

C 3. 数式処理による常微分方程式の解法のための プログラミング技法(1)

渡辺 隼郎 (京都大学数理解析研究所)

§ 1. 序

常微分方程式を数式処理によつて解く時に生ずる様な技法を紹介するのが本稿の目的であり、この続稿もそれに準ずる予定である。本稿では整数係数の一変数多項式を有限回の手続きで既約な多項式の積に因数分解する方法を二つ提案する。有限回で因数分解を行なう方法は既にクロネッカーの方法(文献3参照)として知られているものがあるが、今提案する方法は、特殊な場合にプログラムから見て有利な方法を与える。しかしその優劣は実際の使用報告を待たねばならない。

因数分解は数式処理の様々の分野においてあらわれる。例えば有理数を係数とする有理関数の不定積分を求めるには、その有理関数を部分分数展開しなければならないが、それにはその分母の多項式を有理数の範囲で因数分解することが必要である。また有理数係数の線形常微分方程式の解を演算子法によつて解くときには、演算子のなす有理数係数の多項式を因数分解することが必要である。

さらに常微分方程式を、その特異点に注目して取扱うことを考えたとき、一番弱い特異性をもつ確定特異点における解が、解の特異性である対数項を本当に含むかどうかを判定するアルゴリズムがやはり有理数係数の多項式を因数分解することによつて得られることが示される(文献1, 2参照)。

この事実は、考えている確定特異点が見かけ上のものであるかどうかを判定することであり、常微分方程式を機械化する上で欠く事のできないものである。

今のことをより詳しく説明する。線形常微分方程式

$$(1) \quad (x-a)^n y^{(n)} + (x-a)^{n-1} P_1(x) y^{(n-1)} + \cdots + P_n(x) y = 0$$

において、 $P_1(x), \dots, P_n(x)$ が $x=a$ で正則なるとき、 $x=a$ はこの微分方程式の確定特異点であるといひ、(1) は $x=a$ において (1) に対応する決定方程式

$$(2) \quad f_0(\lambda) \equiv \lambda(\lambda-1)\cdots(\lambda-n+1) + P_1(a)\lambda(\lambda-1)\cdots(\lambda-n+2) + \cdots + P_n(a) = 0$$

の n 個の根 $\lambda = \lambda_1, \dots, \lambda_n$ に対応する、 n 個の互に独立な

$$(3) \quad \phi(x) \equiv (x-a)^{\lambda} \{ \phi_0(x) + \phi_1(x) \log(x-a) + \cdots + \phi_n(x) (\log(x-a))^n \}$$

の形の解をもち、 $\phi_i(x)$ は $x=a$ で正則で $\phi_i(a)$ の中には0でないものがある。以上の

ことが常微分方程式論により分っている。さて解(3)の中に対数項 $\log(x-a)$ が出現するかどうかは、決定方程式 $f_0(\lambda)=0$ が重根をもつかどうか、及びその根の中に互に整数差をもっているかどうかにより判定することができる。これを $f_0(\lambda)$ を因数分解することにより判定することが出来ることを示したのが先にあげた文献である。 $P_i(x)$ が有理数係数の有理関数で a が有理数なら $P_i(a)$ は有理数となるから、結局有理数係数の多項式を因数分解すればよい。これは整数係数の多項式を整数の範囲で因数分解することに帰着される。

§ 2. 因数分解 — 未定係数法 —

$f(\lambda)=a_n\lambda^n+\cdots+a_0$ を因数分解するには $1\leq k\leq (\frac{n}{2})$ なる k を与えたとき、 k 次の多項式で $f(\lambda)$ の因数となりうるものを一つ見出すことができればよい。そこで $f(\lambda)$ が

$$(4) \quad a_n\lambda^n+\cdots+a_0=(b_k\lambda^k+\cdots+b_0)(c_l\lambda^l+\cdots+c_0)$$

$(k+l=n)$ と因数分解できたとして、未定係数 b_i, c_i を有限回の手続きで求めようというのである。(4) の右辺を展開して両辺の λ の指数の等しい所を比較して次の関係を得る。但し $k\leq l$ とする。

$$(5) \quad \begin{cases} a_n=b_k c_l \\ a_{n-1}=b_{k-1} c_l+b_k c_{l-1} \\ \dots\dots\dots \\ a_l=b_0 c_l+b_1 c_{l-1}+\cdots+b_k c_{l-k} \\ \dots\dots\dots \\ a_k=b_0 c_k+b_1 c_{k-1}+\cdots+b_k c_0 \\ a_{k-1}=b_0 c_{k-1}+b_1 c_{k-2}+\cdots+b_{k-1} c_0 \\ \dots\dots\dots \\ a_1=b_0 c_1+b_1 c_0 \\ a_0=b_0 c_0 \end{cases}$$

上の $a_0=b_0 c_0$ と $a_n=b_k c_l$ より可能な組合せ $(b_0, c_0; b_k, c_l)$ が有限個定まる。この一つの組に対して b_0, c_0, b_k, c_l は定数と考えてよい。いま $a_0 \neq 0, a_n \neq 0$ と仮定しているから、 $b_0 \neq 0, b_k \neq 0$ 、これと $a_1=b_0 c_1+b_1 c_0$ より $c_1=r_1 b_1+r_2$ r_i : 有理数なる関係式をうる。さらに $a_2=b_0 c_2+b_1 c_1+b_2 c_0$ より $c_2=b_1$ と b_2 の有理数係数多項式なる関係をうる。こうして(5)の下から順に c_i を b であらわしていくと

$$(6) \quad \begin{cases} c_i=h_i(b_1, b_2, \dots, b_i) & i \leq k \quad i=1, \dots, l-1 \\ c_i=h_i(b_1, b_2, \dots, b_k) & i > k \end{cases}$$

(ここに h_i は中の変数に関する有理数係数多項式)とあらわせる。全く同じ様に(5)の上から c_i を順に b であらわしていくと

$$(7) \begin{cases} c_{l-i} = g_{l-i}(b_{k-1}, b_{k-2}, \dots, b_{k-i}) & i \leq k \quad i=1, \dots, l-1 \\ c_{l-i} = g_{l-i}(b_{k-1}, b_{k-2}, \dots, b_0) & i > k \end{cases}$$

(6) と (7) を等置することにより,

$$(8) \quad m_i(b_1, \dots, b_{k-1}) = 0 \quad i=1, \dots, l-1$$

なる有理数係数の連立代数方程式を得る. (8) は必要ならば適当な数を両辺に掛けることにより整数係数とみなすことができる. (8) の整数解を求めるにはクロネッカーの一般消去法に従つて (文献 3, 4 参照), 変数を一つづつ逐次消去していつて, 最後に一変数の代数方程式が得られる. それを

$$(9) \quad e_m b_i^m + \dots + e_0 = 0$$

とする. この方程式の根の限界は $(\max_{0 \leq i \leq m} e_i) + 1 = M$ だから $b_i = -M, -M+1, \dots, -1, 0, 1, \dots, M$ を順に (9) に代入してみることににより, (9) の整数解を全て求めることができる.

さて, (8) を消去していく途中で b_i と b_j ($i \neq j$) の二つの変数に関する代数方程式が得られているのであるが, この b_i に今の整数解を代入して有限個の b_j に関する代数方程式を得る. 今度これを解くことにより可能な (b_i, b_j) の組合せが全て求まる. これを繰返して可能な $(b_0, b_1, \dots, b_k)$ を有限個得る. これで $f(\lambda)$ を実際に割ることにより $f(\lambda)$ の k 次の因数を全て求めることが出来る. なお (4) で $\lambda=1$ とおいて得られる式

$$(10) \quad a_n + \dots + a_0 = (b_k + \dots + b_0)(c_l + \dots + c_0)$$

より $b_k + \dots + b_0$ のとり得る値が有限個だから, これを利用することも出来る. 整数解がなければ既約なことも明らか.

今までの事から, $k=1, 2, 3$ のときは一つの未知数 b_1 を決定すればよい. $k=1, 2$ のときは一次方程式をとけばよく, $k=3$ のときは b_1 の高次方程式をとけばよい. $k=4$ のときは二つの未知数 b_1, b_2 に関する連立方程式を解けばよい等々が分る.

§ 3. 未定係数法による因数分解の例

$$(11) \quad \lambda^7 + 3\lambda^4 + \lambda^3 + 2\lambda + 2 = (\lambda^3 + 2)(\lambda^4 + \lambda + 1)$$

関係 (5) をこの場合につづてみると

$$(12) \quad \begin{cases} 1 = b_3 c_4 \\ 0 = b_2 c_4 + b_3 c_3 \\ 0 = b_1 c_4 + b_2 c_3 + b_3 c_2 \\ 3 = b_0 c_4 + b_1 c_3 + b_2 c_2 + b_3 c_1 \\ 1 = b_0 c_3 + b_1 c_2 + b_2 c_1 + b_3 c_0 \\ 0 = b_0 c_2 + b_1 c_1 + b_2 c_0 \\ 2 = b_0 c_1 + b_1 c_0 \\ 2 = b_0 c_0 \end{cases}$$

(12) のはじめの式から $b_3=c_4=1$ が分る. (10) は $9=(3+b_1+b_2) \times (2+c_1+c_2+c_3)$ これから $b_1+b_2=-2, 0, 6$ を得る. $b_1+b_2=0$, 及び (12) の最後の式から $b_0=2, c_0=1$ において (8) を作ると

$$(13) \begin{cases} c_1 = -\frac{1}{2} b_1 + 1 = -b_2^3 + 2b_2 b_1 + 1 \\ c_2 = \frac{1}{4} b_1^2 - \frac{1}{2} b_1 - \frac{1}{2} b_2 = b_2^2 - b_1 \\ c_3 = -\frac{1}{8} b_1^3 + \frac{1}{4} b_1^2 + \frac{1}{2} b_1 b_2 - \frac{1}{2} (b_2 + b_3 - 1) = -b_2 \end{cases}$$

(13) の一番はじめの式に $b_2 = -b_1$ を代入して

$$2b_1^3 + 4b_1^2 + b_1 = 0$$

を得る. 根の限界は 5 で, 整数解は $b_1=0$ のみである. よつて $b_2=0$, 従つて $b_3 \lambda^3 + \dots + b_0 = \lambda^3 + 2$ が分る.

$b_1+b_2=-2$ 及び (8) の場合, さらに $b_0=1, c_0=2$ の場合とも因数とにならないことが分るから, これで因数分解ができたことになる.

§ 4. 因数分解 — 基数代入法 —

$f(\lambda) = \lambda^2 - \lambda - 6$ を係数に正負の数を許した λ 進数と見なした時, $\lambda=8$ とおいてみる. $f(8)=50$ これを係数に正負の数を許した 8 進数と見なした数に因数分解してみる.

$$1 \cdot 50 = (8-7)(6 \cdot 8+2) = (8-7)(7 \cdot 8-6)$$

$$2 \cdot 25 = (8-6)(3 \cdot 8+1) = (8-6)(4 \cdot 8-7)$$

$$5 \cdot 10 = (8-3)(8+2) = (8-3)(2 \cdot 8-6)$$

ここでもし $f(\lambda)$ の因数の係数の絶対値が 7 を越えなければ, $f(\lambda)$ は $(\lambda-7)(6\lambda+2)$, $(\lambda-7)(7\lambda-6)$, $(\lambda-6)(3\lambda+1)$, $(\lambda-6)(4\lambda-7)$, $(\lambda-3)(\lambda+2)$, $(\lambda-3)(2\lambda-6)$ のどれかに因数分解されるはずだという推論が基数代入法の基本的な考え方である.

正確にアルゴリズムを記述する為にまず次の性質を証明しよう.

< 正の整数 $\lambda (\geq 2)$, N, n を任意に与えたとき

$$(14) N = d_n \lambda^n + d_{n-1} \lambda^{n-1} + \dots + d_1 \lambda^1 + d_0 \quad (d_i: \text{整数} \mid |d_i| < \lambda \quad d_n \neq 0)$$

とあらわすことはできないか, できるとしても有限個しかなく, それは λ, N, n を与えたとき有限回の手続きで全て求ることができる.>

$d_n \lambda^n + \dots + d_0 \leq \lambda^{n+1} - 1$ だから $N \geq \lambda^{n+1}$ ならば N を (14) のようにあらわすことはできない.

$1 \leq N \leq \lambda^{n+1} - 1$ なるときを考えよう.

$$(15) \tau(\lambda) = t_m \lambda^m + t_{m-1} \lambda^{m-1} + \dots + t_0 \quad |t_i| < \lambda \quad i=0, \dots, m$$

は (1) $|t_i+1| < \lambda$, $|t_{i-1}-\lambda| < \lambda$ なら

$$\tau(\lambda) = t_m \lambda^m + \cdots + (t_i+1) \lambda^i + (t_{i-1}-\lambda) \lambda^{i-1} + \cdots + t_0$$

(2) $|t_i-1| < \lambda$, $|t_{i-1}+\lambda| < \lambda$ なら

$$\tau(\lambda) = t_m \lambda^m + \cdots + (t_i-1) \lambda^i + (t_{i-1}+\lambda) \lambda^{i-1} + \cdots + t_0$$

とやはり (15) の形に書きあらわすことができる. (1), (2) が両立することはない. なぜなら (1) は $1 \leq t_{i-1} \leq \lambda-1$, (2) は $-\lambda+1 \leq t_{i-1} \leq -1$ をあらわすからである. さらに $|t_{i-1} \pm 2\lambda| > \lambda$ だから $\tau(\lambda)$ のあらわし方は高々 2^m 個しかない.

n を固定すると 1 を (15) の形にあらわす方法は 1 通りで,

$$(16) \quad 1 = \lambda^n - (\lambda-1) \lambda^{n-1} - (\lambda-1) \lambda^{n-2} - \cdots - (\lambda-1) \lambda - (\lambda-1)$$

N を λ 進数表示したものを

$$(17) \quad N = t_m \lambda^m + \cdots + t_0 \quad t_m > 0, \quad t_i \geq 0, \quad i=0, \dots, m-1$$

とすると, 今 $m \leq n$ を考えているから (16), (17) により

$$(18) \quad N = \lambda^n - (\lambda-1) \lambda^{n-1} - \cdots - (\lambda-1) \lambda^{m+1} - \{(\lambda-1) - t_m\} \lambda^m - \cdots \\ - \{(\lambda-1) - t_1\} \lambda^1 - (\lambda - t_0)$$

を得る. ここで $\lambda-1 \geq t_i \geq 0$, $i=0, \dots, m$ だから $\lambda > \lambda-1-t_i \geq 0 \quad i=1, \dots, m$ 但し $t_0=0$ のとき一番下の桁から桁上げが起るが $0 \leq (\lambda-1)-t_m \leq \lambda-2$ だから, 桁上げは高々 λ^m で止る. 従つて

$$(19) \quad N = \lambda^n - (\lambda-1) \lambda^{n-1} - \cdots - (\lambda-1) \lambda^{m+1} + d_m \lambda^m + \cdots + d_0 \\ |d_i| < \lambda \quad i=0, \dots, m$$

と書ける. 又前に述べたことから, このあらわし方は高々 2^m 個である. これで証明が終つた.

次に $f(\lambda) = a_n \lambda^n + \cdots + a_0$ を与えたとき $a_i \quad i=0, \dots, n$ と, この多項式の任意の因子の係数 b_j, c_k に対して,

$$(20) \quad M \geq \max \{|a_i|, |b_j|, |c_k|\}$$

をみたす正の整数 M を求めるアルゴリズムを示そう.

$f(\lambda)$ の任意の因子を $g(\lambda)$ とすると $g(i)$ は $f(i)$ を割り切る ($i=0, 1, \dots, n$). 従つて点 $(i, f(i))$ の任意の因数 $i=0, 1, \dots, n$ を通る多項式 $\phi_l(\lambda)$ 中のどれかに $g(\lambda)$ は一致する. ゆえに, $f(\lambda)$ の任意の因子の係数の絶対値の最大なるものは, $\phi_l(\lambda)$ の係数の絶対値の最大なるものを越えない. k 次 ($1 \leq k \leq n$) の $\phi_l(\lambda)$ は点 $\lambda=0, 1, \dots, k$ における $f(\lambda)$ の値からニュートンの補間公式によつて

$$(21) \quad \phi_l(\lambda) = \mu_k \lambda(\lambda-1) \cdots (\lambda-k+1) + \cdots + \mu_1 \lambda + \mu_0$$

とおいた係数 μ_i を次の差分の表より $\mu_i = \frac{d^i f(0)}{\lambda!}$ において得られる.

$$(22) \begin{cases} f(0) & \Delta f(0) & \Delta^2 f(0) & \cdots \\ f(1) & \Delta f(1) & \Delta^2 f(1) & \cdots \\ f(2) & \Delta f(2) & \Delta^2 f(2) & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ f(k) & \Delta f(k) & \Delta^2 f(k) & \cdots \end{cases}$$

ここで $|f(i)| \leq m$ とすると $|\Delta f(0)| \leq 2m$, 一般に $|\Delta^i f(0)| \leq 2^i m$ だから

$|\mu_i| \leq \frac{2^i}{i!} m$ を得る. $\phi_l(\lambda)$ の x^i の係数を c_i とすると

$$(23) \quad |c_i| \leq |\mu_i| + \left(\sum_{j_1=1}^i j_1 \right) |\mu_{i+1}| + \left(\sum_{j_1 \neq j_2}^{i+1} j_1 \cdot j_2 \right) |\mu_{i+2}| + \cdots + \left(\sum_{j_1 \neq j_m}^k j_1 \cdots j_{k-i} \right) |\mu_k| \\ \leq m \left\{ \frac{2^i}{i!} \left[1 + \left(\sum_{j_1=1}^i j_1 \right) \frac{2}{i+1} + \left(\sum_{j_1 \neq j_2}^{i+1} j_1 \cdot j_2 \right) \frac{2^2}{(i+1)(i+2)} + \cdots \right. \right. \\ \left. \left. + \left(\sum_{j_1 \neq j_m}^k j_1 \cdots j_{k-i} \right) \frac{2^{k-i}}{(i+1) \cdots k} \right] \right\}$$

この右辺の $\{ \}$ の中は k, i, n のみに依存する. 定数だから, n を固定したとき $1 \leq i \leq k \leq n$ において i, k を動かしたときの最大値が存在する. すなわち (23) の右辺の $\{ \}$ の中を $M(i, k, n)$ としたとき,

$$(24) \quad M(n) = \max_{1 \leq i \leq k \leq n} M(i, k, n)$$

とおく. このとき (20) の M として

$$(25) \quad M = m \cdot M(n)$$

とおけばよい.

最後に $f(\lambda) = a_n \lambda^n + \cdots + a_0$ を基数代入法によつて因数分解するアルゴリズムを与える. まず $f(\lambda)$ に対して (25) の M を求めて $f(M) = a_n M^n + \cdots + a_0$ の可能な因数を $P_1, P_2, \cdots, P_q$ とする. $P_i > 0$ と考えてよい. いま $f(\lambda)$ の l 次の因子を求めているとすると $P_i = c_l^{(i)} M^l + \cdots + c_0^{(i)}$ と書きあらわす. このあらわし方は (14) により有限個であつて, それを次々に求めるアルゴリズムは知られている. そこで $f(\lambda)$ を $c_l^{(i)} \lambda^l + \cdots + c_0^{(i)}$ で割つてみて割れたらそれが因子であるし, そうでなかつたら別の形に直して割つてみるという繰り返し, さらにこれを $P_1, P_2, \cdots, P_q$ について繰り返す. 以上のどれでも割れなかつたら $f(\lambda)$ が既約であるということが分る.

§ 5. 注 意

以上でのべた, 未定係数法と基数代入法はそのどちらも, 原理だけを述べたものであつて, 反復回数をより少なくする為の工夫については述べなかつた. 例えば因子の次数の見当をつけるのに $\text{mod. } 2, \text{mod. } 3$ 等で割り算をしてみればよいことはよく知られている (文献

3 参照). また未定係数法において, 組 (b_0, c_0, b_k, c_l) を定める時に,

$$(26) \quad a_1 = b_0 c_1 + b_1 c_0$$

$$a_{n-1} = b_{k-1} c_l + b_k c_{l-1}$$

が整数解 b_1, c_1, b_k, c_l を持つように定めるべきである. すなわち b_0 と c_0 の最大公約数 (b_0, c_0) が a_1 を割り切る, また (b_k, c_l) が a_{n-1} を割り切るようにと選ぶと (b_0, c_0, b_k, c_l) なる組の数は少なくなるはずである. このような性質をもつと利用することができる. しかし注意しておかねばならない事は, 判断の数とプログラムの長さが比例し, 判定の数とループを廻る回数は反比例することである. これはその因数分解の目的に応じて様々に選ぶべきであつて, 一通りの答を期待できることではないということである.

これを実現するプログラムは最小限有理数係数の多変数多項式の加減乗除が出来る数式処理用のプログラムでなければならないことを注意しておく.

参 考 文 献

- [1] 福原満州雄: 常微分方程式の解法 II (線型の部), 1941
- [2] 渡辺隼郎: 数式処理の例題, 数研計算機構論シンポジウム予稿集, 1967
- [3] Van der Waerden Modern Algebra Vol. I, II 1953
- [4] Joel, M., Solution of systems of polynomial equations by elimination Comm. ACM (Aug. 1966) 634-637

本 PDF ファイルは 1968 年発行の「第 9 回プログラミング・シンポジウム報告集」をスキャンし、項目ごとに整理して、情報処理学会電子図書館「情報学広場」に掲載するものです。

この出版物は情報処理学会への著作権譲渡がなされていませんが、情報処理学会公式 Web サイトの https://www.ipsj.or.jp/topics/Past_reports.html に下記「過去のプログラミング・シンポジウム報告集の利用許諾について」を掲載して、権利者の検索をおこないました。そのうえで同意をいただいたもの、お申し出のなかったものを掲載しています。

過去のプログラミング・シンポジウム報告集の利用許諾について

情報処理学会発行の出版物著作権は平成 12 年から情報処理学会著作権規程に従い、学会に帰属することになっています。

プログラミング・シンポジウムの報告集は、情報処理学会と設立の事情が異なるため、この改訂がシンポジウム内部で徹底しておらず、情報処理学会の他の出版物が情報学広場 (=情報処理学会電子図書館) で公開されているにも拘らず、古い報告集には公開されていないものが少からずありました。

プログラミング・シンポジウムは昭和 59 年に情報処理学会の一部門になりましたが、それ以前の報告集も含め、この度学会の他の出版物と同様の扱いにしたいと考えます。過去のすべての報告集の論文について、著作権者（論文を執筆された故人の相続人）を探し出して利用許諾に関する同意を頂くことは困難ですので、一定期間の権利者搜索の努力をしたうえで、著作権者が見つからない場合も論文を情報学広場に掲載させていただきたいと思います。その後、著作権者が発見され、情報学広場への掲載の継続に同意が得られなかった場合には、当該論文については、掲載を停止致します。

この措置にご意見のある方は、プログラミング・シンポジウムの辻尚史運営委員長 (tsuji@math.s.chiba-u.ac.jp) までお申し出ください。

加えて、著作権者について情報をお持ちの方は事務局まで情報をお寄せくださいますようお願い申し上げます。

期間：2020 年 12 月 18 日～2021 年 3 月 19 日

掲載日：2020 年 12 月 18 日

プログラミング・シンポジウム委員会

情報処理学会著作権規程

<https://www.ipsj.or.jp/copyright/ronbun/copyright.html>