# 多項式関数を関数値から決定する諸アルゴリズムの比較

野　崎　昭　弘

サイバー大学 IT 総合学部・教授

**要　旨**

$n$ 次多項式で表わされる関数のすべての係数を，与えられた $n+1$ 組の「変数値と関数値のペア」から決定する問題は，いろいろなアルゴリズムで解くことができる。すぐ思いつくのは $n+1$ 元連立１次方程式に帰着させる方法であるが，ほかに関孝和の「累裁招差之法」があるし，「補間多項式」から導く方法も考えられる。ここでは①連立１次方程式を解く通常の方法を詳しく分析して，②「累裁招差之法」と比較し，③分析の結果から招差之法の改良版を示し（ここまでは要点が［１］に述べられている），さらに計算量の観点から，④ラグランジュの補間多項式，⑤エイトケンの補間法，⑥ニュートンの補間公式から導く方法とあわせて比較して見た。その結果，計算量は①，②，④，⑤が $n^3$ のオーダーであるのに対して，③，⑥は $n^2$ のオーダーであるから，これらが圧倒的に速いことがわかった。

**キーワード**：多項式の係数の決定，計算量，累裁招差之法，ニュートンの補間公式

## 1. 多項式関数の決定とは

### A. 問　題

与えられた観測データから理論式のパラメータを決定する問題は，理論式が $n$ 次多項式である場合には，次のように述べられる。

〈GIVEN〉変数値と関数値の $n+1$ 組のペア $(x_k, y_k)$, $0 \leqq k \leqq n$

〈WANTED〉すべての $k$ について $f(x_k) = y_k$ をみたす多項式関数

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0$$

の，すべての係数 $a_k (0 \leqq k \leqq n)$ の値

〈例〉　$x_0 = 2,\ x_1 = 3,\ x_2 = 5,\ x_3 = 11$ に対して

$y_0 = 15,\ y_1 = 52,\ y_2 = 246,\ y_3 = 2652$

原稿受付日：2008 年 12 月 2 日

原稿受理日：2009 年 2 月 13 日

となるような3次関数 $y=a_3x^3+a_2x^2+a_1x+a_0$ のすべての係数を求める。

補間法を使えば，係数を決定しなくても上記のデータだけから，与えられた変数値に対する関数値を計算できる。しかし $n$ 次多項式の関数値をひとつ計算するには，係数がわかっていれば乗算・加減算 $n$ 回ずつでできるのに対して，補間法では（変数値に関係ないパラメータの計算はすんでいるとしても），ラグランジュの補間多項式では乗除算・加減算約 $n^2$ 回ずつを要するし，ニュートンの補間式でも，$n$ 回の乗算と $2n$ 回の加減算が必要である。したがって，多くの変数値について繰り返し関数値を計算したい場合には，すべての係数を求めておいたほうがよい。また「ベキ和の公式を求める」などの場合には，係数を求めること自体が目的になる（[1]）。

ここではまず「連立1次方程式を解いて求める」方法の分析から始めて，そのあとに関孝和の方法，さらに補間式から求める方法との比較を行いたい。

## B. 連立1次方程式とガウスの消去法

さきほどの例について，未知の係数 $a_3$, $a_2$, $a_1$, $a_0$ は，明らかに次の4元連立1次方程式（$n$ 次多項式なら $n+1$ 元の方程式）をみたす。

$$
\begin{aligned}
&(0.0):\quad 2^3a_3+2^2a_2+2a_1+a_0=15\\
&(0.1):\quad 3^3a_3+3^2a_2+3a_1+a_0=52\\
&(0.2):\quad 5^3a_3+5^2a_2+5a_1+a_0=246\\
&(0.3):\quad 11^3a_3+11^2a_2+11a_1+a_0=2652
\end{aligned}
\qquad (\mathrm{I})
$$

これは係数行列が“ヴァンデルモンド型”の，きわめて特殊な方程式であるが，もちろん一般的な解法でも解ける。たとえばガウスの消去法によれば，次の手順に従うことになる（よく知られたことであるが，計算量の評価も含めて，たとえば［2］参照）。

（G1）前進消去：対角要素の標準化と，その下の未知数の消去

$$
\begin{aligned}
&a_3+(1/2)a_2+(1/4)a_1+(1/8)a_0=15/8\\
&a_2+(5/6)a_1+(19/36)a_0=-11/36\\
&a_1+(31/30)a_0=1/30\\
&a_0=1
\end{aligned}
\qquad (\mathrm{II})
$$

これで $a_0=1$ が求まる。ここまでは多項式の次数を $n$ として，乗除算・加減算ともに，およそ $(1/3)(n+1)^3$ 回ずつでできることが知られている。

（G2）後退代入：代入法で，$a_1$, $a_2$, $a_3$ を求める

たとえば第3式から

$$a_1=1/30-(31/30)a_0=1/30-(31/30)=-1$$

という要領で，

$$a_1 = -1,\ a_2 = 0,\ a_3 = 2$$

をその順に求めることができる。

なお後半（G 2）で必要な演算回数は，乗除算・加減算ともに $(1/2)n(n+1)$ 回ずつである。したがって，ガウスの消去法の時間的主要部分は前半（G 1）で，全演算回数は乗除算・加減算ともにおよそ $(1/3)(n+1)^3$ である。しかしもう少し精密に言うと，次のようになる（[ 2 ]，41－42 ページ参照)。

乗除算は　約 $(1/3)n^3+2n^2$ 回,

加減算は　約 $(1/3)n^3+(3/2)n^2$ 回

### C. 逆順の消去法

“ヴァンデルモンド型” の方程式（ I ）は，$a_0$ がすぐ消去できる形になっている。たとえば

$$(1.0) = (0.1)-(0.0):\quad 19a_3+5a_2+a_1 = 37$$

で $a_0$ が消える。同様に

$$(0.2)-(0.1):\quad 98a_3+16a_2+2a_1 = 194$$

も求まるが，この両辺を $x_2-x_1 = 5-3 = 2$ で割っておくと

$$(1.1):\quad 49a_3+8a_2+a_1 = 97$$

となる。さらに

$$(0.3)-(0.2):\quad 1206a_3+96a_2+6a_1 = 2406$$

であるが，この両辺を $x_3-x_2 = 11-5 = 6$ で割っておくと次のようになる。

$$(1.2):\quad 201a_3+16a_2+a_1 = 401$$

こうして得られた方程式 (1.0)，(1.1)，(1.2) は，$a_1$ がすぐに消去できる形になっている：

$$(1.0):\quad 19a_3+5a_2+a_1 = 37$$
$$(1.1):\quad 49a_3+8a_2+a_1 = 97$$
$$(1.2):\quad 201a_3+16a_2+a_1 = 401$$

$$(1.1)-(1.0):\quad 30a_3+3a_2 = 60$$

この両辺を $x_2-x_0 = 5-2 = 3$ で割っておくと

(2.0): $10a_3+a_2 = 20$

(1.2)−(1.1): $152a_3+8a_2 = 304$

この両辺を $x_3-x_1 = 11-3 = 8$ で割っておくと：

(2.1): $19a_3+a_2 = 38$

あとはこれらの差を取って，

(2.2)−(2.1): $9a_3 = 18$

この両辺を $x_3-x_0 = 11-2 = 9$ で割れば

(3.0): $a_3 = 2$

が得られる。

以上の結果から，あとで役に立つ方程式を抜き出すと，次のようになる：

$$
\begin{aligned}
&(0.0):\quad 8a_3+4a_2+2a_1+a_0 = 15\\
&(1.0):\quad 19a_3+5a_2+a_1 = 37\\
&(2.0):\quad 10a_3+a_2 = 20\\
&(3.0):\quad a_3 = 2
\end{aligned}
\qquad (\text{III})
$$

このようにすれば，「係数も変数値もすべて整数」である場合には整数計算だけですむので，手計算はずっとラクになる。またここから先は通常のガウス法と同様で，$a_2, a_1, a_0$ がその順に代入法で，乗除算・加減算ともに $(1/2)n(n+1)$ で求められる。しかしこの形（Ⅲ）を上のような消去法で求める場合は，全体としての演算回数はあまり変わらないので，演算回数のオーダーはやはり $n^3$ である（あとで詳しく調べる)。

## 2. 逆順消去法の分析

### A. 差分商 $\delta_k$

方程式（Ⅰ）から方程式（Ⅲ）に至る途中経過を分析するために，

変数値の列　$x_0, x_1, x_2, x_3, \cdots$

関数値の列　$y_0, y_1, y_2, y_3, \cdots$

に対する作用素 $\delta_k$ を，次のように定義しておく。

〈注意〉　この作用素は差分商（あるいは差商，divided difference）と呼ばれ，ニュートン（1642-1727）の補間公式でも使われているが，関孝和（1642？-1708）の「累裁招差法」でも使われている，おもしろい作用素である（どちらも後出)。

定義 1

(1) $\delta_0 y_t = y_t$

(2) $k > 0$ に対して

$$\delta_k y_t = \frac{\delta_{k-1} y_{t+1} - \delta_{k-1} y_t}{x_{t+k} - x_t}$$

〈例〉 $\delta_1 y_0 = \dfrac{\delta_0 y_1 - \delta_0 y_0}{x_1 - x_0} = \dfrac{y_1 - y_0}{x_1 - x_0}$,

$$\delta_2 y_0 = \frac{\delta_1 y_1 - \delta_1 y_0}{x_2 - x_0} = \frac{1}{x_2 - x_0}\left(\frac{y_2 - y_1}{x_2 - x_1} - \frac{y_1 - y_0}{x_1 - x_0}\right)$$

なお逆順の消去過程における式 $(k.\,t)$ の右辺は，$\delta_k y_t$ に一致する。

定理 1　作用素 $\delta_k$ は，同じ変数値の列 $x_0, x_1, x_2, \cdots$ に対する 2 つの関数値の列

$y_0, y_1, y_2, y_3, \cdots$

$z_0, z_1, z_2, z_3, \cdots$

に対して

$$\delta_k (c \cdot y_t + d \cdot z_t) = c \cdot \delta_k y_t + d \cdot \delta_k z_t$$

が成り立つという意味で，線型（linear）である。

これはよく知られた，ほとんど明らかな事実なので，証明は省略する。

## B. 多項式 $L(t, k, p)$

次に，式 $(k.\,t)$ の左辺の未知数 $a_p$ の係数を求めるために，関数 $y = x^p$ から決まる関数値の列

$x_0^p,\ x_1^p,\ x_2^p,\ x_3^p,\ \cdots$

に対して，差分商 $\delta_k$ がどのように働くかを調べておこう。

〈例〉 $\delta_1 x_t = \dfrac{x_{t+1} - x_t}{x_{t+1} - x_t} = 1$,

$$\delta_2 x_t = \frac{\delta_1 x_{t+1} - \delta_1 x_t}{x_{t+2} - x_t} = \frac{1-1}{x_{t+2} - x_t} = 0,$$

$k > 1$ のとき $\delta_k x_t = 0$,

$$\delta_1 x_1^2 = \frac{x_{t+1}^2 - x_t^2}{x_{t+1} - x_t} = x_{t+1} + x_t,$$

$$\delta_2 x_t^2 = \frac{(x_{t+2} + x_{t+1}) - (x_{t+1} + x_t)}{x_{t+2} - x_t} = \frac{x_{t+2} - x_t}{x_{t+2} - x_t} = 1,$$

$$\delta_3 x_t^2 = \frac{1-1}{x_{t+3}-x_t} = 0,$$

$$k > 2 \text{ のとき } \delta_k x_t^2 = 0$$

定義 2　変数 $x_0, x_1, x_2, x_3, \cdots$ の多項式 $L(t,\ k,\ p)$ を，次のように定義する。

$L(t, k, p) = x_t, x_{t+1}, \cdots, x_{t+k}$ から成る，$p$ 次の項の総和

〈注意〉「$p$ 次の項」とは「重複を許す，$p$ 個の変数の積」という意味である。
式で書けば：

$$L(t, k, p) = \sum_{i0+i1+\cdots+ik=p} x_t^{i0} x_{t+1}^{i1} \cdots x_{t+k}^{ik}$$

ただし $p = 0$ のときは，

$$L(t, k, 0) = 1$$

と定めておく。

〈例〉 $p > 0$ のとき $L(t, 0, p) = x_t^p$

$$L(t, 1, 1) = x_t + x_{t+1},$$

$$L(t, 1, 2) = x_t^2 + x_{t+1}^2 + x_t x_{t+1},$$

$$L(t, 2, 2) = x_t^2 + x_{t+1}^2 + x_{t+2}^2 + x_t x_{t+1} + x_t x_{t+2} + x_{t+1} x_{t+2}$$

これは $x$ のベキ乗の差分商 $\delta_k x_t^p$ と，密接な関係がある。以下にいくつか例を示そう。

$$\delta_0 x_t^2 = x_t^2 = L(t, 0, 2),$$

$$\delta_1 x_t^2 = x_{t+1} + x_t = L(t, 1, 1),$$

$$\delta_2 x_t^2 = 1 = L(t, 2, 0)$$

$$\delta_3 x_t^2 = \frac{1-1}{x_{t+3}-x_t} = 0$$

定理 2　0 以上 $p$ 以下の任意の整数 $k$ に対して，

$$\delta_k x_t^p = L(t, k, p-k)$$

〈証明〉 $k = 0$ の場合は，定義 1, 2 から

$$\delta_0 x_t^p = x_t^p = L(t, 0, p)$$

で，たしかに成り立つので，一般の場合を $k$ についての数学的帰納法で証明する。そこで以下，$0 < k \leqq p$ の場合を考える。帰納法の仮定から

$$\delta_k x_t^p = \frac{\delta_{k-1}x_{t+1}^p - \delta_{k-1}x_t^p}{x_{t+k} - x_t} = \frac{L(t+1,\, k-1, p-k+1) - L(t,\, k-1,\, p-k+1)}{x_{t+k} - x_t}$$

$k$ は何でもよいが，以下 $k = 3$ の例について説明する。分子の第 1 項は

$x_{t+1}, x_{t+2}, x_{t+3}$ の $p-2$ 次の積の和，

第 2 項は

$x_t, x_{t+1}, x_{t+2}$ の $p-2$ 次の積の和

であるが，$x_t, x_{t+3}$ を除く積

$$x_{t+1}^{i1} x_{t+2}^{i2}$$

が等しいものどうしを組み合わせると，それらの差は

$$\{x_{t+3}^{i0} - x_t^{i0}\} x_{t+1}^{i1} x_{t+2}^{i2}$$

となる。ここで

$$i0 + i1 + i2 = p-2$$

である。この差は，$i0 = 0$ の場合は 0 であるが，$i0 > 0$ の場合は分母の $x_{t+3} - x_t$ で割ると，

$$\frac{x_{t+3}^{i0} - x_t^{i0}}{x_{t+3} - x_t} = x_{t+3}^{i0-1} + x_{t+3}^{i0-2} x_t + x_{t+3}^{i0-3} x_t^2 + \cdots + x_t^{i0-1}$$

から，

$$\{(x_{t+3}^{i0} - x_t^{i0})/(x_{t+3} - x_t)\} \times x_{t+1}^{i1} x_{t+2}^{i2} = (x_{t+3}^{i0-1} + x_{t+3}^{i0-1} x_t + \cdots + x_t^{i0-1}) \times x_{t+1}^{i1} x_{t-2}^{i2}$$

となる。これを展開すれば $x_t, x_{t+1}, x_{t+2}, x_{t+3}$ の積の和になるが，各変数の指数は，$x_t$ の指数を j，$x_{t+3}$ の指数を h とすると

$$j,\ i1,\ i2,\ h$$

で，これらは条件

$$i0 > 0,$$
$$j + h = i0 - 1,$$
$$i0 + i1 + i2 = p - 2$$

のもとで任意であるから，全体として条件

$$\begin{aligned} & j + i1 + i2 + h \\ & = (i0 - 1) + i1 + i2 \\ & = i0 + i1 + i2 - 1 \end{aligned}$$

$= (p-2)-1 = p-3$

のもとで任意であり，それらの項の総和は

$x_t$ から $x_{t+3}$ までの，$p-3$ 次の積の総和，

すなわち $L(t, 3, p-3)$ に等しい。一般の $k$ についても同様である。

〈証明終わり〉

系

(1)　$k = p$ のとき，$\delta_p x_t^p = 1$

(2)　$k > p$ のとき，$\delta_k x_t^p = 0$

〈証明〉

(1)　定理 2 から

$$\delta_p x_t^p = L(t, p, 0) = 1$$

(2)　上の系(1)から

$$\delta_{p+1} x_t^p = \frac{\delta_p x_{t+1}^p - \delta_p x_t^p}{x_{t+p+1} - x_t} = \frac{1-1}{x_{t+p+1} - x_t} = 0$$

なので，$k = p+1$ のとき $\delta_k x_t^p = 0$ となる。$k > p+1$ の場合も同様。

〈証明終わり〉

さて，$y_t$ は $x_t$ の $n$ 次多項式で

$$\sum_{p=0}^{n} a_j x_t^p = y$$

と表せるのだから，これに $\delta_k$ を施すと，定理 1 と定理 2（およびその系）から，方程式 $(k.\ t)$ は

$$\sum_{p=k}^{n} L(t, k, p-k) a_p = \delta_k y_t$$

と表される。したがって，次の連立 1 次方程式が得られる $(0 \leqq k \leqq n)$。

$$\sum_{p=k}^{n} L(0, k, p-k) a_p = \delta_k y_0 \qquad \text{（Ⅲ）}$$

これが係数行列が「ヴァンデルモンド型」であるような方程式について成り立つ，方程式（Ⅲ）の一般形である。

## C. 関の方法

関孝和は，遺著『括要算法・巻元』の中で，記法は異なるが実質的に差分商を使い，

$$a_n = \delta_n y_0 \qquad (\text{Ⅲ})$$

を証明なしに用いて，$\delta_n y_0$ の計算によって最高次の係数 $a_n$ を求めている（[ 3 ]，pp. 273-282)。なおこの方法は「累裁招差之法」と呼ばれているが，係数のことを和算の用語で「差」というので，それを「招く」（求める）ことに基づく，という。しかし関は方程式（Ⅲ）は用いていないので，低次の項の係数を求めるには

$$(\text{新})y_t = (\text{旧})y_t - a_n x_t^n$$

を計算して多項式の次数を下げ，差分商の計算から「最高次の係数を求める」ことを繰り返している（それが「累裁」？)。その手順を整理して，実行される演算回数を評価してみると，次のようになる。

（S 0）最初にあとの準備として，$x_t^k(0 \leqq t < n,\ 2 \leqq k \leqq n)$ の値を求めておく。それは $n\times(n-1) = n^2-n$（回）の乗算でできる。

（S 1 A）$\delta_n y_0$ の値を求める：それには，以下の各項の値をすべて計算しなければならない。

$$\delta_1 y_{n-1},\ \delta_1 y_{n-2},\ \cdots,\ \delta_1 y_2,\ \delta_1 y_1,\ \delta_1 y_0,$$
$$\delta_2 y_{n-2},\ \cdots,\ \delta_2 y_2,\ \delta_2 y_1,\ \delta_2 y_0,$$
$$\cdots$$
$$\delta_{n-2} y_2,\ \delta_{n-2} y_1,\ \delta_{n-2} y_0,$$
$$\delta_{n-1} y_1,\ \delta_{n-1} y_0,$$
$$\delta_n y_0$$

どの項も，2 回の減算と 1 回の除算で求められるから，全体として減算は
$2\times(n+(n-1)+\cdots+2+1) = n^2+n$（回），
除算はその半分の $(1/2)n^2+(1/2)n$（回）になる。

（S 1 B）（新）$y_t$ =（旧）$y_t - a_n x_t^n$ の計算（$0 \leqq t \leqq n$）

〈注意〉あとで使うのは（新）$y_0, y_1, \cdots, y_{n-1}$ までである。

これは，$x_t^n$ の値はすでに求めてあるので，乗算・加減算ともに $n$ 回ずつでできる。

そのあとは以下の（S 2 A)，(S 2 B）を，$k = n-1, \cdots, 1$ について繰り返す。

（S 2 A）新しい $\{y_t\}$ に基づいて，$\delta_k y_0$ の値を求める：$(1/2)k(k+1)$ 個の項の値を計算するのは（S 1 A）と同様であるが，今度は分母 $(x_{t+k}-x_t)$ の値がすべて計算済みなので，必要な演算は各 $k$ ごとに

乗除算・加減算ともに $(1/2)k(k+1)$ 回ずつ

ですむ。

（S 2 B）（新）$y_t$ =（旧）$y_t - a_k x_t^k$ の計算（$0 \leqq t \leqq k$）

これは，$x_t^k$ の値はあらかじめ求めてあるので，乗算・加減算 $k$ 回ずつでよい。

以上をまとめると，全体としての演算回数は次のようになる。
乗除算：

$$
\begin{aligned}
&(S0)\ (S1A)+(S2A)\ (S1B)+(S2B)\\
&n^2-n+\sum_{k=1}^{n}\frac{1}{2}k(k+1)+\sum_{k=1}^{n}k\\
&=n^2-n+\frac{1}{2}\sum_{k=1}^{n}k^2+\frac{3}{2}\sum_{k=1}^{n}k\\
&=n^2-n+\frac{1}{6}n^3+\frac{1}{4}n^2+\frac{1}{12}n+\frac{3}{4}n^2+\frac{3}{4}n\\
&=\frac{1}{6}n^3+2n^2-\frac{1}{6}n
\end{aligned}
$$

加減算：

$$
\begin{aligned}
&(S1A)\ (S2A)\qquad (S1B)+(S2B)\\
&(n^2-n)+\sum_{k=1}^{n-1}\frac{1}{2}k(k+1)+\sum_{k=1}^{n}k\\
&=\frac{1}{2}n^2-\frac{1}{2}n+\frac{1}{2}\sum_{k=1}^{n}k^2+\frac{3}{2}\sum_{k=1}^{n}k\\
&=\frac{1}{2}n^2-\frac{1}{2}n+\frac{1}{6}n^3+\frac{1}{4}n^2+\frac{1}{12}n+\frac{3}{4}n^2+\frac{3}{4}n\\
&=\frac{1}{6}n^3+\frac{3}{2}n^2+\frac{4}{3}n
\end{aligned}
$$

やはり $n^3$ のオーダーではあるが，ガウスの消去法のおよそ半分の演算回数で済む！

## 3. 関の方法の改良

### A. $\boldsymbol{L}$（$\boldsymbol{t, k, p}$）についての漸化式

方程式（Ⅲ）を，消去法で求めたのでは演算回数は $n^3$ のオーダーになってしまうが，次の漸化式を使うとずっと簡単に求めることができる。
定理 3　$L(t,\ k,\ p)=L(t,\ k,\ p-1)\times x_{t+k}+L(t,k-1,\ p)$

〈証明〉左辺は

$x_t,\ x_{t+1},\ \cdots,\ x_{t+k}$ の $p$ 次の積の総和

であるが，その積は $x_{t+k}$ を含むものと含まないものに分けられる。
$x_{t+k}$ を含むものの和から $x_{t+k}$ をひとつくくりだせば，残りは

$x_t,\ x_{t+1},\ \cdots,\ x_{t+k}$ の，$p-1$ 次の積の総和

になるから $L(t, k, p-1)$ に等しい。したがって，$x_{t+k}$ を含むものの和は
$L(t,\ k,\ p-1)\times x_{t+k}$
に等しい。また $x_{t+k}$ を含まないものの和は

$x_t, x_{t+1}, \cdots, x_{t+k-1}$ の，$p$ 次の積の総和

であるから $L(t,\ k-1,\ p)$ に等しい。〈証明終わり〉

〈例〉$x_0=2,\ x_1=3, x_2=5,\ x_3=11$ に対して，

$$L(0,0,0)=1,$$
$$L(0,\ 0,\ 1)=x_0=2,$$
$$L(0,\ 0,\ 2)=x_0^2=2^2=4,$$
$$L(0,\ 0,\ 3)=x_0^3=2^3=8$$

である。また

$$L(0,1,0)=1,$$
$$L(0,1,1)=x_0+x_1=2+3=5$$

であるが，後者は上の漸化式から次のように求めることもできる：

$$L(0,1,1)=L(0,1,0)\times x_1+L(0,\ 0,\ 1)=1\times 3+2=3+2=5$$

同様に

$$L(0,2,1)=x_0+x_1+x_2=2+3+5=10$$

も，

$$L(0,2,1)=L(2,\ 2,\ 0)\times x_2+L(0,\ 1,\ 1)=1\times 5+5=10$$

のように求められる。

### B. 関の方法（招差之法）の改良版

以下に新しい手順の各段階と，それぞれについての演算回数を示す。

（RS 1）方程式（Ⅲ）を求める：目標は，$n=3$ の場合，次のような方程式の左辺の係数，右辺の定数をすべて決定することである：

$$L(0,0,3)a_3+L(0,0,2)a_2+L(0,0,1)a_1+L(0,0,0)a_0=y_0$$
$$L(0,\ 1,\ 2)a_3+L(0,1,1)a_2+L(0,1,0)a_1=\delta_1 y_0$$
$$L(0,2,1)a_3+L(0,2,0)a_2=\delta_2 y_0$$
$$L(0,3,0)a_3=\delta_3 y_0$$

(RS 1. 1) 左辺の係数の計算：多項式の次数を $n$ とする。

㈠ $L(0,\ k,\ 0)=1$ は，計算する必要がない。

㈡ まず $L(0,\ 0,\ p)=x_0^p\ (2\leqq p\leqq n)$ を計算する：これには $n-1$ 回の乗算が必要である。

㈢ 次に定理 3 から，第 $k$ 行の成分

$$L(0,\ k,\ p-k)=L(0,\ k,\ p-k-1)\times x_{p-k}+L(0,\ k-1,\ p-k)$$

を計算する $(1\leqq k\leqq n, k\leqq p\leqq n)$。これには，“$1\times$” を除いて

$$(n-2)+(n-3)+\cdots+1=(1/2)(n-1)(n-2)=(1/2)n^2-(3/2)n+1$$

回の乗算と，

$$(n-1)+(n-2)+\cdots+1=(1/2)n(n-1)=(1/2)n^2-(1/2)n$$

回の加算が必要である。これで左辺の係数が全て決定できる。

㈡と㈢を合わせると，この段階での演算回数は乗算・加算とも，それぞれ

$$(1/2)n^2-(1/2)n\ (\text{回})$$

ずつである。

(RS 1. 2) 右辺の定数の計算：これは関の方法（累裁招差之法）の（S 1 A）と同じで，すでに述べたように，

乗除算 $(1/2)n^2+(1/2)n$（回）

加減算 $n^2+n$（回），

で求められる。

(RS 2) 方程式（Ⅲ）を解く：これは「ガウスの消去法」の後半（G 2）とまったく同じなので，乗除算・加減算ともに $(1/2)n(n+1)$ 回ずつでよい。

以上，演算回数は合計して次のようになる。

乗除算 $(3/2)n^2+(1/2)n$（回），

加減算 $2n^2+n$（回）

したがって，全演算回数のオーダーは $n^2$ で，「オーダーの改良」ができたわけである。

## 4. 補間式から求める方法

### A. ラグランジュの補間多項式から

めざす多項式 $f(x)$ は，ラグランジュの補間多項式を展開しても得られる（これもよく知られていることであるが，たとえば［4］，p. 18 参照）：

$$f(x)=\sum_{k=0}^{n}\prod_{j\neq k}\left(\frac{x-x_j}{x_k-x_j}\right)\times y_k$$

（L 0）最初に準備として，差 $(x_s - x_t)$ の値をすべて求めておく $(0 \leqq s < t \leqq n)$。

それには $(1/2)n(n+1)$ 回の減算が必要である。

（L 1）右辺の積 $\Pi$ は $n$ 次式であるから，これを展開する手間は，ひとつの $k$ ごとに，次のように見積もられる。

(1.1) 分子の積の計算：まず 2 次式

$$(x-x_1)(x-x_2) = x^2-(x_1+x_2)x+(x_1\times x_2)$$

の係数を決定するには（“1×” は除いて）1 回の乗算と 1 回の加算が必要である。そのあと 3 次式

$$(x^2+ax+b)(x-x_3) = x^3+(a-x_3)x^2+(b-a\times x_3)x-(b\times x_3)$$

の係数を決定するのは，2 回の乗算と 2 回の加減算でできる。以下同様に，$j$ 次式

（最高次の係数が 1 であるような $j-1$ 次式）$\times(x-x_J)$

を求めるのは乗算・加減算ともに $j-1$ 回ずつでできるので，$n$ 次式を求めるには，乗算・加減算ともに

$$1+2+\cdots+(n-1) = (1/2)n(n-1)\text{（回）}$$

で十分である。

(1.2) 定数 $\alpha = \Pi\, 1/(x_k-x_j)\times y_k$ の計算

これは分母がすべて計算済みなので，$n-1$ 回の乗算と 1 回の除算でできる。

(1.3) (1.1) で得られた $n$ 次式を，$\alpha$ 倍する：すべての係数を $\alpha$ 倍しなければならないので，“1×” を除き，$n$ 回の乗算が必要である。

結局(1)は，ひとつの $n$ 次式 $(\Pi)$ ごとに

乗除算 $(1/2)n^2+(3/2)n$ 回，加減算 $(1/2)n^2-(1/2)n$ 回

を要するわけで，$(n+1)$ 個あるすべての多項式 $(\Pi)$ のすべての係数を求めるのは

その $(n+1)$ 倍の，次の回数を必要とする：

乗除算　$(1/2)n^3+2n^2+(3/2)n$ 回，

加減算　$(1/2)n^3-(1/2)n$ 回

(L 2) $\Sigma(\Pi)$ の計算：$(n+1)$ 個の $n$ 次多項式の総和（の，$n+1$ 個の係数）を求めるのであるから，加減算が $n(n+1)$ 回必要である。

以上を合計すると，準備も含めて

乗除算　$(1/2)n^3+2n^2+(3/2)n$ 回，

加減算　$(1/2)n^3+(3/2)n^2+n$ 回

を要する，ということになる。これは連立 1 次方程式をガウスの消去法で解いた時の演算回数 $(1/3)n^3$ より，さらに大きい。

### B．エイトケンの補間法から

エイトケンの補間法は，次のように説明できる（[ 4 ] pp. 19-20，ただし記号は変更した）。0 次式（定数関数）

$$y_t^{(0)}(x) = y_t \text{（定数関数）}$$

から始めて，$k$ 次式 $y_t^{(k)}(x)$ を次のように定義する。
定義 3

$$y_t^{(k)}(x) = \frac{(x_{t+k}-x)y_t^{(k-1)}(x)-(x_t-x)y_{t+1}^{(k-1)}(x)}{x_{t+k}-x_t} \cdots (\sharp)$$

すると $y_0^{(n)}(x)$ は，$0 \leqq t \leqq n$ に対して

$$y_t = y_0^{(n)}(x_t)$$

をみたす多項式になる。これを求めるには

$$y_0^{(1)}(x),\ y_1^{(1)}(x),\ \cdots,\ y_{n-2}^{(1)}(x),\ y_{n-1}^{(1)}(x),$$
$$y_0^{(2)}(x),\ y_1^{(2)}(x),\ \cdots,\ y_{n-2}^{(2)}(x)$$
$$\cdots$$
$$y_0^{(n-1)}(x),\ y_1^{(n-1)}(x),$$
$$y_0^{(n)}(x)$$

をすべて求めなければならないが，数値 $\delta_k y_t$ と違って「式の計算」をしなければならないので，ずっと煩雑になる。実際，定義の漸化式（＃）を 1 回計算するのに，$y_t^{(k)}(x)$ の最高次の係数が 1 でないため，

乗除算　$3k+1$ 回，加減算　$3k$ 回

を要する。そのため全体としては次のような手間がかかる。

乗除算：$\sum_{k=1}^{n}(3k+1)(n-k+1) = \sum_{k=1}^{n}(3(k-1)+4)(n-(k-1))$

$$= 3n\sum_{k=1}^{n}(k-1)+4n\sum_{k=1}^{n}1-3\sum_{k=1}^{n}(k-1)^2-4\sum_{k=1}^{n}(k-1)$$

$$= \frac{3}{2}n^3-\frac{3}{2}n^2+4n^2-3(\frac{1}{3}n^3-\frac{1}{2}n^2+\frac{1}{6}n)-2n^2+2n$$

$$= \frac{1}{2}n^3+2n^2+\frac{3}{2}n$$

加減算： $\sum_{k=1}^{n} 3k(n-k+1) = 3n\sum_{k=1}^{n} k - 3\sum_{k=1}^{n} k^2 + 3\sum_{k=1}^{n} k$

$$= \frac{3}{2}n^3+\frac{3}{2}n^2-3\left(\frac{1}{3}n^3-\frac{1}{2}n^2+\frac{1}{6}n\right)-\frac{3}{2}n^2+\frac{3}{2}n$$

$$= \frac{1}{2}n^3+\frac{3}{2}n^2+n$$

これはラグランジュの補間式から求めた場合と，まったく変わりない。

### C. ニュートンの補間式から

ニュートンの補間式は，次のように表わされる（［4］20ページ公式（2.10），ただし記法を変更した）。

$$f(x) = y_0+(x-x_0)\delta_1 y_0+(x-x_0)(x-x_1)\delta_2 y_0+(x-x_0)(x-x_1)(x-x_2)\delta_3 y_0+ \cdots+(x-x_0)(x-x_1)\cdots(x-x_{n-1})\delta_n y_0$$

これによって関数値を数値的に求めるには，パラメータ $\delta_k y_0$ の値（$x$ には依存しない）は計算してあるとして，次のようにするとよい。

$$f(x) = (((\cdots((\delta_n y_0\times(x-x_{n-1})+\delta_{n-1}y_0)\times(x-x_{n-2})+\delta_{n-2}y_0)\times\cdots \cdots)\times(x-x_2)+\delta_2 y_0)\times(x-x_1)+\delta_1 y_0)\times(x-x_0)+y_0 \quad\cdots\ (\sharp\sharp)$$

そうすれば $n$ 回の乗算と $2n$ 回の加算で，関数値の評価ができる。

関数 $f(x)$ を式として展開するにも，この計算式（♯♯）が役に立つ。

（N 1）まず $\delta_k y_0$ を計算しておく…これは「招差之法」（S 1 A）と同じで，

乗除算　$(1/2)n^2+(1/2)n$ 回，

加減算　$n^2+n$ 回

でできる（(S 0）は必要ない)。

（N 2）次に1次式

$$\delta_n y_0\times(x-x_{n-1})+\delta_{n-1}y_0 = \delta_n y_0\cdot x-\delta_n y_0\times x_{n-1}+\delta_{n-1}y_0$$

から始めて，多項式の次数を1つずつ上げてゆく。

$k$ 次式$\times(x-x_{n-k-1})+\delta_{n-k-1}y_0$

の計算には，係数の計算としては“×1”を除いて，乗算“$x_{n-k-1}$”が $k$ 回と，（さいごの $+\delta_{n-k-1}y_0$ を含めて）加減算 $k$ 回が必要なので，全体として $(1/2)n(n+1)$ 回の乗算と加減算でできる。したがって，全体としてはおよそ

乗除算：　$n^2$ 回

加減算：　$(3/2)n^2+(1/2)n$ 回

の計算ですむ。この回数は「係数の値をすべて求める」手段としては，最も少ない！

## D. まとめ

以上の計算量の評価をまとめると，次のようになる。

**表　$n$ 次多項式のすべての係数を決定するために必要な演算回数**

| | 乗除算 | 加減算 |
|---|---|---|
| ①　ガウスの消去法 | $(1/3)n^3+(1/2)n^2$ | $(1/3)n^3+(1/2)n^2$ |
| ②　関の方法（累裁招差之法） | $(1/6)n^3+2n^2$ | $(1/6)n^3+(3/2)n^2$ |
| ③　関の方法の改良版 | $(3/2)n^2$ | $2n^2$ |
| ④　ラグランジュの補間多項式 | $(1/2)n^3+2n^2$ | $(1/2)n^3+(3/2)n^2$ |
| ⑤　エイトケンの補間法 | $(1/2)n^3+2n^2$ | $(1/2)n^3+(3/2)n^2$ |
| ⑥　ニュートンの補間公式 | $n^2$ | $(3/2)n^2$ |

〈補足〉　変数値が等間隔である場合には，どの方法でもかなり計算量を減らせる。特に②，③，⑥は，$\delta_k y_t$ の計算の主要部分を「加減算だけの差分計算」におきかえられるが，オーダーは変わらないし「共通に減る」ので，深入りはしないでおく。

**参考文献**


1. 野崎昭弘「ベキ和の公式のいくつかの導き方」，日本数学協会『数学文化』第 11 号 2008 年，pp. 64-83
2. 野崎昭弘『アルゴリズムと計算量』共立出版 1987 年，pp. 39-42
3. 平山諦・下平和夫・広瀬秀雄編・著『関孝和全集』大阪教育図書 1974 年，pp. 273-282
4. 戸川隼人『数値計算法』コロナ社 1981 年，pp. 17-21

# Comparison of the Methods for Evaluating the Coefficients of a Polynomial with Respect to Computational Complexity

Akihiro Nozaki

The problem of evaluating all coefficients of an n-th degree polynomial function by $(n+1)$ pairs of arguments and function values can be solved by various methods. A straight-forward approach may be to solve simultaneous linear equations, but T. Seki, a Japanese mathematician in the 17th century, proposed an interesting algorithm for this problem. Besides, techniques for interpolation are also applicable to this problem. In this paper, we investigate in detail Gaussian elimination for solving simultaneous linear equations related to the evaluation of coefficients of a polynomial, compare it with Seki method, and propose a revised method, which is much better than Gaussian elimination with respect to computational complexity. After that, we compare these methods with the algorithms for solving the same problem based on interpolation techniques, such as Lagrange's interpolation polynomial, Aitken's interpolation scheme, and Newton's interpolation formula, with respect to computational complexity. Our comparison shows that the algorithm based on Newton's interpolation formula and the revised Seki method are of order $n^2$, while all other algorithms are of order $n^3$.

**Keywords:** Evaluation of Coefficients of a Polynomial, Computational Complexity, Seki method, Newton's Interpolation Formula