Asynchronous Schwarz alternating methods with flexible communication for the obstacle problem

P. SPITERI [*] **— J.-C. MIELLOU** [**] **— D. EL BAZ** [***]

[*] *ENSEEIHT-IRIT, LIMA, Institut National Polytechnique de Toulouse, 2, Rue Camichel, 31071 Toulouse Cedex, France*
email: Pierre.Spiteri@enseeiht.fr

[**] *Département de Mathématiques, U.F.R. Sciences et Techniques, Université de Franche-Comté, 16, Route de Gray, 25030 Besançon Cedex, France*
email: Miellou@Math.Univ-FComte.FR

[***] *LAAS du CNRS, 7 avenue du Colonel Roche, 31077 Toulouse Cedex 4, France*
email : elbaz@laas.fr

ABSTRACT. *This paper deals with parallel asynchronous subdomain algorithms with flexible communication for the numerical solution of the obstacle problem. The Schwarz alternating method is studied by using the discrete maximum principle and a connection is made with M-functions for the classical obstacle problem. Finally, a computational experience on a shared memory multiprocessor is presented and analyzed.*

RÉSUMÉ. *Nous considérons des méthodes de sous-domaine asynchrones avec communication flexible pour la résolution du problème de l'obstacle. La méthode alternée de Schwarz est étudiée en utilisant le principe du maximum discret. Le lien est fait avec les M-fonctions pour le problème de l'obstacle. Des résultats expérimentaux obtenus sur un multiprocesseur à mémoire partagée sont présentés et analysés*

KEYWORDS: *parallel iterative algorithms, asynchronous iterations, Schwarz alternating method, subdomain methods, boundary value problems, obstacle problem.*

MOTS-CLÉS : *algorithmes itératifs parallèles, itérations asynchrones, méthode alternée de Schwarz, méthodes de sous-domaines, problèmes aux limites, problème de l'obstacle.*

1. Introduction

The numerical solution of boundary value problems leads typically to large scale algebraic systems. Parallel numerical algorithms seem well adapted to an efficient solution of this type of problems, due to the large amount of computation which is required. In particular, subdomain methods which are classical for the solution of boundary value problems are naturally well suited to parallel computing. Among subdomain methods, one can consider methods with overlapping such as the Schwarz alternating method which has been studied in [BEN97], [BOU88], [CHA89], [DIN80], [DIN81], [EVA86], [EVA91], [GIR91a], [LIO89], [MIE82], [MIE86b] and [MIE91] or methods without overlapping (see [EL 84], [GIR91b], [GIR92], [MIE85a], [SPI83] and [SPI86]). The reader is referred to [HOF93] for a study of parallel Schwarz alternating methods applied to the solution of diffusion and convection-diffusion problems.

In this study, we present original parallel asynchronous Schwarz alternating methods with flexible communication whereby the current value of the components of the iterate vector can be used at any time and without any fixed rule. This class of algorithms was first proposed in [MIE98]. The reader is also referred to [EL 96] for application of this new parallel scheme of computation to optimization problems in a weaker context (see also [FRO98]). Note that one of the interests of the proposed methods is to present very good efficiency without considering particular load balancing techniques (see in particular [EL 96]). The very good performance is due to the non synchronization of the computational tasks and to flexible communication, these features being particularly important in the case of massive parallelism. Another original feature of this class of methods is the use of $\mathcal{A}$-supermappings (i.e. approximate block-relaxation procedures for the solution of systems of equations: $\mathcal{A}(y) = 0$). The concept presented here extends the principle of standard asynchronous iterative methods studied in [BAR82], [BAU78], [BER89], [CHA69], [EL 84], [EL 90], [FRO91], [FRO00], [MIE75a], [MIE75b], [MIE86a], [MIE90], [ROB75] and [SZY98].

In this paper, we concentrate on M-functions as defined by Rheinboldt in [RHE70]. This study uses the discrete maximum principle whereby monotone decreasing sequences of vectors are generated and the convergence of parallel asynchronous iterations with flexible communication can be proved. Note that the convergence of such parallel schemes of computation has been shown by using direct verification in [MIE98] for various methods in the M-function context and problems of the type: $\mathcal{A}(y) = 0$. In this paper, we study general fixed point methods and establish first their monotone convergence in the sequential case. Then, we use the sequential convergence result in order to show that the fixed point mapping considered for the solution of the obstacle problem is an $\mathcal{A}$-supermapping and we derive a new convergence result for asynchronous iterations with flexible communication.

In this context, nonlinear boundary value problems are solved via a combination of the aforementioned asynchronous algorithms with flexible communication and the Schwarz alternating method. More precisely, we consider the obstacle problem which

occurs for example in finance and mechanics:

$$\begin{cases} -\Delta u + b\frac{\partial u}{\partial x} + c\frac{\partial u}{\partial y} + du - f)(u-\psi) = 0 \ \text{ in } \ \Omega, \ d \geq 0, \\ -\Delta u + b\frac{\partial u}{\partial x} + c\frac{\partial u}{\partial y} + du \leq f \ \text{ and } \ u \leq \psi \ \text{ in } \ \Omega, \\ u_{|\partial\Omega} = 0. \end{cases} \tag{1.1}$$

A characterization of the solution of such a problem can be given by:

$$\begin{cases} \text{Sup } (-\Delta u + b\frac{\partial u}{\partial x} + c\frac{\partial u}{\partial y} + du - f, u - \psi) = 0, \text{ everywhere in } \Omega, \\ u_{|\partial\Omega} = 0. \end{cases} \tag{1.2}$$

The reader is referred to [SPI83] and [MIE85b] for different convergence results of standard asynchronous iterations applied to the solution of the obstacle problem by using contraction techniques. Note also that problem (1.1) can be rewritten as the following general Hamilton-Jacobi-Bellman problem:

$$\begin{cases} \text{Max } (A^1 u - f^1, A^2 u - f^2) = 0 \quad \text{ in } \Omega, \\ u_{|\partial\Omega} = 0, \end{cases} \tag{1.3}$$

with $A^2 =$ identity and $f^2 = \psi$.

Nevertheless, the convergence of the respective numerical methods for problems (1.1) and (1.3) does not follow from the same arguments since the discretization matrix can be reducible in the former case, whereas it is irreducible in the latter case.

For the obstacle problem and a convenient discretization, we show that an M-function and a suitable approximate fixed point mapping can be obtained from the system of equations and that the convergence of asynchronous iterations with flexible communication can be deived. Finally, we present and analyze computational results for parallel synchronous and asynchronous iterations with flexible communication carried out on a shared memory multiprocessor IBM 3090-600 with 6 vector processors.

Section 2 deals with generalities concerning M-functions. Asynchronous iterations with flexible communication are briefly presented in Section 3, where a connection is also made with the Schwarz alternating method. In Section 4, a general class of fixed point methods is proposed and application to the nonlinear boundary value problem (1.1) is considered. Section 5 deals with the implementation of parallel algorithms on a shared memory multiprocessor and the analysis of experimental results.

2. Mathematical background

Throughout the paper R^n will denote the n-dimensional real linear space of column vectors and $\mathcal{A}$ a continuous mapping from R^n onto itself. Let $\mathcal{A}$ be a surjective M-function (see [ORT70] and [RHE70]). Then, the mapping $\mathcal{A}$ is off-diagonally monotone decreasing and inverse monotone increasing. We note that the surjectivity

of M-functions can be characterized by means of order-coercivity (see [RHE70]). We consider the solution of the following system of equations:

$$\mathcal{A}(y) = 0. \tag{2.1}$$

Under the above assumptions, problem (2.1) has a unique solution y^* (see [ORT70] and [RHE70]). Let R^n be denoted also by E and consider the following decomposition of E, $E = \prod_{i=1}^{\alpha} E_i$, where α is a positive integer, $E_i = R^{n_i}$, and $\sum_{i=1}^{\alpha} n_i = n$. Each subspace E_i is endowed with the natural partial ordering (i.e. component by component) associated with the cone $K_i = R_+^{n_i}$ of vectors with non negative components in R^{n_i}. Let $w \in E$ and consider the following block-decomposition of w :

$$w = \{w_1, \ldots, w_i, \ldots, w_\alpha\} \in \prod_{i=1}^{\alpha} E_i,$$

and the corresponding block-decomposition of $\mathcal{A}$:

$$\mathcal{A}(w) = \{\mathcal{A}_1(w), \ldots, \mathcal{A}_i(w), \ldots, \mathcal{A}_\alpha(w)\} \in \prod_{i=1}^{\alpha} E_i.$$

For all $i \in \{1, \ldots, \alpha\}$, we introduce the following mapping from E_i onto itself :

$$y_i \to \mathcal{A}_i(y_i; w) = \mathcal{A}_i(w_1, \ldots, w_{i-1}, y_i, w_{i+1}, \ldots, w_\alpha).$$

Since $\mathcal{A}$ is a continuous, surjective M-function, it follows from the Theorem 3.5 in [RHE70] that for all $i \in \{1, \ldots, \alpha\}$ and all $w \in E$, the mapping $y_i \longrightarrow \mathcal{A}_i(y_i; w)$, is a continuous, surjective M-function. Moreover, for all $i \in \{1, \ldots, \alpha\}$ and $w \in E$, the system:

$$\mathcal{A}_i(z_i; w) = 0, \tag{2.2}$$

has a unique solution $\hat{z}_i$. Hence, we can define a fixed point mapping $F : E \to E$, associated with problem (2.1) such that

$$F(w) = \hat{z} = \{\hat{z}_1, \ldots, \hat{z}_i, \ldots, \hat{z}_\alpha\}. \tag{2.3}$$

The mapping F is univoquely defined. We can formulate now a very important property of the mapping F (see [MIE86a]). The mapping F is isotone on E (i.e. for all $x, y \in E$ such that $x \leq y$, $F(x) \leq F(y)$).

3. $\mathcal{A}$-supermappings and asynchronous iterations with flexible communication

3.1. *Asynchronous iterations with flexible communication*

In order to solve problem (2.1), we will consider general fixed point iterative methods. The following concept permits one to define a set of initial points for the iterative

algorithms.

Definition 3.1: A vector $y \in R_+^n$ is an $\mathcal{A}$-supersolution if $\mathcal{A}(y) \geq 0$.

Throughout the paper, asynchronous sequences are studied by using the concept of order intervals which is defined as follows. Let $x_i, y_i \in E_i$ such that $x_i \leq y_i$, then the order interval $\langle x_i, y_i \rangle_i$ is defined as $\{z_i \in E_i \mid x_i \leq z_i \leq y_i\}$ (and similarly for $\langle x, y \rangle$, $x, y \in E$).

We introduce now a particular class of fixed point mappings which is used in the definition of asynchronous iterative methods with flexible communication.

Definition 3.2: Let $\mathcal{A}$ be an M-function. $F^{\mathcal{A}}$ is an $\mathcal{A}$-supermapping associated with F if for all $i \in \{1, \ldots, \alpha\}$ and $y \in E$ such that $\mathcal{A}_i(y) \geq 0$, there exists $F_i^{\mathcal{A}}(y) \in E_i$ such that $F_i^{\mathcal{A}}(y) \leq y_i, \mathcal{A}_i(F_i^{\mathcal{A}}(y); y) \geq 0$, and $F_i^{\mathcal{A}}(y) \neq y_i$ if $F_i(y) \neq y_i$.

The following concepts are central in the study of asynchronous iterations.

Definition 3.3: A steering of block-components of the iterate vector, is a sequence $S = \{s(p)\}, p \in N$, such that $s(p) \in \{1, \ldots, \alpha\}$, for all $p \in N$, where N denotes the set of natural integers.

Definition 3.4: A sequence $R = \{r(p)\}$ of maximum delays in the access to the updated components of the iterate vector is such that

$$r(p) = (r_1(p), \ldots, r_i(p), \ldots, r_\alpha(p)) \in N^\alpha, \text{ for all } p \in N,$$

where:

$$0 \leq r_i(p) \leq p \text{ and } r_i(p) = 0 \text{ if } i = s(p), \text{ for all } i \in \{1, \ldots, \alpha\} \text{ and } p \in N.$$

We introduce now the mappings $\rho_i : N \to N,\ i \in \{1, \ldots, \alpha\}$, defined by

$$\rho_i(p) = p - r_i(p), \text{ for all } i \in \{1, \ldots, \alpha\} \text{ and all } p \in N.$$

We have: $0 \leq \rho_i(p) \leq p$ and $\rho_i(p) = p$ if $i = s(p)$. Hence, we define the mapping $\rho : N^\alpha \to N^\alpha$ with components $\rho_i(p), i \in \{1, \ldots, \alpha\}$ and the sets $K_i^p = \{k \in N \mid s(k) = i, 0 \leq k < p\},\ i \in \{1, \ldots, \alpha\}, p \in N$. The set K_i^p contains all the iteration numbers less than p which are relative to the computation of the i-th block-component.

Remark 3.1: The set $\{K_i^p\}$ is a denumerable sequence of finite elements of the power set of N.

Definition 3.5: The general class of asynchronous iterative methods with flexible communication is defined recursively as follows. For all $p \in N$ and $i \in \{1, \ldots, \alpha\}$, we have:

$$\begin{cases} y_i^{p+1} = F_i^{\mathcal{A}}(\tilde{y}^p) \text{ if } i = s(p), \\ y_i^{p+1} = y_i^p \text{ if } i \neq s(p), \end{cases} \tag{3.1}$$

where S and R are defined according to Definitions 3.3 and 3.4, respectively,

$$\tilde{y}^0 = y^0 \text{ is an } \mathcal{A}\text{-supersolution}, \tag{3.2}$$

$$\tilde{y}^p \in \langle y^p, \min\left(y^{\rho(p)}, \tilde{y}^q\right)\rangle \text{ if } p \geq 1, \tag{3.3}$$

$y^{\rho(p)}$ denotes the vector of E with block-components $y_i^{\rho_i(p)}, i \in \{1, \ldots, \alpha\}$ and $q = \mathrm{Max}_{k \in K_{s(p)}^p} k$.

Remark 3.2: In the particular case where $K_i^p = \emptyset$, we have: $\tilde{y}^p \in \langle y^p, y^{\rho(p)}\rangle$.

Remark 3.3: Asynchronous iterative algorithms with flexible communication defined recursively by (3.1) to (3.3) are general iterative methods whereby iterations are carried out in parallel by up to α processor without any order nor synchronization. The main feature of this class of iterative methods is to allow very flexible communication between the processors. In a typical update of the i-th block-component of the iterate vector at iteration $k+1$, all the values $\tilde{y}_j^p$ of the block-components of the iterate vector are taken anywhere in the order interval $\langle y_j^p, \min\left(y_j^{\rho_j(p)}, \tilde{y}_j^q\right)\rangle_j$, where $\tilde{y}_j^q$ was the value used in the last update of the i-th block-component and $y_j^{\rho_j(p)}$ models the nondeterministic behavior of the iterative scheme. Thus, the values of the block-components of the iterate vector which are used in a computation may come from updates which are in progress. A processor can communicate the current value of the relaxed components at any time. On the other hand, all the processors use the most recent value of the components of the iterate vector. We note that the values of the components of the same block of the iterate vector which are used in a typical update, can be relative to different iteration numbers as opposed to the classical case (see [BAU78], [BER89], [CHA69] and [MIE75a]). Another original feature of the new class of methods presented in this paper is the use of $\mathcal{A}$-supermappings $F^{\mathcal{A}}$, which permits one to approximate the solution obtained by relaxation steps.

Remark 3.4: The classical formulation of asynchronous iterations (see [BAU78], [BER89], [CHA69] and [MIE75a]) carries over without difficulty to the framework

of Definitions 3.3 to 3.5. As a matter of fact, in the classical case we have $s(p) \subset \{1, 2, \ldots, \alpha\}$. We note that we can sequentialize the beginning of the computations of the block-components y_i^p, $i \in s(p)$, since the updates of the different block-components are performed independently. However the delays must be modified accordingly. For more details the reader is referred to [MIE98].

We recall now an important result (see [MIE98]).

Proposition 3.1: Let $\mathcal{A}$ be a continuous surjective M-function, F the fixed point mapping associated with $\mathcal{A}$ defined by (2.2) and (2.3), $F^{\mathcal{A}}$ an $\mathcal{A}$-supermapping associated with F, $y^0 \in E$ an $\mathcal{A}$-supersolution, S and R a steering and a sequence of delays, respectively. Then, the asynchronous iteration $\{y^p\}$ given by (3.1) to (3.3) is well defined and satisfies:

$$y^p \downarrow \overline{y}, p \to \infty, \tag{3.4}$$

where $\overline{y}$ is an $\mathcal{A}$-supersolution of problem (2.1) and (3.4) means:

$$\overline{y} \leq \cdots \leq y^{p+1} \leq y^p \leq \cdots \leq y^0 \text{ and } \lim_{p \to \infty} y^p = \overline{y}.$$

3.2. *A particular class of $\mathcal{A}$-supermappings*

We introduce an order relation between $\mathcal{A}$-supermappings.

Definition 3.6: Two $\mathcal{A}$-supermappings $F^{\mathcal{A}}$ and $F^{\mathcal{B}}$ associated with F satisfy the relation $F^{\mathcal{A}} \alpha F^{\mathcal{B}}$, if for all $i \in \{1, \ldots, \alpha\}$ and $y \in E$ such that $\mathcal{A}_i(y) \geq 0$, we have $F_i^{\mathcal{B}}(y) \in \langle F_i^{\mathcal{A}}(y), y_i \rangle_i$.

We concentrate now on a particular class of $\mathcal{A}$-supermappings.

Definition 3.7: $F^{\mathcal{A}}$ is an $\mathcal{M}$-continuous $\mathcal{A}$-supermapping associated with F if there exists an $\mathcal{A}$-supermapping $F^{\mathcal{B}}$ associated with F such that $F^{\mathcal{A}} \alpha F^{\mathcal{B}}$ and

$$y^p \downarrow y^*, p \to \infty, \text{ implies } F_i^{\mathcal{B}}(y^p) \downarrow F_i^{\mathcal{B}}(y^*), p \to \infty \text{ for all } i \in \{1, \ldots, \alpha\}. \tag{3.5}$$

Remark 3.5: The relation (3.5) can be interpreted as a property of continuity at point y^* of the mapping $F^{\mathcal{B}}$ with respect to the partial ordering.

We recall now a global convergence result for asynchronous iterative methods with flexible communication associated with $\mathcal{M}$-continuous $\mathcal{A}$-supermappings (see [MIE98]).

Proposition 3.2: Let assumptions of Proposition 3.1 hold and $F^{\mathcal{A}}$ be an $\mathcal{M}$-continuous $\mathcal{A}$-supermapping associated with F. Assume that the steering S satisfies:

$$\{p \in N \mid i \in s(p)\} \text{ is infinite, for all } i \in \{1, \ldots, \alpha\}, \tag{3.6}$$

and assume also that we have:

$$\lim_{p \to \infty} \rho_i(p) = +\infty, \text{ for all } i \in \{1, \ldots, \alpha\}. \tag{3.7}$$

Then the sequence $\{y^p\}$ defined by (3.1) to (3.3) satisfies $y^p \downarrow y^*$, where y^* is the unique solution of problem (2.1).

3.3. *Link with the Schwarz alternating method*

We concentrate now on the Schwarz alternating method which is commonly used for the solution of boundary value problems. This method is also well suited to parallel computing and represents an interesting domain of application of the above theoretical study. The mapping $\mathcal{A}$ being a surjective M-function, we study the solution of the following nonlinear simultaneous equations:

$$\mathcal{A}(x) = 0, \tag{3.8}$$

via an asynchronous subdomain method derived from the Schwarz alternating method. The augmentation process of the Schwarz alternating method transforms the nonlinear mapping $\mathcal{A}$ into a nonlinear mapping $\tilde{\mathcal{A}}$ which is also a surjective M-function and the above study can be applied (see [MIE98]).

4. Application to nonlinear partial differential equations

In this Section, our original theoretical contribution to the convergence of asynchronous iterations with flexible communication is presented. First a general approach for building approximate block iterative methods is proposed. Then, a monotone convergence result which is of primary importance for deriving A-supermappings is given. An original application of the general study to the obstacle problem is presented. Finally, a convergence result for asynchronous iterations with flexible communication applied to the obstacle problem is derived.

4.1. ***Results on monotone iterative methods***

Let $\mathcal{A}$ be a continuous surjective M-function from R^n onto itself. Then, it follows from Theorem 3.3 in [RHE70] that the system of equations: $\mathcal{A}(y) = 0$ has a unique solution denoted by y^*. Moreover, let x be an initial point, we define the vectors $s, v, r^0, z^0 \in R^n$ by:

$$s_i = \min(\mathcal{A}_i(x), 0),\ v_i = \max(\mathcal{A}_i(x), 0),\ \text{ for all } i \in \{1, ..., n\},$$
$$r^0 = \mathcal{A}^{-1}(s),\ z^0 = \mathcal{A}^{-1}(v).$$

Thus, we have:

$$\mathcal{A}(r^0) \leq \mathcal{A}(y^*) \leq \mathcal{A}(z^0), \tag{4.1}$$

and by the inverse isotonicity of $\mathcal{A}$, we have:

$$r^0 \leq y^* \leq z^0.$$

Moreover, assume that there exists a mapping $C : \langle r^0, z^0 \rangle \to \mathcal{L}(R^n)$, such that for all $x, z \in R^n$ satisfying $r^0 \leq x \leq z \leq z^0$, we have:

$$\mathcal{A}(z) - \mathcal{A}(x) \leq C(z).(z - x), \tag{4.2}$$

$$x \leq z \quad \text{implies} \quad C(x) \leq C(z), \tag{4.3}$$

and

$$C^{-1}(z) \geq 0, \text{ for all } z \in \langle r^0, z^0 \rangle. \tag{4.4}$$

Remark 4.1: We note that the relation (4.3) implies the inequality (4.2) if we have:

$$\mathcal{A}(z) - \mathcal{A}(x) = C(x + t(z - x)).(z - x), \text{ with } t \in]0, 1[.$$

The above equality is satisfied in particular when $\mathcal{A}$ is G-differentiable and $C(x) = \mathcal{A}'(x)$.
We introduce now a mapping ρ' from N into N such that

$$0 \leq \rho'(k) \leq k, k = 0, 1, ... \tag{4.5}$$

and we consider the algorithm

$$z^{k+1} = z^k - C^{-1}(z^{\rho'(k)}).\mathcal{A}(z^k), k = 0, 1, \ldots \tag{4.6}$$

We can state the following convergence result.

Proposition 4.1: Assume that the hypotheses (4.2) to (4.5) hold. Then, the sequence $\{z^k\}$ defined by (4.6) satisfies $z^k \downarrow y^*,\ k \to \infty$, where y^* is the solution of $\mathcal{A}(y) = 0$.

Proof: We proceed mainly by induction. It follows from (4.1) and (4.4) that $\mathcal{A}(z^0) \geq 0$ and $C^{-1}(z^0) \geq 0$. Hence, $z^1 - z^0 = -C^{-1}(z^0).\mathcal{A}(z^0) \leq 0$. If y^* is such that $A(y^*) = 0$, then it follows from the inequality (4.2) that

$$\begin{aligned} y^* &= y^* - C^{-1}(z^0).\mathcal{A}(y^*) = z^1 - (z^0 - y^*) + C^{-1}(z^0).(\mathcal{A}(z^0) - \mathcal{A}(y^*)) \\ &\leq z^1 - (I - C^{-1}(z^0).C(z^0))(z^0 - y^*) = z^1. \end{aligned}$$

It follows from the inequality (4.2), (4.5) and (4.6) that

$$\mathcal{A}(z^1) \geq \mathcal{A}(z^0) + C(z^0).(z^1 - z^0) = 0.$$

We consider now the following induction hypotheses:

$$y^* \leq \ldots \leq z^k \leq z^{k-1} \leq \ldots \leq z^0, \tag{4.7}$$

$$\mathcal{A}(z^k) \geq 0. \tag{4.8}$$

It follows from (4.6), (4.5), (4.7), (4.4), and (4.8) that

$$z^{k+1} - z^k = -C^{-1}(z^{\rho'(k)}).\mathcal{A}(z^k) \leq 0. \tag{4.9}$$

Moreover it follows from (4.5), (4.7), and (4.3) that $z^k \leq z^{\rho'(k)}$ implies $C(z^k) \leq C(z^{\rho'(k)})$. From the inequality (4.4) it results that

$$C^{-1}(z^{\rho'(k)}).C(z^k) \leq I \text{ and } C(z^k).C^{-1}(z^{\rho'(k)}) \leq I. \tag{4.10}$$

Hence, it follows from the equality (4.6) and the inequalities (4.7), (4.2) and (4.10) that

$$\begin{aligned} y^* &= y^* - C^{-1}(z^{\rho'(k)})\mathcal{A}(y^*) = z^{k+1} - (z^k - y^*) + C^{-1}(z^{\rho'(k)}).(\mathcal{A}(z^k) - \mathcal{A}(y^*)) \\ &\leq z^{k+1} - (I - C^{-1}(z^{\rho'(k)}).C(z^k)).(z^k - y^*) \leq z^{k+1}. \end{aligned}$$

Thus, the inequality (4.7) is satisfied at rank $k+1$.
Furthermore, it follows from the inequality (4.2) that $\mathcal{A}(z^{k+1}) \geq \mathcal{A}(z^k) + C(z^k)(z^{k+1} - z^k)$. Hence, it results from (4.6), (4.10), and (4.8) that

$$\mathcal{A}(z^{k+1}) \geq (I - C(z^k).C^{-1}(z^{\rho'(k)}))\mathcal{A}(z^k) \geq 0.$$

Thus, the inequality (4.8) is satisfied at rank $k+1$ and the induction is complete. It follows from (4.7) that there exists a supersolution $\overline{z} \in R^n$, such that $y^* \leq \overline{z}$ and $z^k \downarrow \overline{z}, k \to \infty$, i.e.

$$\lim_{k \to \infty} (z^k - \overline{z}) = 0. \tag{4.11}$$

It follows from the equality (4.6) that

$$C(z^{\rho'(k)})(z^{k+1} - z^k) + \mathcal{A}(z^k) = 0. \tag{4.12}$$

It follows from (4.8), (4.12), (4.7), and (4.3) that

$$0 \leq \mathcal{A}(z^k) \leq C(z^0)(z^k - z^{k+1}).$$

Hence, it follows from (4.11) that $\lim_{k \to \infty} \mathcal{A}(z^k) = \mathcal{A}(\overline{z}) = 0$ and $\overline{z} = y^*$.

4.2. *A-supermappings associated with the Howard-Mosco-Scarpini's method*

In this subsection, we concentrate on the discretized obstacle problem (1.2) and we consider the following problem:

$$\left\{\begin{array}{l} \text{Find } y \in E \text{ solution of} \\ \text{Max}\Big(\sum_{j=1}^{\alpha} A_{ij}.\, y_j - b_i, y_i - \psi_i\Big) = 0, \text{ for all } i \in \{1,\ldots,\alpha\}, \end{array}\right. \tag{4.13}$$

where

$$A \text{ is a strictly diagonally dominant } M\text{-matrix.} \tag{4.14}$$

We note that $A = (A_{ij})$ results from a block decomposition, $b = (\ldots, b_i, \ldots) \in E, \psi = \{\ldots, \psi_i, \ldots\} \in E$, and $y = (\ldots, y_i, \ldots) \in E$ are decomposed accordingly.

The complementarity problem (4.13) is written in the form of the following discrete Hamilton-Jacobi-Bellman problem where the operator $\mathcal{A}$ is elliptic:

$$\left\{\begin{array}{l} (\mathcal{A}u - f)(u - \psi) = 0, \text{ everywhere in } \Omega, \\ \mathcal{A}u - f \leq 0 \text{ and } u \leq \psi \text{ everywhere in } \Omega, \\ u_{|\partial\Omega} = 0. \end{array}\right.$$

In the case where $\mathcal{A}$ is the convection-diffusion operator, the assumption (4.14) is well satisfied if the first derivative is discretized via appropriate backward or forward decentered schemes and the second derivative is discretized by the classical five points scheme.

Let D_i be the diagonal matrix whose diagonal entries are equal to the corresponding diagonal entries of matrix A_{ii} for all $i \in \{1,\ldots,\alpha\}$. Then at the solution, the problem (4.13) is equivalent to

$$\left\{\begin{array}{l} \text{Find } y \in E \text{ solution of} \\ \text{Max}\Big(\sum_{j=1}^{\alpha} A_{ij}.\, y_j - b_i, D_i y_i - D_i\psi_i\Big) = 0, \text{ for all } i \in \{1,\ldots,\alpha\}. \end{array}\right. \tag{4.15}$$

For all w element of E, we denote by $\mathcal{A}(w) = (\ldots, \mathcal{A}_i(w), \ldots)$ the mapping from E onto itself such that

$$\mathcal{A}_i(w) = \text{Max}\Big(\sum_{j=1}^{\alpha} A_{ij} w_j - b_i, D_i w_i - D_i\psi_i\Big) \in E_i, \text{ for all } i \in \{1,\ldots,\alpha\}.$$

It follows from (4.14) that (4.15) is a problem of type (2.1) which satisfies the assumptions of Section 2. We associate to problem (4.15) a fixed point mapping F which is defined implicitly as follows: for all $w \in E$, consider the discrete Hamilton-Jacobi-Bellman subproblems:

$$\left\{\begin{array}{l} \text{Find } \hat{z} = (\hat{z}_1,\ldots,\hat{z}_i,\ldots,\hat{z}_\alpha) \in E \text{ such that} \\ \text{Max}\Big(A_{ii}\hat{z}_i - (b_i - \sum_{j\neq i} A_{ij} w_j), D_i(\hat{z}_i - \psi_i)\Big) = 0, \forall i \in \{1,\ldots,\alpha\}, \end{array}\right. \tag{4.16}$$

$$\hat{z}_i = F_i(w), \quad \text{for all } i \in \{1, \ldots, \alpha\}. \tag{4.17}$$

We consider the vector $w \in E$ and $i \in \{1, \ldots, \alpha\}$ such that

$$\mathcal{A}_i(w) \geq 0. \tag{4.18}$$

In the sequel we will use the following notation:

$$z_i^0 = w_i \quad \text{and} \quad x_i^0 = \hat{z}_i = F_i(w). \tag{4.19}$$

For all $z_i \in \langle x_i^0, z_i^0 \rangle_i$, we recall that we have:

$$\mathcal{A}_i(z_i\,;\,w) = \ \mathrm{Max}\,\Big(A_{ii}z_i - (b_i - \sum_{j=1, j\neq i}^{j=\alpha} A_{ij}w_j), D_i(z_i - \psi_i)\Big), \tag{4.20}$$

In order to present the generation process of the Howard-Mosco-Scarpini method which corresponds to a linearization of the considered problem, we introduce the set $L_i(z)$ of indices of the rows ℓ for which the second argument is dominant in (4.20); so $\ell \in L_i(z)$ if and only if

$$\mathrm{Max}\,\Big[A_{ii}z_i - (b_i - \sum_{j\neq i} A_{ij}w_j), D_i(z_i - \psi_i)\Big]_\ell = a_{\ell\ell}([z_i]_\ell - [\psi_i]_\ell), \tag{4.21}$$

where $[z_i]_\ell$ and $[\psi_i]_\ell$ denote the ℓ-th component of the subvectors z_i and ψ_i, respectively and $a_{\ell\ell}$ is the ℓ-th entry of the diagonal matrix D_i. We also introduce a matrix $C_i(z_i)$ for all $z_i \in \langle x_i^0, z_i^0 \rangle_i$. If $\ell \in L_i(z)$, then the entries of the ℓ-th row of the matrix $C_i(z_i)$ are equal to the entries of the ℓ-th row of the matrix D_i (i.e. $[C_i(z_i)]_\ell = (0, ..., 0, a_{\ell\ell}, 0, ..., 0)$, $a_{\ell\ell}$ being in the ℓ-th position), else they are equal to the entries of the ℓ-th row of the matrix A_{ii}. It follows from (4.14) and Theorem 3.12 in [VAR62], that $C_i(z_i)$ is an M-matrix. Thus, it follows from (4.20) that $\mathcal{A}_i(z_i\,;\,w)$ is a continuous surjective M-function. We note also that assumption (4.1) is satisfied on $\langle x_i^0, z_i^0 \rangle_i$, indeed, the right inequality follows from (4.18) and the left inequality follows from (4.16) and (4.19). Note that with the special choice of x^0, we have : $\mathcal{A}_i(x^0) = 0$. Let $C_i : \langle x_i^0, z_i^0 \rangle_i \to \mathcal{L}(R^{n_i})$, by using the definition of the matrix $C_i(z_i)$ and majorizing the difference $\mathcal{A}_i(z_i\,;\,w) - \mathcal{A}_i(x_i\,;\,w)$ for any x_i, z_i such that $x_i^0 \leq x_i \leq z_i \leq z_i^0$, we find out that assumption (4.2) is satisfied.

Consider now the case where

$$z_i^0 = w_i \leq \psi_i. \tag{4.22}$$

Lemma 4.1: Let the assumptions (4.14), (4.18), (4.22) and the notations (4.19), (4.20) hold. Then for all $i \in \{1, ..., \alpha\}$ and $x_i, z_i \in \langle x_i^0, z_i^0 \rangle_i$ such that $x_i \leq z_i$, $\mathcal{A}_i(x_i; w) \geq 0$ and $\mathcal{A}_i(z_i, w) \geq 0$, we have:

$$C_i(x_i) \leq C_i(z_i)\,.$$

Proof: It follows from (4.21) and (4.22) that for all $\ell \in L_i(x_i)$ we have:

$$[\mathcal{A}_i(x_i; w)]_\ell = a_{\ell\ell}([x_i]_\ell - [\psi_i]_\ell) \leq 0.$$

We conclude that

$$[\mathcal{A}_i(x_i; w)]_\ell = a_{\ell\ell}([x_i]_\ell - [\psi_i]_\ell) = 0,$$

since $\mathcal{A}_i(x_i; w) \geq 0$. Thus, we have

$$[x_i]_\ell = [\psi_i]_\ell, \forall \ell \in L_i(x_i).$$

Analogously, we can show that $[z_i]_\ell = [\psi_i]_\ell$, for all $\ell \in L_i(z_i)$. Hence, it follows from $x_i \leq z_i \leq \psi_i$ that $L_i(x) \subset L_i(z)$. Thus, it follows from the definition of $C_i(x_i)$ and $C_i(z_i)$ that $C_i(x_i) \leq C_i(z_i)$.

Remark 4.2: The result of Lemma 4.1 is close to (4.3); the only additional restriction being that $\mathcal{A}_i(x_i; w) \geq 0$, $\mathcal{A}_i(z_i, w) \geq 0$. We will see in the sequel that this restriction has no influence on the mappings studied in this subsection.

We consider the approximation of the problem $\mathcal{A}_i(z_i; w) = 0$, by the algorithm (4.6) and define the mapping $F_i^{\mathcal{A}}(w) = z_i^k$, where k is a given natural number and z_i^k is the k-th step associated with the algorithm (4.6). Similarly, we define $F_i^{\mathcal{B}}(w) = z_i^1$, which corresponds to the first step of the algorithm (4.6). The domain of definition D of the mappings $F^{\mathcal{A}}$ and $F^{\mathcal{B}}$ is given by:

$$D = \{w \in E \mid \mathcal{A}(w) \geq 0\}. \tag{4.23}$$

Proposition 4.2: Let assumptions (4.14), (4.18), (4.19), and (4.22) hold. Then $F^{\mathcal{A}}$ is an $\mathcal{M}$-continuous $\mathcal{A}$-supermapping associated with F on the domain D.

Proof: For all $z_i \in \langle x_i^0, z_i^0 \rangle_i$, we have $C_i^{-1}(z) \geq 0$ since $C_i(z)$ is an M-matrix. Hence, it follows from the above study and the fact that the algorithm (4.6) preserves the inequality $\mathcal{A}_i(z_i^k; w) \geq 0$ for all k, that the assumptions of Proposition 4.1 are well satisfied. Thus, it follows from the proof of Proposition 4.1 that $F^{\mathcal{A}}$ and $F^{\mathcal{B}}$ are $\mathcal{A}$-supermappings associated with F such that $F^{\mathcal{A}} \ \alpha \ F^{\mathcal{B}}$. We show now that $w^p \downarrow y^*, \ p \to \infty$ and $\mathcal{A}_i(w^p) \geq 0$, imply $F_i^{\mathcal{B}}(w^p) \downarrow F_i^{\mathcal{B}}(y^*), \ p \to \infty$. We first note that $z_i^1 = F_i^{\mathcal{B}}(w)$ can be equivalently defined by:

$$z_i^1 = w_i - C_i^{-1}(w_i).\mathcal{A}_i(w), \tag{4.24}$$

or by

$$z_i^1 = C_i^{-1}(w_i)\tilde{b}_i(w), \tag{4.25}$$

where $\tilde{b}_i(w)$ is the vector of E_i with components $[\tilde{b}_i]_\ell$ defined as follows:

$$[\tilde{b}_i]_\ell = a_{\ell\ell}[\psi_i]_\ell, \text{ if } \ell \in L_i(w),$$

$$[\tilde{b}_i]_\ell = [b_i]_\ell - [\sum_{j\neq i} A_{ij}w_j]_\ell, \text{ if } \ell \notin L_i(w).$$

As a matter of fact, (4.24) is equivalent to $C_i(w_i).(z_i^1 - w_i) + \mathcal{A}_i(w) = 0$, which implies:

$$\sum_{m=1}^{n_i} c_{\ell m}([z_i^1]_m - [w_i]_m) + \text{ Max } \Big(\sum_{r=1}^{n} a_{\ell r}[w_i]_r - [b_i]_\ell, a_{\ell\ell}([w_i]_\ell - [\psi_i]_\ell)\Big) = 0, \forall \ell; \tag{4.26}$$

where $c_{\ell m}, m = 1, ..., n_i$ are the entries of the ℓ-th row of the matrix $C_i(w_i)$, $[z_i^1]_m$ and $[w_i]_m$ denote the m-th component of the subvectors z_i^1, w_i, respectively, and $a_{\ell,r}$ are the entries of the ℓ-th row of the matrix A.

• if $\ell \notin L_i(w)$, (4.26) can be rewritten as follows:

$$\sum_{m=1}^{n_i} a_{\ell m}([z_i^1]_m - [w_i]_m) + \sum_{r=1}^{n} a_{\ell r}.[w_i]_r - [b_i]_\ell = 0; \tag{4.27}$$

the equality (4.27) can also be written as follows, by considering a block decomposition

$$\sum_{m=1}^{n_i} a_{\ell m}[z_i^1]_m = [b_i]_\ell - \Big[\sum_{j\neq i} A_{ij}.w_j\Big]_\ell,$$

• if $\ell \in L_i(w)$, (4.26) can be written as

$$a_{\ell\ell}([z_i^1]_\ell - [w_i]_\ell) + a_{\ell\ell}([w_i]_\ell - [\psi_i]_\ell) = 0.$$

Let an index i and a sequence $\{w^p\}_{p\in N}$ be such that

$$\begin{cases} w^p \downarrow y^*, \ p \to \infty, \\ w^p \leq \psi, \forall\, p, \\ A_i(w^p) = \text{ Max } \Big(\sum_j A_{ij}.w_j^p - b_i, D_i w_i^p - D_i\psi_i\Big) \geq 0. \end{cases} \tag{4.28}$$

where y^* is the solution of the problem. By the continuity of the M-function $\mathcal{A}$, we have:

$$\text{Max } \Big(\sum_j A_{ij}y_j^* - b_i, D_i(y_i^* - \psi_i)\Big) \geq 0.$$

Moreover it follows from (4.23) that for a sufficiently large p we have:

$$L_i(w^p) = L_i(y^*). \tag{4.29}$$

Then, by the equivalence of (4.24) and (4.25), $z_i^1(p) = F_i^{\mathcal{B}}(w^p)$ is given by

$$z_i^1(p) = C_i^{-1}(w_i^p).\tilde{b}_i(w^p).$$

It follows from (4.29) that for a sufficiently large p we have:

$$z_i^1(p) = C_i^{-1}(w_i^p).\tilde{b}_i(w^p) = C_i^{-1}(y_i^*).\tilde{b}_i(w^p).$$

Hence, we have:

$$\lim_{p\to\infty} F_i^{\mathcal{B}}(w^p) = \lim_{p\to\infty} z_i^1(p) = \lim_{p\to\infty} C_i^{-1}(y_i^*).\tilde{b}_i(w^p) = C_i^{-1}(y_i^*).\tilde{b}_i(y^*) = F_i^{\mathcal{B}}(y^*).$$

Remark 4.3: The reader is referred to [LIO80] for the solution of the Hamilton-Jacobi-Bellman problem via several sequential algorithms. The first two algorithms proposed in [LIO80] can lead to $\mathcal{M}$-continuous $\mathcal{A}$-supermappings when used for the solution of the subproblems presented above. Note also that the methods presented in [LIO80] are not related to relaxation or subdomain methods. Reference is also made to [MIE98] for a study of asynchronous Howard-Mosco-Scarpini's methods with flexible communication applied to the Hamilton-Jacobi-Bellman problem. The discretization matrix being irreducible in this case, the convergence analysis is then easier. Similar results can be obtained for boundary value problems such as non linear diffusion problems solved via Newton like methods (see [SPI95]).

4.3. *Combination of $\mathcal{A}$-supermappings with asynchronous iterations with flexible communication*

In this subsection we combine asynchronous iterations with flexible communication of Section 3 with the method presented in the previous subsection. So we shall consider asynchronous Howard-Mosco-Scarpini iterations with flexible communication.

Proposition 4.3: Consider the solution of problem (4.13). Then, asynchronous Howard-Mosco-Scarpini iterations with flexible communication starting from an $\mathcal{A}$-supersolution defined in (4.18) converge to the solution y^* of the problem.

Proof: The result follows from Proposition 3.2 combined with Proposition 4.2.

Remark 4.5: The result obtained in Proposition 4.3 also holds in the particular case of classical asynchronous iterations studied in [BAU78], [CHA69], and [MIE75a] whereby processors communicate only at the end of each updating phase.

Remark 4.6: We note that the solution of problem (4.13) via the combination of classical asynchronous iterations with approximate Howard-Mosco-Scarpini method may also lead to the use of the current value of the components of the iteration vector. As a matter of fact, the communication of updates at the end of each single step of algorithm (4.6) permits the processors to use the current value of the components; in this case order interval are omitted; however this scheme of computation which needs necessarily a communication after each single step of computation is less flexible and general than the block scheme presented in (3.1) to (3.3).

Remark 4.7: We note that we may have $\rho'(k) < k$ in (4.6) if $C(x)^{-1}$ is not updated at each iteration. In practice, one considers Gauss or Cholesky decomposition of $C(x)$; this decomposition is not necessarily achieved at each iteration.

Remark 4.8: It can be interesting to replace the Gauss decomposition quoted in the previous remark by the use of an iterative algorithm associated with a regular splitting of $C(x)$ which can be accelerated by a monotone aggregation method (see [EL 84] and [MIE80]).

5. Computational experiences

In this Section, we present and analyze computational results for parallel asynchronous and synchronous Howard-Mosco-Scarpini methods applied to the obstacle problem (1.1). The computational experience was carried out on a shared memory machine IBM 3090 with six vector processors.

5.1. *Implementation*

We point out that the routines make use of the ESSL library of IBM. The solution of linear systems is obtained via a band Gauss algorithm which preserves partial ordering. The linearizing matrix being an M-matrix, all it's principal submatrices are also M-matrices and all the principal minors are non zero. Then, it is not necessary to perform row and column permutations since each pivot $a_{k,k}$ considered at the k-th step is non zero. We note also that vectorisation is performed automatically.

In the synchronous parallel implementation, subdomains are updated according to the Gauss-Seidel iterative scheme. One or several adjacent subdomains are assigned to each processor. The maximum number of subdomains updated in parallel is $\alpha/2$. Each processor implements the same process which updates the assigned subdomains in cyclic order. The computational scheme is equivalent to an odd-even ordering of the subdomains. After an initial phase, all processors update odd subdomains in parallel,

then they update even subdomains. The parallelization uses macrotasking facilities of parallel Fortran. The synchronization of parallel tasks uses events technics. The implementation of parallel routines is done by running explicitly the routines as distinct parallel tasks; in this case, instructions permit one to create, activate, and suppress parallel tasks. When a task finishes a computation it uses a PEPOST instruction in order to inform other tasks; on the other hand the tasks which are waiting for this event to come use a PEWAIT instruction.

In the asynchronous implementation with flexible communication, two adjacent subdomains are assigned to each processor and all processors update in cyclic order the assigned subdomains. This scheme of computation is very similar to the odd-even scheme used for synchronous parallel algorithms. However, the computation processes are not synchronized and each processor can read boundary values when needed. This feature guarantees that flexible communication is actually implemented. The implementation of flexible communication is performed via data exchange after one step of the linearized method; this corresponds to an implicit definition of the A-supermapping F^A defined in Section 3. Moreover during the backward substitution, the updated values of the components of a subdomain are stored in the shared memory one by one. Each processor reads the available data just before a new computation. This implies that we can have different delays in the access to the components of the same block of the iterate vector during the computation of the components associated with a given subdomain. This procedure corresponds actually to the concept of flexible communication developed in this study which is different from the concept studied in [MIE75a] where the delays are similar for all components of the same block of the iteration vector.

5.2. *Computational results*

Experimental results for parallel algorithms are given in Table 1 for a problem with 25000 discretization points. We note that speedups and efficencies reported on Table 1 are computed according to an equivalent decomposition in the sequential case. Indeed, the sequential time and average rate of convergence vary in function of the subdomain decomposition. We report that a limitation of the synchronous parallel algorithms designed for the application studied in this paper, is that one can not implement all possible decomposition strategies. In particular, one can not implement a synchronous method with α subdomains and α processors. An alternative solution would be to implement a different iterative scheme, such as the Jacobi method; however the convergence rate would be slower. In the synchronous case, the performance is better when the number of subdomains is even. This is mainly due to the fact that for an odd number of subdomains, some processors are idle since they are waiting for the end of the computation of the others processors (this case is not displayed in Table 1).

We point out that asynchronous algorithms with flexible communication are faster than synchronous algorithms. For a medium granularity, the number of relaxations

of synchronous and asynchronous parallel algorithms is close to the number of relaxations of sequential algorithms. For a low granularity, the number of relaxations of synchronous and asynchronous algorithms is greater than the number of relaxations of sequential algorithms. Moreover we note that the efficiency of synchronous algorithms decreases faster than the efficiency of asynchronous algorithms when the granularity decreases.

processors	subdomains	mode	relaxations	time	speedup	efficiency
2	4	sync.	112	184.75	1.86	0.93
2	4	async.	109	171.46	2.0	1.0
3	6	sync.	192	118.85	2.48	0.83
3	6	async.	186	99.99	2.95	0.98
4	8	sync.	296	51.94	3.43	0.86
4	8	async.	288	46.65	3.82	0.96
5	10	sync.	440	41.89	4.06	0.81
5	10	async.	432	36.87	4.62	0.92
6	12	sync.	612	37.97	4.41	0.73
6	12	async.	623	31.16	5.38	0.90

Table 1 : experimental results for the obstacle problem

Acknowledgements

The authors wish to thank Jean-Marie Perraudin for his technical assistance. Part of this study has been made possible by a support of CINES and IBM France, Montpellier. We thank especially Georges Urbach, Michèle Battle, and Jean-Loic Delhaye.

6. References

[BAR82] R.H. Barlow, D.J. Evans, *Synchronous and asynchronous iterative parallel algorithms for linear systems*, Comput. J., vol. 25, pp. 56-60, 1982.

[BAU78] G. M. Baudet, *Asynchronous iterative methods for multiprocessors*, Journal Assoc. Comput. Mach., vol. 25, pp. 226–244, 1978.

[BEN97] M. Benzi, D. B. Szyld, *Existence and uniqueness of splittings for stationary iterative methods with application to alternating methods*, Numerische Mathematik, vol. 76, pp. 309–321, 1997.

[BER89] D. P. Bertsekas, J. N. Tsitsiklis, *Parallel and Distributed Computation, Numerical Methods*. Prentice Hall Englewood Cliffs N.J., 1989.

[BOU88] M. Boulbrachêne, P. Cortey-Dumont, J.C. Miellou, *Mixing finite elements and finite differences in a subdomain method*, Proceedings of the first International Symposium

on Domain Decomposition Methods for partial differential equations, SIAM, pp. 198-216, 1988.

[CHA89] T. F. Chan, R. Glowinski, J. Perriaux, O.B. Widlund, *Domain decomposition methods*, Proceedings of the second International Symposium on Domain Decomposition Methods for partial differential equations, SIAM, 1989.

[CHA69] D. Chazan, W. Miranker, *Chaotic relaxation*, Linear Algebra Appl., vol. 2, pp. 199–222, 1969.

[DIN80] Q.V. Dinh, R. Glowinski, J. Perriaux, *Application of domain decomposition techniques to the numerical solution of Navier-Stokes equations*, GAMNI, 1980.

[DIN81] Q.V. Dinh, R. Glowinski, B. Mantel, J. Perriaux, P. Perrier, *Sub-domain solutions of nonlinear problems in fluid dynamics on parallel processors*, Colloque international sur les méthodes de calcul scientifique et technique, Versailles, 1981.

[EL 90] D. El Baz, *M-functions and parallel asynchronous algorithms*, SIAM Journal on Numerical Analysis, vol. 27, no. 1, pp. 136–140, 1990.

[EL 96] D. El Baz, P. Spiteri, J.C. Miellou, D. Gazen, *Asynchronous iterative algorithms with flexible communication for nonlinear network flow problems*, Journal of Parallel and Distributed Computing, vol. 38, pp. 1–15, 1996.

[EL 84] M. N. El Tarazi, *Algorithmes mixtes asynchrones. Etude de convergence monotone*, Numer. Math. vol. 44, pp. 363-369, 1984.

[EVA86] D.J. Evans, K. Lishan, S. Jianping, C. Yupin, *The convergence rate of the Schwarz alternating procedure (II) for two dimensional problems*, Int. Jour. Comp. Math. 20, 325–339, 1986.

[EVA91] D.J. Evans, W. Deren, *An asynchronous parallel algorithm for solving a class of nonlinear simultaneous equation*, Parallel Computing vol. 17, pp. 165–180, 1991.

[FRO91] A. Frommer, *On asynchronous iteration in partially ordered spaces*, Numer. Funct. Anal. and Optimiz., vol. 12, pp. 315–325, 1991.

[FRO98] A. Frommer, D. B. Szyld *Asynchronous iterations with flexible communication for linear systems*, Calculateurs Parallèles, Réseaux et Systèmes Répartis, vol. 10, pp. 421–429, 1998.

[FRO00] A. Frommer, D. B. Szyld *On asynchronous iterations*, Journal of computational and Applied Mathematics, to appear.

[GIR91a] L. Giraud, J.C. Miellou, P. Spiteri, *Implementation of domain decomposition methods on shared memory multiprocessors*, High performance computing II. M. Durand and F. El Dabaghi ed., Elsevier, pp. 357–367, 1991.

[GIR91b] L. Giraud, P. Spiteri, *Résolution parallèle de problèmes aux limites non linéaires*, M.2 A.N., vol. 25, pp. 579–606, 1991.

[GIR92] L. Giraud, P. Spiteri, *Implementations of parallel solutions for nonlinear boundary value problems*, in Advances in Parallel Computing, vol. 4, D.J. Evans, G.R. Joubert, and H. Liddel ed., Parallel Computing'91, pp. 203-211. Elsevier Science B.V., North Holland, Amsterdam, 1992.

[HOF93] K.H. Hoffman, J. Zou, *Parallel efficiency of domain decomposition methods*, Parallel Computing, vol. 19, pp. 1375-1391, 1993.

[LIO80] P. L. Lions, B. Mercier, *Approximation numérique des équations de Hamilton-Jacobi-Bellman*, R.A.I.R.O. Analyse numérique, vol. 14, pp. 369-393, 1980.

[LIO89] P.L. Lions, *On the Schwarz alternating method II: stochastic interpretation and order properties*, Domain decomposition methods, T. F. Chan et al. ed., SIAM, pp. 47-70, 1989.

[MIE74] J.C. Miellou, *Itérations chaotiques à retards*, CRAS, Paris vol. 278, pp. 957-960, 1974.

[MIE75a] J. C. Miellou, *Algorithmes de relaxation chaotique à retards*, RAIRO, vol. R1, pp. 55–82, 1975.

[MIE75b] J. C. Miellou, *Itérations chaotiques à retards, étude de la convergence dans le cas d'espaces partiellement ordonnés*, C.R.A.S. Paris, vol. 280, pp. 233–236, 1975.

[MIE80] J.C. Miellou, *Extrapolation-aggregation algorithms of monotone kinds; application to one obstacle's stationary problems*, Proceedings of intensive seminary on free boundary value problems. Magenes ed., pp. 411-437, 1980.

[MIE82] J.C. Miellou, *Variantes synchrones et asynchrones de la méthode alternée de Schwarz*, Rapport ERA n° 070.654, University of Besançon, 1982.

[MIE85a] J. C. Miellou, P. Spiteri, *Un critère de convergence pour des méthodes générales de point fixe*, M2AN, vol. 19, pp. 645–669, 1985.

[MIE85b] J. C. Miellou, P. Spiteri, *Two criteria for the convergence of asynchronous iterations*, in Computers and computing, P. Chenin et al. ed., Wiley Masson, Paris, pp. 91–95, 1985.

[MIE86a] J. C. Miellou, *Asynchronous iterations and order intervals*, In Parallel Algorithms and Architectures, M. Cosnard et al. ed., North Holland, pp. 85-96, 1986.

[MIE86b] J.C. Miellou, G.R. Perrin, P. Spiteri, *An inexpensive method of performance evaluation for subdomain decomposition parallel algorithms of three dimensional elliptic problems*, Journal of systems and software vol. 6, pp. 169-173, 1986.

[MIE90] J.C. Miellou, Ph. Cortey-Dumont, M. Boulbrachêne, *Perturbation of fixed-point iterative methods*, Advance in Parallel Computing vol. 1, JAI Press Inc., pp. 81-122, 1990.

[MIE91] J.C. Miellou, L. Giraud, A. Laouar, P. Spiteri, *Subdomain decomposition methods with overlapping and asynchronous iterations*, Progress in partial differential equations, M. Chipot and J. Saint Jean Paulin ed., Pitman Research Notes in Mathematics Series, pp. 166-183, 1991.

[MIE98] J. C. Miellou, D. El Baz, P. Spiteri. *A new class of iterative algorithms with order intervals*, Mathematics of Computation, vol. 67 pp. 237–255, 1998.

[ORT70] J.M. Ortega, W.C. Rheinboldt, *Iterative solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[RHE70] W.C. Rheinboldt, *On M-functions and their application to nonlinear gauss-seidel iterations and to network flows*, J. Math. Anal. and Appl., vol. 32 pp. 274–307, 1970.

[ROB75] F. Robert, M. Charney, F. Musy, *Itérations chaotiques série-parallèle pour des équations non-linéaires de point fixe,* Aplikace Matematiky, vol. 20 pp. 1–37, 1975.

[SPI83] P. Spiteri, *Simulation d'exécutions parallèles pour la résolution d'inéquations variationnelles stationaires*, Revue E.D.F. Informatique et Mathématique Appliquées, Série C, vol. 1 pp. 149-158, 1983.

[SPI86] P. Spiteri, *Parallel asynchronous algorithms for solving boundary value problems*, Parallel Algorithms, M. Cosnard et al ed., North-Holland, pp. 73-84, 1986.

[SPI95] P. Spiteri, J. C. Miellou, D. El Baz, *Asynchronous alternating Schwarz method for the solution of nonlinear partial differential equations*, Rapport IRIT/LCS/LAAS 95309, 1995.

[SZY98] D. B. Szyld, *Different models of parallel asynchronous iterations with overlapping blocks*, Computational and Applied Mathematics, vol. 17 pp. 101–115, 1998.

[VAR62] R.S. Varga, *Matrix Iterative Analysis*, Prentice Hall 1962.