

電子計算機の気象学ならびに地球物理学への 応用についてのシンポジウム (I)

日 時 : 1958年7月18日 (金)
場 所 : 気象庁第1会議室
司 会 : 正 野 重 方 (東大)

第 1 部

今 村 茂 雄 (日本 IBM) : IBM 704の性能について
磯 部 谷 郎 (気象庁) : IBM 704の簡易使用法
都 田 菊 郎 (東大) : 数値予報プログラム JM S001について
増 田 善 信 (気象研究所) : JM T001 について

第 2 部

横 関 徹 (日本航空) : 電子計算機の航空気象への応用 (最短時間飛行について)
片 山 昭 (東京管区気象台) : 電子計算機の輻射への応用
竹 内 清 彦 (気象庁) : 大気乱流研究への電子計算機の利用
藤 田 敏 夫 (気象研究所) : 気象統計の立場から
朝 倉 正 (気象研究所) : IBM 704をめぐる長期予報
長 尾 隆 (研修所) : 気候学に対する電子計算機の応用
駒 林 誠 (東大) : 降雨理論と計算機
小 平 吉 男 (東京管区気象台) : 水理気象と電子計算機
宮 崎 正 衛 (気象庁) : 潮汐の問題
南 日 俊 夫 (気象研究所) : 海洋力学と電子計算機
湯 村 哲 男 (気象庁) : 地震業務への利用
竹 内 均 (東大) : 地球力学と計算機

司会者挨拶: 今月の例会は数値予報関係ということになっています。来年3月には704が気象庁にはいることになり、関係者はその準備に忙殺されています。1949年の春、丁度翌年からアメリカでJNWPUが発足しようというので準備中の頃とよく似ていますが、UCLAとNSF (National Science Foundation) との協催で、電子計算機の気象学及海洋学への応用というコンフェレンスが開かれたことがあります。これは当時の状態では大変有効であったように思います。IBM 704は高性能でありますから、数値予報の現業に使った外にも計算時間が残ると考えられますが、気象学、海洋学その他地球物理学などの研究に利用できる可能性がありますし、各分野の研究者もこれを利用しようと準備されておられることと思います。そこで電子計算機はいると各分野ではどの

ような問題に使えるか又どのようの有効であるか又はたいていして有効でないかなどといろいろの問題について各分野の研究者が互に意見を交換すれば、お互に裨益する所が多いかと思えます。そこで本日のシンポジウムは「IBM 704をめぐる」という気持ちで各分野の方々からご意見を伺ったらどうかと思います。進行の都合上初めにIBM 704の話や現在数値予報ではどのようなプログラムができてどのように準備中であるかを話して頂き、次に各分野のご意見やご抱負を伺うことにしたいと思えます。

第 1 部

正野: 最初に日本IBMの今村さんに『IBM704について』の話を伺います。IBM 704とはどんな性能をもった

機械か、どれ程の計算ができるか、あるいは、既にどんな方面に利用されているかなどについてお話があることと思います。次に磯部さんから「FORTRAN」について解説して頂きます。FORTRAN というのは計算プログラムを組む方法で、アメリカでもやっとなん年あたりから実用になり始めたようであります。一般に計算プログラムを組むことは大変な仕事であります。基本的には計算機用の symbolic な文字で目的とする計算の過程を書き下すのでありますが、FORTRAN では数式を書くのと殆んど変わりなくでき、大変便利なものであります。そこで、本来ですと、最初に symbolic な組み方の話をして頂いて、次に FORTRAN のやり方の話に進みますと、その差が明瞭になりますが、こゝでは時間もありませんから、前者の解説は別の機会に譲ることにして、ここでは FORTRAN の話を磯部さんからして頂くことにします。これによって計算プログラムはどのようにして組むものであるかを理解することにします。

IBM704 の性能について

今村 茂 男*

明年初春、気象庁に IBM 704 型電子計算機が導入されることは、すでに各方面より注目されており導入後の成果が大いに期待されている所であります。電子計算機には、大別してデジタル型（計算をソロバンのように行うもの）とアナログ型（計算尺のように行うもの）との2種類があって、それぞれの使用目的に応じて使い分けられておりますが、前者によれば、精度の極めて高い計算ができ、数字の組合せにより文字・記号も取扱えるし、論理演算とか翻訳のような事もできるので、デジタル型の方が万能計算機として将来の発展の可能性が大きいわけであります。

IBM 704型も、このデジタル型に属する科学技術用の計算機（もちろん一般事務計算も取扱いうる）で、現在世界中にすでに90台以上が稼働中であります。最も広く用いられているため、計算のためのプログラム（計算機に与える命令順序）や計算方法等が豊富に準備され、使用者相互に交換されております。

1. 数の表現

704 型では数字・文字・記号を取扱っていますが、これは読みこみ、取り出しの時だけで、機械の中ではすべて2進数によって表わしております。したがって通常、

数は10進数で読みこみ、これを2進数に変換しているわけですが、2進数は10進数と異なり0と1とから成り立っていますので、機械内部の演算素子、記憶素子の ON、OFF あるいは +、- という2つの状態で表わすことができるわけですが、数に限らず、たとえばロシア語→英語の翻訳の場合、ロシア語も各単語が2進数におきかえられて記憶され、計算機によって論理的な処理をうけて、英語に相当する2進数が作られあるいは選ばれて記憶され、印刷される時に初めて実際の英語に戻るわけであります。

2. その構成と働き

電子計算機の構成はは大体次の4つの部分から成り立っています。(1) 制御装置 (2) 演算機構 (3) 記憶装置 (4) 入力・出力装置・機械の作動の順序に従って簡単に述べますと、先ず予め計算の方式、順序を機械に命令するプログラム（命令順序）を作成しておき、これと計算に必要なデータ（数値・文字・記号）を入力装置より記憶装置に入れます。記憶装置に入ったプログラムをその順序に従って一つ一つ制御装置に取り出して何の計算を命令しているかを解釈し、この命令に従い記憶装置に入っているデータを演算機構に取り出して計算を進め、一つの計算が終ればまた次の命令を読みとり解釈してデータを計算する……これを繰り返して全部の計算が終れば、結果を（命令によって）出力装置から取り出して印刷するという順序です。従って同じ計算式を取扱う限りはプログラムを変える必要はなく、次々と新しいデータを読み込ませればその結果が続々と印刷されて出てくるわけですが、このように計算のための命令も、計算されるデータも一緒に記憶装置に貯えておくという考え方（これをストアード・プログラム方式という）が現在のデジタル型電子計算機の特長であります。この考え方はフォン・ノイマンによって案出されたといわれております。この命令も実際には数字の形（2進数）で与えられておりますので、命令と数そのものを加えたり、命令と命令とを加え合せて新しい命令を作り出すということも可能です。論理演算はこのような仕組みを利用しているわけですが。

3. 記憶の容量

704型の記憶装置としては(1) 磁気コア・マトリックス (2) 磁気ドラム (3) 磁気テープを採用しています。記憶の単位としては、正負符号および35桁の2進数を以て1単位とし1語と呼んでいます。10進数に直すと、固定小数点の数で約10桁強、浮動小数点の数で約8桁×

* 日本 IBM

(10の±約38乗)の数を表わすこととなります。(1)磁気コア・マトリックスでは、4,096語ないし32,768語を記憶し、(2)磁気ドラムでは8,192語ないし16,384語を記憶します。(3)磁気テープは記憶装置であると同時に、入出力装置としても使用されます。テープの記憶容量は33 $\frac{1}{2}$ 語/インチ(テープの最長は2400フィート)で、その書き込み・読みとり速度は毎秒2,500語という高速度です。

4. 入出力装置

入出力には通常、穿孔カードあるいは磁気テープが用いられますが、出力としてはさらに印刷という形でも取り出せます。通常は毎分18,000字の印刷速度ですが、磁気テープにいったん書き込んで高速度印刷機を使用する場合には毎分60,000字ないし120,000字という高能力が可能となります。印刷は単なる数字の羅列または数表の形に止まらず、機械内部の2進数の結果をプログラムによってアナログ的に変換し、文字・記号によって図表や天気図の形に直して印刷することも可能であります。

5. 計算の速度

704型による計算速度は極めて早く、たとえば固定小数点方式・2進数35桁の計算では、毎秒、加減算41,700回、乗除算4,170回判断41,700回、という驚異的速さであります。我々が中学時代に5元方程式ぐらいの宿題でよく悩まされたものですが、10元方程式を解くのに0.222秒、100元方程式を解くのに(人生僅か50年ではとうてい解けきれまいに)僅かに2分24秒を要するのみであります。このような速い処理能力によって単に計算のみに限定されず、龐大なデータの処理、論理演算、翻訳等にも応用されるようになったわけであります。

6. 電子計算機の応用と将来

1954年1月にはアメリカのジョージタウン大学を中心とするグループが701型計算機(704型の前身)を使ってロシア語→英語の翻訳に試験的に成功しております。1958年1月バリの科学計算研究所では704型により円周率の計算を小数点以下1万桁まで僅か1時間40分で成し遂げております(最後の4桁は5678と出た由であります)。国際地球観測年の協定に基き、アメリカ・ワシントンに世界計算センターが設けられておりますが、ここでも704型が活躍しております。(気象庁のプログラムは現在ここにおいてテスト中であります。)また最近では学術文献を704型によみこませて自動的にそのアブストラクトを作成させるとか、古代の文典を705型(704型と同系統の事務計算用電子計算機)にかけて現代語訳を行い、同時

に辞書を作成する等、意欲的な作業が行われつつあります。このように計算機は単に計算のみに止まらず、広い意味でのデータ処理機械としての傾向をますます強めていくことでありましょう。翻訳機械という子供も生まれてまいりましょう。現にアメリカの原子力研究所ではストレッチと呼ぶ巨大な計算機を作るために704型が使われているという面白い例も見受けられます。

計算のためのプログラムが豊富にでき上ればやがてはプログラム・ライブラリーという全く新しい形の図書館ができ上り、各所にできる計算センター相互の連絡も緊密になり学術の交流がますます盛んになりましょう。

IBM704 型電子計算機の簡易使用法 (FORTRAN の概略)

磯部 谷 郎*

IBM社の704型電子計算機は、計算速度を高めるために、2進法を使って計算が行われる。したがって、計算に用いる数値はもちろん、計算順序を計算機に伝える命令も、2進法に直して記憶装置の中へ読み込まねばならない。FORTRAN (mathematical formula translating system) は、ある表現規約にしたがって、計算機にさせたいと思う内容を書き並べ、カードにパンチし、それをFORTRAN マスター・テープと呼ばれる1本の磁気テープに納められている命令でカードリーダを通じて読み込ませ、解釈させ、2進法の命令を自動的に組み立てさせると言う非常に便利な機械的命令組立法である。FORTRAN は普段使い慣れている言葉や10進法の数値を2進法の命令や数に変換すると言うだけでなく、1枚のカードにパンチした1行の言葉から幾つかの命令を作り出してくれるもので、計算機使用者にとってはまことに有り難いプログラミングの方法である。簡単な例によって、その説明を試みよう。

2次方程式の解

$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

を計算することを考えて見る。

$$A=3, B=10, C=-1$$

と仮定すると

$$x_1 = +0.09716753 \dots$$

$$x_2 = -3.43050086 \dots$$

となる筈である。この計算をもし704型電子計算機にさ

* 気象庁予報部予報課

せるとすると、次のような 8 個の命令文章 (program statement) が必要である。

```
A= 3.0
B=10.0
C=-1.0
X1 =(-B+SQRTF(B**2.0-4.0*A*C))
                                     /(2.0*A)
X2 =(-B-SQRTF(B**2.0-4.0*A*C))
                                     /(2.0*A)

PRINT 100, X1, X2
100 FORMAT (1H+2E12.4)
STOP
```

見てすぐわかるように、* は掛算記号を表し、1 は割算記号を表わす。また $B**2.0$ は B^2 を意味している。SQRTF は square root function の意味で、それに続く括弧内の値の平方根を求める計算順序通りの命令を作り出すための記号である。A, B, C, X1, X2 は変数を表わし、それぞれのとるべき値を貯える場所が計算機の磁気コアの中に定められる。

$A=3.0$ は A に割り当てられた場所に、3.0 を貯えるような命令を作れと指示する命令文章である。もちろん 2 進法に直される。= はいつも、その左辺の変数の記憶場所に右辺を計算した値を貯えることを意味し、= で表わした命令文章を算数式 (arithmetic formula) と呼ぶ。右辺は計算機の理解し得る記号である限り、いくら長く並べてもよい。2 枚以上のカードにわたってパンチしても差し支えない。

算数式の形でさきに定義しておくならば、どんな関数形でも表わすことができる。しかし極く基本的な下記の関数は定義しなくても FORTRAN の中に組みこまれて作られている。

SQRTF	Square root function
SINF	Sine function
COSF	Cosine function
LOGF	Logarithmic function
EXPF	Exponential function
ATANF	Arctangent function
SINHF	Hyperbolic sine function
COSHF	Hyperbolic cosine function
TANHF	Hyperbolic tangent function

これらの他に、絶対値を求める ABSF, 整数部分を求める INTF, $A - \left(\frac{A}{B}\right)$ の整数部分 $\times B$ の値を求める MODF, (A, B), 最大値を求める MAXIF, 最小値を求

める MINIF 等その他若干が用意されている。

上例の 5 番目の命令文章までで X1, X2 の値をコアの中に貯えるまでの命令が作り出される。次の PRINT は附属のプリンターで X1, X2 の値を印刷する命令を作り出す。FORMAT は印刷する文字の配列や数字の桁数を指定するための命令文章である。上例中の FORMAT の左の 100 という数字は 4 桁以内の正の整数ならば何でもよい単なる記号である。PRINT の命令文章の中で使用したい FORMAT を指定するのに使う命令文章の番号である。

FORMAT の括弧内の記号の規定を書くときと長くなるので、この例についてだけ述べると、1H はその次にプリンターの行送りを指定する記号が 1 字あることを示す。+ はプリンターの行を変えないで印刷せよという記号である。E12.4 は、コアの中の浮動小数点で表わされている数を浮動小数点のままプリンター上に 12 字分の場所を使って、小数点以下 4 桁まで印刷せよと指定する。E の前にある 2 は、同じ形式の数が二つ印刷されることを示している。今の場合は X1 と X2 が

$$0.9717E-01 \quad -0.3431E 01$$

と印刷されてくることになる。

STOP はプログラムが終了するとき計算機を停止させる命令を作る命令文章である。

この 8 枚のカードが FORTRAN マスター・テープの中の命令で処理されると card recorder または card punch と呼ばれる装置で 2 進法の命令となってパンチされて来る。その何枚かのカードをもう一度 card reader に載せ START というボタンを押すと、この命令は計算機のコアの中に読み込まれ、読み終ると第一の命令から行われて X1, X2 の値がプリンター上に印刷されるまで進行して停止する。

この例の程度では手で計算した方が速いし、経済的でもあるが、次の例のように 100 組位になると電子計算機の価値が幾分現われよう。

A, B, C が 100 組あると $A=3.0$ の形の命令文章が 300 必要になる。このような場合は次のような方法をとる。

```
DIMENSION A (100), B (100), C (100),
                                     X1(100), X2(100)
READ 1000, A, B, C
1000 FORMAT (7F10.4)
X1 (I) = (- B (I) +SQRTF (B (I) ** 2.0 - 4.0
                                     *A(I)*C(I)))/(2.0*A(I)), I=1, 100
```

```

X2(I) = (-B(I) - SQRTF(B(I)**2.0 - 4.0
      *A(I)*C(I)))/(2.0*A(I)), I=1, 100
PRINT 2000, X1(I), X2(I), I=1, 100
2000 FORMAT(1H+2E12.4/(1H02E12.4))
STOP

```

DIMENSION という命令文章で *A, B, C, X1, X2* に夫々 100個の場所がコアの中に用意される。

READ 1000, A, B, C は命令文章の番号 1000 の *FORMAT* にしたがって、パンチしてあるカードをカード・リーダーを通じて読み *A, B, C* のため用意された場所に貯えるまでの命令を作り出す。

2000 の *FORMAT* の / は行を変えることを意味し、1H0 の 0 は 1 行おきにプリントすることを指示し、(1H0 2 E 12.4)の括弧は第 3 行以下皆この記号によることを意味する。その他は前例と全く同じであるが、ただ *I=1, 100* は *X1(I)* を *I=1* から *I=100* まで繰り返し計算することを示し、*PRINT* のところの *I=1, 100* も *X1, X2* を 100 行繰り返して印刷することを意味する。

命令文章には次の 32 種がある。

算数式 (1 種)

$$a = b$$

配列用命令文章 (3 種)

DIMENSION, EQUIVALENCE, FREQUENCY

入出力用命令文章 (13 種)

READ, READINPUTTAPE, PUNCH, PRINT, WRITE OUTPUTTAPE, READ TAPE, READ DRUM, WRITE TAPE, WRITE DRUM, FORMAT, END FILE, REWIND, BACKSPACE

制御用命令文章 (15 種)

IF (a), IF (SENSE LIGHT), IF (SENSE SWITCH),

IF ACCUMULATOR OVERFLOW, IF QUOTIENT OVERFLOW, IF DIVIDE CHECK, SENSE LIGHT, GO TO (3 種), ASSIGN,

PAUSE, STOP, DO, CONTINUE

これらについて一つ一つ述べる余裕はないが、次に *DO, IF, GO TO, CONTINUE* を含む簡単な例を一つ挙げて見る。

```

DIMENSION A (100), B(100), X (100)
READ 1, A, B

```

```
1 FORMAT (10F7.3)
```

```
DO 30 K=1, 100
```

```
IF (A(K)-B(K)) 10, 20, 20
```

```
10 X(K)=SQRTF (B(K)-A(K))
```

```
GO TO 30
```

```
20 X(K)=SQRTF (A(K)-B(K))
```

```
30 CONTINUE
```

```
PRINT 1000, X(K), K=1, 100
```

```
1000 FORMAT (1H+E11.4/(1H0 E11.4))
```

```
STOP
```

DO 30 K=1, 100 は命令文章番号 30 の *CONTINUE* までを *K=1* から 100 まで繰り返すことを意味する。*IF(A(K)-B(K)) 10, 20, 20* は *A(K)-B(K)* が負なら 10 の命令文章に移り、0 または正なら 20 の命令文章に移ることを意味する。*GO TO 30* は無条件に 30 の命令文章に移り、*CONTINUE* は単なる接ぎの命令文章で、これを経て *DO* の命令文章にもどり、ループで 100 回繰り返されることになる。

この例の場合には、*DO* 以下 *CONTINUE* までのかわりに

$$X(K) = \text{SQRTF}(\text{ABS}(A(K) - B(K))),$$

$$K = 1, 100$$

としても同じ結果になる。

FORTRAN を使うときの細かい規程は一冊の印刷物になって *IBM* 社から発行されている (*FORTRAN Programmer's Reference Manual*). *FORTRAN* を使用する場合には、もちろんその本を熟読する必要がある。