COMPUTATIONAL & APPLIED MATHEMATICS

Volume 26, N. 2, pp. 171–189, 2007
Copyright © 2007 SBMAC
ISSN 0101-8205
www.scielo.br/cam

Generalizations of Aitken's process for accelerating the convergence of sequences

CLAUDE BREZINSKI[1] and MICHELA REDIVO ZAGLIA[2]

[1]Laboratoire Paul Painlevé, UMR CNRS 8524, UFR de Mathématiques Pures et Appliquées,
Université des Sciences et Technologies de Lille, 59655 – Villeneuve d'Ascq cedex, France
[2]Università degli Studi di Padova, Dipartimento di Matematica Pura ed Applicata
Via Trieste 63, 35121 – Padova, Italy

E-mails: Claude.Brezinski@univ-lille1.fr / Michela.RedivoZaglia@unipd.it

Abstract. When a sequence or an iterative process is slowly converging, a convergence acceleration process has to be used. It consists in transforming the slowly converging sequence into a new one which, under some assumptions, converges faster to the same limit. In this paper, new scalar sequence transformations having a kernel (the set of sequences transformed into a constant sequence) generalizing the kernel of the Aitken's Δ^2 process are constructed. Then, these transformations are extended to vector sequences. They also lead to new fixed point methods which are studied.

Mathematical subject classification: Primary: 65B05, 65B99; Secondary: 65H10.

Key words: convergence acceleration, Aitken process, extrapolation, fixed point methods.

1 Introduction

When a sequence (S_n) of real or complex numbers is slowly converging, it can be transformed into a new sequence (T_n) by a *sequence transformation*. Aitken's Δ^2 process and Richardson extrapolation (which gives rise to Romberg's method for accelerating the convergence of the trapezoidal rule) are the most well known sequence transformations. It has been proved that a sequence transformation able to accelerate the convergence of all sequences cannot exist [8] (see also [7]).

#682/06. Received: 09/X/06. Accepted: 05/III/07.

In fact, each transformation is only able to accelerate the convergence of special classes of sequences. This is the reason why several sequence transformations have to be constructed and studied.

For constructing a new sequence transformation, an important object is its *kernel* (we will explain why below). It is the set of sequences (S_n), characterized by a particular expression or satisfying a particular relation between its terms, both involving an unknown parameter S (the limit of the sequence if it converges or its antilimit if it does not converge), that are transformed into the constant sequence $(T_n = S)$. For example, the kernel of the Aitken's Δ^2 process (see its definition below) is the set of sequences of the form $S_n = S + a\lambda^n$, $n = 0, 1, \ldots$, where $a \neq 0$ and $\lambda \neq 1$ or, equivalently, satisfying the relation $a_0(S_n - S) + a_1(S_{n+1} - S)$, for $n = 0, 1, \ldots$, with $a_0 a_1 \neq 0$ and $a_0 + a_1 \neq 0$. If $|\lambda| < 1$, then (S_n) converges to its limit S. Otherwise, S is called the antilimit of the sequence (S_n).

The construction of a sequence transformation having a specific kernel consists in giving the exact expression of the parameter S for any sequence belonging to this kernel. This expression makes use of several consecutive terms of the sequence starting from S_n, and it is valid for all n. Thus, by construction, for all n, $T_n = S$. When applied to a sequence not belonging to its kernel, the transformation produces a sequence (T_n) which, under some assumptions, converges to S faster than (S_n), that is

$$\lim_{n\to\infty} \frac{T_n - S}{S_n - S} = 0.$$

In that case, it is said that the transformation accelerates the convergence of (S_n).

In fact, a sequence transformation is based on interpolation followed by extrapolation. For example, the parameters a, λ and S appearing in the kernel of the Aitken's process are computed by solving the system $S_{n+i} = S + a\lambda^{n+i}$ for $i = 0, 1, 2$. If the sequence (S_n) to be transformed does not belong to the kernel, then the value of S (and also those of a and λ) obtained from this system depends of n and it is denoted by T_n. In order to fully understand the procedure followed for obtaining the transformations given in this paper, let us explain in details how the Aitken's Δ^2 process is derived from its kernel. The sequences of its kernel satisfy $(S_n - S)/\lambda^n = a$. Applying the usual forward difference operator Δ (it is an annihilation operator as will be explained below) to both sides leads to $\Delta((S_n - S)/\lambda^n) = \Delta a = 0$. Thus $S_{n+1} - S = \lambda(S_n - S)$ and it follows $S = (S_{n+1} - \lambda S_n)/(1-\lambda)$. The problem is now to compute λ. Applying the operator Δ to $S_n = S + a\lambda^n$ gives $\Delta S_n = a\lambda^n(\lambda - 1)$, and we obtain $\lambda = \Delta S_{n+1}/\Delta S_n$. Thus, replacing λ by this expression in the formula for S, leads to the transformation

$$T_n = \frac{S_n S_{n+2} - S_{n+1}^2}{S_{n+2} - 2S_{n+1} + S_n}, \quad n = 0, 1, \ldots \tag{1}$$

which, by construction, has a kernel including all sequences of the form $S_n = S + a\lambda^n$ or, equivalently, such that $S_{n+1} - S = \lambda(S_n - S)$ for all n. We see that the denominator in this formula is $\Delta^2 S_n$, thus the name of the transformation.

An important point to notice for numerical applications is that Formula (1) is numerically unstable. It has to be put under one of the equivalent forms

$$\begin{aligned} T_n &= S_n - \frac{(\Delta S_n)^2}{S_{n+2} - 2S_{n+1} + S_n} \\ &= S_{n+1} - \frac{\Delta S_n \Delta S_{n+1}}{S_{n+2} - 2S_{n+1} + S_n} \\ &= S_{n+2} - \frac{(\Delta S_{n+1})^2}{S_{n+2} - 2S_{n+1} + S_n} \end{aligned}$$

which are more stable. Indeed, when the terms S_n, S_{n+1} and S_{n+2} are close to S, a cancellation appears in Formula (1). Its numerator and its denominator are close to zero, thus producing a first order cancellation. A cancellation also appears in the three preceding formulae, but it is a cancellation on a correcting term, that is, in some sense, a second order cancellation (see [5, pp. 400–403] for an extensive the discussion).

Similarly, in the sequel, when a transformation is written as $T_n = N_n/D_n$, it is usually unstable. If the computation of T_n makes use of $S_n, \ldots, S_{n+k}$, then T_n can also be put under one of the forms $T_n = S_{n+i} - (S_{n+i}D_n - N_n)/D_n$, for any $i = 0, \ldots, k$, which, after simplification in the numerator $S_{n+i}D_n - N_n$, is more stable.

By construction, we saw that, for all n, $T_n = S$, if (S_n) belongs to the kernel of the transformation. To prove that this condition is also necessary is more difficult. One has to start from the condition for all n, $T_n = S$, and then to show that it implies that the relation defining the kernel is satisfied. This is why, in this paper, we will only say that the kernels of the transformations studied include all sequences having the corresponding form since additional sequences can also belong to the kernel. Let us mention that, for the Aitken's process, the condition is necessary and sufficient.

Of course, one can ask why the notion of kernel is an important one. Although this result was never proved, it is hoped (and it was experimentally verified) that if a sequence is not "too far away" from the kernel of a certain transformation, then this transformation will accelerate its convergence. For example, the kernel of the Aitken's process can also be described as the set of sequences such that for all n, $(S_{n+1} - S)/(S_n - S) = \lambda \neq 1$. It is easy to prove that this process accelerates the convergence of all sequences for which there exists $\lambda \neq 1$ such that $\lim_{n\to\infty}(S_{n+1} - S)/(S_n - S) = \lambda$. On sequence transformations, their kernels, and extrapolation methods see, for example, [5, 14, 16].

The Aitken's Δ^2 process is one of the most popular and effective convergence acceleration method. In this paper we will construct scalar sequence transformations whose kernels generalize the kernel of this transformation which consists, as we saw above, of sequences such that $S_n = S + a\lambda^n$ for $n = 0, 1, \ldots$. Defining and studying generalizations of Aitken's process leads to interesting applications as those described in [11] and [12]. In this paper, we will consider two new generalizations of the kernel of Aitken's process, namely consisting of sequences of the form $S_n = S + (a + bx_n)\lambda^n$ (Section 2) and $S_n = S + \lambda^n/(a + bx_n)$ (Section 3), where (x_n) is a given known sequence. Compared to the sequences in the kernel of Aitken's process, the additional term bx_n can completely change the behaviour since non monotonic sequences or non strictly alternating ones are now included into the kernel. According to the value of λ and to the choice of (x_n), we can have sequences whose error (in absolute value) first increases, and then tends to zero, or divergent sequences whose error (in absolute value) begins to approach zero, and then goes to infinity, thus imitating asymptotic series. This extra term can also be considered as a subdominant contribution (that is a kind of second order term) to the sequence as pointed out by Weniger [15]. Let us mention that, as showed in particular in [13], there is a strong connection between asymptotics and extrapolation methods.

In Section 5, these transformations will be extended to vector sequences. The related fixed point methods will be studied in Section 6.

The following definitions are needed in the sequel. The forward difference operator Δ is defined by

$$\begin{aligned} \Delta u_n &= u_{n+1} - u_n, \\ \Delta^{k+1} u_n &= \Delta^k u_{n+1} - \Delta^k u_n, \end{aligned}$$

the divided difference operator δ is defined by

$$\begin{aligned} \delta u_n &= \frac{u_{n+1} - u_n}{x_{n+1} - x_n}, \\ \delta^{k+1} u_n &= \frac{\delta^k u_{n+1} - \delta^k u_n}{x_{n+k+1} - x_n} \end{aligned}$$

and the reciprocal difference operators by

$$\varrho^{k+1} u_n = \varrho^{k-1} u_{n+1} + \frac{x_{n+k+1} - x_n}{\varrho^k u_{n+1} - \varrho^k u_n}$$

with $\varrho^{-1} u_n = 0$ and $\varrho^0 u_n = u_n$.

We also remind the Leibniz's rule for the operator Δ

$$\Delta(u_n v_n) = u_{n+1} \Delta v_n + v_n \Delta u_n.$$

2 A first scalar kernel

We will construct a sequence transformation with a kernel containing all sequences of the form

$$S_n = S + (a + b x_n)\lambda^n, \quad n = 0, 1, \ldots \tag{2}$$

where S, a, b and λ are unknown (possibly complex) numbers and (x_n) a known (possibly complex) sequence.

We have, for all n,

$$\delta(a + b x_n) = \delta\left(\frac{S_n - S}{\lambda^n}\right) = b. \tag{3}$$

2.1 First technique

From (3), we obtain

$$b\lambda^{n+1}\Delta x_n = S_{n+1} - S - \lambda(S_n - S). \tag{4}$$

Extracting S from this relation leads to a first transformation whose kernel includes all sequences of the form (2)

$$T_n = \frac{S_{n+1} - \lambda S_n - b\lambda^{n+1}\Delta x_n}{1-\lambda}, \quad n = 0, 1, \ldots \tag{5}$$

The problem is now to compute the unknowns b and λ (or λ and $b\lambda^{n+1}$) appearing in (5). Applying the forward difference operator Δ to (4), we get

$$b\lambda^{n+1}(\lambda\Delta x_{n+1} - \Delta x_n) = \Delta S_{n+1} - \lambda\Delta S_n, \tag{6}$$

which gives b (or $b\lambda^{n+1}$) if λ is known.

Writing down (6) also for the index $n+1$, we obtain a system of two nonlinear equations in our unknowns. The unknown product $b\lambda^{n+1}$ can be eliminated by division and we get, after rearrangement of the terms,

$$\frac{\Delta S_{n+2} - \lambda\Delta S_{n+1}}{\lambda\Delta x_{n+2} - \Delta x_{n+1}} = \lambda\frac{\Delta S_{n+1} - \lambda\Delta S_n}{\lambda\Delta x_{n+1} - \Delta x_n}. \tag{7}$$

This is a cubic equation which provides, in the real case, a unique λ only if it has one single real zero. So, another procedure for the computation of λ has to be given.

It is possible to compute λ by writing this cubic equation for the indexes n, $n+1$ and $n+2$. Thus we obtain a system of 3 linear equations in the 3 unknowns λ, λ^2, and λ^3

$$\alpha_{n+i}\lambda^3 + \beta_{n+i}\lambda^2 + \gamma_{n+i}\lambda = \delta_{n+i}, \quad i = 0, 1, 2$$

with

$$\begin{aligned}
\alpha_{n+i} &= \Delta x_{n+2+i}\Delta S_{n+i} \\
\beta_{n+i} &= -(\Delta x_{n+1+i}\Delta S_{n+1+i} + \Delta x_{n+2+i}\Delta S_{n+1+i} + \Delta x_{n+1+i}\Delta S_{n+i}) \\
\gamma_{n+i} &= \Delta x_{n+1+i}\Delta S_{n+2+i} + \Delta x_{n+i}\Delta S_{n+1+i} + \Delta x_{n+1+i}\Delta S_{n+1+i} \\
\delta_{n+i} &= \Delta x_{n+i}\Delta S_{n+i+2}.
\end{aligned}$$

We solve this system for the unknown λ, then we compute λ^{n+1}, and we finally obtain b by (6).

Another way to proceed is to replace $b\lambda^{n+1}$ by its expression in (5). Then the transformation can also be written as

$$\begin{aligned} T_n &= \frac{\lambda r_n(S_{n+1}-\lambda S_n)-(S_{n+2}-\lambda S_{n+1})}{(\lambda r_n-1)\,(1-\lambda)}, && (8)\\ &= S_{n+1}-\frac{\Delta S_{n+1}-\lambda^2 r_n\Delta S_n}{(\lambda r_n-1)(1-\lambda)}, && (9)\end{aligned}$$

with $r_n=\Delta x_{n+1}/\Delta x_n$, and where λ is computed by solving the preceding linear system. Formula (9) is more stable than Formula (8). In (9), it is also possible to replace λ^2 by its value given as the solution of the preceding linear system instead of squaring λ, thus leading to a different transformation with also a kernel containing all sequences of the form (2).

Remark 1. Obviously, for sequences which do not have the form (2), the value of λ obtained by the previous procedures depends on n.

If Δx_n is constant, $\lambda=1$ satisfies the preceding system but its matrix is singular. In this case, (7) reduces to

$$\Delta S_{n+2}-\lambda\Delta S_{n+1}=\lambda(\Delta S_{n+1}-\lambda\Delta S_n)$$

and λ can be computed by solving the system

$$\Delta S_{n+i}\lambda^2-2\Delta S_{n+1+i}\lambda=-\Delta S_{n+i+2},\quad i=0,1.$$

Then, the transformation given by (8) or (9) becomes

$$\begin{aligned} T_n &= \frac{(S_{n+2}-\lambda S_{n+1})-\lambda(S_{n+1}-\lambda S_n)}{(\lambda-1)^2}, && (10)\\ &= S_{n+1}+\frac{\Delta S_{n+1}-\lambda^2\Delta S_n}{(\lambda-1)^2}. && (11)\end{aligned}$$

Formula (11) is more stable than (10).

Let us give a numerical example to illustrate this transformation. We consider the sequence

$$S_n=S+\lambda^n(2-n^\alpha)+\exp(-n)(1+n^b),\quad n=0,1,\ldots \qquad (12)$$

with $x_n = n^\alpha$. We took $S = 1, \lambda = 1.15$, $\alpha = 3.5$, and $b = 2$. With these values, the first term in (S_n) diverges, while the second one tends to zero and, according to [15], is a subdominant contribution to (S_n). On Figure 1, the solid lines represent, in a logarithmic scale, and from top to bottom, the absolute errors of S_n, of the Aitken's Δ^2 process (which uses 3 consecutive terms of the sequence to accelerate), of its first iterate (which uses 5 terms), and of its second iterate (which uses 7 terms). The dash-dotted line corresponds to the error of (11) (which uses 5 terms) with x_n defined as above (which implies the knowledge of α), and the dashed line to the error of (9) (which uses 6 terms). Iterating a process, such as Aitken's, consists in reapplying it to the new sequence obtained by its previous application. On this example, the numerical results obtained by the Formula (8) and by the more stable Formula (9) are the same. The computations were performed using Matlab 7.3.

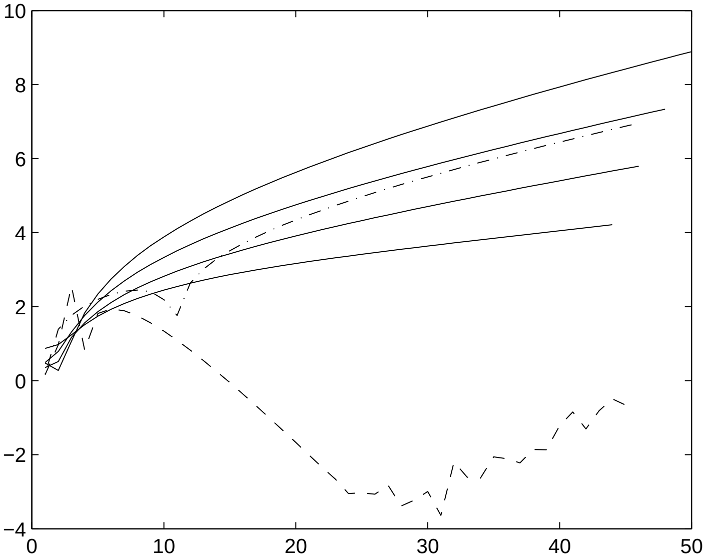

Figure 1 – Transformation (9) in dashed line, and transformation (11) in dash-dotted line, both applied to (12).

The results of Figure 1 show that Aitken's process and its iterates have no impact on the divergence of the sequence (S_n). On the contrary, the dominant contribution has been suppressed at the beginning by the transformations (9) and (11), and they generate sequences which converge before diverging again. Of course, when n grows, the subdominant contribution is almost zero, and this is why these sequence transformations no longer operate. This is particularly visible with transformation (11) which produces a rapidly diverging sequence. On the contrary, transformation (9) exhibits a behavior similar to the behavior of an asymptotic series. Thus stopping its application after 27 or 28 terms leads to an error of the order of 10^{-4}. It must be noticed that the numerical results are quite sensitive to changes in the parameters α, λ, and b if Formula (10) is used, while they are not with Formula (11).

2.2 Second technique

Since, for all n, $\delta(a + bx_n)$ is a constant, then for all n, $\Delta\delta(a + bx_n) = 0$. Thus

$$\begin{aligned} 0 &= \Delta\delta\left(\frac{S_n - S}{\lambda^n}\right) \\ &= \frac{1}{\lambda^{n+2}}\left[\frac{S_{n+2} - S - \lambda(S_{n+1} - S)}{\Delta x_{n+1}} - \lambda\frac{S_{n+1} - S - \lambda(S_n - S)}{\Delta x_n}\right] \end{aligned}$$

and it follows, for $n = 0, 1, \ldots$,

$$\Delta x_n[S_{n+2} - S - \lambda(S_{n+1} - S)] - \lambda\Delta x_{n+1}[S_{n+1} - S - \lambda(S_n - S)] = 0. \quad (13)$$

Extracting S from this relation, we obtain the following sequence transformation whose kernel includes all sequences of the form (2)

$$T_n = \frac{S_{n+2}\Delta x_n - \lambda S_{n+1}(\Delta x_n + \Delta x_{n+1}) + \lambda^2 S_n \Delta x_{n+1}}{\Delta x_n - \lambda(\Delta x_n + \Delta x_{n+1}) + \lambda^2\Delta x_{n+1}} \quad (14)$$

$$= S_{n+1} + \frac{\Delta S_{n+1}\Delta x_n - \lambda^2\Delta S_n\Delta x_{n+1}}{\Delta x_n - \lambda(\Delta x_n + \Delta x_{n+1}) + \lambda^2\Delta x_{n+1}}. \quad (15)$$

Formula (15) is more stable than (14).

The problem is again to express λ from the terms of the sequence (S_n). We assume that an annihilation operator for the sequence (Δx_n) is known, that is a linear operator L such that for all n, $L(\Delta x_n) = 0$. Such operators are quite useful in deriving sequence transformations (as in the case of the Aitken's process where $L \equiv \Delta$). They were introduced by Weniger [14]. Thus, applying L

to (13), we get, for all n,

$$L(S_{n+2}\Delta x_n) - \lambda L(S_{n+1}(\Delta x_n + \Delta x_{n+1})) + \lambda^2 L(S_n \Delta x_{n+1}) = 0. \tag{16}$$

This polynomial equation of degree 2 has 2 solutions and we don't know which solution is the right one. So, we will compute simultaneously λ and λ^2. For that, we write (16) for the indexes n and $n+1$, and we obtain a system of two linear equations in these two unknowns

$$\begin{aligned} \lambda L(S_{n+1}(\Delta x_n + \Delta x_{n+1})) - \lambda^2 L(S_n \Delta x_{n+1}) &= L(S_{n+2}\Delta x_n) \\ \lambda L(S_{n+2}(\Delta x_{n+1} + \Delta x_{n+2})) - \lambda^2 L(S_{n+1}\Delta x_{n+2}) &= L(S_{n+3}\Delta x_{n+1}). \end{aligned}$$

It must be noticed that this approach requires more terms of the sequence than solving directly the quadratic equation (16) for λ.

The solution of the system is

$$\begin{aligned} \lambda &= \left[L(S_{n+3}\Delta x_{n+1})L(S_n\Delta x_{n+1}) - L(S_{n+2}\Delta x_n)L(S_{n+1}\Delta x_{n+2})\right]/D_n \\ \lambda^2 &= \left[L(S_{n+3}\Delta x_{n+1})L(S_{n+1}(\Delta x_n + \Delta x_{n+1}))\right. \\ &\quad \left.-L(S_{n+2}\Delta x_n)L(S_{n+2}(\Delta x_{n+1} + \Delta x_{n+2}))\right]/D_n \end{aligned}$$

with

$$\begin{aligned} D_n &= L(S_n\Delta x_{n+1})L(S_{n+2}(\Delta x_{n+1} + \Delta x_{n+2})) \\ &\quad - L(S_{n+1}\Delta x_{n+2})L(S_{n+1}(\Delta x_n + \Delta x_{n+1})). \end{aligned}$$

Replacing λ and λ^2 in (14) or (15) completely defines our sequence transformation. Let us remark that, for sequences which are not of the form (2), a different transformation is obtained if the preceding expression for λ is used in (14) or (15), and then squared for getting λ^2.

Remark 2. The annihilation operator L used for obtaining (16) from (13) must be independent of n since it has to be applied to a linear combination of terms of the sequence (x_n) whose coefficients can depend on n. So L cannot be $\Delta\delta$ since, although it is an annihilation operator for (x_n), we have $\Delta\delta(\Delta x_n) \neq 0$.

Let us mention that annihilation operators are only known for quite simple sequences. For example, the annihilation operator corresponding to $x_n = n^k$ is $L \equiv \Delta^{k+1}$, since $\Delta^{k+1} n^k = \Delta^k(\Delta n^k)$. If, for all n, x_n is constant, the transformation (11) is recovered.

This construction could be extended to a kernel whose members have the form $S_n = S + P_k(x_n)\lambda^n$, where P_k is a polynomial of degree k. Indeed, since $\delta^k P_k(x_n)$ is a constant, we have, for all n,

$$\Delta\delta^k P_k(x_n) = \Delta\delta^k \left(\frac{S_n - S}{\lambda^n}\right) = 0.$$

However, the difficulty in applying the two techniques described above lies in the derivation and the solution of a system of equations involving λ and its powers. We will not pursue in this direction herein.

3 A second scalar kernel

We will now construct a sequence transformation with a kernel containing all sequences of the form

$$S_n = S + \frac{\lambda^n}{a + bx_n}, \quad n = 0, 1, \ldots \tag{17}$$

where S, a, b and λ are unknown scalars and (x_n) a known sequence.

We have, for all n, $\delta(a + bx_n) = \delta(\lambda^n/(S_n - S)) = b$. Thus

$$\lambda^n \left(\frac{\lambda}{S_{n+1} - S} - \frac{1}{S_n - S}\right) = b\Delta x_n,$$

which is a nonlinear equation in S. Therefore the problem is to bypass such a nonlinearity.

We have, for all n,

$$\Delta\delta(a + bx_n) = \Delta\delta\left(\frac{\lambda^n}{S_n - S}\right) = 0,$$

and, therefore,

$$\lambda\Delta x_n \left(\frac{\lambda}{S_{n+2} - S} - \frac{1}{S_{n+1} - S}\right) - \Delta x_{n+1}\left(\frac{\lambda}{S_{n+1} - S} - \frac{1}{S_n - S}\right) = 0.$$

Setting $e_n = S_n - S$ and reducing to the same denominator, we have

$$\lambda e_n \Delta x_n(\lambda e_{n+1} - e_{n+2}) - e_{n+2}\Delta x_{n+1}(\lambda e_n - e_{n+1}) = 0. \tag{18}$$

3.1 First technique

The main drawback of (18) is that it is a quadratic equation in S. Let us consider the particular case where Δx_n is constant. If we apply the operator Δ, then, by Leibniz's rule, $\Delta(e_{n+p}e_{n+q}) = e_{n+p+1}\Delta e_{n+q} + e_{n+q}\Delta e_{n+p}$. Since $\Delta e_{n+i} = \Delta S_{n+i}$, we obtain a linear expression in S. Remark that, for each product $e_{n+p}e_{n+q}$, we can choose, in Leibniz's rule, either $u_n = e_{n+p}$ and $v_n = e_{n+q}$ or vice versa. However, after simplification, all these choices lead to the same expression.

Applying Δ to (18), we obtain, when Δx_n is constant,

$$\begin{aligned}\lambda^2(e_{n+2}\Delta S_n + e_n\Delta S_{n+1}) - 2\lambda(e_{n+1}\Delta S_{n+2} + e_{n+2}\Delta S_n)&\\ + (e_{n+3}\Delta S_{n+1} + e_{n+1}\Delta S_{n+2}) = 0.&\end{aligned} \tag{19}$$

Define the transformation

$$T_n = \frac{N_n}{D_n} \tag{20}$$

with

$$\begin{aligned}N_n &= \lambda^2 S_{n+1}(S_{n+2} - S_n) - 2\lambda(S_{n+3}S_{n+1} - S_{n+2}S_n) + S_{n+2}(S_{n+3} - S_{n+1})\\ D_n &= \lambda^2(S_{n+2} - S_n) - 2\lambda(S_{n+3} - S_{n+2} + S_{n+1} - S_n) + (S_{n+3} - S_{n+1}).\end{aligned}$$

Then, by (19), the kernel of the transformation (20) includes all the sequences of the form (17).

It remains to compute λ, and the problem can be solved as above. If, in (19), we do not separate λ and S, the unknowns are λ^2, λ, $\lambda^2 S$, λS and S. So, writing (19) for the indexes $n, \ldots, n+4$, leads to a system of 5 linear equations in these 5 unknowns

$$\begin{aligned}&\lambda^2 S_{n+1+i}(S_{n+2+i} - S_{n+i}) - 2\lambda(S_{n+1+i}S_{n+3+i} - S_{n+i}S_{n+2+i})\\ &-\lambda^2 S(S_{n+2+i} - S_{n+i}) + 2\lambda S(S_{n+3+i} - S_{n+2+i} + S_{n+1+i} - S_{n+i})\\ &-S(S_{n+3+i} - S_{n+1+i}) = -S_{n+2+i}(S_{n+3+i} - S_{n+1+i}), \quad i = 0, \ldots, 4,\end{aligned} \tag{21}$$

which provides λ and λ^2.

This system can also be solved directly for the unknown S, thus providing another transformation whose kernel includes all sequences of the form (17).

Let us give a numerical example to illustrate this transformation. We consider the sequence

$$S_n \;=\; S + \frac{\lambda^n}{2+\alpha n} \;+\; b^n n^\beta, \quad n = 0, 1, \ldots \tag{22}$$

We took $S = 1, \lambda = -1.2, b = 0.1, \alpha = 1.1$, and $\beta = 2.5$. With these values, the first term in (S_n) diverges, while the second one tends to zero. This second term is a subdominant contribution to (S_n). On Figure 2, the solid lines represent, in a logarithmic scale, and from top to bottom, the absolute errors of S_n, of the Aitken's Δ^2 process (which uses 3 consecutive terms of the sequence to accelerate), of its first iterate (which uses 5 terms), and of its second iterate (which uses 7 terms). The dashed line corresponds to the error of (20) (which uses 8 terms).

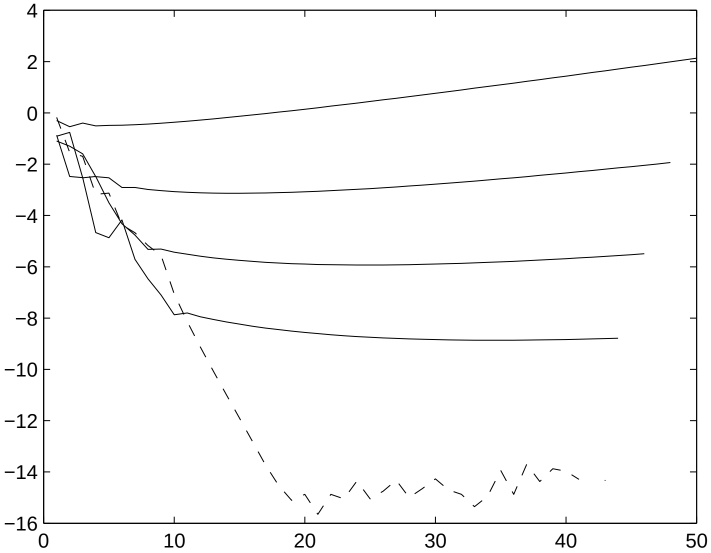

Figure 2 – The transformation (20) applied to (22), in dashed line.

We see that, as in our first example, the dominant contribution has been suppressed at the beginning and that, when n grows, the subdominant contribution is almost zero and this is why the sequence transformation no longer operates.

3.2 Second technique

Equation (18) can be written as

$$\begin{aligned}&\lambda^2 S_n S_{n+1}\Delta x_n - \lambda S_n S_{n+2}(\Delta x_n + \Delta x_{n+1}) + S_{n+1}S_{n+2}\Delta x_{n+1}\\ &\quad -S[\lambda^2(S_n + S_{n+1})\Delta x_n - \lambda(S_n + S_{n+2})(\Delta x_n + \Delta x_{n+1})\\ &\quad +(S_{n+1}+S_{n+2})\Delta x_{n+1}] = S^2(\lambda-1)[\Delta x_{n+1} - \lambda\Delta x_n].\end{aligned} \tag{23}$$

Remark 3. If the reciprocal difference operator is applied to the sequence $1/(a + bx_n) = (S_n - S)/\lambda^n$, we get, for all n, $\varrho^2(1/(a + bx_n)) = \varrho^2((S_n - S)/\lambda^n) = 0$, and (18) is exactly recovered.

For extracting S from (23), we need to know an annihilation operator L for the sequence $(\Delta x_{n+1} - \lambda\Delta x_n)$, and we obtain the following transformation whose kernel includes all sequences of the form (17)

$$T_n = \frac{N_n}{D_n}\,. \tag{24}$$

Here

$$\begin{aligned}N_n &= L(\lambda^2 S_n S_{n+1}\Delta x_n - \lambda S_n S_{n+2}(\Delta x_n + \Delta x_{n+1}) + S_{n+1}S_{n+2}\Delta x_{n+1})\\ D_n &= L(\lambda^2(S_n + S_{n+1})\Delta x_n - \lambda(S_n + S_{n+2})(\Delta x_n + \Delta x_{n+1})\\ &\quad + (S_{n+1}+S_{n+2})\Delta x_{n+1}).\end{aligned}$$

As explained above, if x_n is a polynomial of degree k in n, then $L \equiv \Delta^{k+1}$.

The problem is now to compute λ. However, if the unknowns λ and S are not separated in (23), we obtain, after applying the annihilation operator L, an equation in the unknowns λ^2, λ, $\lambda^2 S$, λS and S. So, writing (23) for the indexes $n, \ldots, n+4$, leads to a system of 5 linear equations in these 5 unknowns. This system can be solved for the unknown λ and its value used in (24). The system can also be solved for the unknowns λ and λ^2 and their values used in (24). Finally, the system can be solved directly for the unknown S. Thus several transformations whose kernels include all sequences of the form (17) can be obtained. When, for all n, $x_n = n$, the transformation (24) reduces to (20) since the λ's obtained from these systems are the halves of the λ corresponding to the system (21).

4 Other transformations

Let us now discuss some additional transformations.

In the particular case $x_n = n$, the following transformation also has a kernel including all sequences of the form (2). It was obtained by Durbin [10], and it is defined as

$$T_n \;=\; \frac{3S_{n+2}^2 - 4S_{n+1}S_{n+3} + S_nS_{n+4}}{6S_{n+2} - 4(S_{n+1} + S_{n+3}) + (S_n + S_{n+4})}\,, \qquad n = 0, 1, \ldots \qquad (25)$$

Let us mention that, in this particular case, the second Shanks transformation (the fourth column of the ε-algorithm) $e_2 : (S_n) \longmapsto \big(e_2(S_n) = \varepsilon_4^{(n)}\big)$ also has a kernel containing all sequences of the form (2), see [4].

We remark that the denominator of (25) is $\Delta^4 S_n$, and it is easy to see that this transformation can also be written as

$$T_n \;=\; \frac{S_n\Delta S_{n+3} - 3S_{n+1}\Delta S_{n+2} + 3S_{n+2}\Delta S_{n+1} - S_{n+3}\Delta S_n}{\Delta S_{n+3} - 3\Delta S_{n+2} + 3\Delta S_{n+1} - \Delta S_n}\,.$$

This expression leads to the idea of other transformations of a similar form with a denominator equal to

$$\Delta^{k+1}S_n \;=\; \Delta^k(\Delta S_n) \;=\; \sum_{i=0}^{k}(-1)^i C_k^i \Delta S_{n+k-i}\,,$$

where $C_k^i = k!/(i!(k-i)!)$ is the binomial coefficient. So, we obtain a whole class of transformations defined by

$$T_n \;=\; \frac{\displaystyle\sum_{i=0}^{k}(-1)^i C_k^i S_{n+i}\Delta S_{n+k-i}}{\displaystyle\sum_{i=0}^{k}(-1)^i C_k^i \Delta S_{n+k-i}}\,.$$

The kernels of these transformations are unknown, but it is easy to see that they all contain the kernel of the Aitken's process. These transformations have to be compared with the Δ^k processes [9] given by

$$T_n = \frac{\Delta^k(S_n/\Delta S_n)}{\Delta^k(1/\Delta S_n)} = \frac{\displaystyle\sum_{i=0}^{k}(-1)^i C_k^i S_{n+k-i}/\Delta S_{n+k-i}}{\displaystyle\sum_{i=0}^{k}(-1)^i C_k^i/\Delta S_{n+k-i}}\,.$$

The case $k = 1$ corresponds to the Aitken's Δ^2 process and, for $k = 2$, the second column $\theta_2^{(n)}$ of the θ-algorithm is recovered (see [5]). The kernel of this transformation is the set of sequences such that, for all n,

$$\Delta^k \left(\frac{S_n - S}{\Delta S_n} \right) = 0 \,,$$

that is

$$\frac{S_n - S}{\Delta S_n} = P_{k-1}(n) \,,$$

where P_{k-1} is a polynomial of degree $k - 1$ in n.

5 The vector case

The sequence transformations described in Sections 2 and 3 will now be used in the case where S_n and S are vectors of dimension p. Obviously, for a vector sequence, a scalar transformation could be used separately on each component. However, such a procedure is usually less efficient than using a transformation specially built for treating vector sequences.

We begin by the first kernel studied in Section 2. Different situations could be considered

$S_n = S + (a + bx_n)\lambda^n$ or $S_n = S + (a + x_n b)\lambda^n$			
$\lambda \in \mathbb{C}$	$a \in \mathbb{C}^p$	$bx_n \in \mathbb{C}^p$	$x_n b \in \mathbb{C}^p$
		$b \in \mathbb{C}, x_n \in \mathbb{C}^p$ $b \in \mathbb{C}^{p\times p}, x_n \in \mathbb{C}^p$	$b \in \mathbb{C}^p, x_n \in \mathbb{C}$ $b \in \mathbb{C}^p, x_n \in \mathbb{C}^{p\times p}$
$\lambda \in \mathbb{C}^p$	$a \in \mathbb{C}$ or $a \in \mathbb{C}^{p\times p}$	$b \in \mathbb{C}, x_n \in \mathbb{C}$	
		$bx_n \in \mathbb{C}^{p\times p}$	$x_n b \in \mathbb{C}^{p\times p}$
		$b \in \mathbb{C}, x_n \in \mathbb{C}^{p\times p}$ $b \in \mathbb{C}^{p\times p}, x_n \in \mathbb{C}^{p\times p}$	$b \in \mathbb{C}^{p\times p}, x_n \in \mathbb{C}$ $b \in \mathbb{C}^{p\times p}, x_n \in \mathbb{C}^{p\times p}$

$S_n = S + \lambda^n(a + bx_n)$ or $S_n = S + \lambda^n(a + x_n b)$			
$\lambda \in \mathbb{C}^{p\times p}$	$a \in \mathbb{C}^p$	$bx_n \in \mathbb{C}^p$	$x_n b \in \mathbb{C}^p$
		$b \in \mathbb{C}, x_n \in \mathbb{C}^p$ $b \in \mathbb{C}^{p\times p}, x_n \in \mathbb{C}^p$	$b \in \mathbb{C}^p, x_n \in \mathbb{C}$ $b \in \mathbb{C}^p, x_n \in \mathbb{C}^{p\times p}$

Since our purpose is only to show how to proceed with vector sequences, we will not treat all these cases. The procedures followed are similar to those described in [6] and [1].

Let us consider the first transformation in the case where λ and b are scalars, and a and x_n vectors. Formulae (5) and (6) are still valid. Taking the scalar product of (6) with two linearly independent vectors y_1 and y_2, and eliminating b gives

$$\begin{aligned}\lambda\big[&(y_1, \Delta x_{n+1})(y_2, \Delta S_{n+1}) - (y_2, \Delta x_{n+1})(y_1, \Delta S_{n+1})\\ &+ (y_1, \Delta x_n)(y_2, \Delta S_n) - (y_2, \Delta x_n)(y_1, \Delta S_n)\big]\\ &+ \lambda^2\big[(y_2, \Delta x_{n+1})(y_1, \Delta S_n) - (y_1, \Delta x_{n+1})(y_2, \Delta S_n)\big]\\ =\ &(y_1, \Delta x_n)(y_2, \Delta S_{n+1}) - (y_2, \Delta x_n)(y_1, \Delta S_{n+1}).\end{aligned}$$

Writing down this relation also for the index $n+1$ leads to a system of two equations in the two unknowns λ and λ^2, which completely defines the first vector transformation.

Another way of computing λ and λ^2 consists in writing down (6) for the indexes n and $n+1$ and taking the scalar products with a unique vector y. Eliminating b gives an equation in our two unknowns. Then, this equation is written down for the indexes n and $n+1$, thus leading again to a system of two equations.

For the second transformation, nothing has to be changed until (16) included. For the computation of λ and λ^2 one can proceed as for the first transformation. The relation (16) can be multiplied scalarly by y_1 and y_2. Thus, a system of two equations is obtained for the unknowns. It is also possible to write down (16) for the indexes n and $n+1$ and to multiply these two equations scalarly by the same vector y.

For the second kernel considered in Section 3, it can be written as $S_n = S + (a + bx_n)^{-1}\lambda^n$, which shows that $a + bx_n$ can be a matrix and λ a vector. Obviously, a and bx_n cannot be vectors. This second kernel can then be treated in a way similar to the first one.

6 Fixed point methods

There is a close connection between sequence transformations and fixed point iterations for finding $x \in \mathbb{R}^p$ such that $x = F(x)$, where F is a mapping of $\mathbb{R}^p$ into itself [2]. In this Section, we will see how to convert the vector sequence transformations of Section 5 into fixed point methods. The procedure is similar to obtaining Steffensen's method from Aitken's Δ^2 process in the case $p = 1$. The transformations obtained in this way are often related to quasi-Newton methods; see [3].

In each sequence transformation, the computation of T_n makes use of $S_n, \ldots, S_{n+m}$, where the value of m differs for each of them. An iteration of a fixed point method based on a sequence transformation consists in the following steps for computing the new iterate x_{n+1} from the previous one x_n

1. Set $S_0 = x_n$.
2. Compute $S_{i+1} = F(S_i)$ for $i = 0, \ldots, m-1$.
3. Apply the sequence transformation to $S_0, \ldots, S_m$, and compute T_0.
4. Set $x_{n+1} = T_0$.

7 Conclusions

In this paper, we discussed how to construct scalar sequence transformations with certain kernels generalizing the kernel of the Aitken's Δ^2 process. As we could see, generalizations for the types of kernels we considered are not so easy to obtain and the corresponding algorithms need some efforts to be implemented. However, we experimented some cases where they were quite effective. The convergence and the acceleration properties of these transformations remain to be studied. Then, we showed how extend some of these transformations to the case of vector sequences. Since we only gave the idea how to proceed, a systematic study of such vector transformations and their applications have to be pursued. Finally, we explained how to convert these vector transformations into fixed point iterations. They also have to be analyzed.

Acknowledgements. We would like to thank Mohamed Ait Tidili for a careful reading of a first version of this work, and pointing out some improvements. We also thank the referee for her/his careful reading, and her/his remarks and suggestions which helped us to clarify some points, and to ameliorate the paper.

REFERENCES

[1] C. Brezinski, Vector sequence transformations: methodology and applications to linear systems. J. Comput. Appl. Math., **98** (1998), 149–175.

[2] C. Brezinski, Dynamical systems and sequence transformations. J. Phys. A: Math. Gen., **34** (2001), 10659–10669.

[3] C. Brezinski, A classification of quasi-Newton methods. Numer. Algorithms, **33** (2003), 123–135.

[4] C. Brezinski and M. Crouzeix, Remarques sur la procédé Δ^2 d'Aitken. C.R. Acad. Sci. Paris, **270 A** (1970), 896–898.

[5] C. Brezinski and M. Redivo Zaglia, *Extrapolation Methods. Theory and Practice*. North-Holland, Amsterdam, 1991.

[6] C. Brezinski and M. Redivo Zaglia, Vector and matrix sequence transformations based on biorthogonality. Appl. Numer. Math., **21** (1996), 353–373.

[7] J.P. Delahaye, *Sequence Transformations*. Springer-Verlag, Berlin, 1988.

[8] J.P. Delahaye and B. Germain-Bonne, Résultats négatifs en accélération de la convergence. Numer. Math., **35** (1980), 443–457.

[9] J.E. Drummond, A formula for accelerating the convergence of a general series. Bull. Aust. Math. Soc., **6** (1972), 69–74.

[10] F. Durbin, Private communication, 20 June 2003.

[11] G. Fikioris, An application of convergence acceleration methods. IEEE Trans. Antennas Propagat., **47** (1999), 1758–1760.

[12] A. Navidi, Modification of Aitken's Δ^2 formula that results in a powerful convergence accelerator, in *Proceedings ICNAAM-2005*. T.E. Simos et al. eds., Viley-VCH, Weinheim, 2005, pp. 413–418.

[13] G. Walz, *Asymptotics and Extrapolation*. Akademie Verlag, Berlin, 1996.

[14] E.J. Weniger, Nonlinear sequence transformations for the acceleration of convergence and the summation of divergent series. Comput. Physics Reports, **10** (1989), 189–371.

[15] E.J. Weniger, Private communication, 12 September 2003.

[16] J. Wimp, *Sequence Transformations and Their Applications*. Academic Press, New York, 1981.