NAL TM-198

UDC 681.32.06

航空宇宙技術研究所資料

TECHNICAL MEMORANDUM OF NATIONAL AEROSPACE LABORATORY

TM-198

等高線を描かせるプログラム

磯　部　俊　夫

1971 年 1 月

航空宇宙技術研究所
NATIONAL AEROSPACE LABORATORY

既　刊　資　料

TM-170	JR エンジン燃焼器に起きた振動燃焼	1970年2月	鈴木邦男，石井浅五郎 山中国雍
TM-171	缶形燃焼器（低圧，低温系）試験設備による航空計器の氷結（着氷）試験	1970年3月	鈴木邦男，相波哲朗 本間幸吉，服部宣夫
TM-172	自動追尾型風向風速測定器	1970年3月	川幡長勝，中谷輝臣
TM-173	FA-200 改機の動安定微係数	1970年3月	遠藤　浩，林　良生 海老沼幸成，中谷輝臣
TM-174	遷音速風胴動安定測定装置の構造と作動	1970年4月	小橋安次郎，河野長正 西　武徳，宮沢政文
TM-175	NAL-16・31D 型ロケットの研究試作	1970年5月	五代富文，近藤洋史 中井暎一，田畑浄治
TM-176	非対称自由流線の一計算法	1970年5月	髙橋　侔
TM-177	小型固体ロケットモータの振動燃焼実験 —パルス法による中周波振動燃焼の研究—	1970年5月	五代富文，伊藤克弥 西村久男，湯沢克宜 柴藤羊二
TM-178	大型低速風胴動安定微係数測定装置	1970年5月	広岡貫一，遠藤　浩 戸田亘洋，岡部祐二郎
TM-179	風胴天秤の試作	1970年5月	金成正好，北出大三
TM-180	ジェットエンジン燃焼器出口ガス流の乱れの測定(II) —レーザのドップラ効果を利用する方法—	1970年5月	相波哲朗
TM-181	航空機の乗り心地について	1970年6月	幸尾治朗
TM-182	排気系障害板の模型実験	1970年9月	田辺義一
TM-183	ロケット模型の大型低速風胴試験	1970年9月	近藤洋史，髙橋　宏 桑野尚明
TM-184	金属線へ衝突する液滴の現象	1970年9月	田丸　卓，乙幡安雄
TM-185	推進エンジン用空気取入口の予備実験	1970年9月	近藤　博，石賀保正
TM-186	NAL-25・31型ロケットの試作と飛しょう試験	1970年11月	宇宙研究グループ
TM-187	推力 300 kg ジンバル液体ロケットエンジンの揺動特性（I）	1970年11月	槙崎哲二，中野富雄 橋本亮平，竹花真一郎
TM-188	自由飛行模型 FFM-10の空力微係数におよぼす機体弾性変形の影響について	1970年11月	河崎俊夫，河本　厳 戸田　勧
TM-189	燃料蒸発管に関する研究(III) 管内の燃料—空気二相流への熱伝達	1970年12月	田丸　卓，乙幡安雄 鈴木邦男
TM-190	NAL-25・31型ロケットのノズル部FRPライナーの接着剝離の検討	1970年12月	中井暎一，五代富文 古田敏康，大竹邦彦
TM-191	大型電子計算機プログラム・ライブラリ	1970年12月	戸川隼人，磯部俊夫
TM-192	極超音速風胴M 9 ノズル較正試験	1971年1月	長洲秀夫，橋本　登 穂積弘一，松崎貴至
TM-194	航空機のSTOL 性に関する一考察	1971年1月	田辺義一
TM-195	曲面壁上の噴流に対する噴出孔形状，配列の効果について	1971年1月	西村英明，臼井　弘 井上重雄，三村富嗣雄
TM-196	NAL-16H 型ロケットの研究試作	1971年1月	宇宙研究グループ
TM-197	二次元煙風胴について	1971年1月	髙橋　宏，戸田亘洋 白井正孝

等高線を描かせるプログラム*

磯 部 俊 夫**

1. ま え が き

電子計算機による計算結果の図式化は，計算結果の解析・検討・整理等に必要なことである。本所計算センターにも *X-Y* プロッタが備えつけられており，そのためのソフトウェアも完備している。これらを使い，$y=f(x)$ の型をした関数の図式化は簡単にできる。しかし，二変数の関数 $z=f(x, y)$ の図式化は簡単にはできない。科学技術計算では，*X-Y* 平面上での現象を計算する問題がひんぱんに取り扱われ，その計算結果を図式化したいという要求がでてくる。二変数関数を図式化する一般的な方法として，等高線の形式で表現する方法がある。計算機に等高線をかかせることは，一種のパターン認識の問題でもあり，非常にむずかしい。等高線を描かせるプログラムは，すでに作られているらしいが，そのアルゴリズムを明らかにした論文は筆者が調べた限りでは見つけることができなく，わずかに解説程度の簡単なもの一件[1)]を捜し当てただけだった。

本文では，計算機に等高線をかかせる方法と，それに基いて作ったプログラム，およびその使用法について述べる。ここでは $z=f(x, y)$ を，*X-Y* 平面を適当な間隔でサンプルした格子点上の値とした行列で表わす。プログラムは，等高線をかかせるための基本プログラムと，基本プログラムの機能変更・追加をするための補助プログラムで構成されたサブルーチンの集まりであり，その処理時間を短かく，かつ使用コア数を最少になるよう種々の考慮を払ってコーデングした。このプログラムを使うことにより簡単に等高線を描くことができ，また図の座標変換，拡大，縮小，一部マスク等が容易にできる。

2. 等 高 線

x, y の二つの値とそれに対応する z の値を三次元空間の一つの点 (x, y, z) で表わし

$$z=f\ (x,\ y)$$

と関数の型で表現する。この関数が，定義されている x, y 平面上の領域で連続であるならば，この関数によって決まる点の集合は，一つの曲面を作る。この三次元空間での曲面を，xy 平面上に等高線の型式であらわす。

等高線の定義として，つぎの二つの定義の方法が考えられる。

1) $z=c$ (=constant) とおき $c=f\ (x,\ y)$ を y について解いたときの解

   $$y=g\ (x,\ c)$$

   の作る曲線

2) xy 平面でのある点 $(x,\ y)$ に対する z の値が，c より大きくなる点の集合 A_o と，そうでない点の集合 B_o に分ける。すると集合 A_o と集合 B_o とにより xy 平面は切断される。関数 $z=f\ (x,\ y)$ が連続であれば，この切断は連続な曲線となる。この切断によってできる曲線。

電子計算機を使って等高線をかかせるという目的からみた場合，定義 1) では，一般の $c=f\ (x,\ y)$ を y について解かなくてはならず現実的でない。一方定義 2) では，集合 A_o と集合 B_o を分けることが容易にできる。ここでは，等高線の定義として 2) を採用する。

3. 等高線の近似

xy 平面上の点 $(x,\ y)$ の集合が連続した閉集合 S_o であるとする。この閉集合の内部に有限個のサンプル点をとり，閉集合の標本とする。サンプル点 p とその点に対する z の値 $f\ (p)$ から $z=f\ (x,\ y)$ の等高線の近似な線を得ることを考える。

まず，格子点および隣り合う格子点をつぎのように定義する。

格子点；直角座標系において，一つの定点の X 軸方向 r の距離にある点が，定義域中に含まれるなら，その点を含み，また Y 軸方向 r の距離にある点が定義域中に含まれるなら，その点を含む全ての点の集合。

隣り合う格子点；ある格子点から r の距離にある格子点。

---

* 昭和45年12月21日受付

** 計測部

ここで，閉集合S_oの標本としてのサンプル点はつぎの条件を満すものであるとする。

1）S_oを位相的に相似性を保つ変換Tを施すことによりサンプル点を格子点にすることができる。

2）一定の正の数εが与えられたとき格子点に変換されたサンプル点pと，その隣り合せた点qとの間に

$$|f(p)-f(q)|<\varepsilon$$

の関係がある。

閉集合S_oに変換Tを行ない，サンプル点を格子点にする。変換を受けたS_oをSとする。領域Sで，$z=c$より大きな値をもつ点の集合A，Aの余集合をBとする。Aはつぎの二つの場合が考えられる。

1）Sの境界を含む集合

2）Sの境界を含まない集合

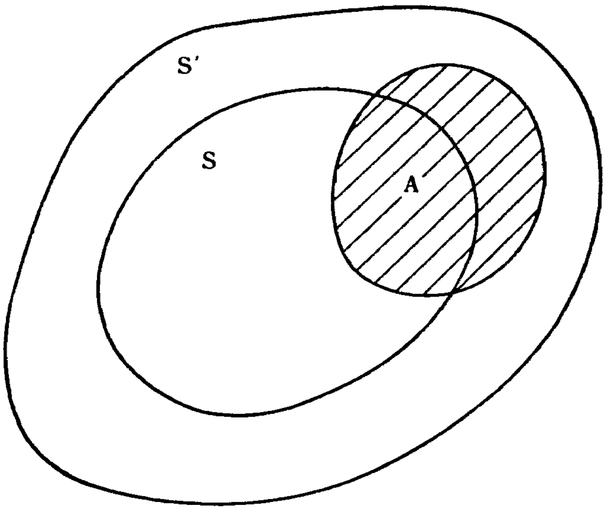

図１　仮想のS'を考え領域AをS'でおおう

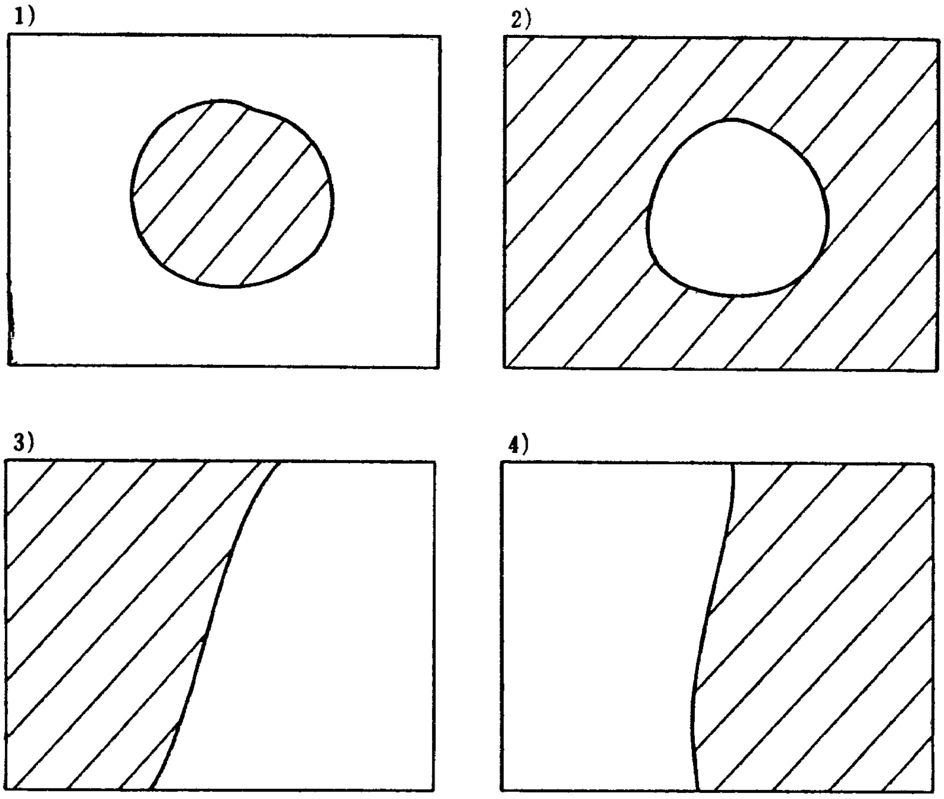

図２　等高線の基本パターン

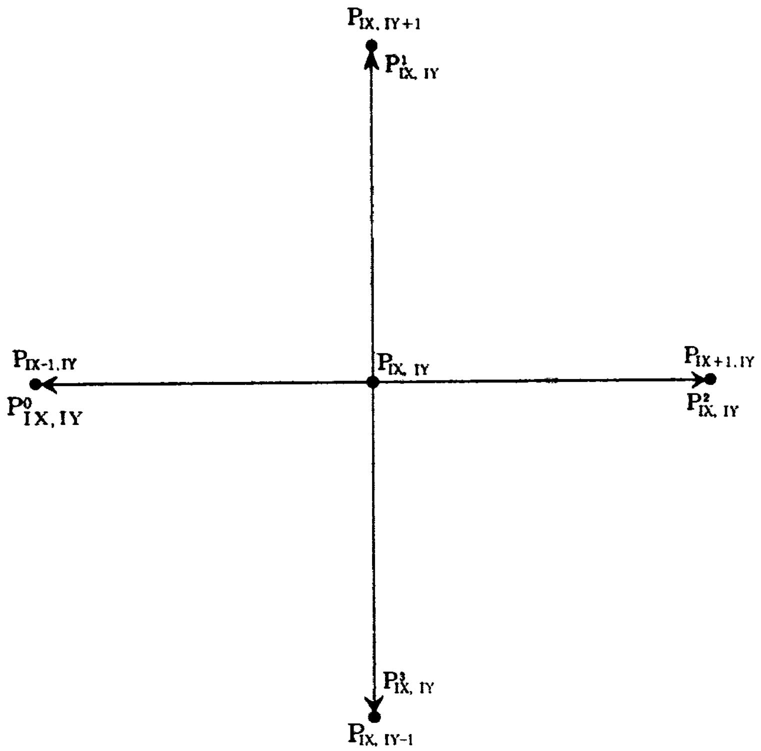

図3　隣り合った格子点

集合 A が 1) の場合，図1のようにSを含む新たな閉集合S'を考え，S'での集合Aが 2) のような集合になるようにすることができる。したがって，集合Aは 2) のような集合として考えても，その一般性を失なわない。

SにおいてAとBとの境界線は，変換Tを受けた$z=c$の等高線L_iになる。集合Aの中にある格子点の集合をPA，Bの中にある格子点の集合をPBとする。PAの中でその隣り合った格子点が全てPAに含まれるような格子点を，PAから除く。するとPAの隣り合った格子点に必ず一つは集合PBに属する格子点が存在する。隣り合ったPAとPBの格子点間には，中間値の定理[2)]

ある区間において連続なる関数 $f(p)$ が，この区間に属する点α，βにおいて相異なる値，$f(\alpha)=a$，$f(\beta)=b$を有するとき，a，bの中間にある任意の値をcとすれば，$f(p)$はα，βの中間のある点γにおいて，このcなる値を取る。すなわち

$\alpha<\gamma<\beta$，$f(\gamma)=c$

なるγが存在する。

より$z=c$の値をとるγが存在する。PAに属する全格子点について，このγの存在すべき位置を適当な近似により求める。求めたあるγの一点γ_0から，たとえば常にPAを進行方向の右側に見るように順にγを結んでゆく。こうすれば必ず出発点γ_0に戻ってくる。これは領域Sで $z=c$の等高線 L_i に近似した多角形となる。この多角形を変換Tの逆変換をすることによりS_0での等高線に近似した多角形が得られる。

4.　等高線をかかせるためのアルゴリズム

4.1　等高線のパターン

ある閉領域内で等高線をかくとき，できる等高線は

1)　その領域内で閉じた等高線

2)　領域の境界から出発して境界で終る等高線

の二つのパターンに分けられる。さらにそれぞれ図2のように分けることができる。どのような等高線でもこの四つのパターンのいずれかである。図2で斜線の領域は，その内部の点での値が外の部分の点の値より大きな値をもつ領域である。

4.2　等高線の近似線を作る手順

ここでは，図2の1，2のように等高線が領域内で閉じている場合について考える。図2の3，4の場合も 3. で述べたように閉じた等高線とみなすことができる。また領域は変換Tを受けサンプル点が格子点で表わされているものとする。格子点の一点 $p_{IX,IY}$の隣り合った点とは，図3に示すように$p_{IX-1,IY}$，p_{IX}，

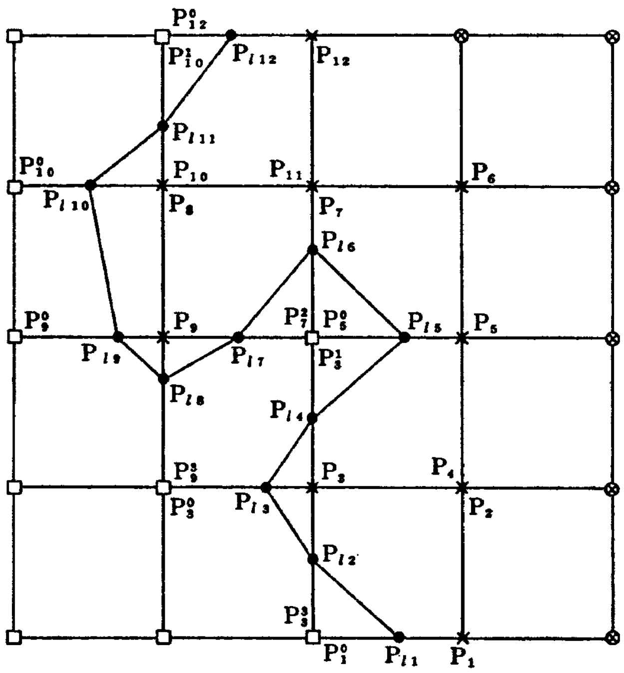

図 4　処理過程の一例

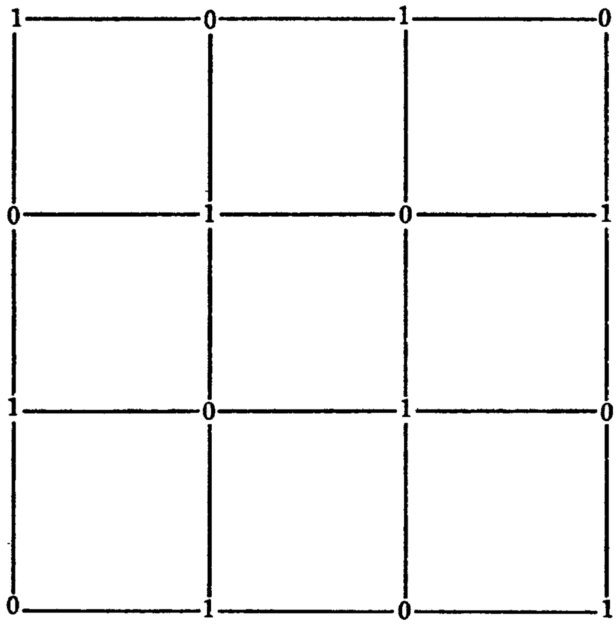

図 5　悪いデータの一例

$_{IY+1}$, $p_{IX+1,IY}$, $p_{IX,IY-1}$ の 4 点である。ここでpの隣り合った点をp^mとあらわす。mを 4 を法とした整数とし

$m=0$ のときは p の左の点

$m=1$ のときに p の上の点

$m=2$ のときは p の右の点

$m=3$ のときは p の下の点

とする。mを 1 づつ増加させると p を中心として時計方向に回転することになる。

さて，集合PAの一点，P_1をとる。P_1^0, P_1^1, P_1^2, P_1^3の少くなくても一つは集合PBに属す。いまなんらかの方法でこれらの点のうちPBに属する点P_1^kを見つける。P_1とP_1^kの間を等高線が通る。そこで

処理1；$\overline{p_n p_n^k}$ と等高線の交わる点を直線近似で求める。この点をp_lとする。

以下つぎの処理により等高線の近似線を作る。

処理2；k に 1 を加える。

処理3；P_n^kがPBに属するかどうかを調らべる。PB

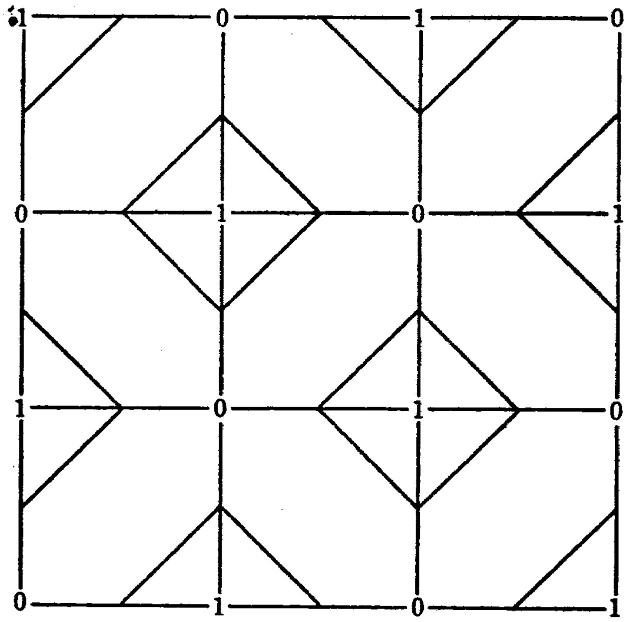

図 6　図 5 のデータでz=0.5の等高線をかかかせた結果

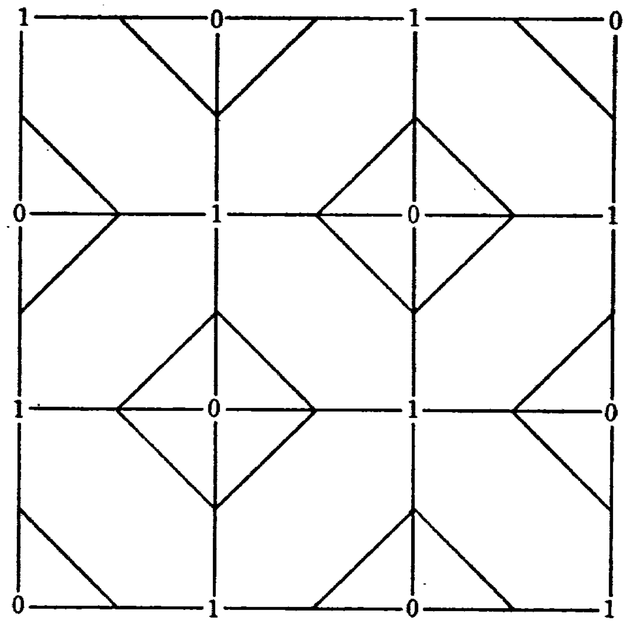

図 7　集合Aでの値を $z=c$ より小さくしたときの z=0.5の等高線

に属するときは処理 1 へ戻る。

処理4；処理 3 で P^kが PA に属するときはP^kを新たにP_{n+1}とする。P_{n+1}をP_nとかく。

処理5；P_n から P_{n-1}を見たとき，P_{n-1}が$P_n^{k'}$ であれば$k=k'+1$ とし処理 3 へ戻る。

以上の処理により得られる点列 P_{l1}, P_{l2}, ……を結び等高線の近似線を作る。図 4 にこの処理の一例を示す。図で×印の格子点は集合 PA に属する点。⊗は集合Aに属するが集合PAではない点。□は集合PBに属する点である。・は計算されたP_lの点である。

図 5 のように格子点に与えられたデータが等高線を描くのに十分でない場合，上で述べた処理を行なってz=0.5 の等高線をかくと図 6 のようになる。

ここでは集合Aを $z=c$ より大きな値をとる点の集合と定義したが，より小さな値をとる点の集合と定義しても同じように等高線の近似線を得ることができ

る。このとき得られる等高線は $z=c$ より大きな値をとる点の集合をAとしたとき得られる等高線とは，異なったものとなる場合がある。たとえば，図5のようなデータで $z=0.5$ の等高線を描くと図7のようになる。しかし，データが等高線を描くのに十分なものであるときは，両者はほぼ一致する。

4.3　出発点と処理の終り

等高線をかきだす最初の P_1 を出発点と名付ける。このP_1は，集合 PA に属する点であるならいずれの点も出発点になりうるが，能率良く等高線をかくためには，若干の工夫が必要である。

1)　領域の境界から出発して境界で終る等高線の場合

出発点は，当然境界上にある方が能率的である。図8の等高線L_1, L_2 の出発点を決める。まず ($LX1$, $LY1$)→(LX, $LY1$)→(LX, LY)→($LX1$, LY)→($LX1$, $LY1$) の順にこの上にある格子点を調べ，つぎのような条件を満す格子点を求める。

PAに属する点で，かつその一つ前の格子点がPBに属する点である点。

図8で斜線を引いてある領域は集合Aを示す。図8

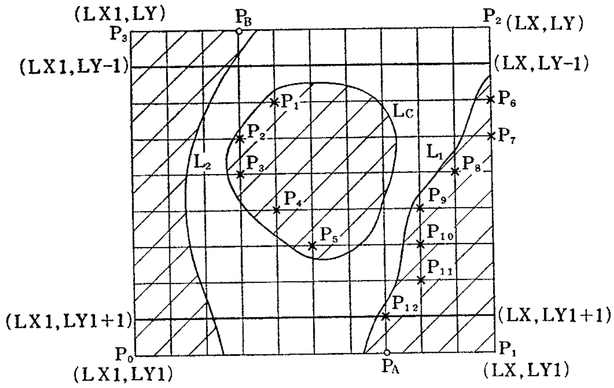

図8　出発点の決め方

でこの条件を満す点は，P_A, P_B でありこの点から出発し等高線を引いてゆく。等高線が境界上に来たとき処理を終る。

2)　閉じた等高線の場合

図8で ($LX1$, $LY1+1$), (LX, $LY1+1$), (LX, $LY-1$), ($LX1$, $LY-1$), ($LX1$, $LY1+1$) で囲まれた範囲の中で集合PAに属しその左側の点が集合PBに属する点を選び集合IAとする。図8では P_1からP_{12}までの×印の付いた点である。P_1 から P_5 までは等高線L_C, P_6から P_{12} までは等高線 L_1に関するものである。一本の等高線では出発点は一つだけでよい。したがってP_6〜P_{12}はL_1をかく過程で集合 IAから取り除いておく必要がある。また，L_C が P_1を出発点としてかかれたものであるときは，これをかく過程でP_2〜P_5をIAから取り除く。これらの点をIAから除く処理は，4.2 の処理3において $k=0$ であるとき，$P_n{}^0$ がPBに属する点であるなら P_nが IAに含まれていれているかどうかを調べ，含まれているなら IA から取り除くことにより行なう。

閉じた等高線の場合は，必ず元のところに戻ってくる。最初の点に戻ったとき処理を終る。

4.4　等高線をえがくプログラム CMLTQ

以上述べた手法で等高線を描くプログラムCMLTQを作った。このプログラムは，等高線の値が ZA から $ZMIN$ までをインターバルで W かでかせるものである。

図9にこのフロー・チャートを示す。

5.　コアにロードされているプログラムを変更するプログラム

いろいろな場合に利用できる一般性をもったプログラムを作ることは，重要なことである。しかし，広い範囲に適応できるプログラムは，プログラム自体大きくなり，その使用法も複雑になる。基本的な使用をする場合は，かえって不便になることがある。

今ある基本サブルーチンAがあり，このサブルーチンに Bの場合の処理もできるようにする。このときサブルーチンにパラメータをもうけ，たとえば $I=1$ のとき Bの場合の処理を行なうものとする。フォートランで,このようなサブルーチンは以下のようにかける。

```
      SUBROUTINE A
      ...................................
99    ...................................
      IF (I.NE.1) GŌ TŌ 100
      （Bの場合の処理）
100   ...................................
      ...................................
      RETURN
      END
```

このようにするとBの場合の処理を実行しない者にとっては，パラメータを セット しなくては ならないという煩わしさ，Bの場合を処理するために生ずるメモリーの占有領域の増大，IF 命令のむだな処理という好ましくない結果を招く。そこで，サブルーチンAは基本的なものだけにしておき，特殊な場合の処理をする場合は，基本サブルーチンAをCALLする前にすでにコアにロードされているサブルーチンAの一部を

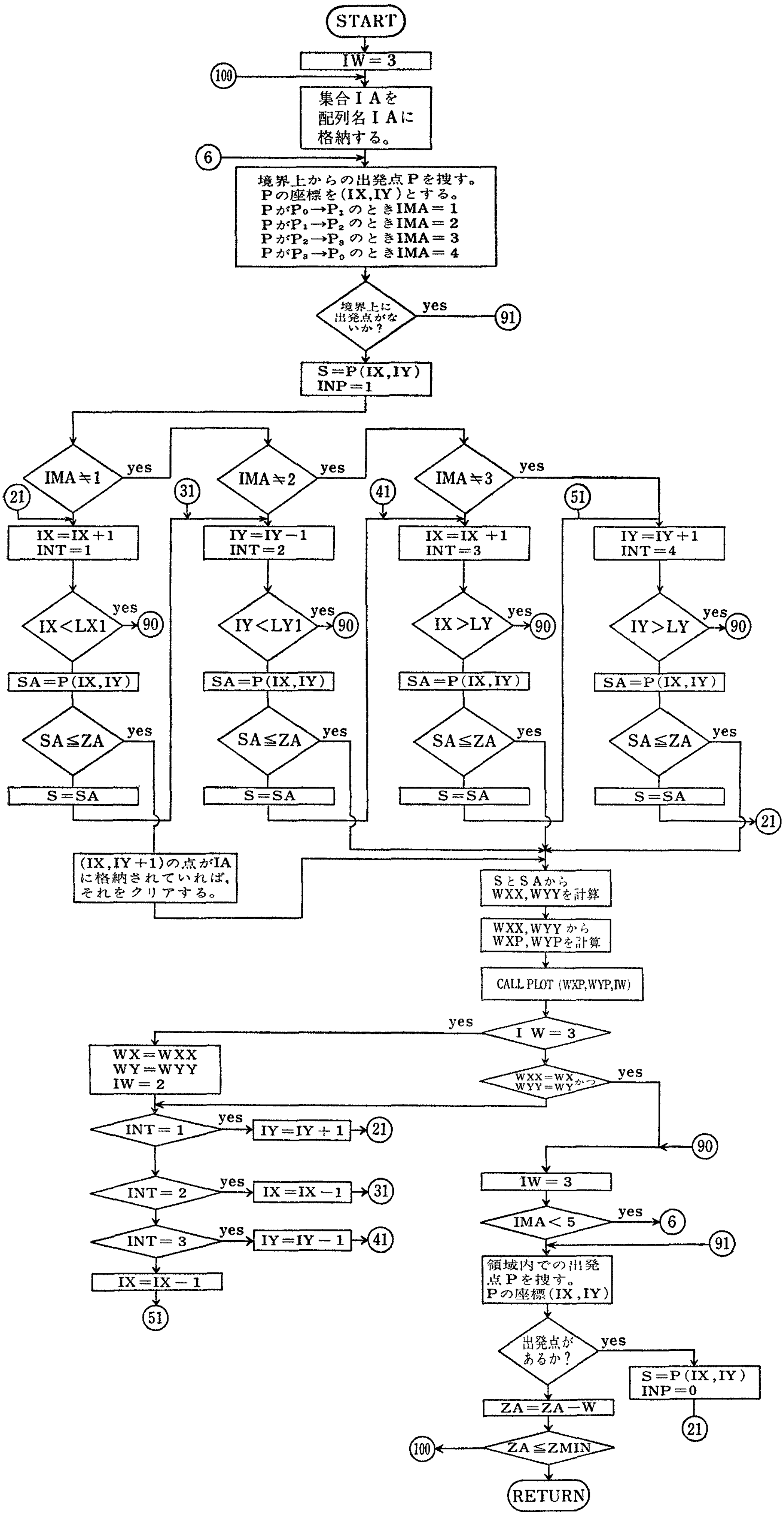
START
IW＝3
集合ＩＡを配列名ＩＡに格納する。
境界上からの出発点Ｐを捜す。
Ｐの座標を(IX,IY)とする。
ＰがP0→P1のときIMA＝1
ＰがP1→P2のときIMA＝2
ＰがP2→P3のときIMA＝3
ＰがP3→P0のときIMA＝4
境界上に出発点がないか？
yes
S＝P(IX,IY)
INP＝1
IMA≒1
IMA≒2
IMA≒3
IX＝IX＋1
INT＝1
IY＝IY－1
INT＝2
IX＝IX＋1
INT＝3
IY＝IY＋1
INT＝4
IX＜LX1
IY＜LY1
IX＞LY
IY＞LY
SA＝P(IX,IY)
SA≦ZA
S＝SA
(IX,IY＋1)の点がIAに格納されていれば、それをクリアする。
ＳとＳＡからWXX,WYYを計算
WXX,WYYからWXP,WYPを計算
CALL PLOT (WXP,WYP,IW)
I W＝3
WX＝WXX
WY＝WYY
IW＝2
WXX＝WX
WYY＝WYかつ
INT＝1
IY＝IY＋1
INT＝2
IX＝IX－1
INT＝3
IY＝IY－1
IX＝IX－1
IW＝3
IMA＜5
領域内での出発点Ｐを捜す。
Ｐの座標(IX,IY)
出発点があるか？
S＝P(IX,IY)
INP＝0
ZA＝ZA－W
ZA≦ZMIN
RETURN

図9　フローチャート

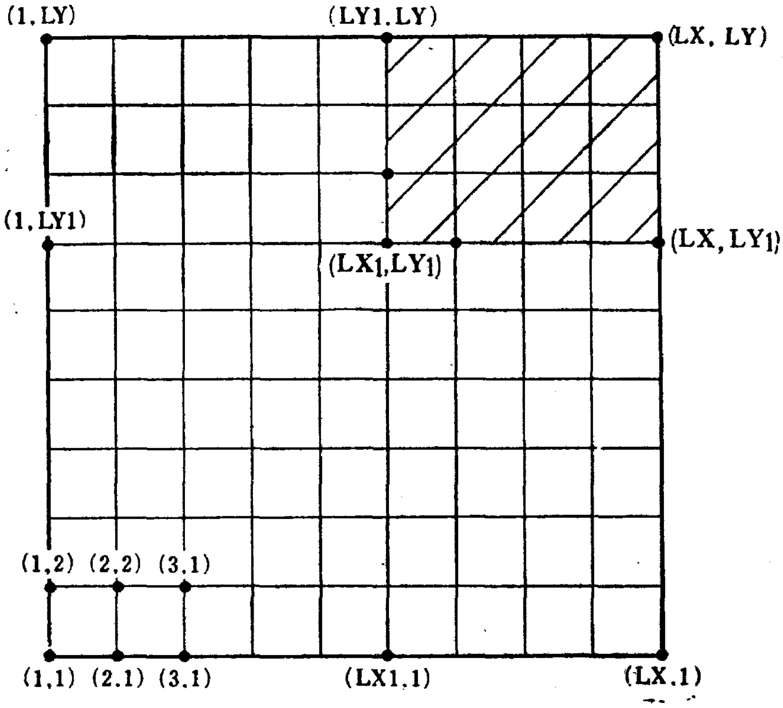

図10　xy平面 $a_x b_y \leqq x \leqq ax \cdot LX + bx, ay + by \leqq y \leqq ay \cdot LY + by$ を (1.1) から (LX, LY) までの格子点で表はす

変更するためのサブルーチンBをCALLして特殊処理を行なわせる。サブルーチンBの形式を以下に示す。

SUBROUTINE B

（サブルーチンAを変更するプログラム）

RETURN

B1 ………………………………

（Bの場合の処理）

サブルーチンAのステートメント番号 100へジャンプ

END

サブルーチンAを変更するプログラムは，サブルーチンAのステイトメント番号99の命令がロードされているコアのパターンそのものをB1 の場所移し，そこにB1 へジャンプする命令を埋めこむ。

サブルーチンBをCALLした後サブルーチンAを実行するとステイトメント番号99の位置でサブルーチンBのB1 へジャンプしBの場合の処理を行ない，処理終了後サブルーチンAのステイトメント番号 100へジャンプする。

このような形式のものにすると，サブルーチンの数が増加し，またプログラムのメンテナンスが複雑になるという欠点が生ずるが，一方基本プログラムには，なんらの修正をも施こさずにその機能の拡大，変更ができるという利点がある。処理速度も速くなる。次章で述べるサブルーチンCONVRT, BOUNDR, NŌTES等はこの形式のサブルーチンである。

なお，この形式のプログラムはフォートランではかくことができず，アセンブラ言語でかく必要がある。

6. プログラムとその使用法

このプログラムは，フォートランのサブルーチン形式でかかれたものである。プログラムは基本プログラムCMLTQと，基本プログラムの初期値のセット，機能拡張または，機能変更を行なうためのサブプログラム群で構成されている。等高線に関する基本的処理は，すべて基本プログラムで処理され，これだけで独立したプログラムである。この基本プログラムは，一般性を持たせるためにフォートランでかき，サブプログラム群は，計算機使用能率すなわち，処理速度のスピード・アップ，使用コア数を小さくするという目的から HITAC5020 アセンブラ言語 HISAP2[3] でかいた。なおこのサブプログラムは，フォートランの通常のサブルーチンと同じものであるが，その内のいくつかは，3で述べたコア・メモリーにすでにロードしてある基本プログラムの一部を修正し，その処理の手順を変更するものである。

入力データ，サブルーチンの引数は，一倍精度のデータおよび変数である。

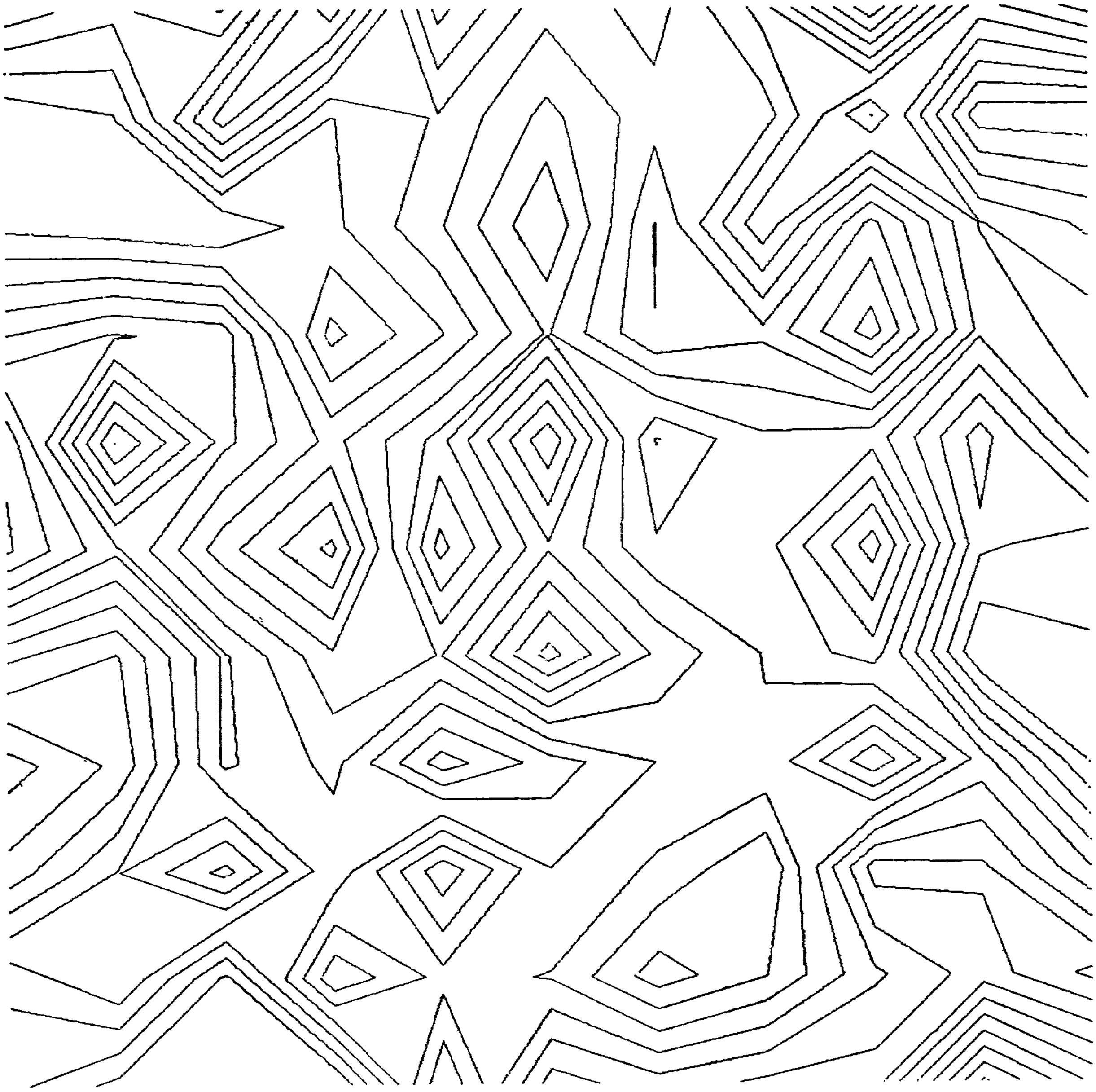

図 11

6.1 データの定義

データは実数型1倍精度2次元配列名に入れる。今2次元配列名Pの添字（IX, IY）の場所にZを入れたとする。すなわち

$P(IX,\ IY)=Z$

としたときこれは，変数関数

$z=f(x,\ y)$

において

$$\begin{cases} x=a_x \cdot IX+b_x \\ y=a_y \cdot IY+b_y \end{cases}$$

の位置の Z の値と対応する（図10)。このようにして $z=f(x,\ y)$ を配列Pに定義する。

つぎに述べるサブルーチン C$\bar{\text{O}}$NT$\bar{\text{O}}$U を使用する前に，必ず等高線をかくべき二変数関数を2次元配列に定義しておかなければならない。

6.2 サブルーチン C$\bar{\text{O}}$NT$\bar{\text{O}}$U

(1) 目的

等高線をかくプログラム。このプログラムの中で基本プログラム CMLTQ を呼びだし等高線をXYプロッタにかかせる。与えられたデータの最大値と最小値の間で10等分し，その値について等高線をかく。図は，X方向，Y 方向それぞれ10インチの大きさのものとなる。このときかいた等高線の値は，ラインプリンタに印字される。

(2) リンクの方法

CALL C$\bar{\text{O}}$NT$\bar{\text{O}}$U (P, LX, LY)

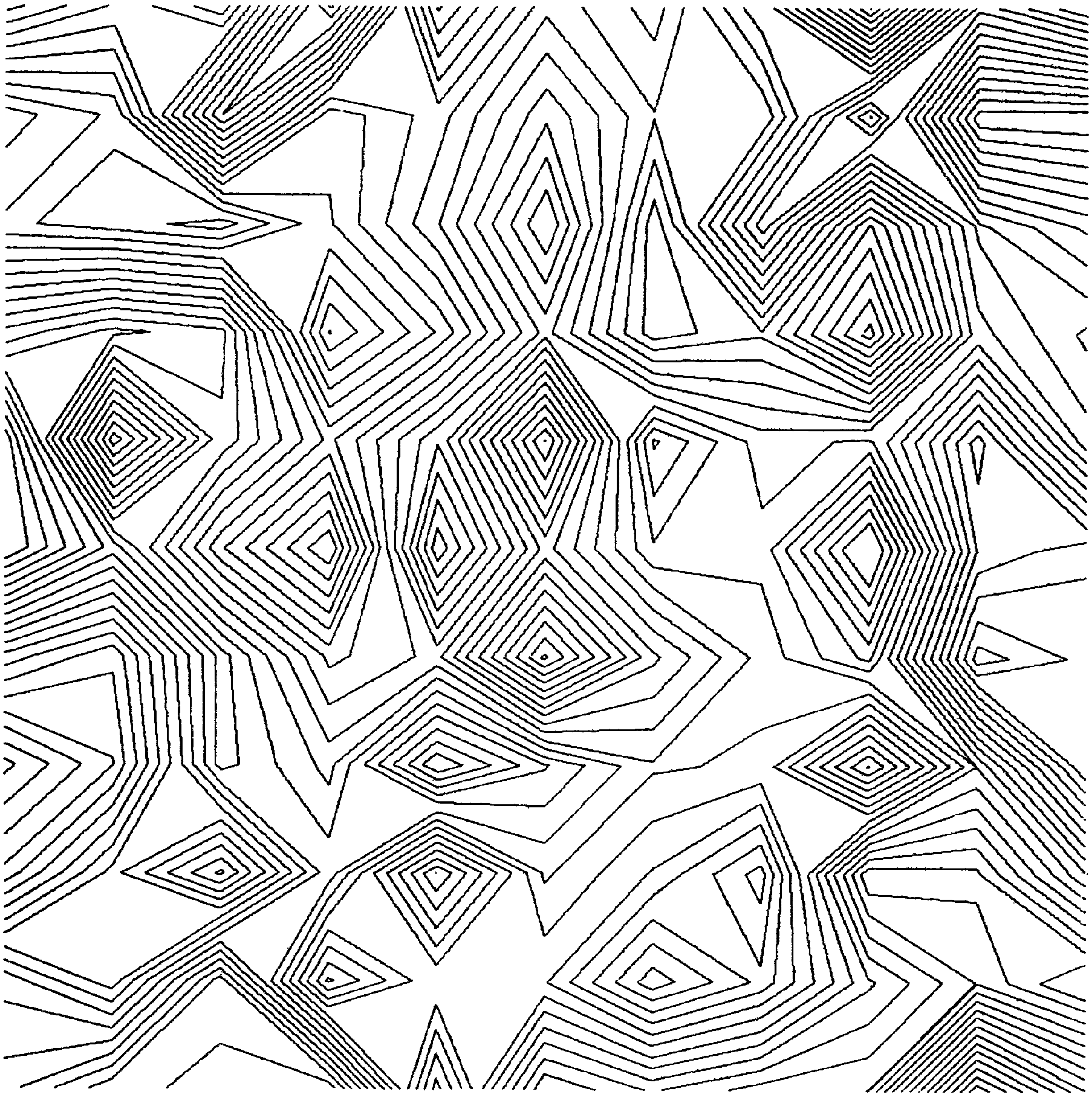

図 12

(3)　パラメータの説明

P　；実数型 2 次元配列名

かかせる等高線のデータを入れておく。

LX, *LY*　；整数型

配列名 *P* に入っているデータの大きさを指定する。配列の大きさは *P* (*LX*, *LY*) となる。*LX*, *LY* > 1

なおこのサブルーチンでは日立製作所の *XY* プロッター・ルーチン PLOTSUB 4) を呼びだしているのでこのサブルーチンを CALL する前に PLŌTS, 使用後に PLŌTV を CALL する必要がある。

(4)　使用例

配列 *P* の *P*(1,1) から *P* (11, 11) に 10.0以下の正の一様乱数を入れて等高線をかかせる。

```
DIMENSIŌNIŌ (11, 11)
(Pに一様乱数を入れる)
CALL PLŌTS (8H…………, 0.0)
CALL CŌNTŌU (P, 11, 11)
CALL PLŌTV
STŌP
END
```

この結果を図11に示す。

6.3　サブルーチン CLDATA

(1)　目的

かくべき等高線の値の最大値と最小値および等高線のインターバルを指定する。

(2) リンクの方法

CALL CLDATA (*ZMAX*, *W*, *ZMIN*)

(3) パラメータの説明

ZMAX； 実数型
かかせるべき等高線の最大の値を指定

W ； 実数型
等高線のインターバル。等高線は
$ZA=ZMAX-k\cdot W$ ($k=1$, 2……)
の値のものをかいてゆく。

ZMIN ； 実数型
*ZA*の値が*ZMIN*より小さくなると処理を終了する。$ZMIN<ZMAX$

(4) 使用例

6.2 の例で最大値 9.5から 10までをインターバル 0.5 で等高線をかかせる。

```
…………………………………………………………
CALL   CLDATA (9.5, 0.5, 1.0)
CALL   CŌNTŌU (P, 11, 11)
…………………………………………………………
```

この結果を図12に示す。

6.4 サブルーチン LDATAD

(1) 目的

サブルーチン CLDATA では等高線のインターバルが一定になる。インターバルが全く不規則な等高線をかけるようにしたのがこのサブルーチンである。

(2) リンクの方法

CALL LDATAD (D, N)

(3) パラメータの説明

D ； 実数型配列名
*D*の中にかかせる等高線の値を入れる。

N ； 整数型
*D*の中に入っているデータの数。

(4) 使用例

6.2 の例で等高線の値が 9.6, 5.3, 4.0, 3.14のものをかかせたいとき。

```
DIMENSIŌN P (11,11), D(5)
D(1)=9.6
D(2)=5.3
D(3)=4.0
D(4)=3.14
…………………………  ……………………………
CALL LDATAD (D, 5)
CALL CŌNTŌU (P, 11, 11)
…………………………………………………………
```

6.5 サブルーチン SIZE

(1) 目的

XY プロッタでかかせる図の大きさを任意のものにするためのサブルーチンである。

(2) リンクの方法

CALL SIZE (*XO*, *YO*, WIDTH, HIGHT)

(3) パラメータの説明

XO, *YO* ； 実数型
P (1, 1)の位地を*XY*プロッタ上の(*XO*, *YO*) に定義する。

WIDTH; 実数型
*X*軸方向の展開巾を定義する。*P*(*LX*, 1) は, *XY*プロッタ上の (*XO*+WIDTH, *YO*) 上に位置する。

HIGHT; 実数型
Y 軸方向の展開巾を定義する。 *P* (1, *LY*) は, *XY*プロッタ上の (*XO*, *YO*+HIGHT) 上に位置する。

(4) 使用例

$z=x^2+y^2$, $-10\leqq x\leqq 10$, $-10\leqq y\leqq 10$
を配列 *P* に定義し, SIZE を使って等高線をかかせる。

```
DIMENSIŌN P (81, 81)
```
($z=x^2+y^2$を配列 *P* (1, 1) から *P* (81, 81) へ定義する。)

```
CALL PLŌTS (8H…………, 0.0)
CALL SIZE (0.0, 0.0, 4.0, 4.0)
CALL CŌNTŌU (P, 81, 81)
CALL SIZE (0.0, 5.0, 10.0, 5.0)
CALL CŌNTŌU (P, 81, 81)
CALL PLŌTV
STŌP
END
```

この結果を図13に示す。

このサブルーチンを使って一枚の図をいくつかに分割してかくことができる。一度にコアにロードできないような大きなデータについても処理が可能になる。

6.6 サブルーチン PŌRIG

(1) 目的

データの入っている配列*P* の処理範囲を変るためのサブルーチンである。

(2) リンクの方法

CALL PŌRIG (*L*×1, *LY* 1)

(3) パラメータの説明

LX 1, *LY* 1 ； 整数型
処理する配列*P* の範囲を図10の斜線部す

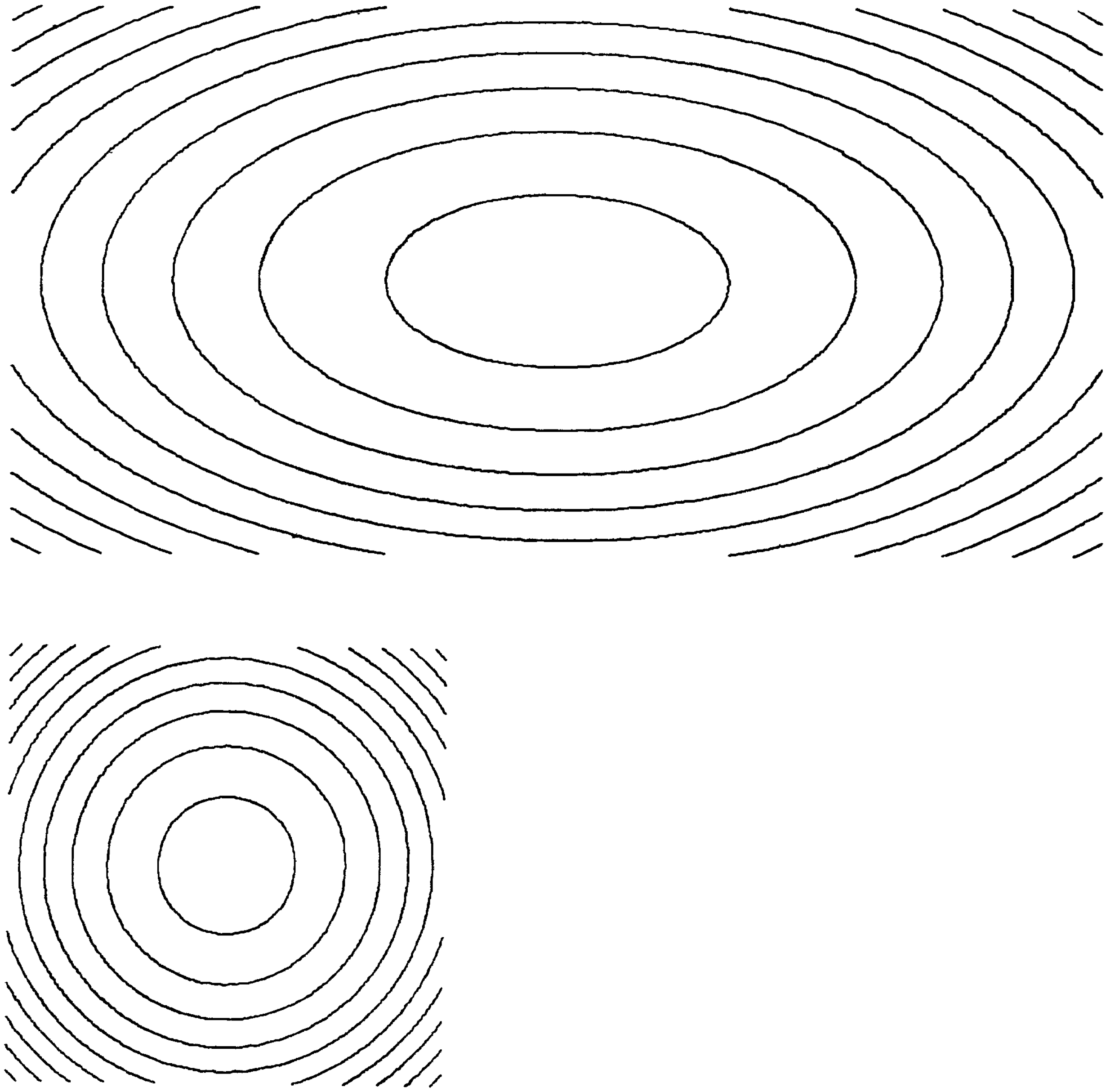

図 13

なわち$P(LX1, LY1)$, $P(LX, LY1)$ $P(LX, LY)$ で囲まれた範囲のみに限る。

(4) 使用例

6.5 での例のデータで $LX1=40$, $LY1=40$ にして等高線をかかせる。

```
..............................................................
CALL PŌRIG (40, 40)
CALL CŌNTŌU (P, 81, 81)
..............................................................
```

この結果を図14に示す。

6.7 サブルーチン WRFŌM

(1) 目的

等高線一本かくとラインプリンタにその等高線の出発点$P(I, J)$ のI, J および$P(I, J)$ の値と実際の線の出発点 P_{Is}と終着点P_{Ie} の位置が印字される。P_{Is}, P_{Ie} の位置を配列名の位置であらわすか，実際のプロッター上の位置であらわすかの選択をする。

(2) リンクの方法

CALL WRFŌM (I)

(3) パラメータの説明

I ; 整数型

I＝－1のとき 印字を省略する。

I＝0のとき プロッタ上の位置で印字。

I＝1のとき 配列名の位置で印字。

6.8 サブルーチン LŌMIT

(1) 目的

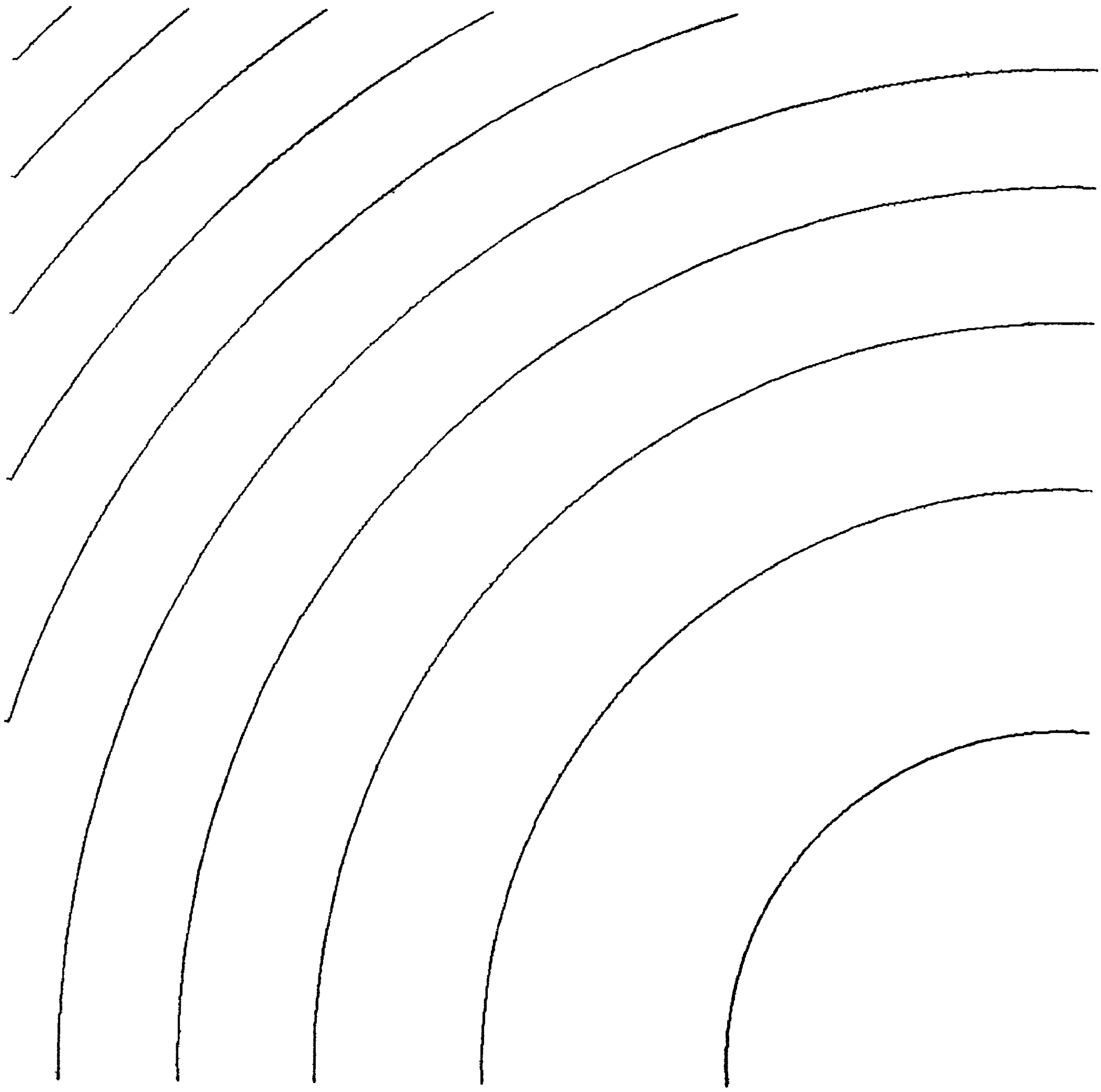

図 14

閉じた等高線あるいは閉じない等高線を省略して，XY プロッタにかかせないためのものである。

(2) リンクの方法

CALL LŌMIT（I）

(3) パラメータの説明

I ; 整数型

I＝1のとき閉じる等高線を省略する。

I＝2のとき閉じていない等高線を省略する。

1が1，2以外のとき元の状態にもどす。

(4) 使用例

6.2 の例で I＝1, I＝2 のときの等高線をかく。

……………………………………………………………

CALL LŌMIT（1）

CALL CŌNTŌU (*P*, 11, 11)

CALL PLŌT (12, 0, 0.0, −3)

CALL LŌMIT（0）

CALL LŌMIT（2）

CALL CŌNTŌU (*P*, 11, 11)

……………………………………………………………

I＝1のときの結果を図15に，I＝2のときの結果を図16に示す。

6.9 サブルーチン CŌNVRT

(1) 目的

座標変換をするためのサブルーチン。基本ルーチンでは xy 平面は直角座標であるが，このサブルーチンにより重角座標から他の座標に変換を行なうことができる。

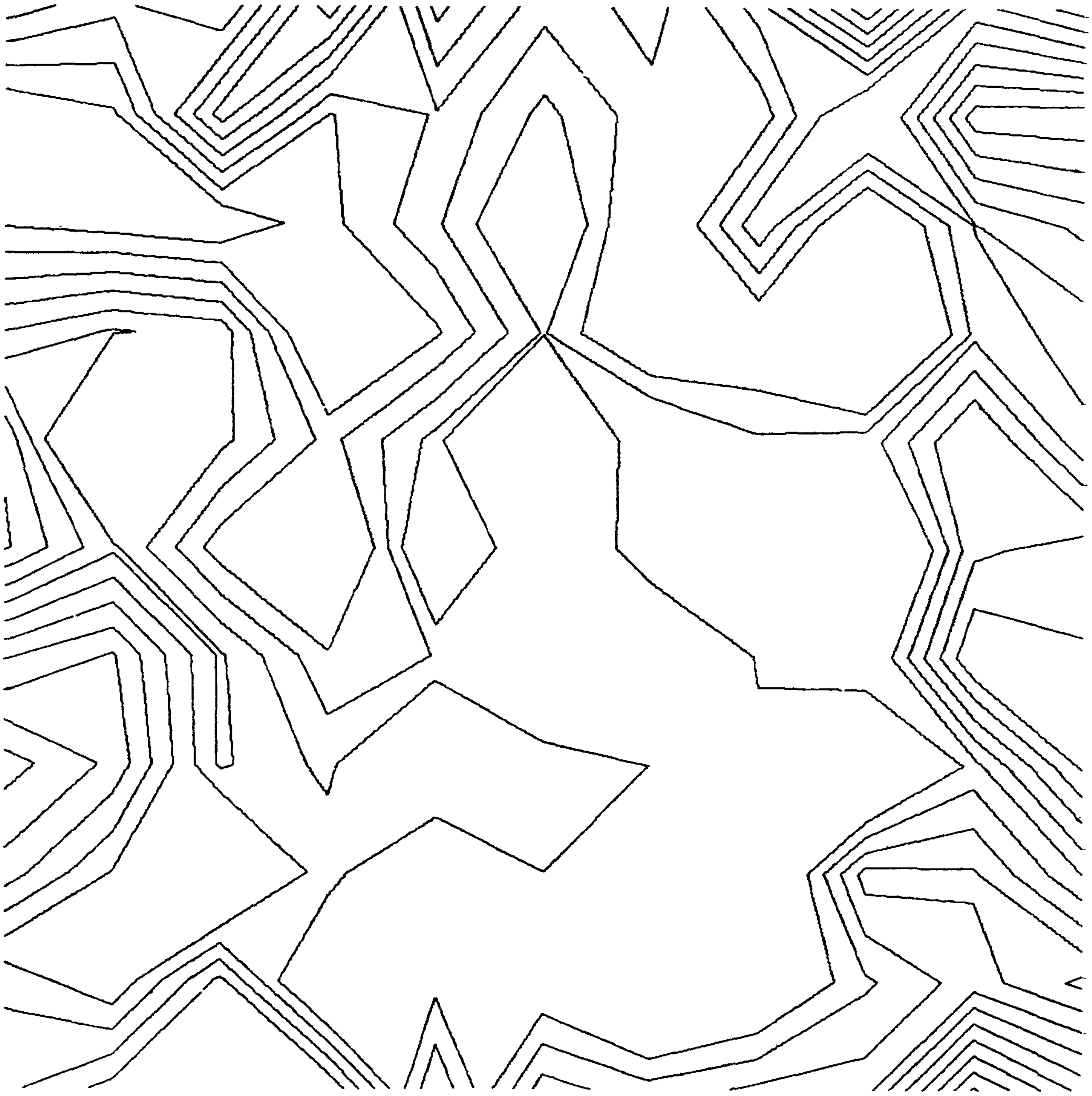

図 15

(2) リンクの方法

CALL CŌNVRT (FUNC)

(2) パラメータの説明

FUNC; サブルーチン名

直角座標系から他の座標系への変換計算を行なうサブルーチンでつぎのような形式のもの。

SUBRŌUTINE FUNC (*X*, *Y*)

*X*1=*X*

*Y*1=*Y*

X=*f* (*X*1, *Y*1)

Y=*g* (*X*1, *Y*1)

RETURN

END

引数に整数型実変数で1をセットすると元の状態にもどる。

(4) 使用例

6.5 の例のデータで $x=X\sin Y$, $y=X\cos Y$ の変換を行なった平面で等高線をかかせる。

....

EXTERNAL FUNC

CALL CŌNVRT (FUNC)

CALL SIZE (0.0, 0.0, 5.0, 6.283)

CALL CŌNTŌU (*P*, 81, 81)

....

SUBRŌUTINE FUNC (*X*, *Y*)

R=*X*

T=*Y*

X=*R* * SIN(*T*)+5.0

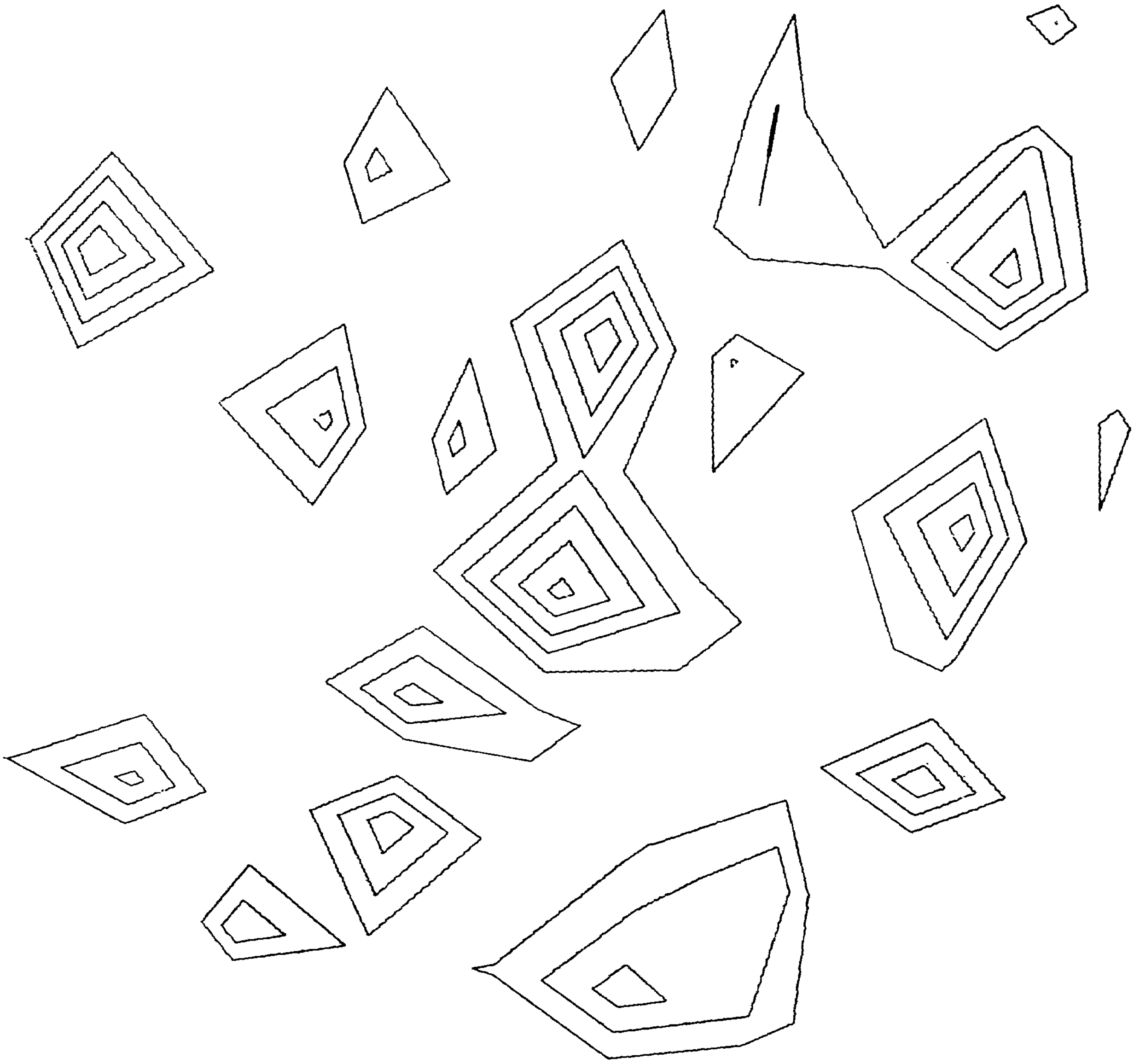

図 16

$Y=R*C\bar{O}S(T)+5.0$

RETURN

END

なおFUNCの入力引数X，Yの範囲はSIZEできめるXとYの範囲である。この結果を図17に示す。

6.10 サブルーチン BŌUNDR

(1) 目的

図18のように任意の一価の関数 $y=f(x)$ あるいは $x=f(y)$ を一つの境界線として等高線をかかせる。

(2) リンクの方法

CALL BŌUNDR (FUNC, I)

(3) パラメータの説明

FUNC; 関数名

　ステート・メント関数，または関数サブプログラムで定義する。関数サブプログラムにしたときは，BŌUNDR をCALLする前に関数サブプログラム名をタイプ・ステートメント EXTERNAL で定義しておく。

I　　; I＝1のとき　図18-1で(1)の領域で等高線をかく。

I＝2のとき　図18-1で(2)の領域で等高線をかく。

I＝3のとき図18-2で(3)の領域で等高線をかく。

I＝4のとき　図18-2で(4)の領域で等高線をかく。

I＝0のとき　プログラムを元の状態にもどす。

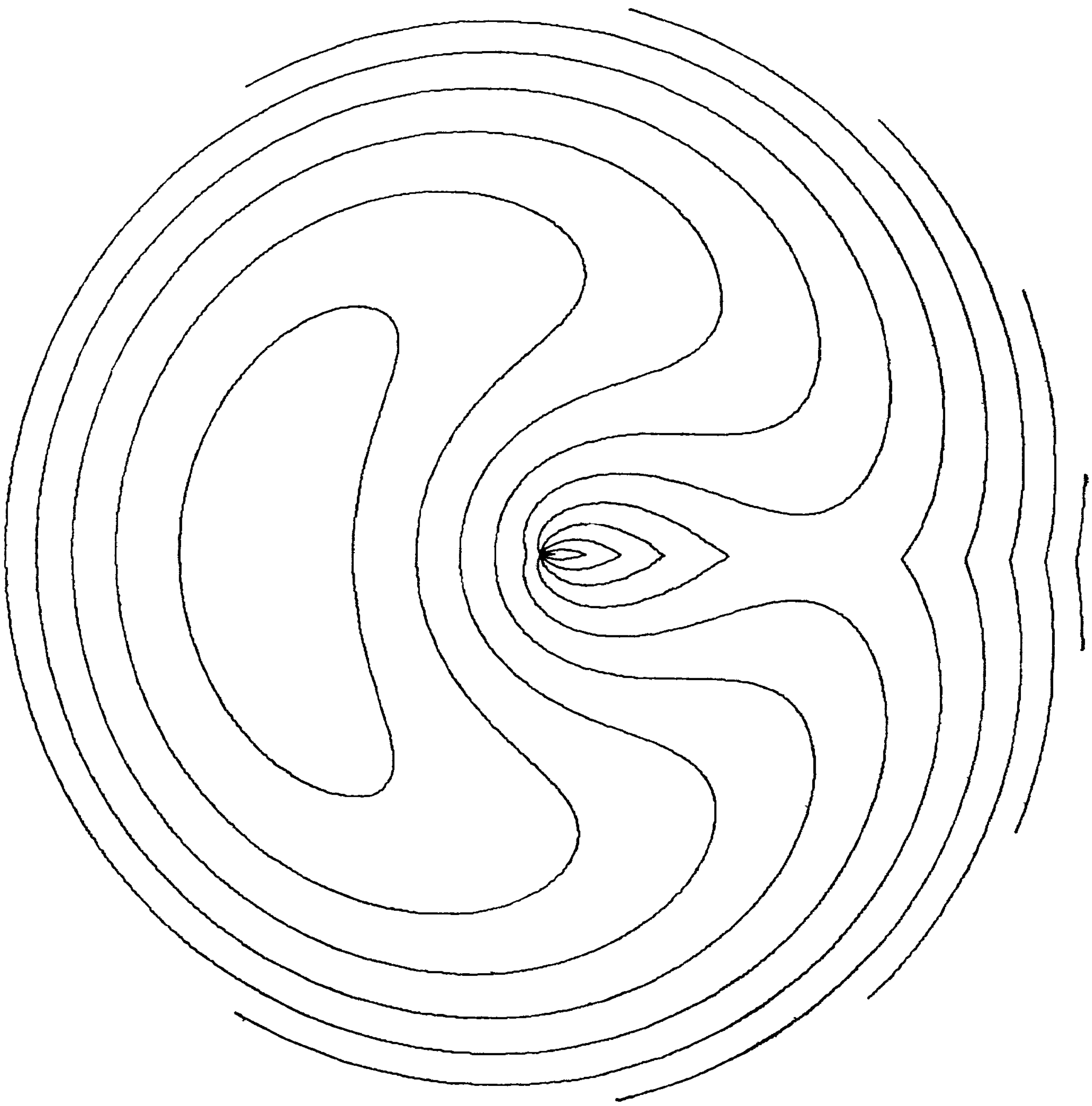

図 17

なお，ここで定義する関数のx，yはプロッタ上に実際にかかる座標系のものであり単位はインチである。

(4) 使用例

の例で $Y \leqq$FUNC (X) の範囲で等高線をかかせる。ただし

FUNC(X)$=X+2$

……………………………………………………………

FUNC(X)$=X+2.0$
CALL CLDATA (9.0, 0.5, 1.0)
CALL BŌUNDR (FUNC, 1)
CALL CŌNTŌU (P, 11, 11)

……………………………………………………………

6.11 サブルーチン NŌTESI

(1) 目的

域内で閉じた等高線の値を図の中にかかせる。

(2) リンクの方法

CALL NŌTESI (K, THETA)

(3) パラメータの説明

K　　; 整数型

$ZA=ZMAX-K\cdot W(N-1)$ の値の等高線について数値をかき込む。

ただし$N=1$, 2, …………

THETA; 実数型

かかれる数字とX軸のなす角度

かかれる数値の大きさは0.07インチである。

6.12 サブルーチン NŌTES

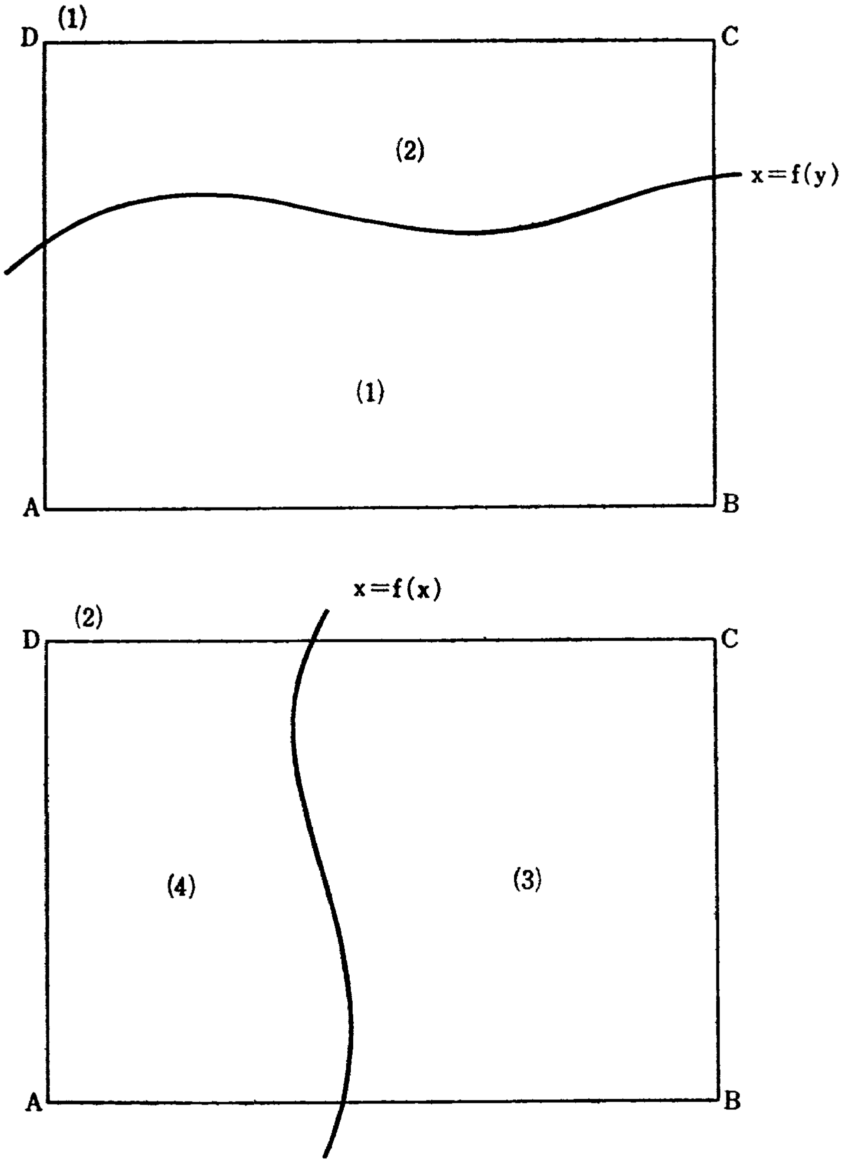

図18　$y=f(x)$ あるいは$x=f(y)$ の境界線

(1)　目的

領域の境界線上から出発する等高線の値を図の周囲にかく。

(2)　リンクの方法

CALL NŌTES (*IX*, *IY*, *K*, HIGH)

(3)　パラメータの説明

IX　；　整数型

$IX=1$ のとき図18の$\overline{AB}$に沿って数値をかく。

$IX=2$ のとき図18の$\overline{CD}$に沿って数値をかく。

$IX=3$ のとき図18の$\overline{AB}$，$\overline{CD}$に沿って数値をかく。

*IX*が1，2，3 以外のとき$\overline{AB}$，$\overline{CD}$のいずれにもかかない。

IY　；　整数型

$IY=1$ のとき図18の$\overline{DA}$に沿って数値をかく。

$IY=2$ のとき図18の$\overline{BC}$に沿って数値をかく。

$IY=3$ のとき図18の$\overline{DA}$，$\overline{BC}$に沿って数値をかく。

*IY*が1，2，3以外のとき$\overline{BC}$，$\overline{DA}$のいずれにもかかない。

K　；　整数型

サブルーチンNŌTESIの*K*と同じ。

HIGH;　かかれる数値の大きさを指定する。

(4)　使用例

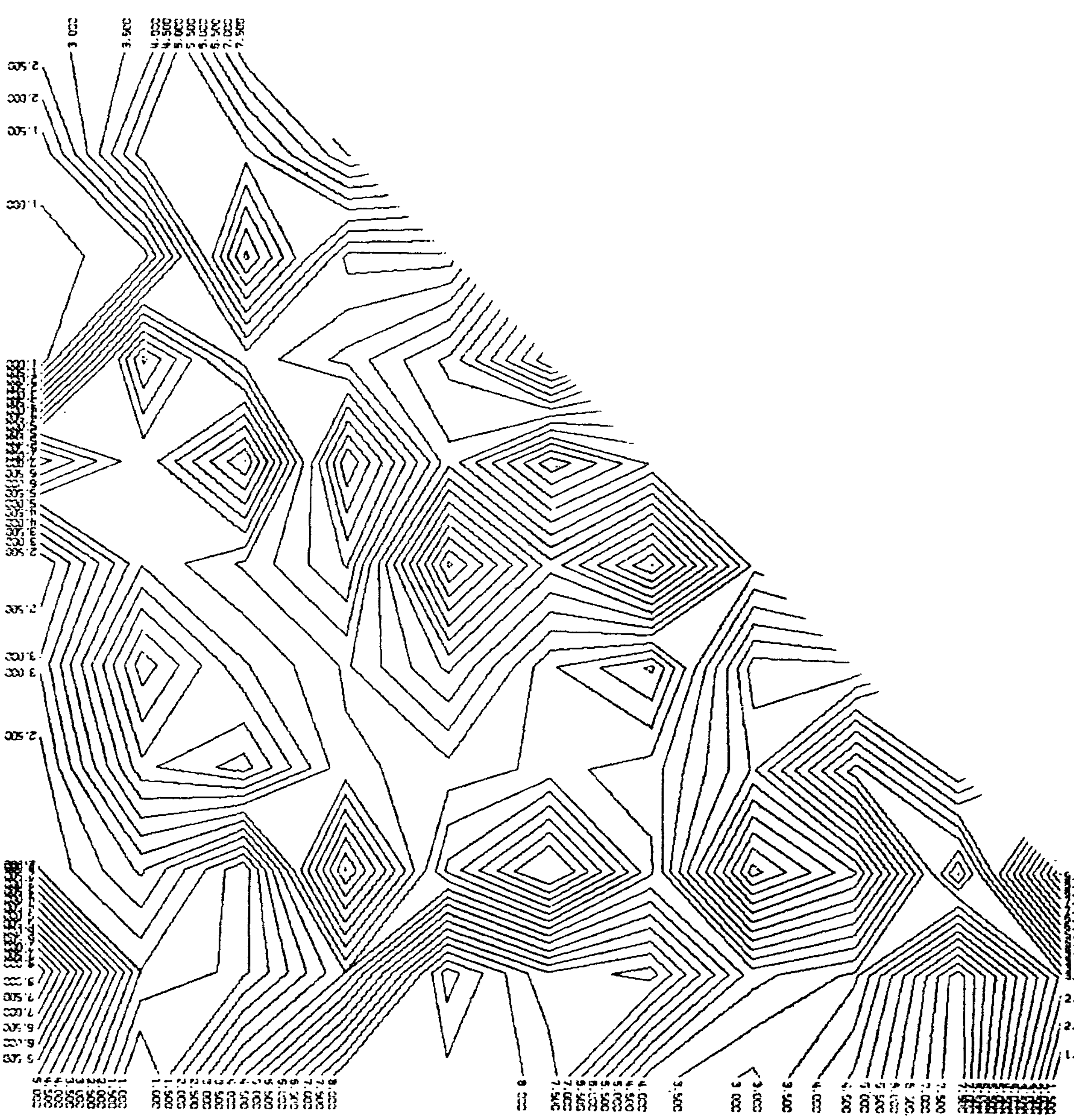

図 19

6.10の例で図の周囲に等高線の値をかく。次のカードを6.10の例のCALL CŌNTŌUの前に入れる。

CALL NŌTES (3, 3, 1, 0.15)

この結果を図19に示す。

7. 使　用　例

〔例 1〕

図20は，空気力学第1部の安喜隆幸技官の計算された「物体まわりの衝撃波回折」の計算結果の一例で円柱のまわりの等密度線を描いた図である。

〔例 2〕

図21は，計測部の石黒登美子技官の計算された「ノズル流」の計算結果の一例で密度分布線を描いた図である。

〔例 3〕

図22は，空気力学第1部の井上健二技官，計測部の相波清子技官の計算された「傾いた楕円のまわりの流れ」の計算結果の一例である。

8. む　す　び

等高線をかかせるためのアルゴリズムを考察し，実際にプログラムを作成した。そして，次のような結果を得た。

(1) Fortran の2次元配列名に格納されたデータを簡単に等高線の形式で図に表わすことができる。

(2) 2次元平面でその周囲の状態を考慮しながらあ

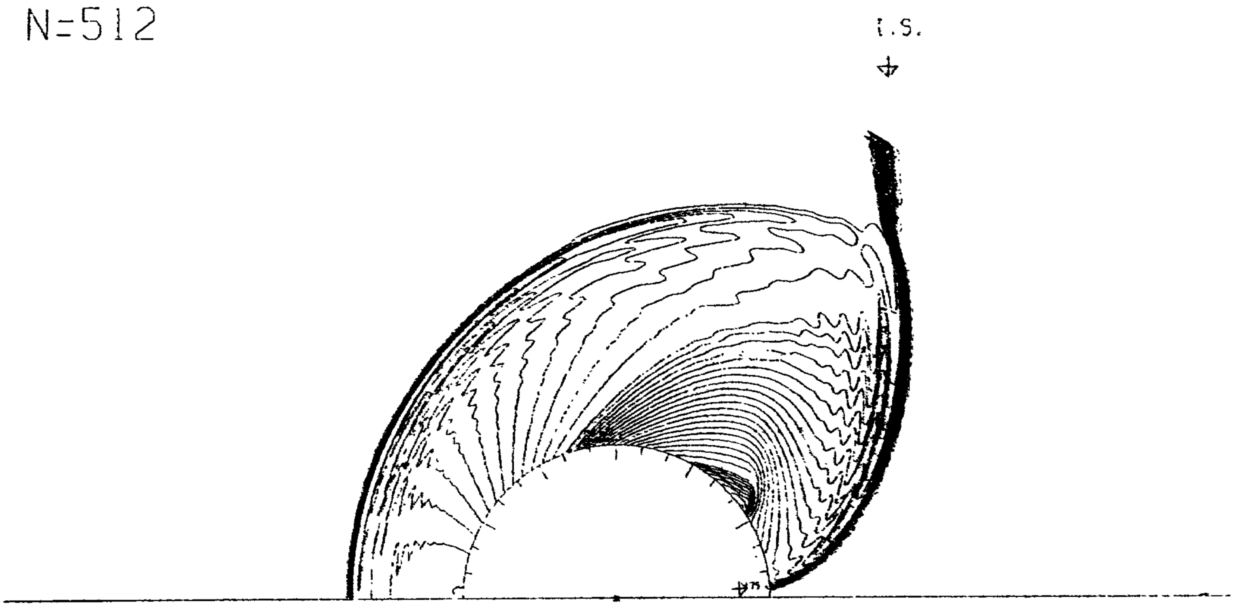

図 20

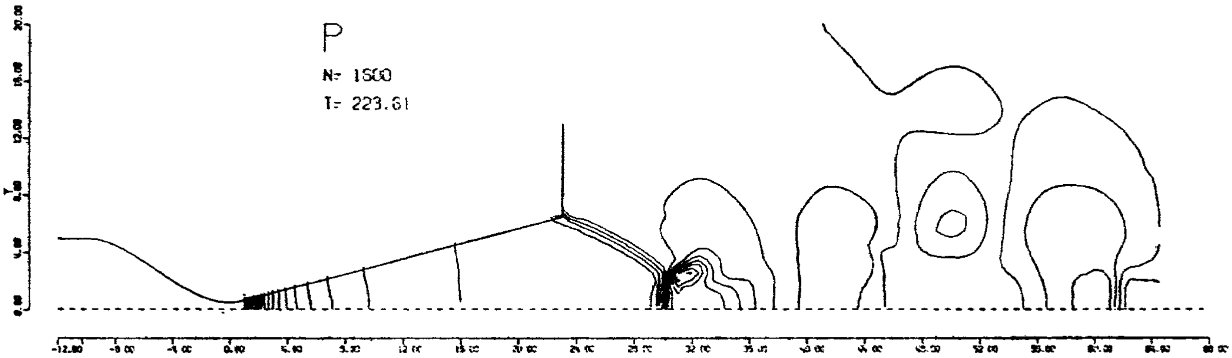

図 21

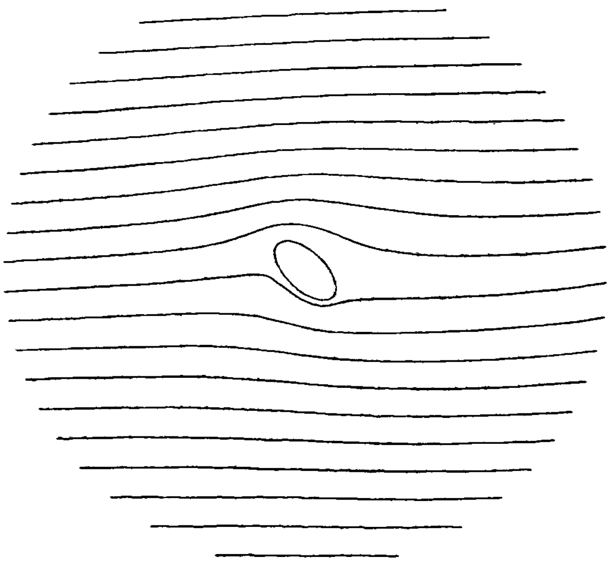

図 22

る処理を行なう一手法を考えた。この手法は等高線を描く問題以外，2次元でのパターン認識等で利用できる可能性がある。

(3) 既にコアにロードされたプログラムを実行中に修正する形式のプログラムは問題によっては非常に有効な手段である。

参 考 文 献

1) M. O. Dayhoff; A Contour-Map Program for X-Ray Crystallogphy, Communications of the ACM. Vol. No. 10 (1963) pp. 620-623
2) 高木貞治，解析概論，(1961) pp 26，岩波書店
3) 日立製作所；HITAC 5020, 5002 E/F アセンブリ・システム (HISAP 2)，(昭和44年)
4) 日立製作所；HITAC 5020X-Y プロッタ・ルーチン (昭和43年)

付録

1. サブルーチン CMLTC

(1) 目的

等高線をかく基本プログラムである。

(2) リンクの方法

CALL CMLTQ (*P, JX, JY, IA, IAMAX*)

(3) パラメータの説明

P ; 実数型2次元配列名
かかせる等高線のデータを入れておく。

JX, *JX* ; 整数型
配列名*P*に入っているデータの大きさと指定する。配列の大きさは*P*(*JX, JY*)となる。

IA ; 整数型1次元配列名
作業用エリア

IAMAX; 整数型
*IA*の大きさを指定。普通500ぐらいのエリアを確保してあれば十分。

さらに CŌMMŌN/CMLTQI/ として以下の11個のパラメータの値をセットする必要がある。

LX 1, *LY* 1 ; 整教型
処理する配列*P*の範囲を*P*(*LX*1, *LY*1)～*P* (*LX, LY*) と指定する。(6.6参照)

ICMAX; 整数型
かかれる等高線の数を指定する。ただし等しい値の等高線は1本とする。

IWRITE; 印字のフォーマットを指定する。(6.7参照)

XŌ, YŌ ; 実数型
XY プロッタ上の原点の指定をする。(6.5 参照)

Z ; 実数型
かかせるべき等高線の最大の値を指定する (6.3 参照)

W ; 実数型
等高線のインターバルを指定する。(6.3 参照)

ZMIN; 実数型
等高線の値の下限を指定する。(6.3 参照)

FCX, *FCY* ; 実数型
格子上の位置を*XY*プロッター上の位置へ変換するための係数。プロッター上の *XY* 座標 (*WXP, WYP*) は *WXP*=(*WXX*+*FLX*1) *FCX*+*X*Ō
WYP= (*WYY*+*FLY*1) *FCY*+*Y*Ō
で計算される。

(4) プログラム中の変数名

LX ; *JX*と等しい
LY ; *JY* と等しい
FLX 1 ; *LX* 1 の実数化したもの
FLY 1 ; *LY* 1 の実数化したもの
ZA ; 処理中の等高線の値
ICŌUNT; 等高線処理の回数，*ZA*=*Z*のときは0で*ZA*から*W*を引くごとに1増加する。
IW ; サブルーチン LŌT のパラメータ
IMB ; 等高線の出発点を捜す処理におけるパラメータ。
IMA ; 等高線出発点の状態を示すパラメータ。
IX, *IY* ; 処理をしている格子点位置。
S ; 基準点の値
SA ; *P* (*IX, IY*) の値
INP ; 閉じた等高線のときは0，開いた等高線のときは1になる。
INT ; 処理の状態を示すパラメータ
WXX, *WYY* ; 格子上での等高線の通る座標点
WX, *WY* ; 閉じた等高線の最初に計算された座標点
WXP, *WYP* ; 実際に *XY*プロッタ上にかかれる等高線の座標
XY ; (*ZA*−*SA*) / (*S*−*SA*)
M ; 配列 *IA* の添字

以上の変数のうち次の22個はCŌMMŌN/CMLTQŌ/として外部から参照可能になっている。

デック名	サブルーチン名	サブルーチンの内容	備　　考
CONTOUR	CONTOU	ここで基本サブルーチンCMLTQのパラメータをセットしCMLTQをCALLする。	サブルーチンCMLTQをCALLする。
CMLTQ	CMLTQ	等高線をかく基本サブルーチン	サブルーチンPLOTをCALLする。
CMLTQ 1	CLDATA	等高線の最大値、最小値、インターバルを指示する。	初期値をセットするための補助サブルーチンの集まり。(HISAP2)
	SIZE	図の大きさを指示する。	
	PORIG	データ格納配列の処理範囲を変更する。	
	WRFOM	処理状況の印字のフォーマットを指定する。	
CMLTQ 2	LOMIT	閉じた等高線は又は閉じていない等高線を省略する。	CMLTQを変更し機能を変えるための補助サブルーチンの集まり。(HISAP2)
	LDATAD	等高線の値を配列で定義する。	
	CONVRT	直角座標系から他の座標系へ交換する。	
BOUNDR	BOUNDR	任意の境界を持った図にする。	
NOTES	NOTESI	閉じた等高線の値を図の中にかかせる。	
	NOTES	境界から出発する等高線の値をかかせる。	

表　1

ICŌUNT, IW, INP, INT, IMA, M, IX, IY, FLX 1, *FLY* 1, *ZA, WXX, WYY, WXP, WYP, WXY, S, XY, LX, LY, WX, WY*

(5) 必要とするサブルーチン

サブルーチン PLŌT が必要。この CMLTQ では CALL PLŌT (*WXP, WYP, IW*) という型で PLLŌT を CALL している。

2. 各サブルーチンの関係

サブルーチンは，次の6つのデックに納められている。

(1) CŌNTŌUR
(2) CMLTQ
(3) CMLTQ 1
(4) CMLTQ 2
(5) BŌUNDR
(6) NŌTES

各デックに入っているサブルーチンは表1の通りである。

サブルーチン CŌNTŌU は，その中で CMLTQ を呼びだし等高線をかく。補助サブルーチンのうち CLDATA, LDATAD, SIZE, PŌRIGは，CŌNTŌU を一度呼びだすと以後それらのサブルーチンで指定した事項は無効になる。したがって，数度 CŌNTŌU を呼びだす必要があるときは，そのつどこれらのサブルーチンでの指定を行なう。

よ各サブルーチンの関係を図23に示す。

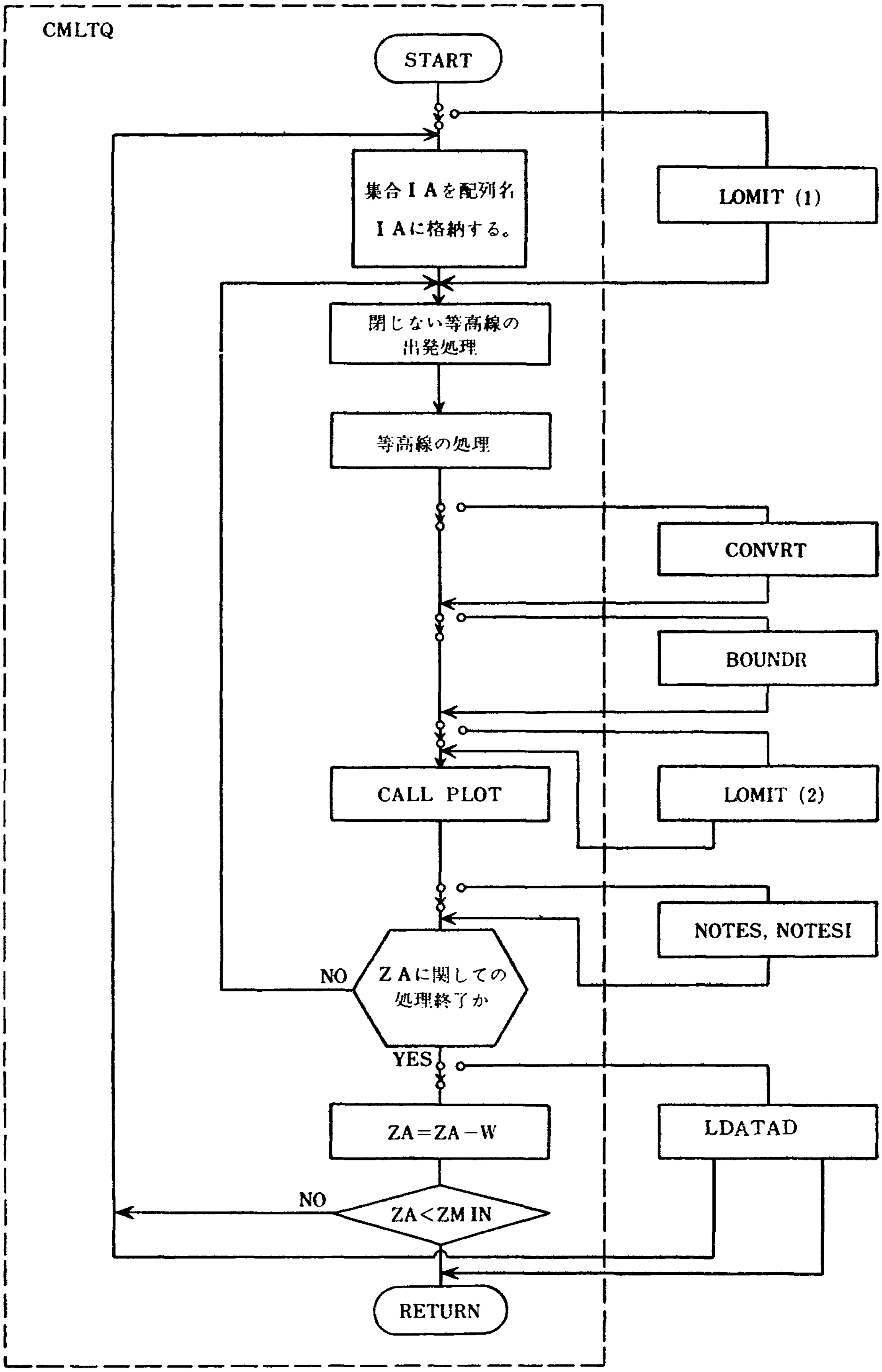
CMLTQ
START
集合ＩＡを配列名
ＩＡに格納する。
LOMIT (1)
閉じない等高線の
出発処理
等高線の処理
CONVRT
BOUNDR
LOMIT (2)
CALL PLOT
NOTES, NOTESI
NO
ＺＡに関しての
処理終了か
YES
ZA＝ZA－W
LDATAD
NO
ZA＜ZM IN
RETURN

図 23

3. CŌNTŌUのリスト

```
      SUBROUTINE CMLTQ(P,JX,JY,IA,IAMAX)
      COMMON/CMLTQI/LX1,LY1,ICMAX,IWRITE,XO,YO,Z,W,ZMIN,FCX,FCY
      COMMON/CMLTQO/ICOUNT,IW,INP,INT,IMA,M,IX,IY,FLX1,FLY1,ZA,WXX,WYY,
     C      WXP,WYP,WXY,S,XY,LX,LY, WX,WY
      DIMENSION P(LX,LY),IA(IAMAX)
      DATA/IM1,I1,I2,I3,I4,I5,I1000/-1,1,2,3,4,5,1000/
  501 FORMAT(1H ,3HZA=,1PE11.4)
  502 FORMAT(1H ,5X,16HLINE OPEN      I=,I3,2X,2HJ=,I3,4X,2HP=,1PE11.4)
  503 FORMAT(1H ,5X,16HLINE CLOSED    I=,I3,2X,2HJ=,I3,4X,2HP=,1PE11.4)
  504 FORMAT(1H+,55X,6HFROM (,F8.4,1H,,F8.4,1H))
  505 FORMAT(1H+,80X,4HTO (  ,F8.4,1H,,F8.4,1H))
      LX=JX
      LY=JY
      FLX1=FLOAT(LX1)
      FLY1=FLOAT(LY1)
      ZA=Z
      ICOUNT=0
      IW=I3
  100 M=0
      WRITE(6,501)ZA
      J1=LY1+I1
      J2=LY-I1
      DO 600 J=J1,J2
      IMB=0
      DO 600 I=LX1,LX
      IF(P(I,J).LE.ZA) GO TO 601
      IF(IMB.NE.I1) GO TO 600
      M=M+I1
      IA(M)=I1000*I+J
      IF(M.GE.IAMAX) GO TO 101
      IMB=0
      GO TO 600
  601 IMB=I1
  600 CONTINUE
  101 IMA=I1
      IMB=0
      IXA=LX1-I1
      IYA=LY1
    1 IXA=IXA+I1
      IF(IXA.LT.LX)  GO TO 5
      IMA=I2
      GO TO 5
    2 IYA=IYA+I1
      IF(IYA.LT.LY)  GO TO 5
      IMA=I3
      GO TO 5
    3 IXA=IXA-I1
      IF(IXA.GT.LX1)  GO TO 5
      IMA=I4
      GO TO 5
    4 IYA=IYA-I1
      IF(IYA.GT.LY1) GO TO 5
      IMA=I5
    5 S=P(IXA,IYA)
      IF(S.GT.ZA) GO TO 7
      IMB=I1
    6 GO TO (1,2,3,4,91),IMA
    7 IF(IMB.NE.1) GO TO 6
```

```
      IMB=0
      IX=IXA
      IY=IYA
      S=P(IXA,IYA)
      INP=I1
      IF(IWRITE.EQ.IM1) GO TO 8
      WRITE(6,502)IX,IY,S
    8 GO TO (21,11,12,13,51),IMA
   11 IF(IY.NE.LY1)GO TO 31
      GO TO 21
   12 IF(IX.NE.LX) GO TO 41
      GO TO 31
   13 IF(IY.NE.LY) GO TO 51
      GO TO 41
   10 IX=IA(N)/I1000
      IY=IA(N)-I1000*IX
      S=P(IX,IY)
      INP=0
      IA(N)=0
      IF(IWRITE.EQ.IM1) GO TO 21
      WRITE(6,503)IX,IY,S
   21 IX=IX-I1
      INT=I1
      IF(IX.LT.LX1)GO TO 90
      SA=P(IX,IY)
      IF(SA.LE.ZA) GO TO 52
      S=SA
   31 IY=IY-I1
      INT=I2
      IF(IY.LT.LY1)GO TO 90
      SA=P(IX,IY)
      IF(SA.LE.ZA) GO TO 60
      S=SA
   41 IX=IX+I1
      INT=I3
      IF(IX.GT.LX) GO TO 90
      SA=P(IX,IY)
      IF(SA.LE.ZA) GO TO 60
      S=SA
   51 IY=IY+I1
      INT=I4
      IF(IY.GT.LY) GO TO 90
      SA=P(IX,IY)
      IF(SA.LE.ZA) GO TO 60
      S=SA
      GO TO 21
   52 IF (M.LE.0) GO TO 60
      IK=I1000*IX+IY+I1000
      DO 602 J= 1,M
      IF(IA(J).NE.IK) GO TO 602
      IA(J)=0
      GO TO 60
  602 CONTINUE
   60 CONTINUE
      WXX=FLOAT(IX)
      WYY=FLOAT(IY)
      XY=(ZA-SA)/(S-SA)
      GO TO (61,62,63,64),INT
```

```
 61 WXX=WXX+XY
    GO TO 165
 62 WYY=WYY+XY
    GO TO 165
 63 WXX=WXX-XY
    GO TO 165
 64 WYY=WYY-XY
165 WXP=(WXX-FLX1)*FCX+XO
166 WYP=(WYY-FLY1)*FCY+YO
118 CALL PLOT(WXP,WYP,IW)
    IF(IW.NE.I3) GO TO 66
 70 WX=WXX
    WY=WYY
    IW=I2
    IF (IWRITE) 110,68,69
 68 WRITE(6,504)WXP,WYP
110 GO TO 67
 69 WRITE(6,504) WXX,WYY
 67 GO TO (50,20,30,40),INT
 20 IY=IY+I1
    GO TO 21
 30 IX=IX-I1
    GO TO 31
 40 IY=IY-I1
    GO TO 41
 50 IX=IX+I1
    GO TO 51
 66 IF(WXX.NE.WX) GO TO 67
    IF(WYY.NE.WY) GO TO 67
 90 IW=I3
    IF(IWRITE) 97,98,96
 98 WXX=WXP
    WYY=WYP
 96 WRITE(6,505)WXX,WYY
 97 IF(IMA.NE.I5) GO TO 6
 91 IF(M.EQ.0) GO TO 94
    DO 603 N= 1,M
    IF(IA(N).NE.0) GO TO 10
603 CONTINUE
 94  ICOUNT=ICOUNT+I1
    IF(ICOUNT.GT.ICMAX) GO TO 123
    IF(W.EQ.0.0) GO TO 123
    ZA=ZA-W
    IF(ZA.GE.ZMIN) GO TO 100
123 RETURN
    END
```

4. CMLTQ のリスト

```
      SUBROUTINE CONTOU(P,LA,LB)
      COMMON/CMLTGA/ISIZE,IDATA,WIDTH,HIGH,IPORIG,ISW(11)
      COMMON/CMLTGI/LX1,LY1,ICMAX,IWRITE,XO,YO,Z,W,ZMIN,FCX,FCY
      COMMON/CMLTQO/ICOUNT,IW,INP,INT,IMA,M,IX,IY,FLX1,FLY1,ZA,WXX,WYY,
     C      WXP,WYP,WXY,S,XY,LX,LY, WX,WY
      DIMENSION P(1,1)
     C     ,IA(512)
      ICMAX=150
      IAMAX=512
      LX=LA
      LY=LB
      IF(IPORIG.EQ.1) GO TO 10
      LX1=1
      LY1=1
   10 IF(ISIZE .EQ.1) GO TO 20
      XO=0.0
      YO=0.0
      HIGH=10.0
      WIDTH=(10.0/FLOAT(LY-LY1))*FLOAT(LX-LX1)
   20 IF(IDATA .EQ.1) GO TO 30
      Z=P(LX1,LY1)
      ZMIN=Z
      DO 600 J=LY1,LY
      DO 600 I=LX1,LX
      IF(ZMIN.GT.P(I,J)) ZMIN=P(I,J)
      IF(Z.LT.P(I,J)) Z=P(I,J)
  600 CONTINUE
      W=(Z-ZMIN)*0.1
      Z=Z-0.5*W
   30 FCX=WIDTH/FLOAT(LX-LX1)
      FCY=HIGH/FLOAT(LY-LY1)
      WRITE(6,500)
  500 FORMAT(1H ,19HCONTOUR MAP PROGRAM)
      CALL CMLTQ(P,LX,LY,IA,IAMAX)
      ISIZE=0
      IDATA=0
      IPORIG=0
      RETURN
      END
```

航空宇宙技術研究所資料198号

昭和46年1月発行

発行所	航空宇宙技術研究所
	東京都調布市深大寺町1880
	電話武蔵野三鷹(0422)44-9171(代表)〒182
印刷所	第一印刷株式会社
	東京都新宿区富久町58

Printed in Japan