Bol. Soc. Paran. Mat. (3s.) **v. 38** 4 (2020): 51–69.
©SPM –ISSN-2175-1188 on line ISSN-00378712 in press
SPM: www.spm.uem.br/bspm doi:10.5269/bspm.v38i4.37018

New Approach for Accelerating Nonlinear Schwarz Iterations

Nabila Nagid and Hassan Belhadj

abstract: The vector Epsilon algorithm is an effective extrapolation method used for accelerating the convergence of vector sequences. In this paper, this method is used to accelerate the convergence of Schwarz iterative methods for stationary linear and nonlinear partial differential equations (PDEs). The vector Epsilon algorithm is applied to the vector sequences produced by additive Schwarz (AS) and restricted additive Schwarz (RAS) methods after discretization. Some convergence analysis is presented, and several test-cases of analytical problems are performed in order to illustrate the interest of such algorithm. The obtained results show that the proposed algorithm yields much faster convergence than the classical Schwarz iterations.

Key Words: Additive Schwarz method, Domain decomposition method, Partial differential equations, Restricted additive Schwarz method, Vector Epsilon algorithm.

Contents

1 Introduction **51**

2 Linear Schwarz iterations **52**

3 Nonlinear Schwarz iterations **54**

4 Vector Epsilon algorithm **54**

5 Vector Epsilon algorithm applied to AS/RAS for linear systems **59**

6 Vector Epsilon algorithm applied to AS/RAS for nonlinear systems **61**

7 Numerical Experiments **62**

8 Conclusion **67**

1. Introduction

In scientific computing, the domain decomposition methods are now commonly used when solving large linear or nonlinear systems arising from discretization of partial differential equations (PDEs) [2,4,5,13]. The first models of these methods have been established by H.A.Schwarz, the idea is to decompose a large problem into a series of smaller subproblems, and therefore more easily resolved. There are

2010 *Mathematics Subject Classification:* 65N55, 65B05.
Submitted May 05, 2017. Published November 10, 2017

several variant of Schwarz method, for example additive Schwarz method (AS), and restricted additive schwarz method (RAS) [1,9,18,20].

So as to accelerate convergence of sequences produced by these methods, the Aitken process appears as an acceleration method suitable for many domain decomposition methods, in the case of linear problems, but this process does not warrant the convergence when the problems are nonlinear. The generalization of Aitken process for nonlinear sequences leads us to focus on the shanks transformation and it's derivatives algorithms. In practice, calculating determinants being very costly, shanks transformation is calculated just for the low values of k, in particular for $k = 1$, where the Δ^2Aitken algorithm is obtained. The most common method for calculating the shanks transformation is the Epsilon algorithm (ε-algorithm) proposed by Peter Wynn [3,6,7,10]. There exist different variants of the Epsilon algorithm that can be used with vector sequences: the vector Epsilon algorithm, or the scalar Epsilon algorithm applied to each component of the vector sequences [7,10].

There have many works that have treated the acceleration of domain decomposition methods, for example in [19], the authors accelerate the nonlinear Schwarz iterations by reduced rank extrapolation method. Another idea was described in [16], to accelerate Schwarz iterations for ordinary differential equations ODEs. There exist many other works that have treated the acceleration of domain decomposition methods, see for examples [8,14,15,17].

The purpose of this paper is to accelerate the nonlinear iterative Schwarz, using the vector Epsilon algorithm for PDEs, this algorithm is applied to the sequences of vectors produced by AS and RAS methods, we show experimentally that the proposed algorithm can provide faster convergence measured both in number of iterations and in CPU Times.

2. Linear Schwarz iterations

We consider the following problem

$$\begin{cases} L(u) = f \ in \ \Omega, \\ Bu = g \ on \ \partial\Omega. \end{cases} \tag{2.1}$$

where L is a linear operator, B is a boundary operator and Ω is a bounded domain of $\mathbb{R}^d$ $(d = 1, 2, ..)$.

H.A.Schwarz proposed an iterative method for the solution of classical boundary value problems. There are several variants of Schwarz algorithms, additive, multiplicative, and several hybrid types, a number of them are discussed in detail in [2,4,5,12,13], in the present work, we have considered the additive Schwarz method .

Let consider these notations, Ω as a union of nonoverlapping domains Ω_j, $j = 1, .., p$, $\Gamma_j = \partial\Omega_j \cap \partial\Omega$, $\Gamma_{ij} = \partial\Omega_i \cap \Omega_j$and τ is the Richardson parameter $(0 < \tau \leq 1/p)$. The additive Schwarz algorithm in the Richardson version is written as follows:

For $n = 0, ...$

For each $j = 1, ..., p$,

$$\text{solve} \begin{cases} L(u^{n+1,j}) = f & in\ \Omega_j, \\ Bu^{n+1,j} = g & on\ \partial\Gamma_j, \\ u^{n+1,j} = v^n & on\ \partial\Omega_j \backslash \Gamma_j. \end{cases}$$

Compute $w^{n+1} = u^{n+1,1} + ... + u^{n+1,p}$.

Update $v^{n+1} = (1 - p\tau)v^n + \tau w^{n+1}$.

The discretization of problem (2.1) leads to a linear system of equations of the form

$$Au = f, \tag{2.2}$$

where A is the discretization matrix by a numerical methods (Finite element, Finite Difference , or Finite volume). We use the same notation f after discretization.

A stationary iterative method for (2.2) is given by

$$u_{n+1} = u_n + M^{-1}(f - Au_n), \tag{2.3}$$

with a given initial approximation u_0 to the solution of (2.2).

Algebraic domain decomposition methods group the unknowns into subsets, $u_j = R_j u$, $j = 1, ..., p$, where R_j are rectangular restriction matrices. Coefficient matrices for subdomain problems are defined by $A_j = R_j A R_j^T$. The additive Schwarz (AS) preconditioner, and the restricted additive Schwarz (RAS) preconditioner (see [1,9,18,20]) are defined by:

$$M_{AS}^{-1} = \sum_{j=1}^{p} R_j^T A_j^{-1} R_j, \quad M_{RAS}^{-1} = \sum_{j=1}^{p} \widetilde{R}_j^T A_j^{-1} R_j, \tag{2.4}$$

where the $\widetilde{R}_j$ correspond to a non-overlapping decomposition, and it consists of zeroes and ones, in such a way that

$$\sum_{j=1}^{p} \widetilde{R}_j^T R_j = I.$$

The additive Schwarz method constructs the sequence of approximations $\{u_n\}_{n\in\mathbb{N}}$ by setting:

$$u_{n+1} = u_n + \sum_{j=1}^{p} R_j^T A_j^{-1} R_j (f - Au_n), \ n = 0, 1, ... \tag{2.5}$$

(without the Richardson acceleration).

The restricted additive Schwarz (RAS) algorithm is given by:

$$u_{n+1} = u_n + \sum_{j=1}^{p} \widetilde{R}_j^T A_j^{-1} R_j \ (f - Au_n), \ n = 0, 1, ... \tag{2.6}$$

3. Nonlinear Schwarz iterations

We consider now the problem (2.1) with a nonlinear operator L. After discretization, we obtain an algebric nonlinear system

$$F(u) = 0, \tag{3.1}$$

we transform this problem to a fixed point form

$$G(u) = u, \tag{3.2}$$

where F and G are two mappings from $\mathbb{R}^n \to \mathbb{R}^n$, using the same notation as before, we define G on each subdomain Ω_j, $j = 1, 2, ..., p$, as follows:

$$G_j(X) = R_j G(\ R_j^T(X)). \tag{3.3}$$

The corresponding nonlinear additive Schwarz method is defined by

$$u_{n+1} = u_n + \sum_{j=1}^{p} R_j^T G_j(R_j(u_n)), \ n = 0, 1, ..., \tag{3.4}$$

and the nonlinear restricted additive Schwarz method is defined by

$$u_{n+1} = u_n + \sum_{j=1}^{p} \tilde{R}_j^T G_j(R_j(u_n)), \ n = 0, 1, ..., \tag{3.5}$$

we also consider for the solution of (3.2) the Schwarz-Newton methods, where in each subdomain, the nonlinear problem is solved by a Newton, see [21].

4. Vector Epsilon algorithm

The vector Epsilon algorithm is a nonlinear extrapolation method for accelerating the convergence of sequences, one can say also that this is a generalization of Aitken method. There exist several versions of the Epsilon algorithm (topological, scalar, and vector Epsilon algorithm). In this work, we are only interested in the vector form. We consider thereafter the fundamental algebraic results in the theory of the vector Epsilon algorithm [3,6,7,10].

First, we recall some results concerning the Aitken's process.
Let $U = (u_n)_{n\in\mathbb{N}}$ is a sequence that converges to u, the convergence acceleration methods consists in transforming $U = (u_n)_{n\in\mathbb{N}}$ into another sequence $(\varepsilon_2^{(n)})$ which converges faster to the same limit u.

Among these transformation methods, the best-known are the Richardson methods and Δ^2Aitken.
We define the operator Δ such as

$$\begin{cases} \Delta^0 u_n = u_n \\ \Delta^{k+1} u_n = \Delta^k u_{n+1} - \Delta^k u_n \end{cases} .$$

Definition 4.1. *Let $U = (u_n)_{n\in\mathbb{N}}$ and $V = (v_n)_{n\in\mathbb{N}}$ two sequences of real numbers that converge to u, we say that $(u_n)_{n\in\mathbb{N}}$ converges faster than $(v_n)_{n\in\mathbb{N}}$ if:*

$$\lim_{n\to\infty} \frac{u_n - u}{v_n - u} = 0 \quad .$$

Definition 4.2. *Let $U = (u_n)_{n\in\mathbb{N}}$ be a sequence of real numbers, the Δ^2Aitken process consists in transforming the sequence (u_n) into a new sequence $(\varepsilon_2^{(n)})$ defined by :*

$$\varepsilon_2^n = \frac{u_{n+2}\, u_n \, - u_{n+1}^2}{\Delta^2 u_n} = u_{n+1} - \frac{\Delta u_{n+1}}{\frac{\Delta u_{n+1}}{\Delta u_n} - 1} .$$

Theorem 4.3. *If we apply the Δ^2Aitken process to the sequence $U = (u_n)_{n\in\mathbb{N}}$ which satisfies the condition*

$$\lim_{n\to\infty}(u_{n+1} \, - u)/(u_n \, - u) = \lim_{n\to\infty} \Delta u_{n+1} \, /\Delta u_n \, = \rho \neq 1,$$

then the sequence ε_2^n converges to u faster than u_{n+1}.

Proof:

Using the definition 4.1 we have:

$$\lim_{n\to\infty} \frac{u_{n+1} - \frac{\Delta u_{n+1}}{\frac{\Delta u_{n+1}}{\Delta u_n} - 1} - u}{u_{n+1} - u} = 0$$
$$\Leftrightarrow \lim_{n\to\infty} \frac{u_{n+2} - u_{n+1}}{u_{n+1} - u} \times \frac{1}{\frac{\Delta u_{n+1}}{\Delta u_n} - 1} = 1$$
$$\Leftrightarrow \lim_{n\to\infty} \frac{\frac{u_{n+2} - u}{u_{n+1} - u} - 1}{\frac{\Delta u_{n+1}}{\Delta u_n} - 1} = 1$$

if the condition of the theorem is satisfied, then (ε_2^n) converges to u faster than (u_{n+1}).

□

Now, we seek the conditions on (u_n) in order that $\varepsilon_2^{(n)} = u$ for $n > N$ (N is a given rank).

We have seen that:

$$\varepsilon_2^{(n)} = \frac{u_{n+2}\, u_n \, - u_{n+1}^2}{\Delta^2 u_n},$$

writing $\varepsilon_2^{(n)}$ based on determinants:

$$\varepsilon_2^{(n)} = \frac{\begin{vmatrix} u_n & u_{n+1} \\ \Delta u_n & \Delta u_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta u_n & \Delta u_{n+1} \end{vmatrix}} ,$$

we want to have

$$\frac{\begin{vmatrix} u_n & u_{n+1} \\ \Delta u_n & \Delta u_{n+1} \end{vmatrix}}{\begin{vmatrix} 1 & 1 \\ \Delta u_n & \Delta u_{n+1} \end{vmatrix}} = u \qquad \forall n \geq N \quad ,$$

therefore

$$\begin{vmatrix} u_n - u & u_{n+1} - u \\ \Delta u_n & \Delta u_{n+1} \end{vmatrix} = \begin{vmatrix} u_n - u & u_{n+1} - u \\ u_{n+1} - u & u_{n+2} - u \end{vmatrix} = 0 \qquad \forall n \geq N \quad ,$$

for this determinant to be zero, it is necessary and sufficient that there exist a_0 and a_1 such that:

$$a_0(u_n - u) + a_1(u_{n+1} - u) = 0 \qquad \forall n > N \ ,$$

if $a_0 + a_1 = 0$ we remark that $u_n = u_{n+1}$ $\forall n$ and then the Δ^2Aitken process cannot be applied to u_n, and if $a_0 + a_1 \neq 0$ then, we have $\varepsilon_2^{(n)} = u$ $\forall n > N$, therefore, we have the following theorems.

Theorem 4.4. *A necessary and sufficient condition to have* $\varepsilon_2^{(n)} = u$ $\forall n > N$, *is that the sequence* (u_n) *verifies*

$$a_0(u_n - u) + a_1(u_{n+1} - u) = 0 \qquad \forall n > N \ , \textit{ with } \ a_0 + a_1 \neq 0.$$

This theorem can be generalized to high order using a nonlinear acceleration method, the Shanks transformation [3,6,7]. This transformation called $e_k^n(U)$ is built such that $e_k^n(U) = u \quad \forall n > N$, and it consists in computing the quantities $e_k^n(U)$ as follows

$$e_k^n(U) = \sum_{i=0}^{k} a_i^{(n,k)} u_{n+i} \qquad \forall n > N \quad \text{with} \quad \sum_{i=0}^{k} a_i^{(n,k)} = 1 \ ,$$

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

$$e_k^n(U) = \frac{\begin{vmatrix} u_n & \dots & u_{n+k} \\ \Delta u_n & \dots & \Delta u_{n+k} \\ . & & . \\ . & & . \\ . & & . \\ \Delta u_{n+k-1} & & \Delta u_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ \Delta u_n & \dots & \Delta u_{n+k} \\ . & & . \\ . & & . \\ . & & . \\ \Delta u_{n+k-1} & & \Delta u_{n+2k-1} \end{vmatrix}} \quad \text{with} \quad \sum_{i=0}^{k} a_i^{(n,k)} = 1 \ ,$$

the transformed expression given above is to a ratio of two determinants having a particular structure, and it's a part of the hankel determinants [6,7].
The previous results leads to the following theorem.

Theorem 4.5. *If for a fixed k, the sequence U is such that there exists $u \in \mathbb{R}$ and $a_0, ..., a_k \in \mathbb{R}$ with $\sum_{i=0}^{k} a_i \neq 0$ satisfying $\sum_{i=0}^{k} a_i(u_{n+i} - u) = 0 \quad \forall n > N$, then*

$$e_k^n(U) = \varepsilon_{2k}^{(n)} = u \ \forall n > N.$$

The proof of theorems 4.4 and 4.5 are given for example in [6,7,8].

Remark 4.6. *A recursive rule for computing the quantities $e_k^n(U)$ of shanks transformation has been given by [6,7], these quantities can be computed by the following Epsilon algorithm:*

$$\varepsilon_{-1}^{(n)} = 0 \qquad \varepsilon_0^{(n)} = U_n \qquad n = 0, 1, ...$$
$$\varepsilon_{k+1}^{(n)} = \varepsilon_{k-1}^{(n+1)} + (\Delta\varepsilon_k^{(n)})^{-1} \qquad n, k = 0, 1, ...$$

where the inverse of a vector y is defined by: $y^{-1} = \frac{\overline{y}}{\|y\|_2^2}$.

Using theorem 4.5, it has been proved that the vector Epsilon algorithm provides a direct method for solving the linear systems of equations [6,7].

Theorem 4.7. *If we apply the vector Epsilon algorithm to the sequence $\{u_n\}$ produced by*

$$u_{n+1} = Au_n + b,$$

with a given u_0 and A is a real square matrix such that $I - A$ is invertible, then we have

$$\varepsilon_{2m}^{(n)} = u \qquad for \ \ n = 0, 1, ...$$

where $u = (I - A)^{-1}\ b$ and m is the degree of minimal polynomial of A for the vector $u_0 - u$.

Proof: Let $p(t) = \sum_{i=0}^{m} a_i t^i$ the minimal polynomial of A for the vector $u_0 - u$, the definition of the minimal polynomial of a matrix for a vector is:

$$\sum_{i=0}^{m} (a_i\ A^i)\ (u_0 - u) = 0,$$

the matrix $I - A$ is invertible, therefore $p(1) = \sum_{i=0}^{m} a_i \neq 0$ on the other hand, we have $u = Au + b$,

so $u_{n+1} - u = A(u_n - u)$,
and $u_k - u = A^k(u_0 - u), \quad \forall k \geqslant 0$
replacing in $p(t)$ we have

$$A^n \sum_{i=0}^{m} a_i \ (u_i - u) = \sum_{i=0}^{m} a_i \ (u_{n+i} - u) = 0 \ \ \forall n,$$

so using theorem 4.5 we prove that

$$\varepsilon_{2m}^{(n)} = u \qquad \forall n \geq 0.$$

□

Now, let to be solve the following nonlinear problem

$$\text{Find } x \in \mathbb{R}^p \text{ such that } x = F(x)$$

where $F: \ \mathbb{R}^p \longrightarrow \ \mathbb{R}^p$ is differentiable in the sense of Frechet in a neighborhood of x, knowing x_0 we set $u_0 = x_n$ and we solve for $k = 1, ..., 2m - r$ the following iterative problem

$$u_k = F(u_{k-1}).$$

To calculate $\varepsilon_{2(m-r)}^{(r)}$ we applied the Epsilon algorithm to the vectors $u_0, ..., u_{2m-r}$, then we take $x_{n+1} = \varepsilon_{2(m-r)}^{(r)}$, where m is the degree of minimal polynomial of $F'(x)$ for the vector $x_n - x$ and r is the multiplicity of the root $(\lambda = 0)$ for this minimal polynomial.
The nonlinear fixed point problems can be accelerated by the Epsilon algorithm if the conditions of theorem 4.8 is verified.

Theorem 4.8. *Let $F: \ \mathbb{R}^p \longrightarrow \ \mathbb{R}^p$ such as there exist $x \in \mathbb{R}^p$ wich satisfies $x = F(x)$, such that F is differentiable in the sense of Frechet in a neighborhood of x, and such that $I - F'(x)$ is invertible. Then there exists a neighborhood V of x such that for any $x_0 \in V$ the previous algorithm converges to x at least quadratically, ie:*

$$\|x_{n+1} - x\| \quad = o(\ \|x_n - x\| \ ^2) \qquad\qquad n = 0, 1, ...$$

Proof: If F is differentiable in the sense of Frechet in a neighborhood of x, we have:

$$u_{k+1} - x = F'(x)(u_k - x) + o(\ \|u_k - x\| \ ^2),$$

where $o(\ \|z_k\| \ ^2)$ refers to a vector $y_k \in \mathbb{R}^p$ such as $\forall k > K \quad \|y_k\| \leq A \, \|z_k\| \ ^2$.
Let $p(t) = \sum\limits_{i=0}^{m} a_i t^i$ the minimal polynomial of $F'(x)$ for the vector $x_n - x$,

since $I - F'(x)$ is invertible, we have $p(1) = \sum\limits_{i=0}^{m} a_i \neq 0$,

we have: $u_1 - x = F'(x)(u_0 - x) + o(\ \|u_0 - x\| \ ^2)$, and $u_k - x = [F(x)']^k(u_0 - x) + \sum_{j=0}^{k-1}[F(x)']^{k-1-j}o(||u_j - x||^2)$, replacing in the minimal polynomial

$$\sum_{i=0}^{m} a_i[F'(x)]^i(x_n - x) = \sum_{i=0}^{m} a_i[F'(x)]^i(u_0 - x) = 0 \text{ (because } u_0 = x_n),$$

and

$$\sum_{i=0}^{m} a_i[F'(x)]^i(u_0 - x) = \sum_{i=0}^{m} a_i(u_i - x) + \sum_{j=0}^{i-1}[F(x)']^{i-1-j}o(||u_j - x||^2) = 0,$$

therefore

$$\sum_{i=0}^{m} a_i u_i = x\sum_{i=0}^{m} a_i + \sum_{j=0}^{i-1}[F(x)']^{i-1-j}o(||u_j - x||^2),$$

using theorems 4.4 and 4.5 we get

$$\varepsilon_{2(m-r)}^{(r)} = x + \sum_{j=0}^{i-1}[F(x)']^{i-1-j}o(||u_j - x||^2).$$

□

5. Vector Epsilon algorithm applied to AS/RAS for linear systems

We consider the following problem:

$$\begin{cases} L(u) = f & in\ \Omega, \\ Bu = g & on\ \partial\Omega. \end{cases} \tag{5.1}$$

In the case where the operator L is linear, a discretization of the equation (5.1) leads to a linear system of equations of the form

$$Au = f. \tag{5.2}$$

The additive Schwarz methods allows to compute the sequence of approximations $\{u_n\}_{n\in\mathbb{N}}$ by setting:

$$u_{n+1} = u_n + M_{AS}^{-1}(f - Au_n), \tag{5.3}$$

with

$$M_{AS}^{-1} = \sum_{j=1}^{p} R_j^T A_j^{-1} R_j,$$

i.e.,

$$\begin{pmatrix} u_{n+1,1} \\ u_{n+1,2} \\ . \\ . \\ . \\ u_{n+1,p} \end{pmatrix} = \begin{pmatrix} u_{n,1} \\ u_{n,2} \\ . \\ . \\ . \\ u_{n,p} \end{pmatrix} + \begin{pmatrix} A_{11} & & & \\ & A_{22} & 0 & \\ & & . & \\ 0 & & . & \\ & & . & \\ & & & A_{pp} \end{pmatrix}^{-1} \begin{pmatrix} r_{n,1} \\ r_{n,2} \\ . \\ . \\ . \\ r_{n,p} \end{pmatrix},$$

we can write

$$u_{n+1} = B\ u_n + F, \tag{5.4}$$

where

$$B = I - M_{AS}^{-1}A.$$

Theorem 5.1. *Suppose that A and M_{AS}^{-1} a real square matrices that have the same size such that $C = M_{AS}^{-1}A$ is non singular. If we apply the vector Epsilon algorithm to the aditive Schwarz sequence (5.4) then $\varepsilon_{2m}^n = u$, where u is the solution of the linear system $Cu = F$, and m the degree of the minimal polynomial of B .*

Proof: The solution u is a fixed point of the operator

$$u \to u + M_{AS}^{-1}(f - Au),$$

let $P = M_{AS} - A$ is the difference between A and M_{AS}, when (5.3) converge, it converges to the solution of the preconditioned system

$$M_{AS}^{-1}Au = M_{AS}^{-1}f,$$

by setting $y_n = u_n - u$ we obtain

$$\begin{aligned} u_{n+1} &= u_n + M_{AS}^{-1}(f - Au_n) \\ &= (I - M_{AS}^{-1}(M_{AS} - P))u_n + M_{AS}^{-1}\ f \\ &= M_{AS}^{-1}Pu_n + M_{AS}^{-1}Au \\ &= M_{AS}^{-1}Pu_n + M_{AS}^{-1}(M_{AS} - P)u \\ &= u + M_{AS}^{-1}P(u_n - u) \end{aligned}$$

the equivalent system becomes

$$y_{n+1} = M_{AS}^{-1}Py_n = By_n,$$

using theorem 4.7, we have $\varepsilon_{2m}^n = u$, where m is the degree of the minimal polynomial of B, and we have

$$P_m(B)(u_0 - u) = \textstyle\sum_{n=0}^{k}\gamma^n B^n(u_0 - u) = \sum_{n=0}^{k}\gamma^n(u_n - u) = 0,$$

where γ^n are the coefficients of the polynomial P_d such that $P_d(1) = 1$.

□

6. Vector Epsilon algorithm applied to AS/RAS for nonlinear systems

We consider now the nonlinear reaction diffusion problem defined by:

$$Lu - f(u) = g \qquad in\ \Omega, \tag{6.1}$$

we can write

$$Lu = G(u) \quad in\ the\ sense\ of\ D'(\Omega). \tag{6.2}$$

The corresponding discretized problem can be written as follows:

$$AU = G(U) \in \mathbb{R}^p, \tag{6.3}$$

where A is the matrix of the discretized operator L, obtained by the finite elements on a regular grid, $G: \mathbb{R}^p \longrightarrow \mathbb{R}^p$ is the nonlinear function, and U is the vector containing the approximation of the solution of the continuous problem to grid points. We remark, that if we put $F = A^{-1}G(.)$, then the problem (6.3) is equivalent to $U = F(U)$.

Let solve the following problem:

$$\text{find } x \in \mathbb{R}^p \text{ such that } x = F(x) \tag{6.4}$$

where $F: \mathbb{R}^p \longrightarrow \mathbb{R}^p$ is differentiable in the sense of Frechet, in a neighborhood of x, m is the degree of minimal polynomial of $F'(x)$ for the vector $x_n - x$ and r is the multiplicity of the root $(\lambda = 0)$ for this minimal polynomial.

Knowing x_0 we set $u_0 = x_n$ and we solve for $k = 1, ..., 2m - r$, the following iterative problem $u_k = F(u_{k-1})$.

To calculate $\varepsilon_{2(m-r)}^{(r)}$ we apply the Epsilon algorithm to the vectors $u_0, ..., u_{2m-r}$. then we take $x_{n+1} = \varepsilon_{2(m-r)}^{(r)}$.

The application of the Epsilon algorithm to the nonlinear RAS provides a method of resolution with quadratic convergence, see [11].

Theorem 6.1. *Let* $u_{n+1} = u_n + \sum_{j=1}^{p} \tilde{R}_j^T F_j(R_j(u_n)) \qquad n = 0, 1, ...,$ *where* $F: \mathbb{R}^p \longrightarrow \mathbb{R}^p$ *is defined on each subdomain* $F_j(X) = R_j F(\ R_j^T(X))$*, and* $G(u) := \mathcal{F}(u) - u = \sum_{j=1}^{p} \tilde{R}_j^T F_j(u) - u = 0$*. If* $\mathcal{F}$ *is differentiable in the sense of Frechet in a neighborhood of* u*, and* $I - \mathcal{F}'(u)$ *is invertible, then there exists a neighborhood* V *of* u *such that* $\forall\ x_0 \in V$

$$\|x_{n+1} - u\| \quad = o(\ \|x_n - u\|^{\ 2}) \quad n = 0, 1, ...$$

Proof: If $\mathcal{F}$ is differentiable in the sense of Frechet in a neighborhood of u, we have:

$$u_{k+1} - u = \mathcal{F}'(u)(u_k - u) + o(\ \|u_k - u\|\ ^2).$$

Let $p(t) = \sum_{i=0}^{m} a_i t^i$ the minimal polynomial of $\mathcal{F}'(u)$ for the vector $u_n - u$, since $I - \mathcal{F}'(u)$ is invertible, thus $p(1) = \sum_{i=0}^{m} a_i \neq 0$,

we have: $u_1 - u = \mathcal{F}'(u)(u_0 - u) + o(\ \|u_0 - u\|\ ^2)$,

and $u_k - u = [\mathcal{F}'(u)]^k (u_0 - u) + o(\ \|u_0 - u\|\ ^2)$,
replacing in the minimal polynomial

$$\sum_{i=0}^{m} a_i [\mathcal{F}'(u)]^i (x_n - u) = \sum_{i=0}^{m} a_i (u_i - u) + o(\ \|u_0 - u\|\ ^2) = 0,$$

therefore $u_0 = x_n$, using theorems 4.4 and 4.5 we get

$$\varepsilon_{2(m-r)}^{(r)} = u + o(\ \|x_n - u\|\ ^2).$$

□

The algorithm (Epsilon-RAS)
In case of convergence , $\lim_{n \longrightarrow \infty} u_n = u$.

1. Choose a starting approximation x_0.
2. Set $u_0 = x_n$ at the iteration n, and

$$u_{k+1} = u_k + \sum_{j=1}^{p} \tilde{R}_j^T F_j(R_j(u_k)) \qquad k = 0, \ldots .2m - r.$$

3. Apply the Epsilon algorithm to the vectors $u_0, \ldots, u_{2m-r}$ to calculate $\varepsilon_{2(m-r)}^{(r)}$.
4. Compute x_{n+1} such that

$$x_{n+1} = \varepsilon_{2(m-r)}^{(r)},$$

($r = 0$ if the partial Frechet derivative of F is invertible).

7. Numerical Experiments

In this section, we compare the performance of Schwarz iterations with those accelerated with the vector Epsilon algorithm in terms of number of iterations and CPU Time.
We treat two different applications, the first one in the linear case and the second one in the nonlinear case. We have implemented the finite element discretization in

two spatial dimensions and all computational experiments presented were carried out using Freefem++.
We compare results for different number of nonoverlapping subdomains and different discretizations.
In all plots, the labels AS and RAS refer to the additive and restricted additive Schwarz methods, respectively. The labels Epsilon-AS and Epsilon-RAS refer to the vector Epsilon algorithm applied to sequences constructed by the AS and RAS methods, respectively.

Application to the Helmholtz Problem

We consider the Helmholtz problem

$$\begin{cases} -k^2u - \Delta u = f & in\ \Omega \\ \frac{\partial u}{\partial n} = g & on\ \partial\Omega \end{cases} \tag{7.1}$$

Let k be a constant, we take $k = 10$, $g = y(y-1)$ and we use a finite element discretization on an equidistant grid on the domain $\Omega = [0,1] \times [0,1]$ with homogeneous Neumann boundary conditions. Figure 1 illustrates the computational result on all domain Ω using FreeFem ++.

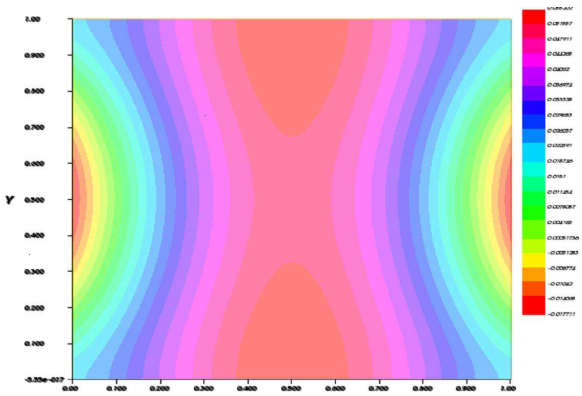

Figure 1: Helmholtz problem: Solution on Ω

Now, using the Epsilon-RAS algorithm, we solve the problem on multiple subdomains. Let p the number of subdomains, figure 2 shows the solution for p=4, and table 1 shows the behaviour of the error norm (L^∞) when we apply the Epsilon-RAS algorithm to the problem (7.1).

Figure 3 shows the behaviour of the error norm using a logarithmic scale versus number of iterations for all algorithms, when p=16.

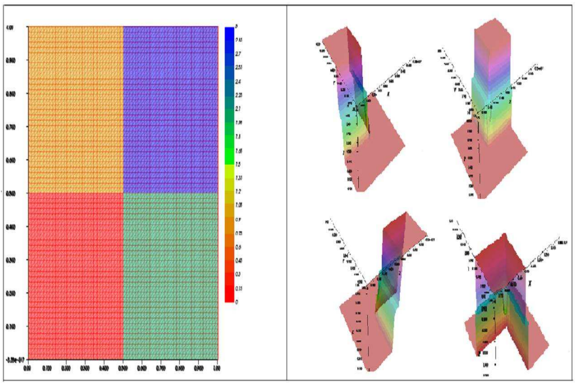

Figure 2: Helmholtz problem: Solution for p=4

Table 1: Helmholtz problem: Error norm for Epsilon-RAS algorithm with p=4

$E_0^{(n)}$	$E_2^{(n)}$	$E_4^{(n)}$	$E_6^{(n)}$	$E_8^{(n)}$	$E_{10}^{(n)}$
0.92×10^{-1}					
0.67×10^{-1}	0.18×10^{-3}				
0.29×10^{-1}	0.86×10^{-4}	0.37×10^{-5}			
0.92×10^{-2}	0.61×10^{-4}	0.22×10^{-5}	0.38×10^{-6}		
0.86×10^{-2}	0.43×10^{-4}	0.88×10^{-6}	0.25×10^{-6}	0.98×10^{-7}	
0.64×10^{-2}	0.23×10^{-4}	0.72×10^{-6}	0.72×10^{-7}	0.83×10^{-8}	0.12×10^{-8}
0.27×10^{-2}	0.91×10^{-5}	0.65×10^{-6}	0.51×10^{-7}	0.44×10^{-8}	
0.86×10^{-3}	0.77×10^{-5}	0.55×10^{-6}	0.28×10^{-7}		
0.64×10^{-3}	0.68×10^{-5}	0.43×10^{-6}			
0.52×10^{-3}	0.45×10^{-5}				
0.44×10^{-3}					

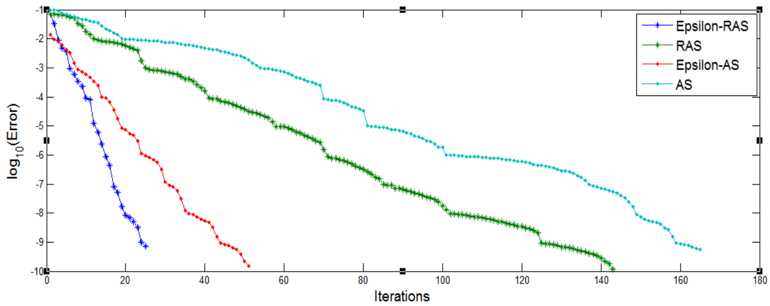

Figure 3: Helmholtz problem. Convergence for p=16

It can be observed that AS and RAS require 165 and 143 iterations, respectively, whereas Epsilon-AS and Epsilon-RAS only need 51 and 25 iterations, respectively, for the same problem.

The vector Epsilon algorithm reduces both the number of iterations and the CPU Time. Similar results about fastness of convergence are observed when we compare the CPU Time versus the number of subdomains for all algorithms. The results are reported in table 2 and figure 4.

Table 2: Helmholtz problem: CPU Time versus Number of subdomains

Subdomains (P)	CPU Time AS	CPU Time Epsilon -AS	CPU Time RAS	CPU Time Epsilon -RAS
P = 4	500.24	462.13	448.37	221.74
P = 9	366.92	341.27	232.13	210.11
P = 16	273.88	244.62	198.46	172.68
P = 25	218.60	161.382	143.15	122.44
P = 36	148.07	119.21	72.64	46.91
P = 49	131.15	92.90	41.12	28.35

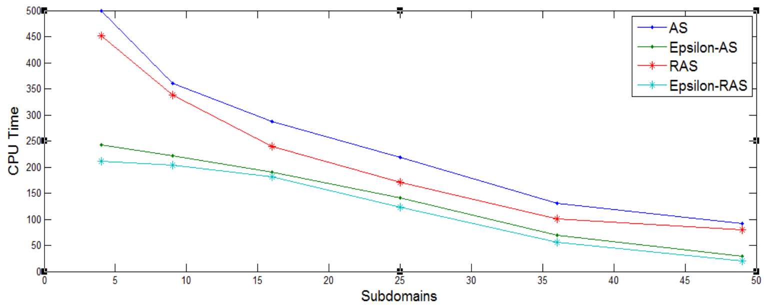

Figure 4: Helmholtz problem. CPU Time versus number of subdomains

Application to the Bratu problem

We consider now the following nonlinear reaction diffusion problem

$$-\Delta u + \lambda e^{u} = f \qquad in \quad \Omega \tag{7.2}$$

The domain is the unit square $\Omega = [0,1] \times [0,1]$ decomposed uniformly into p nonoverlapping subdomains.

b is chosen so that the solution is known to be the vector of all ones, using a finite element discretization, we obtain the following nonlinear system of equations

$$AX + \lambda e^{X} - b = 0 \qquad in \quad \Omega \tag{7.3}$$

For this problem, we use the nonlinear additive and restricted additive Schwarz iterations, respectively; and their acceleration with vector Epsilon algorithm. In each subdomain, we use the nonlinear SSOR method to solve the smaller nonlinear problem.

$$G_j(X) = C_\omega X + \omega(2-\omega)(D_j - \omega U_j)^{-1} D_j (D_j - \omega L_j)^{-1}(b - \lambda e^{X})$$

where
$C_\omega = (D_j - \omega U_j)^{-1}(\omega L_j + (1-\omega)D_j)(D_j - \omega L_j)^{-1}(\omega U_j + (1-\omega)D_j)$ and $A_j =$

$D_j - L_j - U_j$ We chose the values of $\lambda = 6.998$ and $\Omega = 0.5$. Using FreeFem++,

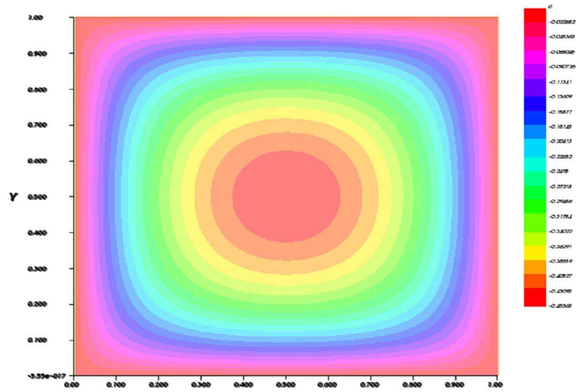

Figure 5: Bratu problem. Solution on Ω

the solution of the problem (7.3) on Ω is presented in figure 5.

Figure 6 shows the solution of problem (7.2) using nonlinear Epsilon-RAS algorithm on p=8 nonoverlapping subdomains.

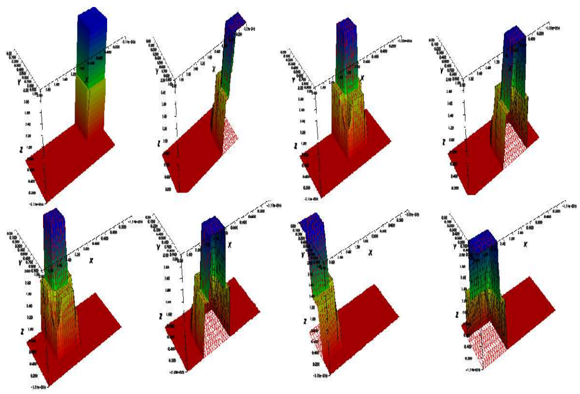

Figure 6: Bratu problem: Solution for p=8

The following results reported in table 3, show the L^{∞} Error norm when we apply the nonlinear Epsilon-RAS algorithm to the problem (7.2).

To show experimentally that the vector Epsilon algorithm can indeed provide a good acceleration, we show the behaviour of the error norm using a logarithmic scale versus number of iterations for all methods, when p=16, see figure 7.

As in the linear case, it can be observed that both nonlinear additive and restricted additive Schwarz iterations take too long to converge, whereas nonlinear Epsilon-AS and Epsilon-RAS require far fewer iterations for convergence.
As in the previous example, When we compare the CPU Time, one can observe

Table 3: Bratu problem: Error norm for nonlinear Epsilon-RAS algorithm with p=8

$E_0^{(n)}$	$E_2^{(n)}$	$E_4^{(n)}$	$E_6^{(n)}$	$E_8^{(n)}$	$E_{10}^{(n)}$	$E_{12}^{(n)}$
0.93×10^{-1}						
0.80×10^{-1}	0.21×10^{-2}					
0.73×10^{-1}	0.88×10^{-3}	0.48×10^{-4}				
0.69×10^{-1}	0.63×10^{-3}	0.33×10^{-4}	0.19×10^{-5}			
0.57×10^{-1}	0.52×10^{-3}	0.27×10^{-4}	0.78×10^{-6}	0.82×10^{-7}		
0.44×10^{-1}	0.49×10^{-3}	0.15×10^{-4}	0.63×10^{-6}	0.66×10^{-7}	0.58×10^{-8}	
0.36×10^{-1}	0.38×10^{-3}	0.93×10^{-5}	0.59×10^{-6}	0.38×10^{-7}	0.31×10^{-8}	0.74×10^{-9}
0.28×10^{-1}	0.22×10^{-3}	0.89×10^{-5}	0.37×10^{-6}	0.15×10^{-7}	0.24×10^{-8}	
0.94×10^{-2}	0.97×10^{-4}	0.68×10^{-5}	0.24×10^{-6}	0.71×10^{-8}		
0.83×10^{-2}	0.82×10^{-4}	0.45×10^{-5}	0.11×10^{-6}			
0.71×10^{-2}	0.73×10^{-4}	0.32×10^{-5}				
0.53×10^{-2}	0.65×10^{-4}					
0.37×10^{-2}						

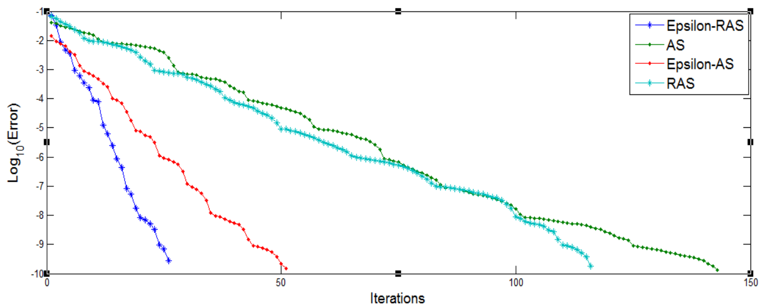

Figure 7: Bratu problem: Convergence for different number of subdomains p

that the Epsilon algorithm performs very well, the application of the extrapolation method yields much faster convergence than the classical Schwarz iterations. The results are reported in table 4 and in figure 8.

Table 4: Bratu problem: CPU Time versus Number of subdomains

Subdomains (P)	CPU Time AS	CPU Time Epsilon -AS	CPU Time RAS	CPU Time Epsilon -RAS
P = 4	572.11	299.98	498.32	282.14
P = 9	482.31	279.11	421.16	224.62
P = 16	322.67	212.04	281.75	184.35
P = 25	242.67	168.22	200.83	131.17
P = 36	154.51	98.14	122.98	83.56
P = 49	98.83	36.44	75.41	21.12

8. Conclusion

We have proposed an accelerated form of Schwarz iterations for nonlinear problems (AS-RAS) using the vector Epsilon algorithm. Comparing CPU-Time and the number of iterations, we show that this accelerated method is fast and it has a better accuracy than the direct classical Schwarz method.
As perspective of the present work, we can generalize the acceleration method for non stationary PDEs, and apply it to a real modelling case.

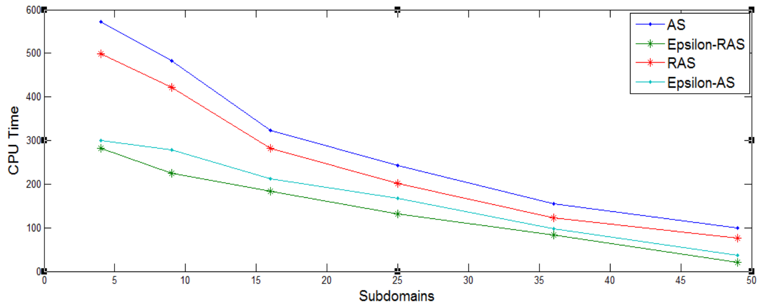

Figure 8: Bratu problem: CPU Time versus number of subdomains

Acknowledgments

The authors are extremely grateful to the referees for their helpful suggestions for the improvement of the paper.

References

1. Formmer, A., Szyld, D.B., *An algebraic convergence theory for restricted additive schwarz methods using weighted max norms*, SIAM journal on numerical analysis 39, 463-479, (2001).

2. Quarteroni, A., Valli, A., *Domain decomposition methods for PDEs*, Oxford University Press, (1999).

3. Salam, A., Graves-Morris, P.R., *On the vector $\varepsilon-$ algorithm for solving linear systems of equations*, Numerical Algorithms 29, 229-247, (2002).

4. Toselli, A., Widlund, O.B., *Domain decomposition methods: algorithms and theory*, Berlin : Springer 34, xvi+−450, (2005).

5. Smith, B., Bjorstad, P., Gropp, W.D., *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*, Cambridge university press, (2004).

6. Brezinski, C., *Méthodes d'accélération de la convergence en analyse numérique*, Centre de Calcul Scientifique de l'Armement, (1971).

7. Brezinski, C., *Variations on Richardson's method and acceleration*, Bulletin of the Belgian Mathematical Society Simon Stevin 3, 33-44, (1996).

8. Roland, C., *Méthodes d'accélération de convergence en analyse numérique et en statistique*, Doctoral dissertation, Université des Sciences et Technologie de Lille-Lille, (2005).

9. Efstathiou, E., Gander, M.J., *Why restricted additive Schwarz converges faster than additive Schwarz*, BIT Numerical Mathematics 43, 945-959, (2003).

10. Gekeler, E., *On the solution of systems of equations by the epsilon algorithm of Wynn*, Mathematics of Computation 26, 427-436, (1972).

11. Leray, J., *La théorie des points fixes et ses applications en analyse*, In Proc. International Congress of Math 2, 202-208, (1950).

12. Lottes, J.W., Fischer, P.F., *Hybrid multigrid/Schwarz algorithms for the spectral element method*, Journal of Scientific Computing 24, 45-78, (2005).

13. Dryja, M., Smith, B.F., *Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions*, SIAM journal on numerical analysis 31, 1662-1694, (1994).

14. Garbey, M., Tromeur-Dervout, D., *On some Aitken-like acceleration of the Schwarz method*, International journal for numerical methods in fluids 40, 1493-1513, (2002).

15. Garbey, M., *Acceleration of the Schwarz method for elliptic problems*, SIAM Journal on Scientific Computing 26, 1871-1893, (2005).

16. Linel, P., *Méthodes de décomposition de domaine en temps et en espace pour la résolution de systèmes d'EDOs non-linéaires*, Thèse de doctorat. Université Claude Bernard-Lyon I, (2011).

17. Linel, P., Tromeur-Dervout, D., *Aitken-schwarz and schur complement methods for time domain decomposition*, In PARCO. IOS Press 19, 75-82, (2010).

18. Nabben, R., Szyld, D.B., *Convergence theory of restricted multiplicative Schwarz methods*, SIAM Journal on Numerical Analysis 40, 2318-2336, (2002).

19. Duminil, S., Sadok, H., Szyld, D.B., *Nonlinear Schwarz iterations with reduced rank extrapolation*, Applied Numerical Mathematics 94, 209-221, (2015).

20. Cai, X.C., Sarkis, M., *A restricted additive Schwarz preconditioner for general sparse linear systems*, Siam journal on scientific computing 21, 792-797, (1999).

21. Cai, X.C., Li, X., *Inexact Newton methods with restricted additive Schwarz based nonlinear elimination for problems with high local nonlinearity*, SIAM Journal on Scientific Computing 33, 746-762, (2011).

Nagid Nabila,
Department of Mathematics,
FST, University of Abdelmalek Essaadi,
Morocco.
E-mail address: nabilanagid@gmail.com

and

Belhadj Hassan,
Department of Mathematics,
FST, University of Abdelmalek Essaadi,
Morocco.
E-mail address: Hassan.belhadj@gmail.com