

研究員報告書

第 2 号

(1981.3)

岡山理科大学
情報処理センター資料室

目 次

1. 植物社会とその成立要因の相関解析 —岡山県北部における森林植生の成立要因に関する統計的解析—	教養部講師	波田善夫	1
2. Fortran用動的解析システムFordapの移植	応用数学科講師	馬野元秀	6
3. 数式処理II —量子統計における図形処理—	応用数学科助教授	坂本 薫	17
	" 助手	青江俊夫	
4. Bessel関数のサブルーチン	応用数学科助教授	一村 稔	23
5. 遷移金属カルコゲナイドにおける化学結合	化学科教授	中平光興	24
	" 講師	林 宏哉	
	" "	大谷 槻男	
6. 錯塩のX-線結晶解析	化学科講師	柴原隆志	25
7. モンテカルロによる多チャンネルカスケードの計算	機械理学科教授	田代 勤	28
8. 表面散乱におけるイオンフォーカシングの効果	応用物理学科教授	山村 泰道	30
	教養部助手	竹内 渉	
9. 生体力学系におけるカオスの挙動	電子理学科助手	太田垣 博一	34
10. CMTによるデータ処理方法の研究・開発	電子理学科講師	山本 勲	38
	応用物理学科助手	中村 改樹	
	情報処理センター助教授	木村 宏	
11. 文字のストローク抽出	電子理学科教授	橋本 禮治	41
12. C言語コンパイラの作成(1)	情報処理センター助教授	木村 宏	42
13. SL-1 言語による通信系のシミュレーション	電子理学科教授	宮垣 嘉也	50
	" 助手	小西 憲一	
14. クラマース・クローニツヒの関係式によるフォノン解析	基礎理学科助教授	若村 国夫	51
	" 助手	財部 健一	
15. "Spectrum-wise"なsuperconvergence algorithmに関する研究	応用数学科情報専攻教授	仁木 滉	53
	" " 講師	沢見 英男	
	" " "	池内 雅紀	
16. LP解法アルゴリズムの開発	教養部教授	成久 洋之	54

1. 植物社会とその成立要因の相関解析

—岡山県北部における森林植生の成立要因に関する統計的解析—

課題番号：50501

教養部講師 波 田 善 夫

森林の成立要因に関する研究は、従来実地踏査により得られる資料により解析されることが主流であった。近年環境庁や地方自治体による自然環境保全基礎調査が行われる機会が多くなったが、これらには航空写真の判読による広範囲の植生図が付随するのが常である。

このような状況下において、植生図、地質図、地形図から判読可能なデータを読み取り、それらと植生との相関を解析することにより、植物群落の成立要因を推定することを試みた。

調査対象とした地域は岡山県北部であり、国土地理院発行の5万分の1地形図、湯元、奥津、智頭、坂根、津山東部、佐用の岡山県内である。

1. 調査項目

読み取りに使用した地図は1/10万地質図、1/5万植生図、1/2.5万地形図である。これらの地図を緯度、経度ともに20"間隔のメッシュをひき、その交点について諸データ読み取った。なお、総交点数は4371であった。

a. 植物群落

第2回自然環境保全基礎調査（環境庁：1979）による植生図6葉を使用した。岡山県の植生は自然植生、二次植生、植林、耕作地、人家、裸地等の44の凡例に区分されている。これについて交点の植生を読み取った。

b. 地 質

地質図は岡山県地質図1/10万（光野・杉田：1979）を使用した。地質図はその性格上、境界線の線びきは、ある程度の幅を持つものである。この点を考慮し、交点が地質境界線より2mm以内に位置した場合は、両者に0.7：0.3の割合にて重み付けをし、境界線上に交点が位置した場合は0.5：0.5の重み付けをした。なお県内の地質は49の凡例に区分されている。

c. 標 高

1/2.5万地形図において、交点の標高を10m単位で読み取った。

d. 傾斜角およびその方向

1/2.5万地形図において、傾斜角は地形図上での標高差と水平距離から傾斜角を求め、その方向については北を0°とし、左まわりに30°間隔で読みとった。

e. 地 形

植生は、その成立する地形も強い影響を与えと考えられるので、交点がどのような地形上に位置するかを読み取った。

尾根・頂上部…… 1 斜面中部…… 2 斜面下部…… 3

f. 人家からの距離

人家の存在が、人為的影響の強を示すものとして、最も近い人家からの水平距離を求めた。

2. プログラム

プログラム1はデータの読み取りと項目ごとの出現回数、およびその出現割合に関する総括表の製作である。プログラム2は、同一のデータについて、植生との相関をプリントアウトするものである。

プログラム登録表

(*欄を作成者が記入する)

ライブラリー名	プログラム1 (Mainroutine)	作成日付*	1981. 12. 22.
プログラム名*	植物社会とその成立要因の統計的解析		
作成者氏名*	教養部講師 波 田 善 夫		
形式*	(a). コンプリート・プログラム c. 関数副プログラム b. サブルーチン・副プログラム d. その他		
使用機器*	(a). 磁気テープ 入力 (1) 本 出力 () 本 b. CRTの種類 () c. 磁気ディスク・ファイル () 個 (d). その他 (カード)		
プログラムサイズ*	a. ソース・プログラム (294) 行 b. オブジェクト・プログラム (52.2) KW		
使用言語*	a. FLAG d. COBOL g. Lisp (b). FORTRAN IV e. METASYMBOL h. PL/I c. FORTRAN 77 f. PASCAL i. その他 ()		
処理形態*	(a). バッチ処理専用 c. バッチ・TSS兼用 b. TSS専用		
使用条件等*			

プログラム登録表

(*欄を作成者が記入する)

ライブラリー名	プログラム2	作成日付*	1981. 12. 22.
プログラム名*	植物社会とその成立要因の統計的解析		
作成者氏名*	教養部講師 波 田 善 夫		
形式*	(a). コンプリート・プログラム c. 関数副プログラム b. サブルーチン・副プログラム d. その他		
使用機器*	(a). 磁気テープ 入力 (1) 本 出力 () 本 b. CRTの種類 () c. 磁気ディスク・ファイル () 個 (d). その他 (カード)		
プログラムサイズ*	a. ソース・プログラム (230) 行 b. オブジェクト・プログラム (3.3) KW		
使用言語*	a. FLAG d. C <small>OB</small> <small>OL</small> g. Lisp (b). F <small>OR</small> TRAN IV e. METASYMB <small>OL</small> h. PL/I c. F <small>OR</small> TRAN 77 f. PASCAL i. その他 ()		
処理形態*	(a). バッチ処理専用 c. バッチ・TSS兼用 b. TSS専用		
使用条件等*			

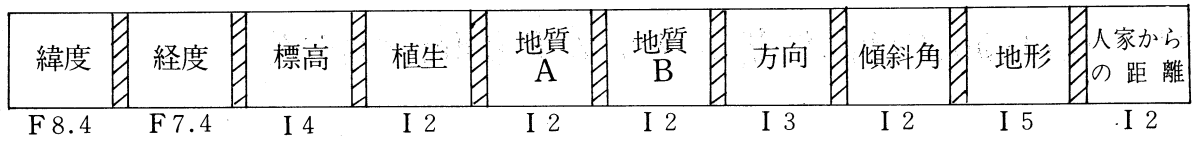


Fig. 1 データファイル

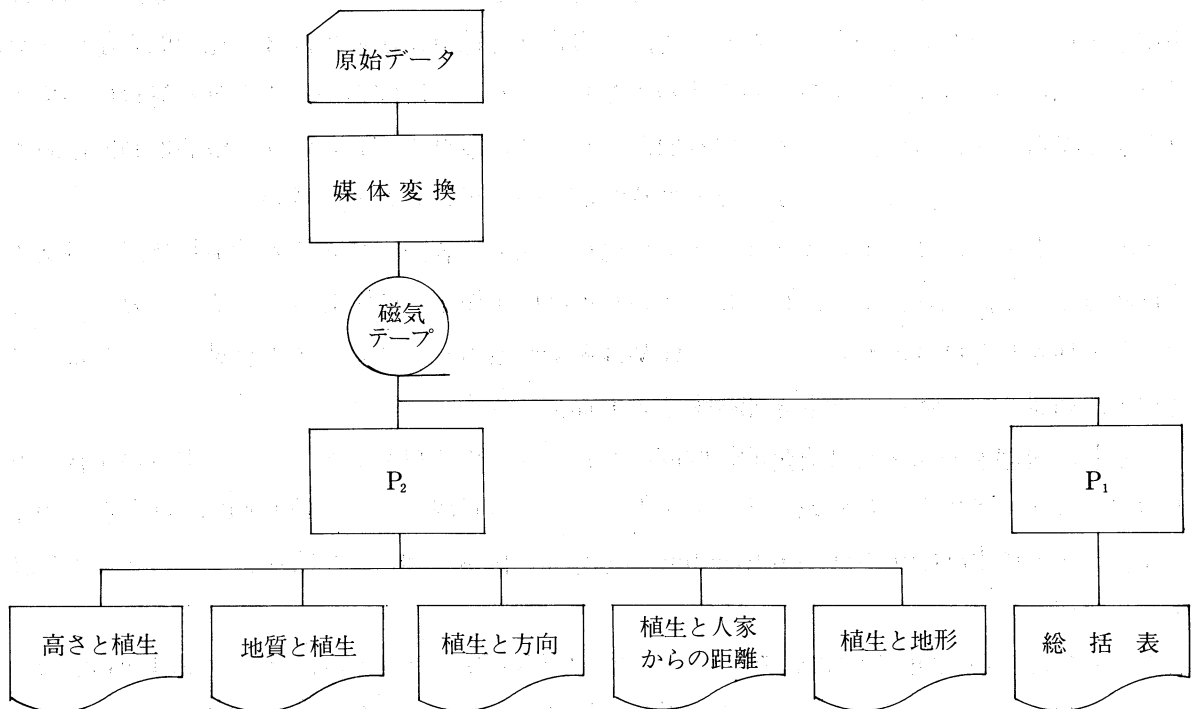


Fig. 2 フローチャート

2. Fortran用動的解析システム Fordap の移植

課題番号：50502

応用数学科講師 馬 野 元 秀

1. はじめに

入出力のあまり多くないプログラムの実行時間のほとんどは、そのテキストの数%の部分に集中していると言われている¹⁾。したがって、このような部分に注目すれば、比較的容易にプログラムの効率の大幅なアップを望むことができる。しかし、時間のかかる部分を見付け出すのはなかなか難しく、「この部分で多くの時間がかかっているだろう」という直観的な判断は誤まっていることが多く、とんでもない部分で時間がかかっていることが多い。

このようなときには、実際にプログラムを実行させて、各副プログラムの実行時間、各文の実行回数などを調べればよい。しかし、そのためには手作業で時間計測ルーチンやカウンタをたくさん挿入しなければならない。この作業は面倒であるし、プログラムを見にくくするだけでなく、結果をソース・リストと比較するのも面倒である。

しかし、言語を固定すれば自動的に時間計測ルーチンやカウンタを挿入し、実行終了後、もとのソース・リストと実行回数を並べて出力することも可能である。Fortranに対するこのようなシステムは Fordap (Fortran dynamic analyzer program) と呼ばれ、多くのところで使用されている。

本研究では、Fordap を情報処理センターの COSMO 700-III に移植した。この Fordap は、もとは九州大学・工学部・情報工学科の FACOM 230-45 S の OSII の上に実現されたもので²⁾、これを本学、応用数学科のミニコン pdp-11/45 の IAS 上の Fortran IV plus に合わせて移植したものをベース³⁾にしている。

Fordap の機能は、Fortran で書かれた誤りのないプログラムに対して、

- (1) 各実行文の実行回数を数える。さらに、論理 IF 文については、論理式が真となった回数も数える。
- (2) 副プログラムごとに、所要時間を計測する。
- (3) 結果の見やすさを考えて、実行回数はソース・リストと並べて出力する。

である。

Fordap を使ってプログラムをどのように解析していくか、Fordap をどのようにして作成するかについては文献2)に詳しく述べられているのでそちらに参照してもらうことにして、本報告では、COSMO 700-III のために拡張した点、使用上の注意点、具体的な使用法についての簡単な説明のみに留める。また、本 Fordap のより詳しい内容については文献4)を参照されたい。

2. COSMO-700III のための拡張

COSMO-700III 上の Fortran-IV と拡張 Fortran の両方で使えるようにするために次のような処理を追加した。

- ① 宣言文：ABNORMAL, CHARACTER (*n), GLOBAL, INTRINSIC, NAMELIST, QUADRUPLE, SAVE, VIRTUAL に対する処理
- ② 関数副プログラムの型：CHARACTER (*n), DOUBLE, DOUBLE INTEGER, QUADRUPLE PRECISION, QUADRUPLE COMPLEX に対する処理
- ③ 実行文：REPEAT k, FOR 条件式 と REPEAT k, WHILE 条件式 に対する処理
- ④ 注釈欄の文字：条件付コンパイル (X, Y, Z), インライン・アセンブラ (S) と注釈文字の拡張 (/ , * , ¥ , † , †† , ~) に対する処理
- ⑤ 複合文：セミコロンによる複合文に対する処理 (ただし, 行の最初の文でカウンタ挿入の要・不要を決定)

3. 使用上の注意点

Fordap では, ソース・プログラムを読んでそれを解釈し, それに従って加工するが, 文の解析部を複雑にしないようにするためと, システムが挿入する文と矛盾しないようにするためにいくつかの注意すべき点がある。

(1) 文の種類判定

- ① FORMAT 文は FORMAT (で判定しているので, 例えば,

```
150 | |6 | FORMAT (1, K) = R
```

という配列要素への代入文は, FORMAT 文とみなされて実行回数は表示されない。これは次の文の実行回数をみるか,

```
150 | |6 | CONTINUE  
    | |   | FORMAT (1, K) = R
```

として, 文番号150の文の実行回数をみればよい。

- ② 文関数定義文の方を代入文より優先して考えるので, 例えば,

```
| |6 | INTEGER F(10)  
    | |   | F(K) = ……
```

という配列要素への代入文は文関数定義文とみなされ, 実行回数は表示されない。これは, ダミーの文番号をつけるか, 前に CONTINUE 文を挿入すればよい。

- ③ IF 文では, (と) の数で括弧の対応を調べているので,

```

6 | IF (CHR. EQ. 1H( ) ) GO TO 100
  | IF (CHR. EQ. ▼ ) ▼ GO TO 200

```

は、ともに括弧の対応が不整の文として処理され、カウンターが挿入されないので、実行回数も表示されない。これは、1H(や ▼)▼ を変数に代入しておいて、その変数と比較すればよい。

- (2) システムが使用する名前との衝突
 - ① システムが使用する文番号は、65535番から下向きに生成するので、65000番台くらいは使用しない方がよい。
 - ② カウンタなどの名前として、SYS900, SYS901, SYS902, SYS903, SYS906, SYS907, SYS908 を使用しているので、SYS900~SYS909は使用しない方がよい。
 - ③ 論理装置番号 23, 24, 25, 89, 90, 91, 92, 93, 94, 95, 96は Fordap が使用しているので、ユーザは使用できない。
- (3) 論理装置番号の割り付け
 - ① 論理装置番号 6 は、Fordap からユーザへのメッセージを端末に表示するために使っているので、ラインプリンタに割り付けない方がよい。
 - ② ファイルからの入力に論理装置番号 5 を使用しない。Fordap の前処理部で異常が起こる。
 - ③ INCLUDE文の入れ子は 4 重までにする。

4. 使用方法

使用は現在のところ TSS からの使用を前提にしており、処理の流れは図 1 のようになる。コマンド・プロシージャの形で利用できるようになっている。以下、まず拡張 Fortran を利用する場合について順に述べる。

- (1) 拡張 Fortran 用のコマンド・プロシージャを指定する。(付録 1-(a) を参照)


```
! XEQ TOOL
```
- (2) 入力ファイル (解析したい Fortran のソース・ファイル) と Fordap が作る出力ファイルの名前を入力する。


```
INPUT FILE NAME?
? 入力ファイル名
OUTPUT FILE NAME?
? 出力ファイル名 (任意でよい)
```
- (3) TOOLNOUT の翻訳にはいとオプションの入力を要求してくるので、


```
OPTIONS> NS, NE, NOMD
```

 と入力する。

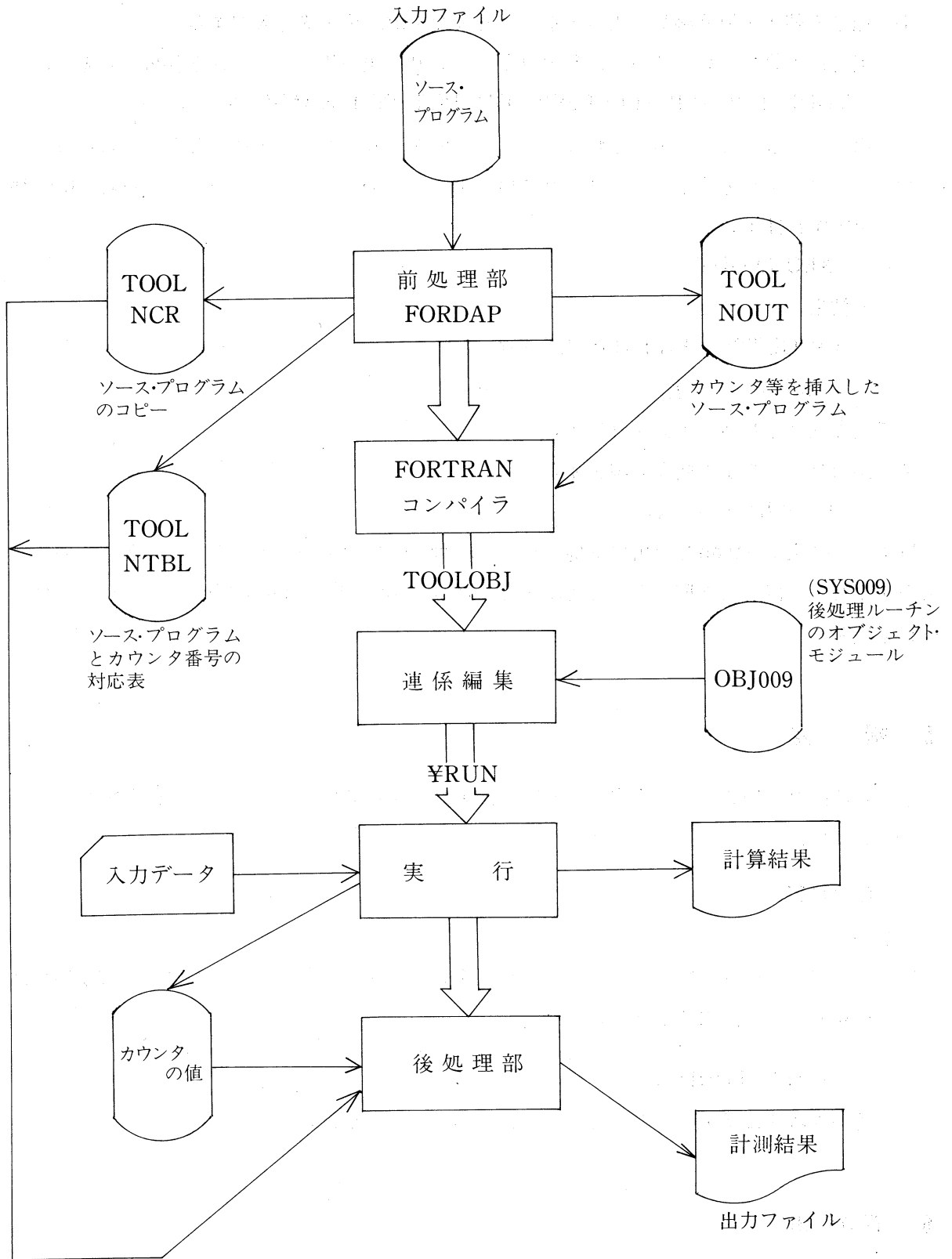


図1. FORDAP システムの流れ

- (4) 通常の翻訳・連係編集・実行となり、必要ならば、データを入力する。
(5) 実行がSTOPで終了すると、後処理部にはいり、出力用のファイルが作成されると、

TOOL FORDAP SUCCESS!! OUTPUT FILE NAME: ×××××

と出力される。これが出力されないときは、何らかのエラーで中断したと考えられる。

- (6) 出力ファイルをライン・プリンタに出力する。コマンド・プロシージャ（付録2-(c)参照）を指定すればよい。

! XEQ LPOUT

この結果、

×× OUTPUT FILE (LP=L1) ××

: 出力ファイル名

で出力ファイル名を入力すればよい。

- (7) 出力ファイルが不要なら消去する。

! D 出力ファイル名

以上は、拡張 Fortran で使用する場合だが、Fortran IV の場合も出力ファイル名を1回多く入力するだけでほとんど同じなので省略する。（付録2-(b)の Fortran IV 用のコマンド・プロシージャ参照）

5. 補 足

- ① 時間計測の中止：プログラムの実行時間を計測する必要があるときには、入力テキストの先頭に、

C-TIM⁶E|

をいれる。

- ② プログラムの異常終了：プログラムが正常終了しないときは、一般的には各文の実行回数が得られない。このときには、

|⁶CALL FORDAP

を適当な場所に挿入しておく、そこまでの実行回数を得られる。

6. 実行例

図2に簡単なプログラムに対するソース・ファイル(a)、カウンタ等を挿入したファイル(b)、および実行回数を並べた出力ファイル(c)を示す。

17:02 MAR 18, '81 DC/TOOLNCR.50502

```
      DIMENSION N(200)
      I=0
1     I=I+1
      READ(5,10) N(I)
-----
      IF(N(I).EQ.9999) GO TO 2
      GO TO 1
2     I=I-1
      CALL SORT(I,N)
      WRITE(6,20) (N(K),K=1,I)
10    FORMAT(I5)
-----
20    FORMAT(10I5)
      STOP
      END
C     SORTING
C
      SUBROUTINE SORT(N,M)
-----
      DIMENSION M(200)
      LOGICAL SW
      I=0
      K=1
      SW=.FALSE.
C
-----
1     IF(SW) RETURN
      SW=.TRUE.
2     I=I+K
      J=I+1
      IF((J.GT.N).OR.(I.LE.0)) GO TO 3
      IF(M(I).LT.M(J)) GO TO 2
-----
      L=M(I)
      M(I)=M(J)
      M(J)=L
      SW=.FALSE.
      GO TO 2
C
-----
3     K=-1*K
      GO TO 1
      END
C
```

図2. (a) ソース・ファイル

```

1 REAL      SYS906( 2000)
2 LOGICAL   SYS907
3 INTEGER   SYS908
4 COMMON   /SYS900/SYS906,SYS907,SYS908
5 DIMENSION N(200)
6 DO 65535 SYS908=1, 2000
7   SYS906(SYS908)=0.0
8 65535 CONTINUE
9 CALL SYS901( 1)
10 SYS906( 3)=SYS906( 3)+1.0
11 I=0
12 1 CONTINUE
13 SYS906( 4)=SYS906( 4)+1.0
14 I=I+1
15 READ(5,10) N(I)
16 SYS906( 5)=SYS906( 5)+1.0
17 SYS907=
18 ¥(N(I)-EN.9999)
19 IF (SYS907) SYS906( 6) = SYS906( 6) + 1.0
20 IF (SYS907)
21   ¥GO TO 2
22 SYS906( 7)=SYS906( 7)+1.0
23 GO TO 1
24 2 CONTINUE
25 SYS906( 8)=SYS906( 8)+1.0
26 I=I-1
27 CALL SORT(I,N)
28 SYS906( 9)=SYS906( 9)+1.0
29 WRITE(6,20) (N(K),K=1,I)
30 FORMAT(15)
31 10 FORMAT(10I5)
32 CALL SYS902 ( 1)
33 CALL SYS903
34 STOP
35 END
36 SURROUTINE SORT(N,M)
37 REAL      SYS906( 2000)
38 LOGICAL   SYS907
39 INTEGER   SYS908
40 COMMON   /SYS900/SYS906,SYS907,SYS908
41 DIMENSION M(200)
42 LOGICAL SW
43 CALL SYS901( 10)
44 SYS906( 13)=SYS906( 13)+1.0
45 I=0
46 K=1
47 SW=.FALSE.
48 1 CONTINUE
49   SYS906( 14)=SYS906( 14)+1.0
50 SYS907=
51 ¥(SW)
52 IF (SYS907) SYS906( 15) = SYS906 ( 15) + 1.0
53 IF (SYS907) GO TO 65535
54 SYS906( 16)=SYS906( 16)+1.0
55 SW=.TRUE.
56 2 CONTINUE
57 SYS906( 17)=SYS906( 17)+1.0
58 I=I+K
59 J=I+1
60 SYS907=
61 ¥((J.GT.N).OR.(I.LE.0))
62 IF (SYS907) SYS906( 18) = SYS906 ( 18) + 1.0
63 IF (SYS907)
64   ¥GO TO 3
65 SYS906( 19)=SYS906( 19)+1.0
66 SYS907=
67 ¥(M(I).LT.M(J))
68 IF (SYS907) SYS906( 20) = SYS906 ( 20) + 1.0
69 IF (SYS907)
70   ¥GO TO 2
71 SYS906( 21)=SYS906( 21)+1.0
72 L=M(I)
73 M(I)=M(J)
74 M(J)=L
75 SW=.FALSE.
76 GO TO 2
77 3 CONTINUE
78 SYS906( 22)=SYS906( 22)+1.0
79 K=-1*K
80 GO TO 1
81 65535 CONTINUE
82 CALL SYS902 ( 10)
83 RETURN
84 END

```

図2. (b) カウンタの挿入されたソース・ファイル

MELCOM COSMO-700 UTS/V/S FORDAP FILE: TEL DATE: 3-19-1981 TIME: 15: 3:40 SUBPROGRAM - 1

* SOURCE STATEMENT *	EXECUTION TIME :	48 MSEC	EXECUTIONS	TRUE
0001	DIMENSION N(200)			
0002	I=0		1	
0003	1 I=I+1		11	
0004	READ(5,10) N(I)		11	
0005	IF(N(I).EQ.9999) GO TO 2		1	(9.1%)
0006	GO TO 1		10	
0007	2 I=I-1		1	
0008	CALL SORT(I,N)		1	
0009	WRITE(6,20) (N(K),K=1,I)		1	
0010	10 FORMAT(15)		1	
0011	20 FORMAT(10I5)			
0012	STOP		1	
0013	END			

MELCOM COSMO-700 UTS/V/S FORDAP FILE: TEL DATE: 3-19-1981 TIME: 15: 3:41 SUBPROGRAM - 2

* SOURCE STATEMENT *	EXECUTION TIME :	4 MSEC	EXECUTIONS	TRUE
0001	C SORTING			
0002	C			
0003	SUBROUTINE SORT(N,H)		1	
0004	DIMENSION M(200)			
0005	LOGICAL SW			
0006	I=0		1	
0007	K=1		1	
0008	SW=.FALSE.		1	
0009	C			
0010	1 IF(SW) RETURN		8	1 (12.5%)
0011	SW=.TRUE.		7	
0012	2 I=I+K		70	
0013	J=I+1		70	
0014	IF((J.GT.N).OR.(I.LE.0)) GO TO 3		70	7 (10.0%)
0015	IF(M(I).LT.M(J)) GO TO 2		63	35 (55.6%)
0016	L=M(I)		28	
0017	M(I)=M(J)		28	
0018	M(J)=L		28	
0019	SW=.FALSE.		28	
0020	GO TO 2		28	
0021	C			
0022	3 K=J*K		7	
0023	GO TO 1		7	
0024	END			

図2. (c) 実行回数つきソース・リスト

〔謝辞〕

本システムの作成にあたって、文献2)の Fordap の pdp-11/45 と COSMO700-III への移植を心よく御許可下さいました九州大学・工学部・牛島和男教授に感謝致します。また、pdp-11/45 と COSMO-700III へ移植については、ゼミ生 谷昭男君(昭和55年3月卒業)と衣笠和明君(昭和56年3月卒業)が事実上ほとんどすべての作業をしてくれた。2人に感謝する。

〔参考文献〕

1. D. E. Knuth, "Structured Programming with go to statements," ACM Computing Surveys, Vol. 6, No.4, pp.261-301 (1974).
2. 牛島和夫, 「Fortran プログラミングツール」, コンピュータサイエンス・ライブラリ, 産業図書 (1979).
3. 谷昭男, 「Fortran プログラミング・ツール Fordap システムの作成」, 岡山理科大学・応用数学科・卒業論文 (1980).
4. 衣笠和明, 「Fortran プログラミング・ツール Fordap の COSMO-700III への移植」, 岡山理科大学・応用数学科・卒業論文 (1981).

付 録

15:10 MAR 19, '81 DC/TOOL.50502

```
!DONT ECHO
!!WRITE ** FORDAP SYSTEM RUN **
!!CONTROL TERM
!S ¥FORDAP
-----
!!WRITE ** SYSTEM FILE COMPILE **
!!WRITE ** OPTIONS --> NS,NE,NOMD **
!EXTFORT TOOLNOUT OVER TOOLOBJ
!!WRITE ** SYSTEM FILE LINK **
!LYNX TOOLOBJ,OBJ009,¥DATE.OKLIB,¥CPUNE.OKLIB OVER ¥RUN ;.EXTLIB
!!WRITE ** SYSTEM FILE RUN **
!!WRITE ** INPUT YOUR DATA ? **
!S ¥RUN
!!WRITE ** SYSTEM FILE DELETED **
!D TOOLNOUT,TOOLNCR,TOOLNTBL,TOOLOBJ
!D ¥RUN
```

(a) コマンド・プロシージャ TOOL

15:22 MAR 19, '81 DC/COMF4.50502

```
!!DCL (FILEN,10)
!DONT ECHO
!RESET
!!WRITE ** OUTPUT FILE NAME ? **
!!READ &FILEN
!!WRITE ** FORDAP SYSTEM RUN **
!!CONTROL TERM
!S ¥FORDAP
-----
!!WRITE ** SYSTEM FILE COMPILE **
!!CONTROL NOTERM
!FORT TOOLNOUT OVER TOOLOBJ4
NS,BO
!!WRITE ** SYSTEM FILE LINK **
!SET F:23/TOOLNCR;IN
!SET F:24/&FILEN;OUT
!SET F:25/TOOLNTBL;IN
!LYNX TOOLOBJ4,OBJ004,¥DATE.OKLIB,¥CPUTIME.OKLIB OVER ¥RUN4
!!CONTROL TERM
!S ¥RUN4
!!WRITE ** SYSTEM FILE DELETED **
!D TOOLNOUT,TOOLNTBL,TOOLNCR,TOOLOBJ4,¥RUN4
```

(b) コマンド・プロシージャ COMF4

15:12 MAR 19, '81 DC/LPOUT,50502

```
!!DCL (FILEN,10)  
!DONT ECHO  
!!WRITE ** OUTPUT FILE ( LP=L1 ) **  
!!READ &FILEN  
-----  
!C &FILEN TO L1  
!LDEV L1,(FORM,FFFF),(ASAVE)  
!PRINT  
!!WRITE ** END **
```

(c) コマンド・プロシージャ LPOUT

3. 数式処理 II

—量子統計における図形処理—

課題番号：50503

応用数学科助教授 坂 本 薫

” 助 手 青 江 俊 夫

ファイマン・ダイアグラムを図示するとき、相互作用の摂動の次数が増加すると、ダイアグラムの数は増加し、図形も非常に複雑になる。そこで私達は、ダイアグラムの生成とそのダイアグラムの作図をコンピュータで処理する研究をした。ここで問題を簡単にするために、外部からの粒子の出入りがないと仮定した。

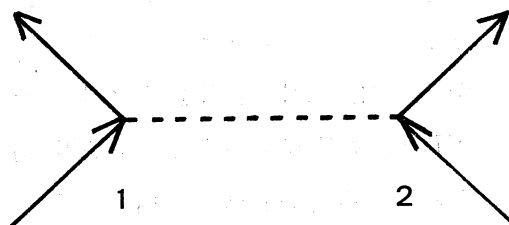
1. ファイマン・ダイアグラムの生成

ファイマン・ダイアグラムを表わすリストは、本学計算センターの Lisp K のプログラムの計算結果によって得られた。¹⁾

2. ファイマン・ダイアグラムの図形処理

本学計算センターのグラフィック・ディスプレイ装置では、画面のサイズの制限があるので、本学応用数学科の X-Y プロッタを利用することにした。

相互作用は、図1のように水平な破線で表現される作用線と、その作用線上の右、左両端の各々のバーテックスに入る1本の有向線（消失演算子を表わす）とバーテックスから出る有向線（生成演算子を表わす）²⁾によって表示される。（図1）



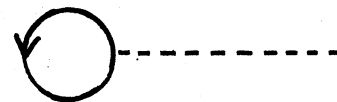
(図1)

2. 1 プログラムの説明

○主プログラム (FDPROG)

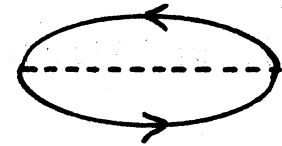
ダイアグラムは基本的には、次の三つのタイプに分類される。

タイプ1. 1本の有向線が、あるバーテックスから出て、もとのバーテックスに入る場合（図2）。



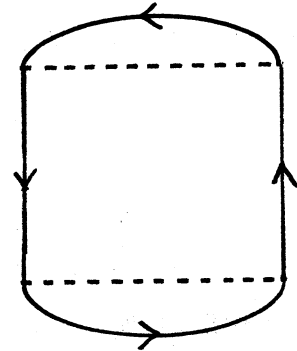
(図2)

タイプ2. ある作用線上の1方のバーテックスから1本の有向線が出て、同一の作用線上のもう1方のバーテックスに入り、そこから再びもとのバーテックスに入る場合(図3)。



(図3)

タイプ3. ある作用線上のバーテックスから1本の有向線が出て、他の作用線上のバーテックスをいくつかを経て、最初の作用線上のもとのバーテックスに入る場合(図4)。



(図4)

1で生成されたデータは生成演算子と消失演算子のコントラクションの積であるので、それらを、互につながったループを形成する組に分け、それぞれの組が上記の3つのタイプの中のどれに相当するかを判別して、各タイプの処理ルーチンに分岐する。

—サブプログラム1 (FDHEAD ルーチン)—

図形の見出しを書くルーチン。これから作図するファイマン・ダイアグラムを表すリストを書き出すと同時に、図がプロッタ用紙におさまるように調整するルーチン。

SUBROUTINE FDHEAD (ND, JCARD, ………)

NDは図の続き番号, JCARDはファイマン・ダイアグラムを表すリストを含む配列名。

—サブプログラム2 (TYPE 1 ルーチン)—

上記タイプ1に相当する図を描くルーチン。すなわち、ある点における生成演算子と、同一の点における消失演算子と結合するように有向線を描く。作図した結果は「円」を示す。

SUBROUTINE TYPE1 (ICR)

ICRは生成演算子の位置を表わす値を持つ。ICRで指定された位置に円を描く。

—サブプログラム3 (TYPE 2 ルーチン)—

上記のタイプ2に相当する図を描くルーチンである。ある点の生成演算子と他の点の消失演算子と結合し、その点の生成演算子ともとの点の消失演算子とを結合する有向線を描く。作図した結果は「楕円」を示す。

SUBROUTINE TYPE2 (ICR, IAN)

ICRは生成演算子の位置を, IANは消失演算子の位置を表わす値を持つ。ICRとIANで示された2点を通る楕円を描く。

—サブプログラム4 (TYPE 3 ルーチン)—

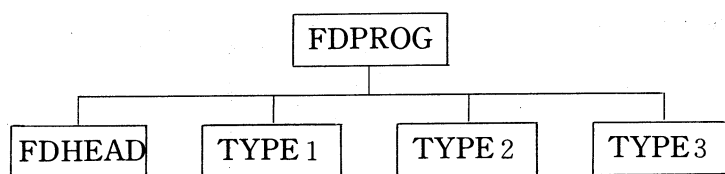
上記のタイプ3に該当する図形を描くルーチンである。このサブプログラムでは、前述のタイプ1及びタイプ2以外のすべての場合を処理する。実際には、指定された生成演算子から消失演算子に向けて有向線をひく処理をする。このとき両演算子の位置関係により図形が「直線」になる場合と「半楕円」になる場合とがある。

SUBROUTINE TYPE3 (ICR, IAN, ………)

ICRは生成演算子の位置を, IANは消失演算子の位置を表わす値を持つ。ICRとIANとで

示された2点を直線又は半楕円で結合する。ICRが奇数番号でIANが偶数番号のとき、又はその逆のとき、直線で結合する。ICRとIANとがともに奇数番号又はともに偶数番号で $|ICR - IAN| = 2$ のときは直線で結合する。その他の場合は半楕円で結合する。

2.2 プログラムの構成



FDHEAD ルーチンは、1つのファイマン・ダイアグラムを描く時にただ1回呼ばれるだけである。1組のファイマン・ダイアグラムを描くとき、プロッタのペンの移動は最短コースを通るように考慮されている。

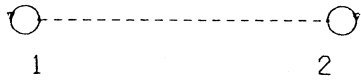
【参考文献】

- 1) 坂本薫, 青江俊夫, “数式処理1 - 量子統計力学におけるファイマン・ダイアグラムの自動生成 -”, 研究員報告書 (岡山理科大学 情報処理センター資料室) 第1号, P.3~6 (1980. 3)
- 2) 坂本薫, 青江俊夫, “量子統計における図形処理”, 情報処理学会第22回大会予稿集, P.605, 1981.

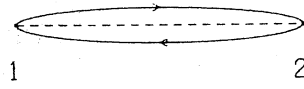
付 録

ファイマン・ダイアグラムの図形処理の結果

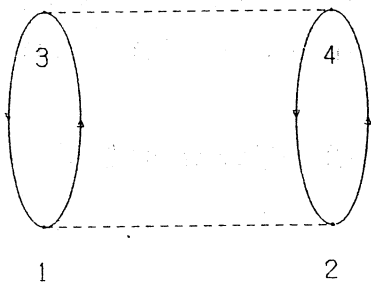
$((1\ 1)(2\ 2))$



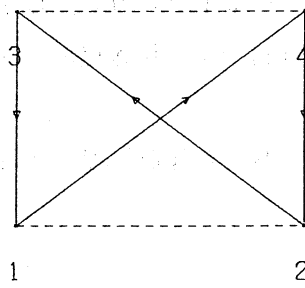
$((1\ 2)(2\ 1))$



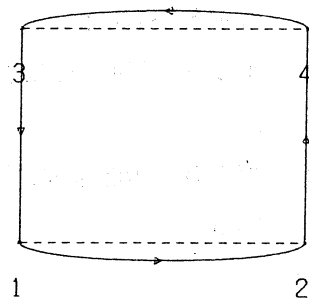
$((1\ 3)(3\ 1)(2\ 4)(4\ 2))$



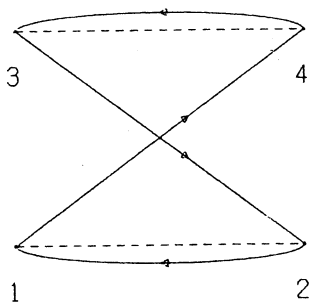
$((1\ 4)(4\ 2)(2\ 3)(3\ 1))$



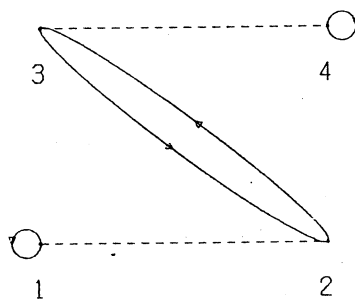
$((1\ 2)(2\ 4)(4\ 3)(3\ 1))$



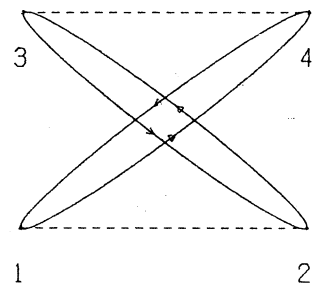
$((1\ 4)(4\ 3)(3\ 2)(2\ 1))$



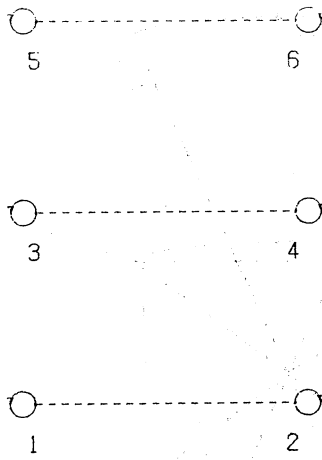
$((1\ 1)(2\ 3)(3\ 2)(4\ 4))$



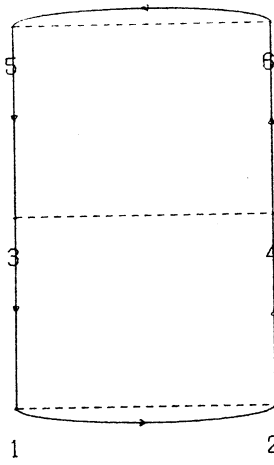
$((1\ 4)(4\ 1)(2\ 3)(3\ 2))$



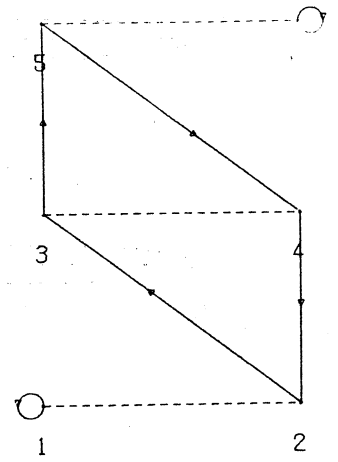
$((1\ 1)(2\ 2)(3\ 3)(4\ 4)(5\ 5)(6\ 6))$



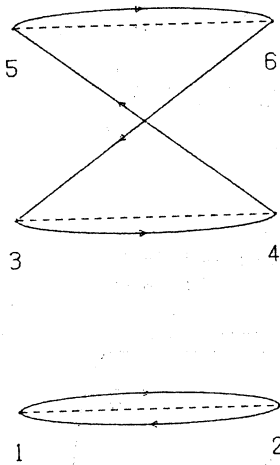
$((1\ 2)(2\ 4)(4\ 6)(6\ 5)(5\ 3)(3\ 1))$



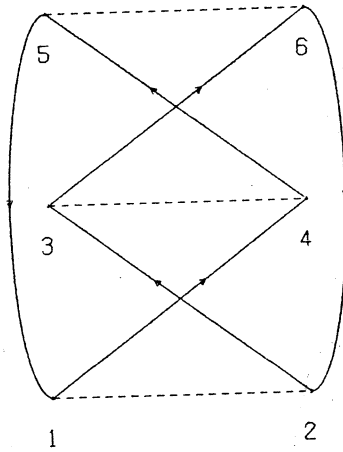
$((1\ 1)(6\ 6)(2\ 3)(3\ 5)(5\ 4)(4\ 2))$



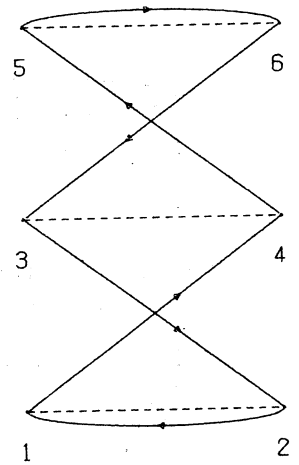
$((1\ 2)(2\ 1)(3\ 4)(4\ 5)(5\ 6)(6\ 3))$



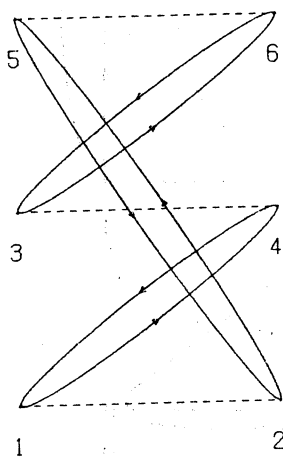
$((1\ 4)(4\ 5)(5\ 1)(2\ 3)(3\ 6)(6\ 2))$



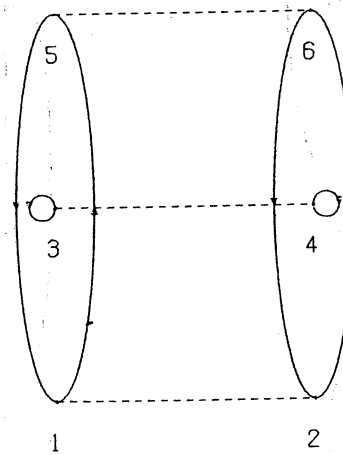
$((1\ 4)(4\ 5)(5\ 6)(6\ 3)(3\ 2)(2\ 1))$



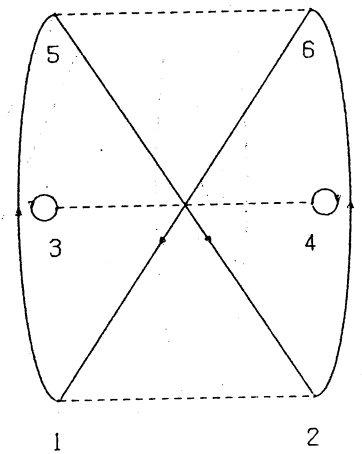
$((1\ 4)(4\ 1)(2\ 5)(5\ 2)(3\ 6)(6\ 3))$



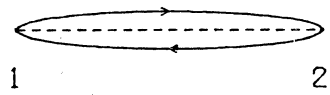
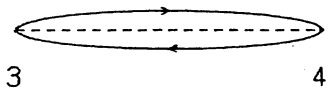
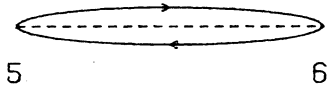
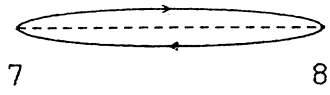
$((1\ 5)(5\ 1)(3\ 3)(6\ 2)(2\ 6)(4\ 4))$



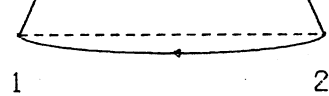
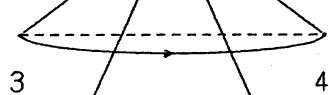
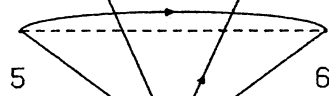
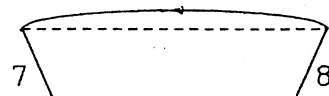
$((1\ 5)(5\ 2)(2\ 6)(6\ 1)(3\ 3)(4\ 4))$



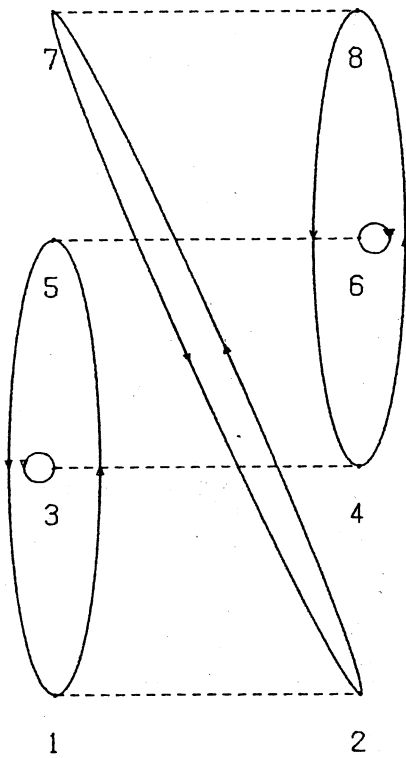
$((1\ 2)\ (2\ 1)\ (3\ 4)\ (4\ 3)\ (5\ 6)\ (6\ 5)\ (7\ 8)\ (8\ 7))$



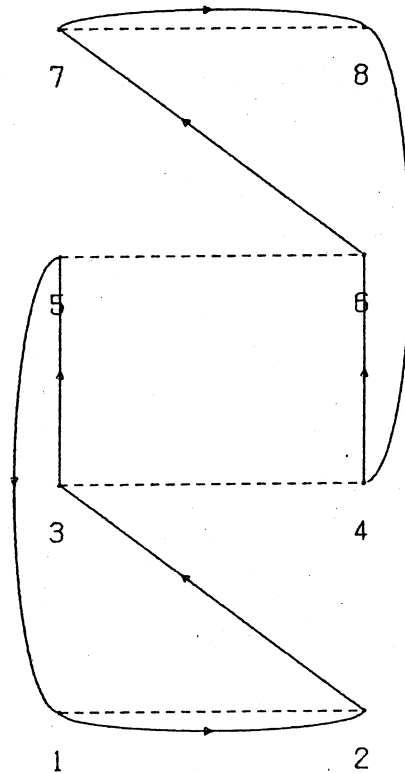
$((1\ 8)\ (8\ 7)\ (7\ 2)\ (2\ 1)\ (3\ 4)\ (4\ 5)\ (5\ 6)\ (6\ 3))$



$((1\ 5)\ (2\ 7)\ (3\ 3)\ (4\ 8)\ (5\ 1)\ (6\ 6)\ (7\ 2)\ (8\ 4))$



$((1\ 2)\ (2\ 3)\ (3\ 5)\ (5\ 1)\ (4\ 6)\ (6\ 7)\ (7\ 8)\ (8\ 4))$



4. Bessel 関数のサブルーチン

課題番号：50504

応用数学科助教授 一 村 稔

締切り日に間に合わなかったので報告は次号にさせていただきます。

5. 遷移金属カルコゲナイドにおける化学結合

課題番号：50505

化学科教授	中	平	光	興
" 講師	林		宏	哉
" "	大	谷	槻	男

締切り日に間に合わなかったので報告は次号にさせていただきます。

6. 錯塩のX-線結晶解析

課題番号：50506

化学科講師 柴原隆志

1) モリブデン元素は窒素固定酵素(ニトロゲナーゼ)等多くの酸化還元酵素の中に含まれている。酵素それ自体の結晶化はなされておらず、モリブデンのまわりの配位形式も明らかにされていない。今回モリブデンとマロン酸を含む新化合物を合成し、X-線結晶解析によりその構造を明らかにしたので報告する。計算に必要なプログラムはUNICS*のRSSFR-5(構造因子およびフーリエ合成)およびHBL5-IV(ブロックダイアゴナル近似最小2乗法)を本学情報処理センター MELCOM-COSMO 700III に conversion して用いた。

※The Universal Crystallographic Computation program System (UNICS), ed.

T. Sakurai, The Crystallographic Society of Japan, 1967.

2)

Synthesis and X-Ray Crystal Structure of a Novel Malonato Bridged Tetrameric Molybdenum(V) Complex, $K_6[(Mo_2O_4(mal)_2)_2(mal)] \cdot 4H_2O$

Many molybdenum(V) compounds containing Mo_2O_4 core, which is dominant in Mo(V) chemistry, have been prepared in aqueous solution [1]. The oxalato complex is the only example so far prepared to contain a dicarboxylic acid as a ligand [2]. We will report here a synthesis and structural study on a novel malonato complex and compare its structure with that of the oxalato complex [3].

The potassium salt of the complex was prepared by the following procedure: to the malonic acid (H_2mal ; 2.1 g or 0.02 mol) in 50 ml of water, $[pyH]_2[MoOCl_5]$ (4.5 g or 0.01 mol) was added and the solution was heated for ca. 40 min at 80–90 °C. After the solution had been cooled to room temperature, its pH was adjusted to ca. 4.7 with dilute KOH solution. Potassium chloride (1.0 g) was added with stirring and then the solution was made turbid by the addition of ethanol. Brick red crystals were obtained by filtration after standing for several days in a refrigerator and washed successively with ethanol and diethyl ether; yield 1.8 g.

The compound crystallizes in the monoclinic system, space group $P2_1$ with cell dimensions $a = 12.832(3)$, $b = 20.848(9)$, $c = 7.471(1)$ Å, $\beta = 94.96(1)^\circ$ and $Z = 2$. Intensity data were collected on a Phillips PW1100 four-circle diffractometer using graphite monochromatized Mo- $K\alpha$ radiation ($\lambda = 0.7107$ Å) with ω -scan technique. The structure was determined by the Patterson and Fourier procedures and refined by block-diagonal least-squares methods to a final R value of 0.043 for 4006 ($F_o^2 > 3\sigma(F_o^2)$) reflections collected in the $2\theta \leq 55^\circ$ range. All but hydrogen atoms were refined anisotropically.

The overall symmetry of the complex ion is close to C_{2v} ; there is an approximate two-fold rotation axis through the C2 atom of the malonato bridge as shown in Fig. 1, and the coordination scheme is totally different from the oxalato complex, $Ba[Mo_2O_4(ox)_2(H_2O)_2] \cdot 3H_2O$ [3], where the ligating oxygen atoms in the oxalato ligand are in *cis* and *trans* positions to the terminal oxygen. The malonato complex anion contains two types of malonato ligands; one is a bidentate chelate ligand which coordinates to one Mo(V) atom to form a six-membered ring and has a boat conformation [4], the other is a quadridentate ligand to form a bridge between two $Mo_2O_4(mal)_2$ moieties. No report, to our knowledge, has so far appeared on the quadridentate malonato complexes.

The distances** Mo1–Mo2 (2.555 Å), Mo1–Ot (1.684 Å), and the displacement (0.37 Å) of Mo1 towards Ot from the mean plane defined by four oxygens (Ob1, Ob2, O3, and O4) are in good agreement with those of Mo_2O_4 cores previously reported [1, 5]. The C–C–C angle (115°) of the bridging malonato anion is essentially the same as the mean angle (116°) of the four non-bridging malonato C–C–C angles. The distance Mo1–O1 (2.321 Å) is evidently longer than that of Mo1–O3 (2.081 Å) indicating the *trans* influence of Ot. The oxalato complex has been known to show no *trans* influence due to the terminal oxo ligand [3], and it is also the case with the di- μ -sulfido oxalato complex, $Cs_2[Mo_2O_2S_2(ox)_2(H_2O)_2] \cdot 2H_2O$ [6]; the coordination scheme is, however, different between the malonato and oxalato complexes and direct comparison should not be made straightforwardly. The angle Ot–Mo1–O1 ($\theta_1 = 171^\circ$) and the dihedral angle between the Mo1, Ob1, and Ob2 plane and the Mo2, Ob1, and Ob2 one ($\theta_2 = 161^\circ$) are considerably larger than those of similar types of complexes having Mo_2O_4 core[†]. These can be attributed to the short O1–O2 distance (2.230 Å) caused by the coordination of the bridging malonato anion as a quadridentate ligand. The UV and visible spectra show a peak and a shoulder at 315 ($\epsilon = 11200$) and 420 nm ($\epsilon = 720$), respectively. The deformation of Mo_2O_4 core by the bridging malonato anion may be responsible for the red shift of the CT band around 300 nm^{††}.

**Average value of two to eight distances.

[†]E.g. for $[Mo_2O_4(ox)_2(H_2O)_2]^{2-}$, $\theta_1 = 159^\circ$ and $\theta_2 = 150.6^\circ$ calculated using the atomic parameters by Cotton and Morehouse [3]; for $[Mo_2O_4(cys)_2]^{2-}$, $\theta_1 = 162.6^\circ$ and $\theta_2 = 151^\circ$.

^{††}The oxalato complex has an absorption peak and a shoulder at 305 ($\epsilon = 7040$) and 384 nm ($\epsilon = 340$), respectively.

References

- 1 E. I. Stiefel, *Progress in Inorganic Chem.*, 22, 1 (1977) and the references therein.
- 2 a) R. G. James and W. Wardlaw, *J. Chem. Soc.*, 2145 (1927).
b) W. P. Griffith, *J. Chem. Soc. (A)*, 211 (1969).
c) P. H. C. Mitchell, *J. Chem. Soc. (A)*, 146 (1969).
- 3 F. A. Cotton and S. M. Morehouse, *Inorg. Chem.*, 4, 1397 (1965).
- 4 a) T. Lis and J. Matuszewski, *J. Chem. Soc. Dalton*, 996 (1980).
b) K. R. Butler and M. R. Snow, *J. Chem. Soc. Dalton*, 259 (1976).
c) K. Matsumoto and H. Kuroya, *Bull. Chem. Soc. Jpn.*, 44, 3491 (1971).
- 5 B. Spivack and Z. Dori, *Coord. Chem. Rev.*, 17, 99 (1975).
- 6 W. S. McDonald, *Acta Crystallogr.*, B34, 2850 (1978).

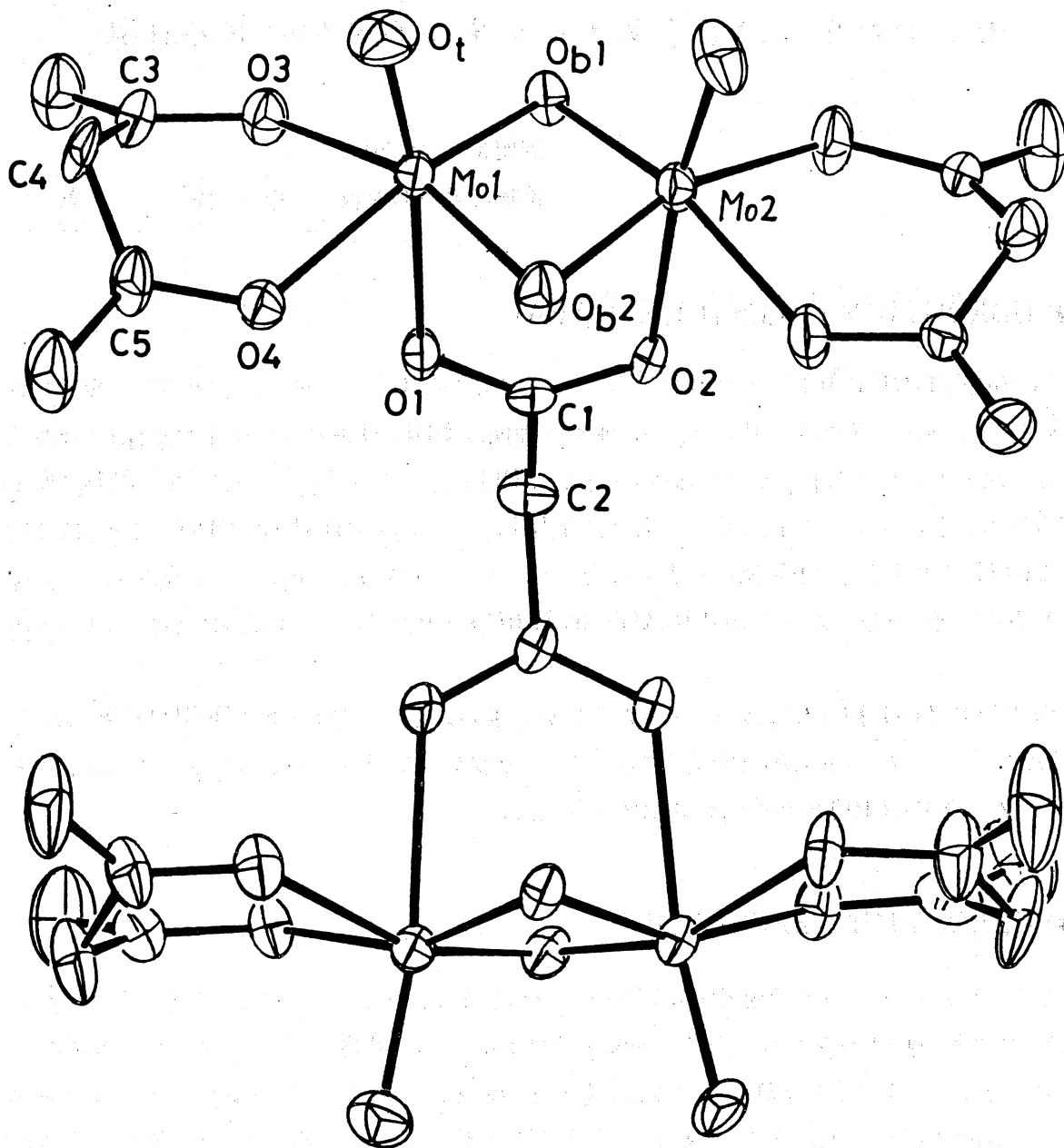


Fig. 1. Structure of the $[\{Mo_2O_4(mal)_2\}_2(mal)]^{6-}$ anion.

3) 謝 辞

UNICSの本学情報センターへのconversionに際し木村助教授に多大のご教示をいただきましたことを感謝いたします。

7. モンテカルロによる多チャンネルカスケードの計算

課題番号：50507

機械理学科助教授 田代 勤

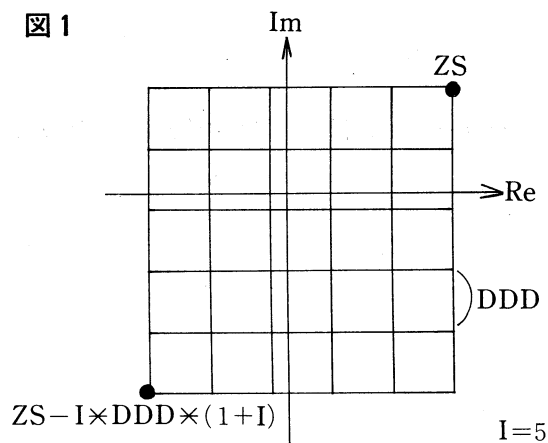
●SUBROUTINE PÔLES (KKK, HABA)

SUBROUTINE PÔLE と積分のサブルーチン（ここでは GL を使った）を用いて、複素関数 $SS(Z)$ のポールを求める。図 1 のように求める範囲は $DDD \times I$ の長さの正方形で右上が ZS である。図の小さな正方形をかこむ線積分の値が CRITE より大きければ、その値を $ZSSS(K, 1)$ 、正方形の右上の位置を $ZPPP(K, 1)$ 、 $K=1 \sim KKK$ にとっておき $HABA=DDD$ とする。積分値が CRITE より大きな KKK 個の正方形について、さらに I 等分して同じことを繰り返す。この操作を NAN 回くり返したあと SUBROUTINE ZERÔS を用いてポールの位置と留数を詳しく決める。

SUBROUTINE PÔLE は格子にそった線積分を求めるものである。SUBROUTINE GL は READ(500) でファイル GAUSSLEG を読んでおく必要がある。 $U=0$ のときは途中の計算を打出さない。 $ZS, DDD, CRITE, NAN$ は MAIN で与える。

●SUBROUTINE ZERÔS (S1, D)

与えられた点 $S1$ の近くで複素関数 $SF(Z)$ が 0 になる点を探すプログラムである。与えられた点 $S1$ で $XX=D/10$ のきざみで $|SF|$ の勾配を計算する。 $-\text{grad}|SF|$ の方向に $Z=S1$ より XX づつずれた点で $|SF(Z)|$ を計算し値の増加する点を求める。この点でまた勾配を計算し上と同様に、 $-\text{grad}|SF|$ の方向に $XX=XX/10$ のきざみで $|SF|$ の増加する点を求める。増加する点に至らない時はきざみを大きくする。この操作をくり返して $SF(Z)=0$ の点を求める。



プログラム登録表

(*欄を作成者が記入する)

ライブラリー名		作成日付*	1981. 9. 9.
プログラム名*	PÔLES, ZERÔS		
作成者氏名*	田 代 勤 三 坪 昭 夫		
形 式*	a. コンプリート・プログラム c. 関数副プログラム ②. サブルーチン・副プログラム d. その他		
使 用 機 器*	a. 磁気テープ 入力 () 本 出力 () 本 b. CRTの種類 () c. 磁気ディスク・ファイル () 個 d. その他 (カード)		
プログラムサイズ*	a. ソース・プログラム (60, 66) 行 b. オブジェクト・プログラム () KW		
使 用 言 語*	a. FLAG d. CÔBÔL g. Lisp ②. FÔRTRAN IV e. METASYMBÔL h. PL/I c. FÔRTRAN 77 f. PASCAL i. その他 ()		
処 理 形 態*	a. バッチ処理専用 ③. バッチ・TSS兼用 b. TSS専用		
使 用 条 件 等*	PÔLES はサブルーチン PÔLE, GL 関数プログラム F, G, SS と共に用い SS にポールを求めようとする関数を入れる。積分路が pole のごく近くを通るときにはあやしいため積分路を変える必要がある。		

8. 表面散乱におけるイオンフォーカシングの効果

課題番号：50508

応用物理学科教授 山村 泰 道
教 養 部 助 手 竹 内 渉

結晶の表面構造解析の可能性を試みるために、表面散乱におけるイオンのフォーカシング効果に着目してモンテカルロ・シミュレーションを行った。以下で簡単に計算モデルとプログラムを報告する。

§1. 計算モデル

(a) 二体間衝突近似

散乱過程は古典的二体間衝突近似に基づいており、散乱角 Θ は重心系において、

$$\Theta = \pi - 2p \int_R^{\infty} dr [r^2 g(r)]^{-1}, \quad (1)$$

time integral τ は

$$\tau = (R^2 - p^2)^{1/2} - \int_R^{\infty} dr \left[[g(r)]^{-1} - \left(1 - \frac{p^2}{r^2}\right)^{-1/2} \right] \quad (2)$$

で与えられる。ここに、

$$g(r) = \left[1 - \frac{p^2}{r^2} - \frac{V(r)}{E_r} \right]^{1/2} \quad (3)$$

である。ただし、 p は衝突係数、 E_r は相対運動エネルギー、 r は原子間距離、 $V(r)$ は原子間力のポテンシャル、 R は $g(R)=0$ を満たす入射粒子のトラジェクトリーの最近接距離である。散乱角 Θ を(1)式、time integral τ を(2)式により、衝突係数 p とエネルギー E との2次マトリックスとして数値計算をして、散乱の都度内挿公式により求めている。尚、(1)と(2)式での数値積分は4点 Gauss-Legendre の積分公式を用いている。

(b) 二体間ポテンシャル

入射イオンと標的原子との二体間ポテンシャルとしては Molière ポテンシャル

$$V(r) = \frac{Z_1 Z_2 e^2}{r} \Phi\left(\frac{r}{a}\right) \quad (4)$$

$$\Phi\left(\frac{r}{a}\right) = 0.35 \exp\left(-0.3 \frac{r}{a}\right) + 0.55 \exp\left(-1.2 \frac{r}{a}\right) + 0.10 \exp\left(-6.0 \frac{r}{a}\right) \quad (5)$$

を採用した。ただし、 Z_1 、 Z_2 はそれぞれ入射粒子と標的原子の原子番号、 $\Phi(r/a)$ は screen-

ing functionを表わす。また、 a は screening lengthであり、Firsovの screening length a_F

$$a_F = \frac{0.4685}{(Z_1^{1/2} + Z_2^{1/2})^{2/3}} \text{ (\AA)} \quad (6)$$

を使用した。

(c) 非弾性エネルギー損失

衝突係数に依存している Firsovの electronic energy loss $Q(E, p)$

$$Q(E, p) = \frac{C_1 E^{1/2}}{(1 + C_2 p)^5} \quad (7)$$

ただし、

$$C_1 = \frac{0.0594(Z_1 + Z_2)^{5/3}}{M_1^{1/2}} \text{ (eV}^{1/2}) \quad (8)$$

$$C_2 = 0.304(Z_1 + Z_2)^{1/3} \text{ (\AA}^{-1}) \quad (9)$$

を用いている。 M_1 は入射粒子の atomic massである。

(d) 熱振動

確率分布関数はガウス分布 $f(u)$

$$f(u) = (2\pi u_1^2)^{-1/2} \exp\left(-\frac{u^2}{2u_1^2}\right) \quad (10)$$

であると仮定する。ただし、 u は変位、 u_1 は1次元の rms (root mean square)の値である。1次元の rmsの値は固体における Debye モデルを使って、

$$u_1 = 12.1 \left[\frac{\varphi(x)/x + 1/4}{M_2 \Theta_D} \right]^{1/2} \quad (11)$$

により求めている。ここで、 M_2 は振動原子の質量、 Θ_D は Debye 温度、 $\varphi(x)$ は Debye 関数、 $x = \Theta_D/T$ である。なお、 T は結晶の温度である。

§2. プログラムの構成

プログラムの概略を簡単に説明する。

MAIN

○ READ 文により各諸量を INPUT する。

○ CALL PARAMA

○ CALL PARAMB

○ CALL PARAMC

○ TVAMP

○ 初期条件として、入射イオンの位置を乱数で発生させて、randomに結晶表面にイオン照

射させる。(図1)

○入射イオンの位置の決定により、標的原子の位置を設定する。

○CALL FIDOUT

○CALL SCAT

○標的から散乱されたイオンを count してイオンのエネルギー分布や空間分布等を求める。

SUBROUTINE PARAMA

○screening length や非弾性エネルギー損失等のパラメーターを決定する。

SUBROUTINE PARAMB

○散乱角 θ や time integral τ を衝突係数 p とエネルギー E の2次マトリックスとして table にする。

SUBROUTINE PARAMC

○小さい散乱角 θ の場合には、運動量近似を用いて θ を求める。

FUNCTION TVAMP

○熱振動による rms の計算。

SUBROUTINE FIDOUT

○標的原子の位置を primitive vector と reciprocal lattice vector で表わし、FCCやBCCの結晶を作成する。そして、入射イオンのバルクでの運動を求める。

SUBROUTINE SCAT

○散乱角 θ と time integral τ の table から内挿公式により、散乱の都度 θ と τ を決定する。
さらに、エネルギー損失(弾性エネルギー損失と非弾性エネルギー損失の両方を含む)を計算する。

SUBROUTINE ZTHET

○ θ と τ を求める際の数値積分を行う。

§3. 計算結果

図2から図4に、Cu単結晶(100)の $\langle 110 \rangle$ に Ar^+ を照射した場合のイオンのトラジェクトリーのシミュレーション結果を表わした。なお、これらの図は $\langle 110 \rangle$ の原子列の横断面内に投影しており、○は標的原子を示す。入射エネルギーの違いによりイオンのフォーカシングが大きく異なる。即ち、イオンのフォーカシング効果が重要である。図5は全反射係数の解析結果、図6はそれのシミュレーション結果である。両者の結果の傾向は一致している。

これらに関連した成果の詳細は、

- (1) Y. Yamamura and W. Takeuchi, Radiat. Eff. 49, 251(1980).
- (2) Y. Yamamura and W. Takeuchi, Phys. Lett. 78A, 105(1980).
- (3) W. Takeuchi and Y. Yamamura, Proc. Fourth Symposium on ISAT'80, P.303(1980).

で報告されている。

図 1

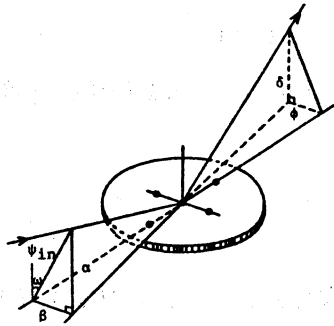


図 2

$E_0 = 30 \text{ KeV}$, $T = 0 \text{ K}$

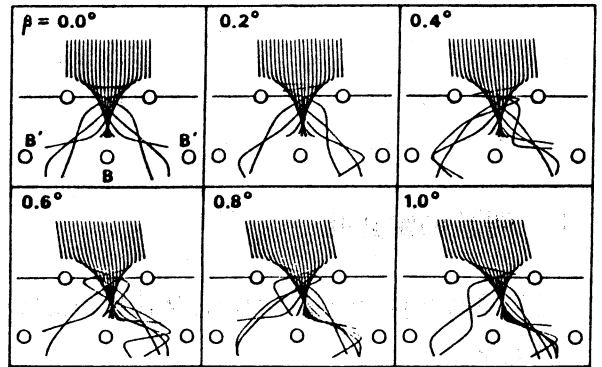


図 3

$E_0 = 50 \text{ KeV}$, $T = 0 \text{ K}$

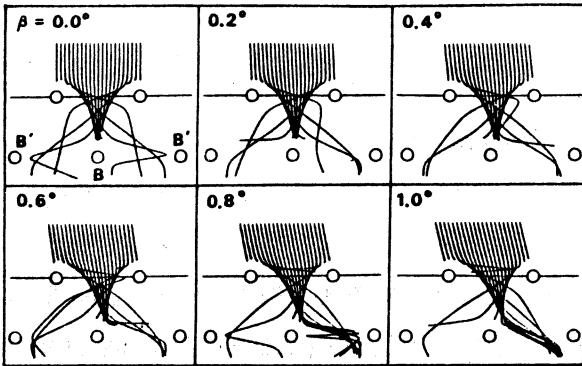


図 4

$E_0 = 200 \text{ KeV}$, $T = 0 \text{ K}$

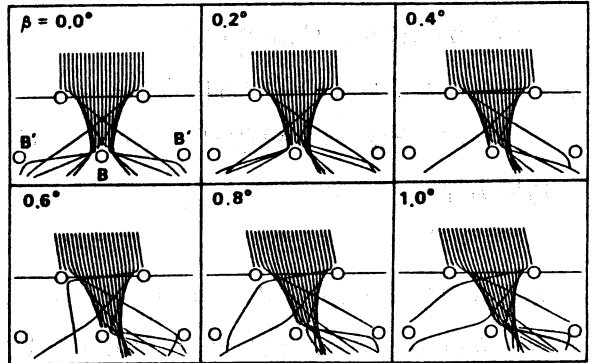


図 5

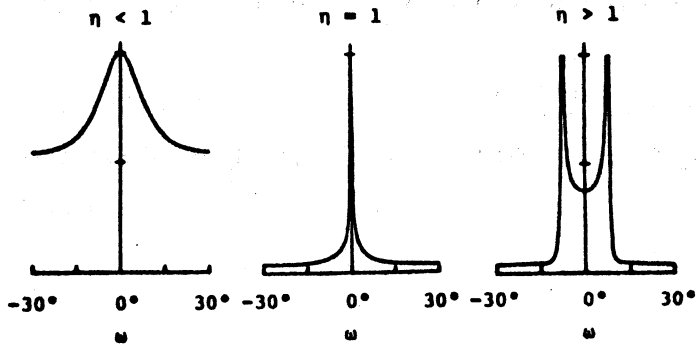


図 6

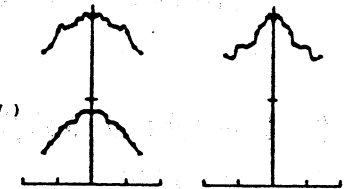
$\text{Ar}^+ \rightarrow \text{Cu}(100)$

$\alpha = 4^\circ$
 $T = 300 \text{ K}$

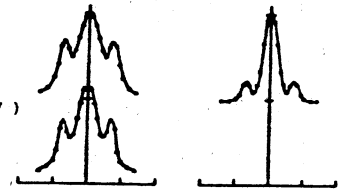
$\frac{E'}{E_0} \geq 0.0$

$\frac{E'}{E_0} \geq 0.95$

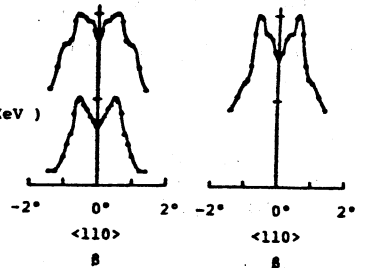
$\eta < 1$
($E_0 = 30 \text{ KeV}$)



$\eta = 1$
($E_0 = 50 \text{ KeV}$)



$\eta > 1$
($E_0 = 200 \text{ KeV}$)



9. 生体力学系におけるカオスの挙動

課題番号：50509

電子理学科助手 太田垣 博 一

1. 問題の定式化

本研究は生体力学系における個体数の変化を記述する方程式として、一次元の Langevin-型の方程式

$$(1-a) \quad \dot{U} = (\epsilon - \beta U)U + \sigma \dot{\xi} \quad t \in [t_0, \infty), \quad \left(\cdot = \frac{d}{dt} \right)$$

で初期条件

$$(1-b) \quad U(t_0) = U_0 : \text{確定的}$$

をもつものおよびその差分方程式であるところの

$$(2-a) \quad X(t_{k+1}) = aX(t_k)\{1 - X(t_k)\} + \sigma B(t_k), \quad k=1, 2, \dots$$

で初期条件

$$(2-b) \quad X(t_0) = X_0$$

をもつものを考える。ただし σ , a は正の定数, $\dot{\xi}$ は正規性の白色雑音過程, B は Gauss 型の不規則変数である。式(2)は式(1)に対応する Euler の方法による差分方程式である。この研究では式(1)と式(2)の解について、パラメータの変化にともなってどのような挙動が発生するかについて考察しているが、ここでは特に、幻影解 (式(2)に対するカオスの挙動) が発生する差分方程式の数値計算について報告する。

2. Euler法による差分方程式

式(1-a)において, $dU = U(t_{k+1}) - U(t_k)$, $dt = \Delta t = h$, $dW(t_k) = W(t_{k+1}) - W(t_k)$ と同定して差分化すると

$$(3) \quad U(t_{k+1}) = U(t_k) + h\{\epsilon - \beta U(t_k)\} + \sigma\{W(t_{k+1}) - W(t_k)\}$$

を得る。 $w(t_{k+1}) - w(t_k)$ は標準 Brown 運動過程の, 時間 $t_{k+1} - t_k$ における増分を表わしており $B(t_k) = B_k$ と同定し, $U(t_k) = U_k$ と書くと

$$(4) \quad U_{k+1} = (1 + \epsilon h - \beta h U_k)U_k + \sigma B_k$$

ここで,

$$(5) \quad \begin{cases} 1 + \epsilon h = a \\ \beta h U_k / a = X_k \\ \beta h \sigma / a = \sigma' \end{cases}$$

とおくと式(4)は

$$(6) \quad X_{k+1} = a(1-X_k)X_k + \sigma' B_k$$

と表わされる。

次に式(1)において σ が一定ではなく、状態変数 $U(t)$ に依存する場合について考える。この場合の差分方程式には Wong and Zakai の修正項を必要とする。すなわち対応する差分方程式は次のようになる。

$$(7) \quad X_{k+1} = a(1-X_k)X_k + \frac{\beta h}{a} \frac{1}{2} \frac{d\sigma(X_k)}{dX_k} \sigma(X_k) + \sigma' B_k$$

ここで $\sigma(X_k) = \sqrt{X_k}$ とすると

$$(8) \quad X_{k+1} = a(1-X_k)X_k + \frac{\beta}{4a} h + \sigma' B_k$$

となる。式(6), (8)に対する確率 Lyapunov 関数として

$$(9) \quad V(X_k) = \frac{1}{2} X_k^2$$

をとる。このとき解 X_k の平衡解への確率的漸近収束性を保証する規範は次のようになる。

$$(10) \quad LV = -aX_k^2(1+X_k) + \frac{1}{2}\sigma^2(X_k) \leq 0$$

式(10)の L は式(1)に対応している微分生成作用素を表わしている。

3. 中点法による差分方程式

ここでは $\sigma=0$ なる場合についてのみ考察する。

式(1)に対する中心差分をとる中点法（一般的には修正 Euler 法といわれている）を用いると式(3)に対すると同様にして、

$$(11-a) \quad Y_{k+1} = Y_{k-1} + 2h a Y_k (1 - Y_k)$$

となる。ここで初期条件は

$$(11-b) \quad \begin{aligned} Y_0 &= U_0 \\ Y_1 &= U_0 + h U_0 (1 - U_0) \end{aligned}$$

で与えられているものとする。ここで式(11-b)における Y_1 は Euler 法によって与えられている。式(11)は次のように書き換えられる。

$$(12) \quad \begin{aligned} X_{k+1} &= Y_k + 2h X_k (1 - X_k) \\ Y_{k+1} &= X_k \end{aligned}$$

式(12)は二次元の可微分同相写像である。

式(6), (8), (10)と(11)に対する数値計算のプログラムを図1に示す。

```

A0=1.0
BE=1.0
U0=0.3
AV=0.3
VN=0.01
SD=SQRT(VN)
VI=12345678
H=0.01
D0 20 T=0,100
A+A0+T*H
B=1/A
C=BE*H/A
SUMRY=0.
SRY=0.
U=U0
X=BE/A+H*v0
WRITE(6,13) A, B, C
DO 10 K=0,200
VX=X*X/2.
VF=X*X/2.-X*X*X/3.
VL=-A*X*X*(1+X)+VN*X/2.
TK=K*H
U=U/(B+C*U)
CALL NORRND ( AV, SD, VI, RY)
FX=A*(1.-X)*X*H+SD*X*RY*SQRT(H)
X=FX
SUMRY=SUMRY+RY
ARY=SUMRY/K
SUMSRY=SUMSRY+RY*RY
SRY=SUMSRY/K
VRY=SRY-ARY*ARY
WRITE (6,11) K, TK, X, U, VX, VF, VL
WRITE (6,12) RY, ARY, VRY
10 CONTINUE
20 CONTINUE

```

```
11 FÖRMAT(1H , 2X, 'A=', F10.5, 2X, 'B=', F10.5, 2X, 'C=', F10.5)
12 FÖRMAT(1H , 2X, 'K=', I4, 2X, 'TK=', F10.5, 2X, 'X=', E15.8, 'V=', E15.8, 2X,
'UX=', E15.8, 2X, 'VF=', E15.8, 2X, 'UL=', E15.8),
13 FÖRMAT(1H , 2X, 3(E15.8, 2X))
STÖP
END
```

(圖 1)

10. CMTによるデータ処理方法の研究・開発

課題番号：50510

電子理学科講師 山 本 勲
応用物理学科助手 中 村 改 樹
情報処理センター助教授 木 村 宏

1. 研究目的と成果

我々は昨年度情報処理センター研究員として、CMTを用いたマイコンとセンターのコンピュータとの連結について報告した。本年は、この成果を発展させるために、マイコン側のCMTの統一的处理に関するソフトウェアを開発したので、引き続き第II報として報告する。

当初は、センター側のCMT入出力装置であるTI733(SILENT700)に遠隔装置制御機構(Remote Device Controller RDC)が増設されるはずであった。従来のTI733がCMTの入力中にリードエラーを検出すると停止し、人間が介入して1ブロック巻戻し、再読み込みさせるという手段を講じなければならず、非常に操作性が悪かったのを、この機構を用いるとホスト側のソフトウェアで自動回復できるようになり操作性の改善がはかれる。我々はこれを用いてREADCT¹⁾の改良を行なう予定を立てていたが、設置時期が遅くなりそうなので、取りあえず本年は、マイコン側でのCMTの統一的处理系(第一版)を作製した。

したがって、本報告に添付されたプログラムは、マイコン(8080/85系)に組込むプログラムである。ただし、これはC²⁾OSMO700III上にインプリメントしたクロスアセンブラでアセンブルして用いるソースプログラムである。

2. CMT処理プログラムの概要

ユーザーが簡便に利用出来ることを目的とし、用意したプログラムは、(1)センター標準形式、(2)センター非標準形式、(3)その他、である。(1)及び(2)は各々リード(CMT→マイコン)とライト(マイコン→CMT)に分けられる。さらに(1)については、ユーザーがプログラムの中に組み込んで使うものと、単独で使えるものに分けた。

(1)の形式によるライトは制御ブロックとデータブロックで構成されるが、ユーザーは(i)0～9までの数字で構成された8桁のName、(ロ)テープに書き込みたいデータの入っているメモリの先頭番地、(ハ)その最終番地、を指定するだけで使える。(2)の形式によるライトは、(ロ)と(ハ)を指定すると使える。(3)は(1)の形式で書かれたテープを管理するために用意した。

3. プログラムの一覧

プログラム名	設 定 値			備 考
	Name	先 頭 番 地	終 了 番 地	
標 準 リ ー ド	必 要	必 要		単 独 使 用
標 準 ラ イ ト (巻もどし付)	必 要	必 要	必 要	単 独 使 用
標 準 ラ イ ト	必 要	必 要	必 要	単 独 使 用
標準ライト (BIN)	必 要			組 込 使 用
標準ライト (END)				組 込 使 用
標準ライト (Data)		必 要		組 込 使 用
非 標 準 リ ー ド		必 要		単 独 使 用
非 標 準 ラ イ ト		必 要	必 要	単 独 使 用
Name 検 索	検 索 Name 必 要			単 独 使 用
Name 検 索	検 索 Name 必 要			組 込 使 用
Name Print				単 独 使 用

設定値は次の番地に入れる。

Name [書き込み時 X'28F0'~X'28F5' (4 バイト)

検索 Name [読み出し時 X'28F4'~X'28F7' (4 バイト)

先頭番地 [X'7F70'…アドレス (下位)

[X'7F71'…アドレス (上位)

終了番地 [X'7F72'…アドレス (下位)

[X'7F73'…アドレス (上位)

4. おわりに

本報告で述べたマイコン用のCMT処理プログラム群はROMに書き込まれ一応テストが終っている。今後は機能拡張及び改良を行って多くの方々に利用していただくことを願っている。なお、このプログラムのROMでの配布は電子理学科の山本が担当することになっている。

〔参考文献〕

1. 「CMTによるデータ処理方法の研究開発」 山本, 中村, 木村; 岡山理科大学情報処理センター研究員報告書, 第1号 (1980); P.20.
2. 「8080系のクロスアセンブラについて」 木村宏; 岡山理科大学情報処理センターニュース Vol. 1. (1979); P. 4.
3. MCS-85TM USER'S MANUAL; Intel: 1978.
4. TEACカセット磁気テープ装置 MT-2 レファレンス・マニュアル; ティアック(株)

11. 文字のストローク抽出

課題番号：50511

電子理学科教授 橋 本 禮 治

文字ストロークの抽出は、未完成であるので、途中段階で用いたサブルーチンのうちで、比較的汎用性があると思われるものをあげる。

(A) Freeman コードまたはチェーン・コード生成プログラム。 $(N \times M)$ の二値画像の輪郭をチェーン・コード化し、座標およびコード値を登録する。

(B) 連結成分抽出プログラム。上記二値画像の一点を与えたとき、それに連結するすべての点を抽出する。最初“*”で与えられた画素のうち、連結成分は“0”に置き換えられる。4-連結、8-連結の他に、任意の点对称連結性が定義できる (DATA 文で)。

(C) 細線化プログラム。内外に多数のアルゴリズムがあり、ソースプログラムも公表されているが、多少の工夫を加えて新たに作成した。しかし、結果は他のものと大差がなかった。(交差点、分岐点などの影響において)。

以上、あまり利用者も予想されないので、プログラムリスト等は省略する。

12. C言語コンパイラの作成 (1)

課題番号：50512

情報処理センター助教授 木 村 宏

1. はじめに

1978年に出版された「The C Programming Language¹⁾」を入手して以来、本学のMELCOM COSMO-700上にC言語処理系をインプリメントしたいと考えていた。その理由は、C言語が次のような性質を有しているからである。

① システム記述言語である。

DEC PDP-11上のUNIX OSの記述用として実用化された言語であり、UNIXシステムが使用できる環境では従来のアセンブラ的な役割をはたしている。

② 構造的プログラミングを意識した言語である。

PASCALに似た制御構造を有しており、FORTRANなどに比べて構造的プログラミングがしやすくなっている。

③ 豊富な演算子群を有する。

本来PDP-11を有効に利用するための演算子群であるが、MELCOMでも簡潔なプログラミングと効率のよいオブジェクト生成を可能にさせられると思われる。

④データ構造が簡潔である。

基本データ型として、整数、実数、文字型があり、この上に配列、構造体(レコード)、共用体(ユニオン、FORTRANのEQUIVALENCEの効果に似ている)、これらへのポインタ及び、これらを値とする関数を定義できる。従来のFORTRANやPASCALとは一味異なる使いやすさが予想される。

このようにCは非常に魅力的な言語であるが、1人が1年間で設計、プログラミング、デバッグを完了させることは、とうてい不可能であり、数年間にわたって継続研究する予定である。

さて著者がCの詳細検討に入った頃、事情により、別途SIMPL-R^{4,5)}言語処理系の作成に取りかからねばならなくなった。このSIMPL-R言語はMaryland大学で開発されたシステム記述言語(主にコンパイラの記述向)であり、著者としては、はからずも同一目標をもった2言語処理系の開発に着手せざるを得なくなった。そのため、取りあえずSIMPL-R処理系を先に完成させ、その過程で得られたknowhowをC処理系の開発に応用することにした。したがって、今回はC及びSIMPL-Rなどの処理系を実用に耐える形態に仕上げる場合に必要となる。コンパイラの初期化の部分についてSIMPL-Rのルーチンを例に上げて報告する。

なお、C言語はその後マイコンを中心に急速に普及し始め、³⁾おそらく本格的16ビットマイコ

ンでは主流の言語となることが予想される。また、MELCOM-COSMO 700上にも、山梨大学の有沢先生の所でインプリメントが開始されたとのことであり、著者としてはその動向も気にしながら SIMPL-R の完成を急いでいる。

2. コンパイラの初期化の部分について

言語処理系を高水準言語で作成すれば、通常はその高水準言語の動作環境で機能する別言語処理系となる。しかし言語処理系は OS と直接インターフェースを持った形態で動作させる方がより使い勝手がよく、またメモリ効率なども向上する。例えば FORTRAN を用いて別言語のコンパイラを作成すれば、Fig. 1 のようにたえず FORTRAN ライブラリをかかえなければコンパイルも実行もできない。これではシステムのサイズが過大となり何かと不便である。

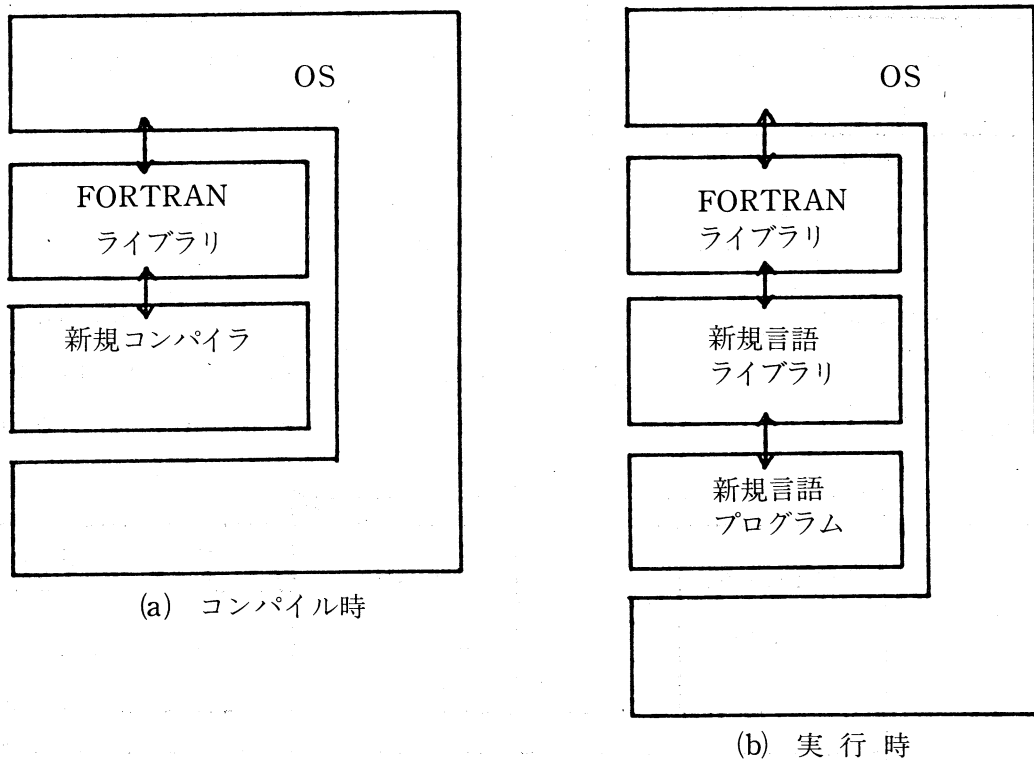


Fig.1 FORTRAN を用いた言語処理系の例

したがって処理速度をあげ、OS とのインターフェースを合理化しようとするれば、少なくとも Fig. 2 のような構成になるであろう。これはコンパイラを作成する際に、OS と当該コンパイラとを結ぶアセンブラ版のサブルーチン群と、実行時ライブラリをアセンブラで作成することを意味する。

さて今回報告するルーチンは、コンパイラの初期化の部分であるが、これが行なう主な作業には次のようなものが考えられる。

- ① 記号表などのコンパイル時のメモリ領域確保

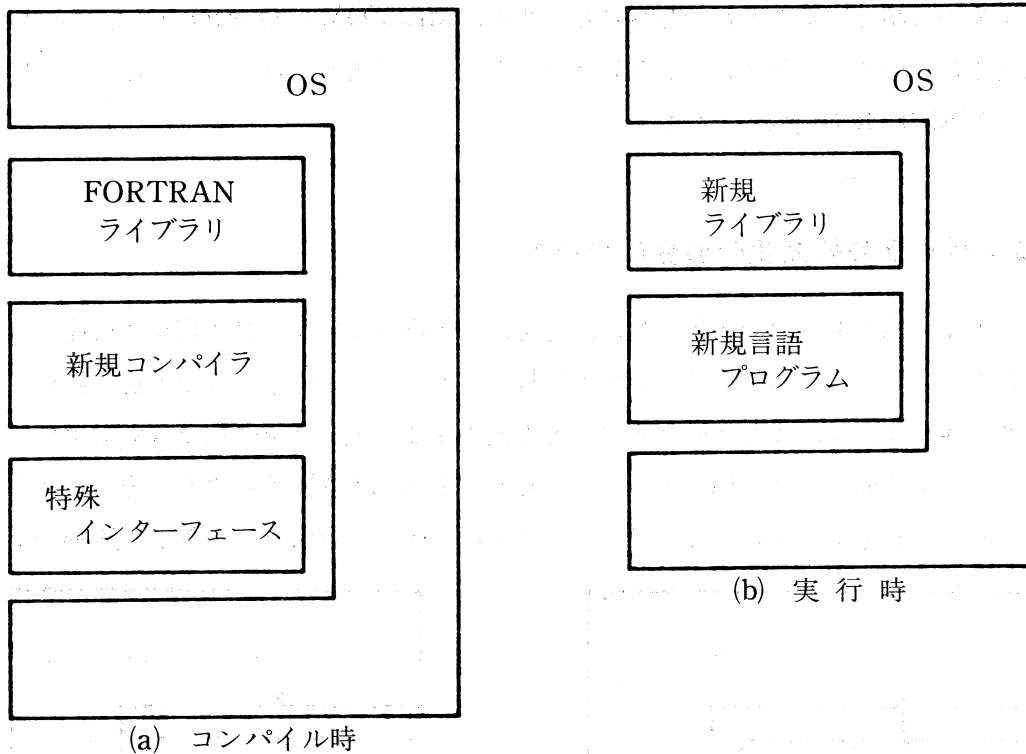


Fig. 2 実用版の言語処理系の例

通常コンパイラを小さくしようと考えれば、コンパイラのロードモジュール中に無駄なデータ領域を持ち込まず、必要になった時点で動的に領域拡張を行なえばよい。MELCOM UTS/VSの下ではロードモジュールは Fig. 3 のように記憶割付が行なわれる。OS に対し、M:GP 又

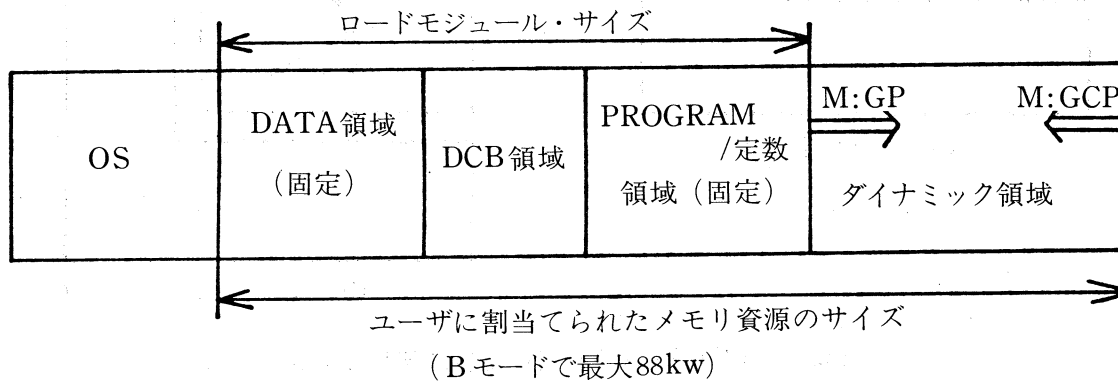


Fig. 3 UTS/VSのメモリ割付け

は M:GCP マクロ命令を出せば、図のダイナミックデータ領域から動的に空間が確保できるので、ここを記号表などに用いればよい。

② CPU タイマの設定

コンパイル時の CPU 時間を測定するためのタイマ設定である。(文献(6)参照)

③ 処理モードの認識

多目的の OS の下では、バッチ処理、TSS 処理などいくつかの処理モードでコンパイラが

動作する。特に処理モードを意識しないでコンパイラを作成してもよいが、それでは使用上いろいろな不都合が生じるので、動作時の処理モードをコンパイラに取り込む必要がある。

④ コンパイル・オプションの入力と解析

現在 COSMO-700III 上で動いている殆どどのプロセッサは、バッチ処理なら、

! processor名 オプション指定

TSS 処理ならば、

! processor in-file $\left\{ \begin{array}{l} \text{ON} \\ \text{OVER} \end{array} \right\}$ rom-file, listing-file

⋮ オプション指定

↑ プロンプト

の形式でコンパイル・オプションを受けつける。同一機種の上で動くプロセッサはできる限りその操作手順を一致させることが望ましいので、コンパイル・オプションの入力方式は結局 OS と関連づけて処理しなければならない。

⑤ CPU の例外処理条件の設定

処理系の環境要因の一つにトラップと称する内部原因の割込み処理がある。トラップには固定小数点オーバーフロー、浮動小数点オーバーフロー、アンダーフロー、スタックポインタのオーバーフロー、アンダーフローなどがある。これらが発生した時の処理ルーチンはユーザによるオウンコーディングが可能であり、各プロセッサは独自のトラップハンドラを備えている。したがって、主言語とは別のトラップ処理ルーチンを設けるとかなりやっかいなトラップハンドラの切換えを行なわねばならない。

これらの条件設定のうち①、③、④についてルーチンの例を提示する。

3. 動的メモリ領域の確保

(SUBROUTINE GETOBJ)

このルーチンは、

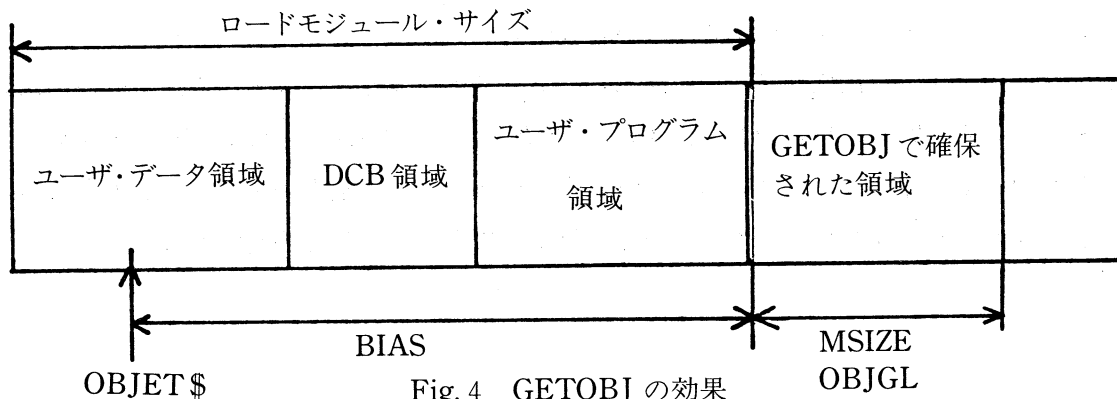


Fig. 4 GETOBJ の効果

CALL GETOBJ (必要なページ数, 異常出口)

の形式で呼ばれる。ここで、ページ数とは 512 語を単位とするメモリサイズのことである。確保されたメモリ領域が指定に満たず 1 ページ以下の場合には異常出口で指定されたロケーションへ制御を移す。このルーチンでは、Fig. 4 のようにアドレス情報を設定する。これらの変数は全て、

COMMON/COBJECT/ ~

で宣言されている。

4. 処理モードの認識及びコンパイル・オプションの入力と解析

(SUBROUTINE OPTION)

OPTION は Tab. 1 に示す 21 種類のコンパイル条件を、コンパイル・オプションの解析と OS からの情報抽出により得ている。この結果は配列 OPT に格納される。ただし、SIMPL-R ではこれらの値を動的に変更するディレクティブが用意されているので、あくまでもコンパイルを開始する時点での初期値である。このルーチンの処理概要を以下に述べる。

① 処理モードの決定：66行～76行

JIT の第 0 語に格納されている処理モードを抽出し OPT(0) を決定する。

② 印刷装置の認識と上段シフトの決定：79行～92行

外部環境で定められた印刷装置の種類を調べ、それが LP ならば小文字を大文字へ自動変換する条件 OPT(12) を設定する。小文字の扱える処理系でかつバッチ処理を許す場合には、このような配慮が必要である。

③ ソース入力装置の認識：94行～99行

TSS 端末からのソース入力では、ソースリスティングが不要であるから OPT(19) に入力装置の種類を記録しておく。

④ コンパイル・オプションの入力：102行～128行

処理モードに従って標準的なオプション指定の入力を行なう。バッチ処理では制御カードを M: C DCB 経由で再読込みしなければならない。

⑤ コンパイル・オプションの解析：130行～205行

各オプション指定を内部サブルーチン GETTKN で取り出し、それぞれの意味に従った処理を行なっている。特に行 191～205 では F: 80 DCB にファイル名を登録し、ソース・プログラムの写しを出力できるようにしている。これは TSS 処理で直接端末からプログラムを入力し、取りあえずコンパイルして後にエラー修正を行なう場合を考慮した機能である。

Tab.1 コンパイラ・オプション設定配列 OPT の構成

番号	意 味	設 定 条 件
0	処理モード	<u>0</u> = バッチ処理, 1 = TSS 処理
1	ソース・リスティング	<u>0</u> = なし, 1 = LS *
2	オブジェクト・リスティング	<u>0</u> = なし, 1 = LO *
3	サマリ・リスティング	<u>0</u> = なし, 1 = SUM *
4	マクロ・リスティング	<u>0</u> = なし, 1 = ML *
5	インクルード・リスティングなし	<u>0</u> = 有, 1 = NIL *
6	ソース・アウトプット	<u>0</u> = なし, 1 = SO (ファイル名指定) *
7	実行制御	{ <u>0</u> = コンパイル・エラーがなければ実行 1 = GO (無条件実行) -1 = CO (コンパイルのみ) } *
8	ライン・トレース	<u>0</u> = なし, 1 = LT *
9	call トレース	<u>0</u> = なし, 1 = CT *
10	case チェック	<u>0</u> = なし, 1 = CC *
11	添高チェック	<u>0</u> = なし, 1 = SC *
12	上段シフト	<u>0</u> = なし, 1 = UC *
13	入力行の長さ	<u>80</u> 字
14	コンパイラのモンタリング	<u>0</u> = なし, 1 = MON *
15	静的統計量収集	<u>0</u> = なし, 1 = SS T *
16	実行時統計量収集	<u>0</u> = なし, 1 = EST *
17	マクロ展開のリスティング	<u>0</u> = なし, <u>1</u> = 有
18	オブジェクト・エリア・サイズ	<u>8</u> KW (値指定) *
19	ソース・入力	<u>0</u> = file, 1 = ME
20	出力装置	<u>0</u> = file, 1 = LP, 2 = ME
21 ┆ 23	予 備	

注) 設定条件の は default 条件, * はオプション指定による設定を示す。

プログラム登録表

(*欄を作成者が記入する)

ライブラリー名	(センター記入)	作成日付*	1981. 12. 22.
プログラム名*	GETŌBJ, ŌPTIŌN		
作成者氏名*	木 村 宏		
形 式*	a. コンプリート・プログラム c. 関数副プログラム ②. サブルーチン・副プログラム d. その他		
使用機器*	a. 磁気テープ 入力 (1) 本 出力 () 本 b. CRTの種類 () c. 磁気ディスク・ファイル () 個 d. その他 (カード)		
プログラムサイズ*	a. ソース・プログラム (343) 行 b. オブジェクト・プログラム (0.3) KW		
使用言語*	a. FLAG d. CŌBŌL g. Lisp b. FŌRTRAN IV ③. METASYMBŌL h. PL/I c. FŌRTRAN 77 f. PASCAL i. その他 ()		
処理形態*	a. バッチ処理専用 ④. バッチ・TSS兼用 b. TSS専用		
使用条件等*	SUBRŌUTINE GETŌBJ 及び ŌPTIŌN は共に FŌRTRAN プログラムから取出して使えるようになっている。 但し, CŌMMŌN 領域及び, DCB名が固定されているので注意してほしい。また, JIT の先頭番地を EQU で設定しているのも、もし ŌS の Version が変われば変更しなければならないかも知れない。なお, ここで言う FŌRTRAN は MELCŌM の FŌRTRAN IV であり, FLAG や FŌRTRAN 77 とはインターフェースが取れない。		

5. おわりに

C言語処理系の開発に着手しながら、事情により SIMPL-R 処理系の開発が先行することになった。今後は早急に C 言語処理系が公開できるよう研究を続けたいと考えている。

なお今回報告したルーチンは SIMPL-R の開発が進展すると共に機能追加を行なう予定である。

【参考文献】

- 1) Kernighan, B. W. and Ritchie, D. M., The C Programming Language, Prentice-Hall (1978).
- 2) Zahn, C. T., C Notes: A Guide to the C Programming Language, Yourdon Press (1978).
- 3) 小川保二：C言語の特徴と機能, インターフェース No. 45 (1981).
- 4) Basili, V. R. and Turner, A. J., SIMPL-T, A Structured Programming Language, Univ. of Maryland, Computer Science Center (1974).
- 5) 原田賢一, 他：コンパイラ作成用の構造的プログラミング言語：SIMPL-T, 情報処理 17-3 (1976).
- 6) 木村宏：CPU 使用時間を取り出すサブルーチンについて, 岡山理科大学情報処理センター, Technical Note, OKLIB-018 (1978).
- 7) UTS/VS システム・プログラミング説明書, 三菱電機, NM-SR01-36A <06A1>
- 8) UTS/VS システム・マクロ文法説明書, 三菱電機, NM-SR01-28A <9XA0>

13. SL-1言語による通信系のシミュレーション

課題番号：50513

電子理学科助教授 宮 垣 嘉 也

” 助 手 小 西 憲 一

上記課題に関する研究を予定していたが、解析的に解くべき問題が発生し、他の研究情勢、緊急度、重要さなどを考えた結果、上記課題は後まわすべきであると考えたので、それに関しては、SL-1 言語文法の修得と簡単な例題による練習を行ったにとどまり報告すべき成果はない。

14. クラマース・クローニツヒの関係式によるフォノン解析

課題番号：50514

基礎理学科助教授 若 村 国 夫

” 助 手 財 部 健 一

測定された反射率, $R(\omega)$ を入力データとし, 表記関係式 ($k-k$ と略す) を用いて, 誘電分散, 分散定数を決める。主として格子振動域 (約 $50\mu\text{m} \sim 10\mu\text{m}$) の波長範囲を扱う。

誘電体界面で反射が生じると, 入射電場と反射電場との位相差, $\theta(\omega_0)$ が生じ, それは $k-k$ 式より, $R(\omega)$ を DATA とし,

$$\theta(\omega_0) = \frac{1}{2\pi} \int_0^{\infty} \ln \left(\frac{\omega + \omega_0}{|\omega - \omega_0|} \right) \frac{dR(\omega)}{d\omega} \cdot d\omega$$

と求められる。 ω , ω_0 は波数単位で測った電場の周波数である。 $\theta(\omega_0)$ がわかると, 例えば屈折率, $n(\omega_0)$ は

$$n(\omega_0) = \frac{1 - R(\omega_0)}{1 + R(\omega_0) - 2\sqrt{R(\omega_0)} \cos \theta(\omega_0)}$$

と計算できる。

今回のプログラム作成はこれらの結果とグラフィック・ディスプレイを結合し, 視覚的にデータが処理できるようにした。

プログラム登録表

(*欄を作成者が記入する)

ライブラリー名		作成日付*	1981. 1. 2.
プログラム名*	SEARCH FOR BEST FITTING PARAMETERS		
作成者氏名*	教養部会講師 波 田 善 夫		
形 式*	a. コンプリート・プログラム c. 関数副プログラム b. サブルーチン・副プログラム (d). その他		
使 用 機 器*	a. 磁気テープ 入力 (1) 本 出力 () 本 b. CRTの種類 (GRAPHIC DISPLAY) c. 磁気ディスク・ファイル () 個 d. その他 (カード)		
プログラムサイズ*	a. ソース・プログラム (654) 行 b. オブジェクト・プログラム () KW		
使 用 言 語*	a. FLAG d. CŌBŌL g. Lisp (b). FŌRTRAN IV e. METASYMBŌL h. PL/I c. FŌRTRAN 77 f. PASCAL i. その他 ()		
処 理 形 態*	a. バッチ処理専用 (c). バッチ・TSS兼用 b. TSS専用		
使 用 条 件 等*			

15. “Spectrum-wise”なsuperconvergence algorithm に関する研究

課題番号：50515

応用数学科情報専攻教授	仁	木	滉
”	”	講師	沢 見 英 男
”	”	”	池 内 雅 紀

本研究は、多次元線固有値問題 $Lu = \lambda Mu$ (L, M は線形偏微分演算子) の、 n 次試行関数を用いた有限要素法にもとづく計算法の改良を目的とするものである。本年度は、簡単な Sturm-Liouville 形方程式をもとに、superconvergence (超収束) を与える1次試行関数および3次試行関数の導出を行なった。しかしながら、それらの isoparametric 要素化、数値積分等について更に考察する必要がある、本研究に関する成果を発表する段階には至っていない。

2. 効率的解法の概要

与えられた条件式の個数を m 、変数を n とすると、実行可能領域での端点の個数は $(m+n)$ 個となる。いま LP 問題を退化 (degenerate) てないものと仮定すると、少なくとも平均的には $\frac{1}{2}(m+n)$ 回の軸操作により最適解を求め得るはずである。しかしながら、 m と n とが可成り大きなものとする、1 回の軸操作による解の改善のための計算時間も相当なものとなる。そこで何らかの方法で最適解の近くの端点まで到達できれば、それ以降単体法により最適解を求め得るはずである。そこでつぎの LP 問題解法手順を考える。

- ① 最適解の近傍実行可能解を探索。
- ② 実行可能解の改善→最適解 (単体法)

①の最適解の近くの実行可能解を与える端点は必ず目的関数の傾斜勾配により等しい条件式を与える超平面が存在するはずである。すなわち、目的関数を与える超平面と最適解およびその近くの端点を構成する条件超平面とのなす角はその他の端点を与える超平面とのなす角より小さいはずである。LP 問題を

$$\max\{Z=Cx \mid Ax \leq b, x \geq 0\}$$

として表示し、 $C=(c_1, c_2, \dots, c_n)$ 、 $A=(a_1, \dots, a_m)^T$ $a_i=(a_{i1}, a_{i2}, \dots, a_{in})$ 、 $i=1, 2, \dots, m$ とする。超平面 $Z=Cx$ と $a_kx=b_k$ とのなす角 θ は $\cos \theta = C \cdot a_k / \|C\| \|a_k\|$ として与えられる。 $\cos \theta_k = \max\{\theta_i\}$ 、 $i=1, \dots, m$ となるような超平面 $a_kx=b_k$ 上には必ず最適解の近くの端点が存在するはずであるから、この超平面上を実行可能領域まで移動して実行可能解を求め得る。

3. アルゴリズムのまとめ

- ステップ1 主制約条件方程式 (目的関数とのなす角度の最小のもの) の決定。
- ステップ2 主制約条件式上で各変数の最大カットを与えるものを軸要素として掃き出す。
- ステップ3 実行不可能度の最大値を与える制約式上で、目的関数を最小限に減少させる変数を選んで掃き出す。
- ステップ4 軸操作の結果、実行可能解であればステップ5へ。そうでなければステップ3を繰り返す。
- ステップ5 実行可能解が最適解となっているかどうか調べ、最適解でなければ単体法の改善ルーチンを用いて最適解を求める。

4. 結果の検討

このアルゴリズムは変数の数に比し条件式が多数個存在する様な LP 問題で、しかもかなり大規模な問題程有効である。問題の種類によっても異なるが32条件式16変数では単体法で11ステ

ップ、新アルゴリズムで6ステップの差が出ている。しかし、主要制約式決定ステップの繰り返し計算を加算すると $\frac{2}{3}$ 程度の計算時間といえる。唯問題の構造によっては主要制約式決定に要する計算時間を考慮すると返って非効率な場合もあり、また大規模な問題に対して実行不可能領域から実行可能領域に至るまでの繰り返しにかなりの計算時間を要する等の現象もあるので今一步の改良の必要がある。ただし、従来の単体法アルゴリズムと並列に使用可能なプログラムとしているので、現在のところ300条件式×400変数までのLP問題は十分にその単体表もプリント可能なプログラムとなっているので、研究上御利用頂けると幸いです。

岡山理科大学情報処理センター

岡山市理大町1-1

TEL (0862) 52-3012