

## 第17章 Richardsonの補外

数値計算のどの分野にも増して、数列の収束加速法は実験的な科学である。この研究は、加速法のアルゴリズムを実際に適用し、それが有用であるかどうかを結果から見極めることになるからだ。

D. Laurie, “The SIAM 100-Digit Challenge”, SIAM

今まで見てきた数値微分・数値積分(第15章)、常微分方程式の初期値問題(第16章)に対する解法は全て次数固定、つまり刻み幅  $h$  が決まれば収束の速さは  $O(h^\gamma)$  であり、この  $\gamma$  が定数であるものばかりであった。これに対し、 $\gamma$  を可変にし、任意に設定できる解法を任意次数アルゴリズムと呼ぶ。本章では任意次数アルゴリズムのうち、最も古典的な Richardson の補外を一般的に導出し、前述した3つの問題に適用した事例を提示する。

### 17.1 Nevilleのアルゴリズムによる導出

$s_c \in \mathbb{R}$  へ収束する数列  $s_1, s_2, \dots, s_n, \dots$  ( $s_i \in \mathbb{R}$ ) の一般項  $s_i$  が、ある多項式関数  $S(x)$  を用いて

$$s_i = S(x_i) = s_c + c_1 x_i + c_2 x_i^2 + \dots + c_m x_i^m + \dots \quad (17.1)$$

と表現できるものとする。本章では  $x_i \rightarrow 0$  の時、この数列  $\{s_i\}$  の収束  $s_n \rightarrow s_c$  のスピードを速めるアルゴリズムの一つである Richardson の補外法 (extrapolation) について議論する。

(17.1) 式より、収束値は  $s_c = S(0)$  と表現できるため、 $S(x)$  を多項式補間によって構成できれば、補間多項式によってより  $s_c$  に近い値を得ることが期待できる。これを、Neville のアルゴリズム (アルゴリズム 34) を使って実際に計算してみよう。

今、無限大に発散する補助数列  $w_1, w_2, \dots, w_n, \dots$  ( $w_i > 0$ ) を用いて  $x_i = h/w_i$  とし、 $\{x_i\}$  を 0 に収束するように設定する。補間点として  $(x_1, S(x_1)), (x_2, S(x_2)), \dots, (x_{m+1}, S(x_{m+1}))$  を取り、 $x = 0$  における補間多項式の値を Neville のアルゴリズムによって求めることを考える。

初期系列を  $T_{11} := S(x_1), T_{21} := S(x_2), \dots, T_{m+1,1} := S(x_{m+1})$  とし、 $x = 0$  における  $m$  次補間多項式の値  $p_m(0)$  を出すための漸化式 (14.6) は

$$\begin{aligned} T_{ij} &:= \frac{(0 - x_{i-j+1})T_{i,j-1} - (0 - x_i)T_{i-1,j-1}}{x_i - x_{i-j+1}} \\ &= \frac{x_i T_{i-1,j-1} - x_{i-j+1} T_{i,j-1}}{x_i - x_{i-j+1}} \end{aligned} \quad (17.2)$$

$$\begin{aligned} &= T_{i,j-1} + \frac{T_{i,j-1} - T_{i-1,j-1}}{\frac{x_{i-j+1}}{x_i} - 1} \\ &= T_{i,j-1} + \frac{T_{i,j-1} - T_{i-1,j-1}}{\frac{w_i}{w_{i-j+1}} - 1} \end{aligned} \quad (17.3)$$

と変形できる。実際に計算する際には、(17.2) 式より (17.3) 式の方が有限桁計算においては良いとされている。

表 17.1: Richardson の補外表

初期系列				
$T_{11}$				
	$\searrow$			
$T_{21}$	$\rightarrow$	$T_{22}$		
$\vdots$		$\vdots$	$\ddots$	
$T_{m1}$	$\rightarrow$	$T_{m2}$	$\cdots$	$T_{mm}$
	$\searrow$			
$T_{m+1,1}$	$\rightarrow$	$T_{m+1,2}$	$\cdots$	$T_{m+1,m}$
				$\searrow$
				$T_{m+1,m+1}$

このように、 $T_{m+1,m+1}$  を求める操作は、補間点  $x_1, x_2, \dots, x_{m+1}$  ( $x_i > 0$ ) の外部  $x = 0$  における値を求めていることになるため、**補外 (extrapolation)** と呼ばれている。

こうして得られた  $T_{m+1,m+1} = p_m(0)$  は、もし  $0$  と補間点  $x_1, x_2, \dots, x_{m+1}$  を含む区間  $I$  において  $S(x)$  が  $m+1$  回連続微分可能である時、定理 14.2.1 によって、

$$|T_{m+1,m+1} - S(0)| = (-1)^{m+1} \frac{S^{(m+1)}(\xi)}{(m+1)!} \prod_{i=1}^{m+1} x_i \quad (17.4)$$

を満足する  $\xi \in I$  が存在することになる。従って、

$$|T_{m+1,m+1} - S(0)| = O\left(\prod_{i=1}^{m+1} x_i\right) = O(|h|^{m+1}) \quad (17.5)$$

となり、 $O(|h|)$  で収束する  $\{s_i\}$  よりも高速に、 $m+1$  次の速さで収束することが期待できる。

もし  $S(x)$  が (17.1) ではなく、定数  $\alpha \geq 1$  に対して

$$s_i = S(x_i) = s_c + c_1 x_i^\alpha + c_2 x_i^{2\alpha} + \cdots + c_m x_i^{m\alpha} + \cdots \quad (17.6)$$

という形で表現できるのであれば、(17.3) は

$$T_{ij} := T_{i,j-1} + \frac{T_{i,j-1} - T_{i-1,j-1}}{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha - 1} \quad (17.7)$$

となる ( $\rightarrow$  演習問題 1)。この時、収束の速さは

$$|T_{m+1,m+1} - S(0)| = O(|h|^{(m+1)\alpha}) \quad (17.8)$$

であることが期待される。

実際に補外計算を行う際には最大段数 (行数)  $L \geq 2$  を決めておき、まず  $T_{11} = S(h/w_1)$  と  $T_{21} = S(h/w_2)$  を求め、

$$\begin{array}{ccc}
 T_{11} & & \\
 & \searrow & \\
 T_{21} & \rightarrow & T_{22}
 \end{array}$$

の  $T_{22}$  の計算において収束判定を行う。もし収束していなければ  $T_{31} = S(h/w_3)$  を求めて

$$\begin{array}{ccccc}
 T_{11} & & & & \\
 & & & & \\
 T_{21} & & T_{22} & & \\
 & \searrow & & \searrow & \\
 T_{31} & \rightarrow & T_{32} & \rightarrow & T_{33}
 \end{array}$$

のように、 $T_{32}, T_{33}$  の計算を収束判定を行いつつ実施する。このように、なるべく計算時間が最も必要な(場合の多い)初期系列の段数を少なく抑えることで、補外計算全体の効率化を図ることが出来る。

以上をまとめると、(17.6)式を前提とした Richardson の補外のアルゴリズムは以下ようになる。

#### アルゴリズム 40 (Richardson の補外)

1. 補助数列  $\{w_i\}$ , 最大段数  $L$ , 相対許容度  $\varepsilon_R$ , 絶対許容度  $\varepsilon_A$  を与える。
2.  $T_{11} := S(h/w_1)$  とする。
3.  $i = 2, \dots, L$

(a)  $T_{i1} := S(h/w_i)$

- (b)  $j = 2, \dots, i$  において以下の計算を行う。

- i. (17.7) を次のように分割して計算する。

$$R_{ij} := \frac{T_{i,j-1} - T_{i-1,j-1}}{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha - 1}$$

$$T_{ij} := T_{i,j-1} + R_{ij}$$

- ii. 収束判定:

$$|R_{ij}| \leq \varepsilon_R |T_{i,j-1}| + \varepsilon_A \quad (17.9)$$

ならば収束したと判断する。

補助数列  $\{w_i\}$  としては、次の3つが提案されている。

**Romberg 数列:**  $1, 2, 4, 8, \dots, 2^i, \dots$

**Bulirsch 数列:**  $1, 2, 3, 4, 6, 8, 12, \dots, 3 \cdot 2^{i/2-1}, 2^{(i-1)/2}, \dots$

**調和 (harmonic) 数列:**  $1, 2, 3, 4, \dots, i+1, \dots$

このうち、調和数列は無限数列を収束させる理論的な根拠となっている Toeplitz 条件

$$\lim_{n \rightarrow \infty} \frac{w_n}{w_{n+1}} < 1$$

を満足しておらず、収束しない場合もありえる。しかし、初期系列の計算時間が  $w_i$  に比例して増加する数値積分や常微分方程式の初期値問題の場合は、他の二つの数列に比べて効率が良いとされている。逆に Romberg 数列は収束条件は良いものの、初期系列の計算時間が大きくなる。この両者の折衷が Bulirsch 数列ということになる。

## 17.2 丸め誤差伝播の性質

収束する数列  $\{s_i\}$  に対する Richardson の補外の初期系列  $T_{11}, T_{21}, \dots$  は先に進むにつれて同じ値に近づいていく。この性質を用いて、初期系列に含まれる丸め誤差  $\varepsilon_{11}, \varepsilon_{21}, \dots$  が補外計算によってどのように伝播し、補助数列の違いによってどのように異なるのかを、Hairer & Wanner[4] が提案する簡易な方法で調べてみる。

初期系列  $T_{i1}$  に含まれる丸め誤差を  $\varepsilon_{i1} = (-1)^{i-1}\varepsilon$  とし、隣接する初期系列同士の差によって打ち消されないものとする。通常は  $\varepsilon \approx \varepsilon_M$  と考えてよい。この時、補外計算 (17.7) によって、 $T_{i2}$  に含まれる丸め誤差  $\varepsilon_{i2}$  は

$$\begin{aligned} \varepsilon_{i2} &= \varepsilon_{i1} + \frac{\varepsilon_{i1} - \varepsilon_{i-1,1}}{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha - 1} \\ &= \left\{ (-1)^{i-1} + \frac{(-1)^{i-1} - (-1)^{i-2}}{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha - 1} \right\} \varepsilon \\ &= \frac{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha + 1}{\left(\frac{w_i}{w_{i-j+1}}\right)^\alpha - 1} \cdot (-1)^{i-1} \varepsilon \\ &= r_{ij} \varepsilon \end{aligned}$$

となる。これを続けて得られる一般の  $\varepsilon_{ij} = r_{ij}\varepsilon$  に対応する  $r_{ij}$  は、初期系列の丸め誤差の拡大率を意味することになる。

$\alpha = 2$  の時の各補助数列に対応する  $|r_{ij}|$  を実際に計算し、プロットしたものが図 17.1 である。

Romberg 数列、Bulirsch 数列では 2 倍以内に抑えられているのに対し、調和数列の場合は 6~7 桁程度まで悪化し、そこで頭打ちになることが分かる。よって、もし調和数列を使用するのであれば、初期系列の打ち切り誤差が丸め誤差に比べて大きい所で止めることが望ましいということになる。

## 17.3 数値微分への適用

Richardson の補外を数値微分に適用してみよう。

無限回微分可能な一変数関数  $f(x)$  の  $m$  階微係数  $f^{(m)}(x)$  を  $N$  点差分商で近似すると、(15.4), (15.5), (15.6) から分かる通り、近似式は一般に

$$f^{(m)}(x) = \frac{1}{h^m} \sum_{s=1}^N b_s f(x_s) + E_T(h) \quad (17.10)$$

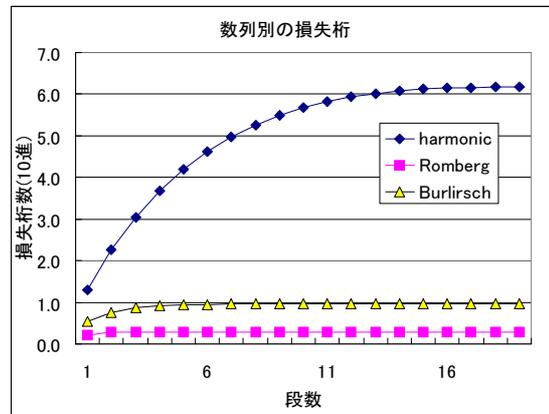


図 17.1: 補助数列ごとの丸め誤差の拡大率 ( $\log_{10} |r_{20,i}|$  のみプロット)

と表現できる。ここで  $b_s$  は定数,  $E_T(h)$  は打ち切り誤差である。補外法のためには初期系列の計算はなるべく軽いものが望ましいので, 中心差分公式を用いて近似する。特に 1 階微係数  $f'(x)$  を 3 点中心差分商を用いて近似すると,  $h_i = h/w_{i-1}$  として

$$T_{i1} = \frac{1}{h_i} \left\{ \frac{1}{2} f(x+h_i) - \frac{1}{2} f(x-h_i) \right\} + E_T(h_i) \approx f'(x)$$

となる。

この時  $E_T(h_i)$  は

$$E_T(h_i) = \sum_{w=0}^{\infty} \frac{f^{(2w+3)}(x)}{(2w+3)!} h_i^{2w+2}$$

と  $h_i$  の 2 のべき乗和として表現できる (→ 演習問題 2)。従って, Richardson 補外を使用すると段数  $\times 2$  に比例して打ち切り誤差が減少していくことが期待される。

数値微分の場合, 初期系列計算に使用する補助数列としては Romberg 数列を使用するのが良い。これは次に理由による。

- 後述する数値積分や常微分方程式の初期値問題とは異なり, 初期系列  $T_{i1}$  の計算における  $f(x)$  の呼び出し回数は近似差分公式にのみ依存し, どの数列を用いても変わらない。
- Romberg 数列は打ち切り誤差 (17.4) ・丸め誤差の増大 (図 17.1) を最も抑えることの出来るものである。

よって, 本書では補外による数値微分には Romberg 数列  $w_i = 2^{i-1}$  のみを用いることにする。

### 17.3.1 数値微分における丸め誤差限界とアルゴリズム

永坂・福井は (17.10) を  $p$  進  $q$  桁の浮動小数点数を用いて数値計算する時に発生する丸め誤差の限界を  $E_R(h)$  を次のように評価した。総和の計算,  $h^m$  の計算, 及びそれらを乗じる際に発生する

丸め誤差を総合して

$$E_R(h) = \frac{N-1}{h^m} \max_s |b_s f(x_s)| \cdot c \cdot p^{-q} \quad (17.11)$$

と評価できる [27]。ここで  $c$  は丸めの方式によって決まる定数で、

$$c = \begin{cases} p & (\text{切り捨て}) \\ \frac{p}{2} & (p/2 - 1 \text{ 捨 } p/2 \text{ 入}) \end{cases}$$

である。

中心差分商を用いて求めた近似値 (17.10) を、Romberg 数列を用いた補外法の初期系列  $T_{i1}$  として使用すると。この時

$$T_{i1} = \frac{\sum_{s=1}^N b_s f(x_s)}{h_i^m}$$

である。この時、補外計算 (17.7) は

$$\begin{array}{ccc} T_{i-1,j-1} & \searrow & \\ & & T_{ij} = T_{i,j-1} + \frac{T_{i,j-1} - T_{i-1,j-1}}{4^{j-1} - 1} \end{array} \quad (17.12)$$

となる。この計算をアルゴリズム 40 に従って計算していくことになる。

但し、収束判定については次の福井の結果を利用して少し改善できる。

この補外計算で混入する丸め誤差限界  $E_R^{ij}$  は

$$E_R^{ij} = \frac{1 + 2^{-m}}{4^{j-1} - 1} \cdot \frac{25}{14} E_R(h/2^{i-1})$$

であることが福井によって証明されている [6]。これは  $p$  進  $q$  桁計算における近似値の絶対誤差の限界値に近い値であるため、実際の数値計算ではこの値になる以前に収束したと判定することが望ましい。よって、収束判定は、ユーザが与えた相対許容度  $\varepsilon_R \geq 0$  と絶対許容度  $\varepsilon_A \geq 0$  も用い、補外計算 (17.23) において (17.9) 式を

$$|R_{ij}| \leq \max(\varepsilon_R |T_{i,j-1}| + \varepsilon_A, E_R^{ij}) \quad (17.13)$$

と改善する。与えられた  $\varepsilon_R$  や  $\varepsilon_A$  が小さすぎる時には、自動的に求められた丸め誤差限界値  $E_R^{ij}$  が歯止めになる仕組みである。

### 17.3.2 数値例

ここで IEEE754 倍精度計算を用いて計算した補外による数値微分の例を二つ提示する。

#### 例題 17.3.1

まず、補外を用いた数値微分の数値例として、15.2 節でも用いた

$$f(x) = \cos(\sin x)$$

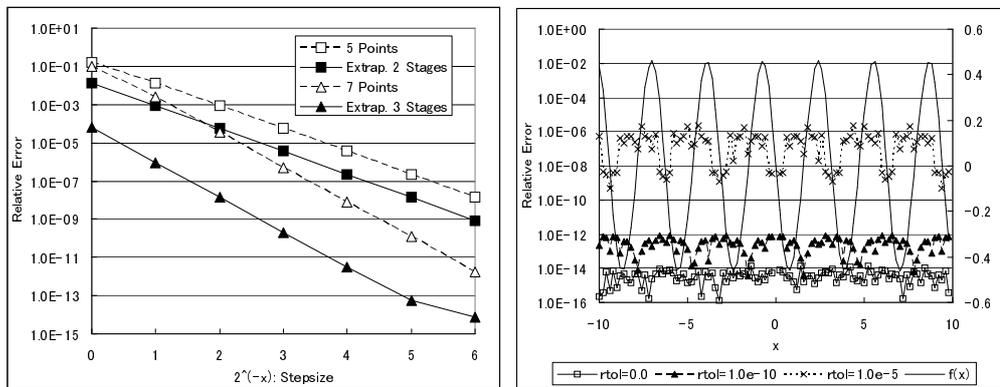


図 17.2: (左) $x = \pi/4$ における次数の比較: 5点公式 vs. 2段補外(4次), 7点公式 vs. 3段補外(6次), (右) 区間  $[-10, 10]$  における誤差の推移

を使用した計算例を示す。図 17.2 の左に  $x = \pi/4$  における導関数の値を  $h = 2^0, 2^{-1}, \dots, 2^{-8}$  と小さくしつつ、5点、7点公式と、次数になる段数  $L = 2, 3$  で止めた補外法によって求めたもので比較する。5点公式と2段補外、7点公式と3段補外の相対誤差が平行して小さくなっていることが分かるとともに、打ち切り誤差そのものの大きさも補外によって小さく抑えられていることが分かる。

また、 $x \in [-10, 10]$  内の各点における数値微分の相対誤差を、相対許容度  $\epsilon_R = rtol = 10^{-5}, 10^{-10}, 0$  としてそれぞれ求めてプロットしたものを図 17.2 の右に示す。参考までに  $f(x)$  もプロットしてある。

これによって、全体的に相対許容度に応じて相対誤差が評価されていることが分かる。また、数値計算の常道では  $\epsilon_R = 0$  とすることは避けるべきとされているが、丸め誤差評価値の採用によってこの場合もほぼ IEEE754 倍精度の  $\epsilon_M$  に近い相対誤差の近似値が得られていることも分かる。

**例題 17.3.2**

$x = 1$  において特異点を持つ不連続な関数

$$f(x) = \frac{1}{1-x}$$

の導関数  $f'(x)$  の値を区間  $[0, 0.999]$  において計算し、その相対誤差をプロットしたものを図 17.3 に示す。左が IEEE754 倍精度計算のものだけをプロットし、右には 10進50桁計算 ( $\epsilon_R = 0$ ) した際の相対誤差をプロットしたものも含めてある。

特異点に近づくにつれ、相対誤差が急激に上がっていることが分かる。特に  $\epsilon_R = 10^{-5}, 10^{-10}$  とすると、相対誤差(主に打ち切り誤差)が  $\epsilon_R$  近辺まで大きくなればそれを抑えるように働いていることが分かる。しかしそれでも特異点に近づくにつれて相対誤差が増大し抑えきれなくなっている。特異点近辺においては相対許容度は小さめに設定し、なおかつ計算精度もなるべく大きく取ることが望ましいことが分かる。

最後に、二つの例題において、収束に要した段数をプロットしたものを図 17.4 に示す。前述したように、 $\epsilon_R = 10^{-5}, 10^{-15}, 0$  としてそれぞれ計算したものである。

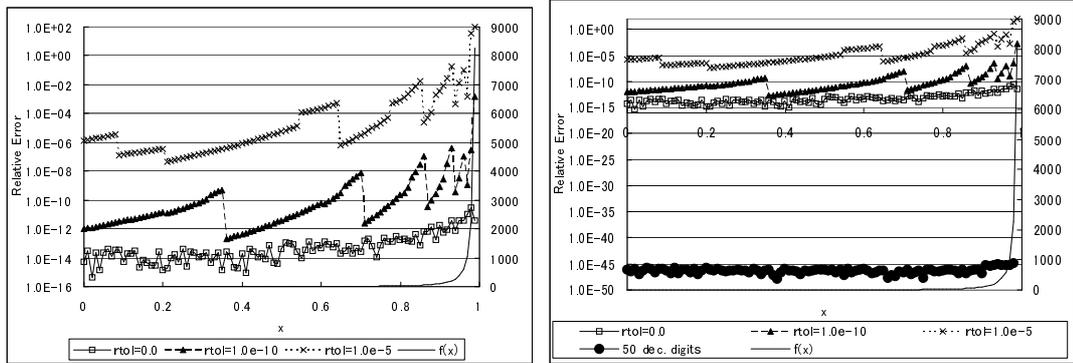


図 17.3: (左)IEEE754 倍精度計算, (右)10 進 50 桁計算 (最下のプロット点)

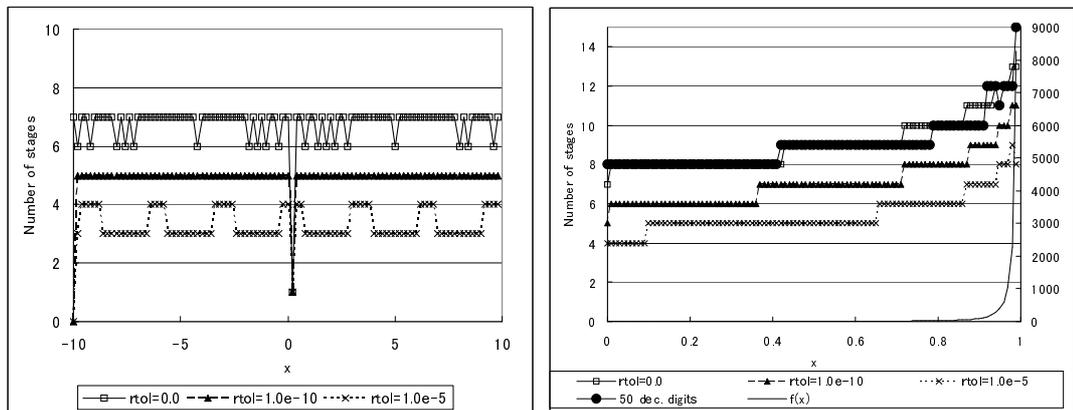


図 17.4: 段数の変化: (左) 例題 1, (右) 例題 2

段数と相対誤差がほぼ連動して推移し、特異点を持つ問題ではその近辺において高い次数(段数)が必要になることが分かる。

更に重要なことは、特異点を持つような問題については十分計算桁数も確保しないと、本来必要な段数以上に補外計算を行ってしまう可能性がある、ということである。ことに図 17.4 の左図において、IEEE754 倍精度の  $\varepsilon_R = 0$  とした時の段数が、10 進 50 桁計算の段数より多い所が特異点に近くなるにつれて見受けられる。補外法を用いて高精度な数値解を得るためには、問題そのものの解析的性質をよく知ることが必要であることがこれによってよく分かる。

### 問題 17.3.1

補助数列として Romberg 数列を用い、打ち切り誤差が 2 のべき乗の多項式で表現できると、補外式 (17.7) は (17.12) 式となることを示せ。(Hint:  $w_i = 2^{i-1}$ ,  $\alpha = 2$  であることを利用せよ。)

## 17.4 数値積分への適用 — Romberg 型数値積分

本節では定積分 (15.7)

$$\int_a^b f(x)dx$$

を、等間隔  $h_i = h/w_{i-1}$  で積分区間を分割し、台形公式 (15.11) を使って求めた積分の近似値を初期系列  $T_{i1}$  とする補外法を考える。この時、 $T_{i1}$  の打ち切り誤差を  $T_{i1} = e(h_i)$  とする。即ち

$$T_{i1} - \int_a^b f(x)dx = e(h_i) \quad (17.14)$$

という関係になる。もし誤差は添字に関係なく

$$e(h) = e_2h^2 + e_4h^4 + e_6h^6 + \dots$$

と  $h$  の 2 のべき乗の多項式で一意に表されるとすると、区間の分割数を  $w_{i-1}$  倍にして求めた初期系列に含まれる誤差も 2 のべき乗の多項式として表現できることになる。これにより、補外計算を行うと

$$\left| T_{ij} - \int_a^b f(x)dx \right| = O(h^{2i})$$

となることが期待できる。

台形公式の誤差項は漸近展開の形で次のように表現できることが知られている。

### 定理 17.4.1 (Euler-Maclaurin 展開)

$f(x)$  が  $[a, b]$  において  $2m+2$  回微分可能であれば、台形公式による定積分 (15.7) の近似によって

$$\begin{aligned} & \frac{h}{2} \left( f(x_0) + 2 \sum_{i=1}^{n-1} f(x_i) + f(x_n) \right) \\ &= \int_a^b f(x)dx + \sum_{l=1}^m h^{2l} \frac{B_{2l}}{(2l)!} (f^{(2l-1)}(x_n) - f^{(2l-1)}(x_0)) \\ & \quad + \frac{(b-a)h^{2m+2} B_{2m+2}}{(2m+2)!} f^{(2m+2)}(\xi) \end{aligned} \quad (17.15)$$

という等式を成立させる  $\xi$  が区間  $(a, b) = (x_0, x_n)$  内に存在する。ここで  $B_i$  は Bernoulli 数 (6.15) である。証明は [34, 22] 等を参照のこと。

以上の定理により、初期系列に台形公式を用いて補外を行うと、 $T_{ij}$  に含まれる打ち切り誤差は  $O(h^{2i})$  で小さくなっていくことが期待できる。

このように、台形則を初期系列に用い、Richardson の補外によって次数を 2 次ずつ上げていく数値積分法を Romberg 型積分と呼ぶことにする。特に補助数列として Romberg 数列を用いるものを Romberg 積分と称している。補助数列の Romberg 数列とは、この数値積分に Romberg が初めて導入したことにその名前の由来がある。

実際に定積分を Romberg 型数値積分で求めた例を以下に示す。定積分としては (1.3) で使用した

$$\int_0^{4/5} \frac{5\sqrt{1-(4/5)^2x^2}}{\sqrt{1-x^2}} dx \quad (17.16)$$

を対象とする。数値的性質を観察するため、あえて収束判定せず、最大段数  $L$  まで無理やり計算させてみる。

補助数列として Romberg 数列と調和数列を用い、積分区間  $[0, 4/5]$  を初期区間幅  $h = 4/5$  とし、IEEE754 倍精度計算で相対誤差をプロットしたものを図 17.5(左) に示す。横軸は段数  $L$  である。前述の数値微分の例と同じく、Romberg 数列を用いたものが最も早く収束していることが分かる。グラフから、実際に収束判定を行って補外計算を行う際には、この例の場合、Romberg 数列を用いると 8 段、調和数列を用いると 12 段程度で止まると、最も精度の良い数値解を得ることができ、それ以上に段数を上げて丸め誤差が増大するだけである。

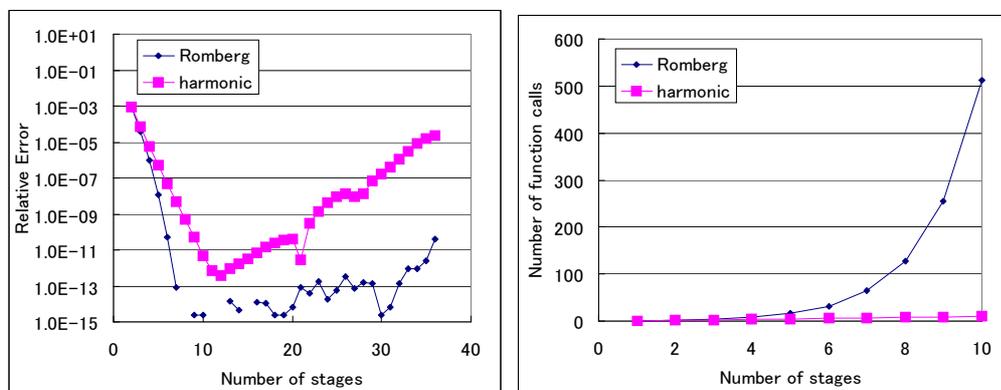


図 17.5: Romberg 型数値積分の相対誤差 (左) と初期系列における関数呼び出し回数 (右)

また、丸め誤差に関する性質も前述の解析結果通り、圧倒的に調和数列を使うと Romberg 数列に比べて精度が悪いことも確認できる。

ではやはり Romberg 数列を使うべきかという、単純にそうとも言えない。そのために、計算量の考察を行う。

Romberg 型数値積分は初期系列  $T_{i1}$  に台形公式を用いるため、関数  $f(x)$  の呼び出し回数は  $w_{i-1} + 1$  回となる。図 17.5(右) は二つの補助数列を用いた場合の呼び出し回数をプロットしたものである。

段数がそれほど違わなければ、圧倒的に調和数列の方が計算量の点で有利である。但し、積分区間が大きいと調和数列では収束しない場合もあり、全ての問題に対して Romberg 数列より短い計算時間で済むかどうかは判然としないが、誤差を  $\varepsilon_M$  より大きく取ってよいならば、一般的には調和数列が有利な場合が多い。

従って、実際に計算を行う場合は

- 積分区間が大きく、相対誤差を  $\varepsilon_M$  に近いレベルにまで落としたいならば Romberg 数列
- 積分区間が比較的小さく、場合によっては積分区間の再分割が可能で、相対誤差が  $\varepsilon_M$  より過大でも問題ない場合は調和数列

を使用し、適切に収束判定を行って丸め誤差領域の手前で段数を止めるようにすることが望ましい。

#### 問題 17.4.1

定積分

$$\int_1^2 \frac{1}{x^2} dx = \frac{1}{2}$$

を Romberg 型数値積分で計算するとき、次の問いに答えよ。

1. この区間を 1, 2, 4 分割してそれぞれ台形則で近似値を IEEE754 倍精度で求めると

6.25000000000000000000e-01

5.347222222222222210e-01

5.08993764172335661e-01

となった。この値を出発値として Romberg 積分を実行せよ。

2. 調和数列を用いて 3 段で数値積分を求めよ。この時、Romberg 数列を用いた結果と比較して相対誤差はどのように減少していくかも合わせて調べよ。

## 17.5 常微分方程式の初期値問題への適用

簡単のため、以下の説明では一次元の常微分方程式の初期値問題

$$\begin{cases} \frac{dy}{dx} = f(x, y) \\ y(x_0) = y_0 \end{cases} \quad (17.17)$$

を対象とするが、多次元問題への対応は容易い。この初期値問題に対し、補外法を用いて求める方法を GBS(Gragg-Bulirsch-Stoer) アルゴリズムと呼ぶ。

数値微分の初期系列に中心差分商を用い、数値積分の初期系列に台形公式を用いた理由は、どちらも打ち切り誤差が刻み幅  $h$  の偶数べき乗展開の形で表現できるため、補外すると 1 段につき 2 次ずつ次数が上昇するからである。常微分方程式でも同様の効果を狙って、偶数べき乗展開の形で打ち切り誤差が表現できる初期系列を選択したい。それが Gragg による midpoint 公式を利用した補外法であり、後に Bulirsch や Stoer らが調和数列を導入して計算量の最適化を図った GBS アルゴリズムである。

### 17.5.1 随伴法と対称法

等間隔  $h$  の刻み幅で、離散点  $x_0, x_1 = x_0 + h, \dots, x_i = x_0 + ih \dots$  における  $y(x_i)$  の近似値  $y_i \approx y(x_i)$  を計算するものとする、第 16 章において導入した一段法は次のような形式で表現できる。

$$y_{i+1} := y_i + h\psi(x_i, y_i, h) \quad (17.18)$$

例えば  $\psi(x_i, y_i, h) = f(x_i, y_i)$  の時は (陽的)Euler 法になる。この意味で、中点法は一段法ではない。(17.18) 式の一段法に対する随伴 (adjoint) 法という概念を導入する。これは次のようにして生成する。

まず、 $h := -h, x_{i+1} := x_i, y_{i+1} := y_i, x_i := x_{i+1}, y_i := y_{i+1}$  と置き換え、 $x_{i+1} \rightarrow x_i$  方向への計算式

$$y_i = y_{i+1} - h\psi(x_{i+1}, y_{i+1}, -h)$$

を作る。未知数を  $y_{i+1}$  と見れば、これは非線型方程式となる。十分小さい  $h$  に対しては解を持つようにできるので、これを  $y_{i+1}$  について解き、 $y_i$  に修正項を加える形で表現した

$$y_{i+1} = y_i + h\psi^*(x_i, y_i, h) \quad (17.19)$$

この漸化式に基づく解法を (17.18) の随伴法と呼ぶ。

#### 例題 17.5.1

例として Euler 法の随伴法を求めてみよう。(17.18) 式は

$$y_{i+1} = y_i + hf(x_i, y_i)$$

となるので、随伴法のもとになる非線型方程式は

$$y_i = y_{i+1} - hf(x_{i+1}, y_{i+1})$$

であるから

$$y_{i+1} = y_i + hf(x_{i+1}, y_{i+1})$$

という非線型方程式を  $y_{i+1}$  に付いて解いたものが Euler 法の随伴法となる。これは陰的 Euler 法と呼ばれている解法である。

当然のことだが、随伴法の随伴法はもとの一段法になる。しかし、随伴法ともとの解法が一致する、即ち

$$\psi = \psi^* \quad (17.20)$$

となる一段法は特殊である。このような特徴を持つ一段法を、対称な (symmetry) 解法と呼ぶ。対称な一段法は  $h$  の偶数べき乗の漸近展開を持つことが知られている [4]。よってこれを初期系列に用いることで、補外一段ごとに 2 次ずつ上げる解法を作ることができる。

### 17.5.2 偶数ステップ数の中点法

対称な一段法として、積分区間  $[x_0, x_0 + H]$  を  $2n$  に等分割した中点法 (16.9)

$$\begin{aligned} y_1 &:= y_0 + hf(x_0, y_0) \\ y_{i+1} &:= y_{i-1} + 2hf(x_i, y_i) \quad (i = 1, 2, \dots, 2n) \end{aligned} \quad (17.21)$$

がある。もちろん  $h = H/(2n)$ ,  $y_{2n} \approx y(x_0 + H)$  であり,  $y_{2n+1}$  は積分区間外の近似値であるが, これを用いて最後に平滑化 (smoothing)

$$\widehat{y}_{2n} := \frac{1}{2}(y_{2n-1} + y_{2n+1})$$

を施すと, 少し A 安定性が改善されることが知られている。

この偶数ステップ数の中点法は対称である。これを次のように 2 次元化することで示そう。

まず,  $\widehat{x}_i := x_{2i}$  とし

$$\begin{aligned} u_i &:= y_{2i} \\ v_i &:= \frac{1}{2}(y_{2i-1} + y_{2i+1}) \\ &= y_{2i-1} + hf(\widehat{x}_i, y_{2i}) \end{aligned}$$

とする。この時, 偶数ステップ数の中点法 (17.21) は

$$\begin{aligned} u_{i+1} &= y_{2i+2} = y_{2i} + 2hf(x_{2i+1}, y_{2i+1}) \\ &= y_{2i} + 2hf(x_{2i+1}, y_{2i-1} + 2hf(x_{2i}, y_{2i})) \\ &= y_{2i} + 2hf(x_{2i+1}, (y_{2i-1} + hf(x_{2i}, y_{2i})) + hf(x_{2i}, y_{2i})) \\ &= u_i + 2hf(\widehat{x}_i + h, v_i + hf(\widehat{x}_i, u_i)) \\ v_{i+1} &= y_{2i+1} + hf(x_{2i+2}, y_{2i+2}) \\ &= y_{2i-1} + 2hf(x_{2i}, y_{2i}) + hf(x_{2i+2}, y_{2i+2}) \\ &= (y_{2i-1} + hf(x_{2i}, y_{2i})) + h(f(x_{2i}, y_{2i}) + f(x_{2i+2}, y_{2i+2})) \\ &= v_i + h(f(\widehat{x}_i, u_i) + f(\widehat{x}_{i+1}, u_{i+1})) \end{aligned}$$

より,

$$\begin{bmatrix} u_{i+1} \\ v_{i+1} \end{bmatrix} = \begin{bmatrix} u_i \\ v_i \end{bmatrix} + h \begin{bmatrix} 2f(\widehat{x}_i + h, v_i + hf(\widehat{x}_i, u_i)) \\ f(\widehat{x}_i, u_i) + f(\widehat{x}_{i+1}, u_{i+1}) \end{bmatrix} \quad (i = 0, 1, \dots, n-1) \quad (17.22)$$

を得る。これは常微分方程式

$$\begin{cases} \frac{d}{dx} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f(x, v) \\ f(x, u) \end{bmatrix} \\ \begin{bmatrix} u(x_0) \\ v(x_0) \end{bmatrix} = \begin{bmatrix} y_0 \\ y_0 \end{bmatrix} \end{cases}$$

に対応する一段法になり, 解は  $[u(x) \ v(x)]^T = [y(x) \ y(x)]^T$  である。

こうして得られた漸化式 (17.22) によって, 偶数ステップ数の中点法 (17.21) は対称であることが分かる (→ 演習問題 4)。

### 17.5.3 GBS アルゴリズム

以上により、2 次ずつ次数を上げる GBS アルゴリズムは次のような手順で計算が進められることになる。ここでは刻み幅制御を簡略化したものを紹介する。

初期値  $f(x_0)$  からステップ幅  $H_1$  だけ進んだ地点の解  $y(x_1)(= y(x_0 + H_1))$  の近似値  $y_1$  を求めるものとする。

まず、ステップ幅  $H_1$  を定め、これを更に小ステップに分割して初期系列を求める。この分割数を定める補助数列  $\{\widehat{w}_i\}$  は偶数でなければならぬため、

$$\widehat{w}_i := 2w_i$$

とする。これにより、各補助数列は次のように変化する

**Romberg 数列:** 2, 4, 8, 16, ...,  $2^{i+1}$ , ...

**Bulirsch 数列:** 2, 4, 6, 8, 12, 16, 24, ...,  $3 \cdot 2^{i/2}, 2^{(i+1)/2}$ , ...

**調和 (harmonic) 数列:** 2, 4, 6, 8, ...,  $2(i+1)$ , ...

各分割数  $n_i$  に対して初期系列  $T_{i1}$  を中点法 (17.21) と必要があれば平滑化を行って求める。ここで小ステップ  $h$  は  $h = H_1/\widehat{w}_i$  となる。

こうして求められた初期系列  $T_{11}, T_{21}, \dots$  を用いて補外計算

$$T_{i,j} := T_{i,j-1} + \frac{T_{i,j-1} - T_{i-1,j-1}}{\left(\frac{\widehat{w}_i}{\widehat{w}_{i-j+1}}\right)^2 - 1} \quad (17.23)$$

を行い、 $h$  に関して 2 次の漸近展開として表現できる打ち切り誤差項を削っていく。

この補外計算の過程で逐次に収束判定を行い、収束条件を満足すれば、そこで補外計算を打ち切って  $y_1$  を決定し、次の区間の積分計算  $y_2 \approx y(y_1 + H_2)$  を同様にして計算する。

もし最大段数に達しても収束条件を満足しなければ、 $H_1 := H_1/2$  としてもう一度最初から計算をやり直す。

こうして、積分区間を  $[x_0, x_1]$  (ステップ幅  $x_1 - x_0 = H_1$ )  $\rightarrow [x_1, x_2]$  (同  $H_2$ )  $\rightarrow \dots \rightarrow [x_{N-1}, x_N]$  (同  $H_N$ ) と順次進めながら近似値  $y_1, y_2, \dots$  を求める。

### 17.5.4 数値例

以上の簡略化した GBS アルゴリズムを使用して、次の共振問題を計算してみよう。

$$\begin{aligned} \frac{d}{dx} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} &= \begin{bmatrix} y_2 \\ \alpha y_1(-y_1 \sin x + 2y_2 \cos x) \end{bmatrix} \\ \mathbf{y}(0) &= [1 \ \alpha]^T \\ \text{積分区間} &: [0, 37] \end{aligned} \quad (17.24)$$

ここで  $\alpha$  は実定数である。解析解は

$$\mathbf{y}(x) = \begin{bmatrix} \frac{1}{1 - \alpha \sin x} \\ \frac{\alpha \cos x}{(1 - \alpha \sin x)^2} \end{bmatrix} \quad (17.25)$$

である。これで分かるように、 $\alpha$  が 1 に近づくと、どちらの要素も共振点  $(2n+1)\pi/2$  ( $n \in \mathbb{Z}$ ) において急激に増加する。実際、解析解の Taylor 展開を求めてみると係数が急激に大きくなる級数になり、収束半径はごく小さいものであると判断できる。室伏 [25] の数値実験によれば、 $\alpha = 0.99999999$  が IEEE754 倍精度で収束するギリギリの線である。従って、このような問題は多倍長計算を用いて行うのがふさわしい。Romberg 数列と調和数列を用いた補外法を用いて、この共振問題を解いた結果を示す。

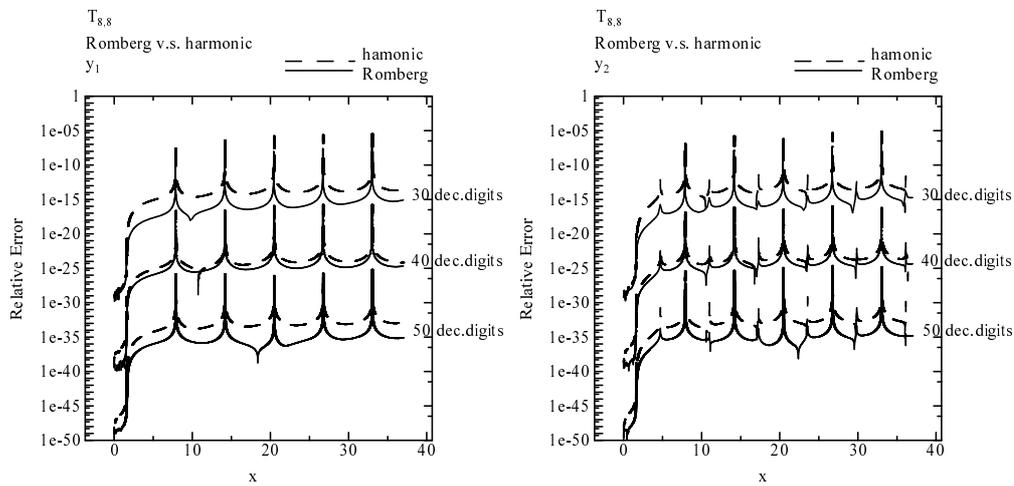


図 17.6: 共振問題  $y_1$ (左) と  $y_2$ (右) の相対誤差 ( $\epsilon_R = 0, 10$  進 30, 40, 50 桁計算)

まず、10 進 30, 40, 50 桁計算した時の近似解の相対誤差の変化を図 17.6 に示す。ここでは全て  $\epsilon_R = 0$  とし、イコール判定 [25] を用いた。この結果から、どちらの補外法においても前述したステップサイズ制御がうまく働き、積分区間において破綻することなく計算できていることがわかる。また、共振点においては刻み幅制御を行っていても誤差が増大していることも併せて見て取れる。どの精度の計算結果でも、大体 15 桁程度は精度が悪化する傾向がある。この場合、全般的に、調和数列の方が、若干相対誤差が大きくなっている。

しかしこの時のステップサイズの変化は、Romberg, 調和数列で全く異なる。この場合、どちらも共振点付近でのみステップサイズが急激に小さくなっているが、それ以外の所では Romberg 数列の方はうまく制御できている一方、調和数列を用いた方はステップサイズ制御がうまくいっておらず、振幅の激しい振動を起こしている。しかも全般的に、調和数列のステップサイズは Romberg 数列のものよりもかなり小さく抑えられてしまっている。このため、本来は計算量が少なく済む調和数列の補外法の方が、トータルの計算時間は Romberg 数列に比べて圧倒的に増大してしまう。

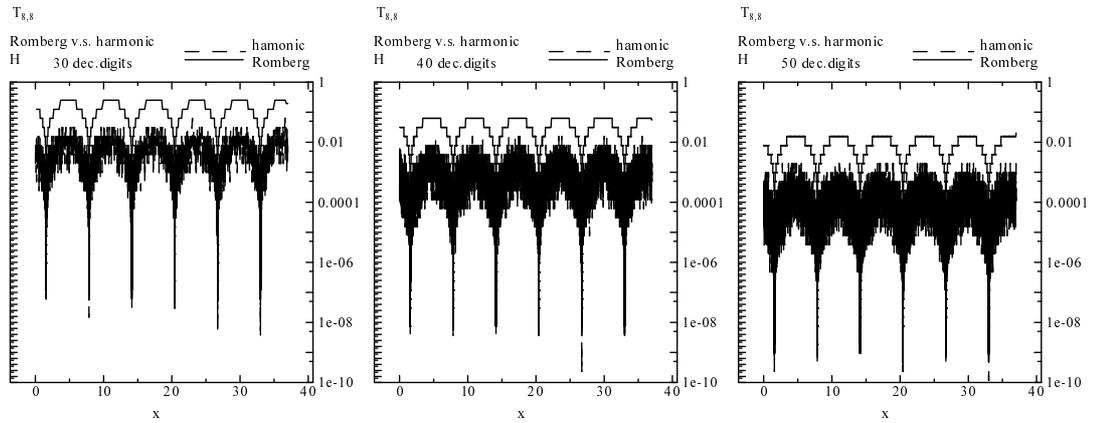


図 17.7: 共振問題の  $H$  の変化 ( $\varepsilon_R = 0$ , 10 進 30(左), 40(中), 50 桁(右) 計算)

これは調和級数利用時の丸め誤差の影響を受けているためである。よって、十分桁数を確保し、 $\varepsilon_R > \varepsilon_M$  と設定することにより回避できる。

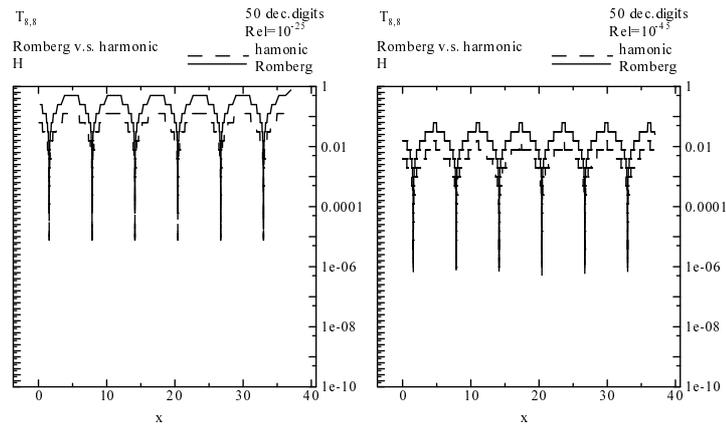


図 17.8: 共振問題の  $H$  の変化 (10 進 50 桁計算,  $\varepsilon_R = 1.0 \times 10^{-25}$ (左),  $1.0 \times 10^{-45}$ (右))

そして、この場合は調和数列を用いても適切な誤差制御ができていることがわかる。

## 演習問題

1.  $S(x)$  の展開式が (17.6) であれば、補外式 (17.3) は (17.7) となることを説明せよ。
2. 中心差分商  $\delta f(x)/h$  を用いて導関数  $f'(x)$  を近似すると、その打ち切り誤差は  $h$  の偶数べき乗の多項式として表現できることを示せ。(Hint:  $x$  の回りで  $f(x \pm h)$  を Taylor 展開 (6.3) し、中心差分商に当てはめよ。)

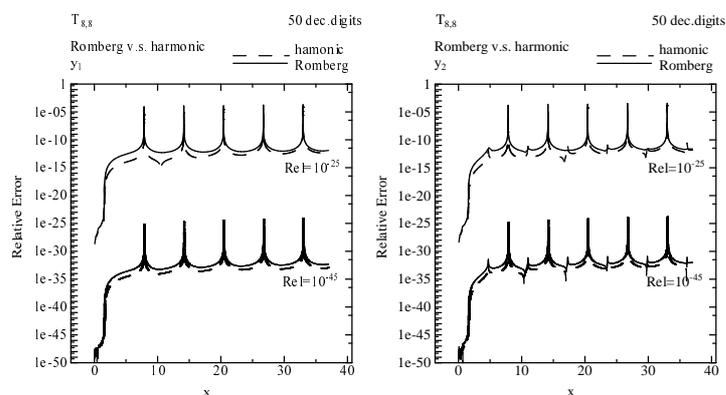


図 17.9: 共振問題  $y_1$ (左) と  $y_2$ (右) の相対誤差 (10 進 50 桁計算)

3. 台形則より次数の高い Newton-Cotes 積分公式 (Simpson の 1/3 公式, 3/8 公式) や Gauss 積分公式を初期系列に用いて補外計算を行ってみよ。
4. 中点法を 2 次元化した (17.22) は対称であることを説明せよ。(Hint:  $[u_{i+1} \ v_{i+1}]^T := [u_i \ v_i]^T$ ,  $[u_i \ v_i]^T := [u_{i+1} \ v_{i+1}]^T$ ,  $\widehat{x_{i+1}} := \widehat{x_i}$ ,  $\widehat{x_i} := \widehat{x_{i+1}}$ ,  $h := -h$  と代入せよ。)
5. Rössler モデル (16.18) の数値計算を GBS アルゴリズムで実行し, 丸め誤差の挙動を調べよ。

## 参考図書

本章の記述は, 永坂秀子・室伏誠・福井義成の研究結果を参考に, 計算量と数値的性質について自分の意見を加えたものである。最も参考にした文献は, 数値微分・Romberg 型積分・GBS アルゴリズムについて網羅的に調査した

有限桁計算における Richardson の補外法による丸め誤差評価の研究

室伏 誠

博士論文 (日本大学)

1998 年

である。