

# 分離における数値計算

## EQUATRAN-M を中心にして

宮原 昶 中\*  
林 田 豊\*  
須藤 精 一\*

### 1. はじめに

近年 CADD (Computer-Aided Drug Design) のように化学の分野にもデザインの手法が採用されるようになった。一方、分離技術など工学の分野では古くからデザインの概念を駆使して化学工業に寄与してきており、そこでは種々の方式が既に確立しているといえる。しかし、

- (1) 連立非線形方程式を解く。
- (2) 任意の組み合わせの変数に値を入力し、残る変数の値を算出する。

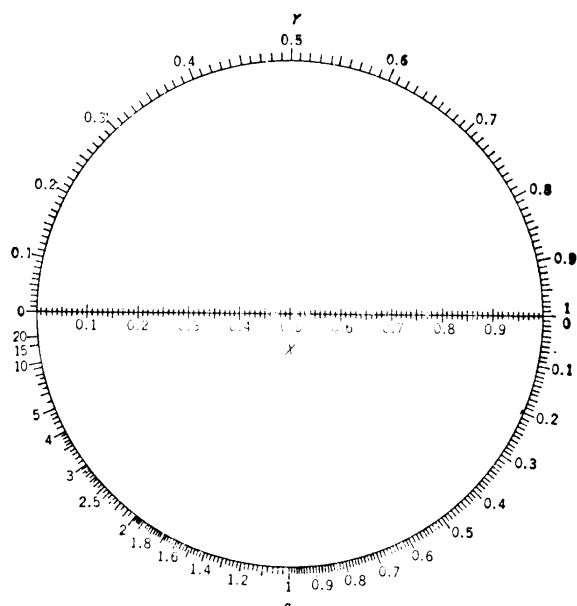


図1  $\frac{Y}{1-Y} = a \frac{X}{1-X}$  形計算図表

など煩雑な式の変形、計算をしばしば余儀なくされる。これを避ける狙いで(特定目的とはいえ)ノモグラフが有効に利用されていた。例えば mol 分率-wt 分率変換では図1に示すノモグラフが利用でき ( $\alpha = M_1/M_2$ , Mは分子量)<sup>1)</sup>、どの変数(X, Y,  $\alpha$ )からも残る変数の値が計算できることが大きな特徴である。とはいえ、

- (1) 適用が特定の目的に限られる。
- (2) 数値の範囲により使い分けが必要。

という制約を持ち合わせている。3変数のノモグラフ、4変数のノモグラフ(どこかの2変数が網状に表示されている)では任意の2, 3変数を指定できるが、5以上の変数の関与するノモグラフでは基本的には3(4)変数ノモグラフの組合せにすぎず、自由度上指定しなければならない変数のうち、論理的に指定することが許される組合せは限られている。

一方、電卓では計算手順さえ考えればどんな対象にも適用できるため、ノモグラフを駆逐してしまったように思われる。しかし、複雑な繰り返し計算を避けて通ることができないことはよく経験するところである。

上述のノモグラフの制約を除き、電卓から煩雑な手順を考えることを省略できれば極めて便利な訳であるが、これを実現したのが EQUATRAN-M であるといえる。

EQUATRAN-Mの発想の原点はもともとフローシートシミュレータを

- (1) もっとフレキシブルに使えるように、プロセスの流れの順序と計算の順序は異なっ

\*三井東圧化学(株)システム部

ていてもよい。

単一流れでも成分毎に計算方向（方法）が異なってもよい。

(2)プログラミングの素養のない人でも使えるように、

ということにあるが、これを更に拡張して、

(3)一般の方程式群も解ける。

ようにとの意図で、ホストコンピュータIBM370用に1975年に開発したものがEQUATRANである。爾来10年間、社内では好んで（プログラミングに疎い）研究者が使用しており、その経験をふまえて使い易さを求め続け、機能拡張につとめて今日に至った。これに会話型（interactive）と即時性を付加、活かすべくパソコンに移植してEQUATRAN-Mが誕生した。EQUATRAN-Mはノモグラフ、電卓、パソコンの長所を兼ね備えたエンジニア、研究者のための強力な武器である（表1）。

端的に言えばEQUATRAN-Mとは方程式さえ書けば（煩雑な）手順を示さなくても式を解読して手順を生成してくれるソフトウェアであり、その特徴は次のように要約できる。

- (1)式の順序は任意でよい（手順型でない！）
- (2)数式の変形をしなくてよい（変形に伴う誤りを犯さなくてすむ）
- (3)仮定したい変数を指定すればその手順を生成してくれる
- (4)入力・出力の関係を逆転させることが容易

これは表2のようにまとめることができる。

EQUATRAN-Mのもう一つの特徴を示すためにプログラミング、パッケージソフトとの比較を

表1 諸手法の比較(1)

	長 所	短 所
ノモグラフ	<ul style="list-style-type: none"> <li>・手順を考えなくてもよい。</li> <li>・指定する変数と計算する変数とを逆転させることが簡単にできる。</li> <li>・即時性がある。</li> </ul>	<ul style="list-style-type: none"> <li>・特定目的で汎用でない。</li> <li>・（目盛の範囲あり）</li> <li>・多変数に弱い。</li> </ul>
電卓	<ul style="list-style-type: none"> <li>・手軽に使える。</li> <li>・即時性がある。</li> </ul>	<ul style="list-style-type: none"> <li>・手順を考えないといけない。</li> <li>・入出力逆転で手順の考え直しが必要。</li> <li>・大きな問題は扱いにくい。</li> </ul>
パソコン	<ul style="list-style-type: none"> <li>・汎用のプログラムが書ける。</li> <li>・大きな問題も扱える。</li> </ul>	<ul style="list-style-type: none"> <li>・プログラム作成時に時間がかかり、今すぐという目的には使えない。</li> <li>・入出力を逆転するとプログラムを書き直さなければならない。</li> </ul>

表2 数値計算ツールの分類

	アナログ型	デジタル型
手順型	計算尺	電卓 FORTRAN } などの高級言語 BASIC }
非手順型	ノモグラフ	EQUATRAN-M

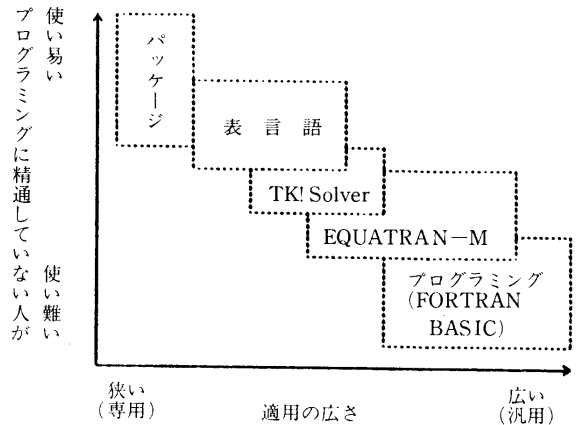


図2 諸手法の比較(2)

表3に示す。EQUATRAN-Mの便利さがどこにあるかが一目瞭然であろう。

さらに最近TK! Solver<sup>2)</sup>というEQUATRAN-Mに機能の酷似した（しかし狭い）ソフトが日本でも発表されたが、この機能の位置づけを図2に示す<sup>3,4)</sup>。

以下簡単な例から順に分離に関するEQUATRAN-Mの適用例を紹介する。実際の業務においては以下に示す例より一般に複雑なものであるが、ここではEQUATRAN-Mがどのように使えるかを示すため簡単な例にとどめてあることを付記しておく。

まず理想溶液の気液平衡関係を取りあげる。液相のモル分率（x）、気相のモル分率（y）と比揮発度（α）の関係は、

$$\frac{y}{1-y} = \alpha \cdot \frac{x}{1-x} \quad (1-1)$$

と表わされるが、これのα、xを与えてyを求めるケースをリスト1(a)に示す。リスト1(b)のようにしても同じ結果を得る。x、yを与えてαを算出するにはリスト1(c)とすればよい。すなわち式本体（…で囲ったもの）はそのまま下線部分のみ変更すればよい。 / \* …… \* / は注釈行、INPUTはこれ

表3 EQUATRAN-Mとプログラミング言語およびパッケージプログラムとの比較

	EQUATRAN-M	プログラミング言語	BASIC FORTRAN	パッケージプログラム
I 適用範囲	方程式（と数表など）で表わせる問題全般 ・多変数の最適化計算も可能。 ・微分方程式、積分方程式はそのままでは扱えない（注1）。	原則的にあらゆる問題に適用可能 ・ただし、プログラムを自分で作ることを覚悟しなければならない。		ピッタリのパッケージが市販されていればよいが、なければお手上げ。
II 連応性・能率	必要な時にすぐ答えが得られる。	答えを得るまでに相当の時間と手間がかかる。		パッケージが手元に準備されていればすぐ答えが得られる。
III 経済性	1本購入すれば広範囲の問題に対応できて経済的。 プログラム開発の人件費が大巾に削減できる。	プログラムの開発と保守に膨大な人件費がかかる。		それぞれの目的に合ったパッケージを購入する必要があり不経済。
IV 答えを得るまでの手続き	① 準備	EQUATRAN-Mの文法をマスター。 ・原則として式をそのまま書けばよいので簡単。 ・一度マスターすれば他の問題の場合もそのまま応用できる。	BASICまたはFORTRANの文法をマスター。 ・十分経験を積まないと正しいプログラムは書けない。 ・一度マスターすれば他の問題の場合もそのまま応用できる。	パッケージプログラムを購入！ 使用方法をマスター。 ・パッケージが変われば使用方法も変わってしまう。
	② 用いる式を用意	文献、単行本、実例集などを参考に自分で用意する。実例パッケージを提供予定。	文献、単行本、実例集などを参考に自分で用意する。	不要。ただし、パッケージで使用している式が望み通りのものかチェックする必要がある。
	③ 解法のアルゴリズムを考える	不要。	式の変形や計算手順はもちろん、収束計算のアルゴリズムなども用意する必要がある。	不要。
	④ プログラム・データの作成	式をそのまま書くだけでよい。 収束計算や最適化の指定が簡単にできる。	文法に従って計算式、入出力などを全てプログラムに組み立てる必要がある。 収束手法や最適化手法のアルゴリズムを組み込むためには専門的な知識も必要。	マニュアルに従ってデータを作成するだけでよい。
	⑤ デバッグ	式全体の書き間違いや、式の過不足を修正する。 ・書式の誤りのほか、式の過不足も自動的にチェックされメッセージされるので容易。	式の変形や計算順序のほかロジックの誤りなども細心の注意をもってチェックする必要がある。 ・書式の誤りしか自動的にチェックされない。	データの誤りを修正するだけでよい。
V 問題を少し変形したい時	入力変数と出力変数とを一部入れ換える	入力、出力の指定を変えるだけでよい。		パッケージプログラムの機能として準備されていない変更の場合はケーススタディで試行錯誤的に解を探さなければならない。
	式を一部変更したりあるいは追加したりする	変更した式だけを書き直し、あるいは追加すればよい。	変更は可能だがアルゴリズムから考え直さなければならないケースが多い。	パッケージプログラムの機能として準備されていない変更は不可能。
	収束計算や最適化の変数を変更する	指定文を変更するだけでよい。		
VI 報告書に用いる時	計算方法のドキュメント	ソーステキスト（式を記述したEQUATRAN-Mへの入力）がそのままドキュメントとなる。 ・ドキュメントと計算内容が完全に一致している。	プログラムのドキュメントを別に作成しなければならない。 ・ドキュメントの作成および保守が大変	パッケージプログラムのドキュメントが用意されている。
	出力をそのまま報告書として用いたい場合	所定の形式でのみ出力される。ファイルへ出力してBASIC等で処理することが可能（注2）。	プログラムを作成すれば自由な形式で出力することができる。	通常報告書として使用可能な出力が用意されている。
	出力をグラフ化したい場合	ファイル出力をBASIC等で処理すれば可能（注3）。	プログラムを作成すれば可能。	パッケージによってはグラフ化の機能を準備している。
VII 総合評価	・フレキシビリティが高く、連応性がよいので非定形な業務（研究、開発など）に最適。	・パッケージもなくEQUATRAN-Mでも取り扱えない時、止むを得ずプログラムを作成する。		・問題に適合したパッケージプログラムがある場合は最も確実に推奨できる。 ・定期的に繰り返し使用しないと不経済。 ・フレキシビリティがないので非定形な業務（研究、開発など）には向かない。
備考	(注1) 離散化した式を作成すれば可能。 常微分方程式に対しては拡張版を提供予定。 (注2, 3) 出力の編集、グラフ化用ソフトを別途提供予定。			

らの値をキーボードから入力することを示している。

る計算例を紹介する。

## 2. Antoine 定数の近似

蒸気圧の近似式には種々あるが、Antoine式によ

リスト2(a)は最小二乗法による Antoine 定数の近似である。簡単にリストの説明をしておこう。

／\*……\*/は注釈行で式の横に書くこともでき

```

1:  /* 理想溶液の気液平衡関係 */
2:
3:  y/(1-y) = A * x/(1-x)
4:
5:  /* x 液相のモル分率
6:     y 気相のモル分率
7:     A 比揮発度 */
8:  input x,A

```

(a)

```

1:  /* 理想溶液の気液平衡関係 */
2:  y1/y2 = A * x1/x2
3:  x1 + x2 = 1
4:  y1 + y2 = 1
5:  input x1,A

```

(b)

```

1:  /* 理想溶液の気液平衡関係 */
2:
3:  y/(1-y) = A * x/(1-x)
4:
5:  input x,y

```

(c)

#### リスト1 理想溶液の気液平衡関係

```

1:  /* Antoine 定数の近似 */
2:  var t(8),pd(8),p(8)
3:  /* t - pd : エタノール蒸気圧データ
4:     出典   : 化学便覧 p.388 (1952) */
5:  t = (-31.3, -12.0, -2.3, 8.0, 26.0, 48.4, 63.5, 78.4) /* 'C */
6:  pd = ( 1, 5, 10, 20, 60, 200, 400, 760) /* mmHg */
7:  /* Antoine 式 */
8:  log10(p) = A - B / (t + C)
9:  C = 231.48
10: /* 最適化 A,B : 独立変数
11:          ss : 目的関数 (誤差の二乗和) */
12: find ( A#10 [0,20], B#2000 [0,5000] ) ..
13: minimize ss until 0.1%
14: ss = sum((log10(p/pd))^2)

```

#### リスト2(a) Antoine 定数の近似

```

1:  /* Antoine 定数の近似 */
2:  /* t - pd : エタノール蒸気圧データ
3:     出典   : 化学便覧 p.388 (1952) */
4:  var t(8),pd(8),p(8)
5:  t = (-31.3, -12.0, -2.3, 8.0, 26.0, 48.4, 63.5, 78.4) /* 'C */
6:  pd = ( 1, 5, 10, 20, 60, 200, 400, 760) /* mmHg */
7:  /* Antoine 式 */
8:  log10(p) = A - B / (t + C)
9:  /* A,B に関する正規方程式 */
10: sum(log10(p/pd)) = 0
11: sum(log10(p/pd)/(t+C)) = 0
12: /* 最適化 C : 独立変数, ss : 目的関数 */
13: find ( C#200 [0,500] ) minimize ss
14: ss = sum((log10(p/pd))^2)

```

[ 計算結果 ]

```

t =
  1) -31.3          2) -12          3) -2.3          4) 8
  5) 26            6) 48.4         7) 63.5         8) 78.4
pd =
  1) 1             2) 5             3) 10            4) 20
  5) 60            6) 200           7) 400           8) 760
p =
  1) 1.004142      2) 4.930812     3) 10.026939    4) 20.139591
  5) 60.390372    6) 198.013604   7) 401.827311   8) 759.311788
C = 259.912104
B = 2029.528879
A = 8.879404
ss = 8.113042E-005

```

リスト2(b) Antoine 定数の近似

る(5, 6行目). 2行目VARはVARIABLE文で, プログラム中で使用する配列変数(1次元または2次元)の宣言を行う. 5, 6行目は配列変数への値の代入であるが, 1次元の場合にはこのようにかっこでくくって書けばよい. また8, 9行目の式ではスカラーと配列(ベクトル)が混在しているが, この場合にも規則に従って適当に拡張解釈してくれる. 8行目はAntoine式であり, tの8つのデータに対する8本の式を表わしている. EQUATRAN—Mでは式の書き方を始め種々の著しい特徴を持っているが, このように表現が非常に簡明でわかりやすい.

さて, パラメータを決定するのに重回帰により正規方程式を作成し, これを解く方法もあるが, ここでは直接目的関数(誤差の二乗和)を最小化する. それにはFIND文(12行目)による最適化機能を使用する(本例ではCは既知としA, Bのみ最適化

する)。独立変数については初期値および探索範囲を指定する。UNTILは収束精度の指定であり、解の精度をあげるにはこの指定を厳しくするか、探索範囲を狭めればよい。本例では使用していないが、さらに独立変数に制約条件を付加することも可能である。

しかしながら、多変数の最適化にはかなり時間がかかる。そこで、一部の変数について正規方程式を作り、残りの変数について最適化するという方法が考えられる。リスト2(b)ではA, Bについて正規方程式を作成し、Cに関して最適化を行っている。これにより計算時間が非常に短縮される。

リスト2(b)と共にその計算結果を示した。出力はOUTPUT文による指定がない場合すべての変数について行われる。OUTPUT文により指定されている場合には、その順番に従って指定された変数のみ出力される。

リスト中のSUM, LOG 10はEQUATRAN-Mの組み込み関数で、ベクトルまたはマトリックスの総和(マトリックスでは結果はベクトル)、常用対数である。組み込み関数はこの他にも多数用意されており簡単に利用できる。

### 3. Wilson パラメータの算出

収束計算を行う場合、独立変数、収束判定のための式の指定および独立変数の初期値、変域の指定が必要であるが、EQUATRAN-Mでは式を書くだけで自動的に収束計算を行ってくれる。しかし、それらの組み合わせが必ずしも最良ではない。そのようなケースにはRESET文により指定を行う(リスト3)。実際には計算の効率上、問題の物理的内容を考慮して積極的に指定してやるのが望ましい。また、本例では場合により解が3組存在する<sup>5)</sup>が、この場合にも変数の初期値、変域を適当に与えることにより3組とも求めることができる。

リスト3は無限希釈の活量係数からWilsonパラメータを求める例であるが一般的な式(2成分Wilson式)を使用して計算している。このように一般的な式を書いておき、ほんの少しの修正でいろいろ違った計算に使えるところがEQUATRAN-Mの便利なところである。なお、9, 10行目のように式をセミコロン(;)で区切れば1行に複数の式を書いてもよい。

```

1: /* Wilson 式のパラメータの算出 */
2: /* Journal of Chemical Engineering of Japan Vol.3 No.2 p.157 */
3:
4: eq1: loge(g1) = -loge(x1+L12*x2)+x2*(L12/(x1+L12*x2)-L21/(L21*x1+x2))
5:      x1=1-x2
6:      loge(g2) = -loge(x2p+L21*x1p)+
7:                x1p*(L21/(x2p+L21*x1p)-L12/(L12*x2p+x1p))
8:      x2p=1-x1p
9:      g1 = 0.4 ; g2 = 0.56
10:     x1 = 0 ; x2p = 0
11:     reset L12 # 2 [ 0.1,10] by eq1

```

リスト3 Wilson パラメータの算出

```

1: /* NRTL 式による 2 成分液液平衡の計算 */
2: Global t12=t12,t21=t21,a12=a12,a21=a21
3: /* NRTL 式 */
4: MACRO NRTL
5:   loge(g1) = X2^2*(t21*exp(-2*t21)/(X1+X2*exp(-t21))^2 ..
6:             + t12*exp(-t12)/(X2+X1*exp(-t12))^2) / a12
7:   loge(g2) = X1^2*(t12*exp(-2*t12)/(X2+X1*exp(-t12))^2 ..
8:             + t21*exp(-t21)/(X1+X2*exp(-t21))^2) / a21
9: END NRTL
10: /* 2 成分液液平衡関係 */
11: x1*g1 = xp1*gp1
12: x2*g2 = xp2*gp2
13: CALL NRTL(g1=g1,g2=g2,X1=x1,X2=x2)
14: CALL NRTL(g1=gp1,g2=gp2,X1=xp1,X2=xp2)
15: x1 + x2 = 1
16: xp1 + xp2 = 1
17: a12 = 0.2 ; t12 = 0.204
18: a21 = 0.2 ; t21 = 0.38
19: output x1,xp1,x2,xp2,g1,gp1,g2,gp2,t12,t21,a12,a21

```

リスト4 2成分液々平衡計算

## 4. 液液平衡

### 4.1 NRTL式による2成分液液平衡

リスト4はNRTL式による2成分液液平衡計算(NRTLパラメータを与えて各相組成, 活量係数を求める)の例である。同様の式が何度も出てくる場合には, この例のようにマクロ(4~9行目)を使用するとより簡単になる。マクロはプログラム中に同じ形の式(群)が現われる場合, それをMACRO文とEND文で囲んで定義しておき, それをCALLして使用するもので, FORTRANにおけるSUBROUTINEのようなものである。ただし, その定義はCALL文よりも前で行う。なおマクロ内の変数はすべてCALL文ごとに異なる変数として取扱われメイン中の変数と同名でも別のものである。これを同一のものとして使用したいときにはGLOBAL文文(2行目)を用いる。GLOBAL文はメイン, マクロを問わずプログラム中の特定のシンボルをその文字列で置き換えるものであるが, リスト4のように指定した場合, 左辺のシンボルは

すべて右辺のもので置き換えられ, メイン, マクロで共通となる。なお, メインのみ, マクロ内のみで同様のことを行うには次の例(リスト5)のようにLOCAL文を使用すればよい。

### 4.2 3成分系溶解度曲線の近似

一般に溶解度曲線は一つの曲線では近似しにくくプレートポイント等で二つに分けて近似する(これの極端なケースがspline近似である)。この場合プレートポイントでの連続性, 滑らかであること(両曲線の微係数が等しい)を制約条件として用いる。リスト5はアセトニトリル-ベンゼン-ヘプタン系の例で, ベンゼン(組成y)をヘプタン(組成x)の3次関数として近似している。なお, 本例では溶解度曲線両端, すなわち2成分系溶解度をも固定し制約条件として用いた。方法はラグランジュの未定乗数法によったが, その目的関数は次式である。

$$Q = \sum_{i=1}^n (y_{ci} - y_i)^2 - \lambda_1 (y_1 - y_{c1}) - \lambda_2 (y_n - y_{cn}) - \mu (y_{n1(a)} - y_{n1(b)}) - \nu (y_{n1(a)'} - y_{n1(b)'}) \quad (4-1)$$

ここで,  $y_i, y_{ci}$  は  $i$  番目のデータ及び計算値で

```
1: /* 3成分系溶解度曲線の近似 */
2: /* n: データ数, n1: プレートポイント */
3: LOCAL n=25, n1=13
4: /* Acetonitrile - Benzene - Heptane 系の溶解度曲線
5: x - y: Heptane - Benzene 液液平衡データ (モル分率 at 25 'C) */
6: VAR x(n), y(n), yc(n)
7: /* Heptane 相, Acetonitrile 相をそれぞれ 3 次関数で近似する */
8:
9: /* ラグランジュの未定乗数法による係数の決定 */
10: L1, L2, Mu, Nu: ラグランジュ乗数
11:
12: /* Acetonitrile 相 */
13: yc(1:n1) = a1 + a2*x(1:n1) + a3*x(1:n1)^2 + a4*x(1:n1)^3
14: /* Heptane 相 */
15: yc(n1:n) = b1 + b2*x(n1:n) + b3*x(n1:n)^2 + b4*x(n1:n)^3
16: /* プレートポイントでの微係数 */
17: a2+2*a3*x(n1)+3*a4*x(n1)^2 = b2+2*b3*x(n1)+3*b4*x(n1)^2
18: /* 2成分系相互溶解度 */
19: yc(1,n) = y(1,n)
20: /* ラグランジュの未定乗数法により導かれる式 */
21: SUM(yc(1:n1)-y(1:n1)) + L1 - Mu = 0
22: SUM((yc(1:n1)-y(1:n1))*x(1:n1)) + L1*x(1) - Mu*x(n1) - Nu = 0
23: SUM((yc(1:n1)-y(1:n1))*x(1:n1)^2) + L1*x(1)^2 - Mu*x(n1)^2 - 2*Nu*x(n1) = 0
24: SUM((yc(1:n1)-y(1:n1))*x(1:n1)^3) + L1*x(1)^3 - Mu*x(n1)^3 - 3*Nu*x(n1)^2 = 0
25: SUM(yc(n1:n)-y(n1:n)) + L2 + Mu = 0
26: SUM((yc(n1:n)-y(n1:n))*x(n1:n)) + L2*x(n) + Mu*x(n1) + Nu = 0
27: SUM((yc(n1:n)-y(n1:n))*x(n1:n)^2) + L2*x(n)^2 + Mu*x(n1)^2 + 2*Nu*x(n1) = 0
28: SUM((yc(n1:n)-y(n1:n))*x(n1:n)^3) + L2*x(n)^3 + Mu*x(n1)^3 + 3*Nu*x(n1)^2 = 0
29: /* データ x, y
30: 出典: A. Sathy, H. T. Cullinan, AIChE J., Vol. 2, No. 3, p. 571 (1975) */
31: x = (0.0381, 0.0442, 0.0514, 0.0600, 0.0608, 0.0723, 0.0724, 0.0849, 0.0984, ..
32: 0.1080, 0.1238, 0.1855, 0.281, 0.4065, 0.5234, 0.5619, 0.6060, 0.6485, ..
33: 0.6897, 0.7128, 0.7669, 0.8010, 0.8607, 0.8896, 0.9371)
34: y = (0.0, 0.0210, 0.0359, 0.0591, 0.0790, 0.1012, 0.1126, 0.1337, 0.1554, ..
35: 0.1729, 0.1906, 0.2346, 0.26, 0.2695, 0.2467, 0.2319, 0.2170, 0.1941, ..
36: 0.1705, 0.1560, 0.1273, 0.0989, 0.0631, 0.0384, 0.0)
37: OUTPUT a1, a2, a3, a4, b1, b2, b3, b4, x, y, yc
```

リスト5 溶解度曲線の近似

$$y_{ci} = a_1 + a_2 x_i + a_3 x_i^2 + a_4 x_i^3$$

(アセトニトリル相) (4-2)

または,

$$y_{ci} = b_1 + b_2 x_i + b_3 x_i^2 + b_4 x_i^3$$

(ヘプタン相) (4-3)

である。添字 n1 はプレートポイント, (a), (b) はそれぞれ (4-2), (4-3) 式からの計算値であることを示す。また, ' は微係数, n はデータ数である。λ<sub>1</sub>, λ<sub>2</sub>, μ, ν はラグランジュ乗数で (4-1) 式第 2, 3 項は 2 成分系溶解度, 4, 5 項はプレートポイントでの制約条件に対応するものである。この目的関数を係数 a<sub>1</sub> ~ a<sub>4</sub>, b<sub>1</sub> ~ b<sub>4</sub> で微分しゼロとおいた式 (リスト 21~28 行目) を 4 つの制約条件と共に解けばよい。

なお, リスト中 (13, 15 行) 1 : n1, n1 : n とあるのは, それぞれ 1 から n1, n1 から n 番目の要素に対する複数の式をまとめて表現したものである。また例えば 21 行目の第 1 項は  $\sum_{i=1}^{n1} (y_{ci} - y_i)$  を, 22 行目の第 1 項は  $\sum_{i=1}^{n1} (y_{ci} - y_i) \cdot x_i$  を表わしている。19 行目の 1 . n は 1 と n 番目の要素に対する 2 本の式を表わす。このように要素番号が連続していない場合でもピリオドで区切って書けばよい (3 つ以上でもよい)。

さて, 本例ではラグランジュ乗数を導入し, 最終的に 12 本 (もともとの独立変数は 8 - 4 = 4 つ) の連立方程式を解くこととしたが, 前出の FIND 文による最適化を使用することも可能である。その場

合の目的関数は (4-1) 式右辺第 1 項のみで, これを 4 つの制約条件のもとに最適化すればよい。4 変数の最適化となるので多少計算時間は犠牲になると考えられるが, 表現は非常に簡単になる。

## 5. 熱交換器

簡単な例として, 向流型熱交換器をとりあげリスト 6 に示す。熱交計算では与えるデータにより種々のケースが生じるが, 熱収支式

$$Q = UA\Delta t = GC(T_i - T_o) = gc(t_o - t_i) \quad (5-1)$$

と平均温度差の式を記述し, 設計問題ならば T<sub>i</sub>, T<sub>o</sub>, t<sub>i</sub>, t<sub>o</sub>, G, g, C, c の中の 7 つと U を, 解析問題ならば Q, U, A, T<sub>i</sub>, T<sub>o</sub>, t<sub>i</sub>, t<sub>o</sub> の中の 6 つと C, c (または G, g) を INPUT 文で与えて残りの変数値を求める。このように EQUATRA N-M では入出力の関係を簡単に逆転して解くことが可能である。

なお, 平均温度差として通常は対数平均が用いられるが, GC = gc (すなわち, T<sub>i</sub> - t<sub>o</sub> = T<sub>o</sub> - t<sub>i</sub>) の場合に計算できないことおよびこれに近い条件の場合に誤差が入り易いことから, リスト中に示す式を用いると便利である。

## 6. McCabe-Thiele 法による蒸留計算

McCabe-Thiele 法で計算するときの一つの困難は, 前もって式 (理論段数) の数が判っていないことである。従って, フィード段の位置, 全段数を

```

1:  /* 熱交換器の計算 */
2:
3:  /*  TI, TO  : 高温流体入口, 出口温度
4:     ti, to  : 低温流体入口, 出口温度
5:     G, g    : 高温, 低温流体流量
6:     C, c    : 高温, 低温流体比熱
7:     DT     : 平均温度差
8:     A      : 伝熱面積
9:     U      : 総括伝熱係数
10:    Q      : 伝熱量
11:
12:    出典 (EQ1 式): Chem., Eng., Sci., Vol.39, No.11,
13:                  p.1635-1636, (1984)      */
14:
15:
16:    Q = A*U*DT
17:    Q = G*C*(TI-TO)
18:    Q = g*c*(to-ti)
19:
20:  EQ1:  DT = ((TI-to)+(TO-ti))/6 + 2/3*SQRT((TI-to)*(TO-ti))
21:
22:  RESET TO  BY EQ1
23:
24:  INPUT  TI,  ti,to,  G,C,g,c,  A
25:  OUTPUT TI,TO,ti,to,DT,G,C,g,c,Q,A,U
26:

```

リスト 6 熱交換器

トライアルで求めることになる。この点ではEQU ATRAN-Mは若干不便な面はあるが、整数段になるように還流比を変えるなど簡単に変更できる点は便利である。リスト7に理想溶液の例を示す。フィード・留出組成から最小還流比を求め、操作線と気液平衡線から各段の気液組成が計算できる。下から数えたフィード段  $n_q$ 、全理論段  $n_p$  をLOCAL文で代入してケーススタディを行なう。最初は  $n_p$ 、 $n_q$  を大きめにしておくとも  $q$  線と操作線との交点と各段の液組成との比較から、まず  $n_q$  が求まり次に  $n_p$  も容易に判る。この例では  $n_p = 11$ 、 $n_q = 3$  が得られる(図3)。気液平衡線と  $q$  線との交点  $(x_f, y_f)$  は繰り返し計算が必要となるが(RES ET文の)独立変数として  $q = 0$  の時は  $x_f$  を、それ

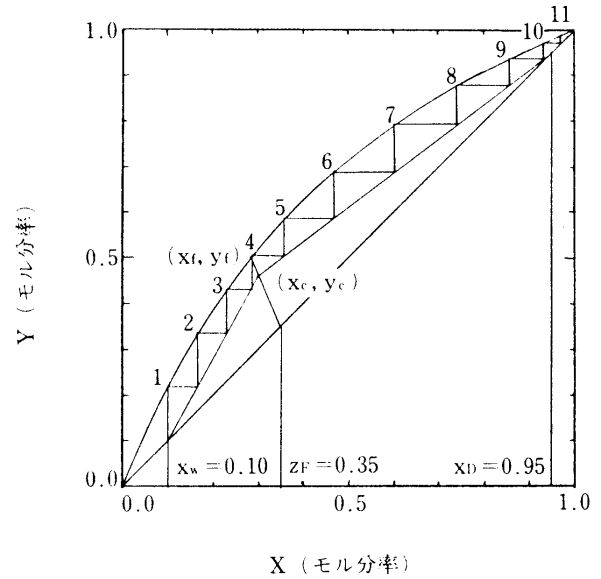


図3 McCabe-Thiele法による作図

```

1:      /* McCabe Thiele法による蒸留計算 */
2:      /*      参考：化学工学便覧 改訂四版 p.594 */
3:
4:      /* ローカル変数の定義 */
5:      LOCAL np=11,nq=3          /* np : 全理論段数 */
6:      LOCAL npm1=np-1,nq1=nq+1,nq2=nq+2 /* nq : 回収部理論段数 */
7:
8:      /* 各段気液組成の定義 */
9:      VAR X(np),Y(np)
10:
11:     /* 各段の気液平衡関係 */
12:     Y = ALPHA*X/(1+(ALPHA-1)*X)
13:
14:     /* 塔全体の収支 */
15:     F*zf = D*xd + W*xw
16:     F = D + W
17:     L = R + D
18:
19:     /* 還流比 */
20:     Rmin/(Rmin+1) = (xd-yf)/(xd-xf)
21:     R = Rmin*1.5
22:
23:     /* 塔底段 */
24:     X(1) = xw
25:
26:     /* 回収部各段 */
27:     Y(1:nq) = ((L+q*F)*X(2:nq1)-W*xw)/(L+q*F-W)
28:
29:     /* 濃縮部各段 */
30:     Y(nq1:npm1) = (R*X(nq2:np)+xd)/(R+1)
31:
32:     /* フィード */
33:     e1:yf/(1-yf) = ALPHA*xf/(1-xf)
34:     zf = q*xf + (1-q)*yf
35:
36:     /* 濃縮部操作線とq-線の交点 */
37:     xc = zf      WHEN q == 1 ..
38:     = (zf/(q-1)+xd/(R+1))/(q/(q-1)-R/(R+1)) WHEN q != 1
39:     yc = (R*xc+xd)/(R+1)
40:
41:     /* 収束計算の指定 */
42:     reset xf# 0.25[0,1] by e1
43:
44:     /* 値の定義 */
45:     zf=0.35 ; xd=0.95 ; xw=0.10 ; F=100 ; q=0.7
46:     ALPHA=2.5
47:

```

リスト7 蒸留計算



```

1: /* 単 蒸 留 の 計 算 */
2: /* 出典 : 化学工学便覧 改訂四版 p.593 (1978) */
3: LOCAL N=10
4: GLOBAL X=X,Y=L,L,DL=DL,LB=LB,DLB=DLB,H=H
5: VAR X(N),Y(N),L(N),DL(N),LB(N),DLB(N),XD(4),YD(4)
6: /* メタノール - 水系気液平衡データ
7: 出典 : J.Gmehling 他,VAPOR-LIQUID EQUILIBRIUM
8: DATA COLLECTION Vol.1 Part 1a,p.61 */
9: TABLE YD=TAB1(XD)
10: XD = (0.2491,0.2841,0.3673,0.4164)
11: YD = (0.6190,0.6530,0.7121,0.7410)
12: /* Modified Euler 法による単蒸留の計算 */
13: MACRO NSTEP
14:   X(I) = X(J) + H
15:   Y(I) = TAB1(X(I))
16:   DL(J) = L(J) / (Y(J)-X(J))
17:   LB(I) = L(K) + 2*H*DL(J)
18:   DLB(I) = LB(I) / (Y(I)-X(I))
19:   L(I) = L(J) + 0.5*H*(DLB(I)+DL(J))
20: END NSTEP
21: /* time step */
22: H = (XN-X(1)) / N
23: /* 2nd step */
24: Y(1) = TAB1(X(1))
25: DL(1) = L(1)/(Y(1)-X(1))
26: X(2) = X(1) + H
27: Y(2) = TAB1(X(2))
28: L(2) = L(1) + H*DL(1)
29: /* 3-N step */
30: CALL NSTEP(I=3:N,J=2:N-1,K=1:N-2)
31: /* distillate */
32: D = L(1) - L(N)
33: BET = D / L(1)
34: INPUT X(1),XN,L(1)
35: OUTPUT X,Y,L,D,BET
36: LB(1.2) = 0
37: DLB(1.2) = 0

```

## リスト 8 単蒸留

以外は $y_i$ を用いるとよい（あるいは $q = 1$ の時 $y_i$ を、それ以外は $x_i$ ）。濃縮部・回収部各段の $X$ と $Y$ の関係はリストのように簡単に表現できる。37, 38行はWHEN文であり、 $q = 1$ のとき $x_c = z_i$ 、 $q \neq 1$ のとき $x_c = 38$ 行目の式であることを意味する。このようにWHEN文を使用して変数の変域により、使用する式を変えることができる。

本例では段の番号づけを下から行なっているが、上からの場合には $Y$ から $X$ を求めることになり困難を伴う。非理想溶液の場合には、活量係数を液組成の関数として表わせばよい。

## 7. 単蒸留

数値積分の例として単蒸留の計算例をリスト 8 に示す。本例は数値求積法により簡単に計算できるが、常微分方程式の初期値問題を解く例としてあえて修正オイラー法を用いて計算している。EQUATRAN-Mでは、現状では微分方程式を書けば解けるようには機能が拡張されていないが、本例のようにマクロを使用して計算することが可能である。

さて、2成分の単蒸留で仕込み量および組成、蒸留後の缶液組成から留出量を求める問題を考える。ある時点の缶残液 $L$ モル、組成 $x$ 、それに平衡な気相組成を $y$ とすると次式が成り立つ。

$$L' = \frac{L}{y-x} \quad \left( L' = \frac{dL}{dx} \right) \quad (7-1)$$

これを修正オイラー法を用いて書くと、

$$\bar{L}_{n+1} = L_{n-1} + 2hL'_n \quad (7-2)$$

$$L_{n+1} = L_n + \frac{1}{2}h(\bar{L}'_{n+1} + L'_n) \quad (7-3)$$

となる。リスト中、 $LB \rightarrow \bar{L}$ 、 $DL \rightarrow L'$ 、 $DLB \rightarrow \bar{L}'$ 、 $H \rightarrow h$ に対応している。 $L_2$ のみ、 $L_2 = L_1 + hL'_1$ により計算している。きざみ $h$ は仕込み組成と蒸留後の組成を $N$ 等分して決めている。

この(7-2),(7-3)式の部分をマクロ化することにより本問題を解くことができる。マクロをCALLしている部分(30行目)は、ここではFORTRANにおけるDOループの記述の役割を果たしている。

9~11行はTABLE文で、このようにデータを数値で与えればそれを折線近似して使用してくれる。本例では気液平衡データ数が少く、間隔が大きい

め計算値は精度不十分であるが、微分方程式への適用例としてあえてとりあげた。

### 8. フローシートシミュレータ

化学プロセスの設計や解析を行なう場合、フローシートシミュレータを用いて物質収支・熱収支の計算がなされる。特に研究や開発段階では装置の組み合わせやフローが頻繁に修正されるため計算が大変であるが、EQUATRAN-Mでは式の修正、入

出力関係の変更等が簡単であるため、それらに容易に対応でき、非常に便利な簡易フローシートシミュレータとなる。

リスト9に簡単な例として、2塩化エチレン生成プロセスの例を示した(図4)。原料のエチレンと塩素が反応し、2塩化エチレン(EDC)が生成される。エチレンの転化率は90%である。精留塔では塩素、エチレン、EDCの98%、92%、0.1%が分離され、リサイクルして反応器に戻るが5%は他の

```

1:  /* プロセス物質収支の計算 */
2:  /* 出典 : JUSE-GIFS プログラム利用の手引(第2版) p.3
3:  /* 日本科学技術研修所計算センター 1988年4月
4:  /* 2塩化エチレン生成プロセス */
5:
6:  var f(9,3) /* f(i,j) flow rate
7:             i = stream number
8:             j = component number
9:             ( CL2=1, C2H4=2, C2H4CL2=3 ) */
10: var conv /* conversion of C2H4 at reactor
11:          C2H4 + CL2 = C2H4CL2 */
12: var coef(3) /* 量論係数 */
13: var sepr(3) /* separation ratio at column */
14: var vent /* vent ratio */
15:
16: /* fresh feed */
17: f(1) + f(8) = f(3)
18: f(2) + f(3) = f(4)
19:
20: /* reactor */
21: f(4) - f(4,2) * conv * coef = f(5)
22:
23: /* column */
24: f(5) * sepr = f(6)
25: f(5) = f(6) + f(9)
26:
27: /* vent */
28: f(6) * vent = f(7)
29: f(6) = f(7) + f(8)
30:
31: /* input data */
32: f(1) = ( 100, 0, 0 )
33: f(2) = ( 0, 100, 0 )
34: coef = ( 1.0, 1.0, -1.0 )
35: sepr = ( 0.98, 0.92, 0.001 )
36: conv = 0.90 ; vent = 0.05
37:

```

[ 計算結果 ]

```

f =
      ( 1)      ( 2)      ( 3)
1) 100          0          0
2) 0            100         0
3) 118.629048  9.577033    0.093777
4) 118.629048  109.577033   0.093777
5) 20.009719   10.957703   98.713107
6) 19.609525   10.081087    0.098713
7) 0.980476    0.504054    0.004936
8) 18.629048   9.577033    0.093777
9) 0.400194    0.876616    98.614394
conv = 0.9
coef =
  1) 1          2) 1          3) -1
sepr =
  1) 0.98       2) 0.92       3) 0.001
vent = 0.05

```

リスト9 プロセス物質収支

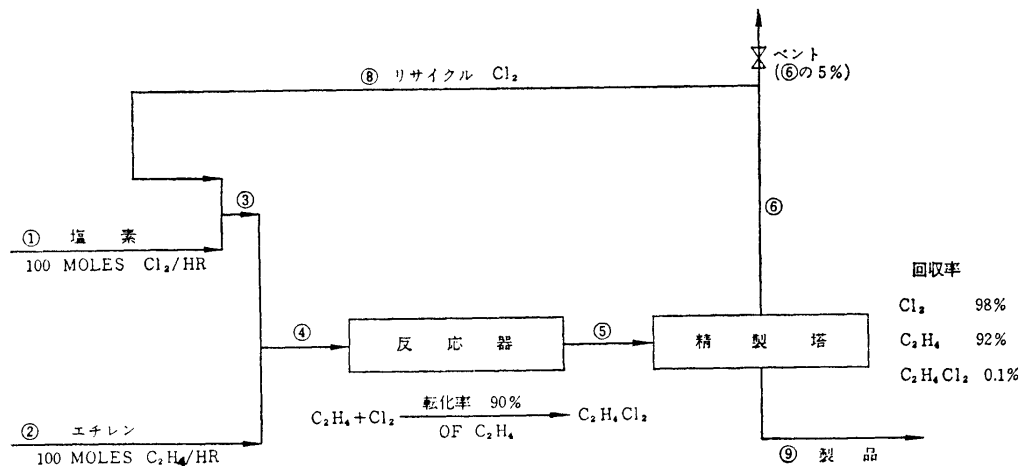


図4 フローシートシミュレータ

工程へ分岐する。この条件で系外（ストリーム⑨）へ流出するEDC量を求める。各ストリームの成分流量を  $f(i, j)$  の2次元配列で表わす。  $i$  はストリーム、  $j$  は成分番号とする。EQUATRAN-Mでは  $f(4, 2)$  はストリーム④の第2成分流量（反応器入口のエチレン流量）を、また  $f(4)$  のように第2添字を省略すると  $f(4, 1) \sim f(4, 3)$  のストリーム④の各成分流量を意味する。従ってフローに沿って物質収支式を記述し、必要な係数を定義すればリストのようなマトリックス形の解が得られる。また、入出力関係を変更して、あるストリーム中の特定成分量（濃度）を固定し、他の変数を求めるようなケーススタディや、コストを最小とする最適化の応用、熱収支式の組み込みなど適用範囲は広い。

### 9. おわりに

以上、分離における数値計算について述べてきたが、より詳しくまたは他分野への利用については他の文献<sup>6-10)</sup>を参照されたい。いくつかの例に示したようにEQUATRAN-Mは式の変形を必要としない「式言語」または「デジタル型ノモグラフ」とでもいべきものである。1セットの方程式を用意すれば簡単に使うことができる。今後、微分方程式に対応するなど、機能拡張に努める予定である。

### 参考文献

- 1) 宮原, 佐渡友, 化学工学, 34, 11, p.1157 (1970)
- 2) Milos Konopasek, Sundaresan Jayaraman The TK! Solver Book McGraw-Hill (1984)
- 3) Stewart M. Goldfarb, Chem. Eng. April 15, p. 91 (1985)
- 4) Robert Hirschel, Chem. Eng., April 15, p.93 (1985)
- 5) K. Miyahara, H. Sadotomo, K. Kitamura, J. of Chem. Eng. of Japan, 3, No. 2, p. 157 (1970)
- 6) G. Oguchi, M. Mitsunaga "A powerful language to solve a set of nonlinear equations" International Congress "Contribution of Computers to the Development of Chemical Engineering and Industrial Chemistry", Paris March 7-10 (1978)
- 7) 小口, 横山, 佐藤 "プロセス物質収支への方程式解法言語の応用" 化学工学協会プロセスシステム工学総合シンポジウム, 1985年3月 於芦屋
- 8) 小口, 横山, 佐藤, "パソコン用方程式解法プログラムの開発" 化学工学協会第50年会1985年3月
- 9) 佐渡友, 宮原, ケミカルエンジニアリング, 1985年8月, vol.30, No. 8, p. 63
- 10) 小口 悟郎, 日経コンピュータ 1985.9.30号(予定)