Numerical Algorithms **29:** 229–247, 2002.

On the vector ε-algorithm for solving linear systems of equations

A. Salam [a] and P.R. Graves-Morris [b]

[a] *Laboratoire de Mathématiques Pures et Appliquées J. Liouville, Université du Littoral, Centre Universitaire de la Mi-Voix, Bâtiment Poincaré, 50 rue F. Buisson, B.P. 699, 62228, Calais Cedex, France*
E-mail: Ahmed.Salam@lmpa.univ-littoral.fr

[b] *Department of Computing, University of Bradford, West Yorkshire BD7 1DP, England*
E-mail: P.R.Graves-Morris@bradford.ac.uk

Received 6 October 2000; revised 10 March 2001

The four vector extrapolation methods, minimal polynomial extrapolation, reduced rank extrapolation, modified minimal polynomial extrapolation and the topological epsilon algorithm, when applied to linearly generated vector sequences are Krylov subspace methods and it is known that they are equivalent to some well-known conjugate gradient type methods. However, the vector ε-algorithm is an extrapolation method, older than the four extrapolation methods above, and no similar results are known for it. In this paper, a determinantal formula for the vector ε-algorithm is given. Then it is shown that, when applied to a linearly generated vector sequence, the algorithm is also a Krylov subspace method and for a class of matrices the method is equivalent to a preconditioned Lanczos method. A new determinantal formula for the CGS is given, and an algebraic comparison between the vector ε-algorithm for linear systems and CGS is also given.

Keywords: extrapolation, vector Padé approximation, projection methods, Krylov subspace methods, Lanczos methods

AMS subject classification: 65F05, 65F10, 65F25, 65B05

1. Introduction

The connection between extrapolation methods for sequences of vectors and methods for solving systems of linear equations has been investigated by several authors. In [5], Brezinski showed that the topological ε-algorithm is equivalent to the Lanczos method. For other extrapolation methods, an extensive study has been developed by Sidi [33]. The extrapolation methods that he considered are the minimal polynomial extrapolation (MPE) of Cabay and Jackson [11], the reduced rank extrapolation (RRE) of Eddy [12] and Mešina [21], the modified minimal polynomial extrapolation (MMPE) of Sidi et al. [34], and the topological ε-algorithm of Brezinski [4]. Sidi showed that the four extrapolation methods, when applied to linearly generated vector sequences are bona fide Krylov subspace methods and conjugate gradient type methods as well [2]. More precisely, he showed that the MPE is equivalent to the Arnoldi method [1,26], the RRE is equivalent to the generalized conjugate residual method GCR of Eisenstat et al. [13] and the topological ε-algorithm is equivalent to Lanczos method [19]. However, no connection between projection methods and the popular vector ε-algorithm, which is a powerful method for accelerating vector sequences, was given. This algorithm was proposed by Wynn [39,40] by extending the scalar epsilon algorithm [38] to vectors by using the Samelson inverse. Note that the first four extrapolation methods above have a familiar formula as a ratio of two determinants. These formulae are at the origin of the work given by Sidi [33] and Brezinski [5]. The aim of this paper is to derive a determinantal formula for the vector ε-algorithm from the determinantal formula of the denominator for the generalized inverse Padé approximant given by Graves-Morris and Jenkins in [18]. Then, we establish new results for the algorithm when applied to linearly generated vector sequences. More precisely, we show that the method is a Krylov subspace method. We show that for the case of skew-symmetric matrices, it is equivalent to a preconditioned Lanczos method. We present also an algebraic comparison in the symmetric case with Lanczos method. In the general case, an algebraic comparison with the CGS [36] will be given. First we recall the extrapolation methods that will be considered. For more details, one can see, for example, [8,14,32,34,35, and references therein].

2. Shanks transformation and vector generalizations

Let $X = (x_i)_{i\in\mathbb{N}}$ be a sequence of real numbers. Shanks' transformation [31] is an extrapolation method for accelerating the convergence of a sequence X. It consists in computing the quantities $e_k^n(X)$ as follows

$$e_k^n(X) = \sum_{j=0}^{k} \gamma_j^{(n,k)} x_{n+j}, \tag{1}$$

where

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} = 1 \tag{2}$$

and satisfying the linear system

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} u_{ij}^{(n)} = 0, \quad i = 0, \ldots, k-1, \tag{3}$$

with $u_{ij}^{(n)} = \Delta x_{n+i+j}$, where Δ is the difference operator $\Delta x_n = x_{n+1} - x_n$. From these equations, it is easy to obtain a determinantal formula for $e_k^n(X)$

$$e_k^n(X) = \frac{\begin{vmatrix} x_n & \dots & x_{n+k} \\ \Delta x_n & \dots & \Delta x_{n+k} \\ \vdots & & \vdots \\ \Delta x_{n+k-1} & \dots & \Delta x_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ \Delta x_n & \dots & \Delta x_{n+k} \\ \vdots & & \vdots \\ \Delta x_{n+k-1} & \dots & \Delta x_{n+2k-1} \end{vmatrix}}, \tag{4}$$

provided that the denominator of the right-hand side of (4) is nonzero. If, for a fixed k, the sequence X is such that there exists $s \in \mathbb{R}$ and $\gamma_0, \dots, \gamma_k \in \mathbb{R}$, with $\sum_{i=0}^k \gamma_i \neq 0$, satisfying

$$\gamma_0(x_n - s) + \dots + \gamma_k(x_{n+k} - s) = 0 \quad \forall n,$$

then $e_k^n(X) = s \ \forall n \in \mathbb{N}$.

The set of such real sequences is called the kernel of the transformation. A recursive rule for computing the quantities $e_k^n(X)$ of Shanks transformation has been given by Wynn [38]. It is the ε-algorithm

$$\varepsilon_{-1}^{(n)} = 0, \qquad \varepsilon_0^{(n)} = x_n, \qquad \varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + \left(\varepsilon_k^{(n+1)} - \varepsilon_k^{(n)}\right)^{-1}, \quad \forall k, n \geqslant 0. \tag{5}$$

We have

$$\varepsilon_{2k}^{(n)} = e_k^n(X).$$

When (x_n) is a real vector sequence, the rule of the ε-algorithm can be used to accelerate the sequence if the inverse of a vector is defined. Wynn takes the Samelson inverse

$$y^{-1} = \frac{y}{\|y\|_2^2}$$

and the resulting algorithm is the vector ε-algorithm (VEA) [39]. The algorithm lacks a determinantal formula. This is a real handicap, since this kind of formula is useful in the study of the properties of the algorithm. It has been conjectured by Wynn and proved by McLeod [20] that if the real vector sequence X has the property that

$$\sum_{i=0}^k \gamma_i(x_{n+i} - s) = 0, \quad \forall n \in \mathbb{N}, \tag{6}$$

then $\varepsilon_{2k}^n = s$. A quite different proof for the complex case was given by Graves-Morris [15].

Another generalization of the ε-algorithm for vector sequences is the topological ε-algorithm of Brezinski [4]. Because the vector ε-algorithm lacked a determinantal formula, Brezinski proposed another approach for generalizing the scalar ε-algorithm to vectors. It leads to a determinantal formula and to a recursive algorithm: the topological ε-algorithm.

The topological ε-algorithm corresponds in computing the double sequence $e_k^n(X)$ in the following way

$$e_k^n(X) = \sum_{j=0}^{k} \gamma_j^{(n,k)} x_{n+j}, \tag{7}$$

where

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} = 1 \tag{8}$$

and satisfying the linear system

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} u_{ij}^{(n)} = 0, \quad i = 0, \dots, k-1, \tag{9}$$

with $u_{ij}^{(n)} = (y, \Delta x_{n+i+j})$, where $(\cdot\,,\cdot)$ denotes the usual scalar product, Δ the difference operator $\Delta x_n = x_{n+1} - x_n$ and y an arbitrary vector.

From these equations, it is easy to obtain a determinantal formula for $e_k^n(X)$

$$e_k^n(X) = \frac{\begin{vmatrix} x_n & \dots & x_{n+k} \\ u_{00}^{(n)} & \dots & u_{0,k}^{(n)} \\ \vdots & & \vdots \\ u_{k-1,0}^{(n)} & \dots & u_{k-1,k}^{(n)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(n)} & \dots & u_{0,k}^{(n)} \\ \vdots & & \vdots \\ u_{k-1,0}^{(n)} & \dots & u_{k-1,k}^{(n)} \end{vmatrix}}. \tag{10}$$

The numerator is a vector obtained by expanding the determinant with respect to its first row using the classical rules.

The quantities $e_k^n(X)$ are computed by the algorithm

$$\begin{aligned} &\varepsilon_{-1}^{(n)} = 0, \qquad \varepsilon_0^{(n)} = x_n, && n = 0, 1, \dots, \\ &\varepsilon_{2k+1}^{(n)} = \varepsilon_{2k-1}^{(n+1)} + \frac{y}{(y, \varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)})}, \\ &\varepsilon_{2k+2}^{(n)} = \varepsilon_{2k}^{(n+1)} + \frac{\varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)}}{(\varepsilon_{2k+1}^{(n+1)} - \varepsilon_{2k+1}^{(n)}, \varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)})}, && n, k = 0, 1, \dots. \end{aligned} \tag{11}$$

Letting ΔX denote the vector sequence (Δx_n), we have [4]

$$e_k^n(X) = \varepsilon_{2k}^{(n)}, \qquad \varepsilon_{2k+1}^{(n)} = \frac{y}{(y, e_k^n(\Delta X))}. \tag{12}$$

If X satisfies (6), then the topological ε-algorithm gives $\varepsilon_{2k}^n = s$, and from (7), (8), we note that $e_k^n(X)$ is a barycentric combination of $x_n, \dots, x_{n+k}$.

3. Polynomial extrapolation methods

The three other polynomial extrapolation methods considered by Sidi et al. are the minimal polynomial extrapolation (MPE) of Cabay and Jackson [11], the reduced rank extrapolation of Eddy and Mešina (RRE) [12,21] and the modified minimal polynomial extrapolation (MMPE) [34]. They have similar structure. Setting $v_i = \Delta x_i$, and $w_i = \Delta v_i$, $\forall i \in \mathbb{N}$, the MPE extrapolated value $e_k^n(X)$ is expressed as a barycentric combination of $k+1$ successive iterates

$$e_k^n(X) = \sum_{j=0}^{k} \gamma_j^{(n,k)} x_{n+j}, \tag{13}$$

where the $\gamma_j^{(n,k)}$ solve the system

$$\sum_{j=0}^{k} \gamma_j^{(n,k)} = 1,$$
$$\sum_{j=0}^{k} \gamma_j^{(n,k)} u_{ij}^{(n)} = 0, \quad i = 0, \dots, k-1,$$

with $u_{ij}^{(n)} = (v_{n+i}, v_{n+j})$. The extrapolated value of RRE is obtained in the same way with $u_{ij}^{(n)} = (w_{n+i}, v_{n+j})$ and with $u_{ij}^{(n)} = (q_i, v_{n+j})$ for MMPE. The set $\{q_i,\ i = 0, \dots, k-1\}$ is a set of arbitrary linearly independent vectors.

The extrapolated values $e_k^n(X)$ of the MPE, RRE and MMPE have a determinantal formula

$$e_k^n(X) = \frac{\begin{vmatrix} x_n & \dots & x_{n+k} \\ u_{00}^{(n)} & \dots & u_{0,k}^{(n)} \\ \vdots & & \vdots \\ u_{k-1,0}^{(n)} & \dots & u_{k-1,k}^{(n)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(n)} & \dots & u_{0,k}^{(n)} \\ \vdots & & \vdots \\ u_{k-1,0}^{(n)} & \dots & u_{k-1,k}^{(n)} \end{vmatrix}}, \tag{14}$$

provided that the denominator of the right-hand side of (14) is nonzero. If the real vector sequence X is such that (6) is satisfied, then $e_k^n(X) = s$ for the three methods.

Let A be a real square matrix in $\mathbb{R}^{d\times d}$, $b \in \mathbb{R}^d$ and consider the system of linear equations expressed by

$$x = Ax + b. \tag{15}$$

Let x_0 be a given starting vector and then the fixed-point iteration

$$x_{j+1} = Ax_j + b \tag{16}$$

generates the Richardson series solution of (15). The residual of x_j is defined by $r(x_j) = (A - I)x_j + b$. We assume that $I - A$ is nonsingular. When the sequence converges, which is the case when the 2-norm of A is less than 1, it converges to the fixed point x solution of the linear system $(I - A)x = b$. When applied to $X = (x_j)$ as in (16), it is known that the MPE, RRE, MMPE and TEA normally lead to the exact solution of the system after a finite number of iterations. In fact, let $e_j = x_j - x$ denote the error. It is easy to see that $e_j = A^j e_0$. Next, let P_k be some polynomial with coefficients γ_j such that $P_k(1) = 1$, $c_k \neq 0$ and

$$P_k(A)(x_0 - x) = \sum_{j=0}^{k} \gamma_j A^j (x_0 - x) = \sum_{j=0}^{k} \gamma_j (x_j - x) = 0.$$

From the Cayley–Hamilton theorem, such a nontrivial polynomial P_k exists. Thus, the sequence X satisfies (6) and the four extrapolation methods give $e_k^0(X) = x$. The results that MPE is mathematically equivalent to the method of Arnoldi, RRE is equivalent to the method Orthomin or Generalized Conjugate Residual, that TEA is equivalent to the Lanczos method and MMPE is equivalent to the Generalized Lanczos method can readily be obtained from the determinantal formulas of those methods, together with orthogonality arguments [1,13,14,19,27,33].

Since some comparison with Lanczos method will be made, we recall briefly some of its properties. Let z_0 be a nonzero arbitrary vector in $\mathbb{R}^d$ and let $r_0 = b - (I - A)z_0$ be its associated residual. To solve (15), Lanczos method generates approximations (z_k^{Lan}) of the solution, defined by

$$z_k^{\mathrm{Lan}} - z_0 \in \mathcal{K}_k(A, r_0) \tag{17}$$

and

$$r_k^{\mathrm{Lan}} = b - (I - A)z_k^{\mathrm{Lan}} \perp \mathcal{K}_k\big(A^{\mathrm{T}}, r_0\big), \tag{18}$$

where $\mathcal{K}_k(A, r) = \mathrm{span}(r, Ar, \ldots, A^{k-1}r)$ and A^{T} denotes the transpose of A.

Let c be the linear functional whose moments are $c_i = (r_0, A^i r_0)$ and let (p_k) be the sequence of orthogonal polynomials with respect to c satisfying $p_k(1) = 1$. The

p_k are called Lanczos polynomials. It is known that p_k can be written as a ratio of two determinants

$$p_k(x) = \frac{\begin{vmatrix} 1 & \dots & x^k \\ (y, r_0) & \dots & (y, A^k r_0) \\ \vdots & & \vdots \\ (y, A^{k-1} r_0) & \dots & (y, A^{2k-1} r_0) \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ (y, r_0) & \dots & (y, A^k r_0) \\ \vdots & & \vdots \\ (y, A^{k-1} r_0) & \dots & (y, A^{2k-1} r_0) \end{vmatrix}}. \tag{19}$$

In [10], it is shown that the topological ε-algorithm iterate $e_k^0(X) = z_k^{\mathrm{Lan}}$, with $x_0 = z_0$ and $y = r_0$. The Lanczos residual can be written as

$$r_k^{\mathrm{Lan}} = p_k(A) r_0. \tag{20}$$

4. VEA is also a Krylov subspace method

Since the VEA can be viewed as a direct method for solving linear systems [3], it is interesting to study its possible links with other direct methods. Similar results for VEA have not been found probably because the determinantal formula is less familiar. In this paper we shall show that one can put the VEA in the same framework as the four extrapolation methods above. Then we try to find similar results when it is applied to linear systems.

4.1. A determinantal formula for VEA

In [15,16] Graves-Morris introduced Generalized Inverse Padé Approximants (GIPA) based on vector rational interpolation, by using vector continued fractions. In [18], a determinantal formula for the denominator of the GIPA and its connection with the vector ε-algorithm were given by Graves-Morris and Jenkins. One can also refer to [23–25,28,29] for other linked developments. Let us recall these two fundamental results which will be used in this paper. The GIPA of type $[n/2k]$ is a rational fraction

$$r^{[n/2k]}(x) = \frac{p^{[n/2k]}(x)}{q^{[n/2k]}(x)}, \tag{21}$$

where $p^{[n/2k]}(x)$ is a polynomial of degree $\partial p^{[n/2k]} \leqslant n$ with vector coefficients, and $q^{[n/2k]}(x)$ is a real polynomial of degree $\partial q^{[n/2k]} \leqslant 2k$. These polynomials satisfy

$$q^{[n/2k]}(0) \neq 0, \tag{22}$$

$$q^{[n/2k]}(x) \mid \left(p^{[n/2k]}(x), p^{[n/2k]}(x)\right), \tag{23}$$

and

$$p^{[n/2k]}(x) - q^{[n/2k]}(x) f(x) = \mathrm{O}\big(x^{n+1}\big), \tag{24}$$

where f is the power series

$$f(x) = \sum_{i=0}^{\infty} c_i x^i, \quad c_i \in \mathbb{R}^d. \tag{25}$$

The first result [18] is that the denominator $q^{[n+2k/2k]}(x)$ of the GIPA of type $[n+2k/2k]$ for the series f is given by

$$q^{[n+2k/2k]}(x) = \begin{vmatrix} x^{2k} & x^{2k-1} & \dots & 1 \\ \mu_{00}^{(n+2k)} & \mu_{01}^{(n+2k)} & \dots & \mu_{0,2k}^{(n+2k)} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{2k-1,0}^{(n+2k)} & \mu_{2k-1,1}^{(n+2k)} & \dots & \mu_{2k-1,2k}^{(n+2k)} \end{vmatrix}, \tag{26}$$

where

$$\begin{cases} \mu_{ij}^{(n+2k)} = \displaystyle\sum_{l=0}^{j-i-1} (c_{i+n+l+1}, c_{j+n-l}), & j > i, \\ \mu_{ij}^{(n+2k)} = -\mu_{ji}^{(n+2k)}, & j < i, \\ \mu_{ii}^{(n+2k)} = 0. \end{cases} \tag{27}$$

The second one is that, if the vector ε-algorithm is initialized by

$$\varepsilon_0^{(j)} = \sum_{i=0}^{j} c_i, \quad j = 0, 1, \dots, \tag{28}$$

and provided that zero divisors are not encountered in the construction of the $\varepsilon_{2k}^{(n)}$, the GIPA $r^{[n+2k/2k]}(x)$ for $f(x)$ exists and one has

$$\varepsilon_{2k}^{(n)} = r^{[n+2k/2k]}(1). \tag{29}$$

Let (s_j) be a vector sequence to be accelerated by the vector ε-algorithm with the classical initializations $\varepsilon_{-1}^{(n)} = 0$, $\varepsilon_0^{(n)} = s_n$, $\forall n \in \mathbb{N}$. Using the Graves-Morris and Jenkins results above, it is easy to obtain a determinantal formula.

Theorem 1. We have

$$\varepsilon_{2k}^{(n)} = \frac{\begin{vmatrix} s_n & \dots & s_{n+2k} \\ u_{00}^{(n)} & \dots & u_{0,2k}^{(n)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(n)} & \dots & u_{2k-1,2k}^{(n)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(n)} & \dots & u_{0,2k}^{(n)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(n)} & \dots & u_{2k-1,2k}^{(n)} \end{vmatrix}}, \tag{30}$$

where

$$\begin{cases} u_{ij}^{(n)} = \sum_{l=0}^{j-i-1} (\Delta s_{i+n+l}, \Delta s_{j+n-l-1}), & j > i, \\ u_{ij}^{(n)} = -u_{ji}^{(n)}, & j < i, \\ u_{ii}^{(n)} = 0. \end{cases} \tag{31}$$

Proof. Let us take $c_0 = s_0$ and $c_n = \Delta s_{n-1}$ for $n > 0$ in the series (25). Thus, we have

$$\varepsilon_0^{(j)} = s_j = \sum_{i=0}^{j} c_i. \tag{32}$$

The accuracy-through-order condition (24), implies that

$$p^{[n+2k/2k]}(x) = \left[q^{[n+2k/2k]}(x) f(x)\right]_0^{n+2k}$$

using the truncation notation $[\phi(x)]_i^j$ [22], in which terms of $\phi(x)$ of powers between i and j inclusively are retained and all others are discarded. So, using (26), we obtain

$$p^{[n+2k/2k]}(x) = \begin{vmatrix} \left[x^{2k} f(x)\right]_0^{n+2k} & \left[x^{2k-1} f(x)\right]_0^{n+2k} & \dots & \left[f(x)\right]_0^{n+2k} \\ \mu_{00}^{(n+2k)} & \mu_{01}^{(n+2k)} & \dots & \mu_{0,2k}^{(n+2k)} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{2k-1,0}^{(n+2k)} & \mu_{2k-1,1}^{(n+2k)} & \dots & \mu_{2k-1,2k}^{(n+2k)} \end{vmatrix}.$$

It is obvious that $[x^j f(x)]_0^{n+2k}|_{x=1}$ is equal to $\sum_{l=0}^{n+2k-j} c_l = s_{n+2k-j}$. Since $c_n = \Delta s_{n-1}$, we have $\mu_{i,j}^{(n+2k)} = u_{ij}^{(n)}$ and from $\varepsilon_{2k}^{(n)} = r^{[n+2k/2k]}(1)$ we obtain (30). □

Note that like Sonneveld's and Van der Vorst's methods [36,37], the VEA is a barycentric combination of $x_n, \dots, x_{n+2k}$ while MPE, RRE, MMPE and TEA are barycentric combinations of only $s_n, \dots, s_{n+k}$ and that the $u_{ij}^{(n)}$'s of the VEA are not linear in the data.

4.2. *VEA as a Krylov subspace method*

We use the definition of [27] for a Krylov subspace method. We recall that a general projection method for solving the linear system

$$Az = b,$$

is a method which seeks an approximate solution z_m from an affine subspace $z_0 + \mathcal{K}_m$ of dimension m by imposing the Petrov–Galerkin condition

$$b - Az_m \perp \mathcal{L}_m,$$

where $\mathcal{L}_m$ is another subspace of dimension m. Here z_0 represents an arbitrary initial guess to the solution. A Krylov subspace method is a method for which the subspace $\mathcal{K}_m$ is the Krylov subspace

$$\mathcal{K}_m = \operatorname{span}\{r_0, Ar_0, \dots, A^{m-1}r_0\},$$

where $r_0 = b - Az_0$.

Let $X = (x_j)$ denote the vector sequence generated by (16), let $r(x) = b-(I-A)x$ be the residual, and let $r_0 = r(x_0)$ and $\varepsilon_k^{(n)}$ be the quantities computed by the VEA applied to the sequence X. We define

$$V_k = [r_0, \dots, A^{2k}r_0] \in \mathbb{R}^{d\times(2k+1)}, \qquad U_k = [u_{ij}^{(0)}] \in \mathbb{R}^{2k\times(2k+1)},$$
$$W_k = U_k\big(\big(V_k^{\mathrm T}\big)V_k\big)^{-1}V_k^{\mathrm T} \in \mathbb{R}^{2k\times d},$$

where $u_{ij}^{(n)}$ are as in (31) by taking x_j instead of s_j for $i = 0, \dots, 2k-1$, $j = 0, \dots, 2k$. Let $\mathcal{P}_{2k}$ be the polynomial

$$\mathcal{P}_{2k}(x) = \frac{\begin{vmatrix} 1 & \dots & x^{2k} \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}. \tag{33}$$

Then we have:

Theorem 2. The VEA, when applied to the vector fixed point $X = (x_j)$, is a Krylov subspace method. More precisely, define $z_k^{\mathrm{VEA}} = \varepsilon_{2k}^{(0)}$, $r_k^{\mathrm{VEA}} = r(z_k^{\mathrm{VEA}})$, then

$$z_k^{\mathrm{VEA}} \in x_0 + \mathcal{K}_{2k}(A, r_0), \tag{34}$$

$$\forall x \in \mathbb{R}, \quad \mathcal{P}_{2k}(x) \geqslant 0, \tag{35}$$

$$r_k^{\mathrm{VEA}} = \mathcal{P}_{2k}(A) r_0, \tag{36}$$

and

$$r_k^{\mathrm{VEA}} \perp \mathrm{Im}\{W_k^{\mathrm{T}}\}, \tag{37}$$

where $\mathrm{Im}\{W_k^{\mathrm{T}}\}$ denotes the vector space of linear combinations of the $2k$ columns of W_k^{T}.

Proof. From the determinantal formula (30) of the VEA, we get

$$z_k^{\mathrm{VEA}} = x_0 + \frac{\begin{vmatrix} 0 & x_1 - x_0 & \dots & x_{2k} - x_0 \\ u_{00}^{(0)} & u_{01}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & u_{2k-1,1}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}.$$

Then, since $\Delta x_n = A^n r_0$ and $x_n - x_0 = \sum_{i=0}^{n-1} \Delta x_i$, we obtain $x_i - x_0 \in \mathcal{K}_i(A, r_0)$. From the same determinantal formula of z_k^{VEA}, we obtain

$$r_k^{\mathrm{VEA}} = b - (I - A) z_k = \frac{\begin{vmatrix} b - (I - A)x_0 & \dots & b - (I - A)x_{2k} \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}.$$

Since $b - (I - A)x_i = \Delta x_i = A^i r_0$, (36) is immediate. Writing $\mathcal{P}_{2k}(x) = \sum_{j=0}^{2k} \gamma_j^{(0,k)} x^j$ and setting $\Gamma_k^{\mathrm{T}} = (\gamma_0^{(0,k)}, \dots, \gamma_{2k}^{(0,k)})$, the residual can be written

$$r_k^{\mathrm{VEA}} = \mathcal{P}_{2k}(A) r_0 = \sum_{j=0}^{2k} \gamma_j^{(0,k)} A^j r_0 = V_k \Gamma_k.$$

From the determinantal formula, the $\gamma_j^{(0,k)}$ satisfy the constraints

$$\sum_{j=0}^{2k} u_{ij}^{(0)} \gamma_j^{(0,k)} = 0, \quad i = 0, \dots, 2k-1.$$

i.e.,

$$U_k \Gamma_k = 0. \tag{38}$$

If $(V_k^{\mathrm{T}})V_k$ is nonsingular, then U_k can be written

$$U_k = W_k V_k, \quad \text{with } W_k = U_k\big((V_k^{\mathrm{T}})V_k\big)^{-1} V_k^{\mathrm{T}}, \tag{39}$$

and then we have

$$U_k \Gamma_k = 0 \quad \Longleftrightarrow \quad W_k V_k \Gamma_k = 0 \quad \Longleftrightarrow \quad W_k r_k^{\mathrm{VEA}} = 0, \tag{40}$$

which means that r_k^{VEA} is orthogonal to the linear subspace $\mathrm{Im}\{W_k^{\mathrm{T}}\}$. The result that $\mathcal{P}_{2k}(x) \geqslant 0$, $\forall x \in \mathbb{R}$ follows from (23) using proof by induction [17]. □

That $\mathcal{P}_{2k} \geqslant 0$ and $r_k^{\mathrm{VEA}} = \mathcal{P}_{2k}(A)r_0$ explain why we are led to compare VEA with the CGS. The open question now is to know if it coincides with a known Krylov subspace method similar to the four transformations above or if it is a new one.

Notice that the corresponding matrix W_k would be V_k^{T} for MPE, $(AW_k)^{\mathrm{T}}$ for RRE, $[q_0, \dots, q_k]$ for MMPE and $[y, A^{\mathrm{T}}y, \dots, (A^{\mathrm{T}})^k y]$ for TEA [14,33]. For the VEA, we do not have an expression other than (39) for W_k.

5. Links between VEA and known Krylov subspace methods

We showed above that the VEA applied to a vector fixed point sequence is a Krylov subspace method. It seems to be a new one since it is different from the classical methods which have determinantal expressions. Even with a determinantal formula available, it is still hard to establish links between the VEA and other known Krylov subspace methods. To begin, we consider two classes of matrices: symmetric and skew-symmetric matrices.

Theorem 3. If the matrix A is symmetric, then $r_k^{\mathrm{VEA}} = \mathcal{P}_{2k}(A)r_0$, with $u_{ij}^{(0)} = (j-i) \times (r_0, A^{i+j-1}r_0)$ in (33), while in the Lanczos method, $r_{2k}^{\mathrm{Lan}} = \mathcal{P}_{2k}(A)r_0$, with $u_{ij}^{(0)} = (r_0, A^{i+j}r_0)$ in (19).

Proof. We have

$$\begin{cases} u_{ij}^{(0)} = \displaystyle\sum_{l=0}^{j-i-1} (\Delta x_{i+l}, \Delta x_{j-l-1}) = \sum_{l=0}^{j-i-1} \big(A^{i+l}r_0, A^{j-l-1}r_0\big), & j > i, \\ u_{ij}^{(0)} = -u_{ji}^{(0)}, & j < i, \\ u_{ii}^{(0)} = 0. & \end{cases}$$

Since A is symmetric, we obtain

$$u_{ij}^{(0)} = \sum_{l=0}^{j-i-1} \big(r_0, A^{j+i-1} r_0\big) = (j-i)\big(r_0, A^{j+i-1} r_0\big). \qquad \square$$

A more interesting result is for the case in which A is skew-symmetric. The following theorem establishs a link between the VEA for solving linear systems and a known Krylov subspace method.

Theorem 4. If A is skew-symmetric, then

$$z_k^{\mathrm{VEA}} = z_{2k}^{\mathrm{Lan}}.$$

Proof. When A is skew-symmetric, it is easy to see that

$$u_{ij}^{(0)} = \frac{(-1)^i - (-1)^j}{2}\big(r_0, A^{i+j-1} r_0\big) \quad \text{for } i = 0, 1, \ldots, 2k-1,\ j = 0, 1, \ldots, 2k.$$

Thus for i, j of the same parity, we have

$$u_{ij}^{(0)} = 0.$$

For i odd and j even, we have

$$u_{ij}^{(0)} = -\big(r_0, A^{i+j-1} r_0\big).$$

For i even and j odd, we have

$$u_{ij}^{(0)} = \big(r_0, A^{i+j-1} r_0\big).$$

For simplicity, we take the example $k = 2$ from which it will be clear that the result is true in general.

The coefficients $\gamma_i^{(0,2)}$, $i = 0, 1, \ldots, 4$, of the polynomial $\mathcal{P}_4$ are given by

$$\sum_{i=0}^{4} \gamma_i^{(0,2)} = 1, \tag{41}$$

$$U_2 \Gamma_2 = 0, \tag{42}$$

where U_2 is the quasi-skew-symmetric matrix

$$U_2 = \begin{pmatrix} 0 & (r_0, r_0) & 0 & \big(r_0, A^2 r_0\big) & 0 \\ -(r_0, r_0) & 0 & -\big(r_0, A^2 r_0\big) & 0 & -\big(r_0, A^4 r_0\big) \\ 0 & \big(r_0, A^2 r_0\big) & 0 & \big(r_0, A^4 r_0\big) & 0 \\ -\big(r_0, A^2 r_0\big) & 0 & -\big(r_0, A^4 r_0\big) & 0 & -\big(r_0, A^6 r_0\big) \end{pmatrix}.$$

From the structure of U_2, system (41)–(42) can be decomposed into two subsystems.

The first subsystem corresponds to the even rows of (42),

$$\begin{pmatrix} (r_0, r_0) & (r_0, A^2 r_0) \\ (r_0, A^2 r_0) & (r_0, A^4 r_0) \end{pmatrix} \begin{pmatrix} \gamma_1^{(0,2)} \\ \gamma_3^{(0,2)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{43}$$

The second subsystem corresponds to the odd rows of (42) and (41),

$$\begin{pmatrix} 1 & 1 & 1 \\ (r_0, r_0) & (r_0, A^2 r_0) & (r_0, A^4 r_0) \\ (r_0, A^2 r_0) & (r_0, A^4 r_0) & (r_0, A^6 r_0) \end{pmatrix} \begin{pmatrix} \gamma_0^{(0,2)} \\ \gamma_2^{(0,2)} \\ \gamma_4^{(0,2)} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{44}$$

If the matrix of the subsystem (43) is nonsingular (that is if the matrix $[r_0, Ar_0]$ is full rank) then $\gamma_1^{(0,2)} = \gamma_3^{(0,2)} = 0$, and solving (41)–(42) becomes equivalent to solving the subsystem (44).

Since $r_2^{\mathrm{VEA}} = \mathcal{P}_4(A) r_0$, we obtain

$$r_2^{\mathrm{VEA}} = \frac{\begin{vmatrix} r_0 & A^2(A) r_0 & A^4(A) r_0 \\ (r_0, r_0) & (r_0, A^2 r_0) & (r_0, A^4 r_0) \\ (r_0, A^2 r_0) & (r_0, A^4 r_0) & (r_0, A^6 r_0) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 \\ (r_0, r_0) & (r_0, A^2 r_0) & (r_0, A^4 r_0) \\ (r_0, A^2 r_0) & (r_0, A^4 r_0) & (r_0, A^6 r_0) \end{vmatrix}}. \tag{45}$$

We compare this with z_4^{Lan}. From (19), (20), its residual is given as

$$r_4^{\mathrm{Lan}} = \frac{\begin{vmatrix} r_0 & Ar_0 & A^2 r_0 & A^3 r_0 & A^4 r_0 \\ (r_0, r_0) & 0 & (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) \\ 0 & (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) & 0 \\ (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) & 0 & (r_0, A^6 r_0) \\ 0 & (r_0, A^4 r_0) & 0 & (r_0, A^6 r_0) & 0 \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 & 1 \\ (r_0, r_0) & 0 & (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) \\ 0 & (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) & 0 \\ (r_0, A^2 r_0) & 0 & (r_0, A^4 r_0) & 0 & (r_0, A^6 r_0) \\ 0 & (r_0, A^4 r_0) & 0 & (r_0, A^6 r_0) & 0 \end{vmatrix}}.$$

By permuting columns and rows of the numerator and denominator, the expression can be written as

$$r_4^{\mathrm{Lan}} = \frac{\begin{vmatrix} r_0 & A^2r_0 & A^4r_0 & Ar_0 & A^3r_0 \\ (r_0, r_0) & (r_0, A^2r_0) & (r_0, A^4r_0) & 0 & 0 \\ (r_0, A^2r_0) & (r_0, A^4r_0) & (r_0, A^6r_0) & 0 & 0 \\ 0 & 0 & 0 & (r_0, A^2r_0) & (r_0, A^4r_0) \\ 0 & 0 & 0 & (r_0, A^4r_0) & (r_0, A^6r_0) \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 & 1 & 1 \\ (r_0, r_0) & (r_0, A^2r_0) & (r_0, A^4r_0) & 0 & 0 \\ (r_0, A^2r_0) & (r_0, A^4r_0) & (r_0, A^6r_0) & 0 & 0 \\ 0 & 0 & 0 & (r_0, A^2r_0) & (r_0, A^4r_0) \\ 0 & 0 & 0 & (r_0, A^4r_0) & (r_0, A^6r_0) \end{vmatrix}}.$$

Thus, this expression is reduced to (45) by using the classical rule of product of determinant.

To our knowledge, this is the first successful result establishing a direct link between the VEA for solving linear systems and known Krylov subspace methods. This result can be viewed as a Lanczos method for a precondioned system. □

Corollary 1. Let x_0^{VEA} be the starting guess for the VEA method for solving the system $(I - A)x = b$, and $x_0^{P\mathrm{Lan}}$ be the starting guess of Lanczos method ($y = r_0^{P\mathrm{Lan}} = C(b - (I - A)x_0^{P\mathrm{Lan}})$) for the preconditioned system

$$C(I - A)x = Cb, \quad \text{with } C = (I + A) \simeq (I - A)^{-1},$$

such that $r_0^{\mathrm{VEA}} = r_0^{P\mathrm{Lan}}$. If A is skew-symmetric, then the two methods are equivalent.

Proof. Since $x_{j+1} = Ax_j + b$, we have

$$x_{2j+2} = A^2x_{2j} + (A + I)b,$$

which is the vector fixed point sequence associated to the system

$$x = A^2x + (A + I)b. \tag{46}$$

Let $r_2^{P\mathrm{Lan}}$ denote the residual of the Lanczos method ($y = r_0^{P\mathrm{Lan}}$) applied to the system $x = A^2x + (A + I)b$. From its determinantal formula (19), and comparing with (45), we get $r_2^{P\mathrm{Lan}} = r_2^{\mathrm{VEA}}$. Since

$$x = A^2x + (A + I)b \quad \Longleftrightarrow \quad (I + A)(I - A)x = (I + A)b,$$

we get the result. □

We shall now give an algebraic comparison with the CGS. This comparison shows that the two algorithms can be written in a unified form.

6. Algebraic comparison with the CGS

Let (p_k) the Lanczos polynomials (19). They are orthogonal polynomials with respect to the linear functional c.

Using the theory of vector orthogonal polynomials, we have the following:

Theorem 5.

$$p_k^2(x) = \frac{\begin{vmatrix} 1 & \dots & x^{2k} \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}, \tag{47}$$

where

$$\begin{cases} u_{ij}^{(0)} = \sum_{l=0}^{j-i-1} c_{i+l}c_{j-l-1}, & j > i, \\ u_{ij}^{(0)} = -u_{ji}^{(0)}, & j < i, \\ u_{ii}^{(0)} = 0. \end{cases} \tag{48}$$

For a succinct proof, see [30].

Example. For $k = 1$, we have

$$p_1^2(x) = \left(\frac{\begin{vmatrix} 1 & x \\ c_0 & c_1 \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ c_0 & c_1 \end{vmatrix}} \right)^2 = \frac{\begin{vmatrix} 1 & x & x^2 \\ 0 & c_0^2 & 2c_0c_1 \\ -c_0^2 & 0 & c_1^2 \end{vmatrix}}{\begin{vmatrix} 1 & 1 & 1 \\ 0 & c_0^2 & 2c_0c_1 \\ -c_0^2 & 0 & c_1^2 \end{vmatrix}}.$$

The CGS [36] algorithm is the algorithm which produces the residual $r_k^{(\mathrm{CGS})} = p_k^2(A)r_0$. In [6,7,9] Brezinski et al. gave a determinantal formula of the residual and proposed several methods to cure breakdown and near-breakdown. We use the theorem above for giving another determinantal formula for the residual of the CGS which allows an algebraic comparison with the VEA for linear systems in corollaries 2, 3, which follow immediately from theorem 5.

Corollary 2. We have

$$r_k^{(CGS)} = \frac{\begin{vmatrix} r_0 & \dots & A^{2k} r_0 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}, \tag{49}$$

where

$$\begin{cases} u_{ij}^{(0)} = \displaystyle\sum_{l=0}^{j-i-1} c_{i+l} c_{j-l-1}, & j > i, \\ u_{ij}^{(0)} = -u_{ji}^{(0)}, & j < i, \\ u_{ii}^{(0)} = 0. \end{cases} \tag{50}$$

Thus, the residual of the CGS and the VEA for linear systems can be written in the same framework. More precisely, we have:

Corollary 3. Let $M = (m_i)$ denote a sequence of scalars, as in (53) (or real vectors as in (54)). Using them, we define the polynomial $q_{2k}(x)$ by

$$q_{2k}(x) = \frac{\begin{vmatrix} 1 & \dots & x^{2k} \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ u_{00}^{(0)} & \dots & u_{0,2k}^{(0)} \\ \vdots & & \vdots \\ u_{2k-1,0}^{(0)} & \dots & u_{2k-1,2k}^{(0)} \end{vmatrix}}, \tag{51}$$

where

$$\begin{cases} u_{ij}^{(0)} = \displaystyle\sum_{l=0}^{j-i-1} (m_{i+l})^{\mathrm{T}} m_{j-l-1}, & j > i, \\ u_{ij}^{(0)} = -u_{ji}^{(0)}, & j < i, \\ u_{ii}^{(0)} = 0. \end{cases} \tag{52}$$

Then

$$r_k^{(CGS)} = q_{2k}(A)r_0, \quad \text{with } m_i = \left(r_0, A^i r_0\right), \tag{53}$$

$$r_k^{(\mathrm{VEA})} = q_{2k}(A)r_0, \quad \text{with } m_i = A^i r_0. \tag{54}$$

References

[1] W.E. Arnoldi, The principle of minimized iterations in the solution of the matrix eigenvalue problem, Quart. Appl. Math. 9 (1951) 17–29.
[2] O. Axelsson, Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations, Linear Algebra Appl. 29 (1980) 1–16.
[3] C. Brezinski, Some results in the theory of the vector ε-algorithm, Linear Algebra Appl. 8 (1974) 77–86.
[4] C. Brezinski, Généralisations de la transformation de Shanks, de la table de Padé et de l'ε-algorithme, Calcolo 12 (1975) 317–360.
[5] C. Brezinski, *Padé-Type Approximation and General Orthogonal Polynomials* (Birkhäuser, Basel, 1980).
[6] C. Brezinski and M. Redivo-Zaglia, Treatment of near-breakdown in the CGS algorithms, Numer. Algorithms 7 (1994) 33–73.
[7] C. Brezinski and M. Redivo-Zaglia, Look-ahead in BiCGSTAB and other product-type methods for linear systems, BIT 35 (1995) 169–201.
[8] C. Brezinski and M. Redivo-Zaglia, *Extrapolation Methods. Theory and Practice* (North-Holland, Amsterdam, 1996).
[9] C. Brezinski and H. Sadok, Avoiding breakdown in the CGS algorithm, Numer. Algorithms 1 (1991) 199–206.
[10] C. Brezinski and H. Sadok, Lanczos-type algorithms for solving systems of linear equations, Appl. Numer. Math. 11 (1993) 443–473.
[11] S. Cabay and L.W. Jackson, A polynomial extrapolation method for finding limits and antilimits of vector sequences, SIAM J. Numer. Anal. 13 (1976) 734–752.
[12] R.P. Eddy, Extrapolating to the limit of a vector sequence, in: *Information Linkage between Applied Mathematics and Industry*, ed. P.C.C. Wang (Academic Press, New York, 1979) pp. 387–396.
[13] S.C. Eisenstat, H.C. Elman and M.H. Schultz, Variational iterative methods for nonsymmetric systems of linear equations, SIAM J. Numer. Anal. 20 (1983) 345–357.
[14] W. Gander, G.H. Golub and D. Gruntz, Solving linear equations by extrapolation, in: *Supercomputing*, Trondheim, 1989, Computer Systems Science, Vol. 62 (Springer, Berlin, 1989) pp. 279–293.
[15] P.R. Graves-Morris, Vector valued rational interpolants I, Numer. Math. 42 (1983) 331–348.
[16] P.R. Graves-Morris, Vector valued rational interpolants II, IMA J. Numer. Anal. 4 (1984) 209–224.
[17] P.R. Graves-Morris, G.A. Baker Jr. and C.F. Woodcock, Cayley's theorem and its application in the theory of vector Padé approximants, J. Comput. Appl. Math. 66 (1996) 255–265.
[18] P.R. Graves-Morris and C.D. Jenkins, Vector-valued, rational interpolants III, Constr. Approx. 2 (1986) 263–289.
[19] C. Lanczos, Solution of systems of linear equations by minimized iteration, J. Res. N.B.S. 49 (1952) 33–53.
[20] J.B. McLeod, A note on the ε-algorithm, Computing 7 (1971) 17–24.
[21] M. Mešina, Convergence acceleration for the iterative solution of the equations $X = AX + f$, Comput. Methods Appl. Mech. Engrg. 10 (1977) 165–173.
[22] J. Nuttall, Convergence of Padé approximants of meromorphic functions, J. Math. Anal. Appl. 31 (1970) 147–153.

[23] D.E. Roberts, Clifford algebras and vector-valued rational forms I, Proc. Roy. Soc. London A 431 (1990) 285–300.

[24] D.E. Roberts, Clifford algebras and vector-valued rational forms II, Numer. Algorithms 3 (1992) 371–381.

[25] D.E. Roberts, On a q-d algorithm, Adv. Comput. Math. 8 (1998) 193–219.

[26] Y. Saad, Krylov subspace methods for solving large unsymmetric linear systems, Math. Comp. 37 (1981) 105–126.

[27] Y. Saad, *Iterative Methods for Sparse Linear Systems* (PWS, Boston, 1996).

[28] A. Salam, Formal vector orthogonal polynomials, Adv. Comput. Math. 8 (1988) 267–289.

[29] A. Salam, Vector Padé-type approximants and vector Padé approximants, J. Approx. Theory 97 (1999) 92–112.

[30] A. Salam, On vector Hankel determinant, Linear Algebra Appl. 313 (2000) 127–139.

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

[32] A. Sidi, Convergence and stability properties of minimal polynomial and reduced rank extrapolation algorithms, SIAM J. Numer. Anal. 23 (1986) 197–209.

[33] A. Sidi, Extrapolation vs. projection methods for linear systems of equations, J. Comput. Appl. Math. 22 (1988) 71–88.

[34] A. Sidi, W.F. Ford and D.A. Smith, Acceleration of convergence of vectors sequences, SIAM J. Numer. Anal. 23 (1986) 178–196.

[35] D.A. Smith, W.F. Ford and A. Sidi, Extrapolation methods for vector sequences, SIAM J. Numer Anal. 23 (1986) 178–196.

[36] P. Sonneveld, CGS, A fast Lanczos-type solver for nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 10 (1989) 35–52.

[37] H.A. Van Der Vorst, BiCGSTAB: A fast and smoothly converging variant of the Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 13 (1992) 631–644.

[38] P. Wynn, On a device for computing the $e_m(S_n)$ transformation, Math. Tables Autom. Comp. 10 (1956) 91–96.

[39] P. Wynn, Acceleration techniques for iterated vector and matrix problems, Math. Comput. 16 (1962) 301–322.

[40] P. Wynn, Vector continued fractions, Linear Algebra Appl. 1 (1968) 357–395.