SHANKS SEQUENCE TRANSFORMATIONS AND ANDERSON ACCELERATION

CLAUDE BREZINSKI *, MICHELA REDIVO-ZAGLIA †, AND YOUSEF SAAD ‡

Abstract. This paper presents a general framework for Shanks transformations of sequences of elements in a vector space. It is shown that the Minimal Polynomial Extrapolation (MPE), the Modified Minimal Polynomial Extrapolation (MMPE), the Reduced Rank Extrapolation (RRE), the Vector Epsilon Algorithm (VEA), the Topological Epsilon Algorithm (TEA), and Anderson Acceleration (AA), which are standard general techniques designed for accelerating arbitrary sequences and/or solving nonlinear equations, all fall into this framework. Their properties and their connections with quasi-Newton and Broyden methods are studied. The paper then exploits this framework to compare these methods. In the linear case, it is known that AA and GMRES are 'essentially' equivalent in a certain sense while GMRES and RRE are mathematically equivalent. This paper discusses the connection between AA, the RRE, the MPE, and other methods in the nonlinear case.

Key words. Acceleration techniques; sequence transformations; Anderson Acceleration; Reduced Rank Extrapolation; quasi-Newton methods; Broyden metods.

AMS subject classifications: 65B05, 65B99, 65F10, 65H10.

1. Introduction. In computational sciences it is often necessary to obtain the limit of a sequence of objects of a vector space (scalars, vectors, matrices, ...) that converges slowly to its limit or even diverges. In some situations, we may be able to obtain a new sequence that converges faster to the same limit by modifying the method that produced the original sequence. However, in many instances, the process by which the sequence is produced is hidden (*black box*) or too cumbersome for this approach to be practical. Another common solution is to transform this sequence, by means of a *sequence transformation*, into a new sequence which, under some assumptions, will converge faster. Notable among these general techniques is Shanks transformation [75]. As stated in [44], the so-called *Shanks transformation* is arguably the best all-purpose method for accelerating convergence of sequences.

The aim of this paper is to present a general framework for Shanks transformation(s) of sequences of elements in a vector space. This framework includes the Minimal Polynomial Extrapolation (MPE), the Modified Minimal Polynomial Extrapolation (MMPE), the Reduced Rank Extrapolation (RRE), the Vector Epsilon Algorithm (VEA), the Topological Epsilon Algorithm (TEA), and Anderson Acceleration (AA). Their application to the solution of systems of linear and nonlinear equations will be discussed throughout the paper. For details on these methods, which are widely used, and their many applications, see, for example, [7, 21, 34, 35, 40, 52, 70, 77].

Section 2, provides a basic background on sequence transformations for accelerating convergence. The general framework containing all the methods mentioned above is presented in Section 3. Their properties and their connections with quasi-Newton methods are also studied. In Section 4, some of the transformations are discussed in more details, in particular Anderson Acceleration which is related to Broyden-type methods. The Vector Epsilon Algorithm is treated in Section 5. Conclusions are drawn in Section 6.

*Université de Lille, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France. E-mail: Claude.Brezinski@univ-lille1.fr.

†Università degli Studi di Padova, Dipartimento di Matematica "Tullio Levi-Civita", Via Trieste 63, 35121-Padova, Italy. E-mail: Michela.RedivoZaglia@unipd.it.

‡University of Minnesota, Dept. of Computer Science and Engineering. E-mail: saad@cs.umn.edu.

2. Prologue on acceleration methods. An *acceleration technique* takes a sequence (s_n) and produces an *accelerated* sequence, or a set of such sequences $(t_n^{(k)})$, indexed by k, that, hopefully, converges faster than the original sequence, see, e.g., [12, 21]. Note that the s_i's can be scalars, or vectors, or matrices, or tensors, or even other elements in general inner-product spaces.

For a historical perspective on acceleration and extrapolation methods, see the article [19]. The literature on acceleration schemes is rich and has a long history. Modern acceleration methods started with Richardson's *deferred approach to the limit* [68,69] followed a little later by Aitken's well-known method for computing zeros of polynomials [1]. In 1955, Shanks [75] defined a generalization of Aitken's procedure. However, his method was not too practical as it relied on ratios of determinants and numerical methods for evaluating these were complicated as well as unstable. Shortly thereafter, Wynn [84] discovered an elegant recursive algorithm to calculate these ratios. This discovery set a new dynamic in motion and many papers followed. Meanwhile, physicists were also developing their own acceleration techniques using a viewpoint akin to that of quasi-Newton methods [1], see [2, 65, 66]. These techniques include *Anderson Acceleration* (or *Anderson mixing*), and *Pulay mixing* also known as *Direct Inversion in the Iterative Subspace* (DIIS). These were widely studied and applied to the solution of various problems in numerical analysis and applied mathematics. The literature on these topics is quite broad and we only mention a few papers to show the variety of results obtained and problems treated [31, 34, 41, 48, 63, 70, 78, 81]. One can distinguish between two classes of methods among those just mentioned. In the traditional acceleration techniques, such as Aitken or Shanks method, a sequence to accelerate is available at the outset and the aim of the method is to produce a faster converging sequence from it. In contrast, in the second class of methods, which includes the quasi-Newton based methods, DIIS, and Anderson Acceleration, the sequence is generated by the method itself.

We now introduce general acceleration methods starting with *Aitken's* Δ^2 *process* [1]. We are given a scalar sequence (s_n) whose limit is $\lim_{n\to\infty} s_n = s$. Aitken's acceleration is based on the observation that it is possible to find this limit exactly in the special situation where consecutive iterates s_n, satisfy the relation

$$s_{n+1} - s - \lambda(s_n - s) = 0, \quad \forall n \tag{2.1}$$

where λ is a constant different from 1. The above relation is the *kernel* of Aitken's process, that is the set of sequences which are transformed into a constant sequence whose terms are all equal to s. The scalar λ, and the limit s can be easily determined from s_n, s_{n+1}, s_{n+2} by writing:

$$\frac{s_{n+1} - s}{s_n - s} = \lambda, \quad \frac{s_{n+2} - s}{s_{n+1} - s} = \lambda \quad \rightarrow \quad \lambda = \frac{s_{n+2} - s_{n+1}}{s_{n+1} - s_n}$$

and, letting $\Delta s_i = s_{i+1} - s_i$ and $\Delta^2 s_i = \Delta s_{i+1} - \Delta s_i = s_{i+2} - 2s_{i+1} + s_i$, we obtain

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

[1] Here we use the term quasi-Newton method in a broad way to describe a method in which a first order derivative, as represented by a Jacobian, is approximated using current secant information.

which can also be written as a ratio of determinants

$$s = \frac{\begin{vmatrix} s_n & s_{n+1} \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta s_n & \Delta s_{n+1} \end{vmatrix}} = \frac{\begin{vmatrix} s_n & \Delta s_n \\ \Delta s_n & \Delta^2 s_n \end{vmatrix}}{\Delta^2 s_n} = s_n - \Delta s_n (\Delta^2 s_n)^{-1} \Delta s_n. \tag{2.2}$$

Although a trivial observation in this case, the third part of the above formula shows that s is the Schur complement of $\Delta^2 s_n$ in the matrix

$$\begin{pmatrix} s_n & \Delta s_n \\ \Delta s_n & \Delta^2 s_n \end{pmatrix},$$

while the second formula is Schur's determinantal formula for the complement. As a background recall that if a square matrix M is partitioned as

$$M = \begin{pmatrix} A & B \\ C & D \end{pmatrix}, \tag{2.3}$$

where D is square and invertible then $\det(M) = \det(D) \times \det(M/D)$ where (M/D) is the Schur complement of D in M, i.e., $(M/D) = A - BD^{-1}C$. Note that A can be a 1×1 matrix as was the case above. More on Schur complements and Schur determinantal formulas, can be found in [15, 62, 86].

Let now (s_n) be a sequence that does not belong to the kernel defined by (2.1). Any of the previous formulas for s can still be used, and its result is denoted by $t_n^{(1)}$. In particular, $t_n^{(1)} = s_n - \Delta s_n (\Delta^2 s_n)^{-1} \Delta s_n$. The *sequence transformation* $(s_n) \longmapsto (t_n^{(1)})$ defines Aitken's Δ^2 process and, by construction, $\forall n, t_n^{(1)} = s$ if and only if (s_n) satisfies (2.1). This kernel can also be written under the form

$$\alpha_0(s_n - s) + \alpha_1(s_{n+1} - s) = 0, \quad \forall n$$

where α_0, α_1 are constants such that $\alpha_0 \alpha_1 \neq 0$ and $\alpha_0 + \alpha_1 \neq 0$.

Shanks [75] extended the above idea by developing a transformation that yields the exact limit for sequences that belong to a $(k+1)$-term kernel, i.e., for sequences that satisfy:

$$\alpha_0(s_n - s) + \alpha_1(s_{n+1} - s) + \cdots + \alpha_k(s_{n+k} - s) = 0, \quad \forall n. \tag{2.4}$$

We now consider $\alpha_0, \ldots, \alpha_k$ and s as unknowns with $\alpha_0 \alpha_k \neq 0$ and $\alpha_0 + \cdots + \alpha_k \neq 0$. Since the α_i's are determined up to a multiplicative scalar, we will impose the following *normalization condition*, a constraint that does not restrict generality

$$\alpha_0 + \alpha_1 + \cdots + \alpha_k = 1. \tag{2.5}$$

From (2.4) and (2.5) we easily obtain the following linear system:

$$\begin{cases} \alpha_0 + \cdots + \alpha_k = 1 \\ s_{n+i}\alpha_0 + \cdots + s_{n+k+i}\alpha_k - s = 0, \quad i = 0, \ldots, k. \end{cases}$$

This is a $(k+2) \times (k+2)$ linear system with unknowns $\alpha_0, \cdots, \alpha_k, s$. The unknown s, which is the desired limit, can be obtained by using Cramer's rule. This process can now be applied to any sequence, not just one that satisfies the kernel relation (2.4), and

in this case we denote the resulting s by $t_n^{(k)}$. This process which transforms an original sequence (s_n) into the new sequence $(t_n^{(k)})$ is known as the *Shanks transformation.* A few row manipulations with determinants will lead to the following expression:

$$t_n^{(k)} = \frac{\begin{vmatrix} s_n & s_{n+1} & \cdots & s_{n+k} \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ \vdots & \vdots & & \vdots \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 & \cdots & 1 \\ \Delta s_n & \Delta s_{n+1} & \cdots & \Delta s_{n+k} \\ \vdots & \vdots & & \vdots \\ \Delta s_{n+k-1} & \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} \end{vmatrix}}.$$

By construction, $t_n^{(k)}$ is such that $\forall n, t_n^{(k)} = s$ if and only if (s_n) satisfies (2.4). Clearly, when $k = 1$ this is just Aitken's process as shown by (2.2).

The above formula can again be expressed using Schur complements. A remarkable result due to Wynn [84] is that, for scalar sequences, $t_n^{(k)}$ can be obtained by the following recursive implementation which he termed the ε-algorithm:

$$\varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + [\varepsilon_k^{(n+1)} - \varepsilon_k^{(n)}]^{-1},$$

with $\varepsilon_{-1}^{(n)} = 0$ and $\varepsilon_0^{(n)} = s_n$ for $n = 0, 1, \ldots$. As it turns out, we have $\varepsilon_{2k}^{(n)} = t_n^{(k)}$ for all k and n. Wynn extended this algorithm to vector sequences by defining the inverse of a vector $v \in \mathbb{C}^p$ as its pseudo-inverse, that is $v^{-1} = (v^*v)^{-1}v^*$. He thus obtained the *vector ε-algorithm* (VEA) [85] that will be discussed in Section 5. However, Shanks transformation does not extend as is to vector sequences. The more general framework of projection will have to be used for this purpose. This is explained next.

3. Shanks transformations in a vector space. Let (s_n) be a sequence of elements of a vector space E on $\mathbb{R}$ or $\mathbb{C}$ satisfying, for a fixed value of k and for all n, the following relation which generalizes (2.4)

$$\alpha_0(s_n - s) + \cdots + \alpha_k(s_{n+k} - s) = 0, \tag{3.1}$$

with $\alpha_i \in \mathbb{R}$, $s \in E$, and $\alpha_0 + \cdots + \alpha_k = 1$, a *normalization condition* which does not restrict generality. The set of such sequences is called the *Shanks kernel.*

For a fixed value of k, we want to transform (s_n) into a new sequence $(t_n^{(k)})$ such that, for sequences belonging to the Shanks kernel, $t_n^{(k)} = s, \forall n$ (now only a sufficient condition). If the coefficients α_i are known it immediately follows, from (3.1) and the normalization condition, that this *Shanks sequence transformation* is given by

$$t_n^{(k)} = \alpha_0 s_n + \cdots + \alpha_k s_{n+k}. \tag{3.2}$$

To determine the $k + 1$ coefficients α_i we will need to set-up a linear system of k (scalar) equations, in addition to the normalization condition. If the sequence to be transformed does not belong to the Shanks kernel, the coefficients α_i can still be computed by the same system but they will then depend on k and n and the transformed sequence will satisfy (3.2).

We will now present a general framework including all sequence transformations whose kernel is the set of sequences satisfying (3.1). Let us mention that this kernel

includes sequences which behave like sums of exponential functions (see [20]), a common feature of many iterative procedures, which explains their efficiency in a number of cases.

The main ingredients for building these schemes are the notions of *Schur complement* and *Schur determinantal formula* [15, 62, 86]. They were extended to matrices M of the form (2.3) where now $A \in E$, B is a row consisting of q elements of E, C is a vector of dimension q, and D a square and invertible $q \times q$ matrix. In this case $\det(M)$ is the element of E obtained by expanding M with respect to its first row of elements of E by the classical rules, and $(M/D) \in E$ [14]. In what follows, Δ is the usual forward difference operator, its powers defined as usual, and it always acts on the lower index when applied to quantities with two indices. When discussing the vector case, we always restrict ourselves to $\mathbb{R}^p$. There is no difficulty in extending the results to $\mathbb{C}^p$.

3.1. Coupled topological Shanks transformations. Let (t_n) be a known sequence of elements of E, called the *coupled sequence*, assumed to satisfy

$$\alpha_0 t_n + \cdots + \alpha_k t_{n+k} = 0, \tag{3.3}$$

for all n, where the coefficients α_i are the same as in (3.1). The corresponding Shanks sequence transformation is called a *Coupled Topological Shanks Transformation* (CTST). The term *topological* is due to historical developments of the transformation [11], and from the fact that, to be able to discuss its convergence properties, the vector space E must be equipped with a topology.

Let y and y_i, $i = 1, \ldots, k$, be linearly independent linear functionals (that is elements of E^*, the algebraic dual space of E) which can depend on n. Obviously, when E is a vector space of dimension p, we must have $k \leq p$. We denote by $\langle \cdot, \cdot \rangle$ the duality product (or bracket) between E^* and E.

Three strategies for writing a linear system that yields the coefficients α_i can be employed and these are discussed in turn.

3.1.1. The polynomial extrapolation strategy. This strategy is obtained from considering the system of linear equations

$$\begin{cases} \alpha_0 & + \cdots + & \alpha_k & = 1 \\ \alpha_0 \langle y_i, t_n \rangle & + \cdots + & \alpha_k \langle y_i, t_{n+k} \rangle & = 0, \quad i = 1, \ldots, k. \end{cases} \tag{3.4}$$

Invoking again Cramer's rule to solve this system, and substituting the resulting α_i's in (3.2) leads to

$$t_n^{(k)} = \begin{vmatrix} s_n & \cdots & s_{n+k} \\ \langle y_1, t_n \rangle & \cdots & \langle y_1, t_{n+k} \rangle \\ \vdots & & \vdots \\ \langle y_k, t_n \rangle & \cdots & \langle y_k, t_{n+k} \rangle \end{vmatrix} \Bigg/ \begin{vmatrix} 1 & \cdots & 1 \\ \langle y_1, t_n \rangle & \cdots & \langle y_1, t_{n+k} \rangle \\ \vdots & & \vdots \\ \langle y_k, t_n \rangle & \cdots & \langle y_k, t_{n+k} \rangle \end{vmatrix}, \tag{3.5}$$

where, as explained above, the determinant in the numerator represents the element of E obtained by developing it with respect to its first row by the usual rules for expanding determinants.

We now replace each of the columns from column $k+1$ down to column 2 by its difference with the preceding column, and we do this both in the numerator and the

denominator of (3.5). This transforms this ratio of determinants into:

$$t_n^{(k)} = \frac{\begin{vmatrix} s_n & \Delta s_n & \cdots & \Delta s_{n+k-1} \\ \langle y_1, t_n\rangle & \langle y_1, \Delta t_n\rangle & \cdots & \langle y_1, \Delta t_{n+k-1}\rangle \\ \vdots & \vdots & & \vdots \\ \langle y_k, t_n\rangle & \langle y_k, \Delta t_n\rangle & \cdots & \langle y_k, \Delta t_{n+k-1}\rangle \end{vmatrix}}{\begin{vmatrix} \langle y_1, \Delta t_n\rangle & \cdots & \langle y_1, \Delta t_{n+k-1}\rangle \\ \vdots & & \vdots \\ \langle y_k, \Delta t_n\rangle & \cdots & \langle y_k, \Delta t_{n+k-1}\rangle \end{vmatrix}}. \tag{3.6}$$

Thus, according to the Schur determinantal formula, $t_n^{(k)}$ can be written as a Schur complement

$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}](Y^T \Delta T_n^{(k)})^{-1} Y^T T_{n,1}^{(k)}, \tag{3.7}$$

with $Y = [y_1, \ldots, y_k]$, $T_n^{(k)} = [t_n, \ldots, t_{n+k-1}]$, and where $T_{n,1}^{(k)}$ is the first column of the matrix $T_n^{(k)}$ (that is t_n in this case). Note that in this notation the matrix $T_n^{(k)}$ has k columns (denoted by an upper index) and that its first column is t_n (which has n as a lower index). An important point to notice is that, in a general vector space E, the notations $(Y^T \Delta T_n^{(k)})$ and $Y^T T_{n,1}^{(k)}$ have to be understood in the sense of the duality product and not in the sense of the usual scalar product between vectors. This means, for example, that $(Y^T \Delta T_n^{(k)})$ is the matrix whose elements are $\langle y_i, \Delta t_{n+j-1}\rangle$ for $i, j = 1, \ldots, k$. Obviously, it has the original meaning when $E = \mathbb{R}^p$. In the matrix case, the duality product becomes the Frobenius inner product defined, for $P, Q \in \mathbb{R}^{p\times q}$, by $\langle P, Q\rangle = \langle P, Q\rangle_F = \mathrm{tr}(P^T Q) = \mathrm{tr}(Q^T P)$. These notational conventions will also be valid below. It is also worthwhile noting that when $E = \mathbb{R}^p$ and $k = p$, this formula simplifies to

$$t_n^{(p)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+p-1}](\Delta T_n^{(p)})^{-1} T_{n,1}^{(p)}.$$

This transformation enters into the framework introduced in [22].

We saw that (3.6) is deduced from (3.5) by replacing each column in the numerator and in the denominator from the last one by its difference with the preceding one. The same treatment can be reapplied several times to (3.6), thus leading to

$$t_n^{(k)} = \begin{vmatrix} s_n & \Delta s_n & \cdots & \Delta^k s_n \\ \langle y_1, t_n\rangle & \langle y_1, \Delta t_n\rangle & \cdots & \langle y_1, \Delta^k t_n\rangle \\ \vdots & \vdots & & \vdots \\ \langle y_k, t_n\rangle & \langle y_k, \Delta t_n\rangle & \cdots & \langle y_k, \Delta^k t_n\rangle \end{vmatrix} \Big/ \begin{vmatrix} \langle y_1, \Delta t_n\rangle & \cdots & \langle y_1, \Delta^k t_n\rangle \\ \vdots & & \vdots \\ \langle y_k, \Delta t_n\rangle & \cdots & \langle y_k, \Delta^k t_n\rangle \end{vmatrix} \tag{3.8}$$

$$= s_n - [\Delta s_n, \ldots, \Delta^k s_n][Y^T[\Delta t_n, \ldots, \Delta^k t_n]]^{-1} Y^T T_{n,1}^{(k)}. \tag{3.9}$$

3.1.2. The Shanks strategy. We will now outline the strategy followed by Shanks to obtain his scalar sequence transformation. Shanks considered extracting the α_i's by solving the system of linear equations:

$$\begin{cases} \alpha_0 & + \cdots + & \alpha_k & = 1 \\ \alpha_0\langle y, t_{n+i}\rangle & + \cdots + & \alpha_k\langle y, t_{n+k+i}\rangle & = 0, \quad i = 0, \ldots, k-1. \end{cases} \tag{3.10}$$

where y is now a fixed vector. Proceeding as before, we solve the system with Cramer's rule and then exploit (3.2) to obtain

$$t_n^{(k)} = \begin{vmatrix} s_n & \cdots & s_{n+k} \\ \langle y, t_n\rangle & \cdots & \langle y, t_{n+k}\rangle \\ \vdots & & \vdots \\ \langle y, t_{n+k-1}\rangle & \cdots & \langle y, t_{n+2k-1}\rangle \end{vmatrix} \Big/ \begin{vmatrix} 1 & \cdots & 1 \\ \langle y, t_n\rangle & \cdots & \langle y, t_{n+k}\rangle \\ \vdots & & \vdots \\ \langle y, t_{n+k-1}\rangle & \cdots & \langle y, t_{n+2k-1}\rangle \end{vmatrix}. \tag{3.11}$$

Replacing each column starting from the last one by its difference with the preceding one allows to write this ratio of determinants as

$$t_n^{(k)} = \frac{\begin{vmatrix} s_n & \Delta s_n & \cdots & \Delta s_{n+k-1} \\ \langle y, t_n\rangle & \langle y, \Delta t_n\rangle & \cdots & \langle y, \Delta t_{n+k-1}\rangle \\ \vdots & \vdots & & \vdots \\ \langle y, t_{n+k-1}\rangle & \langle y, \Delta t_{n+k-1}\rangle & \cdots & \langle y, \Delta t_{n+2k-2}\rangle \end{vmatrix}}{\begin{vmatrix} \langle y, \Delta t_n\rangle & \cdots & \langle y, \Delta t_{n+k-1}\rangle \\ \vdots & & \vdots \\ \langle y, \Delta t_{n+k-1}\rangle & \cdots & \langle y, \Delta t_{n+2k-2}\rangle \end{vmatrix}}. \tag{3.12}$$

Thus, according to the Schur determinantal formula, $t_n^{(k)}$ can be written as a Schur complement (a new result)

$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}](Y^T \Delta T_n^{(k)})^{-1} Y^T T_{n,1}^{(k)}, \tag{3.13}$$

with now

$$Y = \begin{pmatrix} y & z & \cdots & z \\ z & y & \cdots & z \\ \vdots & \vdots & & \vdots \\ z & z & \cdots & y \end{pmatrix} \quad \text{and} \quad T_n^{(k)} = \begin{pmatrix} t_n & t_{n+1} & \cdots & t_{n+k-1} \\ t_{n+1} & t_{n+2} & \cdots & t_{n+k} \\ \vdots & \vdots & & \vdots \\ t_{n+k-1} & t_{n+k} & \cdots & t_{n+2k-2} \end{pmatrix},$$

where $z = 0 \in E^*$, and where $T_{n,1}^{(k)}$ denotes the first column of the matrix $T_n^{(k)}$ as before. In the particular case $t_n = \Delta s_n$, such a formula was already given in [18].

Here are a few observations. A first observation, valid for the case when $E = \mathbb{R}^p$, is based on the fact that $T_{n,1}^{(k)} = T_n^{(k)} e_1$ where e_i is the i-th canonical basis vector of the range of $T_n^{(k)}$, i.e., it consists of zeros except for a one in the i-th entry. For any j, where $0 \le j \le k-1$ we write

$$\begin{aligned} T_{n,1}^{(k)} &= T_n^{(k)}(e_1 - e_2) + T_n^{(k)}(e_2 - e_3) + \cdots + T_n^{(k)}(e_j - e_{j+1}) + T_n^{(k)} e_{j+1} \\ &= T_n^{(k)} e_{j+1} - \sum_{i=1}^{j} \Delta T_n^{(k)} e_i. \end{aligned}$$

When substituted into (3.13) this immediately yields the alternative formula, valid for any j with $0 \le j \le k-1$

$$t_n^{(k)} = s_{n+j} - [\Delta s_n, \ldots, \Delta s_{n+k-1}](Y^T \Delta T_n^{(k)})^{-1} Y^T T_{n+j,1}^{(k)},$$

where $T_{n+j,1}^{(k)} \equiv T_n^{(k)} e_{j+1}$ is the $(j+1)$-st column of $T_n^{(k)}$, or, equivalently, the 1-st column of $T_{n+j}^{(k)}$. A more general result will be proved later (Theorem 3.1).

A second observation will lead to yet another formula for $t_n^{(k)}$, namely one that expresses formulae (3.5) and (3.11) as the Schur complements

$$t_n^{(k)} = \alpha_0(s_n - [s_{n+1}, \ldots, s_{n+k}](Y^T T_{n+1}^{(k)})^{-1} Y^T T_{n,1}^{(k)}),$$

with, for each case, the corresponding matrices Y and $T_n^{(k)}$. This result is easily obtained by dividing their respective numerators and denominators by the determinant of the matrix $Y^T T_{n+1}^{(k)}$. Thus, the numerators of (3.5) and (3.11) can be written as $s_n - [s_{n+1}, \ldots, s_{n+k}](Y^T T_{n+1}^{(k)})^{-1} Y^T T_{n,1}^{(k)}$, and their denominators as $1/\alpha_0 = 1 - [1, \ldots, 1](Y^T T_{n+1}^{(k)})^{-1} Y^T T_{n,1}^{(k)}$, which gives the result.

A last observation is that (3.12) and (3.13) can also be written under a form similar to (3.8) and (3.9).

Finally, we note that the matrix $[\Delta s_n, \ldots, \Delta s_{n+k-1}](Y^T \Delta T_n^{(k)})^{-1} Y^T$ is a projector only when $t_n = s_n, \forall n$, a choice that may not satisfy (3.3).

3.1.3. The least-squares strategy. To discuss the least-squares strategy we begin by expressing the formulas (3.2) and (3.3) in an alternative form that will invoke the differences Δs_{n+j}, for $j = 0, \ldots, k-1$. These definitions for $t_n^{(k)}$ can also be written as follows

$$\begin{aligned} t_n^{(k)} &= s_n + \sum_{j=1}^{k} \alpha_j (s_{n+j} - s_n) \\ &= s_n + \sum_{j=1}^{k} \alpha_j \sum_{i=1}^{j} \Delta s_{n+i-1} \\ &= s_n - \sum_{i=1}^{k} \left(- \sum_{j=i}^{k} \alpha_i \right) \Delta s_{n+i-1}. \end{aligned}$$

In other words, the accelerated sequence will satisfy:

$$t_n^{(k)} = s_n - \sum_{i=1}^{k} \beta_i \Delta s_{n+i-1}, \tag{3.14}$$

in which $\beta_i = -(\alpha_i + \cdots + \alpha_k)$ for $i = 1, \ldots, k$. Note that since the constraint (2.5) has been used to derive (3.14) this new formulation implicitly assumes that the α_i's sum up to one. Proceeding similarly for the sequence t_n, we would obtain the relation

$$t_n - \sum_{i=1}^{k} \beta_i \Delta t_{n+i-1} = 0. \tag{3.15}$$

In the least-squares strategy, the vector $b = (\beta_1, \ldots, \beta_k) \in \mathbb{R}^k$ is obtained by solving the $(p+1) \times k$ least-squares system (3.15), that is

$$[\Delta t_n, \ldots, \Delta t_{n+k-1}]\ b \ =_{\text{LS}}\ t_n, \tag{3.16}$$

where $=_{\text{LS}}$ stands for 'equal in the least-squares sense'. Thus, using the same notation $T_n^{(k)}$ as in the polynomial extrapolation strategy, and assuming that $T_n^{(k)}$ is of full

rank, we get $b = [(\Delta T_n^{(k)})^T \Delta T_n^{(k)}]^{-1} (\Delta T_n^{(k)})^T t_n$. It then follows that the sequence transformation (3.14) is given by

$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}][(\Delta T_n^{(k)})^T \Delta T_n^{(k)}]^{-1} (\Delta T_n^{(k)})^T t_n. \tag{3.17}$$

Since $t_n = T_{n,1}^{(k)}$, this formula is a particular case of (3.7) with, now, $Y = \Delta T_n^{(k)}$. By the Schur determinantal formula, we also have

$$t_n^{(k)} = \begin{vmatrix} s_n & \Delta s_n \cdots \Delta s_{n+k-1} \\ (\Delta T_n^{(k)})^T t_n & (\Delta T_n^{(k)})^T \Delta T_n^{(k)} \end{vmatrix} \Big/ \left| (\Delta T_n^{(k)})^T \Delta T_n^{(k)} \right|, \tag{3.18}$$

which is a particular case of (3.6) with $Y = \Delta T_n^{(k)}$.

As before, the matrix $[\Delta s_n, \ldots, \Delta s_{n+k-1}][(\Delta T_n^{(k)})^T \Delta T_n^{(k)}]^{-1} (\Delta T_n^{(k)})^T$ in (3.17) is a projector only when $t_n = s_n$, $\forall n$, and this choice of t_n may not satisfy (3.3). We also remark that formula (3.17) shows that $t_n^{(k)}$ is the pseudo-Schur complement of $\Delta T_n^{(k)}$ in the matrix [67]

$$\begin{pmatrix} s_n & \Delta S_n^{(k)} \\ t_n & \Delta T_n^{(k)} \end{pmatrix} \quad \text{with} \quad S_n^{(k)} = [s_n, \ldots, s_{n+k-1}].$$

Notice that (3.18) can also be written under a form similar to (3.8).

3.1.4. Choice of the coupled sequence. We will now discuss the choice of the coupled sequence (t_n). There are two common ways of selecting it.

General choice. Writing (3.1) for the indices $n+1$ and n, and subtracting, we see that the sequence $t_n = \Delta s_n$ satisfies (3.3). In fact, any sequence of the form $t_n = \Delta^p s_n$, $p \geq 2$, will also satisfy (3.3) and is therefore a valid choice. It will lead to a transformation proposed in [43, p. 68].

Fixed-point choice. Consider the fixed point problem $s = g(s)$ in E, and assume that the s_n's are given by $s_{n+1} = g(s_n), n = 0, 1, \ldots$ Then when the s_n's satisfy (3.1), the $g(s_n)$'s will also satisfy it, as well as their differences. Thus, we can select $t_n = g(s_n) - s_n$, which leads to variants of MPE [26], MMPE [11, 64], RRE [36, 58], and TEA [11], in the appropriate vector space E. Other possible interesting choices include $t_n = \Delta^p s_{n+m}$ or $t_n = g(s_{n+m}) - s_{n+m}$ where $m \in \mathbb{Z}$. Setting $f(s) = g(s) - s$, this also motivates the choice $t_n = f(s_n)$ where the s_n's are approximations of s.

3.1.5. Choice of the linear functionals. Next we discuss the choice of the linear functionals y and y_i in the cases of the polynomial extrapolation and the Shanks strategies (these functionals do not play a role in the least squares strategy). These functionals may or may not depend on n, thus leading to new transformations which have not yet been studied. When E is $\mathbb{R}^p$, the duality product becomes the usual inner product. In the matrix case, the duality product is replaced by the Frobenius inner product as explained above.

3.2. Summary and nomenclature. In this section, we summarize the various transformations derived from the kernels (3.1) and (3.3), and the corresponding names by which they will be called.

The sequence transformation defined by (3.2) will be denoted by the generic term *Coupled Topological Shanks Transformation* (CTST in short). Each method depends on two selections. First we select one of three possible strategies for writing the linear

system that yields the coefficients α_i. These are the polynomial extrapolation strategy (3.4), the Shanks strategy (3.10) and the least squares-strategy (3.16). The three symbols used for these strategies will be: *Pol*, *Sha*, and *Lsq*, respectively. Second, we have two possibilities for choosing the coupled sequence (t_n) satisfying (3.3): the general choice and the fixed point choice. We will use the symbols *Gen* and *Fxp* for these, respectively. Thus, we end-up with six classes of transformations according to the strategy for the computation of the coefficients α_i, and the choice of the coupled sequence (t_n). The naming for these methods will consist of the acceleration strategy selected followed by the choice made for the coupling sequence, e.g., *Pol-Gen* , for Polynomial acceleration scheme, with the general choice for the coupling. These methods are shown in the following table where the columns determine the accelerations strategy (Polynomial, Shanks, Least-squares) while the rows determine the choice of the coupling sequence t_n (general, fixed point).

	Polynomial	Shanks	Least-squares
t_n : General	*Pol-Gen*	*Sha-Gen*	*Lsq-Gen*
t_n : Fixed Point	*Pol-Fxp*	*Sha-Fxp*	*Lsq-Fxp*

It must be made clear that, even when $E = \mathbb{R}^p$, the choices of the sequence (t_n) and that of y and the y_i's are independent of each other.

We set $S_n^{(k)} = [\Delta s_n, \ldots, \Delta s_{n+k-1}]$. We will now study, in particular, the following methods

- The *Modified Mimimal Polynomial Extrapolation* (MMPE) [11, 64]. It enters into the polynomial extrapolation strategy when the y_i's are arbitrarily fixed linearly independent linear functionals and $t_n = \Delta s_n$. It is given by
$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}](Y^T \Delta^2 S_n^{(k)})^{-1} Y^T \Delta s_n, \quad Y = [y_1, \ldots, y_k].$$
- The *Minimal Polynomial Extrapolation* (MPE) [26] corresponds to the polynomial extrapolation strategy with $t_n = \Delta s_n$ and $y_i = \Delta s_{n+i-1}$, and we have
$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}][(\Delta S_n^{(k)})^T \Delta^2 S_n^{(k)}]^{-1} (\Delta S_n^{(k)})^T \Delta s_n.$$
- The *Reduced Rank Extrapolation* (RRE) [36, 58] is obtained by the choices $t_n = \Delta s_n$ and $y_i = \Delta^2 s_{n+i-1}$. It holds
$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}][(\Delta^2 S_n^{(k)})^T \Delta^2 S_n^{(k)}]^{-1} (\Delta^2 S_n^{(k)})^T \Delta s_n.$$
- *Anderson Acceleration* (AA) [2] is a method for the solution of fixed point problems. Modulo a shift of indices, the vectors $\bar{x}_k$ that it constructs can be seen to belong to the class *Pol-Fxp* and the vectors $\bar{f}_k$ and $\bar{g}_k$ have the form (3.17) from the least-squares strategy *Lsq-Fxp* (see Formulas (4.6), (4.8) and (4.10) of Section 4.3).
- The *Topological Epsilon Algorithm* (TEA) [11]. Its first version falls into the Shanks strategy with a fixed $y \in E^*$, and it is given by (3.11) or (3.12) or (3.13).
- The *Vector Epsilon algorithm* (VEA) [85], discussed in Section 5, also enters into this framework after replacing determinants by designants which generalize them in a noncommutative algebra [73].

The MMPE and the TEA can treat, without any change, sequences of elements of a general vector space, in particular, matrices or tensors, while, in the matrix case, the other transformations need the replacement of the duality product by the Frobenius inner product.

3.3. Recursive implementations. For all methods described above, when n is fixed and k increases, the linear systems (3.4) and (3.10) can be recursively solved by the bordering method described in [16] and [21, pp. 30-31]. Thus, the vector, matrix and tensor cases are treated in the same way.

Of these methods, only three benefit from a specific simple recursive algorithm for their implementation in the case where y and the y_i's are independent of n. These are the MMPE which can be implemented by the $S\beta$-algorithm of Jbilou [49] (see also [51]), the *Sha-Gen* by the TEAs [11] or, by the less expensive STEAs [23, 24], and Henrici's method [46, p. 115] by the H-algorithm [13, 25].

In the general case, that is when the y_i's depend on n, some other recursive algorithms also exist but their implementation is quite tricky [82, pp. 177] and [22].

3.4. Properties. We remark that, in all cases, formulae (3.7), (3.13) and (3.17) have the same structure, independently from the choice of the linear functionals y and y_i, namely

$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}]\gamma,$$

where γ is the solution of the system $(Y^T \Delta T_n^{(k)})\gamma = Y^T T_{n,1}^{(k)}$. The preceding result can be generalized by isolating any column i in the determinants of (3.5) and (3.11), and it leads to

THEOREM 3.1. *The following expression holds for any $i = 0, \ldots, k$,*

$$t_n^{(k)} = \begin{vmatrix} s_{n+i} & \Delta S_n^{(k)} \\ Y^T T_{n+i,1}^{(k)} & Y^T \Delta T_n^{(k)} \end{vmatrix} \Big/ \left| Y^T \Delta T_n^{(k)} \right|, \quad \textit{with} \quad \Delta S_n^{(k)} = [\Delta s_n, \ldots, \Delta s_{n+k-1}],$$

that is

$$t_n^{(k)} = s_{n+i} - \Delta S_n^{(k)} \gamma_i, \quad \textit{with} \quad \gamma_i = (Y^T \Delta T_n^{(k)})^{-1} Y^T T_{n+i,1}^{(k)}.$$

Proof. In (3.5) and (3.11), select any column i for $0 \le i \le k$. For $i = 0$, we have the formulae (3.7) and (3.13) given above. After selecting a column $1 \le i \le k-1$, we subtract the column j from the column $j+1$ for $j = 0, \ldots, i-1$. Then, for $j = i+1, \ldots, k$, we subtract the column j from the column $j-1$. When $i = k$, the subtractions are done only for the preceding columns. Finally, the column i is moved to the first place in both the numerator and the denominator. Since, the exact same operations are performed on the numerator and the denominator, the sign of the ratio is unchanged. The new ratio now appears as a Schur complement and the result follows. For the least-squares strategy, we first have to write (3.18) as a ratio of determinants, thus obtaining a formula similar to (3.6) and (3.12). Then, the determinants have to be modified by adding together their columns, and we get a representation like (3.5) and (3.11). Finally, we proceed with any column i as above for the two other strategies. □

COROLLARY 3.2. *Assume that the s_i's are vectors in $\mathbb{R}^p$ that are generated by the linear recurrence $s_{j+1} = Hs_j + d$, s_0 arbitrary, where and $I - H$ is invertible. Then for all three strategies of Section 3.1, with $t_j = \Delta s_j$ $\forall j$, we have $t_0^{(m)} = s = (I - H)^{-1} d$, where m is the degree of the minimal polynomial of H for the vector $s_0 - s$.*

This result is well-known and it has even been extended to some cases where the matrix H is singular. It is based on the fact that, thanks to the definition of

the minimal polynomial of a matrix for a vector, the s_j's and s satisfy (3.1). The complete results and their proofs can be found in the literature [10, 42].

The corollary means that any of the Shanks transformations will yield the exact solution in at most m steps, and this result is valid even if the original sequence (s_n) does not converge, i.e., without making any particular assumption on M.

The next property we prove is an orthogonality result that will establish a link with projection methods. From Theorem 3.1, we have

$$t_n^{(k)} = s_{n+i} - [\Delta s_n, \ldots, \Delta s_{n+k-1}]\gamma^{(i)}, \quad i = 0, \ldots, k,$$

where $\gamma^{(i)}$ is the solution of the system $(Y^T \Delta T_n^{(k)})\gamma^{(i)} = Y^T T_{n+i,1}^{(k)}$. We set

$$\widetilde{t}_n^{(k)} = s_{n+i+1} - [\Delta s_{n+1}, \ldots, \Delta s_{n+k}]\gamma^{(i)},$$

where $\gamma^{(i)}$ is the same as above. If $t_n = \Delta s_n$, we have

$$\begin{aligned} Y^T(\widetilde{t}_n^{(k)} - t_n^{(k)}) &= Y^T \Delta s_{n+i} - Y^T[\Delta^2 s_n, \ldots, \Delta^2 s_{n+k-1}]\gamma^{(i)} \\ &= Y^T \Delta s_{n+i} - Y^T \Delta T_n^{(k)} (Y^T \Delta T_n^{(k)})^{-1} Y^T T_{n+i,1}^{(k)}. \end{aligned}$$

Thus, we obtain the following Galerkin orthogonality conditions that generalize a property given in [53, eq. (2.4)] (see also [50]), and are valid for all coupled topological Shanks transformations

THEOREM 3.3. *We set*

$$\widetilde{t}_n^{(k)} = s_{n+i+1} - [\Delta s_{n+1}, \ldots, \Delta s_{n+k}]\gamma^{(i)}, \quad i = 0, \ldots, k-1,$$

where $\gamma^{(i)}$ *is the solution of the system* $(Y^T \Delta T_n^{(k)})\gamma^{(i)} = Y^T T_{n+i,1}^{(k)}$. *If* $\forall n, t_n = \Delta s_n$, *then*

$$Y^T(\widetilde{t}_n^{(k)} - t_n^{(k)}) = 0, \quad i = 0, \ldots, k-1.$$

3.5. The quasi-Newton connection. Consider a system of p nonlinear equations in p unknowns $f(x) = g(x) - x = 0 \in \mathbb{R}^p$. Newton's method consists in the iteration $x_{n+1} = x_n - [f'(x_n)]^{-1} f(x_n)$, where $f'(x)$ denotes the Jacobian of f at x.

Under the assumptions of Lipschitz continuity of the Jacobian f' in the neighborhood of x and the boundedness of its inverse, it is known that the sequence (x_n) converges locally to a solution and that the convergence is quadratic, see, e.g., [33], and [30] or [29, pp. 478 ff.] for a detailed study. The main drawback of Newton's method is the need to compute f' and so *quasi-Newton methods* were introduced as a remedy. They replace Newton's iteration by an iteration of the form

$$x_{n+1} = x_n - G_n f(x_n),$$

where G_n is an approximation of $[f'(x_n)]^{-1}$ (see, for example, [17, pp. 287ff]).

We consider the following iterative method for computing the fixed point x of g

1. Set $s_0 = x_n$.
2. Compute $s_{i+1} = g(s_i)$ for $i = 0, \ldots, k-1$.
3. Apply the transformation *Pol-Fxp* (that is $t_i = f_i = g(s_i) - s_i = \Delta s_i$) to the iterates s_i, and compute (3.7) for $n = 0$, that is

$$t_0^{(k)} = s_0 - [\Delta s_0, \ldots, \Delta s_{k-1}](Y^T \Delta T_0^{(k)})^{-1} Y^T T_{0,1}^{(k)}.$$

4. Set $x_{n+1} = t_0^{(k)}$.

Since $T_{0,1}^{(k)} = t_0 = f_0 = f(x_n)$, any of these methods can be considered as a quasi-Newton method with

$$G_n = [\Delta s_0, \ldots, \Delta s_{k-1}](Y^T \Delta T_0^{(k)})^{-1} Y^T \in \mathbb{R}^{p \times p}.$$

The Shanks strategy also leads to a fixed point method by computing $t_0^{(k)}$ by (3.13), and restarting the iterations with $x_{n+1} = t_0^{(k)}$ (a procedure first proposed for the vector ε-algorithm when $k = p$ [8, 9, 42]). Although more complicated (since it needs to compute the s_i's up to $i = 2k$), this method (which is *Sha-Gen* or *Sha-Fxp*) can also be considered as a quasi-Newton method where G_n is as above but with dimension $p \times kp$, and where $T_{0,1}^{(p)}$ is now the vector $(f(s_0)^T, \ldots, f(s_{k-1})^T)^T$ of dimension kp. We will come back to this procedure in Section 4.1.

Among quasi-Newton methods, the Barnes secant method [3] uses an approximation $G_n \in \mathbb{R}^{p \times p}$ that satisfies the conditions

$$G_n \Delta f_i = \Delta s_i, \quad i = 0, \ldots, p-1,$$

where $f_i = f(s_i) = g(s_i) - s_i$. Using the notations of the polynomial extrapolation strategy, this can be written in matrix form as,

$$\begin{aligned} G_n[\Delta f_0, \ldots, \Delta f_{p-1}] &= [\Delta s_0, \ldots, \Delta s_{p-1}] \\ G_n \Delta T_0^{(p)} &= \Delta S_0^{(p)}, \end{aligned}$$

with $t_i = f_i$ and $S_0^{(p)} = [s_0, \ldots, s_{p-1}]$. Thus $G_n = \Delta S_0^{(p)} (\Delta T_0^{(p)})^{-1}$, and the iteration becomes

$$x_{n+1} = x_n - [\Delta s_0, \ldots, \Delta s_{p-1}][\Delta f_0, \ldots, \Delta f_{p-1}]^{-1} f(x_n) = x_n - \Delta S_0^{(p)} (\Delta T_0^{(p)})^{-1} T_{0,1}^{(p)}.$$

As we will see in Section 4.2, this is exactly the RRE when $k = p$ since $t_i = f_i = \Delta s_i$.

As stated by Barnes [3], his method can be identified with the generalized secant method as previously described by Bittner [6] and Wolfe [83]. The matrix G_n is determined by the conditions

$$G_n f_i = s_i - x_{n+1}, \quad i = 0, \ldots, p,$$

which yields, in matrix form,

$$\begin{aligned} G_n[\Delta f_0, \ldots, \Delta f_p] &= [\Delta s_0, \ldots, \Delta s_p] \\ G_n \Delta T_0^{(p+1)} &= \Delta S_0^{(p+1)}. \end{aligned}$$

As explained in [27], since the $p+1$ vectors f_i must be linearly dependent, there exist constants α_i not all zero such that

$$\alpha_0 f_0 + \cdots + \alpha_p f_p = 0,$$

a relation identical to (3.3). The constants α_i can be normalized to sum up to 1. Multiplying the two preceding relations by G_n, which is assumed to exist, we get

$$\alpha_0 (s_0 - x_{n+1}) + \cdots + \alpha_p (s_p - x_{n+1}) = 0, \tag{3.19}$$

which gives $x_{n+1} = \alpha_0 s_0 + \cdots + \alpha_p s_p$. The vector $a = (\alpha_0, \ldots, \alpha_p)^T$ is obtained as the solution of the system of linear equations

$$\begin{pmatrix} 1 & \cdots & 1 \\ f_0 & \cdots & f_p \end{pmatrix} a = e_1 = (1, 0, \cdots, 0)^T \in \mathbb{R}^{p+1}.$$

It is easy to recognize that (3.19) is nothing else than the Shanks kernel (3.1) when starting from s_0 and with $k = p$, and that the procedure falls into the class *Lsq-Fxp* (see Section 3.1.3).

Under some assumptions, all these methods converge quadratically to the fixed point x of g when $k = p$. This is proved in [51] for the RRE and the MPE, in [56] for the TEA, in [8] and [9] for the VEA (although there is a gap in the proof), and in [61, p. 373] for the MMPE with the choice $y_i = e_i$ (which corresponds to a method due to Henrici [46, p. 115], see Section 4.1). As proved in Corollary 3.2, all methods presented in this paper yield the exact solution in one iteration for a system of linear equations when $k = p$, the dimension of the system. Indeed, it is known since the 1980s that RRE and MPE are Krylov subspace methods in the linear case [4, 5, 49] (see also [52, 76]). Analogously, the sequence (x_k) obtained by Lanczos' method [55] for solving the system of linear equations $Ax = (I - M)x = d$ starting from x_0 (which can be implemented by the biconjugate gradient algorithm of Fletcher [39]), and the sequence $(t_0^{(k)})$ obtained by applying the TEA with $y = r_0 = (I - M)x_0 - d$ to the sequence generated by $s_{n+1} = Ms_n + d$ with $s_0 = x_0$ are identical as proved in [12, pp. 186-189]. A simpler proof, already given in [17, pp. 167-8], is based on the fact that $\Delta s_n = -r_n = d - (I - M)x_n$ and $\Delta^i s_n = (-1)^i A^{i-1} r_n$, and on the determinantal expressions of x_k and $t_0^{(k)}$ (see (3.8)), or those of the corresponding Schur complements (see (3.9)), which can be shown to be identical after some algebraic manipulations.

REMARK 1. *The preceding results are still valid if p is replaced by m, where m is the degree of the minimal polynomial of the Jacobian $f'(x)$ for the vector $x_n - x$.*

4. Particular methods. We will now consider particular cases of our general framework that are well-known.

4.1. The Modified Minimal Polynomial Extrapolation. The Modified Minimal Polynomial Extrapolation (MMPE) belongs to the class *Pol-Gen* . It corresponds to the choice $t_n = \Delta s_n$, and linearly independent y_i's.

We now apply MMPE to the solution of the fixed point problem $s = g(s)$ in $\mathbb{R}^p$, and consider the vectors generated by $s_i = g(s_{i-1})$ for $i = 1, 2, \ldots$, with s_0 given. Taking $k = p$ and choosing $y_i = e_i$ (the vectors of the canonical basis of $\mathbb{R}^p$), the first application of the MMPE produces the vector

$$t_0^{(p)} = s_0 - [\Delta s_0, \ldots, \Delta s_{p-1}][\Delta^2 s_0, \ldots, \Delta^2 s_{p-1}]^{-1} \Delta s_0,$$

which can be written as

$$t_0^{(p)} = s_0 - [\Delta s_0, \ldots, \Delta s_{p-1}]\gamma,$$

where γ is the solution of the system

$$[\Delta^2 s_0, \ldots, \Delta^2 s_{p-1}]\gamma = \Delta s_0 \quad \text{that is} \quad \Delta T_0^{(p)} \gamma = \Delta s_0.$$

As mentioned in Section 3.5, for finding the fixed point $s = g(s)$, we consider the iterative method which consists in constructing a sequence (x_n) by setting $s_0 = x_n$,

applying the MMPE as above, defining the next iterate by $x_{n+1} = t_0^{(p)}$, and restarting the process with $s_0 = x_{n+1}$. This method is due to Henrici [46, p. 115] and, under some assumptions, the sequence (x_n) converges quadratically to the fixed point s of g. If g is affine, then $t_0^{(p)} = x_1 = s$. As mentioned in Section 3.5, a similar restarting procedure with the other methods described above leads to methods that, under some assumptions, converge quadratically to the fixed point of g.

Assume now that the vectors s_i are not given by fixed point iterations (they need not even be given a priori but may be generated by the transformation process itself) and that, instead of taking $k = p$ in the system that gives γ, we take $k \leq p$. Then, this system does not have a full rank. Solving it in the least squares sense gives $(\Delta T_0^{(k)})^T \Delta T_0^{(k)} \gamma = (\Delta T_0^{(k)})^T \Delta s_0$, and $t_0^{(k)}$ is nothing else than the first application of the RRE which was discovered in this way [36, 58]. Notice that Formula (3.17) is also recovered for $n = 0$.

4.2. The Reduced Rank Extrapolation. As previously mentioned, the Reduced Rank Extrapolation (RRE) corresponds to setting $y_i = \Delta^2 s_{n+i-1}$ for $i = 1, \ldots, k$ and $t_{n+i} = \Delta s_{n+i}$ for $i = 0, \ldots, k-1$ in the polynomial extrapolation strategy. Therefore, it is a member of the class *Pol-Gen* . Since $Y = \Delta T_n^{(k)} = [\Delta^2 s_n, \ldots, \Delta^2 s_{n+k-1}]$, it follows that

$$t_n^{(k)} = s_n - [\Delta s_n, \ldots, \Delta s_{n+k-1}][(\Delta T_n^{(k)})^T \Delta T_n^{(k)}]^{-1} (\Delta T_n^{(k)})^T \Delta s_n.$$

Using the notation of Theorem 3.1, the vector $\gamma^{(0)} = [(\Delta T_n^{(k)})^T \Delta T_n^{(k)}]^{-1} (\Delta T_n^{(k)})^T \Delta s_n$ is such that

$$\gamma^{(0)} = \text{argmin}_\gamma \|\Delta s_n - \Delta T_n^{(k)} \gamma\|_2.$$

Thus, since $t_n = \Delta s_n$, RRE also coincides with the method *Lsq-Gen* as given by (3.17). Note also that in the case when $\Delta T_n^{(k)}$ is not of full rank, the preceding expression is still valid and the article [67] shows that $t_n^{(k)}$ can be written using pseudo-Schur complements.

As a particular case, assume that we fix n at $n = 0$, and use all forward differences $\Delta s_0, \ldots, \Delta s_k$. In the linear case, $t_0^{(k)}$ is the solution obtained at the k-th step of the full GMRES [72]. Indeed, as proved in [45, Eq. (3.3)], the iterates of the full GMRES for solving the system $Ax = (I - M)x = d$ can be written as a Schur complement. Then, applying the RRE to the sequence generated by $s_{n+1} = Ms_n + d$ with $s_0 = x_0$, one can easily see that, after some algebraic manipulations, the Schur complements of both methods (and thus both methods) are identical since $\Delta s_n = -r_n$ and $\Delta^i s_n = (-1)^i A^{i-1} r_n$ (see (3.9)). Therefore, GMRES can be written under a determinantal form. These authors also showed that GMRES can be considered as a quasi-Newton method. If the linear iterations are restarted from $t_0^{(k)}$, then RRE and GCR(k)/GMRES(k) are mathematically equivalent as proved in [76]. These results were also shown earlier by Beuneu in an unpublished report [4] and in [5] (see also [49, 52]).

According to Theorem 3.1, we have the following corollary.

COROLLARY 4.1. *For any $0 \leq i \leq k$, we have*

$$t_n^{(k)} = s_{n+i} - [\Delta s_n, \ldots, \Delta s_{n+k-1}] \gamma^{(i)},$$

where $\gamma^{(i)} = \text{argmin}_\gamma \|\Delta s_{n+i} - \Delta T_n^{(k)} \gamma\|_2$.

4.3. Anderson Acceleration. Anderson Acceleration (AA) is aimed at the solution of systems of nonlinear equations $f(x) = g(x) - x = 0$, see [2].

Specifically, let x_i, $i = 0, 1, \ldots$, be a given sequence and define $f_i = f(x_i)$. As presented by Walker and Ni [80], or by Ni [59] in his thesis, or by Higham and Strabić [48], AA consists of choosing x_0 and $m \geq 1$, computing $x_1 = g(x_0) = x_0 + \beta_0 f_0$, where $\beta_0 > 0$ is a parameter, and, for $k = 1, 2, \ldots$, after setting $m_k = \min(m, k)$, to compute (using common notation) the vector $\theta^{(k)}$ that solves

$$\min_{\theta \in \mathbb{R}^{m_k}} \|f_k - \Delta F_k \theta\|_2, \tag{4.1}$$

and finally to obtain

$$\bar{x}_k = x_k - \sum_{i=k-m_k}^{k-1} \theta_i^{(k)} \Delta x_i = x_k - \Delta X_k \theta^{(k)} \tag{4.2}$$

$$\bar{f}_k = f_k - \sum_{i=k-m_k}^{k-1} \theta_i^{(k)} \Delta f_i = f_k - \Delta F_k \theta^{(k)}, \tag{4.3}$$

where

$$X_k = [x_{k-m_k}, \ldots, x_{k-1}], \quad F_k = [f_{k-m_k}, \ldots, f_{k-1}],$$

and

$$\theta^{(k)} = (\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T f_k. \tag{4.4}$$

Then, the next iterate of Anderson's method is

$$x_{k+1} = \bar{x}_k + \beta_k \bar{f}_k = x_k + \beta_k f_k - (\Delta X_k + \beta_k \Delta F_k)\theta^{(k)}, \tag{4.5}$$

where β_k is a parameter, usually positive.

We have

$$\bar{x}_k = x_k - [\Delta x_{k-m_k}, \ldots, \Delta x_{k-1}](\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T f_k. \tag{4.6}$$

Thus, $\bar{x}_k$ is the Schur complement of $(\Delta F_k^T \Delta F_k)$ in the matrix M_k given by

$$M_k = \begin{pmatrix} x_k & \Delta x_{k-m_k} \cdots \Delta x_{k-1} \\ \Delta F_k^T f_k & \Delta F_k^T \Delta F_k \end{pmatrix}.$$

Therefore, from the Schur determinantal formula,

$$\bar{x}_k = \frac{\begin{vmatrix} x_k & \Delta x_{k-m_k} & \cdots & \Delta x_{k-1} \\ (\Delta f_{k-m_k}, f_k) & (\Delta f_{k-m_k}, \Delta f_{k-m_k}) & \cdots & (\Delta f_{k-m_k}, \Delta f_{k-1}) \\ \vdots & \vdots & & \vdots \\ (\Delta f_{k-1}, f_k) & (\Delta f_{k-1}, \Delta f_{k-m_k}) & \cdots & (\Delta f_{k-1}, \Delta f_{k-1}) \end{vmatrix}}{\begin{vmatrix} (\Delta f_{k-m_k}, \Delta f_{k-m_k}) & \cdots & (\Delta f_{k-m_k}, \Delta f_{k-1}) \\ \vdots & & \vdots \\ (\Delta f_{k-1}, \Delta f_{k-m_k}) & \cdots & (\Delta f_{k-1}, \Delta f_{k-1}) \end{vmatrix}}. \tag{4.7}$$

A similar expression for $\bar{f}_k$ is obtained by replacing the first row of the determinant in the numerator by $f_k, \Delta f_{k-m_k}, \ldots, \Delta f_{k-1}$, and the following relation holds

$$\bar{f}_k = f_k - [\Delta f_{k-m_k}, \ldots, \Delta f_{k-1}](\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T f_k. \tag{4.8}$$

An alternative way to express the update (4.5) is to rewrite it by defining: $g_i = g(x_i) = x_i + f_i$, and $\bar{g}_k = \bar{x}_k + \bar{f}_k$. This gives,

$$x_{k+1} = \bar{g}_k - \bar{f}_k + \beta_k \bar{f}_k = \bar{g}_k - (1 - \beta_k)\bar{f}_k. \tag{4.9}$$

If we set $G_k = [g_{k-m_k}, \ldots, g_{k-1}]$, we also have $G_k = X_k + F_k$ and $\Delta G_k = \Delta X_k + \Delta F_k = [\Delta g_{k-m_k}, \ldots, \Delta g_{k-1}]$. Then $\bar{g}_k$ satisfies:

$$\begin{aligned} \bar{g}_k &= x_k + f_k - (\Delta X_k + \Delta F_k)\theta^{(k)} \\ &= g_k - \Delta G_k \theta^{(k)} \\ &= g_k - [\Delta g_{k-m_k}, \ldots, \Delta g_{k-1}](\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T f_k. \end{aligned} \tag{4.10}$$

Note also that $\bar{g}_k$ can be expressed by a formula similar to (4.7) in which the first row of the determinant in its numerator is replaced by the row $[g_k, \Delta g_{k-m_k}, \ldots, \Delta g_{k-1}]$.

In practical situations, the *mixing* (also called *damping*) parameter β_k is often set to a fixed nonzero constant. In the case $\beta_k = 0, \forall k$ the iterates simplify to $x_{k+1} = \bar{x}_k$, which is a linear combination of the previous iterates, and this leads to a stagnating sequence. The case $\beta_k = 1, \forall k$ is a common choice in the literature and leads to a new iterate of the form $x_{k+1} = \bar{g}_k$ which is the same as in (4.10). This is the so-called *undamped* iterate.

We now return to the polynomial extrapolation strategy when $E = \mathbb{R}^p$. We replace k by m_k and n by $k - m_k$ in the Schur complement formula (3.7) for $t_n^{(k)}$. By the last expression in Theorem 3.1, we obtain:

$$\begin{aligned} t_{k-m_k}^{(m_k)} = s_k \; - \; & [\Delta s_{k-m_k}, \ldots, \Delta s_{k-1}] \\ & [[y_1, \ldots, y_{m_k}]^T [\Delta t_{k-m_k}, \ldots, \Delta t_{k-1}]]^{-1} [y_1, \ldots, y_{m_k}]^T t_k, \end{aligned}$$

for $k = 1, 2, \ldots$, where $m_k = \min(m, k)$ with $m \geq 1$. That is

$$t_{k-m_k}^{(m_k)} = s_k - [\Delta s_{k-m_k}, \ldots, \Delta s_{k-1}](Y^T \Delta T_{k-m_k}^{(m_k)})^{-1} Y^T t_k. \tag{4.11}$$

Now, consider (4.11) with the fixed point choice $t_i = f_i$. This satisfies (3.3), and $Y = \Delta T_{k-m_k}^{(m_k)} = \Delta F_k$. Comparing this expression with (4.6), (4.8) and (4.10), we see that Anderson Acceleration relates to the polynomial extrapolation strategy. In fact, with the previous choices, when (4.11) is applied to the sequence $s_i = x_i$, the polynomial acceleration yields $t_{k-m_k}^{(m_k)} = \bar{x}_k$. By a similar argument, if we set $s_i = f_i$, we obtain $t_{k-m_k}^{(m_k)} = \bar{f}_k$, and, when $s_i = g_i$, we have $t_{k-m_k}^{(m_k)} = \bar{g}_k$. Thus, by using both relations, we are able to find the new iterate $x_{k+1} = \bar{x}_k + \beta_k \bar{f}_k = \bar{g}_k - (1 - \beta_k)\bar{f}_k$. When $\beta_k = 1$, by only one application of the transformation, we directly obtain the new iterate since $x_{k+1} = \bar{g}_k$. Thus, AA belongs to the class *Pol-Fxp* . From (3.17) with $t_i = f_i$, we see that it is also a method of the class *Lsq-Fxp* .

REMARK 2. *We now comment on the situation where we want to find the fixed point of a mapping $\widetilde{g}$. Let us restrict ourselves to the situation where $\beta_k \equiv \beta$ is a nonzero constant, and set $\widetilde{g}(x) = x + \beta f(x)$. Then a fixed point of $\widetilde{g}$ is also a zero of f. Anderson Acceleration defines x_{k+1} in the first part of (4.5) (i.e., $x_{k+1} = \bar{x}_k + \beta \bar{f}_k$)*

as a natural substitute for $x_{k+1} = \widetilde{g}(\bar{x}_k) = \bar{x}_k + \beta f(\bar{x}_k)$ *which would have been verified if we had* $\bar{f}_k = f(\bar{x}_k)$*. We have instead, as a consequence of (4.5),*

$$x_{k+1} = \widetilde{g}(x_k) - [\Delta\widetilde{g}_{k-m_k}, \ldots, \Delta\widetilde{g}_{k-1}]\theta^{(k)},$$

where $\theta^{(k)}$ *minimizes (4.1) and is equal to (4.4). Obviously, when* $\beta = 1$*, we have* $g = \widetilde{g}$*, and we recover (4.10).*

In [45], the authors also discuss the quasi-Newton Inverse Least Squares method (QN-ILS) proposed in [32]. They proved that it is related to Krylov subspace methods in general, and to GMRES in particular when applied to linear systems.

With our notation, one iteration of the QN-ILS method can be written as

$$x_{k+1} = g_k - [g_k - g_{k-1}, \ldots, g_k - g_{n_k}]\gamma^{(k)}$$

with $n_k = \max(0, k-m)$ and $\gamma^{(k)} = [f_k - f_{k-1}, \ldots, f_k - f_{n_k}]^\dagger f_k$. On the other hand, following [48], the Anderson Acceleration can be written as follows, for $k = 1, 2, \ldots$,

$$u_k = x_k - \sum_{i=1}^{m_k} \theta_i^{(k)}(x_k - x_{k-i}) \tag{4.12}$$

$$v_k = g(x_k) - \sum_{i=1}^{m_k} \theta_i^{(k)}(g(x_k) - g(x_{k-i})) \tag{4.13}$$

$$x_{k+1} = u_k + \beta_k(v_k - u_k), \tag{4.14}$$

with $m_k = \min(m, k)$, $x_1 = g(x_0)$, and where $\theta^{(k)} = (\theta_1^{(k)}, \ldots, \theta_{m_k}^{(k)})^T$ minimizes $\|v_k - u_k\|_2$ that is $\theta^{(k)} = [f_k - f_{k-1}, \ldots, f_k - f_{k-m_k}]^\dagger f_k$. Thus

$$u_k = x_k - [x_k - x_{k-1}, \ldots, x_k - x_{k-m_k}]\theta^{(k)}$$
$$v_k = g_k - [g_k - g_{k-1}, \ldots, g_k - g_{k-m_k}]\theta^{(k)}.$$

When $k < m$, we have $m_k = k$ and $n_k = 0$. When $k \leq m$, that is when $k = m + j$, $j = 0, 1, \ldots$, then $m_k = m$ and $n_k = j$. Thus, the vectors x_k produced by the QN-ILS method are the same as the vectors v_k of AA that are defined above.

It is easy to see that the vectors v_k correspond to the vectors $\bar{g}_k$ as defined in (4.10). Thus, in fact, the QN-ILS method is exactly Anderson Acceleration with $\beta_k = 1$, and its iterates can also be written as ratios of determinants.

4.3.1. Comparison with RRE. We would like to compare the sequence $(t_n^{(k)})$ obtained in RRE with the vector sequence obtained by Anderson Acceleration. In the following we assume that k is fixed and that it is the same for RRE and the Anderson Acceleration.

The article [45] described a method that is identical with RRE and showed that this method is mathematically equivalent to GMRES in the linear case. As discussed earlier, this result was already known in the 1980s, see, e.g., [76]. As proved in [80], when all previous iterates are used, Anderson Acceleration is 'essentially equivalent' (but not completely) in a certain sense to GMRES [71], and thus to RRE. Indeed $x_{k+1}^{AA} = g(x_k^{GMRES})$ and thus also $x_{k+1}^{AA} = g(x_k^{RRE})$. The question now is whether or not there are relations with any one of the extrapolation techniques in the *nonlinear* case.

Let us consider again $t_{k-m_k}^{(m_k)}$ given by Formula (4.11). For the general choice $t_i = \Delta s_i$ and $Y = \Delta T_{k-m_k}^{(m_k)}$, as previously seen, we recover the RRE (which belongs

to the class *Pol-Gen*) expressed with this change in the indices. However, with this procedure, it is not possible to reproduce the vectors $\bar{x}_k$, $\bar{f}_k$ and $\bar{g}_k$ of Anderson Acceleration. Indeed, in particular, we have $f_k = f(x_k)$ in Formulas (4.6), (4.8) and (4.10) of AA, while the RRE needs $t_k = \Delta s_k$ in (4.11). Other combinations of choices for s_i, t_i and Y do not allow to recover the vectors of AA in the general nonlinear case.

Instead, consider RRE in which we set $s_i = g(x_i) = g_i$ for $i = 0, \ldots, k$, and, by using Theorem 3.1, the accelerated member

$$\bar{s}_k = s_k - [\Delta g_{k-m_k}, \ldots, \Delta g_{k-1}]\theta^{(k)} = g_k - [\Delta g_{k-m_k}, \ldots, \Delta g_{k-1}]\theta^{(k)}.$$

If in this formula we were to choose $\theta^{(k)}$ so as to minimize (4.1), we would obtain $\bar{s}_k = \bar{g}_k$ given by AA, and, in the undamped version, we would have $x_{k+1} = \bar{g}_k$.

In RRE, the coefficient $\theta^{(k)}$ satisfies a slightly different optimization criterion, namely, it minimizes $\|\Delta g_k - [\Delta^2 g_{k-m_k}, \ldots, \Delta^2 g_k]\theta\|_2$, where $\Delta^2 g_i = \Delta(g_{i+1} - g_i)$. Thus in the last case, we can also set $x_{k+1} = \bar{s}_k$, compute $s_{k+1} = g(x_{k+1})$ and continue in this way. This AA-like variation of RRE is close, but not quite equivalent, to Anderson acceleration because of the difference in the optimization criterion used to obtain the coefficients θ_i. Note that this difference is subtle. Each vector f_j involved in the least-squares problem (4.1), the right-hand side and column vectors of F_k, is replaced by Δg_j. In the standard RRE, we have $\Delta g_j = g(x_{j+1}) - g(x_j)$ but since $x_{j+1} = g(x_j)$ we would have $\Delta g_j = g(x_{j+1}) - x_{j+1} = f_{j+1}$ which is what is used in the least-squares problem (4.1) of AA. However, in the AA-like variation discussed above the relation $x_{j+1} = g(x_j)$ is no longer true because we defined x_{k+1} as $x_{k+1} = \bar{s}_k \neq g(x_k)$.

An attempt to compare RRE with AA was made in Capehart's PhD thesis [28] using a non-standard interpretation of AA.

4.3.2. The Broyden connection. In *generalized Broyden methods* [37, 79] the authors define a class of Broyden update techniques that give an approximate Jacobian G_k satisfying m secant conditions:

$$G_k \Delta f_i = \Delta x_i, \qquad \text{for } i = k-m, \ldots, k-1,$$

with $f_i = f(x_i)$ and where it is assumed again that the vectors $\Delta f_{k-m}, \ldots, \Delta f_{k-1}$ are linearly independent and $m \leq k$. In matrix form this can be written, using the notations of Anderson Acceleration,

$$G_k \Delta F_k = \Delta X_k,$$

with $X_k = [x_{k-m}, \ldots, x_{k-1}]$ and $F_k = [f_{k-m}, \ldots, f_{k-1}]$ (thus a procedure entering into the class *Pol-Fxp*). A least-change condition is imposed

$$(G_k - G_{k-m})q = 0, \quad \forall q \in \text{span}\{\Delta f_{k-m}, \ldots, \Delta f_{k-1}\}^{\perp}.$$

After calculations we get a rank-m update formula

$$G_k = G_{k-m} + (\Delta X_k - G_{k-m}\Delta F_k)(\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T.$$

The update itself is of the form

$$x_{k+1} = x_k - G_{k-m} f_k - (\Delta X_k - G_{k-m}\Delta F_k)\theta^{(k)}, \quad \theta^{(k)} = (\Delta F_k^T \Delta F_k)^{-1} \Delta F_k^T f_k.$$

Note that it is common in practice to vary m with k (so m could be replaced by m_k).

Setting $G_{k-m} = -\beta_k I$ yields exactly Anderson's original method (4.5). This result was shown by Eyert [37, 79] (see [38]).

5. The Vector Epsilon Algorithm. To complete our overview, let us now discuss the vector ε-algorithm (VEA) [85] as defined in Section 2. When applied to a sequence (s_n) of real vectors (to simplify) satisfying (3.1) the algorithm yields $\varepsilon_{2k}^{(n)} = s$ for all n, a result proved in [57]. Thus, it fits into the general framework laid out in Section 3. However, its algebraic theory is more complicated. The first attempt to express these vectors as a ratio of determinants was proposed in [74], but it involved determinants of dimension $2k+1$ (Formula (30)) instead of $k+1$ as above. The second attempt consisted in working in a noncommutative field, to use designants, which generalize determinants in this setting, and to consider a real Clifford algebra for the theory [73].

There exist left and right designants which were defined and studied in [47]. For example, let $\Delta_r^{(n)}$ be the right designant

$$\Delta_r^{(n)} = \begin{vmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{vmatrix}_r ,$$

where the a_{ij}'s belong to a noncommutative field. This designant can be recursively computed (and thus defined) as follows. We start from

$$\Delta_r^{(2)} = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}_r = a_{22} - a_{12}a_{11}^{-1}a_{21}.$$

Let A_{qr}^p be the right designant of order $p+1$ obtained from $\Delta_r^{(n)}$ by keeping the rows 1 to q, and the columns 1 to p and the column r. Then, we have

$$\Delta_r^{(n)} = \begin{vmatrix} A_{n-1,n-1}^{n-2} & A_{n-1,n}^{n-2} \\ A_{n,n-1}^{n-2} & A_{n,n}^{n-2} \end{vmatrix}_r = A_{n,n}^{n-2} - A_{n-1,n}^{n-2}(A_{n-1,n-1}^{n-2})^{-1}A_{n,n-1}^{n-2}.$$

Obviously, this formula looks like a Schur complement.

Designants are used in the solution of systems of linear equations in a noncommutative field [60]. Thus they are useful in our context, and it was proved by Salam [73] that the vectors $\varepsilon_{2k}^{(n)}$ obtained by applying the vector ε-algorithm to a sequence of vectors (s_n) are given by

$$\varepsilon_{2k}^{(n)} = \begin{vmatrix} \Delta s_n & \cdots & \Delta s_{n+k-1} & S_n \\ \vdots & & \vdots & \vdots \\ \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} & S_{n+k} \end{vmatrix}_r \begin{vmatrix} \Delta s_n & \cdots & \Delta s_{n+k-1} & 1 \\ \vdots & & \vdots & \vdots \\ \Delta s_{n+k} & \cdots & \Delta s_{n+2k-1} & 1 \end{vmatrix}_r^{-1} .$$

A similar result holds with left designants.

6. Concluding remarks. Methods for accelerating the convergence of various processes have been developed by researchers in a wide range of disciplines, often without being aware of similar efforts undertaken elsewhere. Certainly, differences in terminology and notation have played a role in hampering the exchange of ideas across different arenas. In this paper, we gave a general framework for sequence transformations based on kernels of the form (3.1) and (3.3). This framework includes many known and widely used transformations, and it allows to derive new ones. Their connections with quasi-Newton and Broyden methods have been pointed out.

The Anderson Acceleration article appeared about one decade before the Kaniel and Stein [54] version of RRE and 13 years before the RRE paper [36]. It is only recently that the literature has explored the various relations between these methods. To be able to make links between different acceleration schemes, it was necessary to overcome the scientific language barrier. In the case of the RRE, the MPE, and Anderson Acceleration, it was essential to express the RRE and the MPE accelerated sequences differently, specifically as an update from the last iterate instead of a delayed iterate. It is hoped that these alternative expressions will help unravel other, yet unknown, connections.

Acknowledgments: We thank the reviewers for pointing out an error in a previous version of this paper. Their constructive comments encouraged us to expand this study and develop the general framework discussed here. C.B. would like to thank Prof. Yvon Maday for drawing his attention on Anderson Acceleration. The work of C.B. was supported in part by the Labex CEMPI (ANR-11-LABX-0007-01) and, in part, by the University of Padua. The work of M.R.-Z. was partially supported by the University of Padua, Project 2014 No. CPDA143275. The work of Y.S. was supported by the Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research and Basic Energy Sciences DE-SC0008877. This work started as a result of a presentation made by Y.S. at the workshop *"Numerical Methods for Large-Scale Nonlinear Problems and Their Applications*, held at ICERM (Providence, RI, USA) in August 31 - September 4, 2015.

REFERENCES

[1] A. Aitken, *On Bernoulli's numerical solution of algebraic equations*, Proc. Roy. Soc. Edinburgh, 46 (1926), p. 289305.

[2] D. G. Anderson, *Iterative procedures for non-linear integral equations*, Assoc. Comput. Mach., 12 (1965), pp. 547–560.

[3] J. G. P. Barnes, *An algorithm for solving nonlinear equations based on the secant method*, Computer J., 8 (1965), pp. 66–72.

[4] J. Beuneu, *Méthodes de projection à convergence finie. Remarques sur leur forme incomplète*, Tech. Rep. ANO 80, Université de Lille I, 1982.

[5] ———, *Méthodes de projection-minimisation pour les problèmes linéaires*, RAIRO, Analyse Numérique, 17 (1983), pp. 221 – 248.

[6] L. Bittner, *Eine Verallgemeinerung des Sekantenverfahrens (regula falsi) zur näherungsweisen Berechnung der Nullstellen eines nichtlinearen Gleichungssystems*, Wissen. Zeit. der Technischen Hochschule Dresden, 9 (1959), pp. 325–329.

[7] A. Bouhamidi, K. Jbilou, L. Reichel, and H. Sadok, *Vector extrapolation applied to truncated singular value decomposition and truncated iteration*, J. Engrg. Math., 93 (2015), pp. 99–122.

[8] C. Brezinski, *Application de l'ε–algorithme à la résolution des systèmes non linéaires*, C. R. Acad. Sci. Paris, 271A (1970), pp. 1174–1177.

[9] ———, *Sur un algorithme de résolution des systèmes non linéaires*, C. R. Acad. Sci. Paris, 272A (1971), pp. 145–148.

[10] ———, *Some results in the theory of the vector ε-algorithm*, Linear Algebra Appl., 8 (1974), pp. 77–86.

[11] ———, *Généralisation de la transformation de Shanks, de la table de Padé et de l'ε–algorithme*, Calcolo, 12 (1975), pp. 317–360.

[12] ———, *Padé-Type Approximation and General Orthogonal Polynomials*, Birkhäuser-Verlag, Basel-Boston-Stuttgart, 1980.

[13] ———, *About Henrici's method for nonlinear equations.* Symposium on Numerical Analysis and Computational Complex Analysis, Zürich, August 1983, unpublished.

[14] ———, *Some determinantal identities in a vector space, with applications*, in Padé Approxi-

mation and its Applications. Bad-Honnef 1983, H. Werner and H. J. Bünger, eds., Lecture Notes in Mathematics, vol. 1071, Berlin, 1984, Springer-Verlag, pp. 1–11.
[15] ———, *Other manifestations of the Schur complement*, Linear Algebra Appl., 11 (1988), pp. 231–247.
[16] ———, *Bordering methods and progressive forms for sequence transformations*, Zastos. Mat., 20 (1990), pp. 435–443.
[17] ———, *Projection Methods for Systems of Equations*, Elsevier, Amsterdam, 1997.
[18] ———, *Vector sequence transformations: methodology and applications to linear systems*, J. Comput. Appl. Math., 98 (1998), pp. 149–175.
[19] ———, *Convergence acceleration during the 20th century*, J. Comput. Appl. Math., 122 (2000), pp. 1 – 21. Numerical Analysis in the 20th Century Vol. II: Interpolation and Extrapolation.
[20] C. Brezinski and M. Crouzeix, *Remarques sur le procédé Δ^2 d'Aitken*, C. R. Acad. Sci. Paris, 270A (1970), pp. 896–898.
[21] C. Brezinski and M. Redivo-Zaglia, *Extrapolation Methods: Theory and Practice*, North-Holland, Amsterdam, 1991.
[22] C. Brezinski and M. Redivo-Zaglia, *Vector and matrix sequence transformations based on biorthogonality*, Appl. Numer. Math., 21 (1996), pp. 353–373.
[23] ———, *The simplified topological epsilon-algorithms for accelerating sequences in a vector space*, SIAM J. Scient. Comput., 36 (2014), pp. A2227–A2247.
[24] ———, *The simplified topological ε–algorithms: software and applications*, Numer. Algorithms, (2017). To appear, DOI: 10.1007/s11075-016-0238-0.
[25] C. Brezinski and H. Sadok, *Vector sequence transformations and fixed point methods*, in Numerical Methods in Laminar and Turbulent Flows, C. Taylor, ed., Swansea, 1987, Pineridge Press, pp. 3–11.
[26] S. Cabay and L. W. Jackson, *A polynomial extrapolation method for finding limits and antilimits of vector sequences*, SIAM J. Numer. Anal., 13 (1976), pp. 734–752.
[27] C. Cantor, *Variants of the secant method for solving nonlinear systems of equations*, Tech. Rep. X-733-71-48, Goddard Space Flight Center, Greenbelt, Maryland,USA, 1971.
[28] S. R. Capehart, *Techniques for Accelerating Iterative Methods for the Solution of Mathematical Problems*, PhD thesis, Oklahoma State University, Stillwater, Oklahoma, USA, July 1989.
[29] P. G. Ciarlet, *Linear and Nonlinear Functional Analysis with Application*, SIAM, Philadelphia, 2013.
[30] P. G. Ciarlet and C. Mardare, *On the Newton-Kantorovich theorem*, Anal. Appl. (Singap.), 10 (2012), pp. 249–269.
[31] P. Csázár and P. Pulay, *Geometry optimization by direct inversion in the iterative subspace*, J. Mol. Struct., 114 (1984), pp. 31–34.
[32] J. Degroote, K.-J. Bathe, and J. Vierendeels, *Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction*, Comput. & Structures, 87 (2009), pp. 798–801.
[33] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice Hall, Englewood Cliffs, NJ, 1983.
[34] S. Duminil and H. Sadok, *Reduced rank extrapolation applied to electronic structure computations*, Electron. Trans. Numer. Anal., 38 (2011), pp. 347–362.
[35] S. Duminil, H. Sadok, and D. B. Szyld, *Nonlinear Schwarz iterations with reduced rank extrapolation*, Appl. Numer. Math., 94 (2015), pp. 209–221.
[36] R. P. Eddy, *Extrapolation to the limit of a vector sequence*, in Information Linkage Between Applied Mathematics and Industry, P. C. C. Wang, ed., New York, 1979, Academic Press, pp. 387–396.
[37] V. Eyert, *A comparative study on methods for convergence acceleration of iterative vector sequences*, J. Comput. Phys, 124 (1996), pp. 271–285.
[38] H. Fang and Y. Saad, *Two classes of multisecant methods for nonlinear acceleration*, Numer. Linear Algebra Appl., 16 (2009), pp. 197–221.
[39] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis, Dundee 1975, G. Watson, ed., LNM vol. 506, Berlin, 1976, Springer, pp. 73–89.
[40] W. Gander, G. H. Golub, and D. Gruntz, *Solving linear equations by extrapolation*, in Supercomputing, J. Kowalik, ed., vol. 62 of NATO ASI Studies, 1989, pp. 279–293.
[41] A. J. Garza and G. Scuseria, *Comparison of self-consistent field convergence acceleration techniques*, J. Chem. Phys., 137 (2012), p. 054110.
[42] E. Gekeler, *On the solution of systems of equations by the epsilon algorithm of Wynn*, Math. Comput., 26 (1972), pp. 427–436.
[43] B. Germain-Bonne, *Estimation de la Limite de Suites et Formalisation de Procédés*

d'Accélération de Convergence, PhD thesis, Université des Sciences et Techniques de Lille, Lille, France, 1978.
[44] P. R. Graves-Morris and C. D. Jenkins, *Vector-valued rational interpolants, III*, Constr. Approx., 2 (1986), pp. 263–289.
[45] R. Haelterman, J. Degroote, D. V. Heule, and J. Vierendeels, *On the similarities between the quasi-Newton inverse least squares method and GMRES*, SIAM J. Numer. Anal., 47 (2010), pp. 4660–4679.
[46] P. Henrici, *Elements of Numerical Analysis*, Wiley, New York, 1964.
[47] A. Heyting, *Die Theorie der linearen Gleichungen in einer Zahlenspezies mit nichtkommutativer Multiplikation*, Math. Ann., 98 (1927), pp. 465–490.
[48] N. J. Higham and N. Strabić, *Anderson acceleration of the alternating projections method for computing the nearest correlation matrix*, Numer. Algorithms, 72 (2016), pp. 1021–1042.
[49] K. Jbilou, *Méthodes d'Extrapolation et de Projection. Applications aux Suites de Vecteurs*, PhD thesis, Université des Sciences et Techniques de Lille, 1988.
[50] ———, *A general projection algorithm for solving systems of linear equations*, Numer. Algorithms, 4 (1993), pp. 361–377.
[51] K. Jbilou and H. Sadok, *Some results about vector extrapolation methods and related fixed point iteration*, J. Comput. Appl. Math., 36 (1991), pp. 385–398.
[52] ———, *Analysis of some vector extrapolation methods for solving systems of linear equations*, Numer. Math., 70 (1995), pp. 73–89.
[53] ———, *Vector extrapolation methods. applications and numerical comparison*, J. Comput. Appl. Math., 122 (2000), pp. 149–165.
[54] S. Kaniel and J. Stein, *Least-square acceleration of iterative methods for linear equations*, J. Optim. Theory Appl., 14 (1974), pp. 431–437.
[55] C. Lanczos, *Solution of systems of linear equations by minimized iterations*, J. Res. Natl. Bur. Stand., 49 (1952), pp. 33–53.
[56] H. Le Ferrand, *The quadratic convergence of the topological epsilon algorithm for systems of nonlinear equations*, Numer. Algorithms, 3 (1992), pp. 273–284.
[57] J. B. McLeod, *A note on the ε-algorithm*, Computing, 7 (1971), pp. 17–24.
[58] M. Mešina, *Convergence acceleration for the iterative solution of the equations $x = ax + f$*, Comput. Methods Appl. Mech. Engrg., 10 (1977), pp. 165–173.
[59] P. Ni, *Anderson Acceleration of Fixed-point Iteration with Applications to Electronic Structure Computations*, PhD thesis, Worcester Polytechnic Institute, Worcester, Massachusetts, USA, 2009.
[60] O. Ore, *Linear equations in non-commutative fields*, Ann. Math., 32 (1931), pp. 463–477.
[61] J. M. Ortega and W. C. Rheinbolt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.
[62] D. V. Ouellette, *Schur complements and statistics*, Linear Algebra Appl., 36 (1981), pp. 187–295.
[63] F. Potra and H. Engler, *A characterization of the behavior of the Anderson acceleration on linear problems*, Linear Algebra Appl., 438 (2013), p. 1002 1011.
[64] 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 (1978), pp. 199–207.
[65] P. Pulay, *Convergence acceleration of iterative sequences. the case of SCF iteration*, Chem. Phys. Lett., 73 (1980), pp. 393–398.
[66] ———, *Improved SCF convergence*, J. Comput. Chem., 3 (1982), pp. 556–560.
[67] M. Redivo-Zaglia, *Pseudo-Schur complements and their properties*, Appl. Numer. Math., 50 (2004), pp. 511–519.
[68] L. F. Richardson, *The approximate arithmetical solution by finite differences of physical problems including differential equations, with an application to the stresses in a masonry dam*, Philos. Trans. A, 210 (1911), pp. 459–470.
[69] L. F. Richardson and J. A. Gaunt, *The deferred approach to the limit*, Philos. Trans. A, 226 (1927), pp. 636–646.
[70] T. Rohwedder and R. Schneider, *An analysis for the DIIS acceleration method used in quantum chemistry calculations*, J. Math. Chem., 49 (2011), pp. 1889–1914.
[71] Y. Saad, *Iterative Methods for Sparse Linear Systems, 2nd edition*, SIAM, Philadelpha, PA, 2003.
[72] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Scient. Stat. Comput., 7 (1986), pp. 856–869.
[73] A. Salam, *Non-commutative extrapolation algorithms*, Numer. Algorithms, 7 (1994), pp. 225–251.

[74] A. Salam and P. R. Graves-Morris, *On the vector ε-algorithm for solving linear systems of equations*, Numer. Algorithms, 29 (2002), pp. 229–247.
[75] D. Shanks, *Non-linear transformations of divergent and slowly convergent sequences*, J. Math. Phys., 34 (1955), p. 142.
[76] A. Sidi, *Extrapolation vs. projection methods for linear systems of equations*, J. of Comput. and Appl. Math., 22 (1988), pp. 71–88.
[77] D. A. Smith, W. F. Ford, and A. Sidi, *Extrapolation methods for vector sequences*, SIAM Rev., 29 (1987), pp. 199–233.
[78] A. Toth and C. T. Kelley, *Convergence analysis for Anderson acceleration*, SIAM J. Numer. Anal., 53 (2015), p. 805819.
[79] D. Vanderbilt and S. G. Louie, *Total energies of diamond (111) surface reconstructions by a linear combination of atomic orbitals method*, Phys. Rev. B, 30 (1984), pp. 6118–6130.
[80] H. F. Walker and P. Ni, *Anderson acceleration for fixed-point iterations*, SIAM J. Numer. Anal., 49 (2011), pp. 1715–1735.
[81] W. T. J. Willert, W. T. Taitano, and D. Knoll, *Leveraging Anderson Acceleration for improved convergence of iterative solutions to transport systems*, J. Comput. Phys, 273 (2014), p. 278286.
[82] J. Wimp, *Sequence Transformations and their Applications*, Academic Press, New York, 1981.
[83] P. Wolfe, *The secant method for simultaneous nonlinear equations*, Comm. Assoc. Comp. Mach., 2 (1959), pp. 12–13.
[84] P. Wynn, *On a device for computing the $e_m(s_n)$ transformation*, Math. Tables Other Aids Comput., 10 (1956), pp. 91–96.
[85] ———, *Acceleration techniques for iterated vector and matrix problems*, Math. Comput., 16 (1962), pp. 301–322.
[86] F. Zhang, ed., *The Schur Complement and its Applications*, Springer, New York, 2005.