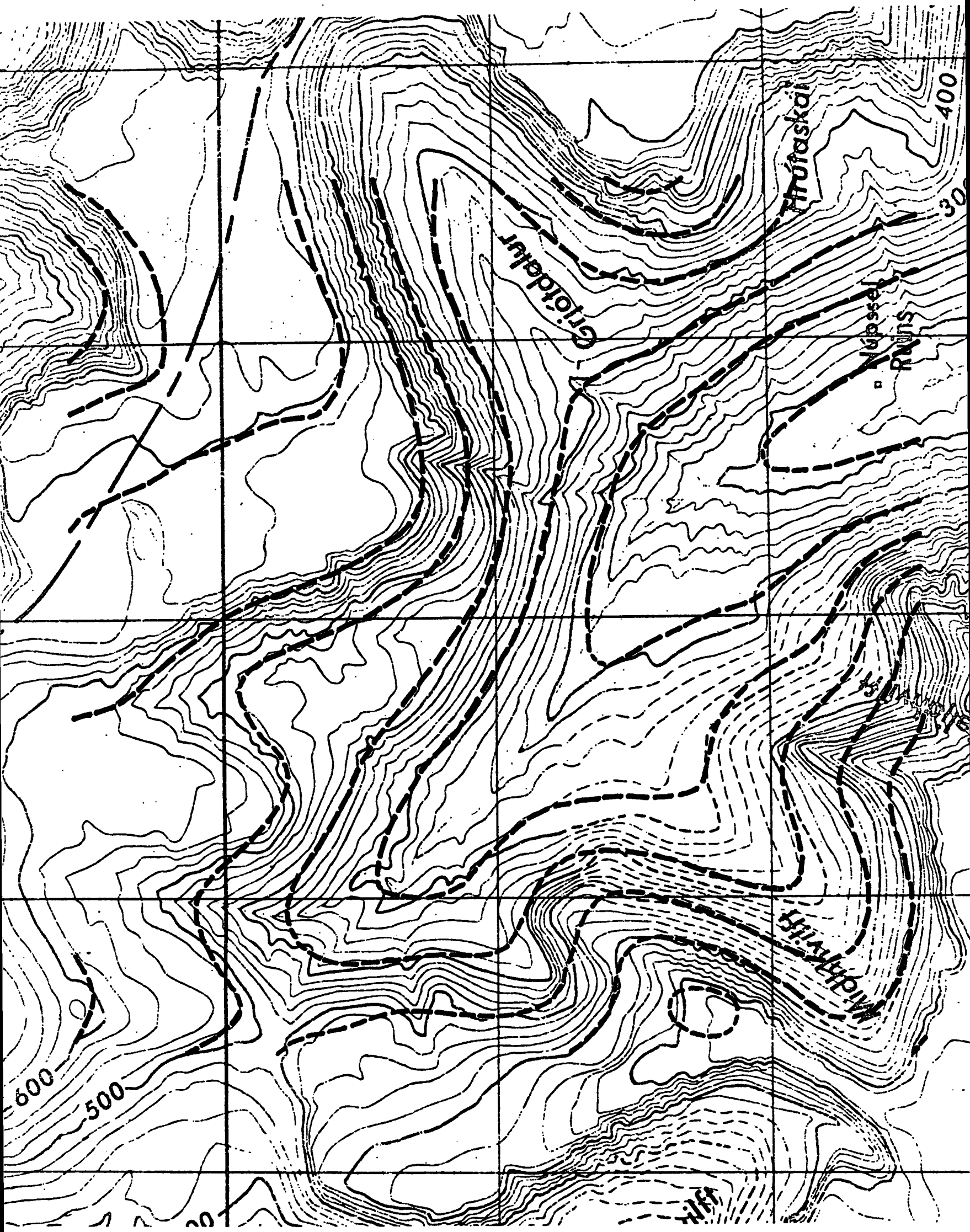


MAP 7

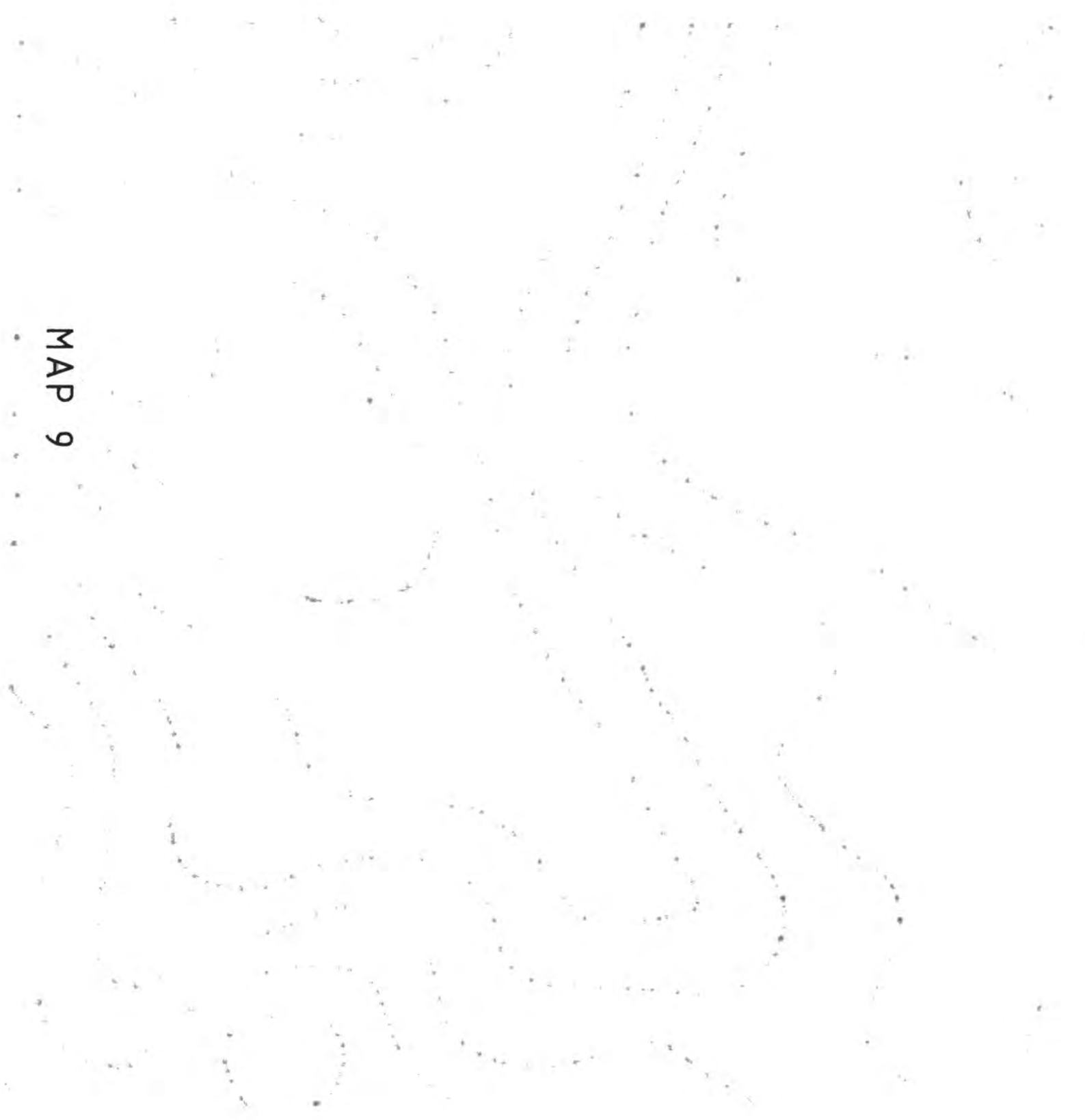


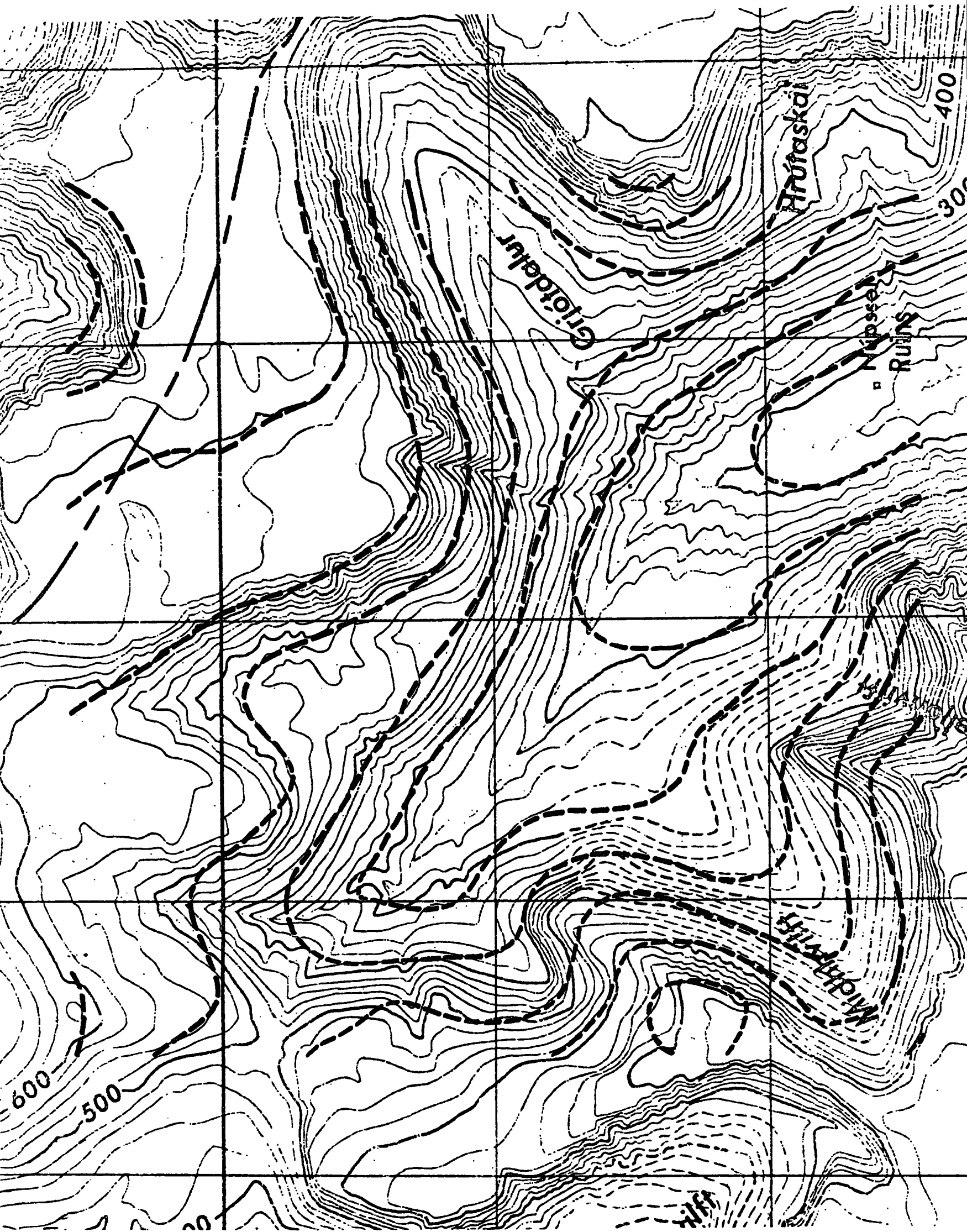
MAP 8



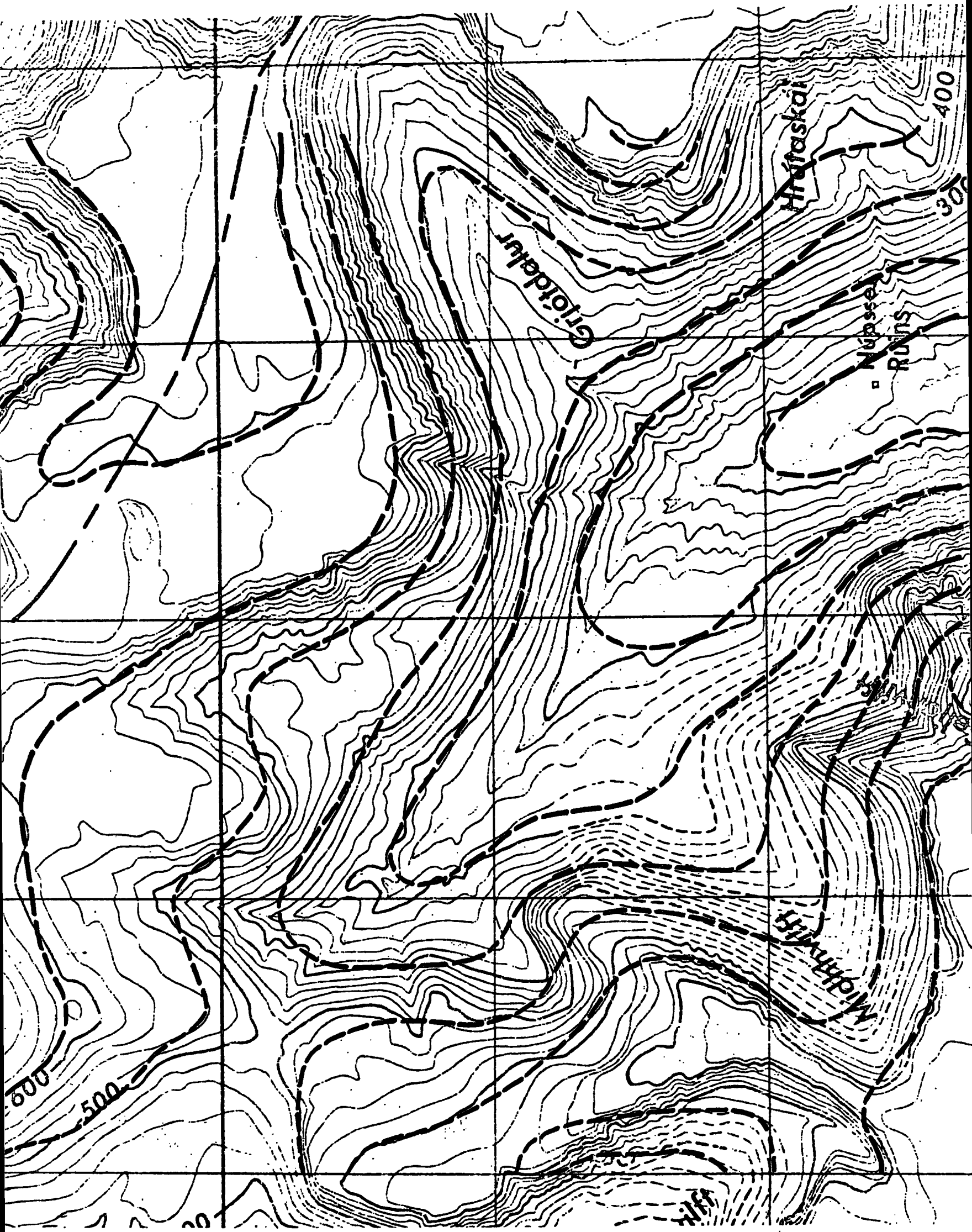


MAP 9

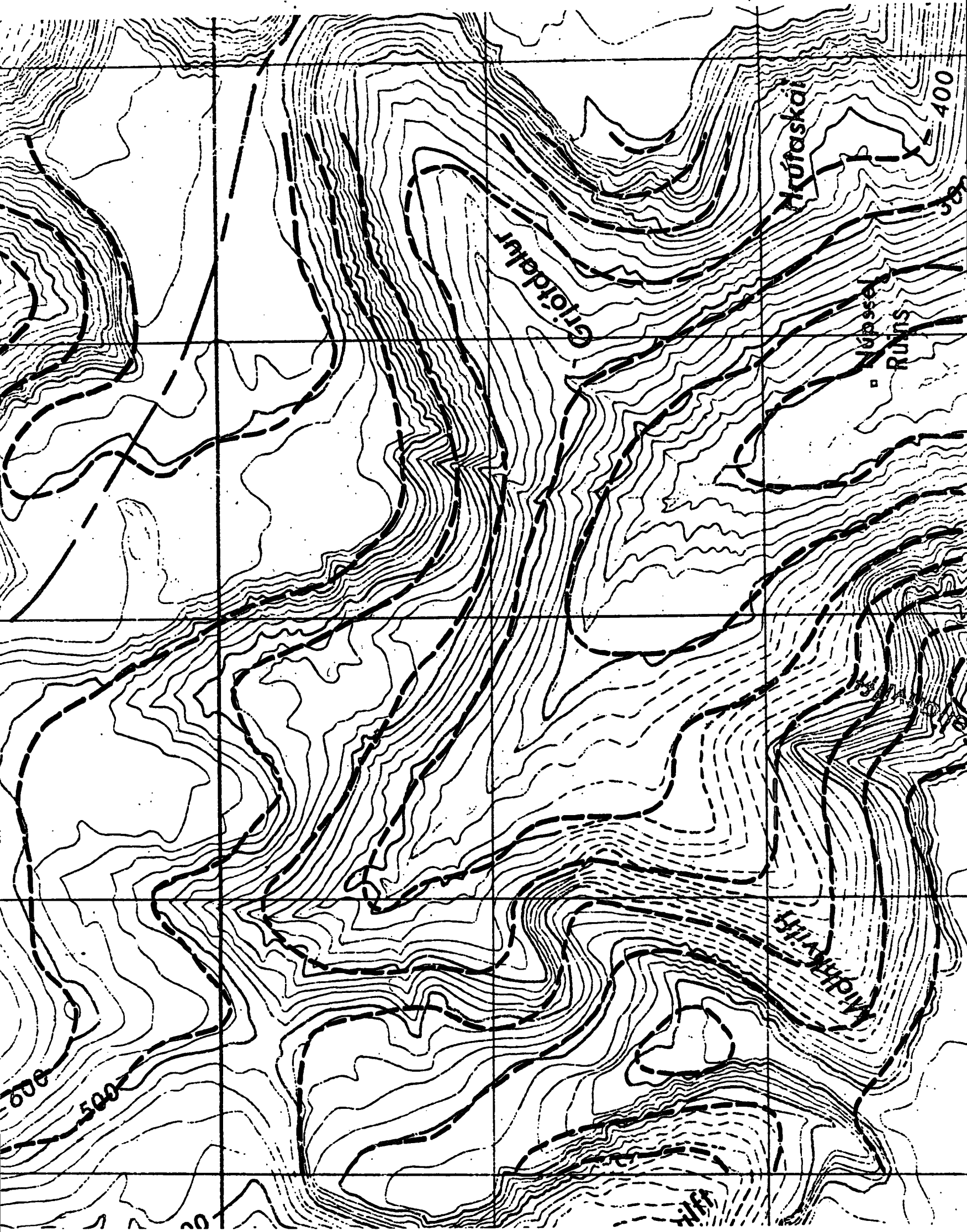




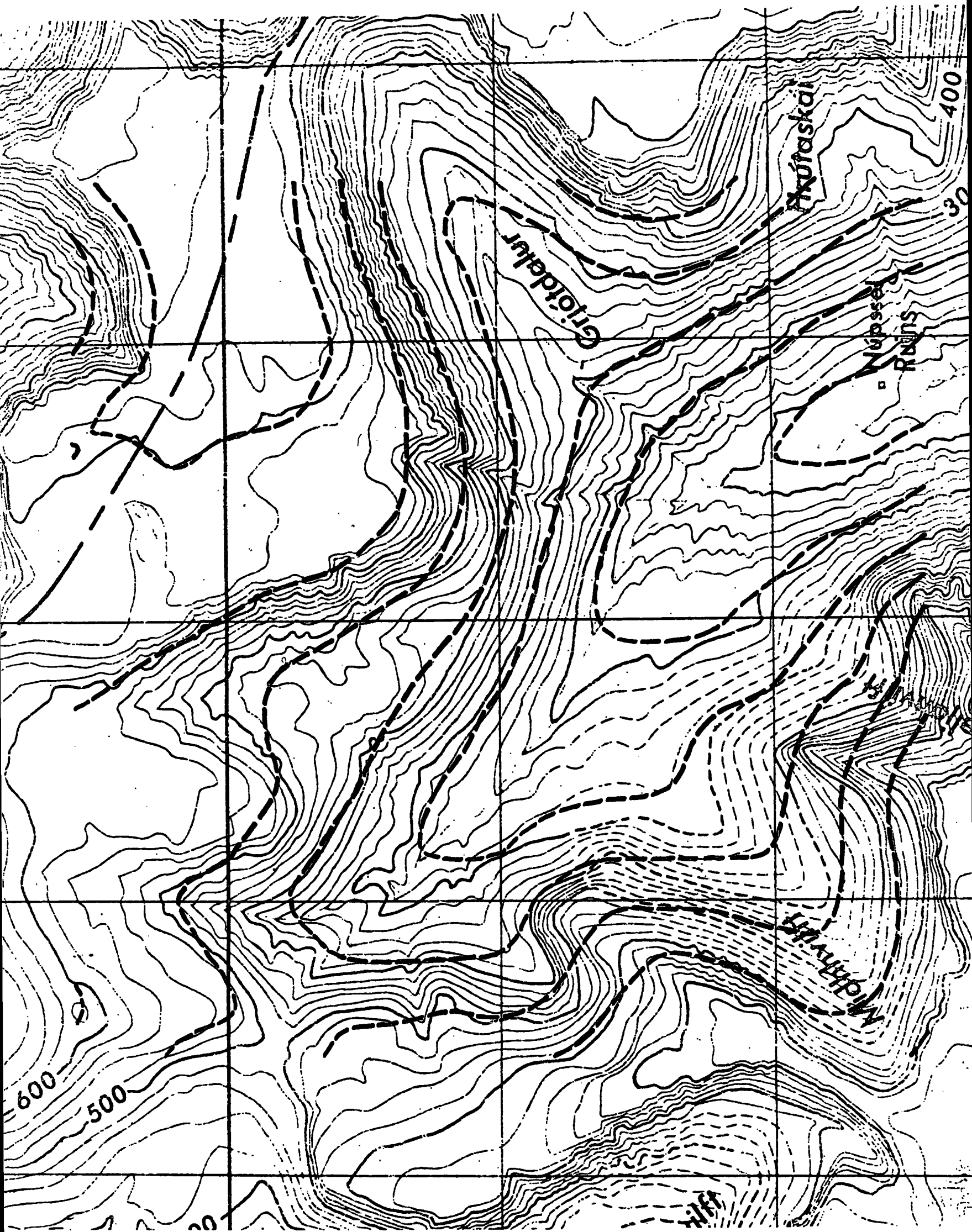
MAP 10



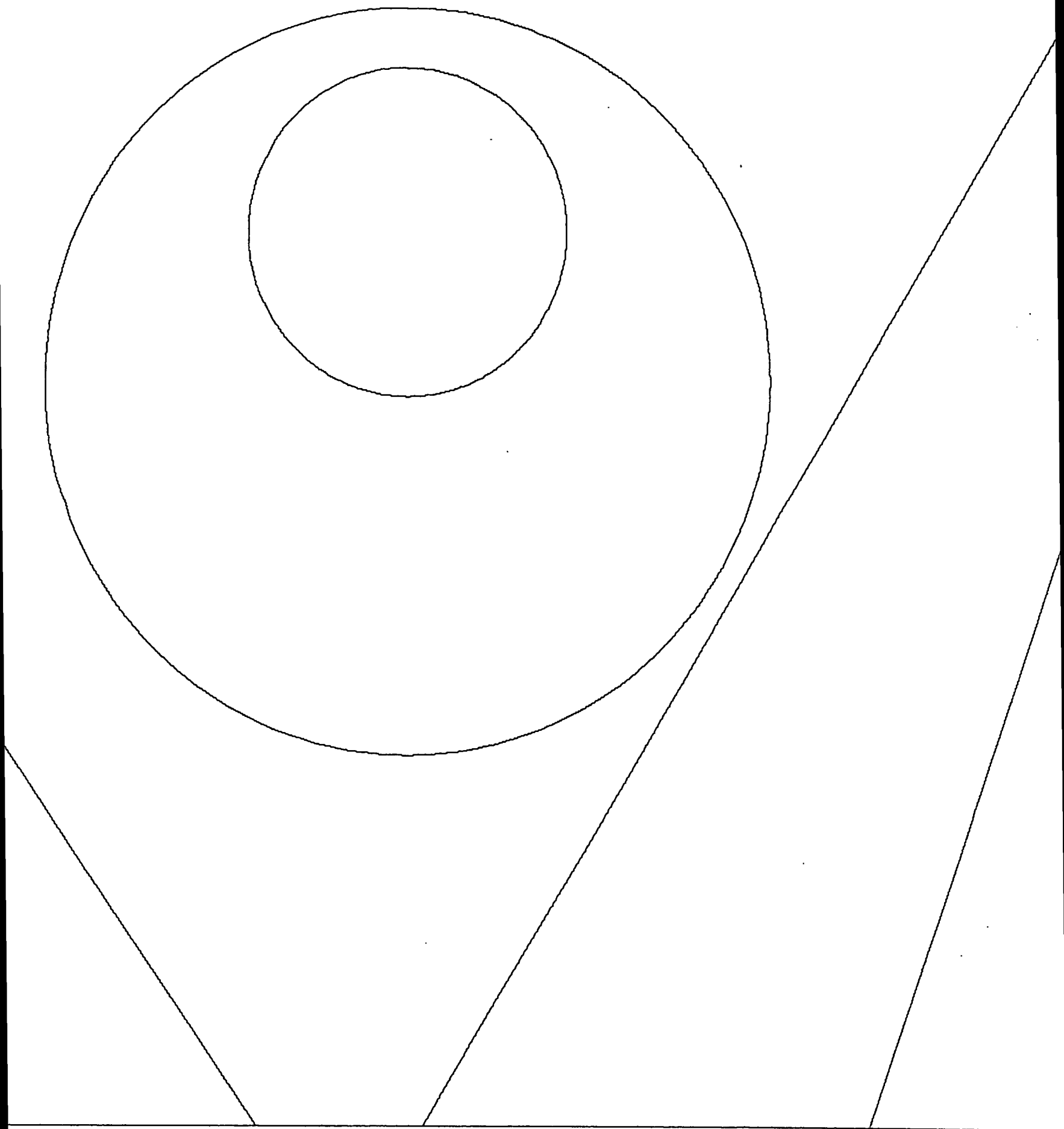
MAP 11



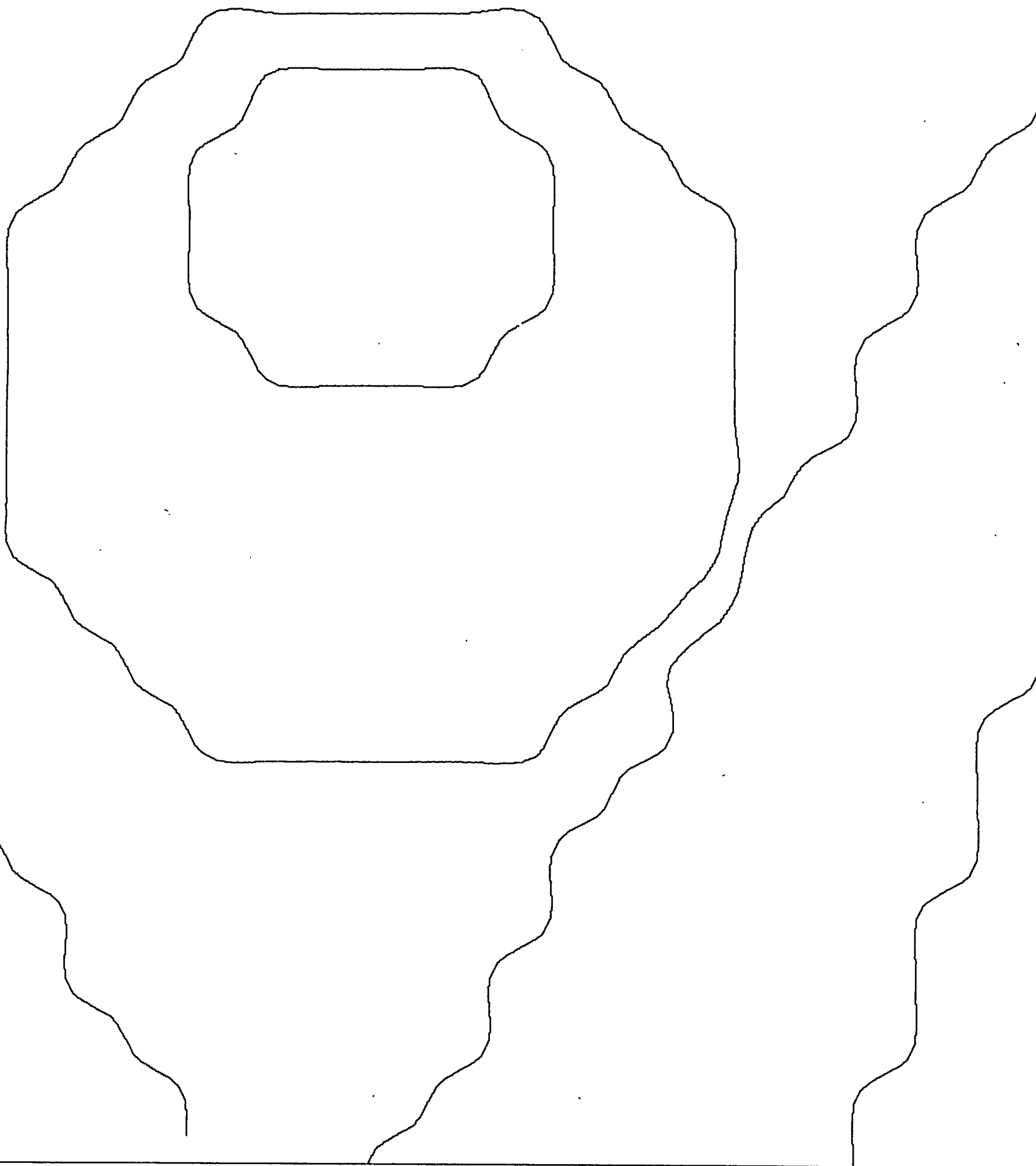
MAP 12



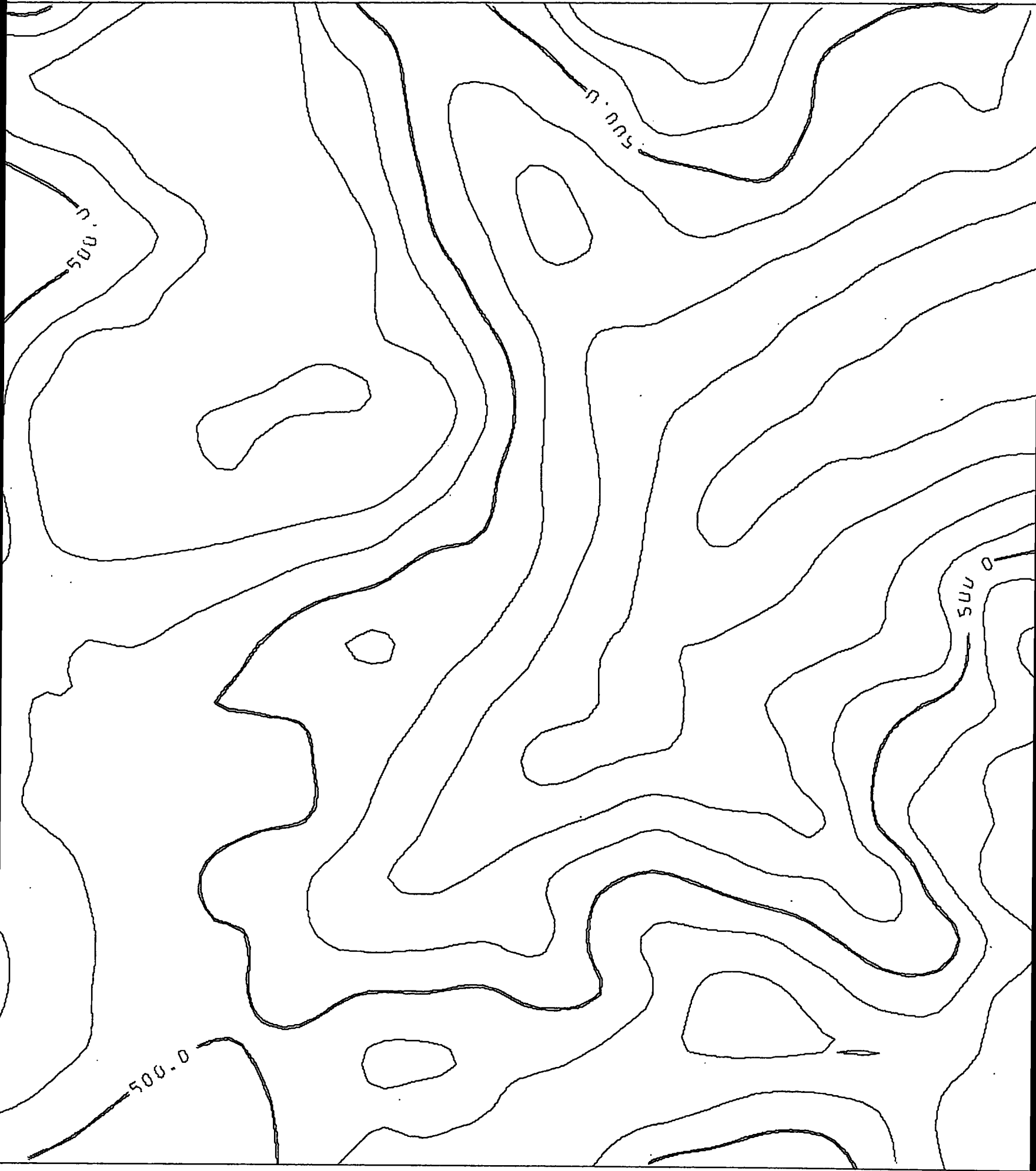
MAP 13



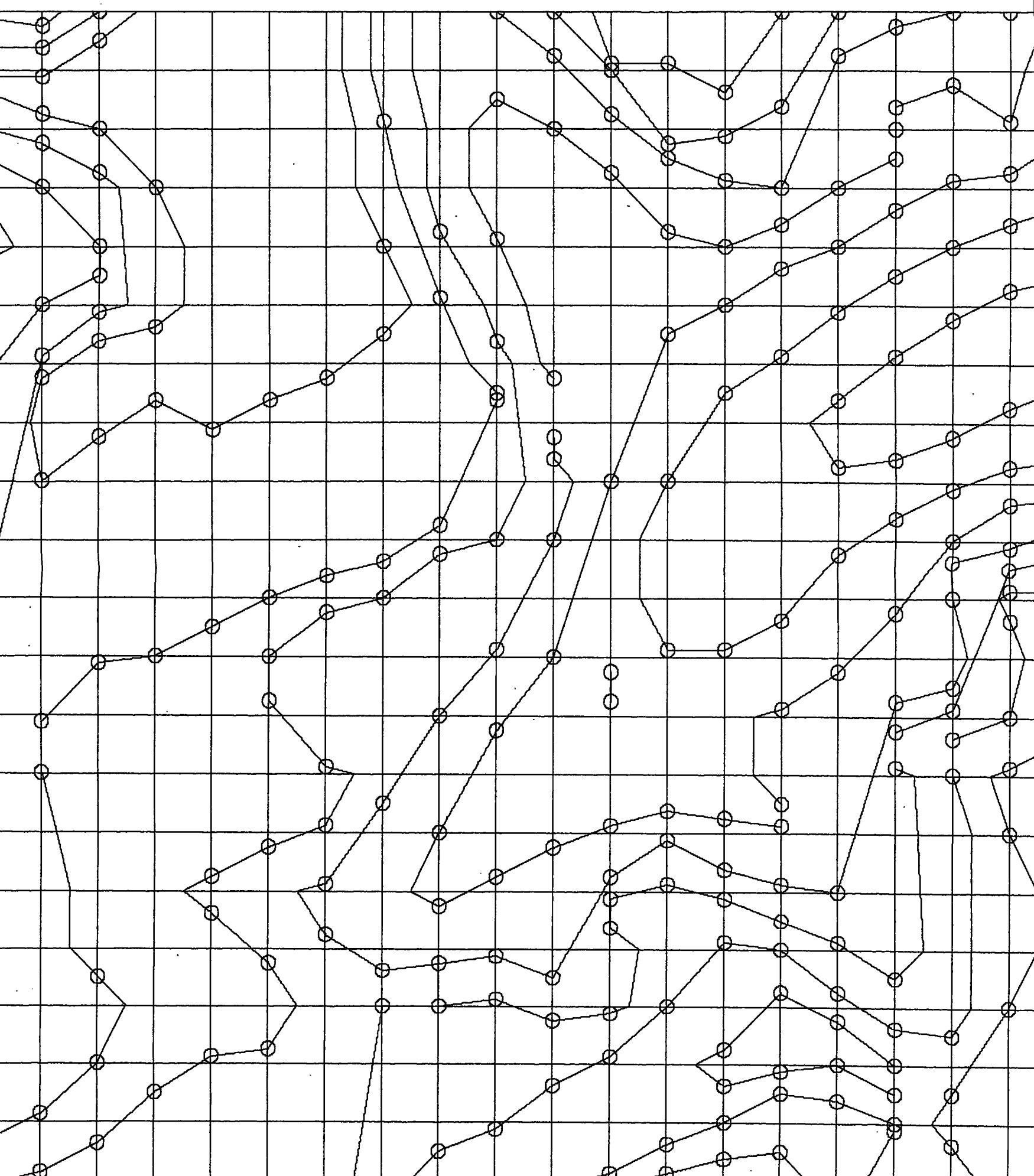
MAP 14



MAP 15



MAP 16



Appendix A. SPLINE INTERPOLATION

Let $f(x)$ be a function of one variable with a given set of values at points $\{x_i\}$ ($i = 1, 2, \dots, N$) (it may be, for example, a set of sample points with the values of the contour lines, obtained from digitizing by raster scanning). The function $f(x)$ is approximated by a sequence of auxiliary functions $\{g_i(x)\}$ where the function $g_i(x)$ is valid over an interval $[x_i, x_{i+1}]$ for $i = 2, 3, \dots, N-2$. The functions $g_i(x)$ are to be defined in such a manner that over two adjacent closed intervals $[x_i, x_{i+1}]$ and $[x_{i+1}, x_{i+2}]$ both the functions $g_i(x)$ and $g_{i+1}(x)$ and their first derivatives $\frac{dg_i}{dx}$ and $\frac{dg_{i+1}}{dx}$ are continuous at the point x_{i+1} . Taking as a class of auxiliary functions the third degree polynomial i.e.

$$g_i(x) = \sum_{k=0}^3 a_{ki} x^k \quad \text{for } x_i \ll x \ll x_{i+1}$$

the coefficients a_{ki} are computed such that

$$g_i(x_i) = f(x_i) = y_i$$

$$g_i(x_{i+1}) = f(x_{i+1}) = y_{i+1}$$

$$\left. \frac{dg_i}{dx} \right|_{x=x_{i+1}} = \left. \frac{df}{dx} \right|_{x=x_i} \quad (1)$$

$$\left. \frac{dg_i}{dx} \right|_{x=x_{i+1}} = \left. \frac{df}{dx} \right|_{x=x_{i+1}}$$

Because the values of derivative $\frac{df}{dx}$ are usually unknown at the points x_i, x_{i+1} they may be estimated as follows

$$\left. \frac{df}{dx} \right|_{x=x_i} \approx \frac{f(x_{i+1}) - f(x_{i-1})}{x_{i+1} - x_{i-1}} = y'_i$$

$$\left. \frac{df}{dx} \right|_{x=x_{i+1}} \approx \frac{f(x_{i+2}) - f(x_i)}{x_{i+2} - x_i} = y'_{i+1} \quad (2)$$

Substituting equations (2) into equations (1) we compute the values of coefficients a_{ki} . To simplify the computations the interval $[x_i, x_{i+1}]$ can be transformed into the interval $[0, \bar{x}_i]$ where $\bar{x}_i = x_{i+1} - x_i$. Because it is a shifting it does not affect the values of derivatives. Thus we obtain the following set of equations

$$\begin{aligned} a_{0i} &= y_i \\ a_{0i} + a_{1i}\bar{x}_i + a_{2i}\bar{x}_i^2 + a_{3i}\bar{x}_i^3 &= y_{i+1} \end{aligned}$$

$$\begin{aligned} a_{1i} &= y'_i \\ a_{1i} + 2a_{2i}\bar{x}_i + 3a_{3i}\bar{x}_i^2 &= y'_{i+1} \end{aligned}$$

Thus

$$a_{0i} = y_i$$

$$a_{1i} = y'_i$$

$$a_{2i} = \frac{3}{2\bar{x}_i^2} (y'_i - y'_i) - \frac{1}{\bar{x}_i} (2y_{i+1} + y'_{i+1})$$

$$a_{3i} = \frac{2}{\bar{x}_i^3} (y_i - y'_i) + \frac{1}{\bar{x}_i^2} (y_{i+1} + y'_{i+1})$$

It is easy to show that the functions $g_i(x)$ and $g_{i+1}(x)$ and their first derivatives are continuous at the point x_{i+1} .

Appendix B. A MACHINE ALGORITHM TO CREATE A QUANTIZED DATA MATRIX FOR DATA OBTAINED BY RASTER SCANNING (with FORTRAN subroutines)

Definition of symbols

- P_k^m - the k-th point situated along the m-th track, $k=1,2,\dots,K_m$
 where K_m is total number of points located on the m-th track.
- $Z(P_k^m)$ - the value of the contour line at the point P_k^m
- M - total number of tracks
- G_{nm} - the n-th grid point situated on the m-th track
- N - total number of grid points situated along one track
- ΔZ - contour spacing

This algorithm is for the determination of quantized data for m-th track where $m=1,2,\dots,M$

Step 1. Find two successive points P_k^m and P_{k+1}^m such that $Z(P_k^m) \neq Z(P_{k+1}^m)$

Step 2. Determine the trend T , i.e. if $Z(P_{k+1}^m) \neq Z(P_k^m)$ then $T = -1$, otherwise $T=1$, store value of T .

Step 3. At points G_{nm} situated between P_k^m and P_{k+1}^m assign value $(Z(P_k^m) + Z(P_{k+1}^m))/2$.

Step 4. Move to the segment (P_{k+1}^m, P_{k+2}^m) . When $Z(P_{k+1}^m) \neq Z(P_{k+2}^m)$ then assign the value $(Z(P_{k+1}^m) + Z(P_{k+2}^m))/2$ to the points G_{nm} situated in this segment; when $Z(P_{k+1}^m) = Z(P_{k+2}^m)$ assign to the points G_{nm} the value $Z(P_{k+1}^m) + T \cdot \Delta Z/2$, and multiply T by -1 to obtain a new trend.

Step 5. Repeat step 4 up to $P_{K_m}^m$.

Step 6. At the points G_{nm} situated in $(P_{K_m}^m, G_{N_m})$ assign value $Z(P_{K_m}^m) - \Delta Z/2$ if $Z(P_{K_m}^m) > Z(P_{K_m-1}^m)$ or $Z(P_{K_m}^m) + \Delta Z/2$ if $Z(P_{K_m}^m) < Z(P_{K_m-1}^m)$. If $Z(P_{K_m}^m) = Z(P_{K_m-1}^m)$ then assign value $Z(P_{K_m}^m) + T \Delta Z/2$.

Step 7. Take the initial value of trend T (step 2) and multiply by -1 .

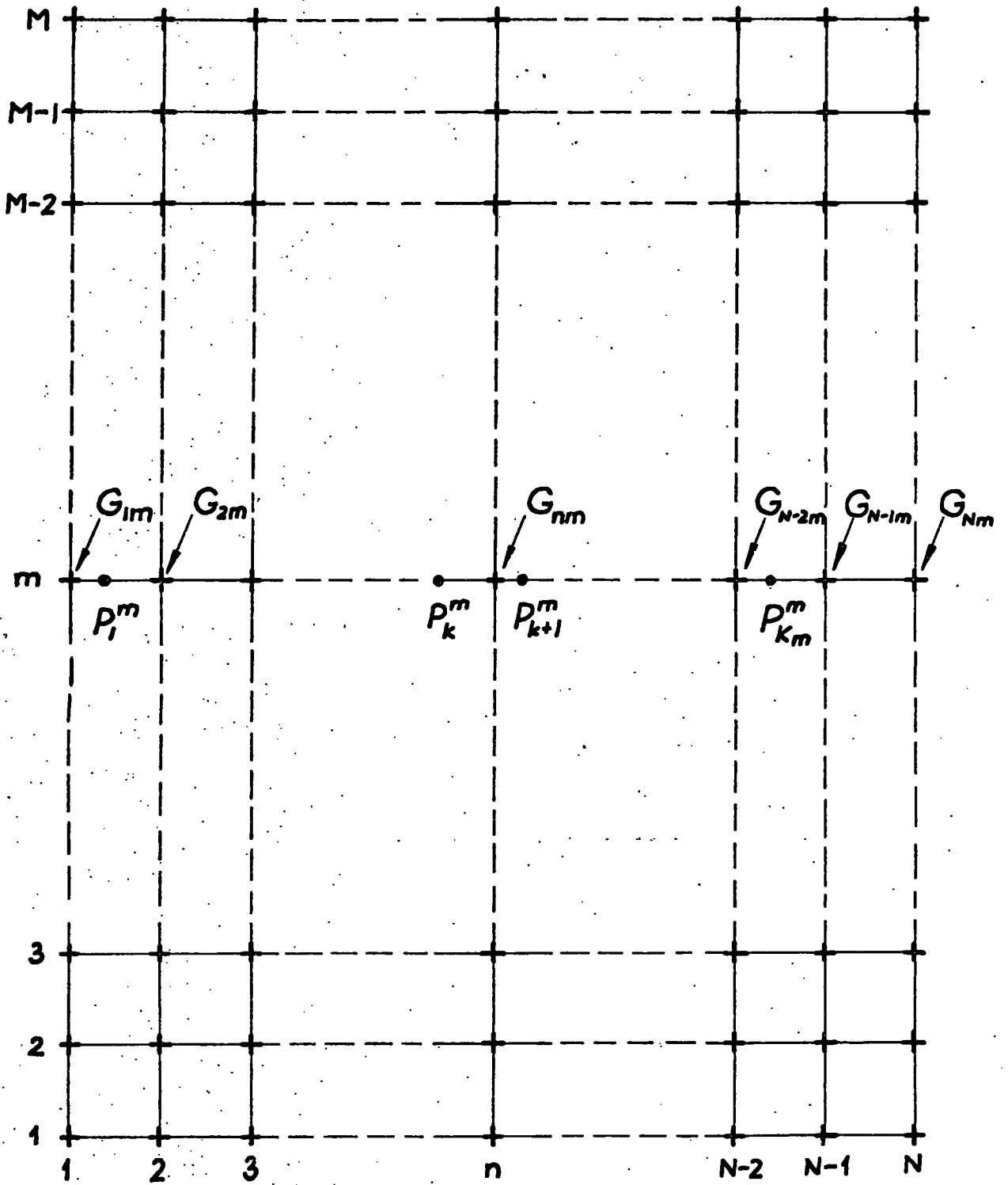


Figure B

Step 8. Move to the segment (P_{k-1}^m, P_k^m) . When $Z(P_{k-1}^m) \neq Z(P_k^m)$ then assign the value $(Z(P_{k-1}^m) + Z(P_k^m))/2$ to the points G_{nm} situated in this segment; when $Z(P_{k-1}^m) = Z(P_k^m)$ assign to the points G_{nm} the value $Z(P_{k-1}^m) + T\Delta Z/2$, and multiply T by -1 to obtain a new trend.

Step 9. Repeat step 8 up to P_1^m .

Step 10. At the points G_{nm} situated in $[G_{1m}, P_1^m)$ assign value $Z(P_1^m) - \Delta Z/2$ if $Z(P_2^m) > Z(P_1^m)$ or $Z(P_1^m) + \Delta Z/2$ if $Z(P_2^m) < Z(P_1^m)$. If $Z(P_1^m) = Z(P_2^m)$ then at the points G_{nm} assign value $Z(P_1^m) + T \cdot \Delta Z/2$.

Step 11. At the points G_{nm} if there is such a k that $G_{nm} = P_k^m$ then assign value $Z(P_k^m)$.

```

1.000 SUBROUTINE QUANT(XMIN,GRID,G,N,P,Z,K,DELZ)
2.000 DIMENSION G(N),P(K),Z(K)
3.000
4.000 C SUBROUTINE COMPUTES QUANTIZED DATA /FOR CNE TRACK/ WHICH ARE
5.000 C STORED IN ARRAY G /N - NUMBER OF GRID POINTS CN ONE TRACK/
6.000 C P(I) - CO-ORDINATE OF THE I-TH SAMPLE POINT,
7.000 C Z(I) - VALUE OF CCNTCUR LINE AT P(I); /I=1,2,....,K/
8.000 C K - NUMBER OF SAMPLE POINTS,
9.000 C XMIN - CO-ORDINATE OF THE FIRST GRID POINT,
10.000 C GRID - GRID SIZE /TRACK SPACING/,
11.000 C DELZ - CCNTCUR INTERVAL.
12.000 C
13.000 DO 11 I=1,N
14.000 XB=XMIN+FLOAT(I-1)*GRID
15.000 DO 11 J=1,K
16.000 IF(XB.EQ.P(J)) G(I)=Z(J)
17.000 11 CONTINUE
18.000 KK=K-1
19.000 DO 1 J=1,KK
20.000 IF(Z(J).NE.Z(J+1)) GO TO 2
21.000 1 CONTINUE
22.000 J=KK
23.000 JP=J
24.000 TT=1
25.000 IF(Z(J+1).LT.Z(J)) TT=-1
26.000 T=TT
27.000 DO 3 J=JP,KK
28.000 IF(Z(J).NE.Z(J+1)) GO TO 4
29.000 ZP=Z(J)+T*DELZ
30.000 T=T*(-1)
31.000 GO TO 6
32.000 4 ZP=(Z(J)+Z(J+1))/2.0
33.000 6 CCNTINUE
34.000 DO 7 I=1,N
35.000 XB=XMIN+FLOAT(I-1)*GRID
36.000 IF(XB.LE.P(J).OR.XE.GE.P(J+1)) GO TO 7
37.000 G(I)=ZP
38.000 7 CONTINUE
39.000 3 CONTINUE
40.000 IF(Z(K).EQ.Z(K-1)) GO TO 8

```

```

41.000 IF(Z(K).GT.Z(K-1)) ZP=Z(K)+DELZ
42.000 IF(Z(K).LT.Z(K-1)) ZP=Z(K)-DELZ
43.000 GO TO 9
44.000 8 ZP=Z(K)+T*DELZ
45.000 9 CONTINUE
46.000 DO 10 I=1,N
47.000 XB=XMIN+FLCAT(I-1)*GRIC
48.000 IF(XB.LE.P(K)) GC TO 10
49.000 G(I)=ZP
50.000 10 CONTINUE
51.000 IF(JP.EQ.1) GC TO 20
52.000 KK=JP-1
53.000 T=TT*(-1)
54.000 DO 20 JJ=1, KK
55.000 J=JP-JJ
56.000 IF(Z(J).NE.Z(J+1)) GO TO 14
57.000 ZP=Z(J)+T*DELZ
58.000 T=T*(-1)
59.000 GO TO 16
60.000 14 ZP=(Z(J)+Z(J+1))/2.0
61.000 16 CONTINUE
62.000 DO 20 I=1,N
63.000 XB=XMIN+FLOAT(I-1)*GRIC
64.000 IF(XB.LE.P(J).OR.XB.GE.P(J+1)) GC TO 20
65.000 G(I)=ZP
66.000 20 CONTINUE
67.000 IF(Z(1).EQ.Z(2)) GC TO 18
68.000 IF(Z(1).LT.Z(2)) ZP=Z(1)-DELZ
69.000 IF(Z(1).GT.Z(2)) ZP=Z(1)+DELZ
70.000 GO TO 19
71.000 18 ZP=Z(1)+T*DELZ
72.000 19 CONTINUE
73.000 DO 21 I=1,N
74.000 XB=XMIN+FLCAT(I-1)*GRIC
75.000 IF(XB.GE.P(1)) GC TO 21
76.000 G(I)=ZP
77.000 21 CONTINUE
78.000 RETURN
79.000 END

```

Appendix C. A MACHINE ALGORITHM TO DISPLAY DATA OBTAINED BY RASTER SCANNING IN CONTOUR MAP FORM (with FORTRAN subroutine).

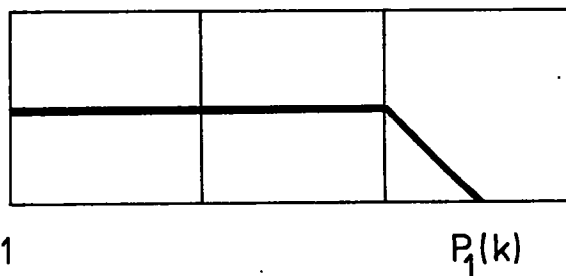
The map is generated in raster mode and only two tracks are used at a time. All variables referring to the lower track (Figure C a) bearing the subscript 1 for example $G_1(1), \dots, G_1(M)$ denote the quantized values at the grid points on the lower track. $P_1(k)$ denotes co-ordinate of the k-th sample point on the lower track and $Z_1(k)$ is the value of the contour line at this point. All variables connected with the top track have the subscript 2 for example $P_2(k), Z_2(k), G_2(m)$.

1. From the sequence of points on the lower track take a point which has not been already used - if all have been used go to 10. Let us denote this point by $P_1(k)$; then the value of contour line at this point is $Z_1(k)$. Find the co-ordinates of grid points such that $P_1(k)$ is situated between them (Figure C b). Denote them by X_m and X_{m+1} .
2. For all vertical sides of grid squares determine whether or not they are crossed by the contour line with value $Z_1(k)$. Store the results of the test in matrix V as follows:

if $(Z_1(k) - G_1(m)) (Z_1(k) - G_2(m)) < 0$ i.e. contour line crosses this side then $V(m) = 0$

3. If $P_1(k)$ is not the first point in the segment (X_m, X_{m+1}) or $V(m) = 0$ go to 5.

If $V(1) = V(2) = \dots = V(m) = 1$ then draw the line joining the point $P_1(k)$ with the point on the left side of the first square (co-ordinate of this point determined using linear interpolation), indicate that the point $P_1(k)$ has been used and go to 1, otherwise go to 4.



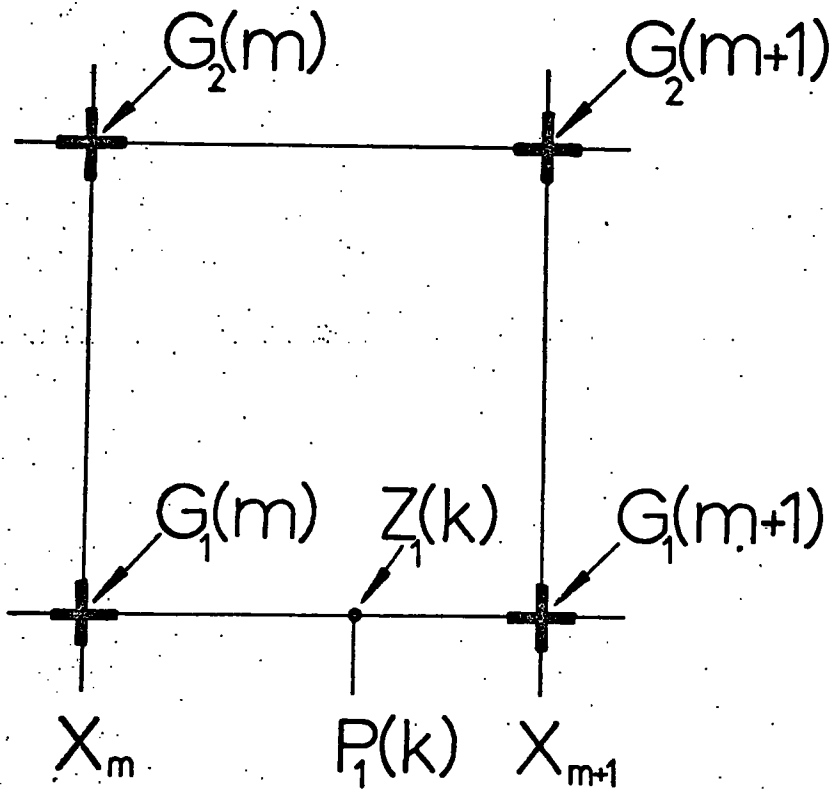
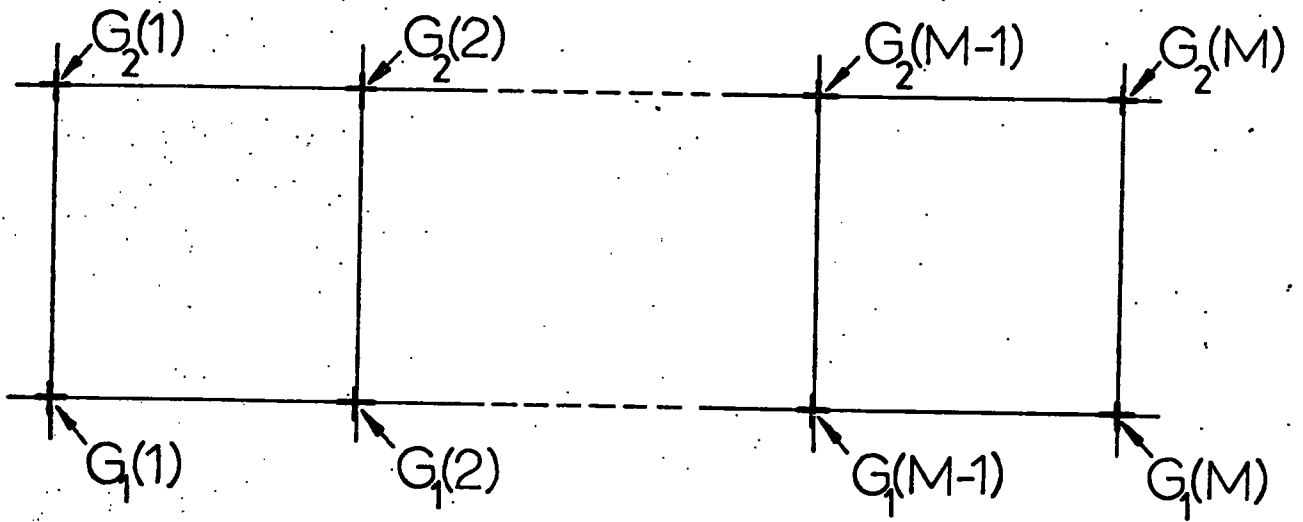
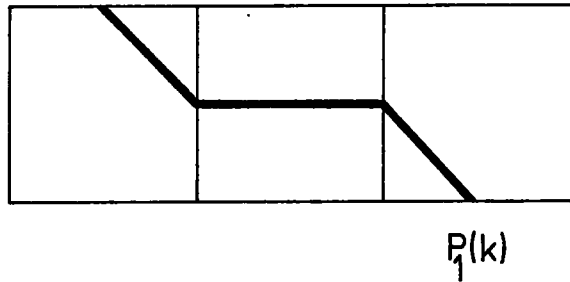


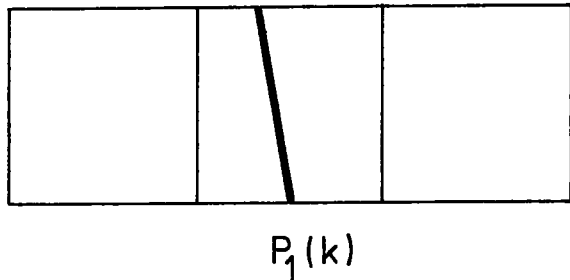
Figure C

4. From a set of points on the top track choose a point so that:
- the value of the contour line is the same as at $P_1(k)$ and
 - it is situated in (X_1, X_m) and
 - it is closest to $P_1(k)$

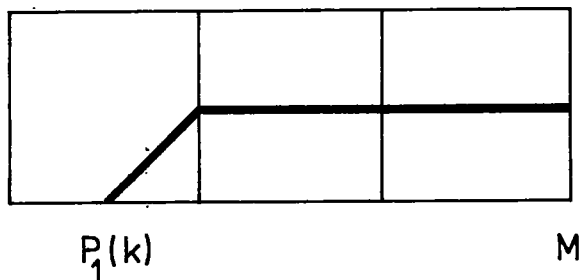
then join the point $P_1(k)$ with this point, indicate that both points have been used and go to 1.



5. If among the points on the top track there is such a point that it is situated in (X_m, X_{m+1}) and the value of the contour line is the same as in $P_1(k)$ then join this point with $P_1(k)$, indicate that both points have been used and go to 1; otherwise go to 6.



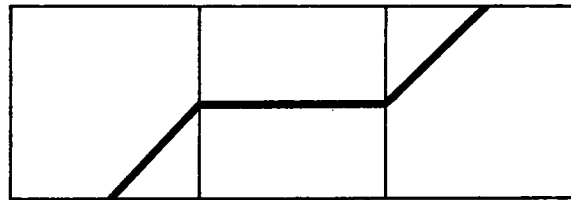
6. If $V_{m+1} = 0$ go to 9 otherwise if $V_{m+2} = \dots = V_M = 1$ then join the point $P_1(k)$ with the point on the right side of the last square (co-ordinate of this point determined using linear interpolation), indicate that the point $P_1(k)$ has been used and go to 1; otherwise go to 7.



7. If $V(l) = 0$ for $l > m+1$ then find if among the points on the top track there is such a point that:

- a. it is situated in (X_{l-1}, X_l) and
- b. the value at the contour line at this point is the same as at $P_1(k)$

if so then join both points, indicate that they have been used and go to 1; otherwise go to 8.

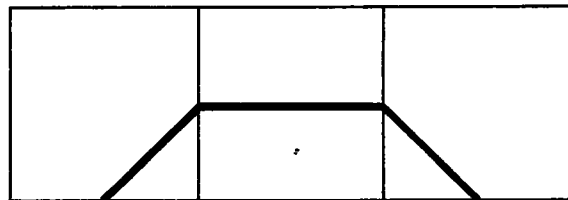


$P_1(k)$

8. Find among the points on the lower track a point such that:

- a. it is situated in (X_{l-1}, X_l) and
- b. the value of the contour line is the same as at $P_1(k)$

then join both points, indicate that they have been used and go to 1.

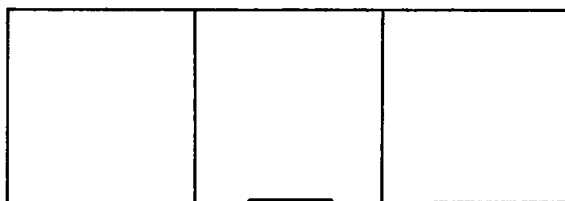


$P_1(k)$

9. Among the points on the lower track find such a point that:

- a. it is situated in (X_m, X_{m+1}) and
- b. the value of the contour line is the same as at $P_1(k)$

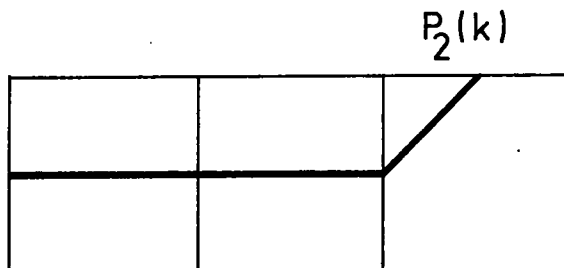
then join both points, indicate that they have been used and go to 1.



$P_1(k)$

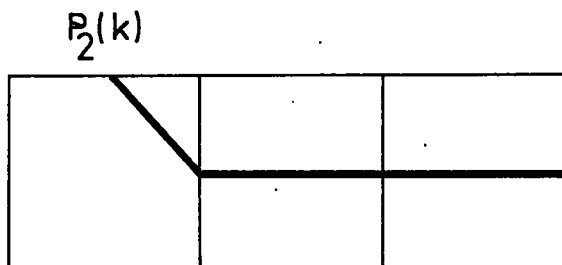
10. From the points situated on the top track take the first point that has not been already used - if all have been used then STOP - determine the segment in which it is situated. Let us denote this segment by (X_m, X_{m+1}) and the point by $P_2(k)$, determine which vertical sides of grid squares are crossed with the contour line $Z_2(k)$, store the results of the test in matrix V (cf. 2) and go to 11.

11. If $V(m) = 0$ go to 12; otherwise join the point $P_2(k)$ with the point on the left side of the first square, indicate that the point $P_2(k)$ has been used and go to 10.



1

12. If $V(m+1) = 0$ go to 14 otherwise if $V(m+2) = V(m+3) = \dots = V(M) = 1$ then join the point $P_2(k)$ with the point on the right side of the last square (co-ordinate of this point determined using linear interpolation) indicate that $P_2(k)$ has been used and go to 10.

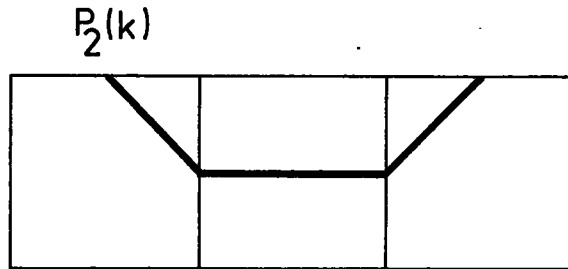


M

13. If $V(l) = 0$ for $l > m+1$ then find among the point on the top track a point such that:

- a. it is situated in (X_{l-1}, X_l) and
- b. the value of the contour line at this point is the same as at $P_2(k)$

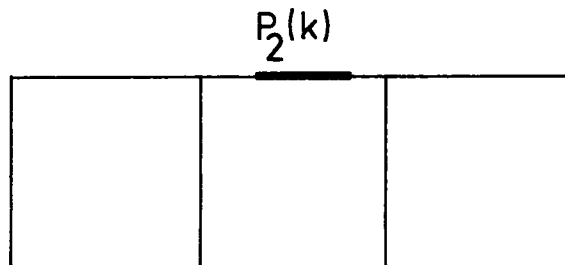
thenjoin them, indicate that they have been used and go to 10.



14. Among the points on the top track find a point such that:

- a. it is situated in (X_m, X_{m+1}) and
- b. the value of the contour line at this point is the same as at $P_2(k)$

thenjoin them, indicate that they have been used and go to 10.



```

1.000 SUBROUTINE DISPSC(XMIN,XMAX,GRID,YMIN,YMAX,DELZ)
2.000 DIMENSION P1(48),Z1(48),IP1(48),P2(48),Z2(48),IP2(48)
3.000 DIMENSION G1(21),G2(21),IV(21),IT(21)
4.000
5.000 C SUBROUTINE DISPLAYS SCAN-MODE DATA IN CONTOUR MAP FORM
6.000 C XMIN - CO-ORDINATE OF THE FIRST GRID POINT CN THE TRACKS,
7.000 C XMAX - CO-ORDINATE OF THE LAST GRID POINT CN THE TRACKS,
8.000 C GRID - GRID SIZE /TRACK SPACING/,
9.000 C YMIN - CO-ORDINATE OF THE LOWEST TRACK,
10.000 C YMAX - CO-ORDINATE OF THE HIGHEST TRACK,
11.000 C DELZ - CONTOUR INTERVAL,
12.000 C INPUT DATA ARE READ FROM UNIT 3 AND THEY SHOULD BE
13.000 C WRITTEN IN THE FOLLOWING FORM - PAIRS OF NUMBERS:
14.000 C CO-ORDINATE OF A SAMPLE POINT /F4.1/ AND VALLE CF
15.000 C CONTOUR LINE AT THIS POINT /F4.0/, EIGHT PAIRS PER ROW
16.000 C VALUE 99.9 AS A CO-ORDINATE OF A SAMPLE PCINT INDICATES
17.000 C THAT THE PREVIOUS SAMPLE PCINT IS THE LAST ONE CN THE
18.000 C TRACK.
19.000 C
20.000 C CALL PAPER(1)
21.000 C CALL PSPACE(0.0,1.0,0.0,1.0)
22.000 C CALL MAP(XMIN,XMAX,YMIN,YMAX)
23.000 C CALL BORDER
24.000 C N=(XMAX-XMIN)/GRID+1.
25.000 C MM=(YMAX-YMIN)/GRID
26.000 C DO 73 K=1,6
27.000 C KL=(K-1)*8+1
28.000 C KR=KL+7
29.000 C READ(3,74)(P2(KK),Z2(KK),KK=KL,KR)
30.000 C FCRMAT(8(F4.1,F4.0))
31.000 C DO 75 KK=KL,KR
32.000 C IF(P2(KK).EQ.99.9) GO TO 78
33.000 C 74 CCNTINUE
34.000 C 73 CONTINUE
35.000 C KK=49
36.000 C 78 K2=KK-1
37.000 C CALL QUANT(XMIN,GRID,G2,N,P2,Z2,K2,DELZ)
38.000 C Y=YMAX
39.000 C DO 111 K=1,K2
40.000 C CALL POSITN(P2(K),Y)

```

```

41.000 111 CALL CIRCLE(0.5)
42.000 DO 105 J=1,M
43.000 YG=Y
44.000 Y=Y-GRID
45.000 CALL POSITN(0.,Y)
46.000 DO 63 K=1,6
47.000 KL=(K-1)*E+1
48.000 KR=KL+7
49.000 READ(3,74)(PI(KK),Z1(KK),KK=KL,KR)
50.000 DO 65 KK=KL,KR
51.000 IF(PI(KK).EQ.99.9) GO TO 68
52.000 65 CONTINUE
53.000 63 CONTINUE
54.000 KK=49
55.000 68 K1=KK-1
56.000 CALL QUANT(XMIN,GRID,G1,N,PI,Z1,K1,DELZ)
57.000 DO 50 K=1,K1
58.000 50 IP1(K)=0
59.000 DO 51 K=1,K2
60.000 51 IP2(K)=0
61.000 DO 52 K=1,N
62.000 52 IT(K)=0
63.000 DO 110 K=1,K1
64.000 CALL POSITN(PI(K),Y)
65.000 110 CALL CIRCLE(0.5)
66.000 CALL POSITN(0.,Y)
67.000 DO 100 K=1,K1
68.000 IF(IP1(K).EQ.1) GC TO 100
69.000 CALL POSITN(PI(K),Y)
70.000 LL=(PI(K)-XMIN)/GRID+1
71.000 LR=LL
72.000 IF(PI(K).GT.FLOAT(LL-1)*GRID+XMIN) LR=LR+J
73.000 DO 1 L=1,N
74.000 IV(L)=0
75.000 IF((Z1(K)-G1(L))*(Z1(K)-G2(L)).LT.0.) IV(L)=1
76.000 1 CONTINUE
77.000 XL=XMIN+FLOAT(LL-1)*GRID
78.000 XR=XL+GRID
79.000 IF(LR.GT.LL) GO TC 5
80.000 LL=LL-1

```

```

81.000 LR=LR+1
82.000 XL=XL-GRID
83.000 5 CONTINUE
84.000 IF(LL.LT.1) GC TO 7
85.000 ITEST=0
86.000 IF(K.EQ.1) GO TC 9
87.000 KM=K-1
88.000 DO 9 KB=1,KM
89.000 IF(Z1(KB).NE.Z1(K)) GO TO 9
90.000 IF(P1(KB).LT.XL.OR.P1(KB).GT.XR) GC TC 9
91.000 ITEST=1
92.000 KS=KB
93.000 9 CONTINUE
94.000 IF(IV(LL).EQ.1.AND.ITEST.EQ.1) GO TO 7
95.000 IF(IV(LL).EQ.C) GO TO 7
96.000 ITEST=0
97.000 DO 2 KK=1,K2
98.000 IF(IP2(KK).EQ.1) GC TC 2
99.000 IF(Z2(KK).NE.Z1(K)) GC TC 2
100.000 IF(P2(KK).GT.XL) GO TO 3
101.000 ITEST=1
102.000 KS=KK
103.000 2 CONTINUE
104.000 3 CONTINUE
105.000 IF(ITEST.EQ.1) GO TO 20
106.000 GO TO 11
107.000 7 CONTINUE
108.000 ITEST=0
109.000 DYST1=2.*(XMAX-XMIN)
110.000 DO 12 KK=1,K2
111.000 IF(IP2(KK).EQ.1) GC TC 12
112.000 IF(Z2(KK).NE.Z1(K)) GC TC 12
113.000 IF(P2(KK).LT.XL.OR.P2(KK).GT.XR) GO TO 12
114.000 ITEST=1
115.000 DYST=SQRT((P1(K)-P2(KK))*2+GRID*GRID)
116.000 IF(DYST.GE.DYST1) GC TC 12
117.000 DYST1=DYST
118.000 KS=KK
119.000 12 CONTINUE
120.000 IF(ITEST.EQ.C) GO TO 30

```

```

141.000 13 IPI(K)=1
142.000   CALL JCIN(P2(KS),YG)
143.000   CALL POSITN(P2(KS),YG)
144.000   IP2(KS)=1
145.000   CALL PCSITN(O.,Y)
146.000   GO TO 100
147.000 11 IF(IV(1).EQ.0) GO TO 45
148.000   YB=((Z1(K)-G1(1))/(G2(1)-G1(1)))*GRIC
149.000   YB=Y+YB
150.000   CALL JCIN(O.,YB)
151.000   IPI(K)=1
152.000   CALL POSITN(O.,Y)
153.000   GO TO 100
154.000 20 CALL JCIN(P2(KS),YG)
155.000   CALL POSITN(P2(KS),YG)
156.000   IPI(K)=1
157.000   IP2(KS)=1
158.000   CALL POSITN(O.,Y)
159.000   GO TO 100
160.000 30 IF(IV(LR).EQ.1) GO TO 37
161.000   ITEST=0
162.000   KM=K+1
163.000   DO 38 KB=KM,K1
164.000     IF(Z1(KB).NE.Z1(K)) GO TO 38
165.000     IF(P1(KB).LT.XL.CR.PI(KB).GT.XR) GO TO 38
166.000     CALL POSITN(PI(K),Y)
167.000     CALL JOIN(PI(KB),Y)
168.000     CALL POSITN(PI(KB),Y)
169.000     IPI(KB)=1
170.000     IPI(K)=1
171.000     GO TO 45
172.000 38 CONTINUE
173.000 37 XL=XR
174.000   XR=XR+GRIC
175.000   KM=K+1
176.000   ITEST=0
177.000   DO 31 KB=KM,K1
178.000     IF(Z1(KB).NE.Z1(K)) GC TC 31
179.000     IF(P1(KB).LT.XL.OR.PI(KB).GT.XR) GO TO 31
180.000     ITEST=1

```

```

161.000 KS=KB
162.000 GO TO 33
163.000 31 CONTINUE
164.000 IF(ITEST.EQ.0) GO TO 34
165.000 33 IF(IV(LR).NE.1) GO TO 45
166.000 YB=((Z1(K)-G1(LR))/(G2(LR)-G1(LR)))*GRID
167.000 YB=Y+YB
168.000 CALL JOIN(XL,YB)
169.000 CALL JOIN(PI(KS),Y)
170.000 CALL POSITN(PI(KS),Y)
171.000 IP1(KS)=1
172.000 CALL POSITN(O.,Y)
173.000 GO TO 100
174.000 34 DO 35 KK=1,K2
175.000 IF(Z2(KK).NE.Z1(K)) GO TO 35
176.000 IF(P2(KK).LT.XL.OR.P2(KK).GT.XR) GO TO 35
177.000 ITEST=1
178.000 KS=KK
179.000 GO TO 36
180.000 35 CONTINUE
181.000 IF(IV(LR).NE.1) GO TO 45
182.000 YB=((Z1(K)-G1(LR))/(G2(LR)-G1(LR)))*GRID
183.000 YB=Y+YB
184.000 CALL JOIN(XL,YB)
185.000 LL=LR
186.000 LR=LR+1
187.000 IF(LR.GE.22) GO TO 45
188.000 GO TO 37
189.000 36 CALL JCIN(P2(KS),YG)
190.000 CALL POSITN(P2(KS),YG)
191.000 IP2(KS)=1
192.000 CALL POSITN(O.,Y)
193.000 GO TO 100
194.000 45 CALL PCSITN(O.,Y)
195.000 100 CONTINUE
196.000 DO 200 KK=1,K2
197.000 IF(IP2(KK).EQ.1) GC TO 200
198.000 LL=(P2(KK)-XMIN)/GRID+1
199.000 LR=LL
200.000 XL=XMIN+(LL-1)*GRID

```

```

201.000 XR=XL+GRID
202.000 IF(P2(KK).GT.FLOAT(LL-1))*GRID) LR=LR+1
203.000 IF(LR.GT.LL) GO TO 250
204.000 LL=LL-1
205.000 LR=LR+1
206.000 XL=XL-GRID
207.000 250 CONTINUE
208.000 DO 201 L=1,N
209.000 IV(L)=0
210.000 IF((Z2(KK)-G1(L))*(Z2(KK)-G2(L)).LT.0) IV(L)=1
211.000 201 CONTINUE
212.000 ITEST=0
213.000 IF(KK.EQ.1) GO TO 202
214.000 KM=KK-1
215.000 DO 202 KB=1,KM
216.000 IF(KB.EQ.KK.OR.IP2(KB).EQ.1) GO TO 202
217.000 IF(P2(KB).LT.XL.CR.P2(KB).GT.XR) GO TO 202
218.000 IF(Z2(KK).NE.Z2(KB)) GC TC 202
219.000 ITEST=1
220.000 KS=KB
221.000 GO TO 205
222.000 202 CONTINUE
223.000 IF(IV(LL).EQ.1) GC TC 210
224.000 206 CONTINUE
225.000 IF(IV(LR).EQ.1) GO TO 220
226.000 GO TO 200
227.000 210 IF(IV(1).EQ.0) GO TO 245
228.000 YB=((Z2(KK)-G1(1))/(G2(1)-G1(1)))*GRID
229.000 CALL POSITN(P2(KK),YG)
230.000 YB=Y+YB
231.000 CALL JOIN(O.,YB)
232.000 CALL POSITN(O.,Y)
233.000 GO TO 200
234.000 205 CALL POSITN(P2(KS),YG)
235.000 CALL JCIN(P2(KK),YG)
236.000 CALL PCITN(P2(KK),YG)
237.000 IF(Z2(KK+1).NE.Z2(KK).OR.IP2(KK+1).EQ.1) GC TC 209
238.000 CALL JOIN(P2(KK+1),YG)
239.000 IP2(KK+1)=1
240.000 KK=KK+1

```



```

241.000 LL=(P2(KK)-XMIN)/GRID+1
242.000 LR=LL+1
243.000 205 CONTINUE
244.000 CALL POSITN(0.,Y)
245.000 IP2(KK)=1
246.000 IP2(KS)=1
247.000 GO TO 206
248.000 220 IF(P2(KK).GE.XMAX) GO TO 245
249.000 CALL POSITN(P2(KK),YG)
250.000 YB=((Z2(KK)-G1(LR))/(G2(LR)-G1(LR)))*GRID+Y
251.000 CALL JOIN(XR,YB)
252.000 IF(LR.EQ.21.OR.KK.EQ.K2) GO TO 245
253.000 ITEST=0
254.000 KM=KK+1
255.000 DO 221 KB=KM,K2
256.000 IF(IP2(KB).EQ.1) GO TO 221
257.000 IF(Z2(KK).NE.Z2(KB)) GO TO 221
258.000 ITEST=1
259.000 KS=KB
260.000 GO TO 222
261.000 221 CONTINUE
262.000 IF(IV(21).EQ.0) GC TO 245
263.000 YB=((Z2(KK)-G1(21))/(G2(21)-G1(21)))*GRID+Y
264.000 CALL JOIN (XMAX,YB)
265.000 CALL POSITN(C.,Y)
266.000 GO TO 200
267.000 222 LL=(P2(KB)-XMIN)/GRID+1
268.000 XL=XMIN+FLOAT(LL-1)*GRID
269.000 IF(P2(KB).EQ.XL) LL=LL-1
270.000 IF(P2(KB).EQ.XL) XL=XL-GRID
271.000 IF(IV(LL).NE.1) GO TO 245
272.000 YB=((Z2(KK)-G1(LL))/(G2(LL)-G1(LL)))*GRID+Y
273.000 CALL JOIN(XL,YB)
274.000 223 CALL JCIN(P2(KB),YG)
275.000 CALL PCSITN(P2(KB),YG)
276.000 IP2(KB)=1
277.000 CALL PCSITN(0.,Y)
278.000 GO TO 200
279.000 245 CALL POSITN(0.,Y)
280.000 200 CONTINUE

```

```
281.000      DC 101 K=1,K1
282.000      P2(K)=P1(K)
283.000      Z2(K)=Z1(K)
284.000      101 CONTINUE
285.000      K2=K1
286.000      DO 102 K=1,N
287.000      102 G2(K)=G1(K)
288.000      105 CONTINUE
289.000      500 CALL GRATSI(GRID,GRID)
290.000      CALL GREND
291.000      RETURN
292.000      END
```

The subroutine DISPSC is written in ANSI FORTRAN IV on IBM computers (360/67 and 370/168) working on NUMAC (Northumbrian Universities Multiple Access Computer) system. It uses subroutine QUANT (Appendix B) and graphic routines from *GHOST package.

