Available at
http://pvamu.edu/aam
Appl. Appl. Math.
ISSN: 1932-9466

Applications and Applied Mathematics:
An International Journal
(AAM)

Vol. 13, Issue 1 (June 2018), pp. 235 – 247

An Accelerate Process for the Successive Approximations Method In the Case of Monotonous Convergence

[1]A. Laouar and [2]I. Mous

[1]Department of Mathematics
Badji Mokhtar University of Annaba
P.O. Box 12
23000 Annaba, Algeria
abdelhamid.laouar@univ-annaba.dz

[2]Department of Mathematics
University 20 Août 1955
21000 Skikda, Algeria
mousilhem@yahoo.fr

Received: September 5, 2016; Accepted: October 19, 2017

Abstract

We study an iterative process to accelerate the successive approximations method in a monotonous convergence framework. It consists in interrupting the sequence of the successive approximations method produced at the k^{th} iteration and substituting it by a combination of the element of the sequence produced at the iterate $k+1$ and an extrapolation vector. The latter uses a parameter which can be calculated mathematically. We illustrate numerically this process by studying a free-boundary problems class.

Keywords: Accelerate process; Monotonous convergence; Method of successive approximations; Method of super/sub-solution

MSC 2010 No.: 49M25, 65B05, 65N22, 65N55

1. Introduction

This paper deals with the study of an iterative process for speeding up the successive approximations method:

$$u^{k+1} = Tu^k + b,\ k = 0, 1, \ldots,$$

where T is a real $m \times m$ matrix and b, $u^k \epsilon R^m$; in the case of monotonous convergence. This process has initially been proposed by Falcone (1984) & Miellou (1979) for solving a linear system and has been extended afterward by El-Tarazi (1986) for a nonlinear system. It consists in interrupting the iterations of the sequence (u^k), $k = 1,2,...$, produced by the iterative method at the k^{th} step and substituting u^k by $\tilde{u}^k$, where $\tilde{u}^k$ is obtained by combining

$$\text{the vector } u^{k+1} \text{ and the extrapolation vector } u^k + \eta^k(u^k - u^{k-1}),$$

with η^k a real parameter to be calculated. We restart the iterative process with $\tilde{u}^k$ as an initial guess; this sequence $\tilde{u}^k$ is a better approximation to the solution that the iterations $u^{k+1}$and u^k, and moreover, it preserves the monotony properties. In Laouar (1988) and Laouar and Abdelli (2005), it has been applied to solve a discrete quasi-variational inequality system. A hybrid acceleration procedure for nonlinear problems, similar to that described above, was developed by Brezinski and Cheheb (1998). Our primary goal, here, is to improve this process for a large system. For this, we illustrate this process by applying it to a class of free boundary problems (see Bensoussan and Lions (1978), Boulbrachene (1998), Chau, et al. (2017), Zhou (2007)). We look for a function u satisfying

$$(P) \begin{cases} a(u, u-v) \geq (f, v-u)_{L^2(\Omega)}, \\ u \leq M(u),\, v \leq M(u), \end{cases} \tag{1}$$

where $a(.,.)$ is a bilinear form, $(.,.)$ being the inner product in $L^2(\Omega)$, f a regular given function in $L^2(\Omega)$. $M(u)$ represents the obstacles which are very important in the study of the quasi-variational inequality system (the impulsion control problems).

The outline of this paper is as follows. Section 2 describes the accelerative process and gives the algorithms. Section 3 presents a free boundary problem and analogous discrete problem. Last section gives the illustration of the accelerate process by considering two examples.

2. An Accelerate Process

2.1. Preliminaries and theory

In this section we introduce some useful materiel which we used in the sequel.

On R^m, the real $m-$dimensional linear space of column vector $u = (u_1, ..., u_m)^T$.

We consider the natural partial ordering defined by

$$\text{for } u, v \in R^m,\ u < v \Longleftrightarrow u_i < v_i,\ i = 1,...,m,$$
$$\text{for } u, v \in R^m,\ u \leq v \Longleftrightarrow u_i \leq v_i,\ i = 1,...,m.$$

On $L(R^n)$ the linear space of real $m \times m$ matrices, we may consider a partial ordering analogous as follows

$$\text{for } A, C \in L(R^m),\, A < C \Longleftrightarrow a_{i,j} < c_{i,j},\quad i, j = 1, ..., m,$$
$$\text{for } A, C \in L(R^m),\, A \leq C \Longleftrightarrow a_{i,j} \leq c_{i,j},\quad i, j = 1, ..., m.$$

Let $T \in L(R^m)$ and $b \in R^m$, and consider the linear system

$$u = Tu + b, \tag{2}$$

and the method of successive approximations

$$\begin{cases} \text{For any initial vector } u^0 \in R^m / \\ u^{k+1} = Tu^k + b,\ k = 0,1,2,\dots, \end{cases} \tag{3}$$

denoting by u^* the fixed point of Equation (2).

It is known that if $\rho(T) < 1$ ($\rho(T)$ is the spectral radius of T) then, the iterative method (3) converges to the solution u^* for any $u^0 \in R^m$; of course, the convergence is not necessarily monotone (see Laouar and Abdelli (2005)).

Definition 2.1.

In (2), for any $z \in R^m$, we say that

- a sub-solution if $z \leq Tz + b$,
- a super-solution if $Tz + b \leq z$.

Remark.

Note that if the spectral radius $\rho(T)$ is less than unity, then the iteration (3) converges to the solution u^* of (2) for any $u^0 \in R^n$. Furthermore, it is classically known (see Ortega and Rheinboldt (1970) for more general cases) that if

$$T \geq 0,\ \rho(T) < 1 \quad \text{and} \quad Tu^0 + b \leq u^0 \quad (\text{resp. } u^0 \leq Tu^0 + b),$$

then the sequence (3) is decreasing (resp. increasing) in partially ordered space R^m and converges to u^*. Whereas when $\rho(T)$ is close to unity, the convergence of (3) may be very slow.

Consider the following hypothesis

$$(H) \ \begin{cases} T \in L(R^m) \text{ with } T \geq 0, \\ \text{and } \rho(T) \text{ is strictly lower than } 1. \end{cases}$$

Lemma 2.3.

Under the hypothesis (H) and for any $u, v \in R^m$; $Tu + b \leq u$ and $Tv + b \leq v$. Let $z \in R^m$ be defined by $z = \min(u, v)$ $[i.e.\ z_i = \min(u_i, v_i), i = 1,...,m]$. Then, we have

$$Tz + b \leq z.$$

Proof:

By definition of z, $z_i = u_i$ (or $z_i = v_i$) for $i = 1, ..., m$. Since, $z \leq u$ (resp. $z \leq v$) and $t_{i,j} \geqslant 0$ ($t_{i,j}$ the coefficients of the matrix T) then, for $j = 1, ..., m$, we have

$$\sum_{i=1}^{m} t_{i,j}(z_j - u_j) \leq 0 \text{ (resp. } \sum_{i=1}^{m} t_{i,j}(z_j - v_j) \leq 0)$$

Then, for $i = 1, ..., m$, we have

$$(Tz + b - z)_i \leq (Tu + b - u)_i \leq 0 \text{ (resp. } (Tz + b - z)_i \leq (Tv + b - v)_i \leq 0) .$$

Hence, we deduce that

$$Tz + b \leq z.$$

■

Proposition 2.4.

We suppose that the hypothesis (H) is verified. Let $Tu^0 + b \leq u^0$ (resp. $v^0 \leq Tv^0 + b$), then the sequence u^k(resp. v^k) defined by

$$u^{k+1} = Tu^k + b \text{ (resp. } v^{k+1} = Tv^k + b),$$

satisfies

$$u^* \leq ... \leq u^{k+1} \leq u^k \leq ... \leq u^0 \text{ and } \lim_{k\to\infty} u^k = u^*$$

$$\text{(resp. } v^0 \leq ... \leq v^k \leq v^{k+1} \leq ... \leq u^* \text{ and } \lim_{k\to\infty} v^k = u^*).$$

Proof:

We have $\rho(T) < 1$ and $T \geq 0$, then $(I - T)^{-1} \geq 0$. So, $(I - T)$ is M-matrix with $Tu^0 + b \leq u^0$ (resp. $v^0 \leq Tv^0 + b$), we can prove easily this proposition. ■

Theorem 2.5.

Let the hypothesis (H) be verified. Let $Tu^0 + b \leq u^0$ and a parameter η^k $(\eta^k > 0)$ defined by

$$\eta^k = \min_{i \in m} \left\{ \frac{[T(u^{k-1} - u^k)]_i}{[(I - T)(u^{k-1} - u^k)]_i} \right\} = \min_{i \in m} \left\{ \frac{[(I - T)u^k - b]_i}{[(I - T)(u^{k-1} - u^k)]_i} \right\}, \tag{4}$$

where the minimum is taken over all i for which

$$\Big[(I - T)(u^{k-1} - u^k)\Big]_i > 0 \quad \text{and} \quad u^{k+1} = Tu^k + b, \text{ for } k = 1,2,...$$

Then

$$\widetilde{u}^k = u^k + \eta^k(u^k - u^{k-1}),$$

satisfies

$$u^* \leq \widetilde{u}^k \leq u^{k+1} \leq u^k \quad \text{and} \quad T\widetilde{u}^k + b \leq \widetilde{u}^k.$$

Proof:

Under the hypothesis (H), u^k satisfies Proposition 2.4 and $u^{k-1} \leq u^k$ for all k. If $u^{k-1} = u^k$, then $u^k = u^*$ and the proof is obvious.

Now, we suppose that $u^{k-1} \neq u^k$. As $(I-T)^{-1} \geqslant 0$, it follows that $(I-T)(u^{k-1}-u^k)$ has at least one positive component. Thus, the parameter η^k is well defined and positive.

From (4), we have

$$\eta^k \Big[(I-T)(u^k - u^{k-1})\Big]_i \leq \Big[(I-T)u^k - b\Big]_i, \quad \text{for } i = 1,...,n,$$

which is true even if

$$\Big[(I-T)(u^{k-1} - u^k)\Big]_i \leq 0, \quad \text{since} \quad (I-T)u^k - b \geqslant 0.$$

Hence,

$$T\Big[u^k + \eta^k(u^k - u^{k-1})\Big] + b \leq u^k + \eta^k(u^k - u^{k-1}), \text{ with } \quad Tu^{k+1} + b \leq u^{k+1}.$$

By using Lemma 2.3, we have

$$T\widetilde{u}^k + b \leq \widetilde{u}^k, \quad \widetilde{u}^{k+1} \leq u^{k+1} \leq u^k$$

and

$$(I-T)u^* - b \leq (I-T)\widetilde{u}^k - b, \quad \text{with } (I-T) \geq 0.$$

Hence, $u^* \leq \widetilde{u}^k$. So, the proof of Theorem 2.5 is completed. ∎

Theorem 2.6.

Let the hypothesis (H) be verified. Let $v^0 \leq Tv^0 + b$ and a parameter θ^k ($\theta^k > 0$) defined by

$$\theta^k = \min_{i \in m}\left\{\frac{[T(v^{k-1} - v^k)]_i}{[(I-T)(v^{k-1} - v^k)]_i}\right\} = \min_{i \in m}\left\{\frac{[(I-T)v^k - b]_i}{[(I-T)(v^{k-1} - v^k)]_i}\right\}, \tag{5}$$

where the minimum is taken over all i for which

$$\Big[(I-T)(v^{k-1} - v^k)\Big]_i < 0 \quad \text{and} \quad v^{k+1} = Tv^k + b, \quad k = 1,2,...$$

Then

$$\widetilde{v}^k = v^k + \theta^k(v^k - v^{k-1}),$$

satisfies

$$v^k \leq v^{k+1} \leq \widetilde{v}^k \leq u^* \quad \text{and} \quad \widetilde{v}^k \leq T\widetilde{v}^k + b.$$

Proof:

In a similar way of Theorem 2.5, we can easily show Theorem 2.6. ∎

Of course under the assumption (H) of the above Theorems 2.5 and 2.6, we can guarantee two sided error estimate

$$v^k \leq \widetilde{v}^k \leq u^* \leq \widetilde{u}^k \leq u^k,$$

which allows us to obtain an approximation to the exact solution u^* to any desired tolerance.

2.2. Original and accelerated algorithms

Denoting by $Alg1$ the original algorithm and $Alg2$ the accelerated algorithm, we now give a description of the $Alg1$ and $Alg2$, respectively.

Algorithm $Alg1$:

$Step\ 0$: Consider the initial guess u^0, $v^0 \in R^m$ such that

$$u^0 \leq Tu^0 + b \quad \text{and} \quad Tv^0 + b \leq v^0.$$

$Step\ 1$: For $k = 0, 1, ...$

$$\begin{cases} u^{k+1} = Tu^k + b, \\ v^{k+1} = Tv^k + b. \end{cases} \tag{6}$$

Algorithm $Alg2$:

$Step\ 0$: Consider the initial guess u^0, $v^0 \in R^m$ such that

$$u^0 \leq Tu^0 + b \quad \text{and} \quad Tv^0 + b \leq v^0.$$

$Step\ 1$: For $k =$ 1,2,...

$$\begin{cases} u^{k+1} = \max\,[Tu^k + b,\ u^k + \eta^k (u^k - u^{k-1})], \\ v^{k+1} = \min\,[Tv^k + b,\ v^k + \theta^k (v^k - v^{k-1})], \end{cases} \tag{7}$$

where the parameter η^k(resp. θ^k) is calculated by the expression (4) (resp. by (5)).

To solve the iterative systems (6) and (7), we use the Jacobi method [$(I - T) = A$, where $A = a_{i,j}$, $1 \leq i, j \leq m$]; we denote by $D = (d_{i,j})_{1 \leq i,j \leq m}$ a diagonal matrix, where $d_{i,j} = a_{i,j}$, if $i = j$ and $d_{i,j} = 0$ if $i \neq j$, $-E$ $(-F)$ is strictly lower (resp. upper) triangular matrix of the A.

Thus, the algorithm is written as

$$\begin{cases} \text{For any } u^0 \in R^m \text{ arbitrary data,} \\ u^{k+1} = (I_n - D^{-1}(E + F))u^k + D^{-1}b,\ \ k = 0, 1, ... \end{cases} \tag{8}$$

For the convergence test of the algorithm, we use the norm $\| u^k - v^k \|_\infty < \varepsilon$, where ε is a tolerance. The initialization process of vectors u^0 and v^0 is defined by

$$(I - T)u^0 < b \quad \text{(resp. } b < (I - T)v^0).$$

$Step$ 0: We can choose the initial vectors $u^0$and v^0 as follows.

$$u^0 = -v^0 = -(\delta, ..., \delta)^t, \quad \text{where} \quad \delta > \max\left(\frac{\overline{b}}{\underline{\lambda}}, \frac{-\underline{b}}{\underline{\lambda}}\right) > 0,$$

with

$$\underline{\lambda} = \min_i \left(\sum_j a_{i,j}\right) > 0, \quad \underline{b} = \min_i(b_i) \quad \text{and} \quad \overline{b} = \max_i(b_i).$$

3. Class of free boundary problems

In order to present the study at least for a large class of the problems, we take an economic problem from the general examples (we refer to the book of Bensoussan and Lions(1978)). Our objective, here, is limited to give numerical examples for illustrating the accelerative process, so we do not intend to present the complete study of this problem (for more details, see Bensoussan and Lions (1978) and Boulbrachene (1998)).

Indeed, we want to solve a class of the following general problem. Let Ω be a bounded domain of R^m, with regular boundary $\partial\Omega$.

$$(P)\begin{cases} \text{Find } u \text{ such that} \\ a(u, u - v) \geq (f, v - u)_{L^2(\Omega)}, \\ u \leq M(u), v \leq M(u), \end{cases} \tag{9}$$

where

$$a(u, v) = \sum_{i,j=1}^{n} \int_\Omega \sigma_{i,j}(x) \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_j} dx + \sum_{j=1}^{n} \int_\Omega \mu_j(x) \frac{\partial u}{\partial x_j} v dx + \int_\Omega \alpha_0(x) uv dx, \tag{10}$$

the coefficients $\sigma_{i,j}$, μ_j and $\alpha_0 \in L^\infty(\Omega)$, for $i, j = 1, ..., n$, with $\alpha_0 \geqslant 0$, $(.,.)$ being the inner product in $L^2(\Omega)$ and f a regular given function in $L^2(\Omega)$. $M(u)$ represents the obstacles which are very important in the study of the quasi-variational inequalities which invert as impulsion control problems.

The general problem (P) is theoretically well understood from mathematical and analytic point of view (see Bensoussan and Lions (1978)) and can be written explicitly under other form of three sub-problems as follows.

For $i = \overline{1, 3}$, find $u_i \in R^m$ such that

$$\begin{cases} A_i u_i \leq f_i, \\ u_i \leq M(u_i), \\ (A_i u_i - f_i)^t . M(u_i) = 0, \end{cases} \tag{11}$$

where A_1, A_2 and A_3 are three operators which will define in the sequel. $M(u)$ is define by

$$M(u_i) = \inf_{i,j} \left\{\Psi_{i,j} + u_i\right\},$$

where $\Psi_{i,j}$ is the obstacles for $i \neq j$ and $i, j \in \{1, 2, 3\}$.

The economic problem means to adjust the period-time in which a regime is used in order to satisfy the demand at the lowest cost (see Bensoussan and Lions (1978)).

Other formulation of the problem (11) is given by

For $i = \overline{1,3}$, find $u_i \in R^m$ such that

$$\max\left[(A_i u_i - f_i)^t.(u_i - \inf_{i,j}(\Psi_{i,j} + u_i))\right] = 0. \tag{12}$$

The problems (11) are presented more explicitly as follows. Find the respective solutions u_1, u_2, u_3 of the three sub-problems such that

$$(Ps)\begin{cases} (P_1)\begin{cases} A_1 u_1 \leq f_1, \\ u_1 \leq \inf(\Psi_{1,2} + u_2,\ \Psi_{1,3} + u_3), \\ (A_1 u_1 - f_1)^t(u_1 - \inf(\Psi_{1,2} + u_2,\ \Psi_{1,3} + u_3)) = 0, \end{cases} \\ (P_2)\begin{cases} A_2 u_2 \leq f_2, \\ u_2 \leq \inf(\Psi_{2,1} + u_1,\ \Psi_{2,3} + u_3), \\ (A_2 u_2 - f_2)^t(u_2 - \inf(\Psi_{2,1} + u_1,\ \Psi_{2,3} + u_3)) = 0, \end{cases} \\ (P_3)\begin{cases} A_3 u_3 \leq f_3, \\ u_3 \leq \inf(\Psi_{3,1} + u_1,\ \Psi_{3,2} + u_2), \\ (A_3 u_3 - f_3)^t(u_3 - \inf(\Psi_{3,1} + u_1,\ \Psi_{3,2} + u_2)) = 0, \end{cases} \end{cases} \tag{13}$$

with boundaries conditions $u_1 = u_2 = u_3 = 0$ and f_1, f_2 and f_3 given positive functions. For $i \neq j$ and $i, j \in \{1, 2, 3\}$, $\Psi_{i,j}$ is the fixed cost and represents the obstacles (see Bensoussan and Lions (1978)).

Remark.

The procedure of passing from one regime to the other is easily described from (Ps); for example, one pays $\Psi_{1,2}$ and $\Psi_{1,3}$ when we pass from the regime (P_1) to the regime (P_2), but if

$$u_1 \leq \inf(\Psi_{1,2} + u_2,\ \Psi_{1,3} + u_3),$$

is not verified, then one projects.

Similarly, one pays $\Psi_{3,1}$ and $\Psi_{3,2}$ when we pass from the regime (P_2) to the regime (P_3), but if

$$u_2 \leq \inf(\Psi_{2,1} + u_1,\ \Psi_{2,3} + u_3),$$

is not verified, then we project (for more details, we refer to the book of Bensoussan and Lions (1978)).

The operators A_1, A_2 and A_3 are given by

$$\begin{cases} A_1 u_1 = -\sum\limits_{i=1} \sigma_{i,j}(x)\dfrac{\partial^2 u_1}{\partial x_i^2} + \sum\limits_{j=1}^{n} \mu_i(x)\dfrac{\partial u_1}{\partial x_i} + \alpha_0(x)u_1, & (I) \\ A_2 u_2 = -\sum\limits_{i=1} \sigma_{i,j}(x)\dfrac{\partial^2 u_2}{\partial x_i^2} + \alpha_0(x)u_2, & (II) \\ A_3 u_3 = -\sum\limits_{i=1} \sigma_{i,j}(x)\dfrac{\partial^2 u_3}{\partial x_i^2}. & (II) \end{cases} \tag{14}$$

Note that the coefficients $\sigma_{i,j}(x)$, $\mu_i(x)$ and $\alpha_0(x)$ are given regular functions which are taken as constant in numerical experiment.

3.1. Analogous discrete problem

The discrete version of the problem (P) can be written as

$$(P_h) \quad \begin{cases} \text{Find } u_h \text{ a discrete solution such that} \\ a_h(u_h, u_h - v_h) \geq (f_h, v_h - u_h), \\ u_h \leq r_h M_h,\ v_h \leq r_h M_h, \end{cases} \tag{15}$$

where r_h is an usual interpolation operator.

The fixed-point application associated to the problem (P_h) is defined by

$$(T_h) \quad \begin{cases} T_h\colon (L^\infty(\Omega_h))^+ \to V_h, \\ \ w \to T_h w = z_h(w), \end{cases} \tag{16}$$

where V_h is a P_1 finite element space approximating $H^1(\Omega_h)$ and $z_h(w)$ the discrete solution of the following problem:

$$(P_h^{'}) \quad \begin{cases} a_h(z_h(w), v_h - z_h(w)) \geq (f_h, v_h - z_h(w)), \\ z_h(w) \leq r_h M_h(w),\ v_h \leq r_h M_h. \end{cases} \tag{17}$$

4. Numerical examples

For illustrating the considered accelerate process, we give two numerical examples and evaluate the execution time of solution by using the following formula

$$\text{q} = \frac{\text{execution time of the algorithm Alg2}}{\text{execution time of the algorithm Alg1}}.$$

Example 4.1.

The first example treats the particular case of the problem $(14-\ I))$ by taking $M(u)$=Ψ. For this we discretize the problem (P_h) by taking $\Omega\ =]0,1[\times]0,1[$ and choosing a regular mesh of pace h ($h = 1/(m+1)$, $m > 0$). We use a standard discretization by the five point-usual finite difference scheme and obtain the following system:

$$A_h u_h = F_h, \tag{18}$$

where A_h is the matrix of discretization, u_h a solution vector and $F_h = (h^2 f_1, ..., h^2 f_m)^T$.

Taking

$$k_1 = 4(\frac{\sigma_1^2}{2} + \frac{\sigma_2^2}{2}) + \alpha_0 h^2, \qquad k_2 = -\frac{\sigma_1^2}{2} + \frac{\mu_1 h}{2}, \qquad k_3 = -\frac{\sigma_1^2}{2} - \frac{\mu_1 h}{2}, \tag{19}$$

$$k_4 = -\frac{\sigma_2^2}{2} + \frac{\mu_2 h}{2}, \qquad k_5 = \frac{\sigma_2^2}{2} - \frac{\mu_2 h}{2}, \tag{20}$$

where $\sigma_1 = \sigma_2 \approx \sigma_{i,j}(x)$ and $\mu_1 = \mu_2 = \mu_j,$ for $i, j = 1, ..., m.$

The coefficients of the matrix are given by

$$\begin{aligned}
&a_{i,i} = k_1 \quad \text{for } i = \overline{1,m}, \\
&a_{i,i+1} = \begin{cases} k_2 & \text{if } i = \overline{1,m} \quad \text{and} \quad i \neq n, 2n, ..., m-1, \\ 0 & \text{if } i = n, 2n, ..., m-1, \end{cases} \\
&a_{i,i-1} = \begin{cases} k_3 & \text{if } i = \overline{2,m} \quad \text{and} \quad i \neq n, 2n, ..., m-1, \\ 0 & \text{if } i = n, 2n, ..., m-1, \end{cases} \\
&a_{i,i+n} = k_4 \quad \text{and} \quad a_{i+n,i} = k_5 \quad \text{for } i = 1, ..., m-1.
\end{aligned}$$

We note that the matrix A_h has a multi-diagonal structure and does not symmetric. To solve the system (18), we use the Jacobi method and project on convex space ($i.e.$, if the iteration $u^k \leq r_h \Psi_h$ and $v^k \leq r_h \Psi_h$, otherwise, we take the iteration u^k and v^k equal to the obstacle, respectively). Thus, we calculate the sub-solution and super-solution. Therefore we are evaluated the cpu execution time in one percent of a second of the original Algorithm $Alg1$ before the acceleration and the execution time of the accelerated Algorithm $Alg2$.

To simulate, we take the following data $\sigma_1 = \sigma_2 = \sqrt{2}$, $\alpha_0 = 100,$ $\mu_1 = \mu_2 = 0.01$, Ψ a positive constant and tolerance $\varepsilon = 10^{-14}$.

We present some results in the following tables.

Table 1. The execution time of the Algorithm Alg1.

m	Iterations	Time cpu (one percent of a second)
64	65	36.8000
169	141	414.960
225	180	896.440
289	202	1272.020

Table 2. The execution time of the Algorithm Alg2.

m	Iterations	Time cpu (one percent of a second)
64	30	26.5600
169	69	256.560
225	88	340.760
289	93	658.670

Note that the iterations number cannot be considered here, as a criterion of comparison because the Algorithm $Alg2$ is different of the Algorithm $Alg1$.

Now, we represent graphically the above result of the tables (see Figure 1).

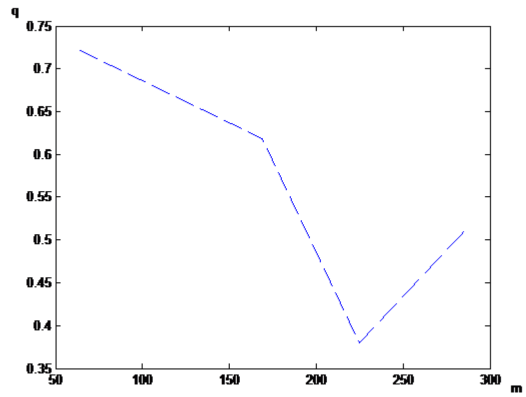

Figure 1. The graph $q = f(m)$.

We remark that the situation, here, is a more favorable that in El-Tarazi (1986) and Laouar and Abdelli (2005), since the result obtained in Tables 1 and 2, shows that there is a better acceleration of convergence (for example $m = 64$, we gain approximately 30% cpu time). We notice that when m increases, the gain in time can reach up to 50%.

Example 4.2.

In the second example, we consider the general problem (13) in three dimensional (i.e., $\Omega =]0,1[\times]0,1[\times]0,1[$). For the discretization of the problems $(14, I)$, $(14, II)$ and $(14, III)$, we use a standard discretization (the nine point-usual finite difference scheme). Thus, we obtain the corresponding systems, respectively.

$$\begin{cases} A_h^{(1)} u_h = F_h^{(1)}, & (I_h) \\ A_h^{(2)} v_h = F_h^{(2)}, & (II_h) \\ A_h^{(3)} w_h = F_h^{(3)}, & (III_h) \end{cases} \tag{21}$$

where $A_h^{(1)}$, $A_h^{(2)}$ and $A_h^{(3)}$ are the discretization matrices of the operators A_1, A_2 and A_3, respectively; $u_{h,}$ v_h and w_h are the solutions vectors of (I_h), (II_h) and (III_h); $F_h^{(1)}$, $F_h^{(2)} F_h^{(3)}$ are the second members.

We solve the systems $(19,\ I_h)$, $(19,\ II_h)$ and $(19,\ III_h)$ by Gauss-Seidel method which is presented as follows.

For $i = \overline{1, m}$, we have

$$\begin{cases} \widetilde{u}_i^{(k+1)} = -\sum\limits_{i<j} a_{i,j} u_i^{(k)} - \sum\limits_{i>j} a_{i,j} u_i^{(k+1)} + F_h^{(i)}, \\ & k = 0,1,2,..., \\ w_i^{(k+1)} = \underset{V}{proj}\ \widetilde{u}_i^{(k+1)}, \end{cases} \tag{22}$$

where

$$V = \left\{ u_i^{(k+1)} / u_i^{(k+1)} \leq \inf(\Psi_{i,j} + v_i^{(k)},\, \Psi_{i,j} + w_i^k) \right\},$$

with v_i^k, w_i^k the calculated iterations for the systems $(19,\ II_h)$ and $(19,\ III_h)$, respectively.

For simulation, we have taken the obstacles $\Psi_{1,2} = \Psi_{2,1} = 3$, $\Psi_{1,3} = \Psi_{3,1} = 4$ and $\Psi_{2,3} = \Psi_{3,2} = 2.5$; the coefficients $\sigma_{i,j} = 2$, $\mu_j = 1$ and $\alpha_0 = 4$; the initial vectors $u^{(0)} = v^{(0)} = w^{(0)} = 1,7$; the tolerance $\varepsilon = 10^{-8}$.

We present some results in the following tables:

Table 3. The execution time of the Algorithm Alg1.

$\mathrm{h} = 1/(\mathrm{m}+1)$	Iterations	Time cpu (second)
$1/8$	13	3.06
$1/11$	12	4.44
$1/16$	15	12.72
$1/32$	45	30.11

Table 4. The execution time of the Algorithm Alg2.

$\mathrm{h} = 1/(\mathrm{m}+1)$	Iterations	Time cpu (second)
$1/8$	12	2.55
$1/11$	8	3.77
$1/16$	9	11.73
$1/32$	30	25.09

As we said previously the iterations number cannot be considered, here, as a criterion of comparison because the Algorithm $Alg2$ is not identical to the Algorithm $Alg1$.

5. Remarks and Conclusion

We remark that our result is a more favorable that in Laouar and Abdelli (2005) & El-Tarazi (1986), since the result obtained, in Tables 1 and 2, shows that there is a better acceleration of convergence (for example $m = 64$, we gain approximately 30% cpu time). We notice that when m increases, the gain in time can reach up to 50%. The results in Tables 3 and 4 show that there is an acceleration of convergence. One gains approximately 20% of the time with respect to the time of the original Algorithm $Alg1$. However the situation is a less favorable as in Example 1. The gain of time is approximately similar to the obtained results in El-Tarazi (1986) & Laouar and Abdelli (2005).

This study introduces a process of convergence acceleration as a numerical algorithm for a class of free boundary problems, where the problem of the obstacle is representative. We were only concerned with setting this acceleration process to test whether it is robust. Indeed, important theoretical questions concerning the sensitivity of this process to the optimal choices of the relaxation parameter were not addressed in this study. This process can be extended to free parabolic boundary problems (the Stefan problem) and optimal control problems of elliptic type with constraints on state and / or control.

Acknowledgement:

The authors are very grateful to the editor and the referees for carefully reading the paper and for their comments and suggestions, which have improved the paper in the present form.

REFERENCES

Bensoussan, A. and Lions, J. L. (1978). Applications des inéquations variationnelles en contrôle stochastique. Ed. Dunod, Paris.

Boulbrachene, M. (1998). The Noncoercive Quasi-Variational Inequalities Related to Impulse Control Problems, Jour. Computers Math. Applic., Vol. 35, No. 12, pp. 101–108.

Brezinski, C. and Chehab, C. (1998). Nonlinear Hybrid Procedure and Fixed Point Iterations, Numer. Funct. Anal. Optimization, pp. 465–487.

Chau, M., Laouar, A., Garcia, Th. and Spiteri, P. (2017). Grid Solution of Problem with Unilateral Contraints, Journal of Numerical Algorithms, Springer.

El-Tarazi, M.N. (1986). On a monotony preserving accelerator process for the successive approximations method. IMA Journal of Numerical Analysis, Vol. 6, pp. 439–446.

Falcone, M. (1984). A numerical approach to infinite horizon problem of deterministic control theory. Conf. CNR-GNAFA, Italy.

Laouar, A. (1988). Aspects de l'analyse numérique de méthodes itératives de point fixe: Erreurs d'arrondi, Accélération de convergence, Sous-domaines. Ph. D. Thesis, Université de Besançon, France.

Laouar, A. and Abdelli, J. (2005). Une méthode d'accélération de convergence appliquée à l'algorithme des approximations successives pour les la résolution des grands systèmes. Acte du 2ème colloque sur "Tendance Mathématiques Appliquées "Tam-Tam05,Tunisia, pp. 341-347.

Laouar, A., Guerziz, A. and Boussaha, A. (2016). Calculation of eigenvalues of Sturm-Liouville equation for simulating hydrodynamic soliton generated by a piston wave maker. Applied Math., SpringerPlus.

Miellou, J.C. (1979). Extrapolation aggregation of monotone kind. Application to one "obstacle's" stationary problems. Proceedings of intensive seminary one free boundary value problems. Pavia, Ed. Magenes, pp. 411–437.

Ortega, J. M. and Reinboldt, W. C (1970). Iterative Solution on Non Linear Equations in Several Variables. Academic Press, New York.

Zhou, S. and Zou, Z. (2007). An iterative algorithm for a quasivariational inequality system related to HJB equation. Jour. of Comp. and Appl. Math., Elsevier.