

# EQUATRAN-Mの機能と構造

小口 梧郎・横山 克己

この連載では、化学工業にみられるさまざまな数値計算上の問題を、方程式解法ソフトを用いて解いてきた。方程式解法ソフトの有用性は十分ご理解頂けたと思うが、その中心として取り上げた EQUATRAN-M に興味を持たれた方もいると思う。今回は、この EQUATRAN-M をひとつのソフトウェアとして見た場合の操作面や内部処理について解説する。

## プログラムの構成と機能

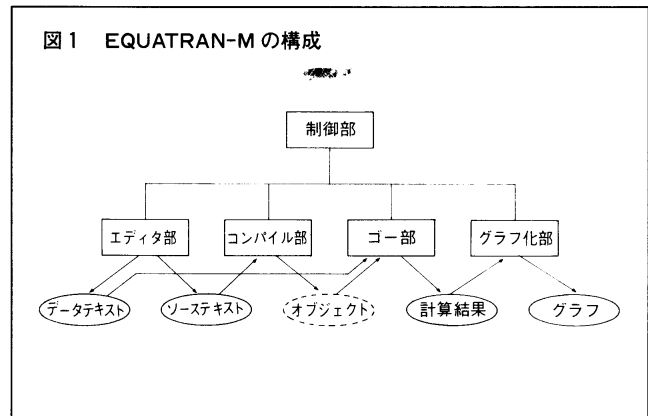
まずは、EQUATRAN-M の大まかな構成とその働きについて説明しよう。

EQUATRAN-M は、図1のように5つの部分から構成されている。各部の上位に位置する制御部は、プログラム全体の流れをコントロールしている。一番左のエディタ部は、EQUATRAN-M に解くべき方程式やデータを入力するためにテキストを編集する部分で、スクリーンエディタを内蔵している。EQUATRAN-M では、方程式を記述したものをソーステキスト、入力データを記述したものをデータテキストと呼んでおり、それぞれ別々にエディットすることができる。次のコンパイル部は、入力されたソーステキストを構文解釈して方程式の構造を解析し、計算順序を決定したのちオブジェクトと呼ぶ中間コードに変換する。この部分が EQUATRAN-M のいわば心臓部で、この内部処理については後ほど説明する。ゴー部では、このオブジェクトに従って(必要ならデータテキストを取り込んで)実際に数値計算を行い、計算結果を出力する。さらにグラフ化する時は、グラフ化部でファイルを介して取り込んだ計算結果のデータを加工して画面にグラフ表示する。

## 取り扱える問題と表現可能な式

EQUATRAN-M では、さまざまな問題を簡潔に記述するためにいくつかのきまり、すなわち文法を持っており、そ

図1 EQUATRAN-Mの構成



の意味では1つの言語となっている。以下に、この観点から、EQUATRAN-M ではどのような問題が扱えるのか、どのような式の表現が可能かを説明しよう。

- **線形連立方程式**：線形連立方程式は方程式を紙に書くようにそのまま記述すればよい。
- **非線形連立方程式**：非線形連立方程式も方程式をそのまま記述すればよい。一般に非線形方程式は直接解くことができないので、繰り返し収束計算が必要となる。EQUATRAN-M には、非線形部分を判別して繰り返し計算のロジックを自動的に生成する機能がある。さらに、ユーザが直接繰り返し計算を指定することもでき、その時は RESET 文を用いる。RESET 文は本連載では度々登場しているが、たとえば、第1回目の図9や第2回目の図2に見られる。
- **常微分方程式**：微分方程式の微分項はアポストロフィ(')を微分記号として、

$$\frac{dx}{dt} \text{ は } x'$$

$$\frac{d^2x}{dt^2} \text{ は } x''$$

などと表現し、高階のものもまったく同様に扱うことが

できる。独立変数(この場合は t)は INTEGRAL 文で指定し、積分手法は 3 種類の中から選択できる。連載の第 3 回・図 8, 第 4 回・図 2 などを参照されたい。

- 最適化計算**：これは、1 つあるいは複数の独立変数の値を、1 つの評価変数の値を最大あるいは最小とするように決定する問題であるが、このような計算をするときは FIND 文を用いて指定する。第 5 回・図 1 にその例が示されている。

次に、式の表現上の文法について説明する。

- 配列変数**：EQUATRAN-M では、1 次元と 2 次元の配列変数を用いることができる。これはベクトルとマトリクスとに相当するが、VARIABLE (省略型 VAR) 文によってあらかじめその要素数を定義しておく(第 1 回・図 10 などを参照)。EQUATRAN-M においては、配列どうしの演算は対応する要素間の演算として扱われ、結果も同じ大きさの配列となる。
- 数表**：工学分野では変数間の関係が図表で与えられるケースが多いが、EQUATRAN-M では 2 次元までの数表を定義して方程式の中で引くことができる。たとえば、第 2 回目の図 3 にその例がある。
- 条件付の式**：式自体が条件により場合分けされている式、たとえば、

$$y = \begin{cases} 2x + b & (x < 0 \text{ のとき}) \\ b & (0 \leq x \leq 1 \text{ のとき}) \\ 0.5x + b - 0.5 & (x > 1 \text{ のとき}) \end{cases}$$

のような式を条件付の式と呼んでいるが、EQUATRAN-M では予約語 WHEN を用いてそのまま表現可能である。第 2 回目の管路網の問題で、管摩擦係数 f の計算はレイノルズ数によって適用する式を使い分けている。

- 組み込み関数**：表 1 に EQUATRAN-M で利用できる組み込み関数の一覧を示した。
- パラメータ**：ソーステキストの一部(たとえば配列変数

表 1 組み込み関数一覧

EQUATRAN-M の表記	内 容
exp (x)	$e^x$
exp 10 (x)	$10^x$
sqre (x)	$x^2$
loge (x)	$\log_e x$
log 10 (x)	$\log_{10} x$
sqrt (x)	$\sqrt{x}$
sin (x)	$\sin x$
cos (x)	$\cos x$
tan (x)	$\tan x$
asin (x)	$\sin^{-1} x$
acos (x)	$\cos^{-1} x$
atan (x)	$\tan^{-1} x$
sinh (x)	$\sinh x$
cosh (x)	$\cosh x$
tanh (x)	$\tanh x$
asinh (x)	$\sinh^{-1} x$
acosh (x)	$\cosh^{-1} x$
atanh (x)	$\tanh^{-1} x$
sum (y)	$\sum_{i=1}^n y_i$
prod (y)	$\prod_{i=1}^n y_i$
max (a, b, ...)	最大の変数値
min (a, b, ...)	最小の変数値
maxof (y)	$\max_i y_i$
minof (y)	$\min_i y_i$
int (x)	整数値に切り捨て
mod (x, y)	x/y の剰余
sign (x, y)	符号の取り出し
if (x)	論理値化
abs (x)	x

の添字など)を仮のシンボルで表しておき、このシンボルに対応する内容を別に GLOBAL 文や LOCAL 文で定義できる。このようなシンボルをパラメータと呼んでいる。第2回・図3, 第4回・図14 参照。

- **マクロ**: マクロは、まとまった計算をする一群の方程式を MACRO 文と END 文ではさんでひとまとめにしてシンボル名をつけておき、これを CALL 文で随時呼び出して使用できる機能である。たとえば、第2回・図7の圧力損失の計算や第3回・図8のPIコントローラの計算など。

これ以外の文として、入出力する変数を指定する INPUT 文, OUTPUT 文, OUTPUT 1 (OUTPUT 2) 文, TREND 文がある。OUTPUT 1 (OUTPUT 2) 文はリザルトファイル(後述)へ出力する変数を指定するときに用い、TREND 文は積分過程を表示するときに用いる。

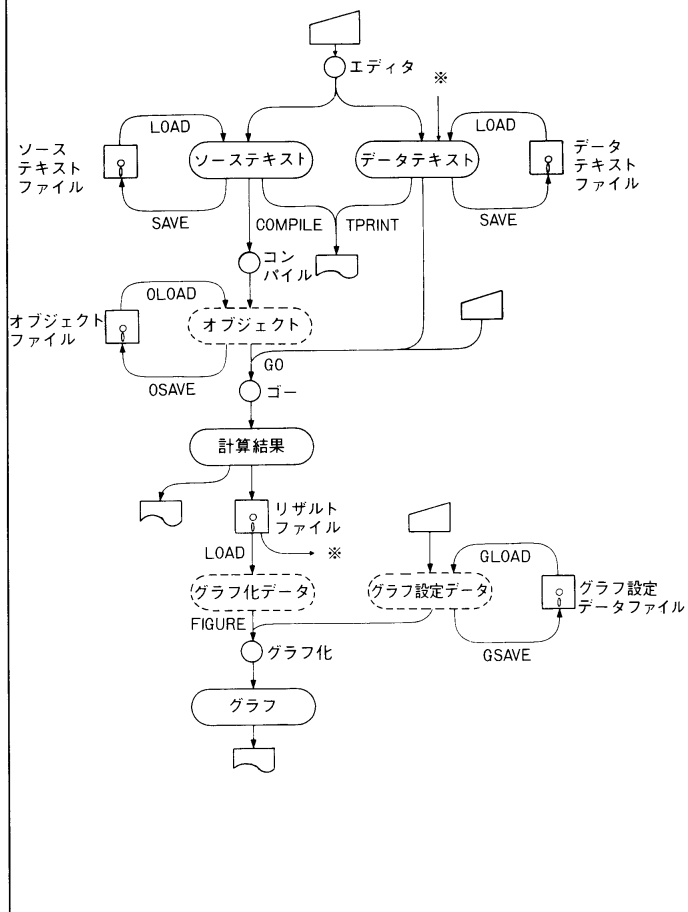
## 操作法

図2におもな操作の流れを示した。図中のコンパイルなどの○印は EQUATRAN-M の内部処理を、ソーステキストなど実線で囲んだものは実際に画面で見られるテキストやグラフを、オブジェクトなど破線で囲んだものは画面で見られないデータやコードをそれぞれ表している。また、EQUATRAN-M ではコマンドを入力して処理を指示するが、おもなコマンドを図中に書き込んでおいた。

上の方から順に説明しよう。まず、解くべき方程式群をキーボードから入力してソーステキストを作成する。この時、内蔵されているスクリーンエディタの機能を使うが、8つのコマンドと特殊キーの操作により高度な編集が可能である。図3にソーステキストを入力したときの画面のようすを示しておいた。作成したソーステキストは SAVE コマンドでディスクに保存でき、LOAD コマンドで読み込むことができる。また、リストとしてプリントアウトするときは、TPRINT コマンドを用いる。

先に説明したように、EQUATRAN-M では答えを得るのに2段階必要とする。すなわち、ソーステキストを解析し

図2 操作の流れと主なコマンド



てオブジェクトに変換するコンパイルステップと、オブジェクトに従って数値計算を行い答えを得るゴーステップである。前者を実行するコマンドが COMPILE コマンドであり、後者を実行するコマンドが GO コマンドである。また、2ステップを続けて実行するときは RUN コマンドを用いればよい。もし、この過程でエラーが発生しても、すぐにエディタの機能を使ってソーステキストを変更できるので、作業を連続して行うことができる。ゴーステップでは、計算に必要な入力データを直接キーボードから対話形式で入力で

図3 基本画面 (ソーステキストを入力したところ)

```

ソース | (S)GOKEN | (D) | (O) | (192)
-----|-----|-----|-----|-----
1: /* 5元連立方程式 */
2:
3: A + B - C + 2*D + E = 7
4: A + 6*B + 2*C - 3*D = 12
5: 2*A - B + 5*C - D + 3*E = -2
6: -A - 2*B + C - 2*E = 3
7: A + B - 5*C + D - 4*E = 1
8:
9: OUTPUT A, B, C, D, E

コマンド>F
    
```

図5 グラフ設定画面 (その1)

```

グラフ | (KYOSHIN) | (KYOSHIN) | (181)
-----|-----|-----|-----
[ タイトル・コメント ]
タイトル /* 共振回路の過渡現象 */
86/12/18

コメント (Y/N) N
横軸 (X/Y) X
横軸の長さ (%) 100
縦軸の長さ (%) 100

[ 軸・罫線 ]
変数名 軸の名称 対数 (Y/N)
X t 時間 [sec] N
Y1 i 電流 [A] N
Y2 V 電圧 [V] N

下限値 上限値 大目盛 小目盛 罫線間隔
X 0 0.04 0.01 0.002 0.01
Y1 -0.2 0.2 0.10 0.02 0.10
Y2 -200 200 100 20 0

コマンド>
    
```

図4 基本画面 (計算結果が出力されたところ)

```

ソース | (S)GOKEN | (D) | (O) | (184)
-----|-----|-----|-----
/* 5元連立方程式 */

[ 計算結果 ]
A = -1.264706
B = 3.738754
C = 3.624567
D = 5.472318
E = -2.794118

コマンド>L
    
```

図6 グラフ設定画面 (その2)

```

グラフ | (KYOSHIN) | (KYOSHIN) | (181)
-----|-----|-----|-----
[ X方向 ]
変数名
t

[ Y方向 ]
変数名 凡例内容
1 i 電流
2 V 電圧
3
4
5

凡例 (Y/N) Y
ソート (Y/N) N

線の種類
0: 線なし
1: 折れ線
2: スプライン(1)
3: スプライン(2)
4: 1次近似線
5: 2次近似線
6: 3次近似線

線のパターン
1: _____
2: .....
3: .....
4: .....
5: .....

種類 マーク 色 Y軸 (1/2)
1 1 0 6 1
2 1 2 5 2
3 1 0 5 2
4 1 0 5 2
5 1 0 5 2

コマンド>
    
```

きるほか、あらかじめテキストの形に作成しておくこともできる。これをデータテキストと呼ぶが、エディットしたりファイルにセーブしたりなど、ソーステキストと同様に扱うことができる。図4のように、得られた計算結果は画面で見ることができ、そのままをプリンタに出力することもできる。計算結果をグラフ化したいときは、ディスクにリザルトファイルとして書き出しておく。GRAPH コマンドでグラフモードに入り、LOAD コマンドでこのリザルトファイルからグラフ化するデータを読み込む。図5、図6に示したふた

つのグラフ設定画面が用意されており、どのようなグラフを表示するかを指定する。自動設定機能があるため、最小限横軸と縦軸の変数名を指定するだけでよい。もちろん、自分の好みに合わせてきめ細かな指定をすることもできる。そして、FIGURE コマンドでグラフの表示が行われ、プリンタにハードコピーを取ることができる。図2にあるように、オブジェクトやグラフの設定データはディスクに保存することができる。また、データテキストファイルやリザルトファイルはMS-DOSのアスキーファ

図7  $p = \log y + 0.5$  のグラフ表現

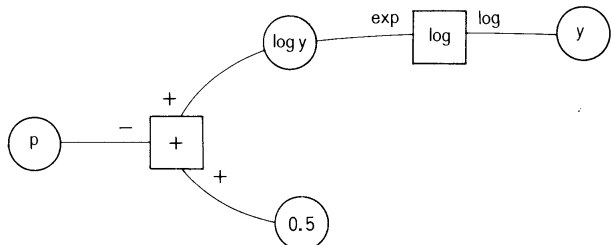
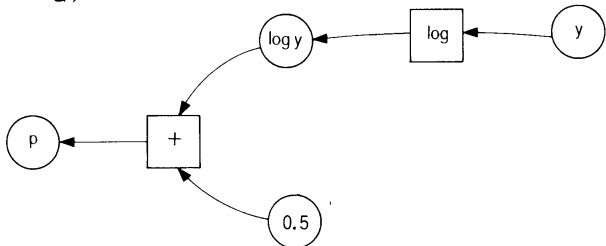


図8 辺の方向づけによる方程式の変形の表現

a)



b)

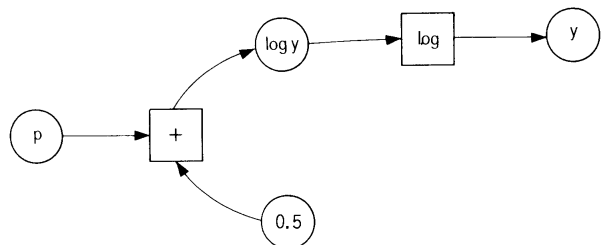


図9 (2)式の例題のためのソースリスト

```

1: /* 例題 */
2:
3:   d = 3
4:   4*y + z = c^2
5:   EXP(z) + d - y = x
6:   x' + 3*y = 0
7:   p = LOGE(y) + 0.5
8:
9:   INPUT c
10:  INTEGRAL t[0,1] STEP 0.01
11:  TREND p,x,y,z STEP 0.05

```

イルであり、他のプログラムとデータのやり取りが可能である。

EQUATRAN-Mには以上説明したコマンドのほかに、各種オプションを設定する OPTION コマンド、MS-DOS の内部コマンドを実行する SYSTEM コマンド、収束計算や積分計算のデフォルト値を表示・変更する DEFAULT コマンド、計算終了後の任意の変数値を表示させる VALUE コマンドなどがある。

## コンパイル部における内部処理

EQUATRAN-M のユーザは、その内部処理について特に意識する必要はないが、大まかな流れを理解しておくことはこのソフトウェアを使いこなすためには有益であろう。コンパイル部の処理は、(1)構文解釈とグラフの作成、(2)グラフ処理による計算順序の決定、(3)中間コード(オブジェクト)の生成、の3つのステップからなる。

### (1) 構文解釈とグラフの作成

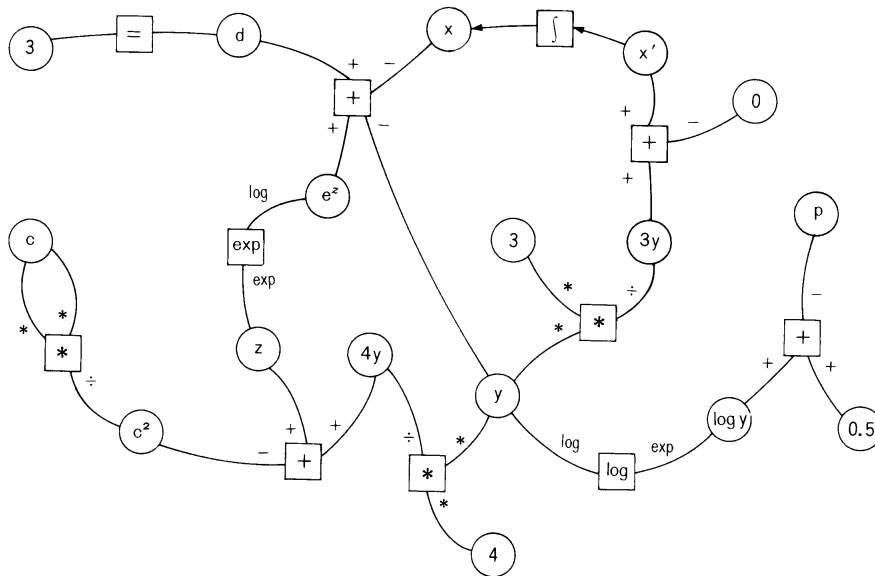
ソーステキスト中に書かれた方程式は、それぞれ演算の優先順位に基づいて構文解釈され、その結果は変数および演算を頂点とするグラフ(本項でいう「グラフ」はいわゆるグラフ理論におけるグラフであって折れ線グラフなどのグラフではない)として表現される。一例として、方程式

$$p = \log y + 0.5 \quad (1)$$

によって作られるグラフを図7に示す。このグラフでは、変数(pやyのほかに定数0.5や、中間変数のlog yも広義の変数とみなされる)の頂点を○で、演算(logのような関数も演算の一種である)の頂点を□で表し区別している。

変数の頂点と演算の頂点を結ぶ辺には、その変数と演算との関係を示すフラグ(図7の+、-、log、expなど)が付加されており、これによってたとえば「0.5とlog yとを加えるとpになる」ことが明示される。グラフ理論では、このように頂点が2つのグループに分けられ、同じグループ内では辺を共有しないグラフを2部グラフということから、

図 10 例題のための計算 2 部グラフ



このグラフを「計算 2 部グラフ」と呼ぶことにしよう。図 7 の計算 2 部グラフは方程式そのものを表現していると考えられるが、この中の辺に方向づけを行うことによって、計算の方向あるいは式の変形を表すことができる。たとえば、図 8 a は (1) 式によって  $p$  を計算することを表し、図 8 b は同じ (1) 式を変形して、 $y = \exp(p - 0.5)$  を計算することを表している。

次に、微分方程式を含む簡単な連立方程式を例にとって考えてみよう。

$$\left. \begin{aligned} d &= 3 \\ 4y + z &= c^2 \\ e^z + d - y &= x \\ \frac{dx}{dt} + 3y &= 0 \\ p &= \log y + 0.5 \end{aligned} \right\} \quad (2)$$

ここで、 $c$  は入力データとして与えるものとし、 $t$  を  $[0, 1]$  の区間で積分するものとする。この例題に対する EQUATRAN-M のソースリストを図 9 に、また、構文解釈の結果

作成される計算 2 部グラフを図 10 に示す。図 10 の中で  $\textcircled{x}$  と  $\textcircled{x'}$  との間の  $\int$  によって表されている演算は積分演算を表すもので、ソースリスト中の INTEGRAL 文によって導入される。積分演算では計算の方向に自由度がないので、辺が既に方向づけされていることに注意してほしい。

## (2) グラフ処理による計算順序の決定

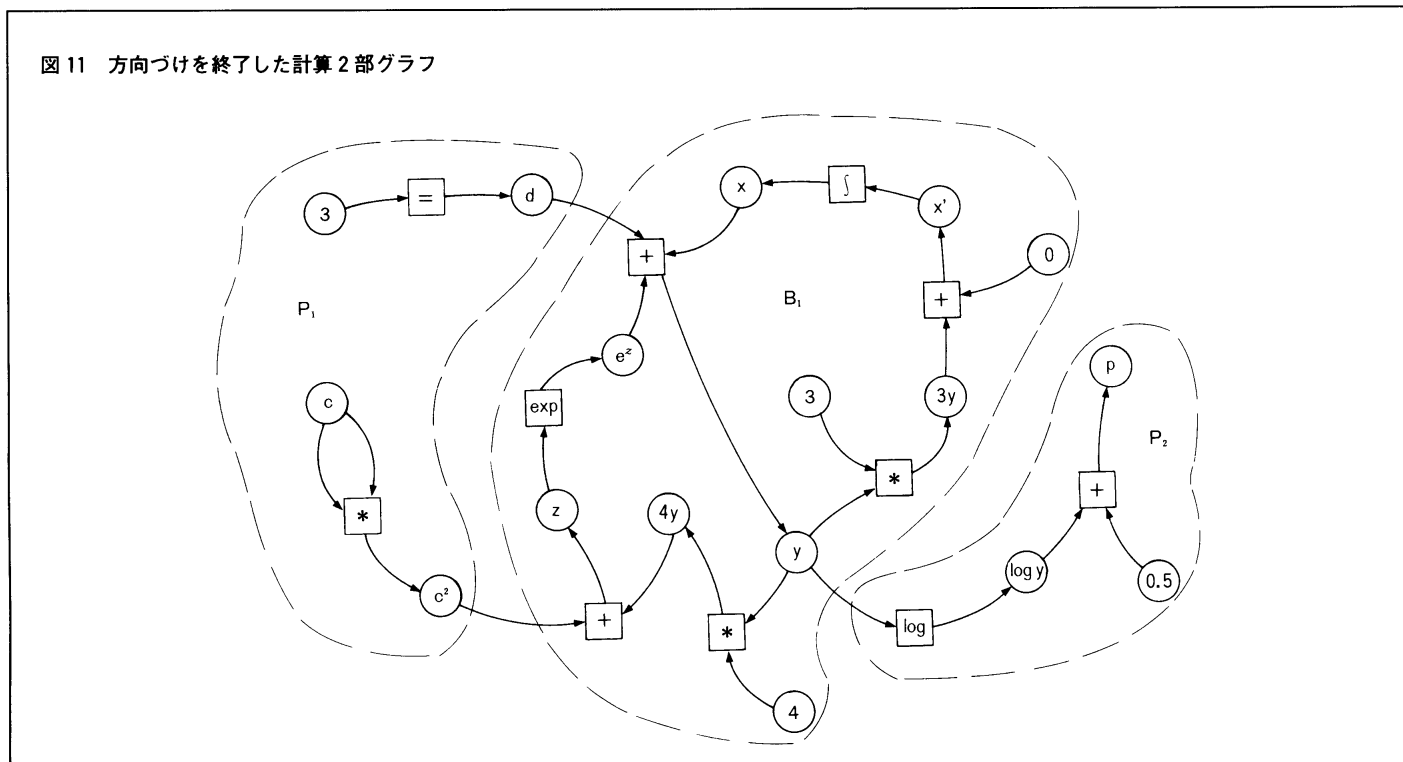
〈グラフの方向づけ〉

グラフ処理の最初のステップは、構文解釈によって作られた計算 2 部グラフの中のすべての辺に方向を与えることである。辺の方向は次の条件を満たすように決める必要がある。

- ① 定数と入力変数の頂点には外向きの辺だけが存在する。
- ② その他の変数の頂点には一つかつ一つの内向きの辺が存在する。
- ③ 演算の頂点には一つかつ一つの外向きの辺が存在する。

これらの条件を満たす方向づけを行うことは、すべての

図 11 方向づけを終了した計算 2 部グラフ



未知変数が必ず一回だけ計算されるように方程式を変形することと同等である。このような方向づけは一通りとは限らないが、方程式の構成上解ける問題であれば少なくとも一つは存在する。一方、方程式全体として、あるいは局部的に式の過不足がある問題ではこれらの条件を満たすように方向づけすることはできず、解答不能の問題であることが判別される。図 11 に (2) 式の例題について方向づけを終了した計算 2 部グラフを示す。

〈サイクリックブロックの判別〉

グラフの方向づけが無事終了すれば、この方向に従って上流の演算の頂点から順次計算して行けばよいわけであるが、図 11 の  $B_1$  の部分のように辺の方向に沿ってループが構成されている部分については、直ちに計算順序を決めることはできない。このような部分をサイクリックブロックと呼ぶが、サイクリックブロックは、その中の任意の 2 つの頂点について、辺の方向に沿って互いに往復可能なパスが存在することから簡単に判別できる。

サイクリックブロック  $B_1$  の判別を行った後、この内部の計算順序の決定をとりあえず保留しておけば、全体の計算の順序は、

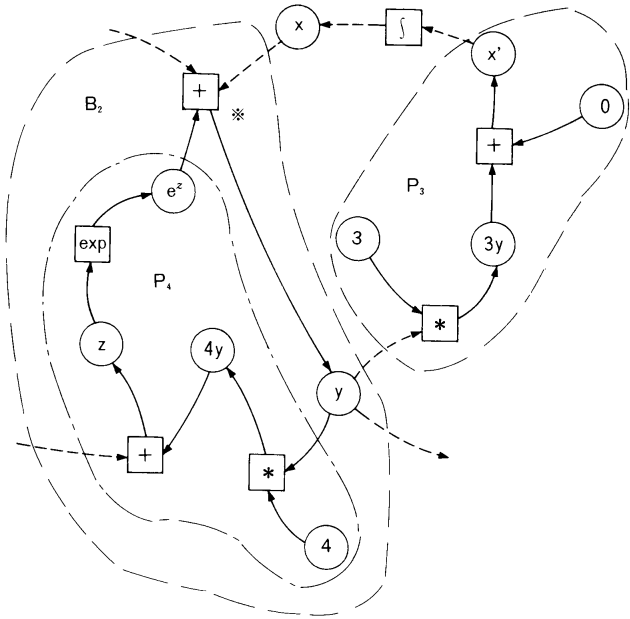
$$\textcircled{P_1} \longrightarrow \textcircled{B_1} \longrightarrow \textcircled{P_2}$$

のように決めることができる。P は非サイクリックブロック (ループを含まない部分) であり、これらの中の計算順序は上流から順次決めることができる。

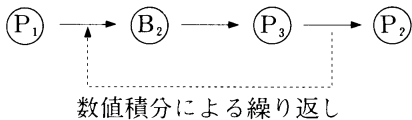
〈サイクリックブロックの分解——その 1〉

図の例題のように、積分演算の頂点が含まれているサイクリックブロックは、この頂点を取り除くことによって分解することができる。数値積分における一つのきざみ点での計算を考えた場合、被積分変数 (例では  $x$ ) の値は前のきざみ点での計算から既知であるとしてよいからである。積分演算の頂点を除いた残りの  $B_1$  の部分について前項のサイクリックブロックの判別の手順を適用すれば、 $B_1$  は図 12 のように  $B_2$  と  $P_3$  とに分解されて全体の計算順序は次のよう

図 12 サイクリックブロック  $B_1$  の分解



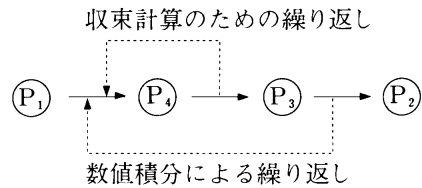
になる。



〈サイクリックブロックの分解——その 2〉

残ったサイクリックブロックは連立して解くべき方程式である。この連立方程式が線形の場合には直接これを解くことができるので、計算順序の決定は完了する。しかし、この例の  $B_2$  のように非線形であれば、この部分に収束計算を組み入れなければならない。収束計算は、連立方程式中のいくつかの変数の値を仮に既知であると考え、このことによって余剰となる式の両辺が等しくなるように変数の仮定値を修正する操作である。グラフ上では、値を仮定する変数と余剰となる式(演算)の頂点を取り除くことによってループを解消する操作に相当している。この例では、変数  $y$  と式  $e^z + d - y = x$  (図 12 の \* 印) の組み合わせがコン

パイラによって自動的に選ばれるが、ユーザの与える RESET 文はこの変数と式の組み合わせを直接指定するものには他ならない。収束計算のための変数と式(演算)の頂点を除いた残りのグラフについて辺の方向づけ以降の操作をやり直せば、この例題のための計算順序は結局次のように決定される。



(3) 中間コードの生成

サイクリックブロックの分解が終了して、計算 2 部グラフ全体が非サイクリックブロックと線形連立方程式のサイクリックブロックとの順序列に分解されてしまえば、これから計算の手順を一つ一つの演算の列である中間コードに生成する作業は容易である。このとき、収束計算や数値積分のためのアルゴリズムを呼び出すコードも同時に組み込まれる。

以上の手続きによって生成された計算手順では、収束計算や積分計算のためのループの中から繰り返し評価する必要のない部分が外へ出され、合理的なコードが生成されることに注目してほしい。たとえば、図 11 の  $P_1$  の部分は計算開始の時点でのみ計算され、 $P_2$  の部分は出力が要求される時点 (TREND 文による出力の周期) でのみ計算される。この意味で EQUATRAN-M による計算の手順は最適化されているということが出来る。

おわりに

EQUATRAN-M を中心に 6 回にわたって方程式解法ソフトとその使用法を紹介してきた。なるべく多彩な使用例をと心掛けたつもりではあるが、なにぶん応用範囲の広いソフトであり、十分その目的を達することはできなかったように思う。表 2 に EQUATRAN-M に関する文献を掲げておいたので、これらも参照してほしい。



方程式解法ソフトの魅力は、プログラミングという煩わしい、そして時に危険な作業を省略して、直接問題に向いあえることであるといえよう。数式モデルさえ作成すれば、モデルの部分的な変更はもちろんのこと、計算条件の与え方を変更した場合の検討も即座に行うことができる。現在の方程式解法ソフトは EQUATRAN-M も含めてもちろん万能とはいえないが、多くの問題をプログラミング(解法の検討からデバッグまで)なしに解くことができる。言い替えば、この場合のプログラミングの作業は、今回の EQUATRAN-M の内部処理でも示したように、コンピュータでもできる機械的な作業である。この意味で方程式解法ソフトを使うことは、コンピュータの代わりに考えるのをやめて、もっと創造的な仕事に時間を使うことだといえるのではなかろうか。

本連載では応用を化学工業の分野に限って取り上げたが、他の産業分野においてもあるいは教育用としても大いに利用されてよいソフトウェアであろう。 [完]

(おぐち ごろう・よこやま かつみ 三井東圧化学株)

表 2 EQUATRAN-M 文献一覧

1. 小口, 横山, 佐藤: プロセス物質収支計算への方程式解法言語の応用  
プロセスシステム工学総合シンポジウム 1985.3 於芦屋
2. 小口, 横山, 佐藤: パソコン用方程式解法言語の応用  
化学工学協会50年会 E 302 1985.3 於横浜
3. 宮原, 林田, 須藤: 分離における数値計算  
分離技術 No. 5 P. 281 (1985)
4. 宮原, 林田, 須藤: 分離技術懇話会夏季研究討論会資料  
1985.8 於浜松
5. 小口: 創造的技術者の生産性を向上させる方程式解法ソフト  
— EQUATRAN-M 日経コンピュータ 9月30日号 P. 207 (1985)
6. 宮原, 小口: 方程式解法ソフト EQUATRAN-M  
OR学会秋期研究発表会 アブストラクト集 (SW-9) P. 228 (1985)
7. 宮原他: 連載「EQUATRAN-M」技術計算用連立方程式解法言語  
(1)~(9) ケミカル・エンジニアリング 8月号(1985)~4月号(1986)
8. 小口: プログラミング不要の技術計算用ソフト (EQUATRAN-M)  
の設計思想と活用事例  
情報処理学会・情報システム研究会報告 No. 10-2
9. 小口他: 連載 化学工業における数値計算 (1)~(6)  
information 8月号(1986)~1月号(1987)
10. 小口, 宮原: EQUATRAN-M (イコートラン・エム)  
— 知的生産性を高める方程式解法ソフト —  
オペレーションズ・リサーチ 11月号 P. 705 (1986)