Extrapolation and prediction of sequences in a vector space

Claude Brezinski, Michela Redivo-Zaglia

Extrapolation and prediction of sequences in a vector space

Claude Brezinski* Michela Redivo-Zaglia†

Dedicated to the memory of our friend Luc Wuytack

Abstract

The aim of this paper is to present a general theoretical framework for the extrapolation and the prediction of sequences of elements in a vector space. Then, particular cases are studied and recursive algorithms for implementing some of the procedures obtained are discussed. Possible extensions of this work are evoked.

Keywords: extrapolation, polynomial extrapolation, Shanks extrapolation, sequence transformation, prediction, E-transformation.

Sequences play an important role in applied mathematics and in numerical analysis. They can be produced by an iterative process, for example in the solution of systems of linear or nonlinear equations, as the partial sums of series or the convergents of continued fractions for computing special functions, or in discretization processes for the approximate solution of integral or differential equations when the step size tends to zero, in quadrature methods, etc.. Their limit is usually the exact solution of the problem concerned. When the convergence is slow, the practical usefulness of these sequences is strongly reduced. However, they can be transformed in new sequences which, under certain assumptions, converge faster to the same limit. Another problem related to sequences is, knowing a certain consecutive number of their terms, to estimate the next ones.

These two problems can be simultaneously treated by *extrapolation*. The principal aim of this paper is to present a general theoretical framework for the extrapolation to the limit of a sequence whose terms are elements of a vector space, and for the prediction of its unknown terms. This functional analysis framework contains two approaches which are based on annihilation and difference operators. Then, particular cases leading to deeper results and to recursive algorithms are studied.

The scene is set up in Section 1. The theoretical framework of extrapolation is presented in Section 2, and used in Section 3 for the prediction of the missing terms of a sequence. Applications

*Université de Lille, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France. E-mail: Claude.Brezinski@univ-lille.fr.

†University of Padua, Department of Mathematics "Tullio Levi-Civita", Via Trieste 63, 35121 Padua, Italy. E-mail: Michela.RedivoZaglia@unipd.it.

of this framework are given in Section 4. The first one concerns the E-transformation which is a very general extrapolation method, the second one is the well known Shanks transformation. In some cases, recursive procedures for the implementation of these extrapolation methods are proposed. Concluding remarks and possible extensions of this work end the paper.

1 The scene

Let $\mathbb{E}$ be a vector space on $\mathbb{K}$ ($\mathbb{R}$ or $\mathbb{C}$), and let $\mathbb{E}^{\mathbb{N}}$ be the vector space of sequences of elements of $\mathbb{E}$. Elements of $\mathbb{E}$ are denoted by bold symbols.

The terms of any $(\mathbf{s}_n) \in \mathbb{E}^{\mathbb{N}}$ can be written under the form

$$\forall n, \quad \mathbf{s}_n = \mathbf{s} + \boldsymbol{\varphi}(n),$$

with $\mathbf{s} \in \mathbb{E}$, and where $\boldsymbol{\varphi}$ is some application from $\mathbb{N}$ to $\mathbb{E}$. If $\lim_{n\to\infty} \boldsymbol{\varphi}(n) = \mathbf{0} \in \mathbb{E}$, the sequence $(\mathbf{s}_n)$ converges, and $\mathbf{s}$ is its *limit*, otherwise it is called its *antilimit*.

Let $\mathbb{D} \subseteq \mathbb{K}^k$, and let now $\boldsymbol{\varphi} : \mathbb{N} \times \mathbb{D} \longmapsto \mathbb{E}$ be a given application depending on an arbitrary vector $a = (a_1, \ldots, a_k)^T \in \mathbb{D}$ of parameters, and let $\mathbb{V}_{\boldsymbol{\varphi},k}$ be the linear variety of sequences of elements of $\mathbb{E}$ such that, for all $a \in \mathbb{D}$,

$$\forall n, \quad \mathbf{s}_n = \mathbf{s} + \boldsymbol{\varphi}(n, a). \tag{1}$$

In this paper, we discuss the two following problems

- *Extrapolation* of the sequence $(\mathbf{s}_n)$, which consists in computing $\mathbf{s}$ (or an approximation of it) from a finite number of its consecutive terms.
- *Prediction* of the sequence $(\mathbf{s}_n)$, which consists in computing its unknown terms (or approximations of them) from a finite number of consecutive known ones.

2 Extrapolation

Let us now show how to compute $\mathbf{s}$, or an approximation of it, by an extrapolation procedure. Two different strategies are explained. The first of them, named *Polynomial Extrapolation*, was introduced in [2] for sequences of elements of a vector space, and then evoked again in [10]. In this paper, we still extend it. The second strategy, named *Shanks Extrapolation*, was never published before. Both strategies make use of annihilation and difference operators, and are related one to each other. They were, in fact, implicitly used before in the construction of extrapolation methods, but never presented for themselves in such a generality. The genesis of the transformations discussed below was described in [14] (see also [16]).

We begin by a definition due to Weniger [41]. Let $L : \mathbb{E}^{\mathbb{N}} \longmapsto \mathbb{E}^{\mathbb{N}}$ be a linear operator. It is named an *annihilation operator* for the sequence $(\mathbf{v} = (\mathbf{v}_n)) \in \mathbb{E}^{\mathbb{N}}$ if $(L\mathbf{v} = (L\mathbf{v}_n)) = (\mathbf{0} = (\mathbf{0})) \in \mathbb{E}^{\mathbb{N}}$, the sequence whose all terms are equal to $\mathbf{0} \in \mathbb{E}$. For example, the kth power $\Delta^k = \Delta(\Delta^{k-1})$ of the forward difference operator Δ defined by $\forall n, \Delta\mathbf{v}_n = \mathbf{v}_{n+1} - \mathbf{v}_n$ is an annihilation operator for

the sequence $(P_{k-1}(n))$ where P_{k-1} is a polynomial of degree $k-1$ at most. The same is true for the power $\nabla^k = \nabla(\nabla^{k-1})$ of the backward difference operator ∇ defined by $\nabla \mathbf{v}_n = \mathbf{v}_n - \mathbf{v}_{n-1}$. More generally, the kth power of the divided difference operator is an annihilation operator for the sequence $(P_{k-1}(x_n))$, where x_n is a sequence of parameters.

Applying an annihilation operator L for the constant sequence $(\mathbf{s})$ to (1) leads to

$$L\mathbf{s}_n = L\boldsymbol{\varphi}(n, a). \tag{2}$$

For computing the vector a, this relation in $\mathbb{E}$ has to be transformed into a scalar one, or, more precisely, into a set of scalar relations. For that purpose, let $\mathbb{E}^*$ be the algebraic dual vector space of $\mathbb{E}$, that is the vector space of linear functionals on $\mathbb{E}$. Its elements are also denoted by bold symbols. Let $\mathbf{y}_i \in \mathbb{E}^*$ for all i be linearly independent, and let $\langle \cdot, \cdot \rangle$ denote the duality product between $\mathbb{E}^*$ and $\mathbb{E}$. Let us mention that the $\mathbf{y}_i$'s can depend on n. Finally, let E be the operator which transforms an element of a sequence $(\mathbf{v}_n)$ of elements of $\mathbb{E}$ or $\mathbb{E}^*$ into its following one, that is $E\mathbf{v}_n = \mathbf{v}_{n+1}$. Its powers are defined as usual, E^0 being the identity. Notice that, from (1), we also have $E^i L\mathbf{s}_n = E^i L\boldsymbol{\varphi}(n, a)$ and $LE^i\mathbf{s}_n = LE^i\boldsymbol{\varphi}(n, a)$ by the linearity of L.

If $(\mathbf{s}_n) \in \mathbb{V}_{\boldsymbol{\varphi},k}$, then $(\mathbf{s}_n)$ has the form (1) with a uniquely defined vector a of parameters. If a is known, then, for all n, we immediately have $\mathbf{s} = \mathbf{s}_n - \boldsymbol{\varphi}(n, a)$. If a is unknown, it is first computed starting from relation (2) by using one of the strategies described in the sequel, and $\mathbf{s}$ follows from the preceding formula.

Now, if $(\mathbf{s}_n) \notin \mathbb{V}_{\boldsymbol{\varphi},k}$, let us consider a sequence $(\mathbf{t}_n = \mathbf{t} + \boldsymbol{\varphi}(n, a)) \in \mathbb{V}_{\boldsymbol{\varphi},k}$, and impose it the *interpolation conditions* $\mathbf{t}_{n+i} = \mathbf{s}_{n+i}$ for $i = 0, \ldots, k$. Assuming that it exists and is unique, we first compute the vector $a \in \mathbb{D}$ as the solution of a system of k scalar equations, and, then, we obtain $\mathbf{t}$. This procedure for obtaining $\mathbf{t}$ is named *extrapolation* of the sequence $(\mathbf{s}_n)$ since $\mathbf{t}$ should be an approximation of $\mathbf{s}$. Obviously, if $\mathbb{E}$ has a finite dimension, k must not be greater than this dimension.

It exists several strategies for computing first the vector a when it is unknown, and then $\mathbf{s}$ or its approximation $\mathbf{t}$.

- The first strategy, named *Polynomial Extrapolation* (PE), for computing the vector a consists in solving the system of equations

$$\langle E^i\mathbf{y}_1, L\mathbf{s}_n \rangle = \langle E^i\mathbf{y}_1, L\boldsymbol{\varphi}(n, a) \rangle, \quad i = 0, \ldots, k-1, \tag{3}$$

  where E^i applies on the lower index of $\mathbf{y}_1$, that is $E^i\mathbf{y}_1 = \mathbf{y}_{i+1}$.

- In the second strategy, called *Shanks Extrapolation* (SE), the vector a is the solution of the system

$$\langle \mathbf{y}_1, E^i L\mathbf{s}_n \rangle = \langle \mathbf{y}_1, E^i L\boldsymbol{\varphi}(n, a) \rangle, \quad i = 0, \ldots, k-1, \tag{4}$$

  where E^i applies on the index n, that is $E^i L\boldsymbol{\varphi}(n, a) = L\boldsymbol{\varphi}(n+i, a)$, and $E^i L\boldsymbol{s}_n = L\boldsymbol{s}_{n+i}$.

It can be noticed that the difference between the two strategies is the position of the operator E^i in the duality products. These two strategies can be mixed up and unified, and lead to obtaining

a as the solution of the following system where the powers of the operator E have been distributed on both sides of the duality product

$$\langle E^i \mathbf{y}_1, E^j L\mathbf{s}_n\rangle = \langle E^i \mathbf{y}_1, E^j L\boldsymbol{\varphi}(n,a)\rangle, \quad i+j=0,\ldots,k-1. \tag{5}$$

If j is fixed at 0, the PE is recovered, while, fixing i at 0, leads to the SE. Here, we will not consider the other choices.

Once a has been obtained, we set $\mathbf{t} = \mathbf{s}_n - \boldsymbol{\varphi}(n,a)$. Obviously, $\mathbf{t}$ depends on n, but also on the function $\boldsymbol{\varphi}$ chosen, and on its fixed number k of parameters. For these reasons, we will denote it by $\mathbf{E}_k(\mathbf{s}_n)$ for remembering the notation used by Shanks [34] in the scalar case. Thus, we have defined the *sequence transformation* (in fact, several transformations) $\mathbf{T}_k : (\mathbf{s}_n) \longmapsto (\mathbf{E}_k(\mathbf{s}_n))$ by

$$\mathbf{E}_k(\mathbf{s}_n) = \mathbf{s}_n - \boldsymbol{\varphi}(n,a), \quad n = 0,1,\ldots$$

The set $\mathbb{V}_{\boldsymbol{\varphi},k}$ is called the *kernel* of the transformation $\mathbf{T}_k$, and we have the following result for the sequences in it since, in this case, the uniquely defined vector a in $\boldsymbol{\varphi}$ has been exactly computed by any of the preceding strategies

Theorem 1
By construction, if $(\mathbf{s}_n) \in \mathbb{V}_{\boldsymbol{\varphi},k}$*, then, for all* n*,* $\mathbf{E}_k(\mathbf{s}_n) = \mathbf{s}$*.*

Remark 1 *Since the SE makes use of* k *successive relations* (1)*, we can also consider the transformations defined by* $\mathbf{E}_{k,i}(\mathbf{s}_n) = \mathbf{s}_{n+i} - \boldsymbol{\varphi}(n+i,a)$ *for* $n = 0,1,\ldots$*. Typical choices for* i *are* $i = 0$ *and* $i = k-1$*.*

Let us consider the particular case $\mathbb{E} = \mathbb{R}^k$, where k is the dimension of the vector a. The relation $L\mathbf{s}_n = L\boldsymbol{\varphi}(n,a)$ is a system of k equations in k unknowns, the components of the vector a. In this case, it is not necessary to introduce neither the dual space $\mathbb{E}^*$ nor the duality product. Solving this system gives a, and thus the corresponding vector $\mathbf{t}$. Thus, we directly obtain the sequence transformation given by $\mathbf{E}_k(\mathbf{s}_n) = \mathbf{s}_n - \boldsymbol{\varphi}(n,a)$, and, if $(\mathbf{s}_n) \in \mathbb{V}_{\boldsymbol{\varphi},k}$, the preceding Theorem holds since it corresponds, in fact, to the Modified Minimal Polynomial Extrapolation (MMPE) [3], which is the PE for the Shanks-transformation (see Section 4.2.1) with the particular choice $\mathbf{y}_i = \mathbf{e}_i$, the vectors of the canonical basis of $\mathbb{R}^k$.

Although the sequences produced by all transformations $\mathbf{T}_k$ are denoted by $(\mathbf{E}_k(\mathbf{s}_n))$, it must be understood that they do not furnish the same results for sequences outside $\mathbb{V}_{\boldsymbol{\varphi},k}$.

3 Prediction

When only a certain number of consecutive terms of a sequence are known, it could be interesting to obtain approximations of some of the next missing ones. This is the problem of *prediction* of the sequence $(\mathbf{s}_n)$. The idea of prediction goes back, as far as we know, to Gilewicz who used Padé approximants to predict the unknown coefficients of a formal power series [25]. The idea was taken back in detail by Sidi and Levin with another rational approximation of power series in [38] and continued in [39]; see also [32]. This approach had been used quite extensively in physics for the approximation of unknown coefficients of quantum theoretical perturbation expansions that

can be extremely demanding [28, 29], but also for tackling more theoretical question [1]. Using extrapolation methods, the idea was also exploited in [7] for scalar sequences, and then in [40]. Let us now generalize it to sequences in $\mathbb{E}$.

As for extrapolation, the principle of prediction consists in interpolating the sequence by a sequence of $\mathbb{V}_{\boldsymbol{\varphi},k}$, which is equivalent to assuming that $(\mathbf{s}_n)$ has the form $\mathbf{s}_n = \mathbf{s} + \boldsymbol{\varphi}(n, a)$. The first step of the prediction consists in estimating $\mathbf{s}$ by one of the extrapolation strategies described in Section 2, thus leading to an approximation $\mathbf{E}_k(\mathbf{s}_n)$ of it for k and n fixed, and a vector $a \in \mathbb{D} \subseteq \mathbb{R}^k$. If the computation of $\mathbf{E}_k(\mathbf{s}_n)$ and a need the knowledge of the elements $\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_p$ of the initial sequence, then an approximation $\mathbf{t}_m$ of $\mathbf{s}_m$ for any index m greater than the greatest index p used in the extrapolation process can be computed by

$$\mathbf{t}_m = \mathbf{E}_k(\mathbf{s}_n) + \boldsymbol{\varphi}(m, a), \quad m = p+1, p+2, \ldots.$$

Obviously, when m increases the quality of the approximation $\mathbf{t}_m$ of $\mathbf{s}_m$ decreases.

In some particular cases, such as the Shanks transformation described in Section 4.2, the vector a is used in $\boldsymbol{\varphi}(m, a)$ together with the vectors $\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_p$. In such a case, there exists another possibility for predicting the unknown terms of the sequence $(\mathbf{s}_n)$. As far as we known, it is new and it is as follows. After having computed $\mathbf{t}_{p+1}$ as described above, the same vector a could be used again in $\boldsymbol{\varphi}(p+2, a)$ but now with the vectors $\mathbf{s}_1, \ldots, \mathbf{s}_p, \mathbf{t}_{p+1}$, thus leading to an approximation $\mathbf{t}_{p+2}$ of $\mathbf{s}_{p+2}$. Then, the process can be repeated again with $\mathbf{s}_2, \ldots, \mathbf{s}_p, \mathbf{t}_{p+1}, \mathbf{t}_{p+2}$, and so on. For example, the Padé approximant $[p/q]$ of the series $f(t) = c_0 + c_1 t + \cdots$ can be used in that way for predicting the unknown coefficients c_{p+q+i} for $i = 1, 2, \ldots$.

4 Applications

Let us now apply the framework presented in Section 2. In the first example, the E-transformation, the function $\boldsymbol{\varphi}$ is the most general linear combination of sequences of elements of $\mathbb{E}$ that has been studied so far. The second example is an important particular case of it for which additional results are obtained.

4.1 The E-transformation

The E-transformation, with its corresponding recursive E-algorithm, was introduced in [5] for scalar sequences together with its extension to vector sequences. Let us enter it into our framework and extend it one step further to sequences in $\mathbb{E}$.

This transformation corresponds to

$$\boldsymbol{\varphi}(n, a) = a_1 \mathbf{g}_1(n) + \cdots + a_k \mathbf{g}_k(n), \quad n = 0, 1, \ldots,$$

where the $(\mathbf{g}_i(n))$'s are given sequences of elements of $\mathbb{E}$. Applying the annihilation operator L for constant sequences to (1), we obtain, since $L\boldsymbol{\varphi}(n, a) = L\mathbf{s}_n$,

$$a_0 L\mathbf{s}_n + a_1 L\mathbf{g}_1(n) + \cdots + a_k L\mathbf{g}_k(n) = \mathbf{0}, \quad n = 0, 1, \ldots, \tag{6}$$

with $a_0 = -1$ for homogenizing the notations.

4.1.1 Polynomial Extrapolation

Let us begin by the PE (3). The linear system to be solved for obtaining the coefficients a_i can be written as

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ \langle E^0\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Solving this system leads to the sequence transformation defined by

$$\mathbf{E}_k(\mathbf{s}_n) = \frac{\begin{vmatrix} \mathbf{s}_n & \mathbf{g}_1(n) & \cdots & \mathbf{g}_k(n) \\ \langle E^0\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{vmatrix}}{\begin{vmatrix} \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \\ \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{vmatrix}}. \tag{7}$$

The numerator is the element of $\mathbb{E}$ obtained by expanding it with respect to its first row by using the usual rules for that purpose.

This ratio can also be written as an extended Schur complement (see [8])

$$\mathbf{E}_k(\mathbf{s}_n) = \mathbf{s}_n - [\mathbf{g}_1(n), \ldots, \mathbf{g}_k(n)] \begin{pmatrix} \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \\ \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{pmatrix}^{-1} \begin{pmatrix} \langle E^0\mathbf{y}_1, L\mathbf{s}_n\rangle \\ \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_n\rangle \end{pmatrix}.$$

Let us now relate what precedes to the general problem of finite interpolation as described in [20, Sect. 2.2] and [9].

Denoting by $D_k^{(n)}$ the denominator of (7), let $\mathbf{I}_k$ be the operator

$$\mathbf{I}_k : \mathbb{E} \longmapsto \mathbb{E}_k = \text{span}\{L\mathbf{g}_1(n), \ldots, L\mathbf{g}_k(n)\}$$

defined by

$$\mathbf{I}_k(\cdot) = -\begin{vmatrix} \mathbf{0} & L\mathbf{g}_1(n) & \cdots & L\mathbf{g}_k(n) \\ \langle E^0\mathbf{y}_1, \cdot\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, \cdot\rangle & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{vmatrix} / D_k^{(n)}.$$

It is easy to see that

$$\mathbf{I}_k(L\mathbf{g}_i(n)) = L\mathbf{g}_i(n), \quad i = 1, \ldots, k \quad \text{and} \quad \mathbf{I}_k(L\mathbf{s}_n) = L\mathbf{s}_n = L\boldsymbol{\varphi}(n, a).$$

Let now $\mathbf{L}_k : \mathbb{E}^* \longmapsto \mathbb{E}_k^* = \text{span}\{E^0\mathbf{y}_1, \ldots E^{k-1}\mathbf{y}_1\}$ be defined by

$$\mathbf{L}_k(\cdot) = -\begin{vmatrix} \mathbf{0} & E^0\mathbf{y}_1 & \cdots & E^{k-1}\mathbf{y}_1 \\ \langle \cdot, L\mathbf{g}_1(n)\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle \cdot, L\mathbf{g}_k(n)\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{vmatrix} / D_k^{(n)}.$$

We have

$$\mathbf{L}_k(E^i\mathbf{y}_1) = E^i\mathbf{y}_1, \quad i = 0, \ldots, k-1,$$

and for all $(\mathbf{y}, \mathbf{u}) \in \mathbb{E}^* \times \mathbb{E}$, it holds

$$\langle \mathbf{y}, \mathbf{I}_k(\mathbf{u})\rangle = \langle \mathbf{L}_k(\mathbf{y}), \mathbf{u}\rangle = -\begin{vmatrix} 0 & \langle E^0\mathbf{y}_1, \mathbf{u}\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, \mathbf{u}\rangle \\ \langle \mathbf{y}, L\mathbf{g}_1(n)\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_1(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_1(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle \mathbf{y}, L\mathbf{g}_k(n)\rangle & \langle E^0\mathbf{y}_1, L\mathbf{g}_k(n)\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{g}_k(n)\rangle \end{vmatrix} / D_k^{(n)},$$

which shows that $\mathbf{L}_k$ is the dual operator of $\mathbf{I}_k$ [9], and we have, for $i = 0, \ldots, k-1$, from what precedes,

$$\langle E^i\mathbf{y}_1, L\mathbf{s}_n - \mathbf{I}_k(L\mathbf{s}_n)\rangle = \langle E^i\mathbf{y}_1 - \mathbf{L}_k(E^i\mathbf{y}_1), L\mathbf{s}_n\rangle = 0,$$

which are our interpolation conditions (3).

4.1.2 Shanks Extrapolation

For the SE (4), the vector a, augmented by a first component a_0, is the solution of the system

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ \langle \mathbf{y}_1, E^0L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^0L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^0L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1}L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_k(n)\rangle \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Solving this system leads, for $i = 0, \ldots, m$ where m is the largest index used to compute the quantities $E^{k-1}Lg_i(n)$ (at least k), to the transformations

$$\mathbf{E}_{k,i}(\mathbf{s}_n) = \frac{\begin{vmatrix} \mathbf{s}_{n+i} & \mathbf{g}_1(n+i) & \cdots & \mathbf{g}_k(n+i) \\ \langle \mathbf{y}_1, E^0L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^0L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^0L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1}L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_k(n)\rangle \end{vmatrix}}{\begin{vmatrix} \langle \mathbf{y}_1, E^0L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^0L\mathbf{g}_k(n)\rangle \\ \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_k(n)\rangle \end{vmatrix}}. \tag{8}$$

The ratio of determinants defining $\mathbf{E}_{k,i}(\mathbf{s}_n)$ can also be written under the form of an extended Schur complement

$$\mathbf{E}_{k,i}(\mathbf{s}_n) = \mathbf{s}_{n+i} - [\mathbf{g}_1(n+i), \ldots, \mathbf{g}_k(n+i)] \begin{pmatrix} \langle \mathbf{y}_1, E^0L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^0L\mathbf{g}_k(n)\rangle \\ \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1}L\mathbf{g}_k(n)\rangle \end{pmatrix}^{-1} \begin{pmatrix} \langle \mathbf{y}_1, E^0L\mathbf{s}_n\rangle \\ \vdots \\ \langle \mathbf{y}_1, E^{k-1}L\mathbf{s}_n\rangle \end{pmatrix}.$$

Contrarily to what was done for the PE, this transformation cannot be put into the framework of the general problem of finite interpolation as described in [9, 20] since it makes use of only one linear functional, say $\mathbf{y}_1$. However, letting $D_k^{(n)}$ be the denominator of (8), and setting

$$\mathbf{I}_k(\cdot) = -\begin{vmatrix} \mathbf{0} & L\mathbf{g}_1(n) & \cdots & L\mathbf{g}_k(n) \\ \langle \mathbf{y}_1, E^0\cdot\rangle & \langle \mathbf{y}_1, E^0 L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^0 L\mathbf{g}_k(n)\rangle \\ \vdots & \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1}\cdot\rangle & \langle \mathbf{y}_1, E^{k-1} L\mathbf{g}_1(n)\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1} L\mathbf{g}_k(n)\rangle \end{vmatrix} / D_k^{(n)},$$

we have, as in the preceding strategy,

$$\mathbf{I}_k(L\mathbf{g}_i(n)) = L\mathbf{g}_i(n), \quad i = 1, \ldots, k, \quad \text{and} \quad \mathbf{I}_k(L\mathbf{s}_n) = L\mathbf{s}_n = L\boldsymbol{\varphi}(n, a).$$

For the preceding reason, it is not possible to define an interpolation operator $\mathbf{L}_k$ in the dual space $\mathbb{E}^*$.

In the particular case $L \equiv \Delta$, there exists a recursive algorithm for computing the $\mathbf{E}_{k,0}(\mathbf{s}_n)$'s (now simply denoted by $\mathbf{E}_k(\mathbf{s}_n)$) without computing the determinants involved in their definition. It is the *E-algorithm* whose rules are

$$\begin{aligned} \mathbf{E}_k^{(n)} &= \mathbf{E}_{k-1}^{(n)} - \frac{\langle \mathbf{y}_1, \Delta\mathbf{E}_{k-1}^{(n)}\rangle}{\langle \mathbf{y}_1, \Delta\mathbf{g}_{k-1,k}^{(n)}\rangle}\mathbf{g}_{k-1,k}^{(n)}, \quad k, n = 0, 1, \ldots, \\ \mathbf{g}_{k,i}^{(n)} &= \mathbf{g}_{k-1,i}^{(n)} - \frac{\langle \mathbf{y}_1, \Delta\mathbf{g}_{k-1,i}^{(n)}\rangle}{\langle \mathbf{y}_1, \Delta\mathbf{g}_{k-1,k}^{(n)}\rangle}\mathbf{g}_{k-1,k}^{(n)}, \quad k, n = 0, 1, \ldots; \quad i = k+1, k+2, \ldots, \end{aligned}$$

with $\mathbf{E}_0^{(n)} = \mathbf{s}_n$ and $\mathbf{g}_{0,i}^{(n)} = \mathbf{g}_i(n)$ for $i = 1, 2, \ldots$
Setting $\mathbf{g}_{0,0}^{(n)} = \mathbf{g}_0(n) = \mathbf{s}_n$, the first rule can be included into the second one which, in this case, has to be used for $i = k, k+1, \ldots$ This algorithm extends the vector E-algorithm derived in [5], and we have the following result whose proof follows the same lines as that given in [5, 6]

Theorem 2

$$\mathbf{E}_k^{(n)} = \mathbf{E}_k(\mathbf{s}_n), \quad \mathbf{g}_{k,i}^{(n)} = \mathbf{E}_k(\mathbf{g}_i(n)), \quad k, n = 0, 1, \ldots$$

By a proof similar to that of Theorem 10 in [5], we have the following result which gives the expansion of the error $\mathbf{E}_k(\mathbf{s}_n) - \mathbf{s}$ if that of $\mathbf{s}_n - \mathbf{s}$ is known

Theorem 3
If $\mathbf{s}_n = \mathbf{s} + a_1\mathbf{g}_1(n) + a_2\mathbf{g}_2(n) + \cdots$, *then*

$$\mathbf{E}_k(\mathbf{s}_n) = \mathbf{s} + a_{k+1}\mathbf{g}_{k,k+1}^{(n)} + a_{k+2}\mathbf{g}_{k,k+2}^{(n)} + \cdots$$

4.2 The Shanks transformation

Shanks transformation for scalar sequences has been introduced by Shanks [34]. It was extended to sequences in $\mathbb{E}$ in [3]. Its kernel is the set of sequences satisfying, for all n, the difference equation

$$\alpha_0(\mathbf{s}_n - \mathbf{s}) + \cdots + \alpha_k(\mathbf{s}_{n+k} - \mathbf{s}) = \mathbf{0}, \tag{9}$$

with $\alpha_0\alpha_k \neq 0$, and $\alpha_0 + \cdots + \alpha_k = 1$, a condition which does not restrict the generality. The interest of this transformation is based on the fact that sequences generated by a linear iterative process of the form $\mathbf{s}_{n+1} = M\mathbf{s}_n + \mathbf{b}$ have this form, and that those produced by a nonlinear one $\mathbf{s}_{n+1} = F(\mathbf{s}_n)$ are close to it.

Equation (9) can also be written under two different forms similar to (1), that is

$$\mathbf{s}_n = \mathbf{s} + a_1\Delta\mathbf{s}_n + \cdots + a_k\Delta\mathbf{s}_{n+k-1}, \tag{10}$$

with $a_i = -\sum_{j=i}^{k}\alpha_j$, or

$$\mathbf{s}_n = \mathbf{s} + \widetilde{a}_1\Delta\mathbf{s}_n + \cdots + \widetilde{a}_k\Delta^k\mathbf{s}_n, \tag{11}$$

where the coefficients $\widetilde{a}_i$ can easily be related to the α_i's.

Remark 2 *Notice that, using the operator ∇, (9) can also be written under the form*

$$\mathbf{s}_{n+k} = \mathbf{s} + \bar{a}_1\nabla\mathbf{s}_{n+k} + \cdots + \bar{a}_k\nabla\mathbf{s}_{n+1},$$

or with the powers of ∇, which allows a different writing for including it into the E-transformation.

The relations (10) and (11) show that the Shanks transformation enters into the framework of the E-transformation for the particular choices $\mathbf{g}_i(n) = \Delta\mathbf{s}_{n+i-1}$ and $\mathbf{g}_i(n) = \Delta^i\mathbf{s}_n$ for $i = 1, \ldots, k$, respectively. Moreover, for the first of these choices, additional results can be obtained as will be seen below.

Applying the annihilation operator L, (10) becomes

$$a_0L\mathbf{s}_n + a_1L\Delta\mathbf{s}_n + \cdots + a_kL\Delta\mathbf{s}_{n+k-1} = \mathbf{0}, \tag{12}$$

with $a_0 = -1$.

4.2.1 Polynomial Extrapolation

Replacing $\mathbf{g}_i(n)$ by $\Delta\mathbf{s}_{n+i-1}$ in (7) leads to a particular transformation given by the same ratio of determinants. Let us add a first row $(1, 0, \ldots, 0)$ in the denominator of (7), and a first column identical to the first column of the numerator from its second term. Then, in the numerator and in the denominator, replacing each column, from the second one, by its sum with the preceding one shows that this transformation can also be written as

$$\mathbf{E}_k(\mathbf{s}_n) = \frac{\begin{vmatrix} \mathbf{s}_n & \mathbf{s}_{n+1} & \cdots & \mathbf{s}_{n+k} \\ \langle E^0\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^0\mathbf{y}_1, L\mathbf{s}_{n+1}\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{s}_{n+k}\rangle \\ \vdots & \vdots & & \vdots \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_{n+1}\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_{n+k}\rangle \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \langle E^0\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^0\mathbf{y}_1, L\mathbf{s}_{n+1}\rangle & \cdots & \langle E^0\mathbf{y}_1, L\mathbf{s}_{n+k}\rangle \\ \vdots & & \vdots & \\ \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_n\rangle & \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_{n+1}\rangle & \cdots & \langle E^{k-1}\mathbf{y}_1, L\mathbf{s}_{n+k}\rangle \end{vmatrix}}. \tag{13}$$

For $L \equiv \Delta$, this transformation was defined by Brezinski [3] in 1975 and by Pugachev [33, Sect. 7, pp. 352-3] in 1978. It was also given and named the *Modified Minimal Polynomial Extrapolation* with the acronym MMPE in 1986 [37]. If the $\mathbf{y}_i$'s are independent of n, this transformation can be recursively implemented by the $S\boldsymbol{\beta}$ algorithm of Jbilou [26] (see also [27]). This algorithm is still valid in our case with this independency assumption on n for the $\mathbf{y}_i$'s and for any annihilation operator L. We set $\mathbf{E}_k^{(n)} = \mathbf{E}_k(\mathbf{s}_n)$ and $\boldsymbol{\beta}_k^{(n)}$ is the element of $\mathbb{E}$ obtained by replacing the first row of (13) by $L\mathbf{s}_n, \ldots, L\mathbf{s}_{n+k}$. The proof exactly follows the same lines as the new proof of the algorithm given in [16, pp.185-6].

Theorem 4
For $k = 1, 2, \ldots$ and $n = 0, 1, \ldots$, we have

$$\begin{aligned} \mathbf{E}_k^{(n)} &= \frac{\mathbf{E}_{k-1}^{(n)} - a_k^{(n)}\mathbf{E}_{k-1}^{(n+1)}}{1 - a_k^{(n)}}, \\ \boldsymbol{\beta}_k^{(n)} &= \frac{\boldsymbol{\beta}_{k-1}^{(n)} - a_k^{(n)}\boldsymbol{\beta}_{k-1}^{(n+1)}}{1 - a_k^{(n)}}, \end{aligned}$$

with $\mathbf{E}_0^{(n)} = \mathbf{s}_n$ and $\boldsymbol{\beta}_0^{(n)} = \Delta\mathbf{s}_n$ for $n = 0, 1, \ldots$, and where

$$a_k^{(n)} = \langle E^{k-1}\mathbf{y}_1, \boldsymbol{\beta}_{k-1}^{(n)}\rangle / \langle E^{k-1}\mathbf{y}_1, \boldsymbol{\beta}_{k-1}^{(n+1)}\rangle.$$

Let us mention that an algorithm more economical than the $S\boldsymbol{\beta}$ algorithm, and also valid for sequences of elements of a general vector space, was proposed in [23].

Remark 3 *If $\mathbb{E} \equiv \mathbb{R}^p$, $p > 1$ arbitrary, if $L \equiv \Delta$, the choice $\mathbf{y}_1 = \Delta\mathbf{s}_n$ leads to the Minimal Polynomial Extrapolation method (MPE) of Cabay and Jackson [18] in 1976 (notice that this method was already given, but not studied, in [3, p. 356]), and the choice $\mathbf{y}_1 = \Delta^2\mathbf{s}_n$ corresponds to the Reduced Rank Extrapolation (RRE) obtained, in chronological order of discovery, by Kaniel and Stein [30] in 1974, Mešina [31] in 1977, Germain-Bonne [24] in 1978, and finally Eddy [22] in 1979. The algorithm given in [23] also allows to implement recursively the MPE and the RRE.*

4.2.2 Shanks Extrapolation

Replacing now $\mathbf{g}_i(n)$ by $\Delta\mathbf{s}_{n+i-1}$ in (8) leads to a similar transformation and a similar extended Schur complement formula. As for the PE, replacing each column in the denominator of the determinantal formula of the transformation (after adding it the first row $(1, 0, \ldots, 0)$ and a first column identical to the first column of the numerator from its second term) and in its numerator by its sum with the following one, we obtain

$$\mathbf{E}_{k,i}(\mathbf{s}_n) = \frac{\begin{vmatrix} \mathbf{s}_{n+i} & \mathbf{s}_{n+i+1} & \cdots & \mathbf{s}_{n+i+k} \\ \langle \mathbf{y}_1, E^0 L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^0 L\mathbf{s}_{n+1}\rangle & \cdots & \langle \mathbf{y}_1, E^0 L\mathbf{s}_{n+k}\rangle \\ \vdots & \vdots & & \vdots \\ \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_{n+1}\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_{n+k}\rangle \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \langle \mathbf{y}_1, E^0 L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^0 L\mathbf{s}_{n+1}\rangle & \cdots & \langle \mathbf{y}_1, E^0 L\mathbf{s}_{n+k}\rangle \\ \vdots & & \vdots & \\ \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_n\rangle & \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_{n+1}\rangle & \cdots & \langle \mathbf{y}_1, E^{k-1} L\mathbf{s}_{n+k}\rangle \end{vmatrix}}. \tag{14}$$

For the choice $L \equiv \Delta$ and $\mathbf{y}_1$ independent of n, the *Topological Shanks Transformations* introduced in [3] are recovered. The first of these transformations corresponds to $i = 0$, and the second one to $i = k$. They can be recursively implemented by the *Topological ε-algorithms* (TEA1 and TEA2, respectively), or, more economically in terms of storage and arithmetical operations, by the *Simplified Topological ε-algorithms* (STEA1 and STEA2) [12, 13]. These transformations can also be implemented by the ETEA given in [36, Chap. 5]. However, this algorithm is not recursive, and it needs that all the computations be done again when k increases, which is not the case with our algorithms.

Let us now show how to recover the rule of the first Simplified Topological ε-Algorithm (STEA1) from the rules of the E-algorithm. We set $\mathbf{E}_k^{(n)} = \mathbf{E}_{k,0}^{(n)} = \mathbf{E}_k(\mathbf{s}_n)$. Using the notation of the STEA1, we also have $\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)} = \mathbf{E}_k^{(n)}$. It has to be reminded that the STEA1 comes out from the first Topological ε-Algorithm (TEA1), and that, in this algorithm, the inverse of a couple $(\mathbf{y}, \mathbf{u}) \in \mathbb{E}^* \times \mathbb{E}$ is the couple $(\mathbf{u}^{-1}, \mathbf{y}^{-1}) \in \mathbb{E}^* \times \mathbb{E}$ where $\mathbf{u}^{-1} = \mathbf{y}/\langle \mathbf{y}, \mathbf{u}\rangle \in \mathbb{E}^*$ and $\mathbf{y}^{-1} = \mathbf{u}/\langle \mathbf{y}, \mathbf{u}\rangle \in \mathbb{E}$. Thus, the inverse of a single element of $\mathbb{E}$ or $\mathbb{E}^*$ has no meaning in itself since it is always defined with respect to another element of its dual space. In the rules of the TEA, the inverse of $\mathbf{y}_1$ is taken with respect to $\Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}$, that is $\mathbf{y}_1^{-1} = \Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}/\langle \mathbf{y}_1, \Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}\rangle = \Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}/\Delta\varepsilon_{2k}^{(n)}$ where $\varepsilon_{2k}^{(n)}$ is obtained by applying the scalar ε-algorithm of Wynn [43] to the scalar sequence $(\langle \mathbf{y}_1, \mathbf{s}_n\rangle)$ (see page 334 of [3]). We indeed have $\langle \mathbf{y}_1, \mathbf{y}_1^{-1}\rangle = 1$. It was this relation that allows us to derive the STEA1 from the TEA1.

We start our journey to the STEA1 from the main rule of the E-algorithm

$$\mathbf{E}_{k+1}^{(n)} = \mathbf{E}_k^{(n)} - \frac{\langle \mathbf{y}_1, \Delta\mathbf{E}_k^{(n)}\rangle}{\langle \mathbf{y}_1, \Delta\mathbf{g}_{k,k+1}^{(n)}\rangle}\mathbf{g}_{k,k+1}^{(n)}. \tag{15}$$

We have

$$\mathbf{g}_{k,k+1}^{(n)} = (-1)^k \frac{\begin{vmatrix} \Delta\mathbf{s}_n & \cdots & \Delta\mathbf{s}_{n+k} \\ \langle \mathbf{y}_1, \Delta^2\mathbf{s}_n\rangle & \cdots & \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+k}\rangle \\ \vdots & & \vdots \\ \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+k-1}\rangle & \cdots & \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+2k-1}\rangle \end{vmatrix}}{\begin{vmatrix} \langle \mathbf{y}_1, \Delta^2\mathbf{s}_n\rangle & \cdots & \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+k-1}\rangle \\ \vdots & & \vdots \\ \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+k-1}\rangle & \cdots & \langle \mathbf{y}_1, \Delta^2\mathbf{s}_{n+2k-2}\rangle \end{vmatrix}}.$$

To simplify, the denominator of this expression will be denoted by $\bar{H}_k^{(n)}$ and the same determinant with all the Δ^2 replaced by Δ by $H_k^{(n)}$. From the last relation on page 338 in the proof of the

Topological ε-algorithm [3], we have

$$\mathbf{g}_{k,k+1}^{(n)} = (-1)^k \mathbf{y}_1^{-1} H_{k+1}^{(n)} / \bar{H}_k^{(n)}.$$

We also have $\langle \mathbf{y}_1, \mathbf{g}_{k,k+1}^{(n)} \rangle = (-1)^k H_{k+1}^{(n)} / \bar{H}_k^{(n)}$, and it follows

$$\langle \mathbf{y}_1, \Delta\mathbf{g}_{k,k+1}^{(n)} \rangle = (-1)^k (H_{k+1}^{(n+1)} \widetilde{H}_k^{(n)} - H_{k+1}^{(n)} \widetilde{H}_k^{(n+1)}) / \widetilde{H}_k^{(n)} \widetilde{H}_k^{(n+1)}.$$

By relation (12) of [4], the numerator of this expression is equal to $H_k^{(n+1)} \widetilde{H}_{k+1}^{(n)}$. From the relations (54) and (55) of [4], it also holds that

$$\begin{aligned} \varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n)} &= -(H_{k+1}^{(n)})^2 / \widetilde{H}_k^{(n)} \widetilde{H}_{k+1}^{(n)} \\ \varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)} &= H_{k+1}^{(n)} H_k^{(n+1)} / \widetilde{H}_k^{(n)} \widetilde{H}_k^{(n+1)}. \end{aligned}$$

Thus, it follows

$$\frac{\varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n)}}{\varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)}} = (-(H_{k+1}^{(n)})^2 / \widetilde{H}_k^{(n)} \widetilde{H}_{k+1}^{(n)}) / (H_{k+1}^{(n)} H_k^{(n+1)} / \widetilde{H}_k^{(n)} \widetilde{H}_k^{(n+1)}) = -\frac{H_{k+1}^{(n)} \widetilde{H}_k^{(n+1)}}{H_k^{(n+1)} \widetilde{H}_{k+1}^{(n)}},$$

a relation that can also be obtained by taking the duality product of (15) with $\mathbf{y}_1$ and using the preceding determinantal expressions. Plugging all these relations into (15), we obtain

$$\begin{aligned} \widehat{\boldsymbol{\varepsilon}}_{2k+2}^{(n)} &= \widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)} - \left(\frac{\Delta\varepsilon_{2k}^{(n)}}{(-1)^k H_k^{(n+1)} \widetilde{H}_{k+1}^{(n)} / \widetilde{H}_k^{(n)} \widetilde{H}_k^{(n+1)}} \right) \left((-1)^k \frac{\Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}}{\Delta\varepsilon_{2k}^{(n)}} \frac{H_{k+1}^{(n)}}{\widetilde{H}_k^{(n)}} \right) \\ &= \widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)} - \frac{H_{k+1}^{(n)} \widetilde{H}_k^{(n+1)}}{H_k^{(n+1)} \widetilde{H}_{k+1}^{(n)}} \Delta\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}, \end{aligned}$$

which is one of the form of the STEA1 [12], namely

$$\widehat{\boldsymbol{\varepsilon}}_{2k+2}^{(n)} = \widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)} + \frac{\varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n)}}{\varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)}} (\widehat{\boldsymbol{\varepsilon}}_{2k}^{(n+1)} - \widehat{\boldsymbol{\varepsilon}}_{2k}^{(n)}).$$

5 Concluding remarks

In this paper, we present a general theoretical framework for the extrapolation and the prediction of sequences of elements belonging to a vector space $\mathbb{E}$. Extrapolation of a sequence aims at constructing a new sequence converging faster to the same limit under some assumptions, or tending to its antilimit if it does not converge. Prediction is a technique for estimating some of the unknown terms of a sequence.

Two strategies for computing the unknown scalars involved in the expression of the error term of the sequence considered were explored: the Polynomial Extrapolation (PE) defined by (3), and the Shanks Extrapolation (SE) given by (4). The PE needs less terms of the sequence but several elements of $\mathbb{E}^*$, the dual space of $\mathbb{E}$. The SE only requires one element of $\mathbb{E}^*$, but more terms of the sequence to be treated.

These two strategies were first applied to sequences whose error term is a linear combination of the most general possible terms. It is the E-transformation whose theoretical aspects are treated in detail. For the PE (3), no recursive procedure for its implementation is known so far, while, in the case of the SE (4), the E-algorithm allows such an implementation.

An important particular case of the E-transformation, the Shanks transformation, was then studied. Both strategies can be implemented by a recursive algorithm: the $S\boldsymbol{\beta}$-algorithm in the case of the PE, and the topological ε-algorithms for the SE. The connection between these ε-algorithms and the E-algorithm was established.

Our framework opens the way to new developments. We showed that the two strategies can be mixed up to issue in a new one, never exploited, defined by (5), which uses several elements of $\mathbb{E}^*$, and a number of elements of the sequence between those needed by the two other strategies. This approach could possibly be of interest for some applications. The sequence transformations (but not the recursive algorithms) studied in this paper are still valid if the $\mathbf{y}_i$'s depend on a fixed value of n. Moreover, in the case where $\mathbb{E}^* \equiv \mathbb{E}$, two possible choices, each of them having two different possibilities, can be of interest. The first one is in the spirit of the MPE: $\mathbf{y}_i = L\mathbf{s}_{n+i-1}$ or $\mathbf{y}_i = \mathbf{g}_i(n)$. The second choice looks like the RRE: $\mathbf{y}_i = L^2\mathbf{s}_{n+i-1}$ or $\mathbf{y}_i = L\mathbf{g}_i(n)$. The concept of *coupled sequence* introduced in [17] can also be exploited in our context without difficulty. However, for simplicity, we have not included it here, and this extension is left to the interested reader. Another extension of this work, is to compute the vector a in (3) and (4) as the solution in the least squares sense of an over determined system of equations obtained by replacing k by a larger integer, thus following what was done by Cordellier for scalar sequences [19]. Finally, these strategies can be used for extending other sequence transformations built for scalar sequences to sequences in $\mathbb{E}$. The case of rectangular matrix sequences satisfying a difference equation with rectangular matrix coefficients was treated in [15]. It can easily be fitted into our framework.

The effectiveness of this framework and of these two strategies can be seen from the many existing algorithms, see, for example, [11,14,16,21,35,36,41,42], and the papers on their applications which are too numerous to be quoted here.

Acknowledgments: C.B. acknowledges support from the Labex CEMPI (ANR-11-LABX-0007-01). The work of M.R.-Z. was partially supported by the University of Padua, Project No. DOR2007788/20 Numerical Linear Algebra and Extrapolation methods with applications. M.R.-Z. is a member of the INdAM (Istituto Nazionale di Alta Matematica "Francesco Severi") Research group GNCS. The authors thank the reviewers for their interesting comments.

References

[1] C.M. Bender, E.J. Weniger, Numerical evidence that the perturbation expansion for a non-Hermitian $\mathcal{PT}$-symmetric Hamiltonian is Stieltjes, J. Math. Phys. 42 (2001) 2167-2183.

[2] C. Brezinski, *Méthodes d'Accélération de la Convergence en Analyse Numérique*, Thèse de Doctorat d'État, Université Scientifique et Médicale de Grenoble, 1971. https://tel.archives-ouvertes.fr/tel-00282774

[3] C. Brezinski, Généralisation de la transformation de Shanks, de la table de Padé et de l'ε-algorithme, Calcolo, 12 (1975) 317-360.

[4] C. Brezinski, Sur le calcul de certains rapports de déterminants, in *Padé Approximation and its Applications*, L. Wuytack ed., Lecture Notes in Mathematics, vol. 765, Springer-Verlag, Heidelberg, 1979, pp. 184-210.

[5] C. Brezinski, A general extrapolation algorithm, Numer. Math., 35 (1980) 175-187.

[6] C. Brezinski, Recursive interpolation, extrapolation and projection, J. Comput. Appl. Math., 9 (1983) 369-376.

[7] C. Brezinski, Prediction properties of some extrapolation methods, Appl. Numer. Math., 1 (1985) 457-462.

[8] C. Brezinski, Other manifestations of the Schur complement, Linear Algebra Appl., 111 (1988) 231-247.

[9] C. Brezinski, *Biorthogonality and its Applications to Numerical Analysis*, Marcel Dekker, New York, 1992.

[10] C. Brezinski, S. Cipolla, M. Redivo-Zaglia, Y. Saad, Shanks and Anderson-type acceleration techniques for systems of nonlinear equations, IMA J. Numer. Anal., published online 25 August 2021, doi.org/10.1093/imanum/drab061.

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

[12] C. Brezinski, M. Redivo-Zaglia, The simplified topological ε-algorithms for accelerating sequences in a vector space, SIAM J. Sci. Comput., 36 (2014) A2227-A2247.

[13] C. Brezinski, M. Redivo-Zaglia, The simplified topological ε-algorithms: software and applications, Numer. Algorithms, 74 (2017) 1237-1260.

[14] C. Brezinski, M. Redivo-Zaglia, The genesis and early developments of Aitken's process, Shanks' transformation, the ε-algorithm, and related fixed point methods, Numer. Algorithms, 80 (2019) 11-133.

[15] C. Brezinski, M. Redivo-Zaglia, Matrix Shanks transformations, Electron. J. Linear Algebra, 35 (2019) 248-265.

[16] C. Brezinski, M. Redivo-Zaglia, *Extrapolation and Rational Approximation. The Works of the Main Contributors*, Springer Nature, Cham, Switzerland, 2020.

[17] C. Brezinski, M. Redivo-Zaglia, Y. Saad, Shanks sequence transformations and Anderson acceleration, SIAM Rev., 60(3) (2018) 646-669.

[18] S. Cabay, L.W. Jackson, A polynomial extrapolation method for finding limits and antilimits of vector sequences, SIAM J. Numer. Anal., 13 (1976) 734-752.

[19] F. Cordellier, *Interpolation Rationnelle et autres Questions : Aspects Algorithmiques et Numériques*, Thèse de Doctorat d'État ès Sciences Mathématiques, Université des Sciences et Techniques de Lille, 1989.

[20] P.J. Davis, *Interpolation and Approximation*, Blaisdell Publ. Comp., Waltham, MA, 1963.

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

[22] R.P. Eddy, Extrapolation to the limit of a vector sequence, in *Information Linkage between Applied Mathematics and Industry*, P.C.C. Wang ed., Academic Press, New York, 1979, pp. 387-396.

[23] W.F. Ford, A. Sidi, Recursive algorithms for vector extrapolation methods, Appl. Numer. Math., 4 (1988) 477–489.

[24] B. Germain-Bonne, *Estimation de la Limite de Suites et Formalisation de Procédés d'Accélération de la Convergence*, Thèse de Doctorat d'État ès Sciences Mathématiques, Université des Sciences et Techniques de Lille, 1978.

[25] J. Gilewicz, Numerical detection of the best Padé approximant and determination of the Fourier coefficients of insufficiently sampled functions, in *Padé Approximants and Their Applications*, P.R. Graves-Morris, ed., Academic Press, New York, 1973, pp. 99-103.

[26] K. Jbilou, *Méthodes d'Extrapolation et de Projection. Applications aux Suites de Vecteurs*, Thèse de 3ème cycle, Université des Sciences et Techniques de Lille, 1988.

[27] K. Jbilou, H. Sadok, Some results about vector extrapolation methods and related fixed point iteration, J. Comp. Appl. Math., 36 (1991) 385-398.

[28] U.D. Jentschura, J. Becher, E.J. Weniger, G. Soff, Resummation of QED perturbation series by sequence transformations and the prediction of perturbative coefficients, Phys. Rev. Lett., 85 (2000) 2446-2449.

[29] U.D. Jentschura, E.J. Weniger, G. Soff, Asymptotic improvement of resummations and perturbative predictions in quantum field theory, J. Phys. G, 26 (2000) 1545-1568.

[30] S. Kaniel, J. Stein, Least-square acceleration of iterative methods for linear equations, J. Optim. Theory Appl., 14 (1974) 431-437.

[31] M. Mešina, Convergence acceleration for the iterative solution of $x = Ax+f$, Comput. Methods Appl. Mech. Eng., 10 (1977) 165-173.

[32] M. Prévost, D. Vekemans, Partial Padé prediction, Numer. Algor., 20 (1999) 23-50.

[33] B.P. Pugachev, Acceleration of convergence of iterative processes and a method of solving systems of non-linear equations, USSR Comput. Maths. Maths. Phys., 17 (5) (1978) 199-207.

[34] D. Shanks, Non linear transformations of divergent and slowly convergent sequences, J. Math. and Phys., 34 (1955) 1-42.

[35] A. Sidi, *Practical Extrapolation Methods. Theory and Applications*, Cambridge University Press, Cambridge, 2003.

[36] A. Sidi, *Vector Extrapolation Methods with Applications*, SIAM, Philadelphia, 2017.

[37] A. Sidi, W.F. Ford, D.A. Smith, Acceleration of convergence of vector sequences, SIAM J. Numer. Anal., 23 (1986) 178-196.

[38] A. Sidi, D. Levin, Rational approximations from the d-transformation, IMA J. Numer. Anal., 2 (1982) 153–167.

[39] A. Sidi, D. Levin, Prediction properties of the t-transformation, SIAM J. Numer. Anal., 20 (1983) 589-598.

[40] D. Vekemans, Algorithm for the E-prediction, J. Comput. Appl. Math., 85 (1997) 181-202.

[41] E.J. Weniger, Nonlinear sequence transformations for the acceleration of convergence and the summation of divergent series, Comput. Phys. Rep., 10 (1989) 189-371.

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

[43] P. Wynn, On a device for computing the $e_m(S_n)$ transformation, Math. Tables Aids Comput., 10 (1956) 91-96.