Acceleration Methods for Slowly Convergent Sequences and their Applications

Naoki Osada

Acceleration Methods for Slowly Convergent Sequences and their Applications

January 1993

Naoki Osada

CONTENTS

INTRODUCTION

Sequence transformations

Convergent numerical sequences occur quite often in natural science and engineering. Some of such sequences converge very slowly and their limits are not available without a suitable convergence acceleration method. This is the *raison d'être* of the study of the convergence acceleration method.

A convergence acceleration method is usually represented as a sequence transformation. Let $\mathscr{S}$ and $\mathscr{T}$ be sets of real sequences. A mapping $T : \mathscr{S} \to \mathscr{T}$ is called a *sequence transformation*, and we write $(t_n) = T(s_n)$ for $(s_n) \in \mathscr{S}$. Let $T : \mathscr{S} \to \mathscr{T}$ be a sequence transformation and $(s_n) \in \mathscr{S}$. T *accelerates* (s_n) if

$$\lim_{n\to\infty} \frac{t_n - s}{s_{\sigma(n)} - s} = 0,$$

where $\sigma(n)$ is the greatest index used in the computation of t_n.

An illustration : the Aitken δ^2 process

The most famous sequence transformation is the Aitken δ^2 process defined by

$$t_n = s_n - \frac{(s_{n+1} - s_n)^2}{s_{n+2} - 2s_{n+1} + s_n} = s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n}, \tag{1}$$

where (s_n) is a scalar sequence. As C. Brezinski pointed out[1] , the first proposer of the δ^2 process was the greatest Japanese mathematician Takakazu Seki 関孝和 (or Kōwa Seki, 1642?-1708). Seki used the δ^2 process computing π in Katsuyō Sampō vol. IV 括要算法巻四, which was edited by his disciple Murahide Araki 荒木村英 in 1712. Let s_n be the perimeter of the polygon with 2^n sides inscribed in a circle of diameter one. From

$$s_{15} = 3.14159\ 26\mathit{487}\ \mathit{76985}\ \mathit{6708},$$

$$s_{16} = 3.14159\ 265\mathit{23}\ \mathit{86591}\ \mathit{3571},$$

$$s_{17} = 3.14159\ 2653\mathit{2}\ \mathit{88992}\ \mathit{7759},$$

Seki computed

[1]C. Brezinski, *History of continued fractions and Padé approximants*, Springer-Verlag, Berlin, 1991. p.90.

$$
\begin{aligned}
t_{15} &= s_{16} + \frac{(s_{16}-s_{15})(s_{17}-s_{16})}{(s_{16}-s_{15})-(s_{17}-s_{16})}, \qquad (2)\\
&= 3.14159\ 26535\ 89793\ 2\mathit{476},
\end{aligned}
$$

and he concluded $\pi = 3.14159\ 26535\ 89$.[2] The formula (2) is nothing but the δ^2 process. Seki obtained seventeen-figure accuracy from s_{15}, s_{16} and s_{17} whose figure of accuracy is less than ten.

Seki did not explain the reason for (2), but Yoshisuke Matsunaga 松永良弼 (1692?-1744), a disciple of Murahide Araki, explained it in Kigenkai 起源解, an annotated edition of Katsuyō Sampō as follows. Suppose that $b = a + ar$, $c = a + ar + ar^2$. Then

$$
b + \frac{(b-a)(c-b)}{(b-a)-(c-b)} = \frac{a}{1-r},
$$

the sum of the geometric series $a + ar + ar^2 + \dots$.[3]

It still remains a mystery how Seki derived the δ^2 process, but Seki's application can be explained as follows. Generally, if a sequence satisfies

$$
s_n \sim s + c_1\lambda_1^n + c_2\lambda_2^n + \dots,
$$

where $1 > \lambda_1 > \lambda_2 > \cdots > 0$, then t_n in (1) satisfies

$$
t_n \sim s + c_2\left(\frac{\lambda_1-\lambda_2}{\lambda_1-1}\right)^2\lambda_2^n. \qquad (3)
$$

This result was proved by J. W. Schmidt[48] and P. Wynn[61] in 1966 independently. Since Seki's sequence (s_n) satisfies

$$
s_n = 2^n \sin\frac{\pi}{2^n} = \pi + \sum_{j=1}^{\infty}\frac{(-1)^j\pi^{2j+1}}{(2j+1)!}(2^{-2j})^n,
$$

(3) implies that

$$
t_n \sim \pi + \frac{\pi^5}{5!}\left(\frac{1}{16}\right)^{n+1}.
$$

[2] A. Hirayama, K. Shimodaira, and H. Hirose(eds.), *Takakazu Seki's collected works*, English translation by J. Sudo, (Osaka Kyoiku Tosho, 1974). pp.57-58.
[3] M. Fujiwara, *History of mathematics in Japan before the Meiji era*, vol. II (in Japanese), under the auspices of the Japan Academy, (Iwanami, 1956) (= 日本学士院編, 藤原松三郎著, 明治前日本数学史, 岩波書店). p.180.

In 1926, A. C. Aitken[1] iteratively applied the δ^2 process finding the dominant root of an algebraic equation, and so it is now named after him. He used (1) repeatedly as follows:

$$T_0^{(n)} = s_n, \quad n \in \mathbb{N},$$
$$T_{k+1}^{(n)} = T_k^{(n)} - \frac{(T_k^{(n+1)} - T_k^{(n)})^2}{T_k^{(n+2)} - 2T_k^{(n+1)} + T_k^{(n)}} \quad k = 0, 1, \ldots ; n \in \mathbb{N}.$$

This algorithm is called the *iterated Aitken δ^2 process.*

Derivation of sequence transformations

Many sequence transformations are designed to be exact for sequences of the form

$$s_n = s + c_1 g_1(n) + \cdots + c_k g_k(n), \quad \forall n, \tag{4}$$

where $s, c_1, \ldots, c_k$ are unknown constants and $g_j(n)$ $(j = 1, \ldots k)$ are known functions of n. Since s is the solution of the system of linear equations

$$s_{n+i} = s + c_1 g_1(n+i) + \cdots + c_k g_k(n+i), \quad i = 0, \ldots, k,$$

the sequence transformation $(s_n) \mapsto (t_n)$ defined by

$$t_n = E_k^{(n)} = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+k) \\ & & \cdots & \\ g_k(n) & g_k(n+1) & \cdots & g_k(n+k) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+k) \\ & & \cdots & \\ g_k(n) & g_k(n+1) & \cdots & g_k(n+k) \end{vmatrix}}$$

is exact for the model sequence (4). This sequence transformation includes many famous sequence transformations as follows:

(i) The Aitken δ^2 process : $k = 1$ and $g_1(n) = \Delta s_n$.

(ii) The Richardson extrapolation : $g_j(n) = x_n^j$, where (x_n) is an auxiary sequence.

(iii) Shanks' transformation : $g_j(n) = \Delta s_{n+j-1}$.

(iv) The Levin u-transformation : $g_j(n) = n^{1-j} \Delta s_{n-1}$.

In 1979 and 1980, T. Håvie[20] and C. Brezinski[10] gave independently a recursive algorithm for the computation of $E_k^{(n)}$, which is called the *E-algorithm.*

By the construction, the E-algorithgm accelerates sequences having the aymptotic expansion of the form

$$s_n \sim s + \sum_{j=1}^{\infty} c_j g_j(n), \tag{5}$$

where $s, c_1, c_2, \ldots$ are unknown constants and $(g_j(n))$ is a known asymptotic scale. More precisely, in 1990, A. Sidi[53] proved that if the E-algorithm is applied to the sequence (5), then for fixed k,

$$\frac{E_k^{(n)} - s}{E_{k-1}^{(n)} - s} = O\left(\frac{g_{k+1}(n)}{g_k(n)}\right), \qquad \text{as } n \to \infty.$$

Some sequence transformations are designed to accelerate for sequences having a certain asymptotic property. For example, suppose (s_n) satisfies

$$\lim_{n\to\infty} \frac{s_{n+1} - s}{s_n - s} = \lambda. \tag{6}$$

The Aitken δ^2 process is also obtained by solving

$$\frac{s_{n+2} - s}{s_{n+1} - s} = \frac{s_{n+1} - s}{s_n - s}$$

for the unknown s. Such methods for obtaining a sequence transformation from a formula with limit was proposed by C. Kowalewski[25] in 1981 and is designated as *thechnique du sous-ensemble*, or *TSE* for short. Recently, the TSE has been formulated by N. Osada[40].

When $-1 \le \lambda < 1$ and $\lambda \ne 0$ in (6), the sequence (s_n) is said to be *linearly convergent sequence*. In 1964, P. Henrici[21] proved that the δ^2 process accelerates any linearly convergent sequence. When $|\lambda| > 1$ in (6), the sequence (s_n) diverges but (t_n) converges to s. In this case s is called the *antilimit* of (s_n).

For particular asymptotic scales $(g_j(n))$ such as $g_j(n) = n^{\theta+1-j}$ or $g_j(n) = \lambda^n n^{\theta+1-j}$ with $\theta < 0$ and $-1 \le \lambda < 1$, various sequence transformations which cannot be represented as the E-algorithm are constructed.

A brief history : sequence transformations and asymptotic expansions

Here, we give a history of sequence transformations for a sequence such that

$$s_n \sim s + n^{\theta} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{7}$$

where $\theta < 0$ and $c_0(\neq 0), c_1, \ldots$ are constants independent of n.

According to K. Knopp[24, p.240], the first sequence transformation was indicated by J. Stirling in 1730. Stirling derived the asymptotic expansion $\sum_{i=1}^{n} \log(1+ia)$ and the recursive procedure of coefficients in its expansion[5, p.156]. The Euler-Maclaurin summation formula, which was discovered by L. Euler and C. Maclaurin independently in about 1740, was a quite important contribution to the study of sequence transformations. In 1755, using this formula, L. Euler found

$$1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} \sim \log n + \gamma + \frac{1}{2n} - \sum_{j=1}^{\infty} \frac{B_{2j}}{2jn^{2j}}, \tag{8}$$

where γ is the Euler constant and B_{2j} are the Bernoulli numbers. Since the sequence $\sum_{i=1}^{n} \frac{1}{i} - \log n$ satisfies (7), the formula (8) is the earliest acceleration for a logarithmic sequence satisfying (7).

For the partial sum $s_n = \sum_{i=1}^{n} a_i$ of an infinite series, it is convenient to consider the asymptotic expansion of $a_n/a_{n-1} = \Delta s_{n-1}/\Delta s_{n-2}$. In 1936, W. G. Bickley and J. C. P. Miller[2] considered an accleration method for a slowly convergent series of positive terms such that

$$\frac{\Delta s_{n-1}}{\Delta s_{n-2}} \sim 1 - \frac{A_1}{n} + \frac{A_2}{n^2} + \frac{A_3}{n^3} + \ldots, \tag{9}$$

where $A_1(> 1), A_2, A_3, \ldots$ are constants. We note that an asymptotic expansion (7) implies (9). They assumed

$$s - s_n \sim \Delta s_{n-1} \left(\alpha_{-1} n + \alpha_0 + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \ldots \right),$$

and determined $\alpha_{-1}, \alpha_0, \alpha_1, \ldots$ by using $A_1, A_2, \ldots$. The Bickley-Miller method requires the coefficients $A_1, A_2, \ldots$ in (9), so it is not applicable for a sequence without explicit form of Δs_n.

In 1952, S. Lubkin[29] studied the δ^2 process and proposed the W transformation. Then he proved that the W transform accelerates any convergent sequence satisfying

$$\frac{\Delta s_n}{\Delta s_{n-1}} \sim \alpha_0 + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \ldots,$$

where $\alpha_0, \alpha_1, \ldots$ are constants, and that the δ^2 process accelerates if $\alpha_0 \neq 1$.

Much earlier, in 1927, L. F. Richardson[44] proposed the *deffered approach to the limit*, which is now called the Richardson extrapolation, and he considered $s_n = ((2n+1)/(2n-1))^n$ with

$$s_n - e \sim \frac{c_2}{n^2} + \frac{c_4}{n^4} + \dots .$$

He assumed that the errors $s_n - e$ are proportional to n^{-2} and extrapolated $s_5 + \frac{16}{9}(s_5 - s_4)$, or equivalently, solved

$$\frac{s_5 - e}{s_4 - e} = \frac{16}{25},$$

then he obtained $e = 2.718\mathit{17}$.

In 1966, P. Wynn[61] applied his ϵ-algorithm to a sequence satisfying

$$s_n \sim s + n^{-1} \sum_{j=0}^{\infty} \frac{c_j}{n^j}. \tag{10}$$

The asymptotic expansion (10) is a special case of (7).

It was difficult to accelerate a sequence satisfying (7) until 1950's. Lubkin's W transformation, proposed in 1952, and the θ-algorithm of Brezinski[8], proposed in 1971, can accelerate such a sequence. And when θ in (7) is negative integer, the ρ-algorithm of Wynn[60], proposed in 1956, works well on the sequence. These transformations do not require the knowledge of θ in (7).

In 1981, P. Bjørstad, G. Dahlquist and E. Grosse[4] proposed the modified Aitken δ^2 formula defined by

$$\begin{aligned} s_0^{(n)} &= s_n \\ s_{k+1}^{(n)} &= s_k^{(n)} - \frac{2k+1-\theta}{2k-\theta} \frac{(s_k^{(n+1)} - s_k^{(n)})(s_k^{(n)} - s_k^{(n-1)})}{s_k^{(n+1)} - 2s_k^{(n)} + s_k^{(n-1)}}, \end{aligned}$$

and proved that if it is applied to a sequence satisfying (7), then for fixed k,

$$s_k^{(n)} - s = O(n^{\theta - 2k}), \qquad \text{as } n \to \infty.$$

In 1990, N. Osada[37] proposed the generalized ρ-algorithm defined by

$$\begin{aligned} \rho_{-1}^{(n)} &= 0, \quad \rho_0^{(n)} = s_n \\ \rho_k^{(n)} &= \rho_{k-2}^{(n+1)} + \frac{k-1-\theta}{\rho_{k-1}^{(n+1)} - \rho_{k-1}^{(n)}} \end{aligned}$$

and proved that if it is applied to a sequence satisfying (7), then for fixed k,

$$\rho_{2k}^{(n)} - s = O((n+k)^{\theta-2k}), \qquad \text{as } n \to \infty.$$

The modified Aitken δ^2 formula and the generalized ρ-algorithm require the knowledge of the exponent θ in (7). But Osada showed that θ can be computed using these methods as follows. For a given sequence (s_n) satisfying (7), θ_n denotes by

$$\theta_n = 1 + \frac{1}{\Delta\left(\dfrac{\Delta s_n}{\Delta^2 s_{n-1}}\right)},$$

then Bjørstad, Dahlquist and Grosse[4] proved that the sequence (θ_n) has the asymptotic expansion of the form

$$\theta_n \sim \theta + n^{-2}\sum_{j=0}^{\infty}\frac{t_j}{n^j}.$$

Thus by applying these methods with the exponent -2 to (θ_n), the exponent θ in (7) can be estimated.

Organization of this paper

In Chapter I, the asymptotic properties of slowly convergent scalar sequences are dealt, and examples of sequence, which will be taken up in Chapter II as an objective of application of acceleration methods, are given. In Section 1, asymptotic preliminaries, i.e., O-symbol, the Euler-Maclaurin summation formula and so on, are introduced. In Section 2, some terminologies for slowly convergent sequences are given. In Section 3 and 4, partial sums of infinite series, and numerical integration are taken up as slowly convergent scalar sequences.

In Chapter II, acceleraton methods for scalar sequences are dealt. Taken up methods are as follows:

(i) Methods added some new results by the author; the ρ-algorithm, the generalized ρ-algorithm, and the modified Aitken δ^2 formula.

(ii) Other important methods; the E-algorithm, the Richardson extrapolation, the ϵ-algorithm, Levin's transforms, the d transform, the Aitken δ^2 process, and the Lubkin W transform.

For these methods, the derivation, convergence theorem or asymptotic behaviour, and numerical examples are given. For methods mentioned in (i), details are described, but for others only important facts are described. For other information, see Brezinski[9], Brezinski and Redivo Zaglia[11], Weniger[56][57], and Wimp[58].

In Section 14, these methods are compared using numerical examples of infinite series. In Section 15, these methods are applied to numerical integration.

For convenience's sake, FORTRAN subroutines of the automatic generalized ρ-algorithm and the automatic modified Aitken δ^2 formula are appended.

The numerical computations in this paper were carried out on NEC ACOS-610 computer at Computer Science Center of Nagasaki Institute of Applied Science in double precision with approximately 16 digits unless otherwise stated.

I. Slowly convergent sequences

1. Asymptotic preliminaries

P. Henrici said "The study of the asymptotic behaviour frequently reveals information which enables one to speed up the convergence of the algorithm."([21, p.10])

In this section, we prepare for asymptotic methods.

1.1 Order symbols and asymptotic expansions

Let a and δ be a real number and a positive number, respectively. The set $\{x \in \mathbb{R} \,|\, 0 < |x - a| < \delta\}$ is called a *deleted neighbourhood of* a. The open interval $(a, a + \delta)$ is called a deleted neighbourhood of $a + 0$. For a positive number M, the open interval $(M, +\infty)$ is called a deleted neighbourhood of $+\infty$.

Let b be one of a, $a \pm 0$, $\pm\infty$. Let V be a deleted neighbourhood of b. If a function $f(x)$ defined on V satisfies $\lim_{x\to b} f(x) = 0$, $f(x)$ is said to be *infinitesimal* at b.

Suppose that $f(x)$ and $g(x)$ are infinitesimal at b. We write

$$f(x) = O(g(x)) \text{ as } x \to b, \tag{1.1}$$

if there exist a constant $C > 0$ and a deleted neighbourhood V of b such that

$$|f(x)| \le C|g(x)|, \quad x \in V. \tag{1.2}$$

And we write

$$f(x) = o(g(x)) \text{ as } x \to b, \tag{1.3}$$

if for any $\epsilon > 0$ there exists a deleted neighbourhood V_ϵ of b such that

$$|f(x)| \le \epsilon|g(x)|, \quad x \in V_\epsilon. \tag{1.4}$$

In the rest of this subsection b is fixed and the qualifying phrase "as $x \to b$" is omitted.

Let $f_1(x) - f_2(x)$ and $f_3(x)$ be infinitesimal at b. We write $f_1(x) = f_2(x) + O(f_3(x))$, if $f_1(x) - f_2(x) = O(f_3(x))$. Similarly we write $f_1(x) = f_2(x) + o(f_3(x))$, if $f_1(x) - f_2(x) = o(f_3(x))$.

If $f(x)/g(x)$ tends to unity, we write

$$f(x) \sim g(x). \tag{1.5}$$

Then g is called an *asymptotic approximation to* f.

A sequence of functions $(f_n(x))$ defined in a deleted neighbourhood V of b is called an *asymptotic scale* or an *asymptotic sequence* if

$$f_{n+1}(x) = o(f_n(x)), \text{ for } n = 1, 2, \ldots, \tag{1.6}$$

is valid. Let $(f_n(x))$ be an asymptotic scale defined in V. Let $f(x)$ be a function defined in V. If there exist constants $c_1, c_2, \ldots,$ such that

$$f(x) = \sum_{k=1}^{n} c_k f_k(x) + o(f_n(x)), \tag{1.7}$$

is valid for any $n \in \mathbb{N}$, then we write

$$f(x) \sim \sum_{k=1}^{\infty} c_k f_k(x), \tag{1.8}$$

and (1.8) is called an *asymptotic expansion* of $f(x)$ with respect to $(f_n(n))$. We note that (1.8) implies $f(x) \sim c_1 f_1(x)$ in the sense of (1.5).

If $f(x)$ has an asymptotic expansion (1.8), then the coefficients (c_k) are unique:

$$c_1 = \lim_{x \to b} f(x)/f_1(x), \tag{1.9a}$$

$$c_n = \lim_{x \to b} \left((f(x) - \sum_{k=1}^{n-1} c_k f_k(x))/f_n(x) \right), \quad n = 2, 3, \ldots. \tag{1.9b}$$

When $f(x) - g(x) \sim \sum_{k=1}^{\infty} c_k f_k(x)$, we often write

$$f(x) \sim g(x) + \sum_{k=1}^{\infty} c_k f_k(x) \tag{1.10}$$

For asymptotic methods, see Bruijn[12] and Olver[34].

1.2 The Euler-Maclaurin summation formula

The Euler-Maclaurin summation formula is a quite useful theorem not only for numerical integration but also for sequences and infinite series. In this subsection we review the Euler-Maclaurin formulæ without proofs, which can be found in Bourbaki[5]. We begin with the Bernoulli numbers.

The Bernoulli numbers B_n are defined by

$$\frac{x}{e^x - 1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} x^n, \quad |x| < 2\pi, \tag{1.11}$$

where the left-hand side of (1.11) equals 1 when $x = 0$. The Bernoulli numbers are computed recursively by

$$B_0 = 1 \tag{1.12a}$$

$$\sum_{k=0}^{n-1} \binom{n}{k} B_k = 0 \quad n = 2, 3, \ldots. \tag{1.12b}$$

By the relations (1.12), we have

$$\begin{aligned}
&B_0 = 1,\ B_1 = -\frac{1}{2},\ B_2 = \frac{1}{6},\ B_3 = 0,\ B_4 = -\frac{1}{30},\ B_5 = 0,\\
&B_6 = \frac{1}{42},\ B_7 = 0,\ B_8 = -\frac{1}{30},\ B_9 = 0,\ B_{10} = \frac{5}{66},\ B_{11} = 0,\\
&B_{12} = -\frac{691}{2730},\ B_{13} = 0,\ B_{14} = \frac{7}{6},\ B_{15} = 0,\ B_{16} = -\frac{3617}{510}.
\end{aligned} \tag{1.13}$$

It is known that

$$|B_{2j}| \le \frac{4(2j)!}{(2\pi)^{2j}} \quad \text{for } \forall j \in \mathbb{N}. \tag{1.14}$$

Theorem 1.1 (The Euler-Maclaurin summation formula)

Let $f(x)$ be a function of C^{2p+2} class in a closed interval $[a, b]$. Then the following asymptotic formula is satisfied.

$$T_n - \int_a^b f(x)dx = \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!} h^{2j} \left(f^{(2j-1)}(b) - f^{(2j-1)}(a) \right) + O(h^{2p+2}), \text{ as } h \to +0, \tag{1.15a}$$

where

$$h = (b-a)/n, \quad T_n = h \left(\frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a + ih) + \frac{1}{2} f(b) \right). \tag{1.15b}$$

T_n in (1.15b) is called an *n-panels compound trapezoidal rule*, or a *trapezoidal rule*, for short. The following modifications of the Euler-Maclaurin summation formula are useful for infinite series.

Theorem 1.2

Let $f(x)$ be a function of C^{2p+1} class in $[n, n+m]$. Then

$$\begin{aligned}
\sum_{i=0}^{m} f(n+i) = &\int_n^{n+m} f(x)dx + \frac{1}{2}\left(f(n) + f(n+m) \right)\\
&+ \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!} \left(f^{(2j-1)}(n+m) - f^{(2j-1)}(n) \right) + R_p(n, m),
\end{aligned} \tag{1.16}$$

$$|R_p(n,m)| \le \frac{4e^{2\pi}}{(2\pi)^{2p+1}} \int_n^{n+m} |f^{(2p+1)}(x)|dx. \tag{1.17}$$

In particular, if $f^{(2p+1)}(x)$ has a definite sign in $[n, n+m]$, then

$$|R_p(n,m)| \le \frac{4e^{2\pi}}{(2\pi)^{2p+1}} |f^{(2p)}(n+m) - f^{(2p)}(n)|. \tag{1.18}$$

Theorem 1.3

Let $f(x)$ be a function of C^{2p+1} class in $[n, n+2m]$. Then

$$\sum_{i=0}^{2m} (-1)^i f(n+i) = \frac{1}{2}\left(f(n) + f(n+2m)\right) + \sum_{j=1}^{p} \frac{B_{2j}(2^{2j}-1)}{(2j)!} \left(f^{(2j-1)}(n+2m) - f^{(2j-1)}(n)\right) + R_p(n,m), \tag{1.19}$$

$$|R_p(n,m)| \le \frac{4e^{2\pi}(2^{2p+1}+1)}{(2\pi)^{2p+1}} \int_n^{n+2m} |f^{(2p+1)}(x)|dx. \tag{1.20}$$

In particular, if $f^{(2p+1)}(x)$ has a definite sign in $[n, n+2m]$, then

$$|R_p(n,m)| \le \frac{4e^{2\pi}(2^{2p+1}+1)}{(2\pi)^{2p+1}} |f^{(2p)}(n+2m) - f^{(2p)}(n)|. \tag{1.21}$$

The following theorem gives the asymptotic expansion of the midpoint rule.

Theorem 1.4

Let $f(x)$ be a function of C^{2p+2} class in a closed interval $[a, b]$. Then the following asymptotic formula is satisfied.

$$M_n - \int_a^b f(x)dx = \sum_{j=1}^{p} \frac{(2^{1-2j}-1)B_{2j}}{(2j)!} h^{2j} \left(f^{(2j-1)}(b) - f^{(2j-1)}(a)\right) + O(h^{2p+2}), \text{ as } h \to +0, \tag{1.22a}$$

where

$$h = (b-a)/n, \quad M_n = h\sum_{i=1}^{n} f(a + (i - \frac{1}{2})h). \tag{1.22b}$$

Proof. Since $M_n = 2T_{2n} - T_n$, it follows from Theorem 1.1. □

M_n in (1.22b) is called an *n-panels compound midpoint rule*, or a *midpoint rule*, for short.

2. Slowly convergent sequences

When one deals with the *speed of convergence* such as *slow convergence* and *convergence acceleration*, it is necessary to quantitatively represent the speed of convergence. To this end, we use the *order of convergence*, the *rate of contraction* and the *asymptotic expansion*.

2.1 Order of convergence

Let (s_n) be a real sequence converging to a limit s. For $p \geq 1$, (s_n) is said to *have order* p or *be* p*-th order convergence* if there exist A, $B \in \mathbb{R}$, $n_0 \in \mathbb{N}$ such that $0 < A \leq B$ and

$$A \leq \frac{|s_{n+1} - s|}{|s_n - s|^p} \leq B \quad \forall n \geq n_0. \tag{2.1}$$

"2-nd order convergence" is usually called a *quadratic convergence* and "3-rd order convergence" is sometimes called a *cubic convergence*.

If there exist $C > 0$ and $n_0 \in \mathbb{N}$ such that

$$|s_{n+1} - s| \leq C|s_n - s|^p \quad \forall n \geq n_0, \tag{2.2}$$

then (s_n) is said to *have at least order* p or *be at least* p*-th order convergence*.

As is well known, under suitable conditions, Newton's iteration

$$s_{n+1} = s_n - \frac{f(s_n)}{f'(s_n)} \tag{2.3}$$

converges at least quadratically to a simple solution of $f(x) = 0$. Similarly, a sequence generated by the secant method

$$s_{n+1} = s_n - f(s_n)\frac{s_n - s_{n-1}}{f(s_n) - f(s_{n-1})} \tag{2.4}$$

has order at least $(1 + \sqrt{5})/2 = 1.618\ldots$. This convergence is sufficiently fast.

A p-th order convergent sequence (s_n) is said to have the *asymptotic error constant* $C > 0$ if

$$\lim_{n\to\infty} \frac{|s_{n+1} - s|}{|s_n - s|^p} = C. \tag{2.5}$$

We note that p-th order convergent sequences not necessarily have the asymptotic error constant. For example, let (s_n) be a sequence defined by

$$s_{n+1} = \left(\frac{1}{2} + \frac{1}{4}(-1)^n\right)(s_n - s)^p + s. \tag{2.6}$$

If $p = 1$, or $p > 1$ and $0 < 3^{1/(1-p^2)}(3/4)^{1/(p-1)}(s_0 - s) < 1$, then (s_n) converges to s and satisfies (2.1) with $A = 1/4$, $B = 3/4$, but $|s_{n+1} - s|/|s_n - s|^p$ does not converge.

If (s_n) has order $p > 1$ and $|s_{n+1} - s|/|s_n - s|$ converges, then

$$\lim_{n\to\infty} \frac{s_{n+1} - s}{s_n - s} = 0. \tag{2.7}$$

A sequence (s_n) satisfying (2.7) is called a *super-linearly convergent sequence.*

We note that a super-linearly convergent sequence has not necessarily order $p > 1$. For example, let (s_n) be a sequence defined by

$$s_{n+1} = \lambda^n(s_n - s) + s, \quad 0 < |\lambda| < 1. \tag{2.8}$$

Then (s_n) converges super-linearly to s, but has not any order.

2.2 Linearly convergent sequences

When

$$\lim_{n\to\infty} \frac{s_{n+1} - s}{s_n - s} = \lambda, \quad -1 \le \lambda < 1, \lambda \neq 0, \tag{2.9}$$

(s_n) is called a *linearly convergent sequence* and λ is called the *rate of contraction.*

In practice, linearly convergent sequences occur in the following situations:

(i) Partial sums s_n of alternating series satisfies (2.9) with the rate of contraction $\lambda = -1$.

(ii) Suppose that $f(x)$ has a zero α of multiplicity $m > 1$ and is of class C^2 in a neighbourhood of α. If s_0 is sufficiently close to α, then Newton's iteration (2.3) converges linearly to α with the rate of contraction $1 - 1/m$.

(iii) Suppose that an equation $x = g(x)$ has a fixed point α, $g(x)$ is of class C^2 in a neighbourhood of α and $0 < |g'(\alpha)| < 1$. If s_0 is sufficiently close to α, then (s_n) generated by $s_{n+1} = g(s_n)$ converges linearly to α with the rate of contraction $g'(\alpha)$.

The convergence of a linearly convergent sequence whose asymptotic error constant is close to 1 is so slow that it is necessary to accelerate the convergence. However, it is easy to accelerate linearly convergent sequences. For example, the Aitken δ^2 process can accelerate any linealy convergent sequence(Henrici[21]).

Some linearly convergent sequences have an asymptotic expansion of the form

$$s_n \sim s + \sum_{j=1}^{\infty} c_j \lambda_j^n, \qquad \text{as } n \to \infty, \tag{2.10}$$

where $c_1, c_2, \ldots$ and $\lambda_1, \lambda_2, \ldots$ are constants independent of n. A sequence (s_n) satisfying (2.10) with known constants $\lambda_1, \lambda_2, \ldots$ can be quite efficiently accelerated by the Richardson extrapolation. When $\lambda_1, \lambda_2, \ldots$ in (2.10) are unknown, the sequence can be efficiently accelerated by the ϵ-algorithm.

Other some linearly convergent sequences such as the partial sums of certain alternating series (see Theorem 3.3) have an asymptotic expansion of the form

$$s_n \sim s + \lambda^n n^\theta \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \qquad \text{as } n \to \infty, \tag{2.11}$$

where $-1 \leq \lambda < 1$, $\lambda \neq 0$, $\theta < 0$ and $c_0(\neq 0), c_1, \ldots$ are constants independent of n. A sequence (s_n) satisfying (2.11) can be efficiently accelerated by the Levin transformations.

2.3 Logarithmically convergent sequences

When the equality

$$\lim_{n\to\infty} \frac{s_{n+1} - s}{s_n - s} = 1 \tag{2.12}$$

holds (s_n) is called a *logarithmically convergent sequence*(Overholt[42]), or a *logarithmic sequence* for short.

A typical example of logarithmically convergent sequences is the partial sums of the Riemann zeta function $\zeta(\sigma)$:

$$s_n = \sum_{j=1}^{n} \frac{1}{j^\sigma}, \quad \sigma > 1. \tag{2.13}$$

As we shall show in Example 3.4, (s_n) has the asymptotic expansion

$$s_n \sim \zeta(\sigma) + n^{1-\sigma} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{2.14}$$

where $c_0, c_1, \ldots$ are constants such as $c_0 = 1/(1-\sigma)$, $c_1 = 1/2$, $c_2 = -\sigma/12$, $c_3 = 0$.

Similarly to (2.14), many logarithmic sequences (s_n) have the asymptotic expansion of the form

$$s_n \sim s + n^\theta \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{2.15}$$

where $\theta < 0$ and $c_0(\neq 0), c_1, \ldots$ are constants. Some other logarithmic sequences (s_n) have the asymptotic expansion of the form

$$s_n \sim s + n^\theta (\log n)^\tau \sum_{j=0}^{\infty} \sum_{i=0}^{\infty} \frac{c_{i,j}}{(\log n)^i n^j}, \tag{2.16}$$

where $\theta < 0$ or $\theta = 0$ and $\tau < 0$, and $c_{0,0}(\neq 0), c_{i,j}, \ldots$ are constants. The asymptotic formula (2.15) or (2.16) is a special case of the following one

$$s_n = s + O(n^\theta), \qquad \text{as } n \to \infty, \tag{2.17}$$

or

$$s_n = s + O(n^\theta (\log n)^\tau), \qquad \text{as } n \to \infty, \tag{2.18}$$

respectively. When θ in (2.17) or (2.18) is close to 0, (s_n) converges very slowly to s, but when $-\theta$ is sufficiently large, e.g. $-\theta > 10$, (s_n) converges rapidly to s.

Furthermore, logarithmic sequences which have the asymptotic formula

$$s_n = s + O((\log(\log n))^\tau), \qquad \text{as } n \to \infty, \quad (\tau < 0) \tag{2.19}$$

occur in some literatures. Sequences satisfying (2.18) with $\theta = 0$ or (2.19) converge quite slowly.

According to their origin we can classify practical logarithmic sequences into the following two categories:

(a) one from continuous problems by applying a discretization method.

(b) one from discrete problems.

There are many numerical problems in the class (a) such as numerical differentiation, numerical integration, ordinary differential equations, partial differential equations, integral equations and so on. Let s be the true value of such a problem and h a mesh size. For many cases an approximation $T(h)$ has an asymptotic formula

$$T(h) = s + \sum_{j=1}^{k} c_j h^{\alpha_j} + O(h^{\alpha_{k+1}}), \tag{2.20}$$

where $c_1, \ldots, c_k$, $0 < \alpha_1 < \cdots < \alpha_{k+1}$ are constants. Setting $s_n = T(c/n)$ and $d_j = c_j c^{\alpha_j}$ for $c > 0$, we have

$$s_n = s + \sum_{j=1}^{k} d_j n^{-\alpha_j} + O(n^{-\alpha_{k+1}}), \tag{2.21}$$

which is a generalization of (2.15). When we put $s'_n = s_{2^n}$ and $\lambda_j = 2^{-\alpha_j}$, we have

$$s'_n = s + \sum_{j=1}^{k} d_j \lambda_j^n + O(\lambda_{k+1}^n), \tag{2.22}$$

therefore the subsequence (s'_n) of (s_n) converges linealy to s.

For example, let $f(x)$ be a function of class C^{2p+2} in a closed interval $[a, b]$. Let T_n be an approximation of $\int_a^b f(x)dx$ by the n-panels compound trapezoidal rule. Then by the Euler-Maclaurin formula (Theorem 1.1),

$$T_n = \int_a^b f(x)dx + \sum_{j=1}^{p} c_j (2^{-2j})^n + O((2^{-2p-2})^n), \qquad \text{as } n \to \infty, \tag{2.23}$$

where $c_1, \ldots, c_m$ are constants independent of n. An application of the Richardson extrapolation to T_{2^n} is called the *Romberg integration*.

Almost all sequences taken up as examples of logarithmic sequences are of class (b). For example, a sequence given by analytic function such as $s_n = (1+1/n)^n$, partial sums of infinite series of positive terms, and an iterative sequence of a singular fixed point problem. Then (s_n) satisfies (2.15), (2.16) or more general form

$$s_n = s + \sum_{j=1}^{k} c_j g_j(n) + O(g_{k+1}(n)), \tag{2.24}$$

such that

$$\lim_{n\to\infty} \frac{g_{j+1}(n)}{g_j(n)} = 0, \qquad j = 1, 2, \ldots, \tag{2.25}$$

and

$$\lim_{n\to\infty} \frac{g_j(n+1)}{g_j(n)} = 1, \qquad j = 1, 2, \ldots. \tag{2.26}$$

A double sequence $(g_j(n))$ satisfying (2.25) and (2.26) is called an *asymptotic logarithmic scale.*

2.4 Criterion of linearly or logarithmically convergent sequences

The formulae (2.9) and (2.12) involve the limit s, so that they are of no use in practice. However, under certain conditions, (2.9) and (2.12) can be replaced $s_n - s$ with $\Delta s_n = s_{n+1} - s_n$. Namely, if $\lim_{n\to\infty} \Delta s_{n+1}/\Delta s_n = \lambda$, and if one of the following conditions

(i) (Wimp[58]) $0 < |\lambda| < 1$, or $|\lambda| > 1$, (For the divergence case $|\lambda| > 1$, s can be any number.)

(ii) (Gray and Clark[17]) $\lambda = -1$ and

$$\lim_{n\to\infty} \frac{1 + \dfrac{\Delta s_{n+1}}{\Delta s_n}}{1 + \dfrac{\Delta s_n}{\Delta s_{n-1}}} = -1, \tag{2.27}$$

is satisfied, then (2.9) holds.

Moreover, if

(iii) (Gray and Clark[17]) (s_n) converges and Δs_n have the same sign,

then (2.12) holds. In particular, if a real monotone sequence (s_n) with limit s satisfies

$$\lim_{n\to\infty} \frac{\Delta s_{n+1}}{\Delta s_n} = 1, \tag{2.28}$$

then (s_n) converges logarithmically to the limit s.

3 Infinite series

Most popular slowly convergent sequences are alternating series as well as logarithmically convergent series. In this section we describe the asymptotic expansions of infinite series and give examples that will be used as test problems.

3.1 Alternating series*

We begin with the definition of the property (C) that is a special case of Widder's completely monotonic functions[4].

Let $a > 0$. A function $f(x)$ has the *property (C)* in (a, ∞), if the following four conditions are satisfied:

(i) $f(x)$ is of class C^∞ in (a, ∞),

(ii) $(-1)^r f^{(r)}(x) > 0$ for $x > a,\ r = 0, 1, \ldots$ (completely monotonic),

(iii) $f(x) \to 0$ as $x \to \infty$,

(iv) for $r = 0, 1, \ldots,\ f^{(r+1)}(x)/f^{(r)}(x) \to 0$ as $x \to \infty$.

If $f(x)$ has the property (C), then for $r = 0, 1, \ldots,\ f^{(r)}(x) \to 0$ as $x \to \infty$, thus $(f^{(r)}(x))_{r=0,1,\ldots}$ is an asymptotic scale.

Theorem 3.1 *Suppose that a function $f(x)$ has the property (C) in (a, ∞). Then*

$$\sum_{i=0}^{\infty}(-1)^i f(n+i) \sim \frac{1}{2}f(n) - \sum_{j=1}^{\infty}\frac{B_{2j}(2^{2j}-1)}{(2j)!}f^{(2j-1)}(n), \quad \text{as } n \to \infty, \tag{3.1}$$

where B_{2j}'s are the Bernoulli numbers.

Proof. Let $m, n, p \in \mathbb{N}$ with $n > a$. By the Euler-Maclaurin formula (Theorem 1.3), we have

$$\sum_{i=0}^{2m}(-1)^i f(n+i) = \frac{1}{2}\left(f(n) + f(n+2m)\right) + \sum_{j=1}^{p}\frac{B_{2j}(2^{2j}-1)}{(2j)!}\left(f^{(2j-1)}(n+2m) - f^{(2j-1)}(n)\right) + R_p(n,m) \tag{3.2}$$

where

$$|R_p(n,m)| \le \frac{4e^{2\pi}(2^{2p+1}+1)}{(2\pi)^{2p+1}}\int_n^{n+2m}|f^{(2p+1)}(x)|dx. \tag{3.3}$$

---

*The material in this subsection is taken from the author's paper: N. Osada, Asymptotic expansions and acceleration methods for alternating series (in Japanese), *Trans. Inform. Process. Soc. Japan*, 28(1987) No.5, pp.431–436. (= 長田直樹, 交代級数の漸近展開と加速法, 情報処理学会論文誌)

[4] See, D. V. Widder, *The Laplace transform*, (Princeton, 1946), p.145.

By the assumption (ii), $f^{(2p+1)}(x) < 0$ for $x > a$, thus

$$|R_p(n,m)| \le \frac{4e^{2\pi}(2^{2p+1}+1)}{(2\pi)^{2p+1}} \left(f^{(2p)}(n) - f^{(2p)}(n+2m) \right). \tag{3.4}$$

Letting $m \to \infty$, the series in the left-hand side of (3.2) converges and $f^{(r)}(n+2m) \to 0$ for $r = 0, 1, \ldots$. So we obtain

$$\sum_{i=0}^{\infty}(-1)^i f(n+i) = \frac{1}{2}f(n) - \sum_{j=1}^{p} \frac{B_{2j}(2^{2j}-1)}{(2j)!} f^{(2j-1)}(n) + O(f^{(2p)}(n)). \tag{3.5}$$

Since (3.5) and the assumption (iv), we obtain (3.1). □

Theorem 3.2 *Suppose that an alternating series is represented as*

$$s = \sum_{i=1}^{\infty}(-1)^{i-1} f(i), \tag{3.6}$$

where $f(x)$ has the property (C) in (a, ∞) for some $a > 0$. Let s_n be the n-th partial sum of (3.6). Then the following asymptotic expansion holds:

$$s_n - s \sim (-1)^{n-1} \left(\frac{1}{2}f(n) + \sum_{j=1}^{\infty} \frac{B_{2j}(2^{2j}-1)}{(2j)!} f^{(2j-1)}(n) \right), \quad \text{as } n \to \infty. \tag{3.7}$$

Proof. Since

$$s_n - s = (-1)^{n-1} f(n) + (-1)^n \sum_{i=1}^{\infty}(-1)^{i-1} f(n+i-1), \tag{3.8}$$

by (3.1) we obtain (3.7). □

Notation (Levin and Sidi[27]) Let $\theta < 0$. We denote $\mathscr{A}^{(\theta)}$ by the set of all functions of class C^∞ in (a, ∞) for some $a > 0$ satisfying the following two conditions:

(i) $f(x)$ has the asymptotic expansion

$$f(x) \sim x^\theta \sum_{j=0}^{\infty} \frac{a_j}{x^j}, \quad \text{as } x \to \infty, \tag{3.9}$$

(ii) Derivatives of any order of $f(x)$ have asymptotic expansions, which can be obtained by differentiating that in (3.9) formally term by term.

Theorem 3.3 *Suppose that $f(x) \in \mathscr{A}^{(\theta)}$ has the property (C). Let the asymptotic expansion of $f(x)$ be (3.9). Let s_n be the n-th partial sum of the series $s = \sum_{i=1}^{\infty}(-1)^{i-1}f(i)$. Then $s_n - s$ has the asymptotic expansion*

$$s_n - s \sim (-1)^{n-1} n^{\theta} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \quad \text{as } n \to \infty, \tag{3.10}$$

where

$$c_j = \frac{1}{2}a_j + \sum_{k=1}^{\lfloor (j+1)/2 \rfloor} \frac{B_{2k}(2^{2k}-1)}{(2k)!} a_{j+1-2k} \prod_{i=1}^{2k-1} (\theta - j + i) \tag{3.11}$$

Proof. Since $f(x) \in \mathscr{A}^{(\theta)}$, we have

$$f^{(2k-1)}(x) \sim \sum_{m=0}^{\infty} a_m \left(\prod_{i=1}^{2k-1} (\theta - m + 1 - i) \right) x^{\theta - m - 2k + 1}. \tag{3.12}$$

Computing the coefficient of $(-1)^{n-1} n^{\theta - j}$,

$$c_j = \frac{1}{2}a_j + \sum_{k,m} \frac{B_{2k}(2^{2k}-1)}{(2k)!} a_m \prod_{i=1}^{2k-1} (\theta - m + 1 - i), \tag{3.13}$$

where the summation in the right-hand side of (3.13) is taken all integers k, m such that

$$k \geq 1, \; m \geq 0, \; m + 2k - 1 = j. \tag{3.14}$$

Since the solutions of (3.14) are only $k = 1, 2, \ldots, \lfloor (j+1)/2 \rfloor$, we obtain the desired result. $\square$

Using Theorem 3.2 and Theorem 3.3, we can obtain the asymptotic expansions of typical alternating series.

Example 3.1 In order to illustrate Theorem 3.2, we consider first a very simple example,

$$\log 2 = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i}. \tag{3.15}$$

Let $f(x) = 1/x$ and s_n be the n-th partial sum of (3.15). Since $f^{(2j-1)}(x) = -(2j-1)!x^{-2j}$, by (3.7) we have

$$s_n - \log 2 \sim (-1)^{n-1} \left(\frac{1}{2n} - \sum_{j=1}^{\infty} \frac{B_{2j}(2^{2j}-1)}{(2j)n^{2j}} \right). \tag{3.16}$$

The first 5 terms of the right-hand side of (3.16) are as follows:

$$s_n - \log 2 = (-1)^{n-1}\frac{1}{n}\left(\frac{1}{2} - \frac{1}{4n} + \frac{1}{8n^3} - \frac{1}{4n^5} + \frac{17}{16n^7} + O(\frac{1}{n^9})\right). \tag{3.17}$$

Example 3.2 In order to illustrate Theorem 3.3, we next consider the Leibniz series

$$\frac{\pi}{4} = \sum_{i=1}^{\infty}\frac{(-1)^{i-1}}{2i-1}. \tag{3.18}$$

Since

$$\frac{1}{2x-1} = \frac{1}{2x}\left(1 + \sum_{j=1}^{\infty}\frac{1}{(2x)^j}\right), \quad |x| > \frac{1}{2}, \tag{3.19}$$

$f(x) = 1/(2x-1)$ belongs to $\mathscr{A}^{(-1)}$. Using Theorem 3.3, we have

$$s_n - \frac{\pi}{4} = (-1)^{n-1}\frac{1}{n}\left(\frac{1}{4} - \frac{1}{16n^2} + \frac{5}{64n^4} - \frac{61}{256n^6} + \frac{1385}{1024n^8} + O(\frac{1}{n^{10}})\right), \tag{3.20}$$

where s_n is the n-th partial sum of (3.18).

Example 3.3 Let us consider

$$(1-2^{1-\alpha})\zeta(\alpha) = \sum_{i=1}^{\infty}\frac{(-1)^{i-1}}{i^\alpha}, \quad \alpha > 0,\ \alpha \neq 1, \tag{3.21}$$

where $\zeta(\alpha)$ is the Riemann zeta function. For $0 < \alpha < 1$, (3.21) is justified by analytic continuation.[5] Putting $f(x) = 1/x^\alpha$ in Theorem 3.2, we have

$$s_n - (1-2^{1-\alpha})\zeta(\alpha) \sim (-1)^{n-1}\left(\frac{1}{2}f(n) + \sum_{j=1}^{\infty}\frac{B_{2j}(2^{2j}-1)}{(2j)!}f^{(2j-1)}(n)\right), \tag{3.22}$$

where s_n is the n-th partial sum of (3.21). The first 4 terms of the right-hand side of (3.22) are as follows:

$$\begin{aligned} & s_n - (1-2^{1-\alpha})\zeta(\alpha) \\ &= (-1)^{n-1}\frac{1}{n^\alpha}\left(\frac{1}{2} - \frac{\alpha}{4n} + \frac{\alpha(\alpha+1)(\alpha+2)}{48n^3} - \frac{\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)}{480n^5}\right) \\ &\quad + O(\frac{1}{n^{\alpha+7}}). \end{aligned} \tag{3.23}$$

[5]See, E. C. Titchmarsh, *The theory of the Riemann zeta function*, 2nd ed. (Clarendon Press, Oxford, 1986), p.21.

3.2 Logarithmically convergent series**

The following theorem is useful for obtaining the asymptotic expansions of certain logarithmically convergent series.

Theorem 3.4 *Suppose that a function $f(x)$ has the property (C) in (a, ∞). Let s_n be the n-th partial sum of the series $s = \sum_{i=1}^{\infty} f(i)$. Suppose that both the infinite integral $\int_a^\infty f(x)dx$ and the series s converge. Then*

$$s_n - s = -\int_n^\infty f(x)dx + \frac{1}{2}f(n) + \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!} f^{(2j-1)}(n) + O(f^{(2p)}(n)), \quad \text{as } n \to \infty, \tag{3.24}$$

where B_{2j}'s are the Bernoulli numbers.

Proof. Let $m, n, p \in \mathbb{N}$ with $n > a$. By the Euler-Maclaurin formula (Theorem 1.2), we have

$$\sum_{i=0}^{m} f(n+i) = \int_n^{n+m} f(x)dx + \frac{1}{2}\left(f(n) + f(n+m)\right) + \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!}\left(f^{(2j-1)}(n+m) - f^{(2j-1)}(n)\right) + R_p(n,m) \tag{3.25}$$

where

$$|R_p(n,m)| \le \frac{4e^{2\pi}}{(2\pi)^{2p+1}} |f^{(2p)}(n+m) - f^{(2p)}(n)|. \tag{3.26}$$

Letting $m \to \infty$, the series in the left-hand side of (3.25) converges and $f^{(r)}(n+m) \to 0$ for $r = 0, 1, \ldots$. So we obtain

$$s - s_{n-1} = \int_n^\infty f(x)dx + \frac{1}{2}f(n) - \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!} f^{(2j-1)}(n) + O(f^{(2p)}(n)). \tag{3.27}$$

By (3.27), we obtain (3.24). □

Theorem 3.5 *Let $f(x)$ be a function belonging to $\mathscr{A}^{(\theta)}$ with $\theta < -1$. Let the asymptotic expansion of $f(x)$ be*

$$f(x) \sim x^\theta \sum_{j=0}^{\infty} \frac{a_j}{x^j}, \quad \text{as } x \to \infty, \tag{3.28}$$

**The material in this subsection is taken from the author's paper: N. Osada, Asymptotic expansions and acceleration methods for logarithmically convergent series (in Japanese), *Trans. Inform. Process. Soc. Japan*, 29(1988) No.3, pp.256–261. (= 長田直樹, 対数収束級数の漸近展開と加速法, 情報処理学会論文誌)

where $a_0 \neq 0$. *Assume that both the series* $s = \sum_{i=1}^{\infty} f(i)$ *and the integral* $\int_n^{\infty} f(x)dx$ *converge. Then the* n*-th partial sum* s_n *has the asymptotic expansion of the form*

$$s_n - s \sim n^{\theta+1} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \quad \textit{as } n \to \infty, \tag{3.29}$$

where

$$\begin{aligned} c_0 &= \frac{a_0}{\theta+1}, \quad c_1 = \frac{a_1}{\theta} + \frac{a_0}{2} \\ c_j &= \frac{a_j}{\theta+1-j} + \frac{1}{2}a_{j-1} + \sum_{k=1}^{\lfloor j/2 \rfloor} \frac{B_{2k}}{(2k)!} a_{j-2k} \prod_{l=1}^{2k-1} (\theta - j + 1 + l), \quad (j > 1) \end{aligned} \tag{3.30}$$

where B_{2j}*'s are the Bernoulli numbers.*

Proof. By the Theorem 3.4, we have

$$s_n - s \sim -\int_n^{\infty} f(x)dx + \frac{1}{2}f(n) + \sum_{j=1}^{\infty} \frac{B_{2j}}{(2j)!} f^{(2j-1)}(n), \quad \textit{as } n \to \infty. \tag{3.31}$$

Integrating (3.28) term by term, we have

$$\int_n^{\infty} f(x)dx \sim \sum_{k=0}^{\infty} \frac{-a_k}{\theta+1-k} n^{\theta+1-k}, \tag{3.32}$$

and by the assumption $f(x) \in \mathscr{A}^{(\theta)}$, we have

$$f^{(2j-1)}(n) \sim \sum_{k=0}^{\infty} a_k \left(\prod_{l=1}^{2j-1} (\theta - k + 1 - l) \right) n^{\theta-k-2j+1}. \tag{3.33}$$

By (3.31),(3.32) and (3.33), the coeficient of $n^{\theta+1-j}$ in (3.29) coinsides with c_j in (3.30). This completes the proof. □

Using Theorem 3.4, we can obtain the well-known asymptotic expansion of the Riemann zeta function.

Example 3.4 (The Riemann zeta function) Let us consider

$$s_n = \sum_{i=1}^{n} \frac{1}{i^{\alpha}}, \quad \alpha > 1, \tag{3.34}$$

Taking $f(x) = x^{-\alpha}$ and $s = \zeta(\alpha)$ in Theorem 3.4, we have

$$s - s_n + n^{-\alpha} + \frac{1}{1-\alpha}n^{1-\alpha} \sim \frac{1}{2}n^{-\alpha} + \sum_{j=1}^{\infty} \frac{B_{2j}}{(2j)!} \left(\prod_{l=1}^{2j-1} (\alpha + l - 1) \right) n^{-\alpha-2j+1}, \tag{3.35}$$

thus we obtain

$$s_n - s \sim \frac{1}{1-\alpha}n^{1-\alpha} + \frac{1}{2}n^{-\alpha} - \sum_{j=1}^{\infty} \frac{B_{2j}}{(2j)!}\alpha(\alpha+1)\cdots(\alpha+2j-2)n^{-\alpha-2j+1}. \tag{3.36}$$

In particular, if $\alpha = 2$ then

$$s_n - \frac{\pi^2}{6} = \frac{1}{n}\left(-1 + \frac{1}{2n} - \frac{1}{6n^2} + \frac{1}{30n^4} - \frac{1}{42n^6} + \frac{1}{30n^8} + O(\frac{1}{n^{10}})\right), \quad \text{as } n \to \infty. \tag{3.37}$$

Next example is more complicated.

Example 3.5 (Gustafson[18]) Consider

$$s = \sum_{i=1}^{\infty} \left(i + e^{1/i}\right)^{-\sqrt{2}}. \tag{3.38}$$

Let $f(x) = (x + e^{1/x})^{-\sqrt{2}}$. Using Maclaurin's expansions of e^x and $(1+x)^{-\sqrt{2}}$, we have

$$f(n) \sim n^{-\sqrt{2}} \left(1 + \sum_{j=1}^{\infty} \frac{1}{(j-1)!n^j}\right)^{-\sqrt{2}} \sim n^{-\sqrt{2}} \sum_{j=0}^{\infty} \frac{a_j}{n^j}, \tag{3.39}$$

where the coefficients a_j in (3.39) are given in Table 3.1. By Theorem 3.5, the n-th partial sum s_n has the asymptotic expansion of the form

$$s_n - s \sim n^{1-\sqrt{2}} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{3.40}$$

where $s = 1.71379\,67355\,40301\,48654$ and the coefficients c_j in (3.40) are also given in Table 3.1.

Table 3.1

Coefficients a_j in (3.39) and c_j in (3.40)

j	a_j	c_j
0	1	$-1-\sqrt{2}$
1	$-\sqrt{2}$	$3/2$
2	$1-(1/2)\sqrt{2}$	$2-(25/12)\sqrt{2}$
3	$1-(1/6)\sqrt{2}$	$-1/2+(1/2)\sqrt{2}$
4	$1/12-(5/12)\sqrt{2}$	$227/840+(71/630)\sqrt{2}$
5	$-(23/120)\sqrt{2}$	$-41/168-(37/168)\sqrt{2}$
6	$11/45+(7/60)\sqrt{2}$	$67/4968+(173/6210)\sqrt{2}$
7	$1/72-(1/240)\sqrt{2}$	$3367/12240+(5111/24480)\sqrt{2}$
8	$-19/160-(221/2520)\sqrt{2}$	$-2924867/14212800-(106187/676800)\sqrt{2}$

There are logarithmic terms in the asymptotic expansions of the following series.

Example 3.6 Let us consider

$$s_n = \sum_{i=2}^{n} \frac{\log i}{i^\alpha}, \quad \alpha > 1. \tag{3.41}$$

Since $d/d\alpha(i^{-\alpha}) = -\log i/i^\alpha$, s_n converges to $-\zeta'(\alpha)$, where $-\zeta'(s)$ is the derivative of the Riemann zeta function. Let $f(x) = \log x/x^\alpha$. Then

$$f'(x) = \frac{-\alpha \log x + 1}{x^{\alpha+1}}, \tag{3.42a}$$

$$f''(x) = \frac{\alpha(\alpha+1)\log x - 2\alpha - 1}{x^{\alpha+2}}, \tag{3.42b}$$

$$f'''(x) = \frac{-\alpha(\alpha+1)(\alpha+2)\log x + 3\alpha^2 + 6\alpha + 2}{x^{\alpha+3}}. \tag{3.42c}$$

By Theorem 3.4,

$$s_n + \zeta'(\alpha) = -\int_n^\infty \frac{\log x}{x^\alpha}dx + \frac{\log n}{2n^\alpha} + \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!} f^{(2j-1)}(n) + O(f^{(2p)}(n)), \tag{3.43}$$

thus

$$\begin{aligned} s_n + \zeta'(\alpha) = &\frac{\log n}{(1-\alpha)n^{\alpha-1}} - \frac{1}{(1-\alpha)^2 n^{\alpha-1}} \\ &+ \frac{\log n}{2n^\alpha} + \frac{-\alpha \log n + 1}{12n^{\alpha+1}} - \frac{-\alpha(\alpha+1)(\alpha+2)\log n + 3\alpha^2 + 6\alpha + 2}{720n^{\alpha+3}} \\ &+ O(\frac{\log n}{n^{\alpha+5}}). \end{aligned} \tag{3.44}$$

In particular, if $\alpha = 2$,

$$s_n + \zeta'(2) = -\frac{\log n}{n} - \frac{1}{n} + \frac{\log n}{2n^2} - \frac{\log n}{6n^3} + \frac{1}{12n^3} + \frac{\log n}{30n^5} - \frac{13}{360n^5} + O(\frac{\log n}{n^7}), \quad (3.45)$$

where $-\zeta'(2) = 0.93754\,82543\,15843$.

Example 3.7 Let us consider

$$s_n = \sum_{i=2}^{n} \frac{1}{i(\log i)^\alpha}, \quad \alpha > 1. \quad (3.46)$$

Similarly to Example 3.6, we have

$$\begin{aligned} s_n - s = & \frac{1}{(1-\alpha)(\log n)^{\alpha-1}} + \frac{1}{2n(\log n)^\alpha} - \frac{\log n + \alpha}{12n^2(\log n)^{\alpha+1}} \\ & + \frac{6(\log n)^3 + 11\alpha(\log n)^2 + 6\alpha(\alpha+1)\log n + \alpha(\alpha+1)(\alpha+2)}{720n^4(\log n)^{\alpha+3}} \\ & + O(\frac{1}{n^6(\log n)^\alpha}). \end{aligned} \quad (3.47)$$

In particular, if $\alpha = 2$, then $s = 2.10974\,28012\,36891\,97448$ and

$$\begin{aligned} s_n - s = & -\frac{1}{\log n} + \frac{1}{2n(\log n)^2} - \frac{1}{12n^2(\log n)^2} - \frac{1}{6n^2(\log n)^3} \\ & + \frac{1}{120n^4(\log n)^2} + \frac{11}{360n^4(\log n)^3} + \frac{1}{20n^4(\log n)^4} \\ & + \frac{1}{30n^4(\log n)^5} + O(\frac{1}{n^6(\log n)^2}). \end{aligned} \quad (3.48)$$

This series converges quite slowly. When $\alpha = 2$, by the first 10^{43} terms, we can obtain only two exact digits. P. Henrici[6] computed this series using the Plana summation formula. The asymptotic expansion (3.47) is due to Osada[38].

[6] P. Henrici, *Computational analysis with the HP-25 Pocket Calculator*, (Wiley, New York, 1977).

4 Numerical integration

Infinite integrals and improper integrals usually converge slowly. Such an integral implies slowly convergent sequence or infinite series by a suitable method. In this section, we deal with the convergence of numerical integrals and give some examples.

4.1 Semi-infinite integrals with positive monotonically decreasing integrands

Let $f(x)$ be a continuous function defined in $[a, \infty)$. If the limit

$$\lim_{b\to\infty}\int_a^b f(x)dx \tag{4.1}$$

exists and is finite, then (4.1) is denoted by $\int_a^\infty f(x)dx$ and the semi-infinite integral $\int_a^\infty f(x)dx$ is said to *converge*.

Suppose that an integral $I = \int_a^\infty f(x)dx$ converges. Let (x_n) be an increasing sequence diverging to ∞ with $x_0 = a$. Then I becomes infinite series

$$I = \sum_{n=1}^{\infty}\int_{x_{n-1}}^{x_n} f(x)dx. \tag{4.2}$$

Some semi-infinite integrals converge linearly in this sense.

Example 4.1 Let us now consider

$$\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx, \quad \alpha > 0. \tag{4.3}$$

Let s_n be defined by $s_n = \int_0^n x^{\alpha-1}e^{-x}dx$. Then

$$s_n - \Gamma(\alpha) = n^{\alpha-1}e^{-n}\left(-1 - \frac{\alpha-1}{n} + O(\frac{1}{n^2})\right), \quad \text{as } n \to \infty. \tag{4.4}$$

In particular, if $\alpha \in \mathbb{N}$, then

$$s_n - \Gamma(\alpha) = -n^{\alpha-1}e^{-n}\left(1 + \sum_{j=1}^{\alpha-1}\frac{(\alpha-1)\cdots(\alpha-j)}{n^j}\right). \tag{4.5}$$

(4.4) or (4.5) shows that (s_n) converges linearly to $\Gamma(\alpha)$ with the contraction ratio $1/e$. Especially when $\alpha = 1$, the infinite series $\sum_{n=1}^{\infty}(s_n - s_{n-1})$ becomes a geometric series with the common ratio $1/e$.

Suppose that $f(x)$ has the asymptotic expansion

$$f(x) \sim x^{\theta} \sum_{j=0}^{\infty} \frac{a_j}{x^j}, \qquad \text{as } x \to \infty, \tag{4.6}$$

where $\theta < -1$ and $a_0 \neq 0$. Suppose also that the integral $I = \int_a^{\infty} f(x)dx$ converges. Then $\int_a^x f(t)dt - I$ has the asymptotic expansion

$$\int_a^x f(t)dt - I \sim x^{\theta+1} \sum_{j=0}^{\infty} \frac{a_j}{(\theta - j + 1)x^j}, \qquad \text{as } x \to \infty. \tag{4.7}$$

It follows from (4.7) that $\int_a^x f(t)dt$ converges logarithmically to I.

Example 4.2 Let us consider

$$\frac{\pi}{2} = \int_0^{\infty} \frac{dx}{1+x^2}. \tag{4.8}$$

Integrating

$$\frac{1}{1+t^2} = \sum_{j=1}^{\infty} \frac{(-1)^{j-1}}{t^{2j}}, \quad t > 1, \tag{4.9}$$

term by term, we have

$$\int_x^{\infty} \frac{dt}{1+t^2} = \sum_{j=1}^{\infty} \frac{(-1)^{j-1}}{(2j-1)x^{2j-1}}, \quad x > 1. \tag{4.10}$$

We note that the right-hand side of (4.10) converges uniformly to $\tan^{-1} 1/x = \pi/2 - \tan^{-1} x$ provided that $x > 1$. The equality (4.10) shows that $\int_0^x dt/(1+t^2)$ converges logarithmically to $\pi/2$ as $x \to \infty$.

4.2 Semi-infinite integrals with oscillatory integrands

Let $\phi(x)$ be an oscillating function whose zeros in (a, ∞) are $x_1 < x_2 < \ldots$. Set $x_0 = a$. Let $f(x)$ be a positive continuous function on $[a, \infty)$ such that the semi-infinite integral

$$I = \int_a^{\infty} f(x)\phi(x)dx \tag{4.11}$$

converges. Let I_n be defined by

$$I_n = \int_{x_{n-1}}^{x_n} f(x)\phi(x)dx. \tag{4.12}$$

If I converges, then the infinite series

$$\sum_{n=1}^{\infty} I_n \tag{4.13}$$

is an alternating series which converges to I.

Some integrals become geometric series.

Example 4.3 Consider

$$I = \int_0^{\infty} e^{\alpha x} \sin x dx, \quad \alpha < 0, \tag{4.14}$$

where $I = 1/(1+\alpha^2)$. Then

$$\int_{(n-1)\pi}^{n\pi} e^{\alpha x} \sin x dx = (-1)^{n-1} \frac{e^{\alpha\pi}+1}{1+\alpha^2} e^{(n-1)\alpha\pi}. \tag{4.15}$$

Therefore the infinite series (4.13) is a geometric series with the common ratio $-e^{\alpha\pi}$.

If an integrand is a product of an oscillating function and a rational function converging to 0 as $x \to \infty$, then the infinite series (4.13) usually becomes an alternating series satisfying $I_{n+1}/I_n \to -1$ as $n \to \infty$.

Example 4.4 Let us consider

$$I = \int_0^{\infty} \frac{\sin x}{x^{\alpha}} dx, \quad 0 < \alpha < 2, \tag{4.16}$$

where $I = \pi/(2\Gamma(\alpha)\sin(\alpha\pi/2))$. Set

$$I_n = \int_{(n-1)\pi}^{n\pi} \frac{\sin x}{x^{\alpha}} dx. \tag{4.17}$$

Substituting $t = n\pi - x$, we have

$$\begin{aligned} I_n &= \int_0^{\pi} \frac{\sin(n\pi - t)}{(n\pi - t)^{\alpha}} dt = (-1)^{n-1}(n\pi)^{-\alpha} \int_0^{\pi} \left(1 - \frac{t}{n\pi}\right)^{-\alpha} \sin t dt \\ &\sim (-1)^{n-1}(n\pi)^{-\alpha} \left(2 + \sum_{j=1}^{\infty} \frac{\alpha(\alpha+1)\ldots(\alpha+j-1)}{j!(n\pi)^j} \int_0^{\pi} t^j \sin t dt\right). \end{aligned} \tag{4.18}$$

By Theorem 3.3,

$$\sum_{k=1}^{n} I_k - I \sim (-1)^{n-1} n^{-\alpha} \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{4.19}$$

where c_j are constants.

4.3 Improper integrals with endpoint singularities

Let $f(x)$ be a continuous function in an interval $(a,b]$. Suppose that $\lim_{x\to a+0} f(x) = \pm\infty$ such that

$$\lim_{\epsilon\to+0}\int_{a+\epsilon}^{b} f(x)dx \tag{4.20}$$

exists and is finite. Then the above limit denotes $\displaystyle\int_a^b f(x)dx$. Such an integral is called an *improper integral*, and the endpoint a is called an *integrable singularity*. Similar for the case $\lim_{x\to b-0} f(x) = \pm\infty$.

The Euler-Maclaurin formula is extended to improper integral $\displaystyle\int_a^b f(x)dx$. Let M_n be the compound midpoint rule M_n defined by

$$M_n = h\sum_{i=1}^{n} f(a+(i-\frac{1}{2})h), \quad h = \frac{b-a}{n}. \tag{4.21}$$

Theorem 4.1 (Navot[33]) *Let $f(x)$ be represented as*

$$f(x) = x^{\alpha}g(x), \quad \alpha > -1, \tag{4.22}$$

where $g(x)$ is a C^{2p+1} function in $[0,1]$. Then

$$\begin{aligned} & M_n - \int_0^1 f(x)dx \\ &= \sum_{j=1}^{p+1}\frac{(2^{1-2j}-1)B_{2j}}{(2j)!n^{2j}}f^{(2j-1)}(1) + \sum_{k=0}^{2p+1}\frac{(2^{-\alpha-k}-1)\zeta(-\alpha-k)}{k!n^{\alpha+k+1}}g^{(k)}(0) \\ & + O(n^{-2p-2}), \end{aligned} \tag{4.23}$$

where $\zeta(-\alpha-k)$ is the Riemann zeta function.

Example 4.5 We apply Theorem 4.1 to the integral

$$I = \int_0^1 x^{\alpha}dx, \quad -1 < \alpha < 0, \tag{4.24}$$

where $I = 1/(1+\alpha)$. By $g(x) = 1$, $g^{(j)}(0) = 0$ for $j = 1, 2, \ldots$. Since $f^{(j)}(x) = \alpha(\alpha-1)\ldots(\alpha-j+1)x^{\alpha-j}$, we have

$$\begin{aligned} & M_n - I \\ &= \frac{(2^{-\alpha}-1)\zeta(-\alpha)}{n^{\alpha+1}} + \sum_{j=1}^{p+1}\frac{(2^{1-2j}-1)B_{2j}}{(2j)!n^{2j}}\alpha(\alpha-1)\ldots(\alpha-2j+2) + O(n^{-2p-2}). \end{aligned} \tag{4.25}$$

Putting $h = 1/n$, we have the asymptotic expansion

$$M_n - \int_0^1 x^\alpha dx \sim c_0 h^{1+\alpha} + \sum_{j=1}^{\infty} c_j h^{2j}. \tag{4.26}$$

Theorem 4.2 (Lyness and Ninham[30]) *Let $f(x)$ be represented as*

$$f(x) = x^\alpha (1-x)^\beta g(x), \quad \alpha,\, \beta > -1, \tag{4.27}$$

where $g(x)$ is a C^{p-1} function in $[0,1]$. Let

$$\psi_0(x) = (1-x)^\beta g(x), \quad \psi_1(x) = x^\alpha g(x), \tag{4.28}$$

Then

$$M_n - \int_0^1 f(x)dx = \sum_{j=0}^{p-1} \frac{\psi_0^{(j)}(0)(2^{-\alpha-j}-1)\zeta(-\alpha-j)}{j!n^{\alpha+j+1}} + \sum_{j=0}^{p-1} \frac{(-1)^j \psi_1^{(j)}(1)(2^{-\beta-j}-1)\zeta(-\beta-j)}{j!n^{\beta+j+1}} + O(n^{-p}). \tag{4.29}$$

An integrable singular point in Theorem 4.1 or Theorem 4.2 is called an *algebraic singularity.*

Example 4.6 Let us consider the Beta function

$$B(p,q) = \int_0^1 x^{p-1}(1-x)^{q-1}dx, \quad p,\, q > 0. \tag{4.30}$$

Without loss of generality, we assume $p \geq q > 0$. Applying Theorem 4.2, we have

$$M_n - B(p,q) \sim \sum_{j=0}^{\infty} \left(a_j n^{-p-j} + b_j n^{-q-j}\right), \qquad \text{as } n \to \infty. \tag{4.31}$$

Theorem 4.3 (Lyness and Ninham[30]) *Let $f(x)$ be represented as*

$$f(x) = x^\alpha (1-x)^\beta \log x g(x), \quad \alpha,\, \beta > -1, \tag{4.32}$$

where $g(x)$ is a C^{p-1} function in $[0,1]$. Then

$$M_n - \int_0^1 f(x)dx = \sum_{j=0}^{p-1} \frac{a_j + b_j \log n}{n^{\alpha+j+1}} + \sum_{j=0}^{p-1} \frac{c_j}{n^{\beta+j+1}} + O(n^{-p}). \tag{4.33}$$

An integrable singular point $x = 0$ in Theorem 4.3 is said to be *algebraic-logarithmic.*

Example 4.7 Consider

$$I = \int_0^1 x^\alpha \log x dx, \quad \alpha > -1, \tag{4.34}$$

where $I = -1/(\alpha+1)^2$. Applying Theorem 4.3 with $\beta = 0$, we have

$$M_n - I = \sum_{j=0}^{p-1} \frac{a_j + b_j \log n}{n^{\alpha+j+1}} + \sum_{j=0}^{p-1} \frac{c_j}{n^{j+1}} + O(n^{-p}). \tag{4.35}$$

II. Acceleration methods for scalar sequences

5. Basic concepts

5.1 A sequence transformation, convergence acceleration, and extrapolation

Let $\mathscr{S}$ and $\mathscr{T}$ be sets of real sequences. A mapping $T : \mathscr{S} \to \mathscr{T}$ is called a *sequence transformation*, and we write $(t_n) = T(s_n)$ for $(s_n) \in \mathscr{S}$. Let $\sigma(n)$ denote the greatest index used in the computation of t_n. For a convergent sequence (s_n), T is *regular* if (t_n) converges to the same limit as (s_n). Suppose T is regular for (s_n). T *accelerates the convergence* of (s_n) if

$$\lim_{n\to\infty} \frac{t_n - s}{s_{\sigma(n)} - s} = 0. \tag{5.1}$$

When (s_n) diverges, s is called the *antilimit* (Shanks[51]) of (s_n).

A sequence transformation can often be implemented by various algorithms. For example, the Aitken δ^2 process

$$t_n = s_n - \frac{(s_{n+1} - s_n)^2}{s_{n+2} - 2s_{n+1} + s_n} \tag{5.2}$$

can be represented as

$$= s_{n+1} - \frac{(s_{n+1} - s_n)(s_{n+2} - s_{n+1})}{s_{n+2} - 2s_{n+1} + s_n}, \tag{5.3}$$

$$= s_{n+2} - \frac{(s_{n+2} - s_{n+1})^2}{s_{n+2} - 2s_{n+1} + s_n}, \tag{5.4}$$

$$= \frac{s_n s_{n+2} - s_{n+1}^2}{s_{n+2} - 2s_{n+1} + s_n}, \tag{5.5}$$

$$= \frac{\begin{vmatrix} s_n & s_{n+1} \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix}}, \tag{5.6}$$

where Δ stands for the forward difference, i.e. $\Delta s_n = s_{n+1} - s_n$. The algorithm (5.6) coincides with Shanks' e_1 tansformation. All algorithms (5.2) to (5.6) are equivalent in theory but are different in numerical computation; e.g., the number of arithmetic operations or rounding errors.

Let T be a sequence transformation satisfying (5.1). Either T or an algorithm for T is called a *convergence acceleration method*, or an *acceleration method* for short, or a *speed-up method.* A method for estimating the limit or the antilimit of a sequence (s_n) is called an *extrapolation method.*

The name 'extrapolation' is explained as follows. We take an 'extrapolation function' $f(n)$ with $k+1$ unknown constants. Under suitable conditions, by solving the system of equations

$$f(n+i) = s_{n+i}, \quad i = 0, \ldots, k, \tag{5.7}$$

we can determine $k+1$ unknown constants. Then, letting n tend to infinity, we obtain the approximation to s as $s = \lim_{n\to\infty} f(n)$. Thus the above process is called *extrapolation to the limit* or *extrapolation* for short.

Usually, 'a sequence transformation', 'convergence acceleration', and 'extrapolation' are not distinguished.

5.2 A classification of convergence acceleration methods

We divide convergence acceleration methods into three cases according to given knowledge of the sequence.

I. An explicit knowledge of the sequence generator is given. Then we can usually accelerate the convergence of the sequence using such a knowledge. For example, if the n-th term of an infinite series is explicitly given as an analytic function $f(n)$, we can accelerate the series by the Euler-Maclaurin summation formula, the Plana summation formula, or the Bickley and Miller method mentioned in Introduction.

II. The sequence has the asymptotic expansion with respect to a known asymptotic scale. Then we can accelerate by using the scale. Typical examples of this case are the Richardson extrapolation and the E-algorithm. The modified Aitken δ^2 process and the generalized ρ-algorithm are also of this case.

III. Neither an explicit knowledge of the sequence generator nor an asymptotic scale is known. Then we have to estimate the limit using consecutive terms of the sequence $s_n, s_{n+1}, \ldots, s_{n+k}$. Almost all famous sequence transformations such as the iterated Aitken δ^2 process, the ϵ-algorithm, Lubkin's W transformation, and the ρ-algorithm are of this case.

In this paper we will deal with the above cases II and III.

6. The E-algorithm

Many sequence transformations can be represented as a ratio of two determinants. The E-algorithm is a recursive algorithm for such transformations and a quite general method.

6.1 The derivation of the E-algorithm

Suppose that a sequence (s_n) with the limit or the antilimit s satisfies

$$s_n = s + \sum_{j=1}^{k} c_j g_j(n). \tag{6.1}$$

Here $(g_j(n))$ is a given auxiliary double sequence which can depend on the sequence (s_n) whereas $c_1, \dots, c_k$ are constants independent of (s_n) and n. The auxiliary double sequence is not necessarily an asymptotic scale.

Solving the system of linear equations

$$s_{n+i} = T_k^{(n)} + c_1 g_1(n+i) + \cdots + c_k g_k(n+i), \quad i = 0, \dots, k, \tag{6.2}$$

for the unknown $T_k^{(n)}$ by Cramer's rule, we obtain

$$T_k^{(n)} = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+k) \\ & & \cdots & \\ g_k(n) & g_k(n+1) & \cdots & g_k(n+k) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ g_1(n) & g_1(n+1) & \cdots & g_1(n+k) \\ & & \cdots & \\ g_k(n) & g_k(n+1) & \cdots & g_k(n+k) \end{vmatrix}}. \tag{6.3}$$

If (s_n) satisfies

$$s_n \sim s + \sum_{j=1}^{\infty} c_j g_j(n), \tag{6.4}$$

where $(g_j(n))$ is a given asymptotic scale, then $T_k^{(n)}$ may be expected to be a good approximation to s.

Many well known sequence transformations such as the Richardson extrapolation, the Shanks transformation, Levin's transformations and so on can be represented as (6.3). In 1975, C. Schneider[50] gave a recursive algorithm for $T_k^{(n)}$ in (6.3). Using different techniques, the same algorithm was later derived by T. Håvie[20], published in 1979, and then by C. Brezinski[10], published in 1980, who called it the *E-algorithm*.

The two dimensional array $E_k^{(n)}$ and the auxiliary three dimensional array $g_{k,j}^{(n)}$ are defined as follows[7]:

$$E_0^{(n)} = s_n, \quad n = 0, 1, \ldots, \tag{6.5a}$$

$$g_{0,j}^{(n)} = g_j(n), \quad j = 1, 2, \ldots; n = 0, 1, \ldots. \tag{6.6a}$$

For $k = 1, 2, \ldots$ and $n = 0, 1, \ldots$

$$E_k^{(n)} = \frac{E_{k-1}^{(n)} g_{k-1,k}^{(n+1)} - E_{k-1}^{(n+1)} g_{k-1,k}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \tag{6.5b}$$

$$g_{k,j}^{(n)} = \frac{g_{k-1,j}^{(n)} g_{k-1,k}^{(n+1)} - g_{k-1,j}^{(n+1)} g_{k-1,k}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \quad j = k+1, k+2, \ldots. \tag{6.6b}$$

The equality (6.5b) is called the *main rule* and (6.6b) is called the *auxiliary rule*.

The following theorem is fundamental.

Theorem 6.1 (Brezinski[10]) *Let $G_{k,j}^{(n)}$, $N_k^{(n)}$, and $D_k^{(n)}$ be defined by*

$$G_{k,j}^{(n)} = \begin{vmatrix} g_j(n) & \cdots & g_j(n+k) \\ g_1(n) & \cdots & g_1(n+k) \\ & \cdots & \\ g_k(n) & \cdots & g_k(n+k) \end{vmatrix}, \tag{6.7}$$

$$N_k^{(n)} = \begin{vmatrix} s_n & \cdots & s_{n+k} \\ g_1(n) & \cdots & g_1(n+k) \\ & \cdots & \\ g_k(n) & \cdots & g_k(n+k) \end{vmatrix}, \quad D_k^{(n)} = \begin{vmatrix} 1 & \cdots & 1 \\ g_1(n) & \cdots & g_1(n+k) \\ & \cdots & \\ g_k(n) & \cdots & g_k(n+k) \end{vmatrix}, \tag{6.8}$$

respectively. Then for $n = 0, 1, \ldots; k = 1, 2, \ldots,$ we have

$$g_{k,j}^{(n)} = G_{k,j}^{(n)} / D_k^{(n)}, \quad j > k, \tag{6.9}$$

$$E_k^{(n)} = N_k^{(n)} / D_k^{(n)}, \quad n = 0, 1, \ldots; k = 1, 2, \ldots. \tag{6.10}$$

[7]When the sequence (s_n) is defined in $n \geq 1$, substitute '$n = 1, 2, \ldots$' for '$n = 0, 1, \ldots$' in the rest of this chapter.

Since the expressions (6.5b) and (6.6b) are prone to round-off-error effects, it is better to compute $E_k^{(n)}$ and $g_{k,j}^{(n)}$ from the following equivalent expressions.

$$E_k^{(n)} = E_{k-1}^{(n)} - g_{k-1,k}^{(n)} \frac{E_{k-1}^{(n+1)} - E_{k-1}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \tag{6.11}$$

and

$$g_{k,j}^{(n)} = g_{k-1,j}^{(n)} - g_{k-1,k}^{(n)} \frac{g_{k-1,j}^{(n+1)} - g_{k-1,j}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}}, \tag{6.12}$$

respectively.

6.2 The acceleration theorems of the *E*-algorithm

When $(g_j(n))$ is an asymptotic scale, the following theorem is valid.

Theorem 6.2 (Sidi[53]) *Suppose the following four conditions are satisfied.*

(i) $\lim_{n\to\infty} s_n = s$,

(ii) *For any* j, *there exists* $b_j \neq 1$ *such that* $\lim_{n\to\infty} g_j(n+1)/g_j(n) = b_j$, *and* $b_i \neq b_j$ *if* $i \neq j$.

(iii) *For any* j, $\lim_{n\to\infty} g_{j+1}(n+1)/g_j(n) = 0$,

(iv) s_n *has the asymptotic expansion of the form*

$$s_n \sim s + \sum_{j=1}^{\infty} c_j g_j(n) \quad as\ n \to \infty. \tag{6.13}$$

Then, for any k,

$$E_k^{(n)} - s \sim c_{k+1} \left(\prod_{j=1}^{k} \frac{b_{k+1} - b_j}{1 - b_j} \right) g_{k+1}(n) \quad as\ n \to \infty \tag{6.14}$$

and

$$\frac{E_k^{(n)} - s}{E_{k-1}^{(n)} - s} = O\left(\frac{g_{k+1}(n)}{g_k(n)} \right) \quad as\ n \to \infty. \tag{6.15}$$

A logarithmic scale does not satisfy the assumption (ii) of Theorem 6.2, but satisfies the assumptions of the next theorem.

Theorem 6.3 (Matos and Prévost[31]) *If the conditions (i)(iii)(iv) of theorem 6.2 are satisfied, and if for any j, p and any $n \geq N$,*

$$\begin{vmatrix} g_{j+p}(n) & \cdots & g_j(n) \\ \vdots & & \vdots \\ g_{j+p}(n+p) & \cdots & g_j(n+p) \end{vmatrix} \geq 0, \tag{6.16}$$

then for any $k \geq 0$,

$$\lim_{n\to\infty} \frac{E_{k+1}^{(n)} - s}{E_k^{(n)} - s} = 0. \tag{6.17}$$

The above theorem is important because the following examples satisfy the assumption (6.16). (Brezinski et al.[11, p.69])

(1) Let $(g(n))$ be a logarithmic totally monotone sequence, i.e. $\lim_{n\to\infty} g(n+1)/g(n) = 1$ and $(-1)^k \Delta^k g(n) \geq 0 \, \forall k$. Let $(g_j(n))$ be defined by $g_1(n) = g(n), g_j(n) = (-1)^j \Delta^j g(n)$ $(j > 1)$.

(2) $g_j(n) = x_n^{\alpha_j}$ with $1 > x_1 > x_2 > \cdots > 0$ and $0 < \alpha_1 < \alpha_2 < \ldots$.

(3) $g_j(n) = \lambda_j^n$ with $1 > \lambda_1 > \lambda_2 > \cdots > 0$.

(4) $g_j(n) = 1/((n+1)^{\alpha_j}(\log(n+2))^{\beta_j})$ with $0 < \alpha_1 \leq \alpha_2 \leq \ldots$ and $\beta_j < \beta_{j+1}$ if $\alpha_j = \alpha_{j+1}$.

7. The Richardson extrapolation

The Richardson extrapolation, as well as the Aitken δ^2 process, is the most available extrapolation method in numerical computation. Nowadays, the basic idea of the Richardson extrapolation eliminating the first several terms in an asymptotic expansion is used for obtaining various sequence transformations.

7.1 The birth of the Richardson extrapolation

Similar to the Aitken δ^2 process, the Richardson extrapolation originated in the process of computing π.

Let T_n be the perimeter of the regular polygon with n sides inscribed in a circle of diameter C. Let U_n be the perimeter of the regular polygon with n sides circumscribing a circle of diameter C. C. Huygens proved geometrically[8] in his *De Circuli Magnitudine Inventa*, published in 1654, that

$$T_{2n} + \frac{1}{3}\left(T_{2n} - T_n\right) < C < \frac{2}{3}T_n + \frac{1}{3}U_n. \tag{7.1}$$

The left-hand side of (7.1) is the Richardson extrapolation.

The iterative application of the Richardson extrapolation was first used by Katahiro Takebe 建部賢弘 (or Kenkō Takebe, 1664-1739), a disciple of T. Seki. Let s_n be the perimeter of the regular polygon with 2^n sides inscribed in a circle of diameter one and let $\sigma_n = s_n^2$. Takebe used the Richardson extrapolation iteratively in Tetsujyutsu Sankei Chapter 11 Tan'ensū 綴術算経 探圓数 第十一, published in 1722. He presumed that $(\sigma_{n+2} - \sigma_{n+1})/(\sigma_{n+1} - \sigma_n)$ converged to 1/4. By $\sum_{i=1}^{\infty}(1/4)^i = 1/3$, he constructed

$$\sigma_1^{(n)} = \sigma_{n+1} + \frac{1}{3}\left(\sigma_{n+1} - \sigma_n\right), \quad n = 2, 3, \ldots. \tag{7.2}$$

Then he presumed that $(\sigma_1^{(n+2)} - \sigma_1^{(n+1)})/(\sigma_1^{(n+1)} - \sigma_1^{(n)})$ converged to 1/16, and he constructed

$$\sigma_2^{(n)} = \sigma_1^{(n+1)} + \frac{1}{15}\left(\sigma_1^{(n+1)} - \sigma_1^{(n)}\right), \quad n = 2, 3, \ldots. \tag{7.3}$$

Similarly, he constructed

$$\sigma_k^{(n)} = \sigma_{k-1}^{(n+1)} + \frac{1}{2^{2k} - 1}\left(\sigma_{k-1}^{(n+1)} - \sigma_{k-1}^{(n)}\right), \quad n = 2, 3, \ldots; k = 2, 3, \ldots. \tag{7.4}$$

[8] A. Hirayama, *History of circle ratio* (in Japanese), (Osaka Kyoiku Tosho, 1980), pp.75-76. (=平山諦, 円周率の歴史 (改訂新版), 大阪教育図書)

From $\sigma_2, \ldots, \sigma_{10}$, he obtained π to 41 exact digits,[9] i.e.

$$\sqrt{\sigma_8^{(2)}} = 3.14159\ 26535\ 89793\ 23846\ 26433\ 83279\ 50288\ 41971\ \mathit{2} \tag{7.5}$$

7.2 The derivation of the Richardson extrapolation

Let $T(h)$ be an approximation depending on a parameter $h > 0$ to a fixed value s. Suppose that $T(h)$ satisfies the asymptotic formula

$$T(h) = s + \sum_{j=1}^{m} c_j h^{2j} + O(h^{2m+2}), \qquad \text{as } h \to +0, \tag{7.6}$$

where $c_1, c_2, \ldots$ are unknown constants independent of h. For given $h_0 > h_1 > 0$, by computing $h_1^2 T(h_0) - h_0^2 T(h_1)$, we obtain

$$\frac{h_1^2 T(h_0) - h_0^2 T(h_1)}{h_1^2 - h_0^2} = s + \sum_{j=2}^{m} c_j \frac{h_1^2 h_0^{2j} - h_0^2 h_1^{2j}}{h_1^2 - h_0^2} + O(h_0^{2m+2}). \tag{7.7}$$

We define $T_1(h_0, h_1)$ by the left-hand side of (7.7). Then $T_1(h_0, h_1)$ is a better approximation to s than $T(h_1)$ provided that h_0 is sufficiently small. When we set $h_0 = h$ and $h_1 = h/2$, (7.7) becomes

$$T_1(h, h/2) = s + \sum_{j=2}^{m} c'_j h^{2j} + O(h^{2m+2}), \tag{7.8}$$

where

$$T_1(h, h/2) = \frac{T(h) - 2^2 T(h/2)}{1 - 2^2}, \tag{7.9}$$

and $c'_j = c_j(1 - 2^{2-2j})/(1 - 2^2)$.

Similarly, we define $T_2(h, h/2, h/4)$ by

$$T_2(h, h/2, h/4) = \frac{T_1(h, h/2) - 2^4 T_1(h/2, h/4)}{1 - 2^4}, \tag{7.10}$$

then we have

$$T_2(h, h/2, h/4) = s + \sum_{j=3}^{m} c''_j h^{2j} + O(h^{2m+2}), \tag{7.11}$$

where $c''_j = c'_j(1 - 2^{4-2j})/(1 - 2^4)$. By (7.6),(7.8) and (7.11), when h is sufficiently small, $T_2(h, h/2, h/4)$ is better than both $T_1(h/2, h/4)$ and $T(h/4)$.

[9] M. Fujiwara, op. cit., pp.296-298.

The above process was considered by L. F. Richardson in 1927. The formula $T_1(h_0, h_1)$ was named h^2-*extrapolation* and $T_2(h_0, h_1, h_2)$ was named h^4-*extrapolation*. Such extrapolations were also called the *deffered approach to the limit*. Nowadays this process is called the *Richardson extrapolation*.

Extending of (7.9) and (7.10), we define the two dimensional array $(T_k^{(n)})$ by

$$T_0^{(n)} = T(2^{-n}h) \qquad n = 0, 1, \ldots, \tag{7.12a}$$

$$T_k^{(n)} = \frac{T_{k-1}^{(n)} - 2^{2k} T_{k-1}^{(n+1)}}{1 - 2^{2k}} \qquad n = 0, 1, \ldots; k = 1, 2, \ldots, \tag{7.12b}$$

where h is an initial mesh size. The equation (7.12b) becomes

$$T_k^{(n)} = T_{k-1}^{(n+1)} + \frac{1}{2^{2k} - 1}\left(T_{k-1}^{(n+1)} - T_{k-1}^{(n)}\right). \tag{7.13}$$

Takebe's algorithm (7.4) coincides with (7.13). It is clear that $T_1^{(n)} = T_1(h/2^n, h/2^{n+1})$ and $T_2^{(n)} = T_2(h/2^n, h/2^{n+1}, h/2^{n+2})$. The following table is called the T-*table*:

$$\begin{array}{ccccc} T_0^{(0)} & & & & \\ T_0^{(1)} & T_1^{(0)} & & & \\ T_0^{(2)} & T_1^{(1)} & T_2^{(0)} & & \\ T_0^{(3)} & T_1^{(2)} & T_2^{(1)} & T_3^{(0)} & \\ \vdots & & & & \ddots \end{array} \tag{7.14}$$

Using mathematical induction on k, we have

$$T_k^{(n-k)} = s + \sum_{j=k+1}^{m} c_j \left(\prod_{i=1}^{k} \frac{1 - 2^{2i-2j}}{1 - 2^{2i}}\right) h^{2j} 2^{-2j(n-k)} + O(2^{-2(m+1)(n-k)}), \tag{7.15}$$

where h is an initial mesh size. In particular, we have an asymptotic approximation to $T_k^{(n-k)} - s$:

$$T_k^{(n-k)} - s \sim c_{k+1} \left(\prod_{i=1}^{k} \frac{1 - 2^{2i-(2k+2)}}{1 - 2^{2i}}\right) h^{2k+2} 2^{-(n-k)(2k+2)}. \tag{7.16}$$

Numerical quadrature — an example. Suppose that a function $f(x)$ is of C^∞ in a closed interval $[a, b]$. Then the trapezoidal rule

$$T_n = T(h) = h\left(\frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(a + ih) + \frac{1}{2} f(b)\right), \quad h = \frac{b-a}{n} \tag{7.17}$$

and the midpoint rule

$$M_n = T(h) = h\sum_{i=1}^{n} f(a+(i-\frac{1}{2})h), \quad h = \frac{b-a}{n} \tag{7.18}$$

have the asymptotic expansion of the form

$$T(h) - \int_a^b f(x)dx \sim \sum_{j=1}^{\infty} c_j h^{2j}, \text{ as } h \to +0. \tag{7.19}$$

Thus we can apply the Richardson extrapolation to $T(h)$. This method using the trapezoidal rule is called the *Romberg quadrature* and its algorithm is as follows:

$$T_0^{(n)} = T_{2^n} \qquad n = 0, 1, \ldots, \tag{7.20a}$$

$$T_k^{(n)} = \frac{T_{k-1}^{(n)} - 2^{2k}T_{k-1}^{(n+1)}}{1-2^{2k}} \qquad n = 0, 1, \ldots; k = 1, 2, \ldots, \tag{7.20b}$$

where T_{2^n} is the 2^n panels trapezoidal rule.

By the Euler-Maclaurin formula and (7.16), we have

$$T_k^{(n)} - \int_a^b f(x)dx$$
$$\sim \frac{B_{2k+2}}{(2k+2)!}\left(f^{(2k+1)}(b) - f^{(2k+1)}(a)\right)(b-a)^{2k+2}2^{-n(2k+2)}\left(\prod_{i=1}^{k}\frac{1-2^{2i-2k-2}}{1-2^{2i}}\right). \tag{7.21}$$

Now we illustrate by a numerical example. Numerical computations in this section were carried out on the NEC personal computer PC-9801DA in double precision with approximately 16 digits. Throughout this section, the number of functional evaluations will be abbreviated to "f.e.".

Example 7.1 We apply the Romberg quadrature on a proper integral

$$I = \int_0^1 e^x dx = e - 1 = 1.71828\,18284\,59045. \tag{7.22}$$

We give errors of the T-table and the number of functional evaluations in Table 7.1. The Romberg quadrature is quite efficient for such proper integrals.

Table 7.1

The errors of the T-table by the Romberg quadrature

n	f.e.	$T_0^{(n)} - I$	$T_1^{(n-1)} - I$	$T_2^{(n-2)} - I$	$T_3^{(n-3)} - I$	$T_4^{(n-4)} - I$
0	2	1.41×10^{-1}				
1	3	3.56×10^{-2}	5.79×10^{-4}			
2	5	8.94×10^{-3}	3.70×10^{-5}	8.59×10^{-7}		
3	9	2.24×10^{-3}	2.23×10^{-6}	1.38×10^{-8}	3.35×10^{-10}	
4	17	5.59×10^{-4}	1.46×10^{-7}	2.16×10^{-10}	1.34×10^{-12}	3.32×10^{-14}

By (7.21), the asymptotic error approximation of $T_4^{(0)}$ is

$$\frac{B_{10}}{10!}\frac{1-2^{2-10}}{1-2^2}\frac{1-2^{4-10}}{1-2^4}\frac{1-2^{6-10}}{1-2^6}\frac{1-2^{8-10}}{1-2^8}(e-1) = 3.42 \times 10^{-14}, \tag{7.23}$$

which is close to $T_4^{(0)} - I$ in Table 7.1.

7.3 Generalizations of the Richardson extrapolation

Nowadays, the Richardson extrapolation (7.4) or (7.13) is extended to three types of sequences as follows:

I. Polynomial extrapolations.

$$s_n \sim s + \sum_{j=1}^{\infty} c_j x_n^j, \tag{7.24}$$

where (x_n) is a known auxiliary sequence[10] whereas c_j's are unknown constants.

II. Extended polynomial extrapolations.

$$s_n \sim s + \sum_{j=1}^{\infty} c_j x_n^{\alpha_j}, \tag{7.25}$$

where α_j's are known constants, c_j's are unknown constants, and (x_n) is a known particular auxiliary sequence.

In particular, when $x_n = \lambda^{-n}$ and $\alpha_j = j$, then (7.25) becomes

$$s_n \sim s + \sum_{j=1}^{\infty} c_j (\lambda^{-j})^n. \tag{7.26}$$

[10]In this paper, x_i^j means either j th power of a scalar x_i or the i th component of a vector x^j. In (7.24), $x_n^j = (x_n)^j$.

This is a special case of

$$s_n \sim s + \sum_{j=1}^{\infty} c_j \lambda_j^n, \tag{7.27}$$

where λ_j's are known constants, c_j's are unknown constants.

III. General extrapolations.

$$s_n \sim s + \sum_{j=1}^{\infty} c_j g_j(n), \tag{7.28}$$

where $(g_j(n))$ is a known asymptotic scale whereas c_j's are unknown constants.

7.3.1 Polynomial extrapolation

Suppose that a sequence (s_n) satisfies

$$s_n \sim s + \sum_{j=1}^{\infty} c_j x_n^j, \tag{7.29}$$

where (x_n) is a known auxiliary sequence and c_j's are unknown constants. Solving the system of equations

$$s_{n+i} = T_k^{(n)} + \sum_{j=1}^{k} c_j x_{n+i}^j, \quad i = 0, \ldots, k, \tag{7.30}$$

for the unknown $T_k^{(n)}$ by Cramer's rule, we have

$$T_k^{(n)} = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ x_n & x_{n+1} & \cdots & x_{n+k} \\ & & \cdots & \\ x_n^k & x_{n+1}^k & \cdots & x_{n+k}^k \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ x_n & x_{n+1} & \cdots & x_{n+k} \\ & & \cdots & \\ x_n^k & x_{n+1}^k & \cdots & x_{n+k}^k \end{vmatrix}}. \tag{7.31}$$

By Theorem 6.1 and using the Vandermonde determinants,

$$T_k^{(n)} = T_{k-1}^{(n+1)} + \frac{1}{x_n/x_{n+k} - 1}\left(T_{k-1}^{(n+1)} - T_{k-1}^{(n)}\right), \quad k = 1, 2, \ldots; n = 0, 1, \ldots. \tag{7.32}$$

In Takebe's algorithm (7.4) or in the Romberg scheme (7.20), $x_n = 2^{-2n}$.

Example 7.2 We apply the Richardson extrapolation (7.32) with $x_n = 1/n$ to the logarithmically convergent series

$$s_n = \sum_{i=1}^{n} \frac{1}{i^2}. \tag{7.33}$$

As is well known, (s_n) converges to $s = \zeta(2) = \pi^2/6$ very slowly. More precisely, by Example 3.4, we have

$$s_n - \frac{\pi^2}{6} \sim -\frac{1}{n} + \frac{1}{2n^2} - \frac{1}{6n^3} + \frac{1}{30n^5} - \frac{1}{42n^7} + \frac{1}{30n^9} - \frac{5}{66n^{11}} + \dots. \tag{7.34}$$

We give errors of $T_k^{(n-k)} - \pi^2/6$ in Table 7.2. By the first 12 terms, we obtain 10 significant digits. And, between $n = 1$ to 16, the best result is $T_{11}^{(2)} - s = -2.78 \times 10^{-13}$. This method tends to be affected by cancellation of significant digits.

Table 7.2

The errors of the T-table by (7.32)

n	$T_0^{(n)} - s$	$T_1^{(n-1)} - s$	$T_2^{(n-2)} - s$	$T_3^{(n-3)} - s$	$T_4^{(n-4)} - s$
1	-6.45×10^{-1}				
2	-3.95×10^{-1}	-1.45×10^{-1}			
3	-2.84×10^{-1}	-6.16×10^{-2}	-1.99×10^{-2}		
4	-2.21×10^{-1}	-3.38×10^{-2}	-6.05×10^{-3}	-1.42×10^{-3}	
5	-1.81×10^{-1}	-2.13×10^{-2}	-2.57×10^{-3}	-2.58×10^{-4}	3.12×10^{-5}
6	-1.54×10^{-1}	-1.47×10^{-2}	-1.32×10^{-3}	-7.30×10^{-5}	1.96×10^{-5}
7	-1.33×10^{-1}	-1.07×10^{-2}	-7.67×10^{-4}	-2.67×10^{-5}	8.06×10^{-6}
8	-1.18×10^{-1}	-8.14×10^{-3}	-4.84×10^{-4}	-1.15×10^{-5}	3.57×10^{-6}
9	-1.05×10^{-1}	-6.40×10^{-3}	-3.25×10^{-4}	-5.64×10^{-6}	1.74×10^{-6}
10	-9.52×10^{-2}	-5.17×10^{-3}	-2.28×10^{-4}	-3.01×10^{-6}	9.24×10^{-7}
11	-8.69×10^{-2}	-4.26×10^{-3}	-1.66×10^{-4}	-1.73×10^{-6}	5.24×10^{-7}
12	-8.00×10^{-2}	-3.57×10^{-3}	-1.25×10^{-4}	-1.05×10^{-6}	3.14×10^{-7}

n	$T_5^{(n-5)} - s$	$T_6^{(n-6)} - s$	$T_7^{(n-7)} - s$	$T_8^{(n-8)} - s$	$T_9^{(n-9)} - s$
6	1.73×10^{-5}				
7	3.43×10^{-6}	1.12×10^{-6}			
8	8.82×10^{-7}	3.18×10^{-8}	-1.23×10^{-7}		
9	2.80×10^{-7}	-2.14×10^{-8}	-3.65×10^{-8}	-2.57×10^{-8}	
10	1.04×10^{-7}	-1.33×10^{-8}	-9.79×10^{-9}	-3.11×10^{-9}	-6.01×10^{-10}
11	4.36×10^{-8}	-6.70×10^{-9}	-2.95×10^{-9}	-3.86×10^{-10}	2.18×10^{-10}
12	2.01×10^{-8}	-3.38×10^{-9}	-1.01×10^{-9}	-3.30×10^{-11}	8.48×10^{-11}

When $x_n = 1/n$, the subsequence $s'_n = s_{2^n}$ has the asymptotic expansion of the form

$$s'_n - s \sim \sum_{j=1}^{\infty} c_j (2^{-n})^j. \tag{7.35}$$

Then we can apply the Richardson extrapolation with $x_n = 2^{-n}$. This method requires many terms but usually gives high accuracy.

For a sequence (s_n) satisfying (7.27), the Richardson extrapolation is as follows.

$$T_0^{(n)} = s_n \tag{7.36a}$$

$$T_k^{(n)} = T_{k-1}^{(n+1)} + \frac{\lambda_k}{1-\lambda_k}\left(T_{k-1}^{(n+1)} - T_{k-1}^{(n)}\right). \tag{7.36b}$$

Similar to (7.16), we have an asymptotic approximation to $T_k^{(n-k)} - s$:

$$T_k^{(n-k)} - s \sim c_{k+1}\left(\prod_{i=1}^{k} \frac{\lambda_{i+1}-\lambda_i}{1-\lambda_i}\right)\lambda_{k+1}^n. \tag{7.37}$$

Example 7.3 We apply the Richardson extrapolation with $x_n = 2^{-n}$ to $\zeta(2)$ once more. And we give errors of $T_k^{(n-k)} - \pi^2/6$ in Table 7.3.

Table 7.3

The errors of the T-table by the Richardson extrapolation with $x_n = 2^{-n}$

n	terms	$T_0^{(n)} - s$	$T_1^{(n-1)} - s$	$T_2^{(n-2)} - s$	$T_3^{(n-3)} - s$	$T_4^{(n-4)} - s$
0	1	-6.45×10^{-1}				
1	2	-3.95×10^{-1}	-1.45×10^{-1}			
2	4	-2.21×10^{-1}	-4.77×10^{-2}	-1.53×10^{-2}		
3	8	-1.18×10^{-1}	-1.37×10^{-2}	-2.36×10^{-3}	-5.16×10^{-4}	
4	16	-6.06×10^{-2}	-3.66×10^{-3}	-3.17×10^{-4}	-2.46×10^{-5}	-8.76×10^{-6}
5	32	-3.08×10^{-2}	-9.46×10^{-4}	-4.04×10^{-5}	-8.97×10^{-7}	-1.32×10^{-7}
6	64	-1.55×10^{-2}	-2.40×10^{-4}	-5.08×10^{-6}	-2.93×10^{-8}	-1.34×10^{-9}
7	128	-7.78×10^{-3}	-6.06×10^{-5}	-6.36×10^{-7}	-9.28×10^{-10}	-1.14×10^{-11}
8	256	-3.90×10^{-3}	-1.52×10^{-5}	-7.95×10^{-8}	-2.91×10^{-11}	-9.06×10^{-14}

n	terms	$T_5^{(n-5)} - s$	$T_6^{(n-6)} - s$	$T_7^{(n-7)} - s$	$T_8^{(n-8)} - s$
5	32	-6.44×10^{-8}			
6	64	-3.10×10^{-10}	-1.84×10^{-10}		
7	128	-8.87×10^{-13}	-2.82×10^{-13}	-1.92×10^{-13}	
8	256	-1.94×10^{-15}	-2.22×10^{-16}	-8.33×10^{-17}	-5.55×10^{-17}

Using 8 partial sums $s_1, s_2, s_4, s_8, s_{16}, s_{32}, s_{128}$, and s_{256}, we obtain 16 significant digits. This method is hardly affected by rouding errors such as cancellation of significant digits.

7.3.2 Extended polynomial extrapolation

Let us consider a sequence satisfying

$$s_n \sim s + \sum_{j=1}^{\infty} c_j n^{-\alpha_j}, \tag{7.38}$$

where α_j's are known constants whereas c_j's are unknown constants. The Richardson extrapolation for (7.38) is defined by

$$T_0^{(n)} = s_{2^n} \qquad n = 0, 1, \ldots, \tag{7.39a}$$

$$T_k^{(n)} = T_{k-1}^{(n+1)} + \frac{1}{2^{\alpha_k} - 1}\left(T_{k-1}^{(n+1)} - T_{k-1}^{(n)}\right). \qquad n = 0, 1, \ldots; k = 1, 2, \ldots. \tag{7.39b}$$

If we set $h = 1/n$ and $T(h) = s_n$, then (7.38) becomes

$$T(h) \sim s + \sum_{j=1}^{\infty} c_j h^{\alpha_j}. \tag{7.40}$$

Similar to (7.39), the Richardson extrapolation for (7.40) is defined by

$$T_0^{(n)} = T(2^{-n}h) \qquad n = 0, 1, \ldots, \tag{7.41a}$$

$$T_k^{(n)} = T_{k-1}^{(n+1)} + \frac{1}{2^{\alpha_k} - 1}\left(T_{k-1}^{(n+1)} - T_{k-1}^{(n)}\right), \qquad n = 0, 1, \ldots; k = 1, 2, \ldots, \tag{7.41b}$$

where h is an initial mesh size.

Example 7.4 We apply the Richardson extrapolation (7.41) to the improper integral

$$I = \int_0^1 \frac{dx}{\sqrt{x}} = 2. \tag{7.42}$$

Let M_n be the n panels midpoint rule for (7.42). By Example 4.5, we have

$$M_n - 2 \sim c_0 h^{1/2} + \sum_{j=1}^{\infty} c_j h^{2j}. \tag{7.43}$$

We show the T-table in Table 7.4. Using 63 functional evaluations, we obtain $T_5^{(0)} = 1.99999\,9999\mathit{3}\,\mathit{25}$, whose error is -6.75×10^{-10}.

Table 7.4

Applying the Richardson extrapolation (7.41) to $\int_0^1 \frac{dx}{\sqrt{x}}$

n	f.e.	$T_0^{(n)}$	$T_1^{(n-1)}$	$T_2^{(n-2)}$	$T_3^{(n-3)}$	$T_4^{(n-4)}$	$T_5^{(n-5)}$
0	1	1.41					
1	3	1.57	1.971				
2	7	1.69	1.9921	1.99914			
3	15	1.78	1.9979	1.999930	1.999983		
4	31	1.84	1.99949	1.9999952	1.99999957	1.99999983	
5	63	1.89	1.99987	1.99999969	1.9999999920	1.9999999986	1.99999999932

Example 7.5 Next we apply the Richardson extrapolation to the Beta function

$$B(2/3, 1/3) = \int_0^1 x^{-1/3}(1-x)^{-2/3}dx, \qquad (7.44)$$

where $B(2/3, 1/3) = 2\pi/\sqrt{3} = 3.62759\,87284\,68436$. Let M_n be the n-panels midpoint rule for (7.44). By Example 4.6, we have

$$M_n - B(2/3, 1/3) \sim \sum_{j=1}^{\infty}(c_{2j-1}n^{-1/3-j+1} + c_{2j}n^{-2/3-j+1}). \qquad (7.45)$$

We give the T-table in Table 7.5. Using 255 functional evaluations, we obtain $T_7^{(0)} = 3.62759\,\mathit{69010}$, whose error is -1.83×10^{-6}. This integral converges more slowly than that in Example 7.4.

Table 7.5

Applying the Richardson extrapolation to $B(2/3, 1/3)$

n	f.e.	$T_0^{(n)}$	$T_1^{(n-1)}$	$T_2^{(n-2)}$	$T_3^{(n-3)}$	$T_4^{(n-4)}$	$T_5^{(n-5)}$	$T_6^{(n-6)}$	$T_7^{(n-7)}$
0	1	2.00							
1	3	2.34	3.68						
2	7	2.62	3.70	3.74					
3	15	2.84	3.691	3.665	3.615				
4	31	3.01	3.672	3.639	3.6230	3.6262			
5	63	3.14	3.657	3.631	3.6262	3.6277	3.6280		
6	127	3.25	3.646	3.6289	3.62720	3.62765	3.62764	3.627563	
7	255	3.33	3.639	3.6280	3.62748	3.62761	3.627601	3.6275935	3.6275969

7.3.3 General extrapolation

The E-algorithm described in the preceding section is the first algorithm for a sequence satisfying (7.28). Subsequently, in 1987, W. F. Ford and A. Sidi[16] gave more efficiently algorithm. See, for details, Sidi[53].

8. The ϵ-algorithm

The ϵ-algorithm is a recursive algorithm for the Shanks transformation and is one of the most familiar convergence acceleration methods.

8.1 The Shanks transformation

Suppose that a sequence (s_n) with the limit or the antilimit s satisfies

$$\sum_{i=0}^{k} a_i s_{n+i} = \left(\sum_{i=0}^{k} a_i\right) s, \quad \forall n \in \mathbb{N}, \tag{8.1a}$$

$$\sum_{i=0}^{k} a_i \neq 0, \tag{8.1b}$$

where a_i's are unknown constants independent of n. Then $a_0, \ldots, a_k$ satisfy the system of linear equations

$$\begin{aligned} a_0(s_n - s) + a_1(s_{n+1} - s) + \cdots + a_k(s_{n+k} - s) &= 0 \\ a_0(s_{n+1} - s) + a_1(s_{n+2} - s) + \cdots + a_k(s_{n+k+1} - s) &= 0 \\ \cdots & \\ a_0(s_{n+k} - s) + a_1(s_{n+k+1} - s) + \cdots + a_k(s_{n+2k} - s) &= 0. \end{aligned} \tag{8.2}$$

By (8.1b) and (8.2), we have

$$\begin{vmatrix} s_n - s & s_{n+1} - s & \cdots & s_{n+k} - s \\ s_{n+1} - s & s_{n+2} - s & \cdots & s_{n+k+1} - s \\ & & \cdots & \\ s_{n+k} - s & s_{n+k+1} - s & \cdots & s_{n+2k} - s \end{vmatrix} = 0, \tag{8.3}$$

thus we obtain

$$s \begin{vmatrix} 1 & 1 & \cdots & 1 \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix} = \begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}. \tag{8.4}$$

Therefore,

$$s = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}. \tag{8.5}$$

The right-hand side of (8.5) is denoted $e_k(s_n)$ by D. Shanks[51]:

$$e_k(s_n) = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ & & \cdots & \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}. \tag{8.6}$$

The sequence transformation $(s_n) \mapsto (e_k(s_n))$ is called *Shanks' e-transformation*, or the *Shanks transformation*. In particular, e_1 is the Aitken δ^2 process. By construction, e_k is exact on a sequence satisfying (8.1).

If a sequence (s_n) satisfies

$$s_n = s + \sum_{j=1}^{k} c_j(n)\lambda_j^n, \quad \forall n \in \mathbb{N}, \tag{8.7}$$

where $c_j(n)$ are polynomials in n and $\lambda_j \neq 1$ are constants, then (s_n) satisfies (8.1). On concerning the necessary and sufficient condition for (8.1), see Brezinski and Redivo Zaglia[11, p.79].

The Shanks transformation was first considered by R.J. Schmidt[49] in 1941. He used this method for solving a system of linear equations by iteration but not for accelerating the convergence. For that reason his paper was neglected until P. Wynn[59] quoted it in 1956. The Shanks transformation did not receive much attention until Shanks[51] published in 1955.

The main drawback of the Shanks transformation is to compute determinants of large order. This drawback was recovered by the ϵ-algorithm proposed by Wynn.

8.2 The ϵ-algorithm

Immediately after Shanks rediscovered the e-transformation, P. Wynn proposed a recursive algorithm which is named the ϵ-*algorithm*. He proved the following theorem by using determinantal identities.

Theorem 8.1 (Wynn[59]) *If*

$$\epsilon_{2k}^{(n)} = e_k(s_n), \tag{8.8a}$$

$$\epsilon_{2k+1}^{(n)} = \frac{1}{e_k(\Delta s_n)}, \tag{8.8b}$$

then

$$\epsilon_{k+1}^{(n)} = \epsilon_{k-1}^{(n+1)} + \frac{1}{\epsilon_k^{(n+1)} - \epsilon_k^{(n)}}, \quad k = 1, 2, \ldots, \tag{8.9}$$

provided that none of the denominators vanishes.

For a sequence (s_n), the ϵ-algorithm is defined as follows:

$$\epsilon_{-1}^{(n)} = 0, \quad \epsilon_0^{(n)} = s_n, \quad n = 0, 1, \ldots, \tag{8.10a}$$

$$\epsilon_{k+1}^{(n)} = \epsilon_{k-1}^{(n+1)} + \frac{1}{\epsilon_k^{(n+1)} - \epsilon_k^{(n)}}, \quad k = 0, 1, \ldots. \tag{8.10b}$$

There are many research papers on the ϵ-algorithm. See, for details, Brezinski[9], Brezinski and Redivo Zaglia[11].

8.3 The asymptotic properties of the ϵ-algorithm

P. Wynn gave asymptotic estimates for the quantities $\epsilon_{2k}^{(n)}$ produced by application of the ϵ-algorithm to particular sequences.

Theorem 8.2 (Wynn[61]) *If the ϵ-algorithm is applied to a sequence (s_n) satisfying*

$$s_n \sim s + \sum_{j=1}^{\infty} a_j (n+b)^{-j}, \quad a_1 \neq 0, \tag{8.11}$$

then for fixed k,

$$\epsilon_{2k}^{(n)} \sim s + \frac{a_1}{(k+1)(n+b)}, \quad \textit{as } n \to \infty. \tag{8.12}$$

Theorem 8.2 shows that the ϵ-algorithm cannot accelerate a logarithmically convergent sequence satisfying (8.11). For, by (8.12)

$$\lim_{n\to\infty} \frac{\epsilon_{2k}^{(n)} - s}{s_{n+2k} - s} = \frac{1}{k+1}. \tag{8.13}$$

Theorem 8.3 (Wynn[61]) *If the ϵ-algorithm is applied to a sequence (s_n) satisfying*

$$s_n \sim s + (-1)^n \sum_{j=1}^{\infty} a_j (n+b)^{-j}, \quad a_1 \neq 0, \tag{8.14}$$

then for fixed k,

$$\epsilon_{2k}^{(n)} \sim s + \frac{(-1)^n (k!)^2 a_1}{4^k (n+b)^{2k+1}}, \quad \textit{as } n \to \infty. \tag{8.15}$$

Theorem 8.3 shows that the ϵ-algorithm works well on partial sums of altenating series

$$s_n = \sum_{i=1}^{n} (-1)^{i-1} \frac{1}{ai+b}, \tag{8.16}$$

where $a \neq 0$ and b are constants.

Theorem 8.4 (Wynn[61]) *If the ϵ-algorithm is applied to a sequence (s_n) satisfying*

$$s_n \sim s + \sum_{j=1}^{\infty} a_j \lambda_j^n, \tag{8.17}$$

where $1 > \lambda_1 > \lambda_2 > \cdots > 0$, then for fixed k,

$$\epsilon_{2k}^{(n)} \sim s + \frac{a_{k+1}(\lambda_{k+1}-\lambda_1)^2 \ldots (\lambda_{k+1}-\lambda_k)^2 \lambda_{k+1}^n}{(1-\lambda_1)^2 \ldots (1-\lambda_k)^2}, \quad \text{as } n \to \infty. \tag{8.18}$$

Theorem 8.5 (Wynn[61]) *If the ϵ-algorithm is applied to a sequence (s_n) satisfying*

$$s_n \sim s + (-1)^n \sum_{j=1}^{\infty} a_j \lambda_j^n, \tag{8.19}$$

where $1 > \lambda_1 > \lambda_2 > \cdots > 0$, then for fixed k,

$$\epsilon_{2k}^{(n)} \sim s + (-1)^n \frac{a_{k+1}(\lambda_{k+1}-\lambda_1)^2 \ldots (\lambda_{k+1}-\lambda_k)^2 \lambda_{k+1}^n}{(1+\lambda_1)^2 \ldots (1+\lambda_k)^2}, \quad \text{as } n \to \infty. \tag{8.20}$$

The above theorems are further extended by J. Wimp. The following theorem is an extension of Theorem 8.3.

Theorem 8.6 (Wimp[58, p.127]) *If the ϵ-algorithm is applied to a sequence (s_n) satisfying*

$$s_n \sim s + \lambda^n (n+b)^{\theta} \sum_{j=0}^{\infty} c_j n^{-j}, \tag{8.21}$$

where $c_0 \neq 0$, $\lambda \neq 1$, and $\theta \neq 0, 1, \ldots, k-1$, then for fixed $k > 0$,

$$\epsilon_{2k}^{(n)} \sim s + \frac{c_0 \lambda^{n+2k} (n+b)^{\theta-2k} k! (-\theta)_k}{(\lambda-1)^{2k}} \left[1 + O(\frac{1}{n})\right], \tag{8.22}$$

where $(-\theta)_k = (-\theta)(-\theta+1)\cdots(-\theta+k-1)$, which is called the Pochhammer's symbol.

8.4 Numerical examples of the ϵ-algorithm

Numerical computations reported in the rest of this paper were carried out on the NEC ACOS-610 computer in double precision with approximately 16 digits.

Example 8.1 We apply the ϵ-algorithm to the partial sums of alternating series

$$s_n = \sum_{i=1}^{n} \frac{(-1)^{i-1}}{2i-1}. \tag{8.23}$$

As we described in Example 3.2,

$$s_n = \frac{\pi}{4} + (-1)^{n-1}\frac{1}{n}\left(\frac{1}{4} - \frac{1}{16n^2} + O(\frac{1}{n^4})\right). \tag{8.24}$$

We give s_n and $\epsilon_{2k}^{(n-2k)}$ in Table 8.1, where $k = \lfloor (n-1)/2 \rfloor$. By the first 13 terms, we obtain 10 exact digits. And by the first 20 terms, we obtain 15 exact digits.

Table 8.1

The ϵ-algorithm applying to (8.23)

n	s_n	$\epsilon_{2k}^{(n-2k)}$
1	1.000	
2	0.666	
3	0.866	0.791
4	0.723	0.7833
5	0.834	0.78558
6	0.744	0.78534 7
7	0.820	0.78540 3
8	0.754	0.78539 68
9	0.813	0.78539 832
10	0.760	0.78539 8126
11	0.808	0.78539 81682
12	0.764	0.78539 81623
13	0.804	0.78539 81635 4
14	0.767	0.78539 81633 67
15	0.802	0.78539 81634 01
20	0.772	0.78539 81633 97448
∞	0.785	0.78539 81633 97448

Example 8.2 Let us consider

$$s_n = \sum_{i=1}^{n} \frac{(-1)^{i-1}}{\sqrt{i}}. \tag{8.25}$$

As we described in Example 3.3,

$$s_n = (1-\sqrt{2})\zeta(\frac{1}{2}) + (-1)^{n-1}\frac{1}{\sqrt{n}}\left(\frac{1}{2} - \frac{1}{8n} + \frac{5}{128n^3} + O(\frac{1}{n^5})\right), \tag{8.26}$$

where $(1-\sqrt{2})\zeta(\frac{1}{2}) = 0.60489\ 86434\ 21630$. Theorem 8.6 and (8.26) show that the ϵ-algorithm accelerates the convergence of (s_n). We give s_n and $\epsilon_{2k}^{(n-2k)}$ in Table 8.2, where $k = \lfloor (n-1)/2 \rfloor$. The results are similar to Example 8.1.

Table 8.2

The ϵ-algorithm applying to (8.25)

n	s_n	$\epsilon_{2k}^{(n-2k)}$
1	1.00	
2	0.29	
3	0.87	0.6107
4	0.37	0.6022
5	0.81	0.60504
6	0.40	0.60484 9
7	0.78	0.60490 2
8	0.43	0.60489 74
9	0.76	0.60489 875
10	0.45	0.60489 8614
11	0.75	0.60489 86466
12	0.46	0.60489 86426
13	0.74	0.60489 86435 1
14	0.47	0.60489 86433 99
15	0.73	0.60489 86434 243
20	0.49	0.60489 86434 21630
∞	0.60	0.60489 86434 21630

The ϵ-algorithm can extrapolate certain divergent series.

Example 8.3 Let us now consider

$$s_n = \sum_{i=1}^{n} \frac{(-2)^{i-1}}{i}. \tag{8.27}$$

(s_n) diverges from $\frac{1}{2}\log 3 = 0.54930\ 61443\ 34055$, which is the antilimit of (s_n). We apply the ϵ-algorithm on (8.27). We give s_n and $\epsilon_{2k}^{(n-2k)}$ in Table 8.3, where $k = \lfloor (n-1)/2 \rfloor$.

Table 8.3

The ϵ-algorithm applying to (8.27)

n	s_n	$\epsilon_{2k}^{(n-2k)}$
1	1.0	
2	0.0	
3	1.3	0.571
4	−0.6	0.533
5	2.5	0.5507
6	−2.8	0.5485
7	6.3	0.54940
8	−9.6	0.54926
9	18.7	0.54931 2
10	−32.4	0.54930 32
11	60.6	0.54930 661
12	−109.9	0.54930 595
23	119788.2	0.54930 61443 34119
24	−229737.0	0.54930 61443 34055
∞	0.5	0.54930 61443 34055

9. Levin's transformations

In a favorable influence survey [54], Smith and Ford concluded that the Levin u-transform is the best available across-the-board method. In this section we deal with Levin's transformations.

9.1 The derivation of the Levin T transformation

If a sequence (s_n) satisfies

$$s_n = s + R_n \sum_{j=0}^{k-1} \frac{c_j}{n^j}, \tag{9.1}$$

where R_n are nonzero functions of n and $c_0(\neq 0), \dots, c_{k-1}$ are constants independent of n, then the limit or the antilimit s is represented as

$$s = \frac{\begin{vmatrix} s_n & \dots & s_{n+k} \\ R_n & \dots & R_{n+k} \\ \vdots & \ddots & \vdots \\ R_n/n^{k-1} & \dots & R_{n+k}/(n+k)^{k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ R_n & \dots & R_{n+k} \\ \vdots & \ddots & \vdots \\ R_n/n^{k-1} & \dots & R_{n+k}/(n+k)^{k-1} \end{vmatrix}}. \tag{9.2}$$

Thus, if (s_n) satisfies the asymptotic expansion

$$s_n \sim s + R_n \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{9.3}$$

then the ratio of determinants

$$T_k^{(n)} = \frac{\begin{vmatrix} s_n & \dots & s_{n+k} \\ R_n & \dots & R_{n+k} \\ \vdots & \ddots & \vdots \\ R_n/n^{k-1} & \dots & R_{n+k}/(n+k)^{k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ R_n & \dots & R_{n+k} \\ \vdots & \ddots & \vdots \\ R_n/n^{k-1} & \dots & R_{n+k}/(n+k)^{k-1} \end{vmatrix}} \tag{9.4}$$

may be expected to be a good approximation to s. The transformation of (s_n) into a set of sequences $\{(T_k^{(n)})\}$ is called the *Levin T transformation.*

By properties of determinants, R_n can be multiplied by any constant without affecting $T_k^{(n)}$. Levin[26] proposed three different choices for R_n: $R_n = \Delta s_n$, $R_n = n\Delta s_{n-1}$ and $R_n = \Delta s_{n-1}\Delta s_n/(\Delta s_n - \Delta s_{n-1})$, where $\Delta s_n = s_{n+1} - s_n$. These transformations are called the *Levin t-transform*, the *Levin u-transform* and the *Levin v-transform*, respectively.

There is no need to compute the determinants in (9.4). Levin himself gave the following formula:

$$T_k^{(n)} = \frac{\displaystyle\sum_{j=0}^{k}(-1)^j \binom{k}{j} \left(\frac{n+j}{n+k}\right)^{k-1} \frac{s_{n+j}}{R_{n+j}}}{\displaystyle\sum_{j=0}^{k}(-1)^j \binom{k}{j} \left(\frac{n+j}{n+k}\right)^{k-1} \frac{1}{R_{n+j}}}. \tag{9.5}$$

The formula $T_k^{(n)}$ can be recursively computed by the E-algorithm which was described in Section 6.

$$E_0^{(n)} = s_n, \quad g_{0,j}^{(n)} = n^{1-j} R_n, \quad n = 1, 2, \ldots; j = 1, 2, \ldots, \tag{9.6a}$$

$$E_k^{(n)} = E_{k-1}^{(n)} - g_{k-1,k}^{(n)} \frac{\Delta E_{k-1}^{(n)}}{\Delta g_{k-1,k}^{(n)}}, \quad n = 1, 2, \ldots; k = 1, 2, \ldots, \tag{9.6b}$$

$$g_{k,j}^{(n)} = g_{k-1,j}^{(n)} - g_{k-1,k}^{(n)} \frac{\Delta g_{k-1,j}^{(n)}}{\Delta g_{k-1,k}^{(n)}}, \qquad n = 1, 2, \ldots; k = 1, 2, \ldots, j > k, \tag{9.6c}$$

where Δ operates on the upper index n. By Theorem 6.1, we have $E_k^{(n)} = T_k^{(n)}$.

9.2 The convergence theorem of the Levin transformations

We denote by $\mathscr{A}$ the set of all functions f defined on $[b, \infty)$ for $b > 0$ that have asymptotic expansions in inverse powers of x of the form

$$f(x) \sim \sum_{j=0}^{\infty} \frac{c_j}{x^j} \quad \text{as } x \to \infty. \tag{9.7}$$

The next theorem was given by Sidi[52] and Wimp[58], independently.

Theorem 9.1 (Sidi[52];Wimp[58]) *Suppose that the sequence* (s_n) *has an asymptotic expansion of the form*

$$s_n - s \sim \lambda^n n^{\vartheta} \left[c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \ldots\right], \tag{9.8}$$

where $0 < |\lambda| \le 1$, $\vartheta < 0$, *and* $c_0(\neq 0), c_1, \ldots$ *are constants independent of* n.

(1) *Suppose that* $\lambda \neq 1$. *If we set* $R_n = \Delta s_n$ *or* $R_n = \Delta s_{n-1}\Delta s_n/(\Delta s_n - \Delta s_{n-1})$ *then*

$$T_k^{(n)} - s = O\left(\lambda^n n^{\vartheta - 2k}\right) \quad \text{as } n \to \infty. \tag{9.9}$$

(2) *Suppose that* $\lambda = 1$. *If we set* $R_n = n\Delta s_{n-1}$ *or* $R_n = \Delta s_{n-1}\Delta s_n/(\Delta s_n - \Delta s_{n-1})$ *then*

$$T_k^{(n)} - s = O\left(n^{\vartheta - k}\right) \quad \text{as } n \to \infty. \tag{9.10}$$

Theorem 9.1(1) shows that the Levin t- and v-transforms accelerate certain alternating series.

Recently, N. Osada[41] has extended the Levin transforms to vector sequences. These vector sequence transformations satisfy asymptotic properties similar to Theorem 9.1.

Example 9.1 Let us consider

$$\begin{aligned} s_0 &= 0, \\ s_n &= \sum_{i=1}^{n} \frac{(-1)^{i-1}}{\sqrt{i}}, \quad n = 1, 2, \ldots. \end{aligned} \tag{9.11}$$

We apply the Levin u-, v-, and t-transforms on (9.11). For the Levin u-transform, we take $T_{n-1}^{(1)}$, while $T_{n-2}^{(1)}$ for v- and t-transforms. We give significant digits of s_n, defined by $-\log_{10}|s_n - s|$, and those of $T_{n-k}^{(1)}$ in Table 9.1.

Table 9.1

The significant digits of the Levin transforms applying to (9.11)

n	s_n	Levin u $T_{n-1}^{(1)}$	Levin v $T_{n-2}^{(1)}$	Levin t $T_{n-2}^{(1)}$
1	0.40			
2	0.51	0.99		
3	0.58	2.30	2.30	2.23
4	0.63	3.69	4.14	3.12
5	0.67	5.93	4.45	4.19
6	0.71	6.33	5.53	5.44
7	0.74	7.52	6.93	6.95
8	0.77	9.46	9.17	8.82
9	0.79	10.14	9.42	9.43
10	0.81	11.27	10.70	10.77
11	0.83	13.06	13.52	12.68
12	0.85	13.94	13.18	13.16
13	0.87	14.89	14.36	14.40
14	0.88	15.56	15.56	15.52

Theorem 9.1(2) shows that the Levin u- and v-transforms accelerate logarithmically convergent sequences of the form

$$s_n - s \sim n^{\vartheta}\left[c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \dots\right], \tag{9.12}$$

where $\vartheta < 0$, and $c_0(\neq 0), c_1, \dots$ are constants.

Example 9.2 Let us now consider partial sums of $\zeta(1.5)$:

$$s_n = \sum_{i=1}^{n} \frac{1}{i\sqrt{i}}. \tag{9.13}$$

We apply the Levin u-, v-, and t-transforms on (9.13) with $s_0 = 0$. Similar to the above example, we show significant digits of s_n and $T_{n-k}^{(1)}$ in Table 9.2. Both the u- and v-transforms accelerate but the t-transform cannot.

Table 9.2

The significant digits of the Levin transforms applying to $\zeta(1.5)$

n	s_n	Levin u $T_{n-1}^{(1)}$	Levin v $T_{n-2}^{(1)}$	Levin t $T_{n-2}^{(1)}$
1	−0.21			
2	−0.10	0.39		
3	−0.03	1.22	1.22	0.08
4	0.03	2.37	2.26	0.18
5	0.07	3.74	2.35	0.27
6	0.11	4.16	3.00	0.35
7	0.14	5.42	3.93	0.42
8	0.16	6.18	5.29	0.48
9	0.19	6.93	6.25	0.53
10	0.21	9.01	7.08	0.58
11	0.23	8.72	8.34	0.62
12	0.25	8.89	8.55	0.66
13	0.26	8.14	7.57	0.70
14	0.28	7.58	6.97	0.73

When the asymptotic expansion of logarithmically convergent sequence has logarithmic terms such as $\log n/n$, the Levin u-, v-transforms do not work effectively.

Example 9.3 Consider a sequence defined by

$$s_n = \sum_{j=2}^{n} \frac{\log j}{j^2}. \tag{9.14}$$

As we described in Example 3.6, the sequence (s_n) converges to $-\zeta'(2)$, and has the asymptotic expansion

$$s_n \sim -\zeta'(2) - \frac{\log n}{n} - \frac{1}{n} + \frac{\log n}{2n^2} - \frac{\log n}{6n^3} + \frac{1}{12n^3} - \dots, \tag{9.15}$$

where $\zeta(s)$ is the Riemann zeta function and $\zeta'(s)$ is its derivative. We apply the Levin u-, v-, and t-transforms on (9.14) and show significant digits in Table 9.3.

Table 9.3

The significant digits of the Levin transforms applying to (9.14)

n	s_n	Levin u $T^{(2)}_{n-2}$	Levin v $T^{(2)}_{n-3}$	Levin t $T^{(2)}_{n-3}$
1	0.03			
2	0.12			
3	0.19	−0.47	0.12	0.45
4	0.26	0.49	0.49	0.47
5	0.31	0.71	0.33	0.58
6	0.36	1.77	1.18	0.69
7	0.40	1.82	1.62	0.79
8	0.43	2.23	3.18	0.87
9	0.47	2.28	2.50	0.95
10	0.50	2.40	2.72	1.02
11	0.52	2.50	2.81	1.09
12	0.55	2.59	2.91	1.15
13	0.57	2.67	3.00	1.21
14	0.60	2.75	3.08	1.26
15	0.62	2.82	3.16	1.31

9.3 The *d*-transformation

Levin and Sidi[27] extended the Levin T transformation. Suppose that a sequence (s_n) satisfies

$$s_n = s + \sum_{i=0}^{m-1} R_i^{(n)} \sum_{j=0}^{k-1} \frac{c_{i,j}}{n^j}, \tag{9.16}$$

where $R_i^{(n)}$ $(i = 0, \dots, m-1)$ are nonzero functions of n and $c_{i,j}$ are unknown constants.

Using Cramer's rule, s can be represented as the ratio of two determinants.

$$s = \frac{\begin{vmatrix} s_n & \dots & s_{n+mk} \\ R_0^{(n)} & \dots & R_0^{(n+mk)} \\ R_1^{(n)} & \dots & R_1^{(n+mk)} \\ \vdots & & \vdots \\ R_{m-1}^{(n)} & \dots & R_{m-1}^{(n+mk)} \\ R_0^{(n)}/n & \dots & R_0^{(n+mk)}/(n+mk) \\ \vdots & & \vdots \\ R_{m-1}^{(n)}/n^{k-1} & \dots & R_{m-1}^{(n+mk)}/(n+mk)^{k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ R_0^{(n)} & \dots & R_0^{(n+mk)} \\ R_1^{(n)} & \dots & R_1^{(n+mk)} \\ \vdots & & \vdots \\ R_{m-1}^{(n)} & \dots & R_{m-1}^{(n+mk)} \\ R_0^{(n)}/n & \dots & R_0^{(n+mk)}/(n+mk) \\ \vdots & & \vdots \\ R_{m-1}^{(n)}/n^{k-1} & \dots & R_{m-1}^{(n+mk)}/(n+mk)^{k-1} \end{vmatrix}}. \tag{9.17}$$

We denote by $T_{mk,n}^{(m)}$ the right-hand side of (9.17). The transformation of (s_n) into a set of sequences $\{(T_{mk,n}^{(m)})\}$ is called the *$d^{(m)}$-transformation*.

On the analogy of the Levin u-transform, we take $R_{q-1}^{(n)} = n^q \Delta^q s_{n-q}$ $(q = 1, \dots, m)$ for the $d^{(m)}$-transformation. The $d^{(m)}$-transform can be computed by the E-algorithm. For $n = 1, 2, \dots,$

$$E_0^{(n)} = s_n, \tag{9.18a}$$

$$g_{0,pm+q}^{(n)} = n^{q-p} \Delta^q s_{n-q}, \quad p = 0, 1, \dots; q = 1, \dots, m \tag{9.18b}$$

$$E_k^{(n)} = E_{k-1}^{(n)} - g_{k-1,k}^{(n)} \frac{\Delta E_{k-1}^{(n)}}{\Delta g_{k-1,k}^{(n)}}, \quad k = 1, 2, \dots, \tag{9.18c}$$

$$g_{k,j}^{(n)} = g_{k-1,j}^{(n)} - g_{k-1,k}^{(n)} \frac{\Delta g_{k-1,j}^{(n)}}{\Delta g_{k-1,k}^{(n)}}, \quad k = 1, 2, \dots, j > k, \tag{9.18d}$$

where Δ operates on the index n. Then $E_{mk}^{(n)} = T_{mk,n}^{(m)}$.

Example 9.4 Consider $s_n = \sum_{j=2}^{n} \log j / j^2$ once more. By (9.15), $n^k \Delta^k s_n$ have asymptotic expansion of the form

$$\sum_{j=1}^{\infty} \frac{a_j \log n + b_j}{n^j}. \tag{9.19}$$

Thus (s_n) can be represented as

$$s_n \sim -\zeta'(2) + \sum_{k=1}^{\infty} n^k \Delta^k s_{n-k} \sum_{j=0}^{\infty} c_{kj} n^{-j}. \tag{9.20}$$

This asymptotic expansion suggests that the d-transformation works well on (s_n). We apply the Levin u, which can be regarded as the $d^{(1)}$-transform, the $d^{(2)}$-transform, and the $d^{(3)}$-transform to (s_n) and show significant digits in Table 9.4.

Table 9.4

The significant digits of the d-transforms applying to (9.14)

n	s_n	Levin u	$d^{(2)}$	$d^{(3)}$
1	0.03			
2	0.12			
3	0.19	−0.47		
4	0.26	0.49		
5	0.31	0.71	0.67	
6	0.36	1.77	0.88	
7	0.40	1.82	2.06	1.12
8	0.43	2.23	2.11	1.31
9	0.47	2.28	3.59	1.54
10	0.50	2.40	4.23	2.68
11	0.52	2.50	4.99	3.21
12	0.55	2.59	5.36	3.99
13	0.57	2.67	7.07	5.17
14	0.60	2.75	6.66	5.65
15	0.62	2.82	5.75	5.99

10. The Aitken δ^2 process and its modifications

The Aitken δ^2 process is the most famous as well as the oldest nonlinear sequence transformation. It can accelerate linearly convergent sequences but cannot accelerate some logarithmically convergent sequences. In this section we make clear the above fact and describe some modifications of the Aitken δ^2 process.

10.1 The acceleration theorem of the Aitken δ^2 process

First we define two difference oprators Δ and ∇. The *forward defference operator* Δ is defined by

$$\Delta^0 s_n = s_n \tag{10.1a}$$

$$\Delta^k s_n = \Delta^{k-1} s_{n+1} - \Delta^{k-1} s_n \quad k = 1, 2, \ldots. \tag{10.1b}$$

Similarly, the *backward defference operator* ∇ is defined by

$$\nabla^0 s_n = s_n \tag{10.2a}$$

$$\nabla^k s_n = \nabla^{k-1} s_n - \nabla^{k-1} s_{n-1} \quad k = 1, 2, \ldots; n \geq k. \tag{10.2b}$$

Suppose that a sequence (s_n) satisfies $s_n - s = c\lambda^n$. Then we deduce

$$\frac{s_{n+2} - s}{s_{n+1} - s} = \frac{s_{n+1} - s}{s_n - s}. \tag{10.3}$$

Solving (10.3) for the unknown s, we have

$$s = \frac{s_n s_{n+2} - s_{n+1}^2}{\Delta^2 s_n} = s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n}. \tag{10.4}$$

The *Aitken δ^2 process* is defined by

$$t_n = s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n}, \tag{10.5}$$

or equivalently

$$t_n = s_{n+1} - \frac{\Delta s_{n+1} \nabla s_{n+1}}{\Delta s_{n+1} - \nabla s_{n+1}}, \tag{10.6}$$

$$= s_{n+2} - \frac{(\nabla s_{n+2})^2}{\nabla^2 s_{n+2}}. \tag{10.7}$$

As is well known, the Aitken δ^2 process accelerates any linearly convergent sequence.

Theorem 10.1 (Henrici[21]) *Let (s_n) be a sequence satisfying*

$$s_{n+1} - s = (A + \epsilon_n)(s_n - s), \tag{10.8}$$

where A is a constant, $A \neq 1$, and $\epsilon_n \to 0$ as $n \to \infty$. Let (t_n) be defined by (10.5). Then

$$\lim_{n\to\infty} \frac{t_n - s}{s_{n+2} - s} = 0. \tag{10.9}$$

Proof. See [21, p.73]. □

The Aitken δ^2 process can be applied iteratively as follows. The two dimensional array $(T_k^{(n)})$ is defined by

$$T_0^{(n)} = s_n, \quad n = 1, 2, \dots, \tag{10.10a}$$

$$T_{k+1}^{(n)} = T_k^{(n)} - \frac{(T_k^{(n+1)} - T_k^{(n)})^2}{T_k^{(n+2)} - 2T_k^{(n+1)} + T_k^{(n)}}, \quad k = 0, 1, \dots; n = 1, 2, \dots. \tag{10.10b}$$

The algorithm (10.10) is called the *iterated Aitken δ^2 process.* Since $T_1^{(n)} = \epsilon_2^{(n)}$ for any $n \in \mathbb{N}$, by Theorem 8.6, we have the following theorem.[11]

Theorem 10.2 *Suppose that a sequence (s_n) is satisfied*

$$s_n \sim s + \lambda^n (n+b)^\theta \sum_{j=0}^{\infty} c_j (n+b)^{-j}, \quad c_0 \neq 0, \tag{10.11}$$

where $-1 \leq \lambda < 1$, and $\theta \neq 0, 2, \dots, 2k-2$, then

$$T_k^{(n)} \sim s + \frac{c_0 \lambda^{n+2k} (n+b)^{\theta-2k} (-\theta)(-\theta+2)\cdots(-\theta+2k-2)}{(\lambda-1)^{2k}} \left[1 + O(\frac{1}{n+b})\right]. \tag{10.12}$$

Proof. By induction on k, the proof follows from Theorem 8.6. □

Example 10.1 We apply the iterated Aitken δ^2 process to the partial sums of alternating series

$$s_n = \sum_{i=1}^{n} \frac{(-1)^{i-1}}{2i-1}. \tag{10.13}$$

We give s_n and $T_k^{(n-2k)}$ in Table 10.1, where $k = \lfloor (n-1)/2 \rfloor$. By the first 11 terms, we obtain 10 significant digits. For comparison, the significant digits, abbreviated to "SD ",

[11] Theorem 10.2 was given by K. Murota and M. Sugihara[32].

of $\epsilon_{2k}^{(n-2k)}$ are also given in Table 10.1. The iterated Aitken δ^2 process is better than the ϵ-algorithm, because $0 < (-\theta)(-\theta+2)\ldots(-\theta+2k-2) < k!(-\theta)_k$.

Table 10.1

The iterated Aitken δ^2 process applying to (10.13)

n	s_n	$T_k^{(n-2k)}$	SD of Aitken δ^2	SD of ϵ-algorithm
1	1.000			
2	0.666			
3	0.866	0.791	2.20	2.20
4	0.723	0.7833	2.69	2.69
5	0.834	0.78552	3.89	3.73
6	0.744	0.78536 25	4.45	4.30
7	0.820	0.78539 98	5.78	5.25
8	0.754	0.78539 77	6.35	5.87
9	0.813	0.78539 8178	7.82	6.78
10	0.760	0.78539 8159	8.39	7.43
11	0.808	0.78539 81634 9	8.31	8.31
12	0.764	0.78539 81633 69	10.55	8.98
13	0.804	0.78539 81633 9779	12.46	9.84
14	0.767	0.78539 81633 9731	12.86	10.52
15	0.802	0.78539 81633 97444	14.37	11.37
∞	0.785	0.78539 81633 97448		

10.2 The derivation of the modified Aitken δ^2 formula

On the other hand, the Aitken δ^2 process cannot accelerate logarithmic sequences. Thus some modifications of δ^2 process have been considered. In order to illustrate we take up a model sequence (s_n) satisfying

$$s_n = s + c_0 n^\theta + c_1 n^{\theta-1} + c_2 n^{\theta-2} + O(n^{\theta-3}), \tag{10.14}$$

where $\theta < 0$ and $c_0 (\neq 0), c_1, c_2$ are constants.

Lemma 10.3 *Suppose that a sequence satisfies (10.14). Then the following asymptotic formulae hold.*

(1) $s_n - \dfrac{(\Delta s_n)^2}{\Delta^2 s_n} = s + \dfrac{c_0}{1-\theta} n^\theta + O(n^{\theta-1})$.

(2) $s_n - \dfrac{\theta-1}{\theta}\dfrac{(\Delta s_n)^2}{\Delta^2 s_n} = s + O(n^{\theta-1})$.

(3) $s_n - \dfrac{\theta-1}{\theta}\dfrac{\Delta s_n \nabla s_n}{\Delta s_n - \nabla s_n} = s + O(n^{\theta-2})$.

Proof. See Appendix A. □

A sequence transformation

$$s_n \longmapsto s_n - \frac{\theta-1}{\theta}\frac{(\Delta s_n)^2}{\Delta^2 s_n} \tag{10.15}$$

was considered by Overholt[42]. A sequence transformation

$$s_n \longmapsto s_n - \frac{\theta-1}{\theta}\frac{\Delta s_n \nabla s_n}{\Delta s_n - \nabla s_n} \tag{10.16}$$

was first considered by Drummond[15]. For a sequence satisfying (10.14), Drummond's modification (10.16) is better than Overholt's modification (10.15).

Drummond's modification has been extended by Bjørstad, Dahlquist and Grosse[4] as follows:

$$s_0^{(0)} = 0, \tag{10.17a}$$

$$s_0^{(n)} = s_n, \quad n = 1, 2, \ldots, \tag{10.17b}$$

$$s_{k+1}^{(n)} = s_k^{(n)} - \frac{2k+1-\theta}{2k-\theta}\frac{\Delta s_k^{(n)} \nabla s_k^{(n)}}{\Delta s_k^{(n)} - \nabla s_k^{(n)}}, \quad k = 0, 1, \ldots; n \geq k+1, \tag{10.17c}$$

where Δ and ∇ operate on the upper indexes.[12] The formula (10.17) is called the *modified Aitken δ^2 formula.* The following two theorems are fundamental.

Theorem 10.4 (Bjørstad, Dahlquist and Grosse[3, p.7]) *If $s_k^{(n)}$ is represented as*

$$s_k^{(n)} - s = n^{\theta-2k}\left[c_0^{(k)} + \frac{c_1^{(k)}}{n} + \frac{c_2^{(k)}}{n^2} + O(\frac{1}{n^3})\right], \quad c_0^{(k)} \neq 0, \tag{10.18}$$

then

$$s_{k+1}^{(n)} - s = n^{\theta-2k-2}\left[c_0^{(k+1)} + O(\frac{1}{n})\right], \tag{10.19}$$

where

$$c_0^{(k+1)} = \frac{c_0^{(k)}}{12}(1+\theta-2k) - \frac{(c_1^{(k)})^2}{c_0^{(k)}(2k-\theta)^2} + \frac{2c_2^{(k)}}{(2k-\theta)(2k+1-\theta)}. \tag{10.20}$$

[12] s_n^k defined by Bjørstad, Dahlquist and Grosse agrees with $s_k^{(n)}$ in (10.17).

Theorem 10.5 (Bjørstad, Dahlquist and Grosse) *With the above notation, if* $c_0^{(j)} \neq 0$ *for* $j = 0, \ldots, k-1$, *then*

$$s_k^{(n)} - s = O(n^{\theta - 2k}) \quad as\ n \to \infty. \tag{10.21}$$

Poof. By induction on k, the proof follows from Theorem 10.4. □

Example 10.2 We apply the iterated Aitken δ^2 process and modified Aitken δ^2 process to the partial sums of $\zeta(1.5)$. We give s_n, $T_k^{(n-2k)}$ and $s_l^{(n-l)}$ in Table 10.2, where $k = \lfloor (n-1)/2 \rfloor$ and $l = \lfloor n/2 \rfloor$. By the first 11 terms, we obtain 10 exact digits by the modified Aitken δ^2 formula.

Table 10.2

The iterated Aitken δ^2 process and
the modified Aitken δ^2 formula applying to $\zeta(1.5)$

n	s_n	$T_k^{(n-2k)}$	$s_l^{(n-l)}$
1	1.00		
2	1.35		2.640
3	1.54	1.77	2.6205
4	1.67	1.90	2.6159
5	1.76	2.14	2.61232 9
6	1.82	2.19	2.61237 657
7	1.88	2.33	2.61237 560
8	1.92	2.36	2.61237 53431
9	1.96	2.44	2.61237 53475 5
10	1.99	2.46	2.61237 53487 16
11	2.02	2.539	2.61237 53487 10
12	2.04	2.524	2.61237 53487 17
13	2.06	2.525	2.61237 53486 04
14	2.08	2.522	2.61237 53486 13
∞	2.61	2.612	2.61237 53486 85

10.3 The automatic modified Aitken δ^2 formula

The main drawback of the modified Aitken δ^2 process is that it need the explicit knowledge of the exponent θ such that

$$s_n \sim s + n^\theta \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{10.22}$$

where $\theta < 0$ and $c_0(\neq 0), c_1, \ldots$ are constants. Drummond[15] commented that for a sequence satisfying (10.22),

$$\theta \fallingdotseq \frac{1}{\Delta\left(\dfrac{\Delta s_n}{\Delta^2 s_{n-1}}\right)} + 1, \tag{10.23}$$

where the sign $\fallingdotseq$ denotes approximate equality. Moreover Bjørstad et al.[4] show that

$$\theta_n \sim \theta + n^{-2} \sum_{j=0}^{\infty} \frac{t_j}{n^j}, \tag{10.24}$$

where θ_n is defined by the right-hand side of (10.23) and $t_0(\neq 0), t_1, \ldots$ are constants.

Thus the sequence (θ_n) itself can be accelerated by the modified Aitken δ^2 process with $\theta = -2$ in (10.17c).

Suppose that the first n terms $s_1, \ldots, s_n$ of a sequence satisfying (10.22) are given. Then we define $(t_k^{(m)})$ as follows:
Initialization. $t_0^{(0)} = 0$.
For $m = 1, \ldots, n-2$,

$$t_0^{(m)} = \frac{1}{\Delta\left(\dfrac{\Delta s_m}{\Delta^2 s_{m-1}}\right)} + 1, \tag{10.25a}$$

$$t_{k+1}^{(m)} = t_k^{(m)} - \frac{2k+3}{2k+2} \frac{\Delta t_k^{(m)} \nabla t_k^{(m)}}{\Delta t_k^{(m)} - \nabla t_k^{(m)}}, \tag{10.25b}$$
$$k = 0, \ldots, \lfloor m/2 \rfloor - 1,$$

where $\Delta t_k^{(m)} = t_k^{(m+1)} - t_k^{(m)}$ and $\nabla t_k^{(m)} = t_k^{(m)} - t_k^{(m-1)}$. Then we put

$$\alpha_n = \begin{cases} t_{k-1}^{(k)} & \text{if } n \text{ is odd}, \\ t_k^{(k)} & \text{if } n \text{ is even}, \end{cases} \tag{10.26}$$

where $k = \lfloor (n-1)/2 \rfloor$. Substituting α_n for θ in the definition of the modified Aitken δ^2 formula (10.17), we can obtain the followings:
Initialization. $s_{n,0}^{(0)} = 0$.
For $m = 1, \ldots, n$,

$$s_{n,0}^{(m)} = s_m, \tag{10.27a}$$

$$s_{n,k+1}^{(m)} = s_{n,k}^{(m)} - \frac{2k+1-\alpha_n}{2k-\alpha_n} \frac{\Delta s_{n,k}^{(m)} \nabla s_{n,k}^{(m)}}{\Delta s_{n,k}^{(m)} - \nabla s_{n,k}^{(m)}}, \tag{10.27b}$$
$$k = 0, \ldots, \lfloor n/2 \rfloor - 1,$$

where $\Delta s_{n,k}^{(m)} = s_{n,k}^{(m+1)} - s_{n,k}^{(m)}$ and $\nabla s_{n,k}^{(m)} = s_{n,k}^{(m)} - s_{n,k}^{(m-1)}$.

This scheme is called the *automatic modified Aitken δ^2-formula.* The data flow of this scheme is as follows (case $n = 4$):

$$
\begin{array}{ccccccccc}
s_1 & & & & & & & & \\
s_2 & \searrow & 0 & = & t_0^{(0)} & & & & \\
 & \searrow & & & & & & & \\
s_3 & \to & \theta_1 & = & t_0^{(1)} & \searrow & & & \\
 & \searrow & & & & \searrow & & & \\
s_4 & \to & \theta_2 & = & t_0^{(2)} & \to & t_1^{(1)} & = & \alpha_4
\end{array}
$$

$$
\begin{array}{ccccccc}
s_1 & = & s_{4,0}^{(1)} & & & & \\
 & & & \searrow & & & \\
s_2 & = & s_{4,0}^{(2)} & \to & s_{4,1}^{(1)} & & \\
 & & & \searrow & & & \\
s_3 & = & s_{4,0}^{(3)} & \to & s_{4,1}^{(2)} & \searrow & \\
 & & & \searrow & & \searrow & \\
s_4 & = & s_{4,0}^{(4)} & \to & s_{4,1}^{(3)} & \to & s_{4,2}^{(2)}
\end{array}
$$

For a given tolerance ϵ, this scheme is stopped if n is even and

$$|s_{n,k}^{(k)} - s_{n,k-1}^{(k)}| < \epsilon, \tag{10.28a}$$

or if n is odd and

$$|s_{n,k}^{(k+1)} - s_{n,k}^{(k)}| < \epsilon, \tag{10.28b}$$

where $k = \lfloor n/2 \rfloor$.

A FORTRAN subroutine of the automatic modified Aitken δ^2 formula is given in Appendix.

Example 10.3 We apply the automatic modified Aitken δ^2 formula to the partial sums of $\zeta(1.5)$. We give s_n, α_n in (10.26) and $s_{n,k}^{(n-k)}$ in Table 10.3, where $k = \lfloor n/2 \rfloor$. By the first 11 terms, we obtain 11 exact digits by the automatic modified Aitken δ^2 formula.

Table 10.3

The automatic modified Aitken δ^2 formula

applying to $\zeta(1.5)$

n	s_n	α_n	$s_{n,k}^{(n-k)}$
1	1.00		
2	1.35		
3	1.54	−0.544	2.55
4	1.67	−0.5071	2.604
5	1.76	−0.50015	2.61218
6	1.82	−0.50001 3	2.61236 00
7	1.88	−0.49999 9938	2.61235 4
8	1.92	−0.49999 9967	2.61237 541
9	1.96	−0.50000 017	2.61237 525
10	1.99	−0.50000 0068	2.61237 5314
11	2.02	−0.50000 00001 3	2.61237 53486 35
12	2.04	−0.49999 99992 3	2.61237 53488 2
13	2.06	−0.49999 99999 23	2.61237 53487 2
14	2.08	−0.50000 00000 79	2.61237 53486 20
∞	2.61	−0.50000 00000 00	2.61237 53486 8549

11. Lubkin's *W* transformation

Lubkin's W transformation is the first nonlinear sequence transformation that can accelerate not only linear sequences but also some logarithmic sequences.

11.1 The derivation of Lubkin's *W* transformation

Almost all logarithmically convergent sequences that occur in practice satisfy

$$\lim_{n\to\infty}\frac{s_{n+1}-s}{s_n-s}=\lim_{n\to\infty}\frac{\Delta s_{n+1}}{\Delta s_n}=1. \tag{11.1}$$

Suppose that a sequence (s_n) satisfies

$$\lim_{n\to\infty}\frac{1-\dfrac{s_{n+2}-s}{s_{n+1}-s}}{1-\dfrac{\Delta s_{n+1}}{\Delta s_n}}=\sigma, \tag{11.2}$$

where $\sigma(\neq 0)$, $s\in\mathbb{R}$. As Kowalewski[25, p.268] proved, $0<\sigma\leq 1$. The equality (11.2) is equivalent to

$$\lim_{n\to\infty}\frac{\Delta s_{n+1}\Delta s_n}{(s_{n+1}-s)\Delta^2 s_n}=\sigma. \tag{11.3}$$

If σ in (11.3) is known, solving the equation

$$\frac{\Delta s_{n+1}\Delta s_n}{(s_{n+1}-s)\Delta^2 s_n}=\sigma, \tag{11.4}$$

for the unknown s, we have

$$s=s_{n+1}-\frac{1}{\sigma}\frac{\Delta s_n\Delta s_{n+1}}{\Delta^2 s_n}. \tag{11.5}$$

When $\sigma=1$, the right-hand side of (11.5) coincides with the Aitken δ^2 process. When $0<\sigma<1$, that of (11.5) coincides with the modified Aitken δ^2 formula.

If σ in (11.3) is unknown, solving the equation

$$\frac{\Delta s_{n+2}\Delta s_{n+1}}{(s_{n+2}-s)\Delta^2 s_{n+1}}=\frac{\Delta s_{n+1}\Delta s_n}{(s_{n+1}-s)\Delta^2 s_n} \tag{11.6}$$

for the unknown s, we obtain

$$s=s_{n+1}-\frac{\Delta s_{n+1}\Delta s_n\Delta^2 s_{n+1}}{\Delta s_{n+2}\Delta^2 s_n-\Delta s_n\Delta^2 s_{n+1}}. \tag{11.7}$$

A sequence transformation

$$W : s_n \longmapsto s_{n+1} - \frac{\Delta s_{n+1}\Delta s_n \Delta^2 s_{n+1}}{\Delta s_{n+2}\Delta^2 s_n - \Delta s_n \Delta^2 s_{n+1}} \tag{11.8}$$

is called *Lubkin's W transformation*[29], or the W *transform* for short. The relation between the W transform and the modified Aitken δ^2 process will be treated in subsection 11.3.

For a sequence (s_n), we define W_n by[13]

$$W_n = s_{n+1} - \frac{\Delta s_{n+1}\Delta s_n \Delta^2 s_{n+1}}{\Delta s_{n+2}\Delta^2 s_n - \Delta s_n \Delta^2 s_{n+1}}. \tag{11.9}$$

W_n can be represented as various forms.

$$W_n = s_{n+2} - \frac{\Delta s_{n+1}\Delta s_{n+2} \Delta^2 s_n}{\Delta s_{n+2}\Delta^2 s_n - \Delta s_n \Delta^2 s_{n+1}} \tag{11.10}$$

$$= \frac{\Delta^2 \left(\dfrac{s_n}{\Delta s_n}\right)}{\Delta^2 \left(\dfrac{1}{\Delta s_n}\right)} \tag{11.11}$$

$$= s_{n+2} - \frac{1 - \dfrac{\Delta s_{n+1}}{\Delta s_n}}{\dfrac{1}{\Delta s_{n+2}} - \dfrac{2}{\Delta s_{n+1}} + \dfrac{1}{\Delta s_n}}. \tag{11.12}$$

Since the Aitken δ^2 process can be represented as

$$t_n = \frac{\Delta \left(\dfrac{s_n}{\Delta s_n}\right)}{\Delta \left(\dfrac{1}{\Delta s_n}\right)}, \tag{11.13}$$

the formula (11.11) means that the W transform is a modification of the Aitken δ^2 process. Lubkin[29], Tucker[55] and Wimp[58] studied the relationship between the accelerativeness for the W transform and the Aitken δ^2 process.

We remark that W_n is also represented as

$$W_n = \frac{\begin{vmatrix} s_{n+1} & s_{n+2} \\ \Delta s_n \Delta s_{n+1}/\Delta^2 s_n & \Delta s_{n+1}\Delta s_{n+2}/\Delta^2 s_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta s_n \Delta s_{n+1}/\Delta^2 s_n & \Delta s_{n+1}\Delta s_{n+2}/\Delta^2 s_{n+1} \end{vmatrix}}, \tag{11.14}$$

[13] W_n in (11.9) coincides with Lubkin's original W_{n+2}.

which is the Levin v-transform $T_1^{(n+1)}$ with $R_n = \Delta s_{n-1}\Delta s_n/\Delta^2 s_{n-1}$ in (9.5).

11.2 The exact and acceleration theorems of the W transformation

By construction, the W transform is exact on (s_n) satisfying (11.6). More precisely, the following theorem holds.

Theorem 11.1 (Cordellier[13]) *Suppose that $\Delta s_{n+2}\Delta^2 s_n \neq \Delta s_n \Delta^2 s_{n+1}$ for $\forall n \in \mathbb{N}_0$. Then the W transform is exact on (s_n) if and only if s_n can be represented as*

$$s_n = s + K\prod_{j=1}^{n-1}\frac{ja+b+1}{ja+b}, \tag{11.15}$$

where K is a nonzero real, $a \leq 0$ and

$$\begin{cases} b < -\frac{1}{2} \text{ and } b \neq -1 \text{ if } a = 0, \\ ja + b \neq 0, -1 \text{ for } j \in \mathbb{N}, \text{ if } a < 0. \end{cases}$$

Proof. See [13, p.391] or Osada[40, Theorem 2]. □

Example 11.2 Let K be a nonzero real.

(1) (Wimp[58]) Setting $a = 0$ in (11.15), the W transform is exact on $s_n = s + K(1 + 1/b)^{n-1}$, even if (s_n) deverges.

(2) Setting $a = -1$ and $b = -1$ in (11.15), the W transform is exact on $s_n = s + K/n$.

We cite two theorems that were proved by Lubkin.

Theorem 11.3 (Lubkin[29, Theorem 10]) *Suppose that a sequence (s_n) converges and $\lim_{n\to\infty} \Delta s_{n+1}/\Delta s_n = \rho$. Suppose that one of the following three conditions holds:*

(i) $\rho \neq 0, \pm 1$,

(ii) $\rho = 0$ *and* $\Delta s_{n+1}/\Delta s_n$ *is of constant sign for sufficiently large* n,

(iii) $\rho = -1$ *and* $(1 + \Delta s_{n+2}/\Delta s_{n+1})/(1 + \Delta s_{n+1}/\Delta s_n) > 1$.

Then the W transform accelerates (s_n).

Theorem 11.4 (Lubkin[29, Theorem 12]) *Suppose that a sequence (s_n) converges and $\Delta s_{n+1}/\Delta s_n$ has an asymptotic expansion of the form*

$$\Delta s_n/\Delta s_{n-1} \sim c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \ldots, \tag{11.16}$$

where $c_0, c_1, \ldots$ are constants. Then the W transform accelerates (s_n).

The preceding theorems show that the W transform accelerates not only linear sequences (Theorem 11.3 (i)(iii)) but also a large class of logarithmic sequences (the case $c_0 = 1$ in Theorem 11.4). However, the Aitken δ^2 process has not this property.

11.3 The iteratation of the W transformation

The W transform can be applied iteratively as follows[14] : For $n = 0, 1, \ldots,$[15]

$$W_0^{(n)} = s_n, \tag{11.17a}$$

$$W_{k+1}^{(n)} = W_k^{(n+1)} - \frac{\Delta W_k^{(n+1)} \Delta W_k^{(n)} \Delta^2 W_k^{(n+1)}}{\Delta W_k^{(n+2)} \Delta^2 W_k^{(n)} - \Delta W_k^{(n)} \Delta^2 W_k^{(n+1)}}, \quad k = 0, 1, \ldots, \tag{11.17b}$$

where $\Delta W_k^{(n)} = W_k^{(n+1)} - W_k^{(n)}$. The algorithm (11.17) is called the *iterated W transformation.*

In order to give an asymptotic formula of the iterated W transform, we define the δ_α transformation (Sablonnière[47]). For a sequence (s_n), the δ_α transform is defined by

$$\delta_\alpha(s_n) = s_{n+1} - \frac{\alpha + 1}{\alpha} \frac{\Delta s_n \Delta s_{n+1}}{\Delta^2 s_n}, \tag{11.18}$$

where α is a positive parameter. For a sequence (s_n) satisfying

$$s_n \sim s + n^\theta \sum_{j=0}^{\infty} \frac{c_j}{n^j}, \tag{11.19}$$

$\delta_{-\theta}(s_n) = s_1^{(n+1)}$ in (10.17).

Lemma 11.5 (Sablonnière) *Under the above notation,*

$$W_n = s_{n+1} - \frac{\Delta s_{n+1}(s_{n+1} - \delta_\alpha(s_n))}{\Delta s_{n+1} - \Delta(\delta_\alpha s_n)}, \quad \forall \alpha > 0. \tag{11.20}$$

Proof. By the definition of δ_α, we have

$$s_{n+1} - \delta_\alpha(s_n) = \frac{\alpha + 1}{\alpha} \frac{\Delta s_n \Delta s_{n+1}}{\Delta^2 s_n}, \tag{11.21}$$

and

$$\Delta s_{n+1} - \Delta(\delta_\alpha s_n) = \frac{\alpha + 1}{\alpha} \left(\frac{\Delta s_{n+2} \Delta s_{n+1}}{\Delta^2 s_{n+1}} - \frac{\Delta s_n \Delta s_{n+1}}{\Delta^2 s_n} \right). \tag{11.22}$$

Thus we can obtain the result (11.20). $\square$

---

[14] $W_k^{(n)}$ in (11.17) coincides with $\mathscr{T}_k^{(n)}$ of Weniger[57] and $W_{n+3k,k}$ of Osada[39,p.363].
[15] When the sequence s_n is defined in $n \geq 1$, substitute '$n = 1, 2, \ldots$ ' for '$n = 0, 1, \ldots$ '.

Using the above lemma and the asymptotic formula of the modified Aitken δ^2 formula (Theorem 10.4), Sablonnière has proved the following theorem.

Theorem 11.6 (Sablonnière[47]) *Suppose that a sequence* (s_n) *satisfies (11.19). If* $W_k^{(n)}$ *is represented as*

$$W_k^{(n)} - s = n^{\theta-2k}\left[c_0^{(k)} + \frac{c_1^{(k)}}{n} + \frac{c_2^{(k)}}{n^2} + O(\frac{1}{n^3})\right], \quad c_0^{(k)} \neq 0, \tag{11.23}$$

then

$$W_{k+1}^{(n)} - s = n^{\theta-2k-2}\left[c_0^{(k+1)} + O(\frac{1}{n})\right], \tag{11.24}$$

where

$$\begin{aligned} c_0^{(k+1)} &= \frac{c_0^{(k)}(1+\theta-2k)}{6(\theta-2k)} - \frac{2(c_0^{(k)}k(\theta-2k)-c_1^{(k)})^2}{c_0^{(k)}(\theta-2k)^3} \\ &+ \frac{c_0^{(k)}k^2(\theta-2k)(\theta-2k-1) - 4c_1^{(k)}k(\theta-2k-1) + 4c_2^{(k)}}{(\theta-2k)^2(\theta-2k-1)} \end{aligned} \tag{11.25}$$

Proof. See Appendix B. □

Theorem 11.7 (Sablonnière[47]) *With the above notation, if* $c_0^{(j)} \neq 0$ *for* $j = 0, 1, \ldots, k-1$*, then*

$$W_k^{(n)} - s = O(n^{\theta-2k}) \quad \text{as } n \to \infty \tag{11.26}$$

Proof. By induction on k, the proof follows from Theorem 11.6. □

Sablonnière has also given an asymptotic formula of the W transform applying to a sequence satisfying

$$s_n \sim s + n^{\theta}\sum_{j=0}^{\infty}\frac{c_j}{n^{j/2}}, \tag{11.27}$$

where $\theta < 0$ and $c_0(\neq 0), c_1(\neq 0), c_2, \ldots$ are constants.

Theorem 11.8 (Sablonnière[47]) *Suppose that a sequence* (s_n) *satisfies (11.27). If* $W_k^{(n)}$ *has an asymptotic formula of the form*

$$W_k^{(n)} - s = c_0^{(k)}n^{\theta-k/2} + c_1^{(k)}n^{\theta-k/2-1/2} + O(n^{\theta-k/2-1}), \quad c_0^{(k)} \neq 0, \tag{11.28}$$

then

$$W_{k+1}^{(n)} - s = c_0^{(k+1)}n^{\theta-k/2-1/2} + c_1^{(k+1)}n^{\theta-k/2-1} + O(n^{\theta-k/2-3/2}), \tag{11.29}$$

where

$$c_0^{(k+1)} = \frac{c_1^{(k)}}{(k-2\theta)^2(k+2-2\theta)} \tag{11.30}$$

and

$$c_1^{(k+1)} = \frac{(c_1^{(k)})^2(k-1-2\theta)(k+1-2\theta)^2}{c_0^{(k)}(k-2\theta)^4(k+2-2\theta)^2}. \tag{11.31}$$

Recently, Osada[39] has extended the iterated W transform to vector sequences; the Euclidean W transform and the vector W transform. A similar property to Theorem 11.7 holds for both transforms.

11.4 Numerical examples of the iterated *W* transformation

For linearly convergent sequences the W transform works well.

Example 11.1 Let us consider

$$s_n = \sum_{i=1}^{n} (-1)^{i-1} \frac{1}{\sqrt{i}}. \tag{11.32}$$

We apply the iterated W transform to (11.32). We give s_n, $W_k^{(n-3k)}$ in Table 11.1, where $k = \lfloor (n-1)/3 \rfloor$. By the first 17 terms, we obtain 15 exact digits.

Table 11.1

The iterated W transform applying to (11.32)

n	s_n	$W_k^{(n-3k)}$
1	1.00	
2	0.29	
3	0.87	
4	0.37	0.6061
5	0.81	0.60442
6	0.40	0.60511
7	0.78	0.60490 0
8	0.43	0.60489 79
9	0.76	0.60489 888
10	0.45	0.60489 86446
11	0.75	0.60489 86430 6
12	0.46	0.60489 86435 4
13	0.74	0.60489 86432 2192
14	0.47	0.60489 86432 2155
17	0.72	0.60489 86432 21630
∞	0.60	0.60489 86432 21630

Example 11.2 We apply the iterated W transform to the partial sums of $\zeta(1.5)$. We give s_n, $W_k^{(n-3k)}$ in Table 11.2, where $k = \lfloor (n-1)/3 \rfloor$. By the first 15 terms, we obtain 9 exact digits. For this series, the iterated Aitken δ^2 process cannot accelerate but the iterated W transform can do. However, the W transform is inferior to the automatic modified Aitken δ^2 formula.

Table 11.2

The iterated W transform applying to $\zeta(1.5)$

n	s_n	$W_k^{(n-3k)}$
1	2.0	
2	1.35	
3	1.54	
4	1.67	2.590
5	1.76	2.6019
6	1.82	2.6063
7	1.88	2.61234 3
8	1.92	2.61236 2
9	1.96	2.61236 90
10	1.99	2.61237 527
11	2.02	2.61237 5326
12	2.04	2.61237 5337
13	2.06	2.61237 5330
14	2.08	2.61237 5365
15	2.10	2.61237 53440
∞	2.61	2.61237 53486

Example 11.3 We apply the iterated W transform to the partial sums of $\zeta(1.5)+\zeta(2) =$ 4.25730 94155 33714:

$$s_n = \sum_{i=1}^{n} \frac{\sqrt{i}+1}{i^2}. \tag{11.33}$$

We give s_n, $W_k^{(n-3k)}$ in Table 11.3, where $k = \lfloor (n-1)/3 \rfloor$. By the first 20 terms, we obtain 4 exact digits. For comparison, we show Levin v-transform $T_{n-2}^{(1)}$. The W transform is slightly better than the Levin v-transform.

Table 11.3

The iterated W transform applying to $\zeta(1.5)+\zeta(2)$

n	s_n	$W_k^{(n-3k)}$	$T_{n-2}^{(1)}$
1	1.0		
2	2.6		
3	2.9		4.05
4	3.09	4.14	4.18
5	3.22	4.17	4.212
6	3.31	4.19	4.226
7	3.39	4.2568	4.234
8	3.45	4.2590	4.240
9	3.50	4.2596	4.243
10	3.54	4.2596	4.246
11	3.58	4.2596	4.248
12	3.61	4.2596	4.249
13	3.63	4.2596	4.2509
14	3.66	4.2596	4.2518
15	3.70	4.2591	4.2525
20	3.77	4.25727 2	4.2546
∞	4.25	4.25730 9	4.25730

12. The ρ-algorithm

As Smith and Ford[54] pointed out, the ρ-algorithm of Wynn works well on some logarithmic sequences but fails on another logarithmic sequences. In this section we make clear this fact.

12.1 The reciprocal differences and the ρ-algorithm

Since the ρ-algorgithm is a special case of reciprocal differences, we begin with the definition of reciprocal differences. Let $f(x)$ be a function. The *reciprocal differences* of $f(x)$ with arguments $x_0, x_1, \dots$ are defined recursively by

$$\rho_0(x_0) = f(x_0), \tag{12.1a}$$

$$\rho_1(x_0, x_1) = \frac{x_0 - x_1}{\rho_0(x_0) - \rho_0(x_1)}, \tag{12.1b}$$

$$\begin{aligned}\rho_k(x_0, \dots, x_k) &= \rho_{k-2}(x_1, \dots, x_{k-1}) \\ &\quad + \frac{x_0 - x_k}{\rho_{k-1}(x_0, \dots, x_{k-1}) - \rho_{k-1}(x_1, \dots, x_k)}, \quad k = 2, 3, \dots .\end{aligned} \tag{12.1c}$$

Substituting x for x_0 in (12.1), we have the following continued fraction.

$$f(x) = f(x_1) + \cfrac{x - x_1}{\rho_1(x_1, x_2) + \cfrac{x - x_2}{\rho_2(x_1, x_2, x_3) - \rho_0(x_1) + \cfrac{x - x_3}{\ddots}}} \tag{12.2a}$$

The last two constituent partial fractions are as follows:

$$\cfrac{x - x_{l-1}}{\rho_{l-1}(x_1, \dots, x_l) - \rho_{l-3}(x_1, \dots, x_{l-2}) + \cfrac{x - x_l}{\rho_l(x, x_1, \dots, x_l) - \rho_{l-2}(x_1, \dots, x_{l-1})}}. \tag{12.2b}$$

The equality of (12.2) holds for $x = x_1, \dots, x_l$. The right-hand side of (12.2) is called *Thiele's interpolation formula.*

The ρ-algorithm of Wynn[60] is defined by substituting s_n for $f(x_n)$ and $\rho_k^{(n)}$ for $\rho_k(x_n, \dots, x_{n+k})$ in the reciprocal difference:

$$\rho_0^{(n)} = s_n, \tag{12.3a}$$

$$\rho_1^{(n)} = \frac{1}{\rho_0^{(n+1)} - \rho_0^{(n)}}, \tag{12.3b}$$

$$\rho_k^{(n)} = \rho_{k-2}^{(n+1)} + \frac{k}{\rho_{k-1}^{(n+1)} - \rho_{k-1}^{(n)}}, \quad k = 2, 3, \dots . \tag{12.3c}$$

Thiele's interpolation formula implies the continued fraction.

$$s_n = s_m + \cfrac{n-m}{\rho_1^{(m)} + \cfrac{n-m-1}{\rho_2^{(m)} - \rho_0^{(m)} + \cfrac{n-m-2}{\rho_3^{(m)} - \rho_1^{(m)} + \ddots}}}. \tag{12.4}$$

Neglecting the term

$$\cfrac{n-m-2k}{\rho_{2k+1}^{(m)} - \rho_{2k-1}^{(m)} + \cfrac{n-m-2k-1}{\ddots}}, \tag{12.5}$$

we obtain

$$s_n \fallingdotseq \frac{\rho_{2k}^{(m)} n^k + a_1 n^{k-1} + \cdots + a_k}{n^k + b_1 n^{k-1} + \cdots + b_k}, \tag{12.6}$$

where $a_1, \ldots, a_k, b_1, \ldots, b_k$ are constants independent of n and the sign $\fallingdotseq$ denotes approximate equality. By construction, the equality of (12.6) holds for $n = m, \ldots, m + 2k$.

Suppose a sequence (s_n) with the limit s satisfies

$$s_n = \frac{s n^k + a_1 n^{k-1} + \cdots + a_k}{n^k + b_1 n^{k-1} + \cdots + b_k}. \tag{12.7}$$

Then, by the above discussion, $\rho_{2k}^{(m)} = s$ for any m. Thus the ρ-algorithm is a rational extrapolation which is exact on a sequence satisfying (12.7).

A sequence satisfying (12.7) has an asymptotic expansion of the form

$$s_n \sim s + n^{\theta} \left(c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \ldots \right), \qquad \text{as } n \to \infty \tag{12.8}$$

where θ is a negative integer and c_j's are constants independent of n. Conversely, suppose that θ is a negative integer. If we truncate the terms up to c_k/n^k in (12.8), then s_n satisfies (12.7). This fact suggests that the ρ-algorithm works well on (12.8) if and only if θ is a negative integer, which will be proved in the end of this section.

Recently, Osada[39] has extended the ρ-algorithm to vector sequences; the vector ρ-algorithm and the topological ρ-algorithm.

12.2 The asymptotic behavior of the ρ-algorithm

In order to describe the asymptotic behavior of the ρ-algorithm, we shall use the following sequence. For a given non-integer θ and a given nonzero real c, we define the sequence (C_n) as follows:

$$C_{-1} = 0, \tag{12.9a}$$

$$C_0 = c, \tag{12.9b}$$

$$C_{2k-1} = C_{2k-3} + \frac{2k-1}{\theta C_{2k-2}} \quad k = 1, 2, \ldots, \tag{12.9c}$$

$$C_{2k} = C_{2k-2} + \frac{2k}{(1-\theta)C_{2k-1}} \quad k = 1, 2, \ldots, \tag{12.9d}$$

This sequence (C_n) is called the *associated sequence of the ρ-algorithm with respect to θ and c.*

For the associated sequence of the ρ-algorithm, the following two theorems hold.

Theorem 12.1 *Under the above notation,*

$$C_{2k-1} = \frac{k(2-\theta)(3-\theta)\cdots(k-\theta)}{c\theta(1+\theta)\cdots(k-1+\theta)}, \quad k = 1, 2, \ldots \tag{12.10a}$$

$$C_{2k} = \frac{c(1+\theta)\cdots(k+\theta)}{(1-\theta)(2-\theta)\cdots(k-\theta)}, \quad k = 1, 2, \ldots. \tag{12.10b}$$

Proof. By induction on k. For $k = 1$, $C_1 = C_{-1} + 1/c\theta = 1/c\theta$, $C_2 = C_0 + 2/(1-\theta)C_1 = c(1+\theta)/(1-\theta)$. Assuming that they are valid for $k > 1$. By the induction hypothesis, we have

$$\begin{aligned} C_{2k+1} &= C_{2k-1} + \frac{2k+1}{\theta C_{2k}} \\ &= \frac{k(2-\theta)\cdots(k-\theta)}{c\theta(1+\theta)\cdots(k-1+\theta)} + \frac{(2k+1)(1-\theta)\cdots(k-\theta)}{c\theta(1+\theta)\cdots(k+\theta)} \\ &= \frac{(k+1)(2-\theta)\cdots(k-\theta)(k+1-\theta)}{c\theta(1+\theta)\cdots(k+\theta)}. \end{aligned} \tag{12.11}$$

Similarly,

$$C_{2k+2} = \frac{c(1+\theta)(2+\theta)\cdots(k+1+\theta)}{(1-\theta)\cdots(k+1-\theta)}. \tag{12.12}$$

This completes the proof. □

We remark that Theorem 12.1 is still valid when θ is an integer and $k < |\theta|$.

Theorem 12.2 *Under the above notation,*

$$\lim_{k\to\infty} \frac{C_{2k}}{k^{2\theta}} = -\frac{c\Gamma(-\theta)}{\Gamma(\theta)}, \tag{12.13}$$

where $\Gamma(x)$ is the Gamma function.

Proof. By means of Euler's limit formula for the Gamma function

$$\Gamma(x) = \lim_{k\to\infty} \frac{k!k^x}{x(x+1)\cdots(x+k)}, \tag{12.14}$$

we obtain the result. □

Now we have asymptotic behavior of the ρ-algorithm.

Theorem 12.3 *Let (s_n) be a sequence satisfying*

$$s_n \sim s + n^\theta \left(c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \dots\right), \qquad \text{as } n \to \infty. \tag{12.15}$$

Let (C_n) be the associated sequence of the ρ-algorithm with respect to θ and c_0 in (12.15). Let $A = (1-\theta)(-1/2 + c_1/c_0\theta)$. Then the following formulae are valid.

(1)

$$\rho_1^{(n)} = C_1(n+1)^{1-\theta}\left[1 + \frac{A}{n+1} + \frac{B_1}{(n+1)^2} + O((n+1)^{-3})\right], \tag{12.16}$$

where

$$B_1 = \frac{\theta^2-1}{12} + \frac{c_1(1-\theta)}{2c_0} + \frac{(1-\theta)^2{c_1}^2}{{c_0}^2\theta^2} + \frac{c_2(2-\theta)}{c_0\theta}. \tag{12.17}$$

(2)

$$\rho_2^{(n)} = s + C_2(n+1)^\theta\left[1 + \frac{c_1}{c_0(n+1)} + \frac{B_2}{(n+1)^2} + O((n+1)^{-3})\right], \tag{12.18}$$

where

$$B_2 = -\frac{c_0\theta(1+\theta)}{6(1-\theta)} + \frac{2{c_1}^2}{c_0\theta(1-\theta)} + \frac{c_2(5-\theta^2)}{(1-\theta)^2}. \tag{12.19}$$

(3) *Suppose that* $\theta \neq -1, \ldots, 1-k$. *For* $j = 1, \ldots, k$,

$$\rho_{2j-1}^{(n)} = C_{2j-1}(n+j)^{1-\theta}\left[1 + \frac{A}{n+j}\right] + O((n+j)^{-1-\theta}) \tag{12.20}$$

$$\rho_{2j}^{(n)} = s + C_{2j}(n+j)^{\theta}\left[1 + \frac{c_1}{c_0(n+j)}\right] + O((n+j)^{\theta-2}) \tag{12.21}$$

Proof. (1) Using the binomial expansion, we have

$$\begin{aligned} &s_{n+1} - s_n \\ &= c_0\theta(n+1)^{\theta-1}\left[1 - \frac{A}{n+1} + \left(-\frac{1-\theta}{6} + \frac{c_1(1-\theta)}{2c_0\theta} + \frac{c_2}{c_0\theta}\right)\frac{\theta-2}{(n+1)^2}\right] \\ &\quad + O((n+1)^{\theta-3}). \end{aligned} \tag{12.22}$$

Hence, we obtain

$$\rho_1^{(n)} = C_1(n+1)^{1-\theta}\left[1 + \frac{A}{n+1} + \frac{B_1}{(n+1)^2} + O((n+1)^{-3})\right]. \tag{12.23}$$

(2) and (3). Similarly to (1). □

By Theorem 12.3, when θ in (12.15) is non-integer, for fixed k,

$$\frac{\rho_{2k}^{(n)} - s}{s_{n+2k} - s} \sim C_{2k} \quad \text{as } n \to \infty, \tag{12.24}$$

where the sign $\sim$ means asymptotic approximate. When $-\theta$ is small non-integer, the ρ-algorithm cannot be available.

When θ is a negative integer, say $-k$, we have $C_0 \neq 0, \ldots, C_{2k-2} \neq 0$ and $C_{2k} = 0$. Thus, it follows from Theorem 12.3 that

$$\rho_{2k}^{(n)} = s + O((n+k)^{-k-2}), \quad \text{as } n \to \infty. \tag{12.25}$$

As illustrations, we give two examples.

Example 12.1 We apply the ρ-algorithm to the partial sums of $\zeta(2)$:

$$s_n = \sum_{i=1}^{n} \frac{1}{i^2}. \tag{12.26}$$

We give s_n and $\rho_{2k}^{(n-2k)}$ in Table 12.1, where $k = \lfloor (n-1)/2 \rfloor$. By the first 12 terms, we obtain 12 exact digits.

Table 12.1

The ρ-algorithm applying to $\zeta(2)$

n	s_n	$\rho_{2k}^{(n-2k)}$
1	1.00	
2	1.25	
3	1.36	1.650
4	1.42	1.6468
5	1.46	1.64489
6	1.49	1.64492 2
7	1.511	1.64493 437
8	1.527	1.64493 414
9	1.539	1.64493 40643
10	1.549	1.64493 40662 8
11	1.558	1.64493 40668 64
12	1.564	1.64493 40668 418
13	1.570	1.64493 40668 56
14	1.575	1.64493 40668 82
15	1.580	1.64493 40668 56
20	1.596	1.64493 40668 50
∞	1.644	1.64493 40668 48

Example 12.2 We apply the ρ-algorithm to the partial sums of $\zeta(1.5)$:

$$s_n = \sum_{i=1}^{n} \frac{1}{i\sqrt{i}}. \tag{12.27}$$

We give s_n and $\rho_{2k}^{(n-2k)}$ in Table 12.2, where $k = \lfloor (n-1)/2 \rfloor$. Since $\theta = -0.5$, the ρ-algorithm cannot accelerate (12.27).

Table 12.2

The ρ-algorithm applying to $\zeta(1.5)$

n	s_n	$\rho_{2k}^{(n-2k)}$
1	2.0	
2	1.35	
3	1.54	2.19
4	1.67	2.25
5	1.76	2.40
6	1.82	2.42
7	1.88	2.48
8	1.92	2.49
9	1.96	2.525
10	1.99	2.528
11	2.02	2.520
12	2.04	2.553
13	2.06	2.552
14	2.08	2.553
15	2.10	2.564
∞	2.61	2.612

13. Generalizations of the ρ-algorithm*

Let (s_n) be a sequence satisfying

$$s_n \sim s + n^\theta \left(c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \dots \right), \qquad \text{as } n \to \infty \tag{13.1}$$

where $\theta < 0$ and $c_0(\neq 0), c_1, \dots$ are constants independent of n. As we proved in the preceding section, the ρ-algorithm works well on a sequence satisfying (13.1) if and only if θ is a negative integer. In this section we extend ρ-algorithm that works well on (13.1) for any $\theta < 0$.

13.1 The generalized ρ-algorithm

For a sequence (s_n) satisfying (13.1), we put $s_0 = 0$ if s_0 is not defined. We define $\rho_k^{(n)}$ as follows:

$$\rho_{-1}^{(n)} = 0, \tag{13.2a}$$

$$\rho_0^{(n)} = s_n, \tag{13.2b}$$

$$\rho_k^{(n)} = \rho_{k-2}^{(n+1)} + \frac{k-1-\alpha}{\rho_{k-1}^{(n+1)} - \rho_{k-1}^{(n)}}, \quad k = 1, 2, \dots. \tag{13.2c}$$

This procedure is called the *generalized ρ-algorithm with a parameter α* [37]. It is obvious that, when $\alpha = -1$, the generalized ρ-algorithm coincides with the ρ-algorithm defined in (12.3).

We now derive asymptotic behaviors for the quantities $\rho_k^{(n)}$ produced by applying the generalized ρ-algorithm with parameter θ to the sequence satisfying (13.1).

Theorem 13.1 (Osada) *If $\rho_{2k-1}^{(n)}$ and $\rho_{2k}^{(n)}$ satisfy the asymptotic formulae of the forms*

$$\rho_{2k-1}^{(n)} = -\frac{1}{d_0^{(k-1)}} (n+k-1)^{2k-1-\theta} \left[1 + O((n+k-1)^{-1}) \right], \tag{13.3}$$

$$\rho_{2k}^{(n)} = s + (n+k)^{\theta-2k} \left[d_0^{(k)} + \frac{d_1^{(k)}}{n+k} + \frac{d_2^{(k)}}{(n+k)^2} + O((n+k)^{-3}) \right] \tag{13.4}$$

with $d_0^{(k-1)} \neq 0$ and $d_0^{(k)} \neq 0$, then

$$\rho_{2k+2}^{(n)} = s + (n+k+1)^{\theta-2k-2} \left[d_0^{(k+1)} + O((n+k+1)^{-1}) \right], \tag{13.5}$$

*The material in this section is taken from the author's paper: N. Osada, A convergence acceleration method for some logarithmically convergent sequences, *SIAM J. Numer. Anal.* 27(1990), pp.178-189.

where

$$d_0^{(k+1)} = -\frac{d_0^{(k)}}{12}(2k-\theta-1) - \frac{(d_1^{(k)})^2}{d_0^{(k)}(2k-\theta)^2} + \frac{2d_2^{(k)}}{(2k-\theta)(2k-\theta+1)} + \frac{(d_0^{(k)})^2(2k-\theta-1)}{d_0^{(k-1)}(2k-\theta+1)}. \tag{13.6}$$

Proof. Using (13.4) and the binomial expansion, we have

$$\begin{aligned}\rho_{2k}^{(n+1)} - \rho_{2k}^{(n)} &= -d_0^{(k)}(2k-\theta)(n+k+1)^{\theta-2k-1}\\ &\times\left[1+\left(\frac{1}{2}+\frac{d_1^{(k)}}{d_0^{(k)}(2k-\theta)}\right)\frac{2k-\theta+1}{n+k+1}\right.\\ &\quad+\left(\frac{2k-\theta-1}{6}+\frac{d_1^{(k)}(2k-\theta+1)}{2d_0^{(k)}(2k-\theta)}+\frac{d_2^{(k)}}{d_0^{(k)}(2k-\theta)}\right)\frac{2k-\theta+2}{(n+k+1)^2}\\ &\quad\left.+O((n+k+1)^{-3})\right].\end{aligned} \tag{13.7}$$

Hence, we obtain

$$\begin{aligned}\frac{2k-\theta}{\rho_{2k}^{(n+1)}-\rho_{2k}^{(n)}} &= -\frac{1}{d_0^{(k)}}(n+k+1)^{2k+1-\theta}\\ &\times\left[1-\left(\frac{1}{2}+\frac{d_1^{(k)}}{d_0^{(k)}(2k-\theta)}\right)\frac{2k-\theta+1}{n+k+1}\right.\\ &\quad+\left(\frac{2k-\theta-1}{12}+\frac{d_1^{(k)}}{2d_0^{(k)}}+\frac{(d_1^{(k)})^2(2k-\theta+1)}{(d_0^{(k)})^2(2k-\theta)^2}\right.\\ &\quad\left.-\frac{d_2^{(k)}(2k-\theta+2)}{d_0^{(k)}(2k-\theta)(2k-\theta+1)}\right)\frac{2k-\theta+1}{(n+k+1)^2}\\ &\quad\left.+O((n+k+1)^{-3})\right].\end{aligned} \tag{13.8}$$

By means of (13.2) and (13.8),

$$\begin{aligned}\rho_{2k+1}^{(n)} &= -\frac{1}{d_0^{(k)}}(n+k+1)^{2k+1-\theta}\\ &\times\left[1-\left(\frac{1}{2}+\frac{d_1^{(k)}}{d_0^{(k)}(2k-\theta)}\right)\frac{2k-\theta+1}{n+k+1}\right.\\ &\quad+\left(\frac{2k-\theta-1}{12}+\frac{d_1^{(k)}}{2d_0^{(k)}}+\frac{(d_1^{(k)})^2(2k-\theta+1)}{(d_0^{(k)})^2(2k-\theta)^2}\right.\\ &\quad\left.-\frac{d_2^{(k)}(2k-\theta+2)}{d_0^{(k)}(2k-\theta)(2k-\theta+1)}+\frac{d_0^{(k)}}{d_0^{(k-1)}(2k-\theta+1)}\right)\frac{2k-\theta+1}{(n+k+1)^2}\\ &\quad\left.+O((n+k+1)^{-3})\right].\end{aligned} \tag{13.9}$$

Similarly we obtain

$$
\begin{aligned}
\rho_{2k+2}^{(n)} &= \rho_{2k}^{(n+1)} + \frac{2k-\theta+1}{\rho_{2k+1}^{(n+1)} - \rho_{2k+1}^{(n)}} \\
&= s + (n+k+1)^{\theta-2k-2}\Bigg[-\frac{d_0^{(k)}}{12}(2k-\theta-1) - \frac{(d_1^{(k)})^2}{d_0^{(k)}(2k-\theta)^2} \\
&\qquad + \frac{2d_2^{(k)}}{(2k-\theta)(2k-\theta+1)} + \frac{(d_0^{(k)})^2(2k-\theta-1)}{d_0^{(k-1)}(2k-\theta+1)} \\
&\qquad + O((n+k+1)^{-1})\Bigg]. \qquad (13.10)
\end{aligned}
$$

This completes the proof. □

Theorem 13.2 (Osada) *With the notation above, if $d_0^{(j)} \neq 0$ for $j = 0, 1, \ldots, k$, then*

$$\rho_{2k}^{(n)} = s + O((n+k)^{\theta-2k}). \qquad (13.11)$$

Proof. By means of the induction on k, the proof follows from Theorem 13.1. □

It is easy to see that $\rho_2^{(n)} = s_1^{(n+1)}$ in (10.17) when $\alpha = \theta$. Moreover, under the assumption of Theorem 13.2, $\rho_{2k}^{(n)} - s$ has the same order as $s_k^{(n+k)} - s$ defined in (10.17).

For another information of the generalized ρ-algorithm, see Weniger[57].

Example 13.1 We apply the generalized ρ-algorithm to the partial sums of $\zeta(1.5)$. We give s_n, $\rho_{2k}^{(n-2k)}$ in Table 13.1, where $k = \lfloor n/2 \rfloor$. For comparison, we also give the modified Aitken δ^2 formula $s_l^{(n-l)}$, where $l = \lfloor n/2 \rfloor$.

Table 13.1

The generalized ρ-algorithm and
the modified Aitken δ^2 formula applied to $\zeta(1.5)$

n	k	s_n	$\rho_{2k}^{(n-2k)}$	$s_k^{(n-k)}$
1	0	1.00		
2	1	1.35	2.640	2.640
3	1	1.54	2.6205	2.6205
4	2	1.67	2.61215	2.61217
5	2	1.76	2.61232 3	2.61232 9
6	3	1.82	2.61237 71	2.61237 657
7	3	1.88	2.61237 572	2.61237 560
8	4	1.92	2.61237 5334	2.61237 53431
9	4	1.96	2.61237 53458	2.61237 53475 5
10	5	1.99	2.61237 53488 0	2.61237 53487 16
11	5	2.02	2.61237 53487 2	2.61237 53487 10
12	6	2.04	2.61237 53489 2	2.61237 53487 17
13	6	2.06	2.61237 53487 0	2.61237 53486 04
14	7	2.08	2.61237 53486 848	2.61237 53486 13
∞		2.61	2.61237 53486 854	2.61237 53486 85

13.2 The automatic generalized ρ-algorithm

The generalized ρ-algorithm requires the knowledge of the exponent θ in (13.1). But, as described in Section 10, θ can be computed using the generalized ρ-algorithm with parameter -2.

For a given sequence (s_n) satisfying

$$s_n \sim s + n^\theta \left(c_0 + \frac{c_1}{n} + \frac{c_2}{n^2} + \dots\right), \qquad \text{as } n \to \infty \tag{13.1}$$

where $\theta < 0$ and $c_0(\neq 0), c_1, \dots$ are unknown constants independent of n. We define θ_n by

$$\theta_n = 1 + \frac{1}{\Delta\left(\dfrac{\Delta s_n}{\Delta^2 s_{n-1}}\right)}. \tag{13.12}$$

The sequence (θ_n) has the asymptotic expansion of the form

$$\theta_n \sim \theta + n^{-2}\left(t_0 + \frac{t_1}{n} + \frac{t_2}{n^2} + \dots\right), \qquad \text{as } n \to \infty, \tag{13.13}$$

where $t_0(\neq 0), t_1, \dots$ are unknown constants independent of n. Thus by applying the generalized ρ-algorithm with parameter -2 on (θ_n), we can estimate the exponent θ.

Suppose that the first n terms of a sequence (s_n) satisfying (13.1) are given. Then we put $s_0 = 0$ and define $\rho_k^{(m)}$ as follows:

$$\rho_{-1}^{(m)} = 0, \tag{13.14a}$$

$$\rho_0^{(m)} = \theta_m, \quad m \geq 1, \tag{13.14b}$$

$$\rho_k^{(m)} = \rho_{k-2}^{(m+1)} + \frac{k+1}{\rho_{k-1}^{(m+1)} - \rho_{k-1}^{(m)}}, \quad k = 1, 2, \dots. \tag{13.14c}$$

Next, we define $\alpha_n (n \geq 3)$ by

$$\alpha_n = \begin{cases} \rho_{n-3}^{(1)} & \text{if } n \text{ is odd,} \\ \rho_{n-2}^{(0)} & \text{if } n \text{ is even.} \end{cases} \tag{13.15}$$

Then we can apply the generalized ρ-algorithm with parameter α_n to (s_m).

$$\rho_{n,0}^{(m)} = s_m, \quad m = 0, \dots, n \tag{13.16a}$$

$$\rho_{n,1}^{(m)} = \frac{-\alpha_n}{\rho_{n,0}^{(m+1)} - \rho_{n,0}^{(m)}}, \quad m = 0, 1, \dots, n-1, \tag{13.16b}$$

$$\rho_{n,k}^{(m)} = \rho_{n,k-2}^{(m+1)} + \frac{k-1-\alpha_n}{\rho_{n,k-1}^{(m+1)} - \rho_{n,k-1}^{(m)}}, \quad k = 2, \dots, n; m = 0, \dots, n-k. \tag{13.16c}$$

This scheme is called the *automatic generalized ρ-algorithm*. The data flow of this scheme is as follows (case $n = 4$):

$$\begin{array}{ccccccccccc}
s_1 & & & & & & & & & & \\
s_2 & \searrow & & & & & & & & & \\
 & \searrow & & & & & & & & & \\
s_3 & \rightarrow & \theta_1 & = & \rho_0^{(1)} & \rightarrow & \rho_1^{(0)} & & & & \\
 & \searrow & & & & \searrow & & \searrow & & & \\
s_4 & \rightarrow & \theta_2 & = & \rho_0^{(2)} & \rightarrow & \rho_1^{(1)} & \rightarrow & \rho_2^{(0)} & = & \alpha_4 \\
 & & & & & & & & & & \\
s_1 & = & \rho_{4,0}^{(1)} & \rightarrow & \rho_{4,1}^{(0)} & & & & & & \\
 & & & \searrow & & \searrow & & & & & \\
s_2 & = & \rho_{4,0}^{(2)} & \rightarrow & \rho_{4,1}^{(1)} & \rightarrow & \rho_{4,2}^{(0)} & & & & \\
 & & & \searrow & & \searrow & & \searrow & & & \\
s_3 & = & \rho_{4,0}^{(3)} & \rightarrow & \rho_{4,1}^{(2)} & \rightarrow & \rho_{4,2}^{(1)} & \rightarrow & \rho_{4,3}^{(0)} & & \\
 & & & \searrow & & \searrow & & \searrow & & \searrow & \\
s_4 & = & \rho_{4,0}^{(4)} & \rightarrow & \rho_{4,1}^{(3)} & \rightarrow & \rho_{4,2}^{(2)} & \rightarrow & \rho_{4,3}^{(1)} & \rightarrow & \rho_{4,4}^{(0)}
\end{array}$$

For a given tolerance ϵ, the stopping criterion of this scheme is as follows:

(i) n is even and $|\rho_{n,n}^{(0)} - \rho_{n,n-2}^{(1)}| < \epsilon$,

(ii) n is odd and $|\rho_{n,n-1}^{(0)} - \rho_{n,n-1}^{(1)}| < \epsilon$.

Example 13.2 We apply the automatic generalized ρ-algorithm to the partial sums of $\zeta(1.5)$. We give s_n, α_n in (13.15) and $\rho_{n,2k}^{(n-2k)}$ in Table 13.2, where $k = \lfloor n/2 \rfloor$.

Table 13.2

The automatic generalized ρ-algorithm applied to $\zeta(1.5)$

n	s_n	α_n	$\rho_{n,2k}^{(n-2k)}$
1	1.00		
2	1.35		
3	1.54	−0.544	2.55
4	1.67	−0.5071	2.604
5	1.76	−0.50015	2.61217
6	1.82	−0.50001 4	2.61236 60
7	1.88	−0.50000 0052	2.61237 568
8	1.92	−0.49999 9980	2.61237 53453
9	1.96	−0.49999 99946	2.61237 53487 1
10	1.99	−0.50000 00002 3	2.61237 53487 1
11	2.02	−0.50000 00001 1	2.61237 53488 9
12	2.04	−0.50000 00002 9	2.61237 53488 2
13	2.06	−0.49999 99999 76	2.61237 53487 2
14	2.08	−0.50000 00000 93	2.61237 53486 49
∞	2.61	−0.50000 00000 00	2.61237 53486 85

A FORTRAN subroutine of the automatic generalized ρ-algorithm is given in Appendix.

14. Comparisons of acceleration methods*

In this section we compare acceleration methods for scalar sequences using wide range of slowly convergent infinite series.

14.1 Sets of sequences

Whether an acceleration method works effectively on a given sequence or not depends on the type of the asymptotic expansion of the sequence. Conversely, when we know the type of expansion we can choose a suitable method.

The sets of sequences $\mathscr{S}_\lambda$ and LOGSF are defined by

$$\mathscr{S}_\lambda = \{\ (s_n) \mid s_n \sim s + \lambda^n n^\theta \sum_{j=0}^{\infty} c_j n^{-j}, \quad c_0 \neq 0, \lambda \neq 0\}, \tag{14.1}$$

$$\mathrm{LOGSF} = \{\ (s_n) \mid \lim_{n\to\infty} \frac{s_{n+1}-s}{s_n - s} = \lim_{n\to\infty} \frac{\Delta s_{n+1}}{\Delta s_n} = 1\ \}, \tag{14.2}$$

respectively. For $(s_n) \in \mathscr{S}_\lambda$, (s_n) converges if $0 < |\lambda| < 1$, and (s_n) diverges if $|\lambda| > 1$.

We consider subsets of $\mathscr{S}_\lambda$ and LOGSF as follows:

$$\mathrm{Alt} = \{\ (s_n) \mid s_n \sim s + (-1)^n n^\theta \sum_{j=0}^{\infty} c_j n^{-j}, \quad c_0 \neq 0, \theta < 0\}, \tag{14.3}$$

$$\mathscr{L}_1 = \{\ (s_n) \mid s_n \sim s + n^\theta \sum_{j=0}^{\infty} c_j n^{-j}, \quad c_0 \neq 0, -\theta \in \mathbb{N}\}, \tag{14.4}$$

$$\mathscr{L}_2 = \{\ (s_n) \mid s_n \sim s + n^\theta \sum_{j=0}^{\infty} c_j n^{-j}, \quad c_0 \neq 0, \theta < 0\}, \tag{14.5}$$

$$\mathscr{L}_3 = \{\ (s_n) \mid s_n \sim s + \sum_{i=1}^{m} n^{\theta_i} \sum_{j=0}^{\infty} c_{ij} n^{-j}, \quad 0 > \theta_1 > \theta_2 > \cdots > \theta_m\}, \tag{14.6}$$

$$\mathscr{L}_4 = \{\ (s_n) \mid s_n \sim s + n^\theta \sum_{j=0}^{\infty} \frac{a_j + b_j \log n}{n^j}, \quad \theta < 0\}, \tag{14.7}$$

$$\mathscr{L}_5 = \{\ (s_n) \mid s_n \sim s + n^\theta (\log n)^\tau \sum_{j=0}^{\infty}\sum_{i=0}^{\infty} \frac{c_{ij}}{(\log n)^i n^j}, \quad \theta < 0 \text{ or } \theta = 0 \text{ and } \tau < 0\}. \tag{14.8}$$

*The material in this section is an improvement of the author's informal paper: N. Osada, Asymptotic expansion and acceleration methods for certain logarithmically convrgent sequences, *RIMS Kokyuroku* 676(1988), pp.195-207.

We remark that Alt$= \mathscr{S}_{-1}$ and $\mathscr{L}_1 = \mathscr{S}_1$. We also remark that

$$\mathscr{L}_1 \subset \mathscr{L}_2 \subset \mathscr{L}_3 \subset \text{LOGSF}, \quad \mathscr{L}_2 \subset \mathscr{L}_4 \subset \text{LOGSF}, \quad \mathscr{L}_2 \subset \mathscr{L}_5 \subset \text{LOGSF}. \tag{14.9}$$

14.2 Test series

We take up infinite series for each set. Test series are shown in Table 14.1. Some of them, No. 1,3,4,5, and 8 were tested by Smith and Ford[54].

Table 14.1

Test series

set	No.	partial sum of series	sum	asymptotic expansion
$\mathscr{S}_\lambda$	1	$\sum_{i=1}^{n} \frac{(0.8)^i}{i}$	$\log 5$	$\theta = -1,\ \lambda = 0.8$
	2	$\sum_{i=1}^{n} (-1)^{i-1} \frac{4^i}{i}$	$\log 5$	$\theta = -1,\ \lambda = 4$
Alt	3	$\sum_{i=1}^{n} (-1)^{i-1} \frac{1}{i}$	$\log 2$	$\theta = -1$ Example 3.1
	4	$\sum_{i=1}^{n} (-1)^{i-1} \frac{1}{\sqrt{i}}$	0.60489 86434 21630	$\theta = -0.5$ Example 3.3
$\mathscr{L}_1$	5	$\sum_{i=1}^{n} \frac{1}{i^2}$	$\pi^2/6$	$\theta = -1$ Example 3.4
	6	$\sum_{i=1}^{n} \frac{1}{i^3}$	1.20205 69031 59594	$\theta = -2$ Example 3.4
$\mathscr{L}_2$	7	$\sum_{i=1}^{n} \frac{1}{i\sqrt{i}}$	2.61237 53486 85488	$\theta = -0.5$ Example 3.4
	8	$\sum_{i=1}^{n} \left(i + e^{1/i}\right)^{-\sqrt{2}}$	1.71379 67355 40301	$\theta = 1 - \sqrt{2}$ Example 3.5
$\mathscr{L}_3$	9	$\sum_{i=1}^{n} \frac{\sqrt{i}+1}{i^2}$	4.25730 94155 33714	$\theta_1 = -0.5,\ \theta_2 = -1$
$\mathscr{L}_4$	10	$\sum_{i=1}^{n} \frac{\log i}{i^2}$	0.93754 82543 15844	$\theta = -1$ Example 3.6
	11	$\sum_{i=1}^{n} \frac{\log i}{i\sqrt{i}}$	3.93223 97374 31101	$\theta = -0.5$ Example 3.6
$\mathscr{L}_5$	12	$\sum_{i=2}^{n+1} \frac{1}{i(\log i)^2}$	2.10974 28012 36892	$\theta = -1$ Example 3.7

14.3 Numerical results

Let (s_n) converge to s or diverge from s. Let T be a sequence transformation (t_n) the transformed sequence by T, where t_n depends on $s_1, \ldots, s_n$ but does not on $s_{n+k}, k > 0$.

The maximum significant digits from m terms is defined by

$$\max_{1\le n\le m} \{ -\log_{10} |t_n - s| \}. \tag{14.10}$$

Acceleration methods taken up in this section are as follows. The ϵ-algorithm, the Levin u-, v-, and t-transforms, the $d^{(2)}$- and $d^{(3)}$-transforms, the iterated Aitken δ^2 process, the automatic modified Aitken δ^2 formula, the W transform, the ρ-algorithm, and the automatic generalized ρ-algorithm. All methods require no knowledge of asymptotic expansion of objective sequence. We compare with the quantities listed in Table 14.2.

Table 14.2

Acceleration methods

acceleration method	definition	quantity
ϵ-algorithm	(8.10)	$\epsilon_{2k}^{(n-2k)}$, $k = \lfloor (n-1)/2 \rfloor$
Levin u-transform	§9.1	$T_{n-1}^{(1)}$
Levin v-transform	§9.1	$T_{n-2}^{(1)}$
Levin t-transform	§9.1	$T_{n-2}^{(1)}$
$d^{(2)}$-transform	§9.3	$E_{2l-2}^{(n-2l+2)}$, $l = \lfloor n/2 \rfloor$
$d^{(3)}$-transform	§9.3	$E_{3m-3}^{(n-3m+3)}$, $m = \lfloor n/3 \rfloor$
iterated Aitken δ^2 process	(10.10)	$T_k^{(n-2k)}$, $k = \lfloor (n-1)/2 \rfloor$
automatic modified Aitken δ^2 formula	§10.3	$s_{n,l}^{(n-l)}$, $l = \lfloor n/2 \rfloor$
iterated Lubkin W transform	(11.17)	$W_p^{(n-3p)}$, $p = \lfloor (n-1)/3 \rfloor$
ρ-algorithm	(12.3)	$\rho_{2l}^{(n-2l)}$, $l = \lfloor n/2 \rfloor$
automatic generalized ρ-algorithm	§13.2	$\rho_{n,2k}^{(n-2k)}$, $k = \lfloor (n-1)/2 \rfloor$

For each acceleration method we show the maximum significant digits from 20 terms of each test series in Table 14.3. In table 14.3, the “number of terms” is abbreviated to “NT”, and the “number of significant digits” is abbreviated to “SD”. Numerical computations reported in this section were carried out on the NEC ACOS-610 computer in double precision with approximately 16 digits.

Table 14.3

The maximum significant digits

		partial sum		ϵ-algorithm		Levin u		Levin v	
		NT	SD	NT	SD	NT	SD	NT	SD
$\mathscr{S}_\lambda$	1	20	2.72	20	8.01	19	10.78	20	10.40
	2	diverge		20	7.81	16	10.78	16	11.58
Alt	3	20	1.61	20	15.11	15	15.90	15	15.86
	4	20	0.96	20	15.74	14	15.56	14	15.56
$\mathscr{L}_1$	5	20	1.31	20	2.06	14	11.46	12	9.64
	6	20	2.92	20	4.17	12	11.49	12	11.01
$\mathscr{L}_2$	7	20	0.35	20	0.78	10	9.01	12	8.55
	8	20	0.17	20	0.53	13	8.44	13	7.58
$\mathscr{L}_3$	9	20	0.31	20	0.75	20	2.58	19	2.96
$\mathscr{L}_4$	10	20	0.71	20	1.62	20	3.11	20	3.47
	11	20	−0.35	20	−0.01	20	0.98	20	1.24
$\mathscr{L}_5$	12	20	0.49	20	0.69	20	1.07	20	1.13

Table 14.3 (Continued)

		Levin t		$d^{(2)}$-trans		$d^{(3)}$-trans		Aitken δ^2	
		NT	SD	NT	SD	NT	SD	NT	SD
$\mathscr{S}_\lambda$	1	20	10.49	20	7.16	20	6.45	17	9.62
	2	16	10.54	19	8.36	19	6.18	17	11.77
Alt	3	14	15.95	20	13.85	20	12.84	19	16.08
	4	15	15.54	20	15.90	19	15.31	16	16.08
$\mathscr{L}_1$	5	20	2.28	18	11.29	20	10.69	20	3.18
	6	20	4.60	16	12.44	16	12.20	19	5.72
$\mathscr{L}_2$	7	20	0.89	14	9.82	19	10.20	19	1.42
	8	20	0.16	15	10.76	18	9.63	7	1.05
$\mathscr{L}_3$	9	20	0.87	12	5.54	15	8.54	19	1.35
$\mathscr{L}_4$	10	20	1.52	13	7.07	20	6.99	15	1.33
	11	20	0.03	13	4.98	17	4.80	20	−0.18
$\mathscr{L}_5$	12	20	0.74	19	1.29	20	1.59	20	0.81

Table 14.3 (Continued)

		aut mod δ^2		Lubkin W		ρ-algorithm		aut gen ρ	
		NT	SD	NT	SD	NT	SD	NT	SD
$\mathscr{S}_\lambda$	1	13	6.76	20	10.69	decelerate		20	8.06
	2	18	10.52	19	11.13	decelerate		20	7.99
Alt	3	19	16.01	19	16.26	decelerate		19	14.51
	4	19	15.50	17	15.54	decelerate		19	14.59
$\mathscr{L}_1$	5	12	11.02	15	9.66	18	11.66	20	12.18
	6	19	12.53	18	10.92	20	12.81	14	13.05
$\mathscr{L}_2$	7	11	9.79	15	8.34	18	1.52	9	10.51
	8	19	9.70	13	8.38	20	1.03	17	11.29
$\mathscr{L}_3$	9	19	3.04	20	4.43	17	1.54	18	3.08
$\mathscr{L}_4$	10	16	3.03	12	2.25	20	3.03	19	3.58
	11	14	0.75	20	0.46	18	0.44	17	1.27
$\mathscr{L}_5$	12	20	1.13	20	1.31	20	0.89	20	1.15

14.4 Extraction

As Delahahe and Germain-Bonne[14] proved, there is no acceleration method that can accelerate all sequences belonging to LOGSF. However, if a logarithmic sequence (s_n) satisfies the asymptotic form

$$s_n = s + O(n^\theta), \tag{14.11}$$

or

$$s_n = s + O(n^\theta (\log n)^\tau), \tag{14.12}$$

where $\theta < 0$, then the subsequence (s_{2^n}) of (s_n) satisfies

$$s_{2^n} = s + O((2^\theta)^n), \tag{14.13}$$

or

$$s_{2^n} = s + O((2^\theta)^n n^\tau), \tag{14.14}$$

respectively. Both (14.13) and (14.14) converges linearly to s with contraction ratio 2^{θ}. In particular, if a sequence (s_n) belongs to $\mathscr{L}_3$, the subsequence (s_{2^n}) of (s_n) satisfies the asymptotic expansion of the form

$$s_{2^n} \sim s + \sum_{j=1}^{\infty} c_j \lambda_j^n, \tag{14.15}$$

where c_j and $1 > \lambda_1 > \lambda_2 > \cdots > 0$ are constants. By Theorem 8.4, the ϵ-algorithm and the iterated Aitken δ^2 process can accelerate the subsequence efficiently.

If (s_n) satisfies

$$s_n = s + O((\log n)^{\tau}), \tag{14.16}$$

where $\tau < 0$, then

$$s_{2^n} = s + O(n^{\tau}), \tag{14.17}$$

that is, (s_{2^n}) converges logarithmically. Therefore the d-transform or the automatic generalized ρ-algorithm are expected to accelerate the convergence of (s_{2^n}).

In Table 14.4, we take up the Levin t-transform, the ϵ-algorithm, the iterated Aitken δ^2 process, and the $d^{(2)}$-transform as acceleration methods, and we apply to the last 6 series in Table 14.1. Though we do not list in Table 14.4, the Levin t-transform is slightly better than the Levin u-, v-transforms. The $d^{(2)}$-transform is slightly better than the $d^{(3)}$-transform.

Table 14.4

The maximum significant digits

series	partial sum		Levin t		ϵ-algorithm		Aitken δ^2		$d^{(2)}$	
No.	NT	SD	NT	SD	NT	SD	NT	SD	NT	SD
7	16384	1.81	16384	6.49	16384	12.02	4096	11.45	16384	6.96
8	16384	1.36	16384	5.28	16384	10.92	8192	10.77	8192	4.31
9	16384	1.80	16384	5.81	16384	9.56	16384	8.09	16384	6.32
10	16384	3.18	16384	8.52	4096	9.86	8192	7.41	16384	8.91
11	16384	0.74	16384	5.14	8192	6.69	2048	5.97	8192	4.14
12	16384	0.99	16384	1.67	8192	1.49	512	1.45	16384	4.63

Table 14.5

The maximum significant digits for $\sum_{i=2}^{2^n+1} \frac{1}{i(\log i)^2}$.

series	Levin u		Lubkin W		aut. gen. ρ		aut. mod. δ^2		$d^{(3)}$	
No.	NT	SD	NT	SD	NT	SD	NT	SD	NT	SD
12	16384	2.60	4096	3.48	8192	3.91	2048	3.11	8192	4.04

14.5 Conclusions

Table 14.3 show that the best available methods are the $d^{(2)}$- and $d^{(3)}$-transforms. For $\mathscr{S}_\lambda$, all tested methods except the ρ-algorithm work well. For $\mathscr{L}_1$ and $\mathscr{L}_2$, the automatic generalized ρ-algorithm is the best.

The Levin u-, v-transforms, the automatic generalized ρ-algorithm, and the automatic modified Aitken δ^2 formula, and Lubkin's W transforms are good methods. The automatic generalized ρ-algorithm and the automatic modified Aitken δ^2 formula are generalizations of the sequence transformation

$$s_n \mapsto s_n - \frac{\theta - 1}{\theta} \frac{\Delta s_n \Delta s_{n-1}}{\Delta^2 s_{n-1}}, \tag{14.18}$$

therefore they perform similarly.

The ϵ-algorithm, the Levin t-transform, and the iterated Aitken δ^2 process perform similarly. Because these three methods are extensions of the Aitken δ^2 process.

For a sequence (s_n) belonging to $\mathscr{L}_3$ or $\mathscr{L}_4$, any acceleration method listed in Table 14.2 cannot give high accurate result. However, when we apply the ϵ-algorithm or the iterated Aitken δ^2 process to the subsequence (s_{2^n}), we can obtain good result.

For the last series, the $d^{(2)}$-, $d^{(3)}$-transforms, the automatic generalized ρ-algorithm, Lubkin's W transform, and the automatic modified Aitken δ^2 formula accelerate the convergence of (s_{2^n}).

15. Application to numerical integration

Infinite integrals and improper integrals usually converge slowly. Such an integral yields a slowly convergent sequence or an infinite series. Various acceleration methods have been applied to such slowly convergent integrals. In this section we deal with the application of acceleration methods to numerical integration.

15.1 Introduction

Acceleration methods are applied to numerical integration in the following ways.

I. The semi-infinite integral $I = \int_a^\infty f(x)dx$.

Let $a = x_0 < x_1 < x_2 < \ldots$ be an increasing sequence diverging to ∞. Then I becomes to an infinite series

$$I = \sum_{j=1}^{\infty} \int_{x_{j-1}}^{x_j} f(x)dx = \sum_{j=1}^{\infty} I_j. \tag{15.1}$$

Let S_n be the nth partial sum of (15.1).

I-1. Suppose that $f(x)$ converges monotonically to zero as $x \to \infty$. Let $a = x_0 < x_1 < x_2 < \ldots$ be equidistant points. Then S_n sometimes either converges linearly or satisfies

$$S_n \sim I + n^\theta \sum_{j=1}^{\infty} c_j n^{-j}. \tag{15.2}$$

As we described in the preceding section, we can accelerate (S_n) or (S_{2^n}).

I-2. Suppose that $f(x)$ is a product of an oscillating function and a positive decreasing function. Let $x_1 < x_2 < \ldots$ be zeros of $f(x)$. Then the infinite series (15.1) becomes to alternating series. Thus it is easy to accelerate (S_n).

The first proposer of this method was I. M. Longman[28]. In 1956, he applied the Euler transformation to semi-infinite integrals involving Bessel functions.

In this paper we consider $f(x) = g(x)\sin\omega x$ or $f(x) = g(x)\cos\omega x$, where $g(x)$ converges monotonically to zero as $x \to \infty$ and $\omega > 0$ is a known constant.

II. The finite integral $I = \int_a^b f(x)dx$.

Let S_n be an approximation of I by applying n panels compound integral formula such as n panels midpoint rule. As we described in Section 4, S_n has often the asymptotic expansion of the form

$$S_n \sim I + \sum_{j=0}^{\infty} c_j n^{\theta_j}, \tag{15.3}$$

or

$$S_n \sim I + n^{\theta} \sum_{j=0}^{\infty} \frac{a_j + b_j \log n}{n^j} + n^{\tau} \sum_{j=0}^{\infty} \frac{c_j + d_j \log n}{n^j}. \tag{15.4}$$

If $f(x)$ is of class C^{∞} in $[a, b]$ and if the quadrature formula is either the trapezoidal rule or the midpoint rule, then $\theta_j = -2j - 2$ in (15.3).

II-1. When θ_j's in (15.3), or θ and τ in (15.4) are known, by applying the Richardson extrapolation or the E-algorithm to (S_n) or (S_{2^n}), the convergence of (S_n) is accelerated.

In 1955, W. Romberg[45] applied the Richardson extrapolation to (S_{2^n}) when $\theta_j = -2j - 2$ in (15.3). Since 1961, many authors such as I. Navot[33] and H. Rutishauser[46] applied it to improper integrals, see Joyce's survey paper[22].

II-2. When the asymptotic scale in the asymptotic expansion is unknown, acceleration methods taken up in the previous section are applied to (S_n) or (S_{2^n}). As we saw in the previous section, when there are integers i and j such that $\theta_i - \theta_j$ is not integer, we can not obtain a high accurate result by applying an acceleration method to (S_n) itself. However, for (15.3) and (15.4) it is expected to obtain a good result by applying the ϵ-algorithm or the iterated Aitken δ^2 process to (S_{2^n}). The first proposer of this method was C. Brezinski[6,7]. In 1970 and 1971, he applied the ρ-algorithm with parameter to (S_{2^n}) and the ϵ-algorithm when $\theta_j = -2j - 2$ in (15.3). Subsequently many authors such as D. K. Kahaner[23] applied acceleration methods to finite integrals.

III. For another way of applying extrapolation methods to numerical integration, see Brezinski and Redivo Zaglia[11, p.366–386] and Rabinowitz[43].

15.2 Application to semi-infinite integrals

We apply the above method I-1 to integrals listed in Table 15.1.

Table 15.1

Semi-infinite integrals

with a monotonically decreasing integrand

No.	integral	exact	$S_n = \int_0^{2n} f(x)dx$
1	$\int_0^\infty e^{-x}dx$	1	geometric series
2	$\int_0^\infty x^2 e^{-x}dx$	2	linearly convergent sequence
3	$\int_0^\infty \frac{dx}{1+x^2}$	$\frac{\pi}{2}$	$S_n \sim \frac{\pi}{2} + \sum_{j=1}^\infty \frac{c_j}{n^{2j-1}}$

We take $x_j = 2j$ and compute $\int_{2j-2}^{2j} f(x)dx$ by the Romberg method. Acceleration methods that we apply are the Levin u-transform, the ϵ-algorithm, Lubkin's W transform, the iterated Aitken δ^2-process, the $d^{(2)}$-transform and the automatic generalized ρ-algorithm. The tolerances are $\epsilon = 10^{-6}$ and 10^{-12}. For $\int_0^\infty dx/(1+x^2)$, we take $\epsilon = 10^{-9}$ instead of 10^{-12}. The stopping criterion is $|T_n - T_{n-1}| < \epsilon$, where (T_n) is the accelerated sequence. The results are shown in Table 15.2. Throughout this section, the number of terms is abbreviated to "T", and the number of functional evaluations is abbreviated to "FE".

Table 15.2

Number of terms, functional evaluations, and errors

$\epsilon = 10^{-6}$

	Levin	u-transform			ϵ-algorithm			Lubkin's W transform	
No.	T	FE	error	T	FE	error	T	FE	error
1	5	43	6.16×10^{-9}	4	36	6.19×10^{-9}	5	43	6.19×10^{-9}
2	8	96	-1.24×10^{-9}	8	96	-6.60×10^{-10}	8	96	5.05×10^{-8}
3	10	102	6.46×10^{-9}			failure	9	95	9.63×10^{-7}

	Aitken	δ^2 process			$d^{(2)}$-transform			automatic generalized ρ	
No.	T	FE	error	T	FE	error	T	FE	error
1	3	29	6.16×10^{-9}	7	57	6.16×10^{-9}	5	43	6.16×10^{-9}
2	8	96	1.91×10^{-9}	8	96	-6.58×10^{-10}	8	96	7.75×10^{-10}
3			failure	10	102	-3.24×10^{-8}	9	95	-4.17×10^{-8}

$\epsilon = 10^{-12}$

	Levin		u-transform		ϵ-algorithm		Lubkin's W transform		
No.	T	FE	error	T	FE	error	T	FE	error
1	5	187	-2.78×10^{-16}	4	156	-8.33×10^{-17}	5	187	-8.33×10^{-17}
2	11	325	1.29×10^{-14}	8	280	1.40×10^{-14}	12	340	8.88×10^{-16}

	Aitken		δ^2 process		$d^{(2)}$-transform		automatic generalized ρ		
No.	T	FE	error	T	FE	error	T	FE	error
1	3	125	-8.33×10^{-17}	7	233	-9.71×10^{-17}	5	187	-8.33×10^{-17}
2	11	325	4.93×10^{-14}	9	295	1.20×10^{-14}	10	310	1.38×10^{-14}

$\epsilon = 10^{-9}$

	Levin		u-transform		$d^{(2)}$-transform		automatic generalized ρ		
No.	T	FE	error	T	FE	error	T	FE	error
3	14	226	6.20×10^{-11}	14	226	-1.29×10^{-11}	14	226	-1.21×10^{-10}

When $\epsilon = 10^{-12}$, all methods except the automatic generalized ρ-algorithm (T= 27, FE= 557, error= 3.08×10^{-13}), failure on $\int_0^\infty dx/(1+x^2)$.

Next we apply the above method I-2 to integrals listed in Table 15.3. All integrals were tested by Hasegawa and Torii[19].

Table 15.3

Semi-infinite integrals with an oscillatory integrand

No.	integral	exact
1	$\int_0^\infty e^{-x} \cos x dx$	0.5
2	$\int_0^\infty \frac{x \sin x}{x^2+1} dx$	$\pi/(2e)$
3	$\int_0^\infty \frac{\cos x}{x^2+1} dx$	$\pi/(2e)$
4	$\int_0^\infty \frac{\cos x}{\sqrt{x^2+1}} dx$	0.42102 44382 40708
5	$\int_1^\infty \frac{\sin x}{x^2} dx$	0.50406 70619 06919

We compute integrals between two consecutive zeros by the Romberg method. Acceleration methods considered here are the Levin u-transform, the ϵ-algorithm, and the $d^{(2)}$-transform. These methods require no knowledge of asymptotic expansion of the integrand or the integral. The tolerances are $\epsilon = 10^{-6}$ and 10^{-12}. The results are shown in Table 15.4.

Table 15.4

Number of terms, functional evaluations, and errors

$\epsilon = 10^{-6}$

	Levin		u-transform		ϵ-algorithm			$d^{(2)}$-transform	
No.	T	FE	error	T	FE	error	T	FE	error
1	5	81	-7.38×10^{-9}	4	73	-7.38×10^{-9}	5	81	-7.44×10^{-9}
2	8	129	1.54×10^{-7}	10	145	1.13×10^{-7}	9	137	1.46×10^{-7}
3	7	89	1.62×10^{-7}	9	105	7.37×10^{-9}	8	97	-7.75×10^{-8}
4	8	145	-3.01×10^{-8}	10	177	-9.09×10^{-8}	9	161	-1.21×10^{-7}
5	7	89	-5.01×10^{-8}	9	105	4.77×10^{-8}	8	97	-1.05×10^{-8}

$\epsilon = 10^{-12}$

	Levin	u	transform		ϵ-algorithm			$d^{(2)}$-transform	
No.	T	FE	error	T	FE	error	T	FE	error
1	5	353	-2.50×10^{-16}	4	321	4.16×10^{-17}	6	358	-2.78×10^{-16}
2	12	641	-4.53×10^{-14}	18	833	-9.40×10^{-15}	14	705	2.16×10^{-14}
3	12	897	1.82×10^{-14}	17	1217	8.10×10^{-14}	13	961	1.56×10^{-13}
4	12	897	-2.01×10^{-13}	18	1281	-6.90×10^{-14}	14	1025	4.33×10^{-15}
5	12	1025	-9.01×10^{-15}	17	1345	3.68×10^{-14}	12	1025	-1.71×10^{-13}

The Levin v- and t-transforms perform similar to the Levin u-transform. The iterated Aitken δ^2 process and Lubkin's W trasnsform are slightly better than the ϵ-algorithm. The $d^{(2)}$-transform is better that the $d^{(3)}$-transform. The best acceleration methods we tested for semi-infinite oscillating integrals are the Levin transforms.

These results are less than Hasegawa and Torii's results[19], but they are available in practice because they require no knowledge of the integrand.

15.3 Application to improper integrals

We apply the above method II-2 to integrals listed in Table 15.5.

Table 15.5

Improper integrals

No.	integral	exact	$S_n = h\sum_{i=1}^{n} f(a+(2i-1)h),\ \ h=(b-a)/n$
1	$\int_0^1 \sqrt{x}dx$	$2/3$	$S_n \sim I + c_0 n^{-1.5} + \sum_{j=1}^{\infty} c_j n^{-2j}$
2	$\int_0^1 \frac{dx}{\sqrt{x}}$	2	$S_n \sim I + c_0 n^{-0.5} + \sum_{j=1}^{\infty} c_j n^{-2j}$
3	$B(\frac{2}{3}, \frac{1}{3})$	$2\pi/\sqrt{3}$	$S_n \sim I + \sum_{j=1}^{\infty}(c_{2j-1} n^{-1/3-j+1} + c_{2j} n^{-2/3-j+1})$
4	$\int_0^1 \frac{\log x}{\sqrt{x}}dx$	-4	$S_n \sim I + \sum_{j=0}^{\infty}(a_j n^{-1/2-j} + b_j n^{-1/2-j}\log n$ $+c_j n^{-j-1})$

We use the midpoint rule as the quadrature formula. Acceleration methods are the Levin u-transform, the ϵ-algorithm, Lubkin's W transform, the iterated Aitken δ^2-process, the $d^{(2)}$-transform and the automatic generalized ρ-algorithm. The tolerance is $\epsilon = 10^{-6}$ and the maximum number of terms is 15. The results are shown in Table 15.6.

Table 15.6

Number of terms, functional evaluations, and errors

$\epsilon = 10^{-6}$

	Levin	u-transform		ϵ-algorithm			Lubkin's W transform		
No.	T	FE	error	T	FE	error	T	FE	error
1	8	255	3.70×10^{-9}	7	127	-4.86×10^{-9}	8	255	3.30×10^{-9}
2	13	8191	-2.09×10^{-7}	8	255	1.78×10^{-8}	11	2047	-2.46×10^{-10}
3	15	32767	7.21×10^{-6}	13	8191	1.13×10^{-7}		failure	
4	14	16383	-4.90×10^{-7}	11	2047	3.91×10^{-9}	15	32767	2.04×10^{-6}

	Aitken δ^2 process			$d^{(2)}$-transform			automatic generalized ρ		
No.	T	FE	error	T	FE	error	T	FE	error
1	6	63	-4.22×10^{-7}	8	255	-1.20×10^{-7}	8	255	2.24×10^{-11}
2	9	511	9.10×10^{-10}	14	16383	-3.36×10^{-8}	15	32767	-8.26×10^{-9}
3	15	32767	-6.56×10^{-6}	15	32767	1.64×10^{-5}	15	32767	-9.97×10^{-6}
4	15	32767	-2.01×10^{-6}	15	32767	-1.06×10^{-7}	14	16383	3.20×10^{-7}

The ϵ-algorithm is the best. For the tolerance $\epsilon = 10^{-9}$, only the ϵ-algorithm succeeds on all integrals listed in Table 15.5, provided that the number of terms is less than or equals to 15.

CONCLUSIONS

In this paper we studied acceleration methods for slowly convergent scalar sequences from asymptotic view point, and applied these methods to numerical integration.

In conclusion, our opinion is as follows.

1. Suppose that a sequnce (s_n) has an asymptotic expansion of the form

$$s_n \sim s + \sum_{j=1}^{\infty} c_j g_j(n). \tag{1}$$

Let T be a sequence transformation and $(t_n) = T(s_n)$. If we know an asymptotic scale $(g_j(n))$, then we can often obtain an asymptotic formula

$$t_n = s + O(g(n)). \tag{2}$$

2. By the above 1, if we know $(g_j(n))$, we can choose a suitable acceleration method for (s_n).

3. We show the most suitable methods in the following Table 1. We append the number of theorem giving the asymptotic formula.

4. For a logarithmically convergent sequence (s_n), we can usually obtain higher accuracy when we apply to (s_{2^n}).

5. There is no all-purpose acceleration method. The best method of all we treated is the d-transform, and the second best method is the automatic generalized ρ-algorithm. In application, we can usually determine a type of an aymptotic expansion of an objective sequence. For example, when we apply the midpoint rule to an improper integral with endpoint singularity, the objective sequence has the asymptotic expansion of the form

$$s_n \sim s + n^{\theta} \sum_{j=0}^{\infty} c_j n^{-j} + n^{\tau} \sum_{j=0}^{\infty} d_j n^{-j}. \tag{3}$$

In such a case, we recommend the methods listed in Table 1.

6. If (s_n) satisfies (3) and the ϵ-algorithm is applied to (s_{2^n}), $(\epsilon_{2k}^{(n-2k)})$ gives high accurate result. In particular, it is a good method that the ϵ-algorithm is applied to M_{2^n}, where M_{2^n} is the 2^npanels midpoint rule.

Table 1

Suitable acceleration methods

asymptotic expansion $s_n - s$		asymptotic scale known	asymptotic scale unknown
$\sum_{j=1}^{\infty} c_j \lambda_j^n$	(s_n)	Richardson extrapolation[1)]	ϵ-algorithm[2)]
$\lambda^n n^\theta \sum_{j=0}^{\infty} c_j n^{-j}$	(s_n)	E-algorithm[3)]	Levin transforms[4)]
$n^\theta \sum_{j=0}^{\infty} c_j n^{-j}$	(s_n)	generalized ρ-algorithm[5)] modified Aitken δ^2 formula[6)]	automatic generalized ρ-algorithm
	(s_{2^n})	Richardson extrapolation[1)]	ϵ-algorithm[2)]
$\sum c_{ij} n^{\theta_i - j}$	(s_n)	E-algorithm[3)]	d-transform
	(s_{2^n})	Richardson extrapolation[1)]	ϵ-algorithm[2)]
$n^\theta \sum_{j=0}^{\infty} (a_j + b_j \log n) n^{-j}$	(s_n)	E-algorithm[3)]	d-transform
	(s_{2^n})	E-algorithm[3)]	ϵ-algorithm[2)]
$\sum_{i,j} c_{ij} (\log n)^{\tau - i} n^{\theta - j}$	(s_n)	E-algorithm[3)]	
	(s_{2^n})	E-algorithm[3)]	d-transform

[1)] Formula (7.37), [2)] Theorem 8.4, [3)] Theorem 6.2, [4)] Theorem 9.1, [5)] Theorem 13.2, [6)] Theorem 10.5.

We raise the following questions.

1. Find an efficient algorithm for the automatic generalized ρ-algorithm.

2. Find an acceleration method that works well on a sequence (s_n) satisfying

$$s_n \sim s + \sum_{i=1}^{m} n^{\theta_i} \sum_{j=0}^{\infty} \frac{a_{ij} + b_{ij} \log n}{n^j}. \tag{4}$$

Such sequences occur in numerical integration.

3. Find an acceleration method that works well on a sequence (s_n) satisfying

$$s_n \sim s + n^\theta \sum_{j=0}^{\infty} \sum_{i=0}^{j} \frac{c_{i,j} (\log n)^i}{n^j}. \tag{5}$$

Such sequences occur in singular fixed point problems.

4. Extend results for scalar sequences to vector sequences. In particular, study acceleration methods for logarithmically convergent vector sequences.

Acknowledgements

I would like to express my deepest gratitude to Professor T. Torii of Nagoya University for his constant guidance and encouragement in the preparation of this thesis.

I would like to heartly thank Professor T. Mitsui of Nagoya University for his advice which improved this thesis.

I am indebted to Professor S. Kuwabara of Nagoya University for his taking the trouble to referee this thesis.

I would like to give my thank Professor C. Brezinski of Unversité des Sciences et Technologies de Lille for the invitation to an international congress in Luminy, September 1989, that made the turning point of my research.

I am grateful to Professor M. Iri of the University of Tokyo and Professor K. Nakashima of Waseda University for their help and encouragement. I would also like to thank Professor I. Ninomiya of Chubu University who taught me fundamentals of numerical analysis on occasion at various meetings when I began to study it.

REFERENCES

[1] A.C. Aitken, On Bernoulli's numerical solution of algebraic equations, *Proc. Roy. Soc. Edinburgh* Ser. A 46(1926), 289–305.

[2] W.G. Bickley and J.C.P. Miller, The numerical summation of slowly convergent series of positive terms, *Philos. Mag.* 7th Ser. 22(1936), 754–767.

[3] P. Bjørstad, G. Dahlquist and E. Grosse, *Extrapolation of asymptotic expansions by a modified Aitken δ^2-formula*, STAN-CS-79-719 (Computer Science Dept., Stanford Univ., 1979).

[4] P. Bjørstad, G. Dahlquist and E. Grosse, Extrapolation of asymptotic expansions by a modified Aitken δ^2-formula, *BIT* 21(1981), 56–65.

[5] N. Bourbaki, *Éléments de mathématique, Fonctions d'une variable réelle*, (Hermann, Paris, 1961).

[6] C. Brezinski, Application du ρ-algorithme à la quadrature numérique, *C. R. Acad. Sc. Paris* t.270(1970), 1252–1253.

[7] C. Brezinski, Etudes sur les ϵ- et ρ-algorithmes, *Numer. Math.* 17(1971), 153–162.

[8] C. Brezinski, Accélération de suits à convergence logarithmique *C. R. Acad. Sc. Paris* t.273(1971), 727–730.

[9] C. Brezinski, *Accélération de la convergence en analyse numérique* Lecture notes in Math., 584 (Springer, Berlin, 1977).

[10] C. Brezinski, A general extrapolation algorithm, *Numer. Math.* 35(1980), 175–187.

[11] C. Brezinski and M. Redivo Zaglia, *Extrapolation methods: theory and practice*, (Elsevier, Amsterdam, 1991).

[12] N. G. de Bruijn, *Asymptotic Methods in Analysis* (Dover Publ., New York, 1981).

[13] F. Cordellier, Caractérisation des suites que la première étape du θ-algorithme transforme en suites constantes, *C. R. Acad. Sc. Paris* t.284(1971), 389–392.

[14] J. P. Delahaye and B. Germain-Bonne, Résultats négatifs en accélération de la convergence, *Numer. Math.* 35(1980), 443–457.

[15] J.E. Drummond, Summing a common type of slowly convergent series of positive terms, *J. Austral. Math. Soc.* 19 Ser. B(1976), 416–421.

[16] W.F. Ford and A. Sidi, An algorithm for a generalization of the Richardson extrapolation process, *SIAM J. Numer. Anal.* 24(1987), 1212-1232.

[17] H.L. Gray and W.D. Clark, On a class of nonlinear transformations and their applications to the evaluation of infinite series, *J. Res. Nat. Bur. Stand.* 73B(1969), 251–274.

[18] S. Gustafson, A method of computing limit values, *SIAM J. Numer. Anal.* 10(1973), 1080–1090.

[19] T. Hasegawa and T. Torii, Indefinite integration of oscillatory functions by the Chebyshev series expansion, *J. Comput. Appl. Math.* 17(1987), 21–29.

[20] T. Håvie, Generalized Neville type extrapolation schemes, *BIT* 19(1979), 204–213.

[21] P. Henrici, *Elements of numerical analysis*, (John Wiley and Sons, New York, 1964).

[22] D.C. Joyce, Survey of extrapolation processes in numerical analysis, *SIAM Rev.* 13(1971), 435–490.

[23] D. K. Kahaner, Numerical quadrature by the ϵ-algorithm, *Math. Comp.* 26(1972), 689–693.

[24] K. Knopp, *Theory and application of infinite series, 2nd. English ed.*, (Dover Publ., New York, 1990).

[25] C. Kowalewski, Accélération de la convergence pour certaines suites à convergence logarithmique, *Lecture note in Math.* 888(1981), 263–272.

[26] D. Levin, Development of non-linear transformations for improving convergence of sequences, *Intern. J. Computer Math.* 3(1973), 371–388.

[27] D. Levin and A. Sidi, Two new classes of nonlinear transformations for accelerating the convergence of infinite integrals and series, *Appl. Math. and Comp.* 9(1981), 175–215.

[28] I. M. Longman, Note on a method for computing infinite integrals of oscillatory functions, *Proc. Cambridge Phil. Soc.* 52(1956), 764–768.

[29] S. Lubkin, A method of summing infinite series, *J. Res. Nat. Bur. Stand.* 48(1952), 228–254.

[30] J.N. Lyness and B.W. Ninham, Numerical quadrature and asymptotic expansions, *Math. Comp.* 21(1967), 162–178.

[31] A.C. Matos and M. Prévost, Acceleratoon property for the columns of the E-algorithm, *Numer. Algorithms* 2(1992), 393–408.

[32] K. Murota and M. Sugihara, A remark on Aitken's δ^2-process, *Trans. Inform. Process. Soc. Japan* 25(1984), 892–894. (in Japanese) (= 室田一雄, 杉原正顕, Aitken 加速に関する一つの注意, 情報処理学会論文誌).

[33] I. Navot, An extension of the Euler-Maclaurin summation formula to functions with a branch singularity, *J. Math. and Phys.* 40(1961), 271–276.

[34] F.W.J. Olver, *Asymptotics and Special functions*, (Academic Press, New York, 1974).

[35] N. Osada, Asymptotic expansions and acceleration methods for alternating series (in Japanese), *Trans. Inform. Process. Soc. Japan* 28(1987), 431–436. (= 長田直樹, 交代級数の漸近展開と加速法, 情報処理学会論文誌).

[36] N. Osada, Asymptotic expansions and acceleration methods for logarithmically convergent series (in Japanese), *Trans. Inform. Process. Soc. Japan*, 29(1988), 256–261. (= 長田直樹, 対数収束級数の漸近展開と加速法, 情報処理学会論文誌).

[37] N. Osada, A convergence acceleration method for some logarithmically convergent sequences, *SIAM J. Numer. Anal.* 27(1990), 178–189.

[38] N. Osada, Accelerable subsets of logarithmic sequences, *J. Comput. Appl. Math.* 32(1990), 217–227.

[39] N. Osada, Acceleration methods for vector sequences, *J. Comput. Appl. Math.* 38 (1991), 361–371.

[40] N. Osada, A method for obtaining sequence transformations, *IMA J. Numer. Anal.* 12(1992), 85–94.

[41] N. Osada, Extension of Levin's transformations to vector sequences, *Numer. Algorithms* 2(1992), 121–132.

[42] K.J. Overholt, Extended Aitken acceleration, *BIT* 5(1965), 122–132.

[43] P. Rabinowitz, Extrapolation methods in numerical integration, *Numer. Algorithms* 3(1992), 17–28.

[44] L.F. Richardson, The deferred approach to the limit, Part I Single lattce, *Philos. Trans. Roy. Soc. London* Ser A 226(1927), 299–349.

[45] W. Romberg, Vereinfachte numerische Integration, *Kgl. Norske Vid. Selsk. Forhandlinger* 32(1955), 30–36.

[46] H. Rutishauser, Ausdehnung des Rombergschen Prinzips, *Numer. Math.* 5(1963), 48–54.

[47] P. Sablonnière, Asymptotic behaviour of iterated modified Δ^2 and θ_2 transforms on some slowly convergent sequences, *Numer. Algorithms* 3(1992), 401–410.

[48] J.W. Schmidt, Asymptotische Einschließung bei konvergenzbeschleunigenden Verfahren, *Numer. Math.* 8(1966), 105–113.

[49] R.J. Schmidt, On the numerical solution of linear simultaneous equations by an iterative method, *Phil. Mag.* 32(1941), 369–383.

[50] C. Schneider, Vereinfachte Rekursionen zur Richardson-Extrapolation in Spezialfällen, *Numer. Math.* 24(1975), 177–184.

[51] D. Shanks, Non-linear transformations of divergent and slowly convergent sequences, *J. Math. Phys.* 34(1955), 1–42.

[52] A. Sidi, Analysis of convergence of the T-transformation for power series, *Math. Comp.* 35(1980), 833–850.

[53] A. Sidi, On a generalization of the Richardson extrapolation process *Numer. Math.* 57(1990), 365–377.

[54] D.A. Smith and W.F. Ford, Acceleration of linear and logarithmic convergence, *SIAM J. Numer. Anal.* 16(1979), 223–240.

[55] R.R. Tucker, The δ^2-process and related topics II, *Pacific J. Math.* 28(1969), 455–463.

[56] E.J. Weniger, Nonlinear sequence transformations for the acceleration of convergence and the summation of divergent series, *Comput. Phys. Rep.* 10(1989), 189–371.

[57] E.J. Weniger, On the derivation of iterated sequence transformations for the acceleration of convergence and the summation of divergent series, *Comput. Phys. Comm.* 64(1991), 19–45.

[58] J. Wimp, *Sequence transformations and their applications*, (Academic Press, New York, 1981).

[59] P. Wynn, On a device for computing the $e_m(S_n)$ transformaion, *MTAC* 10(1956), 91–96.

[60] P. Wynn, On a procrustean technique for the numerical transformaion of slowly convergent sequences and series, *Proc. Camb. Phil. Soc.* 52(1956), 663–671.

[61] P. Wynn, On the convergence and stability of the epsilon algorithm, *SIAM J. Numer. Anal.* 3(1966), 91–122.

Appendix A. Asymptotic formulae of the Aitken δ^2 process

Lemma 10.3 *Suppose that a sequence* (s_n) *satisfies*

$$s_n = s + c_0 n^{\theta} + c_1 n^{\theta-1} + c_2 n^{\theta-2} + O(n^{\theta-3}), \tag{A.1}$$

where $\theta < 0$ *and* $c_0(\neq 0), c_1, c_2$ *are constants. Then the following asymptotic formulæ hold.*

(1) $s_n - \dfrac{(\Delta s_n)^2}{\Delta^2 s_n} = s + \dfrac{c_0}{1-\theta} n^{\theta} + O(n^{\theta-1})$.

(2) $s_n - \dfrac{\theta-1}{\theta}\dfrac{(\Delta s_n)^2}{\Delta^2 s_n} = s + O(n^{\theta-1})$.

(3) $s_n - \dfrac{\theta-1}{\theta}\dfrac{\Delta s_n \nabla s_n}{\Delta s_n - \nabla s_n} = s + O(n^{\theta-2})$.

Proof. (1) Using (A.1) and the binomial expansion, we have

$$\begin{aligned}
\Delta s_n &= c_0 n^{\theta}\left[\left(1+\frac{1}{n}\right)^{\theta} - 1\right] + c_1 n^{\theta-1}\left[\left(1+\frac{1}{n}\right)^{\theta-1} - 1\right] \\
&\qquad + c_2 n^{\theta-2}\left[\left(1+\frac{1}{n}\right)^{\theta-2} - 1\right] + O(n^{\theta-4}) \\
&= c_0\theta n^{\theta-1} + \left(\frac{1}{2}c_0\theta + c_1\right)(\theta-1)n^{\theta-2} \\
&\quad + \left[\frac{1}{6}c_0\theta(\theta-1) + \frac{1}{2}c_1(\theta-1) + c_2\right](\theta-2)n^{\theta-3} + O(n^{\theta-4})
\end{aligned} \tag{A.2}$$

and

$$\begin{aligned}
\Delta^2 s_n &= c_0\theta n^{\theta-1}\left[\left(1+\frac{1}{n}\right)^{\theta-1} - 1\right] \\
&\quad + \left(\frac{1}{2}c_0\theta + c_1\right)(\theta-1)n^{\theta-2}\left[\left(1+\frac{1}{n}\right)^{\theta-2} - 1\right] + O(n^{\theta-4}) \\
&= c_0\theta(\theta-1)n^{\theta-2}\left[1 + \left(1 + \frac{c_1}{c_0\theta}\right)(\theta-2)\frac{1}{n} + O(\frac{1}{n^2})\right].
\end{aligned}$$

By (A.2),

$$(\Delta s_n)^2 = c_0^2\theta^2 n^{2\theta-2}\left[1 + \left(1 + \frac{2c_1}{c_0\theta}\right)(\theta-1)\frac{1}{n} + O(\frac{1}{n^2})\right],$$

then we have

$$
\begin{aligned}
\frac{(\Delta s_n)^2}{\Delta^2 s_n} &= \frac{c_0\theta}{\theta-1} n^\theta \left[1 + \left(1+\frac{c_1}{c_0\theta}\right)(\theta-2)\frac{1}{n} + O(\frac{1}{n^2})\right] \\
&\quad \times \left[1 - \left(1+\frac{2c_1}{c_0\theta}\right)(\theta-1)\frac{1}{n} + O(\frac{1}{n^2})\right] \\
&= \frac{c_0\theta}{\theta-1} n^\theta \left[1 + \left(1+\frac{c_1}{c_0}\right)\frac{1}{n} + O(\frac{1}{n^2})\right]. \qquad \text{(A.3)}
\end{aligned}
$$

Thus we obtain

$$
s_n - \frac{(\Delta s_n)^2}{\Delta^2 s_n} = s + \frac{c_0}{1-\theta} n^\theta + O(n^{\theta-1}).
$$

(2) By (A.3), we have

$$
s_n - \frac{\theta-1}{\theta}\frac{(\Delta s_n)^2}{\Delta^2 s_n} = s - c_0 n^{\theta-1} + O(n^{\theta-2}).
$$

(3) Similarly,

$$
\begin{aligned}
\nabla s_n &= c_0\theta n^{\theta-1} + \left(-\frac{1}{2}c_0\theta + c_1\right)(\theta-1)n^{\theta-2} \\
&\quad + \left[\frac{1}{6}c_0\theta(\theta-1) - \frac{1}{2}c_1(\theta-1) + c_2\right](\theta-2)n^{\theta-3} + O(n^{\theta-4}). \qquad \text{(A.4)}
\end{aligned}
$$

By (A.2) and (A.4) we have

$$
\Delta s_n \nabla s_n = c_0^2\theta^2 n^{2\theta-2}\left[1 + \frac{2c_1}{c_0\theta}(\theta-1)\frac{1}{n} + O(\frac{1}{n^2})\right],
$$

and

$$
\Delta s_n - \nabla s_n = c_0\theta(\theta-1)n^{\theta-2}\left[1 + \frac{c_1(\theta-2)}{c_0\theta}\frac{1}{n} + O(\frac{1}{n^2})\right].
$$

Thus

$$
\frac{\Delta s_n \nabla s_n}{\Delta s_n - \nabla_n} = \frac{c_0\theta}{\theta-1}n^\theta\left[1 + \frac{c_1}{c_0}\frac{1}{n} + O(\frac{1}{n^2})\right].
$$

Therefore we obtain

$$
s_n - \frac{\theta-1}{\theta}\frac{\Delta s_n \nabla s_n}{\Delta s_n - \nabla_n} = s + O(n^{\theta-2}),
$$

as desired. □

Appendix B. An asymptotic formula of Lubkin's W transformation

For a sequence (s_n), the δ_α transform is defined by

$$\delta_\alpha(s_n) = s_{n+1} - \frac{\alpha+1}{\alpha}\frac{\Delta s_n \Delta s_{n+1}}{\Delta^2 s_n},$$

where α is a positive parameter.

Theorem 11.6 (Sablonnière) *Suppose that a sequence* (s_n) *satisfies*

$$s_n \sim s + n^\theta \sum_{j=0}^{\infty} \frac{c_j}{n^j}. \tag{B.1}$$

If $W_{n,k}$ *is represented as*

$$W_{n,k} - s = n^{\theta-2k}\left[c_0^{(k)} + \frac{c_1^{(k)}}{n} + \frac{c_2^{(k)}}{n^2} + O(\frac{1}{n^3})\right], \quad c_0^{(k)} \neq 0, \tag{B.2}$$

then

$$W_{n,k+1} - s = n^{\theta-2k-2}\left[c_0^{(k+1)} + O(\frac{1}{n})\right],$$

where

$$\begin{aligned} c_0^{(k+1)} &= \frac{c_0^{(k)}(1+\theta-2k)}{6(\theta-2k)} - \frac{2(c_0^{(k)}k(\theta-2k) - c_1^{(k)})^2}{c_0^{(k)}(\theta-2k)^3} \\ &+ \frac{c_0^{(k)}k^2(\theta-2k)(\theta-2k-1) - 4c_1^{(k)}k(\theta-2k-1) + 4c_2^{(k)}}{(\theta-2k)^2(\theta-2k-1)} \end{aligned} \tag{B.3}$$

Proof. Let $\tilde{c}_0^{(k)}, \tilde{c}_1^{(k)}$ and $\tilde{c}_2^{(k)}$ be defined by

$$W_{n,k} - s = (n+k)^{\theta-2k}\left[\tilde{c}_0^{(k)} + \frac{\tilde{c}_1^{(k)}}{n+k} + \frac{\tilde{c}_2^{(k)}}{(n+k)^2} + O(\frac{1}{(n+k)^3})\right].$$

Then

$$\begin{aligned} W_{n,k} - s &= \tilde{c}_0^{(k)} n^{\theta-2k} + \left(\tilde{c}_0^{(k)} k(\theta-2k) + \tilde{c}_1^{(k)}\right) n^{\theta-2k-1} \\ &+ \left(\frac{1}{2}\tilde{c}_0^{(k)} k^2(\theta-2k)(\theta-2k-1) + \tilde{c}_1^{(k)} k(\theta-2k-1) + \tilde{c}_2^{(k)}\right) n^{\theta-2k-2} \\ &+ O(n^{\theta-2k-3}) \end{aligned} \tag{B.4}$$

By (B.2) and (B.4), we have

$$\begin{aligned}
\tilde{c}_0^{(k)} &= c_0^{(k)}, \\
\tilde{c}_1^{(k)} &= -c_0^{(k)} k(\theta - 2k) + c_1^{(k)}, \\
\tilde{c}_2^{(k)} &= \frac{1}{2} c_0^{(k)} k^2 (\theta - 2k)(\theta - 2k - 1) - c_1^{(k)} k(\theta - 2k - 1) + c_2^{(k)}.
\end{aligned}$$

By Theorem 10.4,

$$\delta_{2k-\theta}(W_{n,k}) - s = d_0^{(k+1)} (n + k + 1)^{\theta - 2k - 2} + O((n + k + 1)^{\theta - 2k - 3}).$$

where

$$\begin{aligned}
d_0^{(k+1)} &= \frac{\tilde{c}_0^{(k)}(1 + \theta - 2k)}{12} - \frac{(\tilde{c}_1^{(k)})^2}{\tilde{c}_0^{(k)}(\theta - 2k)^2} + \frac{2\tilde{c}_2^{(k)}}{(\theta - 2k)(\theta - 2k - 1)}, \\
&= \frac{c_0^{(k)}(1 + \theta - 2k)}{12} - \frac{(c_1^{(k)} - c_0^{(k)} k(\theta - 2k))^2}{c_0^{(k)}(\theta - 2k)^2} \\
&\quad + \frac{c_0^{(k)} k^2 (\theta - 2k)(\theta - 2k - 1) - 2c_1^{(k)} k(\theta - 2k - 1) + 2c_2^{(k)}}{(\theta - 2k)(\theta - 2k - 1)}. \qquad \text{(B.5)}
\end{aligned}$$

By (B.2),

$$\begin{aligned}
&\Delta W_{n+1,k} \\
&= c_0^{(k)} (\theta - 2k) n^{\theta - 2k - 1} \left[1 + \left(\frac{3}{2}(\theta - 2k - 1) + \frac{c_1^{(k)}(\theta - 2k - 1)}{c_0^{(k)}(\theta - 2k)} \right) \frac{1}{n} \right. \\
&\quad + \left(\frac{7}{6}(\theta - 2k - 1) + \frac{3c_1^{(k)}(\theta - 2k - 1)}{2c_0^{(k)}(\theta - 2k)} + \frac{c_2^{(k)}}{c_0^{(k)}(\theta - 2k)} \right) \frac{\theta - 2k - 2}{n^2} \\
&\quad \left. + O\left(\frac{1}{n^3}\right) \right], \qquad \text{(B.6)}
\end{aligned}$$

and

$$\begin{aligned}
&W_{n+1,k} - \delta_{2k-\theta}(W_{n,k}) \\
&= c_0^{(k)} n^{\theta - 2k} \left[1 + \left(\theta - 2k + \frac{c_1^{(k)}}{c_0^{(k)}} \right) \frac{1}{n} \right. \\
&\quad + \left(\frac{1}{2}(\theta - 2k)(\theta - 2k - 1) + \frac{c_1^{(k)}(\theta - 2k - 1)}{c_0^{(k)}} + \frac{c_2^{(k)}}{c_0^{(k)}} - \frac{d_0^{(k+1)}}{c_0^{(k)}} \right) \frac{1}{n^2} \\
&\quad \left. + O\left(\frac{1}{n^3}\right) \right]. \qquad \text{(B.7)}
\end{aligned}$$

We put

$$W_{n,k+1} - s = W_{n+1,k} - N/D.$$

Then

$$\begin{aligned}N &= \Delta W_{n+1,k}\left(W_{n+1,k} - \delta_{2k-\theta}(W_{n,k})\right)\\ &= (c_0^{(k)})^2(\theta-2k)n^{2\theta-4k-1}\Bigg[1 + \left(\frac{5\theta}{2} - 5k - \frac{3}{2} + \frac{c_1^{(k)}(2\theta-4k-1)}{c_0^{(k)}(\theta-2k)}\right)\frac{1}{n}\\ &\quad + \Bigg((\theta-2k-1)(\frac{19\theta}{6} - \frac{19k}{3} - \frac{7}{3}) + \frac{c_1^{(k)}(\theta-2k-1)(5\theta-10k-3)}{c_0^{(k)}(\theta-2k)}\\ &\quad + \frac{2c_2^{(k)}(\theta-2k-1)}{c_0^{(k)}(\theta-2k)} + \frac{(c_1^{(k)})^2(\theta-2k-1)}{(c_0^{(k)})^2(\theta-2k)} - \frac{d_0^{(k+1)}}{c_0^{(k)}}\Bigg)\frac{1}{n^2} + O(\frac{1}{n^3})\Bigg].\end{aligned} \tag{B.8}$$

Similarly, we have

$$\begin{aligned}D &= \Delta W_{n+1,k} - \Delta\delta_{2k-\theta}(W_{n,k})\\ &= c_0^{(k)}(\theta-2k)n^{\theta-2k-1}\Bigg[1 + \left(\frac{3}{2}(\theta-2k-1) + \frac{c_1^{(k)}(\theta-2k-1)}{c_0^{(k)}(\theta-2k)}\right)\frac{1}{n}\\ &\quad + \left(\frac{7}{6}(\theta-2k-1) + \frac{3c_1^{(k)}(\theta-2k-1)}{2c_0^{(k)}(\theta-2k)} + \frac{c_2^{(k)}}{c_0^{(k)}(\theta-2k)} - \frac{d_0^{(k+1)}}{c_0^{(k)}(\theta-2k)}\right)\frac{\theta-2k-2}{n^2}\\ &\quad + O(\frac{1}{n^3})\Bigg].\end{aligned} \tag{B.9}$$

By (B.8) and (B.9),

$$\begin{aligned}\frac{N}{D} &= c_0^{(k)}n^{\theta-2k}\Bigg[1 + \left(\theta-2k + \frac{c_1^{(k)}}{c_0^{(k)}}\right)\frac{1}{n}\\ &\quad + \left(\frac{1}{2}(\theta-2k)(\theta-2k-1) + \frac{c_1^{(k)}(\theta-2k-1)}{c_0^{(k)}} + \frac{c_2^{(k)}}{c_0^{(k)}} - \frac{2d_0^{(k+1)}}{c_0^{(k)}(\theta-2k)}\right)\frac{1}{n^2}\\ &\quad + O(\frac{1}{n^3})\Bigg].\end{aligned} \tag{B.10}$$

Since $W_{n,k+1} - s = W_{n+1,k} - N/D$, we obtain

$$W_{n,k+1} - s = \frac{2d_0^{(k+1)}}{\theta-2k}n^{\theta-2k-2} + O(n^{\theta-2k-3}).$$

This completes the proof. □

FORTRAN PROGRAM

Here we give a FORTRAN program that includes the subroutines GENRHO, the generalized ρ-algorithm, and MODAIT, the modified Aitken δ^2 formula. The main routine given below is an example of applications of the subroutines to the series

$$s_n = \sum_{i=1}^{n} \frac{1}{i\sqrt{i}}.$$

The parameters `NMAX, EPSTOR, DMINTOR, XTV` in GENRHO and MODAIT are as follows:

`NMAX`	The maximum number of iterations.
`EPSTOR`	The absolute error tolerance.
`DMINTOR`	A positive number. For a variable x, if $\lvert x\rvert <$`DMINTOR` then the program stop.
`XTV`	The true value, i.e. the limit of the sequence.

The variables `N, ILL, TH` in GENRHO and MODAIT are as follows:

`N`	A positive integer n such that s_n is the n-th term.
`ILL`	A non-negative integer. If `ILL`> 0, then the program stop.
`TH`	A real number. The exponent θ in the asymptotic expansion.

The variables `XX, RHO, KOPT` in GENRHO are as follows:

`XX`	The n-th term s_n. (input) $\rho_{2k}^{(n-2k)}$, where $k = \lfloor n/2 \rfloor$. (output)
`RHO`	An array of dimension (0:1,0:`NMAX`) `RHO`(1,k):$\rho_k^{(n-1-k)}$ (input) `RHO`(1,k):$\rho_k^{(n-k)}$ (output)
`KOPT`	A non-negative integer such that $\rho_{2k}^{(n-2k)}$ is the accelerated value where $k =$`KOPT`.

The variables `XX, S, KOPT, DS` in MODAIT are as follows:

`XX`	The n-th term s_n. (input) $s_k^{(n-k)}$, where $k = \lfloor n/2 \rfloor$. (output)

`S` An array of dimension (0:1,0:`NMAX`)
`S`(1,k):$s_k^{(n-1-k)}$ (input)
`S`(1,k):$s_k^{(n-k)}$ (output)

`KOPT` A non-negative integer such that $s_k^{(n-k)}$ is the accelerated value where k =`KOPT`.

`DS` An array of dimension (0:1,0:`NMAX`)
`DS`(1,k):$s_k^{(n-k-1)} - s_k^{(n-k-2)}$ (input)
`DS`(1,k):$s_k^{(n-k)} - s_k^{(n-k-1)}$ (output)

The function TERM(N) returns the n-th term of infinite series.

```
C     ACCELERATION METHODS FOR INFINITE SERIES
      PROGRAM INFSER
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*4 (I-N)
      CHARACTER CEQ*60,CACCL*60
      PARAMETER (NMAX=20,EPSTOR=1.0D-12,DMINTOR=1.0D-30)
      EXTERNAL TERM
      REAL*8 X(1:NMAX)
      REAL*8 RHO(0:1,0:NMAX)
      REAL*8 TRHO(0:1,0:NMAX)
      REAL*8 S(0:1,0:NMAX),DS(0:1,0:NMAX)
      REAL*8 TS(0:1,0:NMAX),DTS(0:1,0:NMAX)
      CEQ=' A_I=1/SQRT(I)/I  '
      XTV=2.61237534868549D0
      DO 101 IACCL=1,2
          GO TO (102,103),IACCL
 102      CACCL=' AUTOMATIC GENERALIZED RHO ALGORITHM  '
          GO TO 109
 103      CACCL=' AUTOMATIC MODIFIED AITKEN DELTA SQUARE '
          GO TO 109
 109      CONTINUE
          WRITE (*,3000)
          WRITE (*,*) CEQ
          WRITE (*,*) CACCL
          WRITE (*,3100)
          ILL=0
          XX=0.0D0
          DO 201 N=1,NMAX
            X0=XX
            XX=XX+TERM(N)
            X(N)=XX
            IF (N.EQ.1) THEN
              DX=XX
              GO TO 209
            ENDIF
            IF (N.EQ.2) THEN
              DX0=DX
              DX=XX-X0
```

```
            D2X=DX-DX0
            DD=DX/D2X
            GO TO 209
          ENDIF
          DX0=DX
          DX=XX-X0
          D2X=DX-DX0
          DD0=DD
          DD=DX/D2X
          ALPHA=1.0D0/(DD-DD0)+1.0D0
          TH=-2.0D0
          NN=N-2
          GO TO (202,203),IACCL
202       CALL GENRHO(ALPHA,TRHO,NN,DMINTOR,KOPT,NMAX,ILL,TH)
          GO TO 209
203       CALL MODAIT(ALPHA,TS,DTS,NN,DMINTOR,KOPT,NMAX,ILL,TH)
          GO TO 209
209       CONTINUE
          ERX=ABS(XX-XTV)
          SDXER=-LOG10(ERX)
          XP=XX
          IF (N.LE.2) GO TO 229
          GO TO (210,220),IACCL
210       CONTINUE
          DO 211 NN=1,N
            XP=X(NN)
            TH=ALPHA
            CALL GENRHO(XP,RHO,NN,DMINTOR,KOPT,NMAX,ILL,TH)
211       CONTINUE
          GO TO 229
220       CONTINUE
          DO 221 NN=1,N
            XP=X(NN)
            TH=ALPHA
            CALL MODAIT(XP,S,DS,NN,DMINTOR,KOPT,NMAX,ILL,TH)
221       CONTINUE
          GO TO 229
229       CONTINUE
          ER=ABS(XP-XTV)
          SDER=-LOG10(ER)
          IF (N.LE.2) GO TO 232
231       WRITE (*,2000) N,XX,ALPHA,XP,SDXER,SDER,KOPT
          GO TO 239
232       WRITE (*,2010) N,XX
          GO TO 239
239       CONTINUE
          DXP=ABS(XP-XP0)
          XP0=XP
          IF (DXP.LT.EPSTOR) GO TO 300
          IF (ILL.GE.1) GO TO 300
          IF (N.LE.2) GO TO 201
201     CONTINUE
300     CONTINUE
```

```
        WRITE (*,3100)
        IF (ILL.GE.1) THEN
            WRITE (*,*) '  ABNORMALLY ENDED  '
        ENDIF
        WRITE (*,3100)
101   CONTINUE
2000 FORMAT (I4,3D25.15,2F7.2,I5)
2010 FORMAT (I4,1D25.15)
3000 FORMAT (1H1)
3100 FORMAT (/)
9999 STOP
     END

     FUNCTION TERM(N)
     REAL*8 X,TERM
     X=DBLE(N)
     TERM=1.0D0/X/SQRT(X)
     RETURN
     END

     SUBROUTINE GENRHO(XX,RHO,N,DMINTOR,KOPT,NMAX,ILL,TH)
     REAL*8 DMINTOR,ER,TH
     REAL*8 RHO(0:1,0:NMAX)
     REAL*8 DRHO,XX
     KOPT=0
     IF (N.EQ.1) GO TO 110
     KEND=N-1
     DO 101 K=KEND,0,-1
       RHO(0,K)=RHO(1,K)
 101 CONTINUE
 110 CONTINUE
     RHO(1,0)=XX
     IF (N.EQ.1) THEN
       RHO(1,1)=-TH/XX
       GO TO 199
     ENDIF
     DRHO=RHO(1,0)-RHO(0,0)
     ER=ABS(DRHO)
     KOPT=0
     IF (ER.LT.DMINTOR) THEN
       ILL=1
       GO TO 199
     ENDIF
     RHO(1,1)=-TH/DRHO
     KEND=N
     DO 121 K=2,KEND
       DRHO=RHO(1,K-1)-RHO(0,K-1)
       ER=ABS(DRHO)
       IF ((ER.LT.DMINTOR).AND.(MOD(K,2).EQ.1)) THEN
         KOPT=K-1
         GO TO 140
       ENDIF
       IF ((ER.LT.DMINTOR).AND.(MOD(K,2).EQ.0)) THEN
```

```
          ILL=1
          GO TO 199
        ENDIF
        RHO(1,K)=RHO(0,K-2)+(DBLE(K-1)-TH)/DRHO
  121 CONTINUE
      IF (MOD(N,2).EQ.0) THEN
          KOPT=N
        ELSE
          KOPT=N-1
      ENDIF
  140 CONTINUE
      XX=RHO(1,KOPT)
  199 RETURN
      END

      SUBROUTINE MODAIT(XX,S,DS,N,DMINTOR,KOPT,NMAX,ILL,TH)
      REAL*8 DMINTOR,TH,COEF
      REAL*8 W1,W2,XX
      REAL*8 S(0:1,0:NMAX),DS(0:1,0:NMAX)
      KOPT=0
      IF (N.EQ.1) GO TO 110
      KEND=INT((N-1)/2)
      DO 101 K=0,KEND
        S(0,K)=S(1,K)
        IF ((MOD(N,2).EQ.1).AND.(K.EQ.KEND)) GO TO 101
        DS(0,K)=DS(1,K)
101   CONTINUE
110   CONTINUE
      S(1,0)=XX
      IF (N.EQ.1) THEN
        DS(1,0)=XX
        GO TO 199
      ENDIF
      DS(1,0)=XX-S(0,0)
      KEND=INT(N/2)
      DO 111 K=1,KEND
        W1=DS(0,K-1)*DS(1,K-1)
        W2=DS(1,K-1)-DS(0,K-1)
        IF (ABS(W2).LT.DMINTOR) THEN
          ILL=1
          GO TO 199
        ENDIF
        COEF=(DBLE(2*K-1)-TH)/(DBLE(2*K-2)-TH)
        S(1,K)=S(0,K-1)-COEF*W1/W2
        IF (N.EQ.2*K-1) GO TO 111
        DS(1,K)=S(1,K)-S(0,K)
 111  CONTINUE
 120  KOPT=INT(N/2)
 140  CONTINUE
      XX=S(1,KOPT)
 199  RETURN
      END
```