Applications of the Simplified Topological ε–algorithms

Michela Redivo–Zaglia [1] **Claude Brezinski** [2]

[1]Dipartimento di Matematica, Università degli Studi di Padova, Italy.

[2]Laboratoire Paul Painlevé, Université de Lille 1 - Sciences et Technologies, France

Numerical Linear Algebra and Applications (NL2A)
October 24-28, 2016

Shanks transformation (1955) is a well–known sequence transformation for accelerating the convergence of sequences of numbers. It can be recursively implemented by the **scalar ε–algorithm** of **Wynn** (1956).

They were both extended to sequences of elements of a vector space by **Brezinski** (1975).

The idea is as follows.

Let $(\mathbf{S}_n)$ be a sequence of elements of a vector space E converging to $\mathbf{S}$, and assume that it satisfies, for a fixed value of k, the difference equation

$$a_0(\mathbf{S}_n - \mathbf{S}) + \cdots + a_k(\mathbf{S}_{n+k} - \mathbf{S}) = \mathbf{0} \in E, \quad n = 0, 1, \ldots$$

with $a_0 a_k \neq 0$ and $a_0 + \cdots + a_k \neq 0$.

We want to transform $(\mathbf{S}_n)$ into a new sequence $(\mathbf{e}_k(\mathbf{S}_n))$ such that

$$\mathbf{e}_k(\mathbf{S}_n) = \mathbf{S}, \quad \text{for all } n.$$

If $(\mathbf{S}_n)$ satisfies the difference equation above then $\mathbf{S}$ is given by

$$\mathbf{S} = (a_0\mathbf{S}_n + \cdots + a_k\mathbf{S}_{n+k})/(a_0 + \cdots + a_k), \quad \forall n.$$

This linear combination can be computed even if $(\mathbf{S}_n)$ does not satisfy the difference equation, thus defining a sequence transformation.

We now have to compute the coefficients $a_0, \ldots, a_k$ (now depending on k and n).

It holds, $\forall n$,

$$a_0 \Delta \mathbf{S}_n + \cdots + a_k \Delta \mathbf{S}_{n+k} = \mathbf{0}.$$

We need to transform this relation in E into a system of scalar relations.

Let $\mathbf{y}$ be an element of the dual space E^* of E (which means that it is a linear functional).
Taking the duality product of this relation with $\mathbf{y}$, we have, $\forall n$,

$$a_0\langle \mathbf{y}, \Delta \mathbf{S}_n\rangle + \cdots + a_k\langle \mathbf{y}, \Delta \mathbf{S}_{n+k}\rangle = 0.$$

Writing this relation for the indices $n, \ldots, n+k-1$ and adding that $a_0 + \cdots + a_k = 1$ (which does not restrict the generality), leads to the system

$$\begin{cases} a_0 + \cdots + a_k = 1 \\ a_0\langle \mathbf{y}, \Delta \mathbf{S}_{n+i}\rangle + \cdots + a_k\langle \mathbf{y}, \Delta \mathbf{S}_{n+k+i}\rangle = 0, \quad i = 0, \ldots, n+k-1, \end{cases}$$

where the unknowns depend on n (and k) if $(\mathbf{S}_n)$ does not satisfy the difference equation.

The **first topological Shanks transformation** is then defined by

$$\mathbf{e}_k(\mathbf{S}_n) = a_0\mathbf{S}_n + \cdots + a_k\mathbf{S}_{n+k},$$

and the **second topological Shanks transformation** by

$$\tilde{\mathbf{e}}_{\mathbf{k}}(\mathbf{S}_n) = a_0\mathbf{S}_{n+k} + \cdots + a_k\mathbf{S}_{n+2k}.$$

By construction, $\forall n, \mathbf{e}_k(\mathbf{S}_n) = \tilde{\mathbf{e}}_{\mathbf{k}}(\mathbf{S}_n) = \mathbf{S}$ if

$$\forall n, a_0(\mathbf{S}_n - \mathbf{S}) + \cdots + a_k(\mathbf{S}_{n+k} - \mathbf{S}) = \mathbf{0}.$$

Each of these transformations can be implemented by a recursive algorithm which generalizes the scalar ε–algorithm of Wynn **(Brezinski, 1975)**. They are named the
Topological ε–algorithms (**TEA**).
However, these algorithms are **quite complicated**:

- they possess two rules,
- they require the storage of elements of E and of E^*,
- the duality product with $\mathbf{y}$ is recursively used in their rules.

Each of these transformations can be implemented by a recursive algorithm which generalizes the scalar ε–algorithm of Wynn **(Brezinski, 1975)**. They are named the **Topological ε–algorithms** (**TEA**).

However, these algorithms are **quite complicated**:

- they possess two rules,
- they require the storage of elements of E and of E^*,
- the duality product with $\mathbf{y}$ is recursively used in their rules.

Recently, simplified versions of these algorithms were obtained **(Brezinski and R.-Z., 2014)**, and called the **Simplified Topological ε–algorithms** (**STEA**).

- only one recursive rule,
- they require less storage than the initial algorithms and only elements of E,
- the elements of the dual vector space E^* no longer have to be used in the recursive rules (only in their initializations) ,
- numerical stability is improved (thanks to particular rules of Wynn).

The rule of the **First Simplified Topological ε–algorithm**, denoted by **STEA1**, is

$$\boldsymbol{\varepsilon}_{2k+2}^{(n)} = \boldsymbol{\varepsilon}_{2k}^{(n+1)} + \frac{\varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n+1)}}{\varepsilon_{2k}^{(n+1)} - \varepsilon_{2k}^{(n)}}(\boldsymbol{\varepsilon}_{2k}^{(n+1)} - \boldsymbol{\varepsilon}_{2k}^{(n)}), \quad k, n = 0, 1, \ldots,$$

with $\boldsymbol{\varepsilon}_0^{(n)} = \mathbf{S}_n \in E,\ n = 0, 1, \ldots$

The scalars $\varepsilon_{2k}^{(n)}$ are computed by the **scalar ε–algorithm** of Wynn whose rule is

$$\begin{cases} \varepsilon_{-1}^{(n)} & = & 0, \quad n = 0, 1, \ldots, \\ \varepsilon_0^{(n)} & = & \langle \mathbf{y}, \mathbf{S}_n \rangle, \quad n = 0, 1, \ldots, \\ \varepsilon_{k+1}^{(n)} & = & \varepsilon_{k-1}^{(n+1)} + (\varepsilon_k^{(n+1)} - \varepsilon_k^{(n)})^{-1}, \quad k, n = 0, 1, \ldots \end{cases}$$

The rule of the **Second Simplified Topological ε–algorithm**, denoted by **STEA2**, is

$$\widetilde{\varepsilon}_{2k+2}^{(n)} = \widetilde{\varepsilon}_{2k}^{(n+1)} + \frac{\varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n+1)}}{\varepsilon_{2k}^{(n+2)} - \varepsilon_{2k}^{(n+1)}} (\widetilde{\varepsilon}_{2k}^{(n+2)} - \widetilde{\varepsilon}_{2k}^{(n+1)}), \quad k, n = 0, 1, \ldots,$$

with $\widetilde{\varepsilon}_0^{(n)} = \mathbf{S}_n \in E$, $n = 0, 1, \ldots$, and the numbers $\varepsilon_{2k}^{(n)}$ are the same as above.

The rule of the **Second Simplified Topological ε–algorithm**, denoted by **STEA2**, is

$$\widetilde{\varepsilon}_{2k+2}^{(n)} = \widetilde{\varepsilon}_{2k}^{(n+1)} + \frac{\varepsilon_{2k+2}^{(n)} - \varepsilon_{2k}^{(n+1)}}{\varepsilon_{2k}^{(n+2)} - \varepsilon_{2k}^{(n+1)}} (\widetilde{\varepsilon}_{2k}^{(n+2)} - \widetilde{\varepsilon}_{2k}^{(n+1)}), \quad k, n = 0, 1, \ldots,$$

with $\widetilde{\varepsilon}_0^{(n)} = \mathbf{S}_n \in E$, $n = 0, 1, \ldots$, and the numbers $\varepsilon_{2k}^{(n)}$ are the same as above.

These algoritms are related to the topological Shanks transformation by

$$\varepsilon_{2k}^{(n)} = \mathbf{e}_k(\mathbf{S}_n) \quad \text{and} \quad \widetilde{\varepsilon}_{2k}^{(n)} = \widetilde{\mathbf{e}}_{\mathbf{k}}(\mathbf{S}_n).$$

Remark 1: These simplified algorithms allow to prove easily convergence and acceleration results.
Remark 2: For STEA1 there exist **four different equivalent formulas**. The same for STEA2.

The ε-array

$$
\begin{array}{cccccc}
\varepsilon_{-1}^{(0)} = 0 & & & & & \\
 & \color{red}{\varepsilon_0^{(0)}} & & & & \\
\varepsilon_{-1}^{(1)} = 0 & & \varepsilon_1^{(0)} & & & \\
 & \color{red}{\varepsilon_0^{(1)}} & & \varepsilon_2^{(0)} & & \\
\varepsilon_{-1}^{(2)} = 0 & & \varepsilon_1^{(1)} & & \varepsilon_3^{(0)} & \\
 & \color{red}{\varepsilon_0^{(2)}} & & \varepsilon_2^{(1)} & & \varepsilon_4^{(0)} \\
\varepsilon_{-1}^{(3)} = 0 & & \varepsilon_1^{(2)} & & \varepsilon_3^{(1)} & \vdots \\
 & \color{red}{\varepsilon_0^{(3)}} & & \varepsilon_2^{(2)} & \vdots & \\
\varepsilon_{-1}^{(3)} = 0 & & \varepsilon_1^{(3)} & \vdots & & \\
 & \color{red}{\varepsilon_0^{(4)}} & \vdots & & & \\
\varepsilon_{-1}^{(4)} = 0 & \vdots & & & & \\
\vdots & & & & &
\end{array}
$$

TEA1 vs STEA1

Odd rule TEA1 | Even rule TEA1 | STEA1

$$\begin{tikzcd}[row sep=small, column sep=small]
 & \varepsilon_{2k}^{(n)} \arrow[dr] & \\
\varepsilon_{2k-1}^{(n+1)} \arrow[ur] \arrow[dr] & & \boxed{\varepsilon_{2k+1}^{(n)}} \\
 & \varepsilon_{2k}^{(n+1)} \arrow[ur] &
\end{tikzcd}
\qquad
\begin{tikzcd}[row sep=small, column sep=small]
\varepsilon_{2k}^{(n)} \arrow[dr] & & \\
 & \varepsilon_{2k+1}^{(n)} \arrow[dr] & \\
\varepsilon_{2k}^{(n+1)} \arrow[ur] \arrow[dr] & & \boxed{\varepsilon_{2k+2}^{(n)}} \\
 & \varepsilon_{2k+1}^{(n+1)} \arrow[ur] &
\end{tikzcd}
\qquad
\begin{tikzcd}[row sep=small, column sep=small]
\varepsilon_{2k}^{(n)} \arrow[dr] & \\
\varepsilon_{2k}^{(n+1)} \arrow[r] & \boxed{\varepsilon_{2k+2}^{(n)}}
\end{tikzcd}$$

Similar for STEA2.

Exploitation of the algorithms

The algorithms will be used in two different ways

- **Acceleration method (AM)**
- **Restarted method (RM)**

Exploitation of the algorithms

The algorithms will be used in two different ways

- **Acceleration method (AM)**
- **Restarted method (RM)**

The **Acceleration Method (AM)** can be applied to the solution of systems of linear and nonlinear equations, to the computation of matrix functions, ...

$$
\left\{
\begin{array}{l}
\text{Choose } 2k \text{ and } \mathbf{x}_0. \\
\text{For } n = 1, 2, \ldots \\
\quad \text{Compute } \mathbf{x}_n. \\
\quad \text{Apply the STEA1 to } \mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots \text{ and compute the sequence} \\
\quad \text{of extrapolated values} \\
\quad \varepsilon_0^{(0)} = \mathbf{x}_0, \varepsilon_0^{(1)}, \varepsilon_2^{(0)}, \varepsilon_2^{(1)}, \ldots, \varepsilon_{2k}^{(0)}, \varepsilon_{2k}^{(1)}, \varepsilon_{2k}^{(2)}, \ldots \\
\quad \text{or similar quantities by the STEA2.} \\
\text{end}
\end{array}
\right.
$$

The **Restarted Method (RM)** is used for **fixed point problems**, that is for solving systems of linear and nonlinear equations.
Let $F : \mathbb{R}^m \longmapsto \mathbb{R}^m$. We have to find the solution of equations of the form

$$\mathbf{x} = F(\mathbf{x}) \quad \text{or} \quad f(\mathbf{x}) = F(\mathbf{x}) - \mathbf{x} = \mathbf{0} \quad \text{or} \quad \mathbf{x} = \mathbf{x} + \alpha f(\mathbf{x}).$$

The **Restarted Method (RM)** is used for **fixed point problems**, that is for solving systems of linear and nonlinear equations.
Let $F : \mathbb{R}^m \longmapsto \mathbb{R}^m$. We have to find the solution of equations of the form

$$\mathbf{x} = F(\mathbf{x}) \quad \text{or} \quad f(\mathbf{x}) = F(\mathbf{x}) - \mathbf{x} = \mathbf{0} \quad \text{or} \quad \mathbf{x} = \mathbf{x} + \alpha f(\mathbf{x}).$$

$$\left\{\begin{array}{l}
\text{Choose } 2k \text{ and } \mathbf{x}_0. \\
\text{For } i = 0, 1, \ldots \quad (\textit{cycle or outer iterations}) \\
\quad \text{Set } \mathbf{u}_0 = \mathbf{x}_i \\
\quad \text{For } n = 1, \ldots, 2k \quad (\textit{inner iterations}) \\
\qquad \text{Compute } \mathbf{u}_n = F(\mathbf{u}_{n-1}) \\
\qquad \text{Apply the STEA to } \mathbf{u}_0, \ldots, \mathbf{u}_{2k} \\
\quad \text{end} \\
\quad \text{Set } \mathbf{x}_{i+1} = \varepsilon_{2k}^{(0)} \quad \text{or} \quad \widetilde{\varepsilon}_{2k}^{(0)} \\
\text{end}
\end{array}\right.$$

Generalized Steffensen Method (GSM)

A particular case of the RM is the **Generalized Steffensen Method (GSM)** that corresponds to take in the RM $k = m$ (the dimension of the system).

In this case, under some assumptions, the sequence $(\mathbf{x}_i)$ of the vertices of the successive ε–arrays asymptotically **converges quadratically** to the fixed point $\mathbf{x}$ of F **(Le Ferrand, 1992)**.

Remark: This method can also be applied when $F : \mathbb{R}^{m\times s} \longmapsto \mathbb{R}^{m\times s}$ but the quadratic convergence of the GSM has not yet been proved in this case.

References

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

References

- C. Brezinski, M. Redivo–Zaglia, The simplified topological ε–algorithms for accelerating sequences in a vector space, SIAM J. Sci. Comput., 36 (2014) A2227–A2247.
- C. Brezinski, M. Redivo–Zaglia, The simplified topological ε–algorithms: software and applications, submitted.

References

- C. Brezinski, M. Redivo–Zaglia, The simplified topological ε–algorithms for accelerating sequences in a vector space, SIAM J. Sci. Comput., 36 (2014) A2227–A2247.
- C. Brezinski, M. Redivo–Zaglia, The simplified topological ε–algorithms: software and applications, submitted.

A Matlab toolbox called **EPSfun** has been submitted with the paper. It contains:

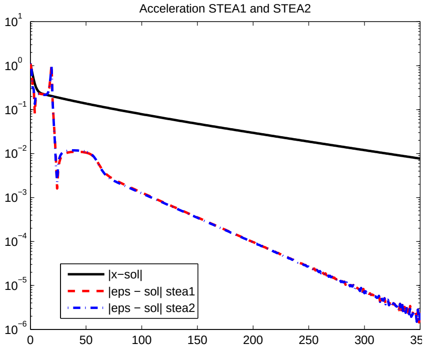
- All the STEA functions and SEAW function.
- Script for users.
- Demo scripts (producing all the examples of the paper).
- TEA and VEAW functions for comparisons.

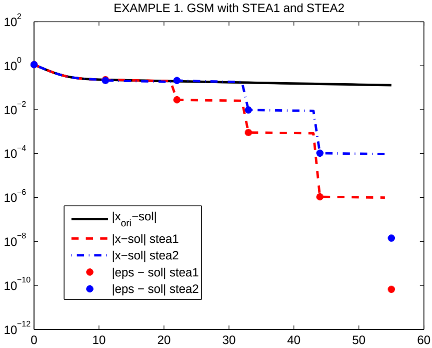
Example 1

Let us consider the following nonlinear system

$$\begin{cases} f_i(\mathbf{x}) = x_i + \sum_{j=1}^{m} x_j - (m+1), \quad i = 1, \ldots, m-1, \\ f_m(\mathbf{x}) = \prod_{j=1}^{m} x_j - 1, \end{cases}$$

whose solution is $\mathbf{x} = (1, \ldots, 1)^T$.
For $m = 5$, starting from $\mathbf{x}_0 = (1/2, \ldots, 1/2)^T$, taking $\alpha = -0.05$, and $k = 2$ for the **AM**, we obtain the results of the next Figure. After 350 iterations, an error of $7.70 \cdot 10^{-3}$ is obtained by the iterative procedure while the AM goes down to $1.4 \cdot 10^{-6}$.
The **GSM** needs 5 iterations (thus a total of 50 basic iterations) to achieve an error of $1.43 \cdot 10^{-8}$ for the STEA2 and $6.66 \cdot 10^{-11}$ for the STEA1. $\mathbf{y}$ is the mean value of a vector (also in Examples 2:5).





Acceleration Method

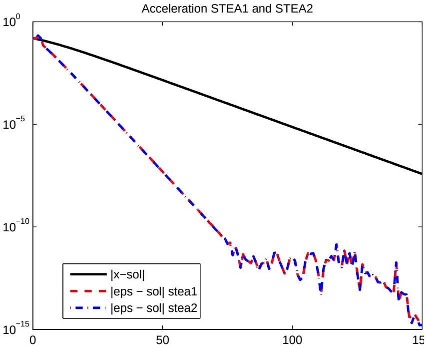
Generalized Steffensen Method

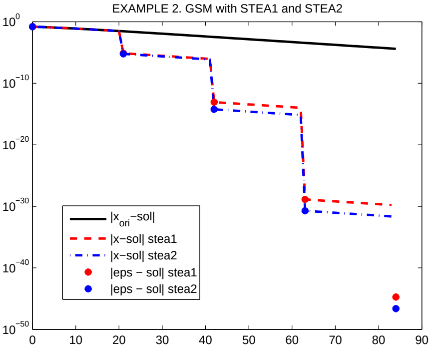
Example 2

We consider the nonlinear system

$$f_i(\mathbf{x}) = m - \sum_{j=1}^{m} \cos x_j + i(1 - \cos x_i) - \sin x_i, \quad i = 1, \ldots, m,$$

whose solution is zero.
For $m = 10$ and $\mathbf{x}_0 = (1/(2n), \ldots, 1/(2n))^T$, we obtain with $\alpha = 0.1$ and $k = 2$ for the AM, the results of the next Figure. They show that instability occurs for the AM after iteration 75 (where the error attains $1.48 \cdot 10^{-11}$), and that the GSM achieved a much better precision with a fewer number of iterations (after 2 iterations the STEA1 has an error of $8.54 \cdot 10^{-14}$, and the STEA2 an error of $5.78 \cdot 10^{-15}$).





Acceleration Method

Generalized Steffensen Method

Example 3

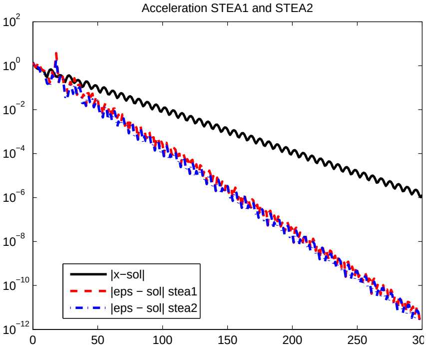
We consider the nonlinear system

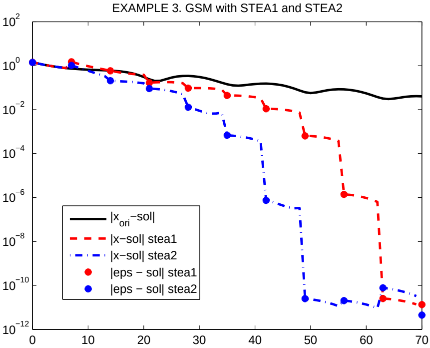
$$\begin{cases} x_1 &= x_1 x_2^3/2 - 1/2 + \sin x_3 \\ x_2 &= (\exp(1 + x_1 x_2) + 1)/2 \\ x_3 &= 1 - \cos x_3 + x_1^4 - x_2, \end{cases}$$

whose solution is $(-1, 1, 0)^T$.

Starting from $\mathbf{x}_0 = \mathbf{0}$, we obtain the results of the next Figure (left) for the **AM** with $\alpha = 0.2$ and $k = 4$.

For the **GSM** with $\alpha = 0.1$ and $k = 3$ the STEA2 gives better results than the STEA1 as shown on the Figure (right).





Acceleration Method

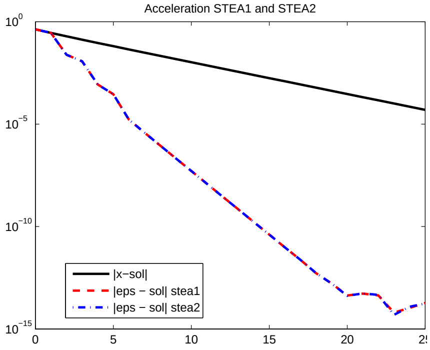
Generalized Steffensen Method

Example 4

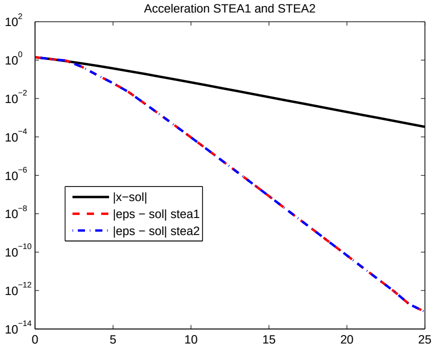
Consider now the non–differentiable system

$$\begin{cases} |x_1^2 - 1| + x_2 - 1 = 0 \\ x_2^2 + x_1 - 2 = 0. \end{cases}$$

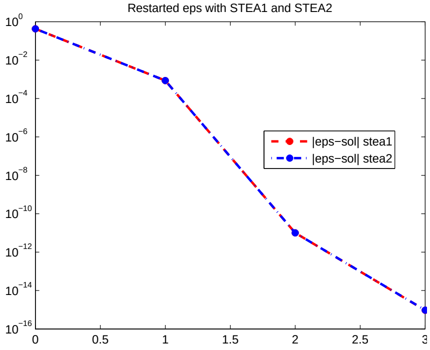
It has two solutions:
$(1,1)^T$ for which we are starting from $\mathbf{x}_0 = (1.3, 1.3)^T$, with $\alpha = -0.1$ (First Figure - left),
and $(-2,-2)$ for which we are starting from $\mathbf{x}_0 = (-1,-1)^T$, with $\alpha = 0.1$ (First Figure - right).
For the **AM** we took $k = 3$. We see that, for both solutions, the AM works quite well.
The **GSM** with $k = 2$ achieves a precision of 10^{-15} in three iterations.



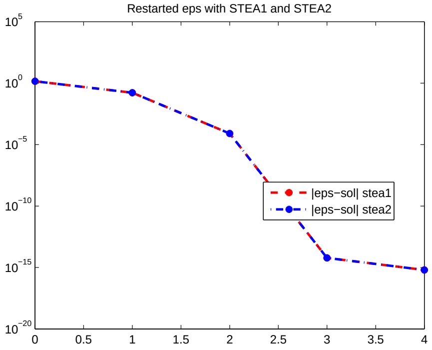
AM, first solution



AM, second solution



GSM, first solution



GSM second solution

Example 5

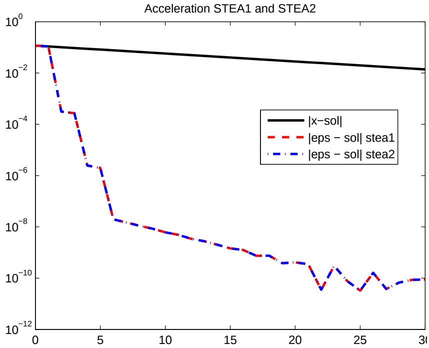
For the nonlinear system

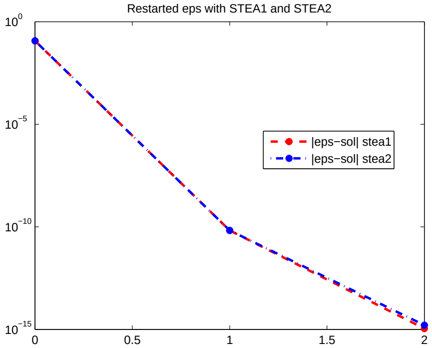
$$\sum_{j=1}^{7} x_j - (x_i + e^{-x_i}) = 0, \quad i = 1, \ldots, 7,$$

the solution is the vector with all components equal to 0.1442749507208862350.

Starting from $\mathbf{x}_0 = (1, \ldots, 1)^T/10$, $\alpha = -0.01$ and with $k = 3$ for the **AM**, we obtain the results of the following Figure.

Notice that the with **GSM** with $k = 7$ achieves with full precision in only two iterations.





Acceleration Method

Generalized Steffensen Method

Example 6

We consider the linear system $AX = B$,

- A is the *parter* matrix of dimension 5 divided by 3 (its spectral radius is 0.9054 and its condition number is 2.149)
- X is formed by the first two columns of the matrix *pei*
- B is computed accordingly.

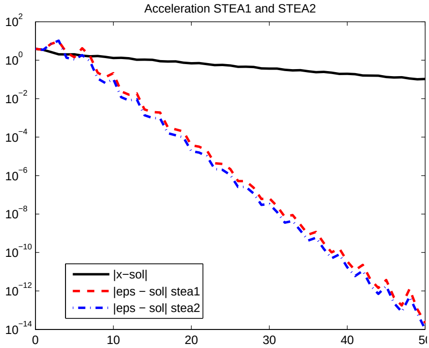
We perform the iterations

$$X_{n+1} = (I - A)X_n + B$$

starting from $X_0 = 0$.

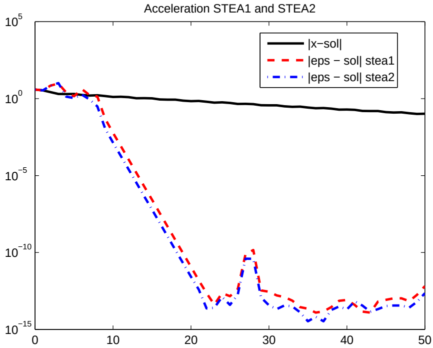
With **y** defined as the linear functional associating to a matrix the sum of its elements, we obtain for the **AM** the results of next Figure, with $k = 3$ (left) and $k = 4$ (right).

For $k = 5$ (that is for column 10 of the ε–array), the solution is obtained with full precision in one iteration by the **GSM** as stated by the theory.



Acceleration STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 3$



Acceleration STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 4$

Example 7

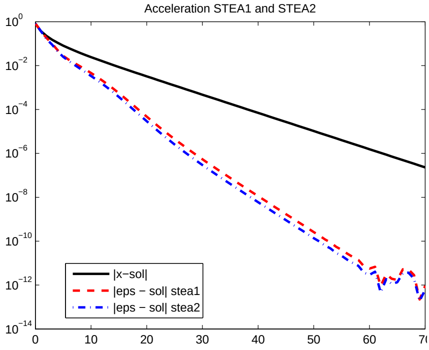
We are looking for the maximal Hermitian positive definite solution X_+ of the matrix equation

$$f(X) = X + A^* X^{-1} A - Q = 0,$$

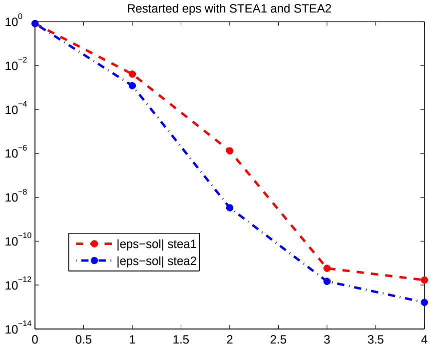
where $A, Q \in \mathbb{C}^{m \times m}$ with Q Hermitian positive definite.
For $Q = I + A^* A$, $X_+ = I$ if and only if $\rho(A) < 1$, a result by **C.-H. Guo (2001)** which provides an easy way of constructing numerical examples by taking $A = S/r$ with $r > \rho(S)$ and S any matrix.
He also proposed the following iterative method

$$\begin{aligned} X_0 &= Q, \\ X_{n+1} &= Q - A^* X_n^{-1} A, \quad n = 0, 1, \ldots \end{aligned}$$

that converges slowly if the spectral radius of A is close to 1.
We take as matrix S the *prolate* matrix, $m = 5$. For the **AM**, we took $k = 2$ and for the **GSM** $k = 5$. **y** is the trace of a matrix.



Acceleration Method



Generalized Steffensen Method

Example 8

We now want to solve the matrix equation

$$X = \sum_{i=0}^{\infty} A_i X^i. \tag{1}$$

We use the algorithm of **Z.-Z. Bai (1997)** which consists, for $n = 1, 2, \ldots$, in the iterations

$$\begin{aligned} Q_n &= I - \sum_{i=1}^{\infty} A_i X_{n-1}^{i-1}, \\ B_n &= 2B_{n-1} - B_{n-1} Q_n B_{n-1}, \\ X_n &= B_n A_0, \end{aligned}$$

starting from a given A_0, $B_0 = I$ and $X_0 = B_0 A_0$. He proved that the sequence (X_n) converges to the minimal nonnegative solution of the matrix equation.

We consider the numerical example treated by **C.-H. Guo (1999)**,were

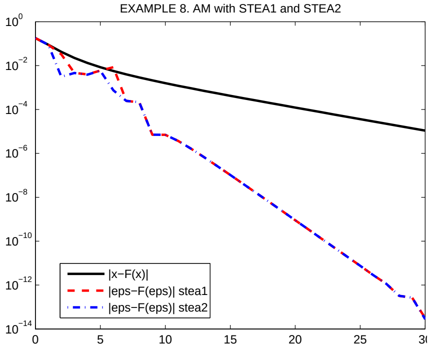
$$A_0 = \frac{4}{3}(1-p)\begin{pmatrix} 0.05 & 0.1 & 0.2 & 0.3 & 0.1 \\ 0.2 & 0.05 & 0.1 & 0.1 & 0.3 \\ 0.1 & 0.2 & 0.3 & 0.05 & 0.1 \\ 0.1 & 0.05 & 0.2 & 0.1 & 0.3 \\ 0.3 & 0.1 & 0.1 & 0.2 & 0.05 \end{pmatrix},$$

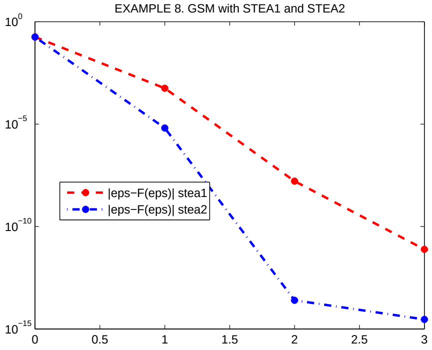
$A_i = p^i A_0$ for $i = 1, 2, \ldots,$ and $p = 0.49$.

The linear functional **y** used in the duality product corresponds to the trace of the matrix (also in all the Examples presented in the sequel).

The infinite sum in the computation of Q_n was stopped after 100 terms.

The results are given in the next Figure with $k = 3$ for the **AM**, and $k = 5$ for the **GSM**.





Acceleration Method

Generalized Steffensen Method

Example 9

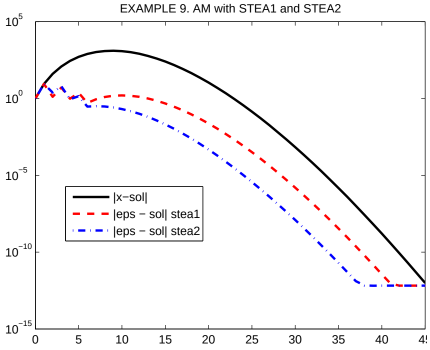
We now consider the computation of the matrix exponential by its series expansion

$$e^{At} = I + \frac{At}{1!} + \frac{A^2 t^2}{2!} + \cdots$$

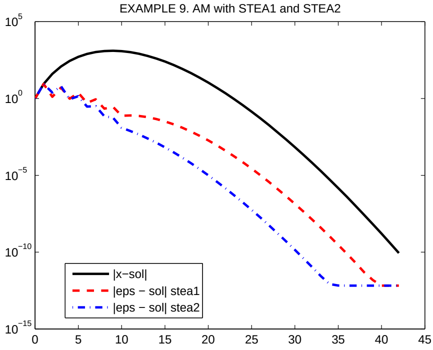
In order to be able to compute the error, we take $A = UDU^{-1}$, where D is a diagonal matrix, so that $e^{At} = Ue^{Dt}U^{-1}$.

For D, we took the *frank* matrix and for U the matrix *orthog*, both of dimension 100.

With $t = -0.099$, we obtain the results of the following Figure with $k = 3$ and $k = 5$ for the **AM**.


EXAMPLE 9. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 3$


EXAMPLE 9. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

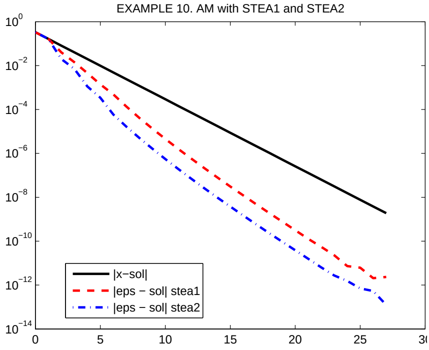
AM, $k = 5$

Example 10

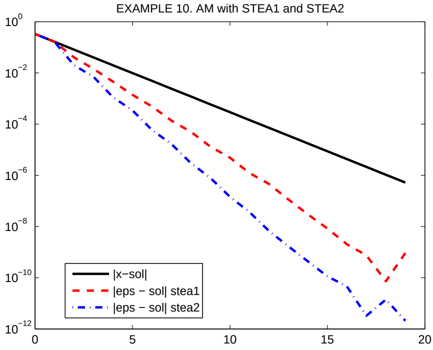
We want now to accelerate the series

$$(I - At)^{-1} = 1 + (At) + (At)^2 + \cdots$$

With the same matrix A as in Example 9, but of dimension 50 and with $t = -0.0099$, we obtain for the **AM** the results of the next Figure (on the left $k = 3$, and on the right $k = 6$).


EXAMPLE 10. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 3$


EXAMPLE 10. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 6$

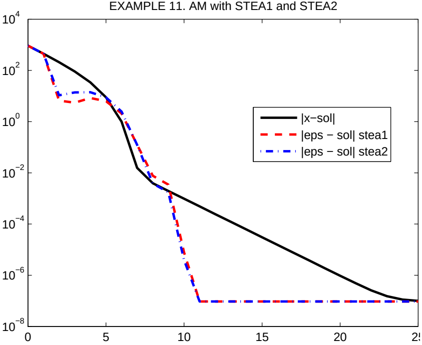
Example 11

We want now to compute the square root of a symmetric positive definite matrix A by the iterative method, denoted as the IN iteration by **N.J Higham (2008)**, which is a variant of Newton's method

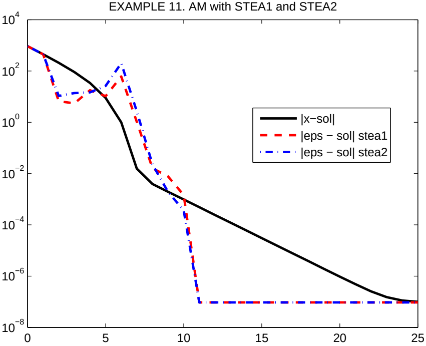
$$\begin{aligned} X_{n+1} &= X_n + E_n \\ E_{n+1} &= -\frac{1}{2} E_n X_{n+1}^{-1} E_n. \end{aligned}$$

with $X_0 = A$ and $E_0 = (I - A)/2$.

With the matrix *moler* of dimension 50, the **AM** gives the results of the next Figure with $k = 1$ on the left and $k = 2$ on the right.


EXAMPLE 11. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 1$


EXAMPLE 11. AM with STEA1 and STEA2
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 2$

Example 12

The binomial iteration for computing the AM, square root of $I - C$, where $\rho(C) < 1$, consists in the iterations

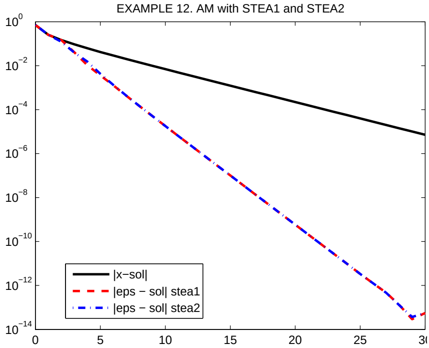
$$X_{n+1} = \frac{1}{2}(C + X_n^2), \quad k = 0, 1, \ldots,$$

with $X_0 = 0$.
The sequence (X_n) converges linearly to $X = I - (I - C)^{1/2}$ and X_n reproduces the series

$$(I - C)^{1/2} = \sum_{i=0}^{\infty} \binom{1/2}{i} (-C)^i = I - \sum_{i=1}^{\infty} \alpha_i C_i, \quad \alpha_i > 0,$$

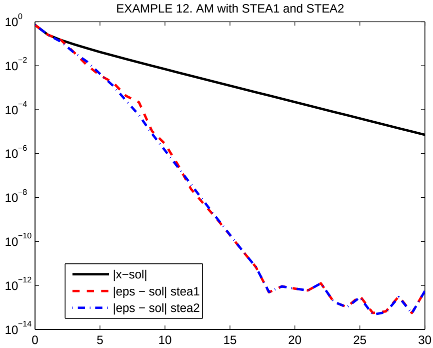
up to and including the term C^n.

For C, we took the matrix *moler* of dimension 500 divided by $1.1 \cdot 10^5$ so that $\rho(C) = 0.9855$. The **AM** gives the results of the following Figure with $k = 2$ on the left and $k = 4$ on the right, for the acceleration of the sequence (X_n).



EXAMPLE 12. AM with STEA1 and STEA2
10^0
10^−2
10^−4
10^−6
10^−8
10^−10
10^−12
10^−14
0
5
10
15
20
25
30
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 2$



EXAMPLE 12. AM with STEA1 and STEA2
10^0
10^−2
10^−4
10^−6
10^−8
10^−10
10^−12
10^−14
0
5
10
15
20
25
30
|x−sol|
|eps − sol| stea1
|eps − sol| stea2

AM, $k = 4$

THANK YOU !